
**Faculdade de Tecnologia de Garça "Deputado Júlio Julinho
Marcondes de Moura"**

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

PEDRO DOS SANTOS ALENCAR JÚNIOR

APLICATIVO ANDROID PARA REALIZAÇÃO DE TESTES DE REDE.

**GARÇA
2017
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**Faculdade de Tecnologia de Garça "Deputado Júlio Julinho
Marcondes de Moura"**

PEDRO DOS SANTOS ALENCAR JÚNIOR

APLICATIVO ANDROID PARA REALIZAÇÃO DE TESTES DE REDE

Relatório técnico apresentado à Faculdade de Tecnologia de Garça – FATEC, como requisito para conclusão do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas.

Orientador: Prof Adriano Nakamura.

**GARÇA
2017**

**Faculdade de Tecnologia de Garça "Deputado Júlio Julinho
Marcondes de Moura"**

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

PEDRO DOS SANTOS ALENCAR JÚNIOR

APLICATIVO ANDROID PARA REALIZAÇÃO DE TESTES DE REDE

Trabalho de Conclusão de Curso
apresentado à Faculdade de
Tecnologia "Dep. Julio Julinho
Marcondes de Moura" – FATEC,
como requisito para conclusão do
Curso de Tecnologia em Análise e
Desenvolvimento de Sistemas,
examinado pela seguinte comissão
de professores:

Data da Aprovação: ____/____/____

Orientador: _____

**Profª Adriano Nakamura
FATEC Garça**

Membro Titular: _____

**Prof. Convidado 1
FATEC Garça**

Membro Titular: _____

**Prof. Convidado 2
FATEC Garça**

Alencar Júnior, Pedro dos Santos

Aplicativo android para realização de testes de rede. / Pedro dos Santos Alencar Júnior. - Garça, 2017.

26 f.

Relatório Técnico (Trabalho de Conclusão de Curso de Tecnologia em Análise e Desenvolvimento de Sistemas) – Faculdade de Tecnologia de Garça – FATEC, 2017.

Orientador: Adriano Nakamura

AGRADECIMENTOS

Chegando ao fim de mais essa etapa da vida, gostaria de agradecer a todos que compartilharam os momentos bons e difíceis superando a cada dia mais um novo desafio. Dessa forma gostaria de agradecer a todos os colegas de sala, os professores, minha família e a Deus. Gostaria de destacar o apoio do orientador Prof. Adriano Nakamura.

RESUMO

Aliando desenvolvimento *Android* e Redes, o presente projeto visa disponibilizar um sistema capaz de realizar testes de conexão pelo próprio usuário de forma prática e de acesso fácil através de um aparelho com o sistema Android utilizando o endereço do site/host ou número de IP de maneira simples em um dispositivo que utilize o sistema operacional *Android* possibilitando a sua utilização por pessoas comuns.

Palavras-chave: *Android*. Redes e conexões. Internet.

LISTA DE ILUSTRAÇÕES

FIGURA 1: DIAGRAMA DE CASO DE USO **ERRO! INDICADOR NÃO DEFINIDO.**2

FIGURA 2: FATIAS DE MERCADO DOS PRINCIPAIS SISTEMAS OPERACIONAIS PARA
DISPOSITIVOS MÓVEIS **ERRO! INDICADOR NÃO DEFINIDO.**3

FIGURA 3: COMPARAÇÃO DAS LINGUAGENS DE PROGRAMAÇÃO **ERRO! INDICADOR NÃO
DEFINIDO.**4

FIGURA 4: OS COMPONENTES DE UMA APLICAÇÃO ANDROID **ERRO! INDICADOR NÃO
DEFINIDO.**7

FIGURA 5: ARQUITETURA DO ANDROID **ERRO! INDICADOR NÃO DEFINIDO.**9

FIGURA 6: INTERFACE DO APLICATIVO 21

FIGURA 7: MENUS ADICIONAIS 22

FIGURA 8: MENU MEU IP 23

FIGURA 9: LISTAR EQUIPAMENTOS CONECTADOS A REDE..... 23

Sumário

1 INTRODUÇÃO.....	9
1.1 Objetivos	10
1.2 Escopo e Delimitação do Trabalho	10
2 EMBASAMENTO TEÓRICO	11
2.1 O recursos técnico utilizados:	11
3 DESENVOLVIMENTO DO PROJETO	11
3.1 Diagrama de Caso de Uso	11
3.2 Diagrama de Classe	Erro! Indicador não definido.
3.3 Sistema Operacional	12
3.4 Linguagem de Programação e <i>Framework</i>	13
3.5 Android	13
3.6 A plataforma Android.....	14
3.7 Visão Geral do SDK	15
3.8 Estrutura das Aplicações Android	15
4 ESPECIFICAÇÃO DO PROJETO	19
4.1 <i>Interface</i>	19
4.2 Menus	20
4.5 Recurso Adicional	23
5 CONSIDERAÇÕES FINAIS	24
6 REFERÊNCIAS	25

1 INTRODUÇÃO

O uso do telefone celular para acessar a internet ultrapassou o computador é o que aponta o Suplemento de Tecnologias de Informação e Comunicação (TIC) da Pesquisa Nacional por Amostra de Domicílios (Pnad) pelo Instituto Brasileiro de Geografia e Estatística (IBGE). Dessa forma é o principal meio de acesso a internet sendo assim através dos dispositivos móveis deve ser capaz de auxiliar nos diagnósticos de rede e resolução de problemas.

Ranking completo, com todos os dados de vendas e presença de mercado dos sistemas:

Operating System	2Q16 Units	2Q16 Market Share (%)	2Q15 Units	2Q15 Market Share (%)
Android	296,912.8	86.2	271,647.0	82.2
iOS	44,395.0	12.9	48,085.5	14.6
Windows	1,971.0	0.6	8,198.2	2.5
Blackberry	400.4	0.1	1,153.2	0.3
Others	680.6	0.2	1,229.0	0.4
Total	344,359.7	100.0	330,312.9	100.0

FONTE(S)GARTNER:

<http://www.gartner.com/newsroom/id/3415117>

1.1 Objetivos

Como o celular tornou-se o principal dispositivo de acesso à Internet, esse projeto tem como objetivo o desenvolvimento de um aplicativo para realização de testes de conexão de forma prática e de acesso fácil através de um aparelho com o sistema Android utilizando o endereço do site/host ou número de IP.

1.2 Escopo e Delimitação do Trabalho

No universo do desenvolvimento para dispositivos móveis existem diferentes categorias de dispositivos como Smartphones, PDAs e Tablets.

Existe também uma variada disponibilidade de plataformas de desenvolvimento para estes dispositivos, dentre as quais podemos citar Android, iOS, Symbian, webOS entre outras.

O fato do escopo deste projeto focar no desenvolvimento de um protótipo para funcionar em um dispositivo específico (Smartphone) baseado em uma plataforma específica (Android) constitui-se em uma das restrições deste trabalho.

Outra restrição é relacionada ao fato de que o protótipo resultante deste trabalho não pretende ser um aplicativo de software completo, a pretensão é que este seja utilizado como um estudo de caso para projeto e melhorias subsequentes que serão descritas como trabalhos futuros.

2 EMBASAMENTO TEÓRICO

2.1 O recursos técnico utilizados:

Aplicação dos conhecimentos adquiridos durante as aulas, Androids.help Português Comunidade Android, QUERINO F^o, L. C. Desenvolvendo seu Primeiro Aplicativo Android. Novatec Editora, 2013.

3 DESENVOLVIMENTO DO PROJETO

O desenvolvimento do projeto baseou-se inicialmente em processos de UML para comunicar a ideia a ser projetada no *software*. Segundo Medeiros (2010, p. 10) “Utilizando a UML conseguimos pensar um software em um local e codifica-lo em outro”. Sendo assim, após a estruturação da ideia foi realizado o levantamento quanto ao sistema operacional que iria hospedar a aplicação, a linguagem de programação a ser utilizada, o *framework* para desenvolvimento.

3.1 Diagrama de Caso de Uso

O Diagrama de Casos de Uso é um modelo das funções pretendidas do sistema e de seu ambiente. Segundo Sommerville (2011) o diagrama de caso de uso deve representar uma tarefa discreta que envolva a interação externa com um sistema. Basicamente o diagrama deve ser composto por atores e setas que direcionam o fluxo de cada ação realizada.

O diagrama ilustrado na Figura 1, demonstra a possibilidade do usuário ao acessar o aplicativo executar o teste de ping de conexão, retorno do seu número de IP, listar equipamentos da rede e sair da aplicação.

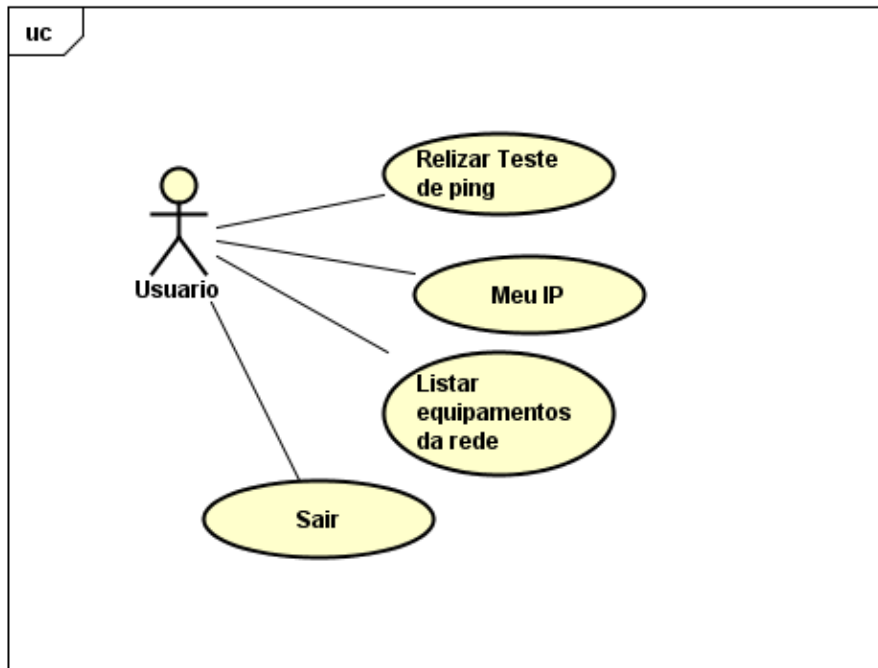
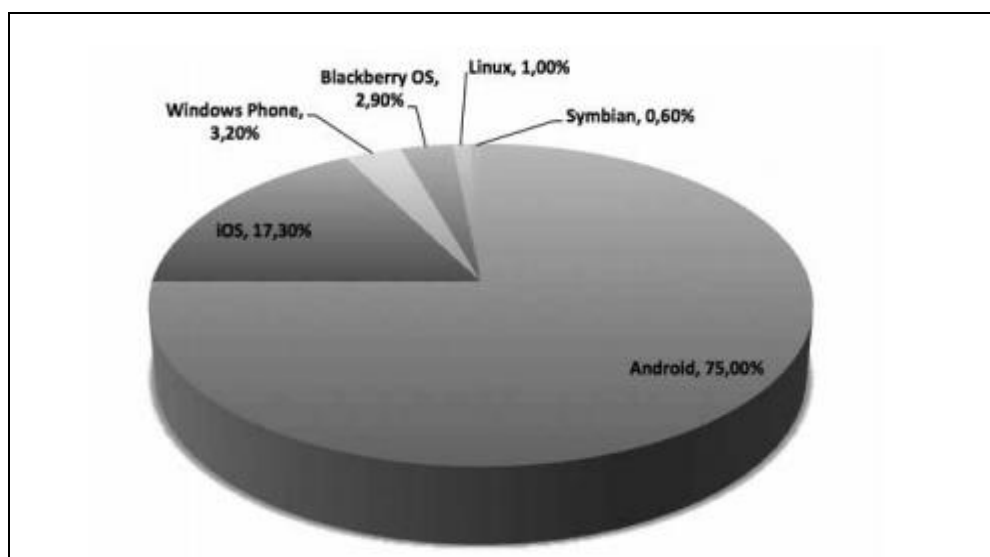


Figura 1 : Diagrama de caso de uso

3.3 Sistema Operacional

O aplicativo foi desenvolvido para ser hospedado no sistema operacional *Android*; tal escolha se deu pela fatia de usuários alcançados por este sistema *mobile*, como é possível verificar no gráfico ilustrado na Figura 2.

Figura 2: Fatias de mercado dos principais sistemas operacionais para dispositivos móveis

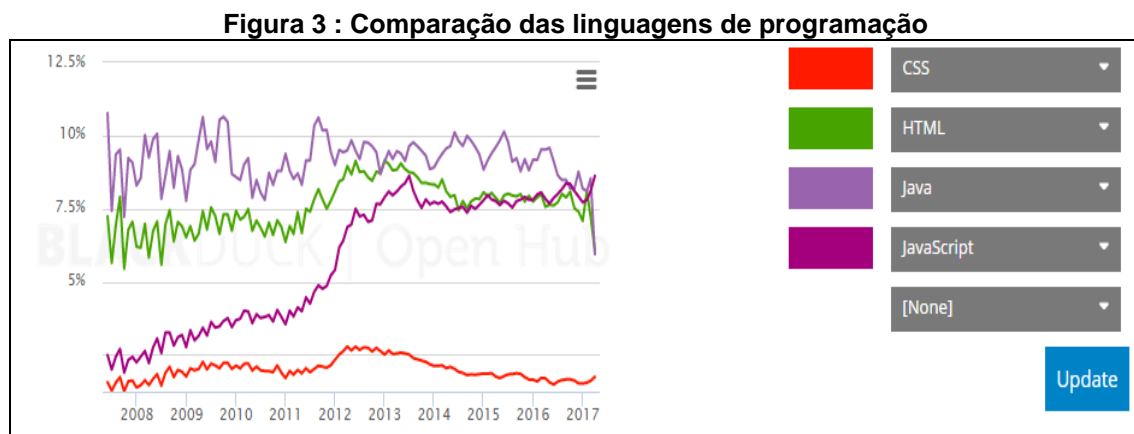


Fonte: QUERINO FILHO (2014)

3.4 Linguagem de Programação e *Framework*

Definido o sistema operacional foi feita a escolha da linguagem de desenvolvimento a programação Java acompanhada da IDE oficial para o desenvolvimento de aplicativos *Android* o *Android Studio*.

Conforme a consideração acima iniciou-se um estudo quanto à possibilidade de desenvolvimento nativo, onde, como demonstra a Figura 3, a linguagem Java manteve-se estável ao longo dos anos se comparada com as linguagens utilizadas em desenvolvimento híbrido, e por ser a linguagem oficial acompanhada do sistema de desenvolvimento do *software Android* permite o acesso a recursos de rede facilmente.



Fonte: BLACKDUCK (2016)

3.5 Android

O Android foi um projeto inicialmente desenvolvido por uma startup americana do Vale do Silício chamada Android Inc. Esta pequena empresa foi adquirida pelo Google no ano de 2005, que por sua vez tratou de amadurecer o projeto e o tornou público em meados de 2007 com o objetivo de apresentar a primeira plataforma open source de desenvolvimento para dispositivos móveis. Atualmente o Android é mantido por um grupo denominado Open Handset Alliance (OHA), que é formado por mais de 40 empresas das 20 quais figuram o próprio Google e outras de importância nos ramos de telefonia (Telefónica),

fabricação de semicondutores (Intel) e fabricação de celulares (Motorola), dentre outras.

3.6 A plataforma Android

O Android é uma plataforma para desenvolvimento de aplicativos voltados para funcionar em dispositivos móveis baseados em um núcleo de Linux, sendo que as aplicações a serem geradas são escritas em linguagem Java. Estas aplicações são compiladas em bytecodes Dalvik e executadas em uma máquina virtual desenvolvida especialmente para utilização em dispositivos móveis denominada Máquina Virtual Dalvik.

Disponibiliza um kit de desenvolvimento denominado Android SDK que proporciona as APIs e ferramentas necessárias para o desenvolvimento de aplicações, tendo como principais recursos:

- Application framework que proporciona a reutilização de componentes;
- Dalvik virtual machine que é otimizada para dispositivos móveis;
- Um browser integrado baseado no webkit engine;
- Gráficos otimizados através de utilização de bibliotecas 2D; e 3D baseada na especificação OpenGL ES 1.0 (aceleração de hardware opcional).
- SQLite para armazenamento de banco de dados estruturados;
- Suporte multimídia para áudio, vídeo e formatos de imagem (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF);
- Ambiente para desenvolvimento rico, apresentando emulador de dispositivo, ferramentas de depuração, memória, desempenho e um plugin para o Eclipse (ADT).

Os seguintes recursos também estão presentes, porém dependentes de hardware: telefonia GSM, Bluetooth, EDGE, 3G, WiFi, câmera, GPS, compasso e acelerômetro.

3.7 Visão Geral do SDK

O kit de desenvolvimento para Android (Android SDK) está disponível para Windows, Linux e MacOS, provendo ao desenvolvedor um conjunto rico de ferramentas que inclui um depurador, bibliotecas, emulador de smartphone, documentação, código de exemplo e tutoriais.

3.7.1 Android Runtime

O mecanismo de runtime do Android é baseado em dois grupos fundamentais que são as suas bibliotecas centrais e sua máquina virtual criada para que cada dispositivo móvel possa executar múltiplas VM.

Cada aplicação desenvolvida em Android executa em um único processo o qual é uma instância desta máquina virtual, executando arquivos no formato '.dex' (abreviação de Dalvik Executable), formato otimizado para carregamento rápido e com consumo mínimo de memória.

A VM Dalvik executa classes compiladas em linguagem Java, porém transformadas no formato '.dex', usando ainda o kernel do Linux para aumentar as suas funcionalidades como o uso de threads e gestão de memória de baixo nível.

3.8 Estrutura das Aplicações Android

Aplicações desenvolvidas em Android são baseadas em uma arquitetura de componentes chave demonstrada pela figura abaixo, porém, não necessariamente uma aplicação deve obrigatoriamente utilizar-se de todos estes componentes, no geral as aplicações são compostas por uma combinação destes.

Em conjunto com estes componentes existe um arquivo XML denominado AndroidManifest.xml de existência obrigatória, e no qual são feitas configurações gerais da aplicação e dos componentes utilizados por ela.

Juntam-se a esta estrutura dois outros itens importantes que fazem estes quatro componentes chave funcionarem que são as Intents e as Views.



Figura 4 – Os componentes de uma aplicação Android.

Activities formam a base para o desenvolvimento visual de uma aplicação, sendo o componente mais comum em uma aplicação. Uma activity é quem realiza os tratamentos dos eventos da tela e que define qual view será desenhada na tela [9]. Para uma aplicação que possua múltiplas telas, cada tela deverá ser representada por uma activity que é implementada como uma subclasse de Activity.

Services são códigos executados em segundo plano e que não apresentam uma interface visual. Cada service é executado na thread principal do processo que o criou, não causando bloqueio ou interferência em outros componentes. Cada serviço é uma classe que herda de Service.

Broadcast Receivers são componentes receptores de ocorrências de eventos do sistema e que reagem a estes eventos. Cada receiver é uma classe que herda de BroadcastReceiver.

Content Providers são componentes que tornam um conjunto específico de dados da aplicação disponível para outras aplicações. Cada provider é uma classe que herda de ContentProvider e disponibiliza um

conjunto padrão de métodos para que outras aplicações possam recuperar e armazenar dados do tipo que o provedor controla.

Intents são mensagens responsáveis por ativar os componentes Service, Activity e BroadcastReceiver de uma aplicação. Essas mensagens são utilizadas para facilitar a ligação entre os componentes da aplicação ou de aplicações diferentes, em tempo de execução.

Views são elementos utilizados para definir objetos gráficos exibidos na tela, com o objetivo de prover interação com o usuário. Exemplo destes elementos são botões, caixas de diálogo, mapas entre outros.

AndroidManifest.xml é o arquivo de manifesto escrito em XML, obrigatório e único para a aplicação. Nele são descritos os componentes que fazem parte da aplicação, definidos nomes para as activities, o modo de orientação da tela, bem como declaradas permissões para acesso a recursos como o GPS ou Internet. Este arquivo lista também as bibliotecas que a aplicação vai usar e qual activity principal irá iniciar a aplicação.

A classe **android.view.View** é a classe mãe de todos os componentes visuais e suas subclasses irão compor uma interface com o usuário no Android. Estas subclasses implementam o método `onDraw(Canvas)`, responsável por desenhar os componentes na tela. Estes componentes dividem-se em dois grupos, os widgets e os gerenciadores de layout. O primeiro é um componente simples que herda diretamente da classe View, como exemplos temos as classes Button, ImageView e TextView, já os gerenciadores de layout são subclasses de `android.view.ViewGroup`.

3.8.1 Visão Geral da Arquitetura

Android é uma plataforma que apresenta desde sistema operacional, até middleware e aplicativos, conforme pode ser observado na figura 5. Sua arquitetura é dividida em diversas camadas as quais são: núcleo do sistema

operacional, bibliotecas, runtime, framework e aplicativos.

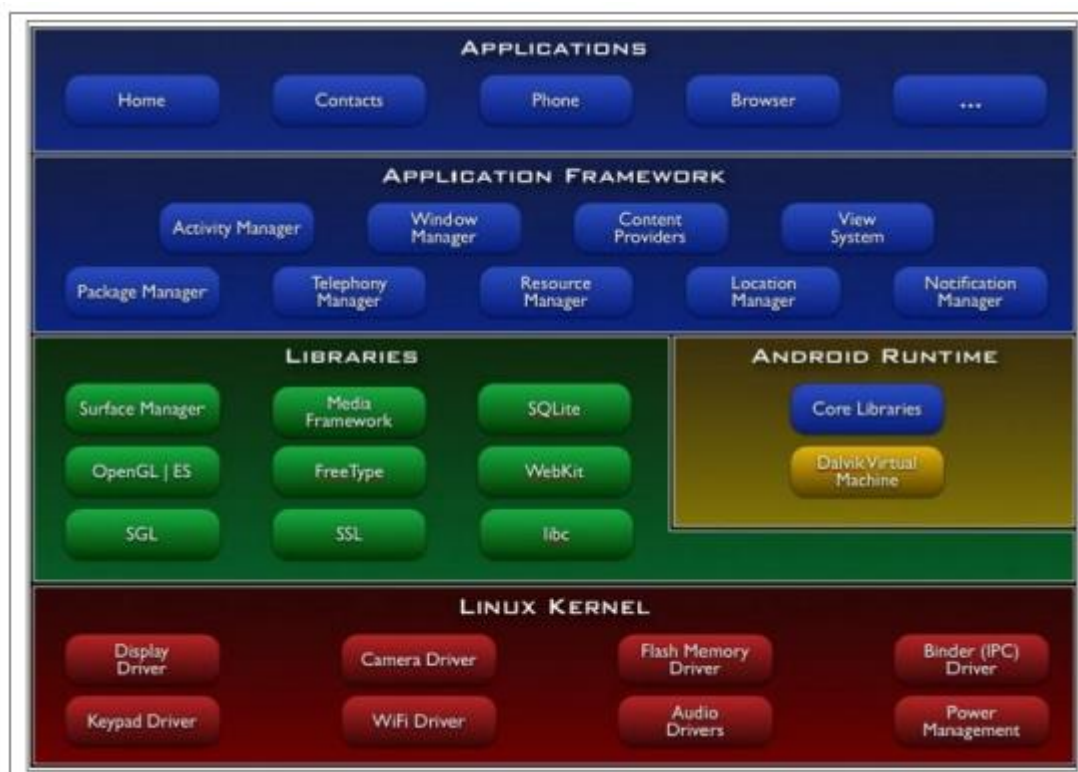


Figura 5 – Arquitetura do Android

Na camada do núcleo (Linux Kernel), baseada em Linux, localiza-se o sistema operacional da plataforma, responsável por serviços denominados de baixo nível como gerenciamento de processos, gerenciamento de memória, segurança, etc.

Na camada de bibliotecas (Libraries), ficam as APIs desenvolvidas em C/C++ e que dão suporte dentre outros recursos à renderização 3D (OpenGL ES), gerenciamento de base de dados (SQLite) e suporte aos diversos formatos de vídeo e áudio.

Na camada de runtime (Android Runtime), encontram-se componentes como as core libraries, que disponibilizam a API Java necessária para a escrita do código de programação das aplicações, bem como a Dalvik Virtual Machine, que é a máquina virtual que dará condições para que a aplicação Java desenvolvida possa ser executada.

Na camada de framework (Application Framework), estão localizadas as APIs que serão utilizadas pelas aplicações que executam sobre a plataforma do Android, como por exemplo, os gerenciadores de telefonia, localização e notificação.

Na camada restante, as de aplicativos (Applications), estarão representadas as aplicações que são executadas sobre a plataforma, sejam elas nativas como o caso da calculadora, do gerenciador de contatos, do calendário, etc., ou aplicações desenvolvidas por terceiros como é o caso do protótipo que será desenvolvido neste trabalho. Para a camada de aplicativos, não existe diferença entre aplicações nativas e aplicações de terceiros, todas são escritas com as mesmas APIs e executadas no mesmo runtime, inclusive tendo a possibilidade da troca de uma aplicação nativa por outra que tenha a mesma finalidade e seja desenvolvida por um terceiro ou pelo próprio usuário.

4 ESPECIFICAÇÃO DO PROJETO

Todos os aspectos vistos até o momento culminam no aplicativo nomeado como “*Android Ping Teste*” que será apresentado nos tópicos seguintes, de forma detalhada, através da especificação as funcionalidades existentes.

4.1 Interface

Ao iniciar o aplicativo, como ilustra a Figura 6, é disparado na *interface*.

Figura 6: Interface

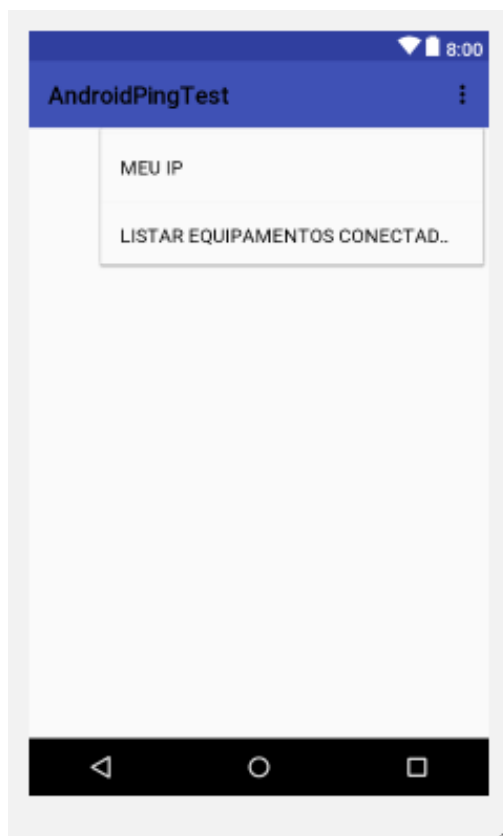


Fonte: do próprio autor

4.1 Interface

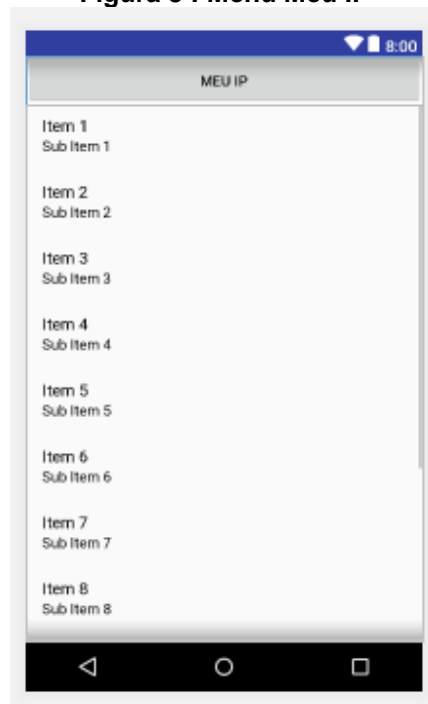
Na lateral superior direita existe o menu com os recursos adicionais MEU IP e LISTAR EQUIPAMENTOS CONECTADOS conforme demonstrado na Figura 7.

Figura 7 : Menus Adicionais



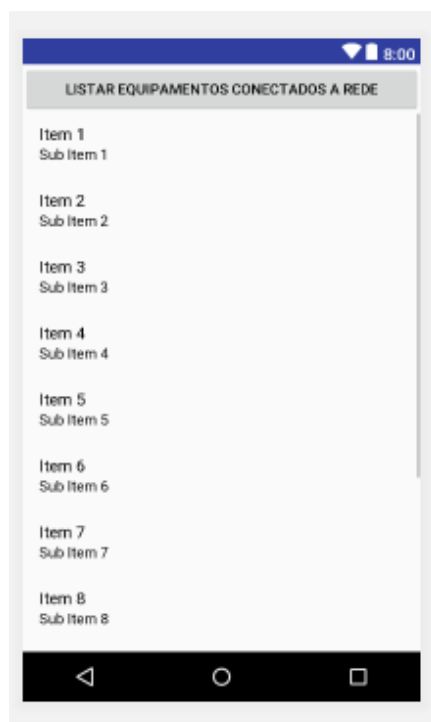
Fonte: do próprio autor

Ao clicar na opção MEU IP será chamado a nova tela com o resultado da consulta do ip do dispositivo conforme mostra a figura 8.

Figura 8 : Menu Meu IP

Fonte: do próprio autor

Ao clicar na opção LISTAR EQUIPAMENTOS CONECTADOS A REDE será chamado a nova tela com o resultado da consulta de ips conectado a mesma rede do dispositivo conforme mostra a figura 9.

Figura 9 : LISTAR EQUIPAMENTOS CONECTADOS A REDE

Fonte: do próprio autor

4.5 Recurso Adicional

Permissões necessárias para acesso a internet do dispositivo, precisa de um acesso especial.

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"></uses-permission>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"></uses-permission>
<uses-permission android:name="android.permission.INTERNET"></uses-permission>
```

Função principal para execução dos comandos.

```
public void ExecutarPing(View view) {
    Editable host = edtIP.getText();
    List<String> listaRespostaPing = new ArrayList<>();
    //Cria o adapter para a ListView
    ArrayAdapter<String> adapterLista = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, listaRespostaPing);

    try {
        String cmdPing = "ping -c 4 "+host;
        Runtime r = Runtime.getRuntime();
        Process p = r.exec(cmdPing);
        BufferedReader in = new BufferedReader( new InputStreamReader(p.getInputStream()));
        String inputLinhe;

        while((inputLinhe = in.readLine()) != null){
            listaRespostaPing.add(inputLinhe);
            //adiciona para cada linha
            listaPing.setAdapter(adapterLista);
        }

        Toast.makeText(this, "Comando executado com sucesso!", Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
        Toast.makeText(this, "Erro: "+e.getMessage().toString(), Toast.LENGTH_SHORT).show();
    }
}
```

5 CONSIDERAÇÕES FINAIS

O resultado deste projeto foi o desenvolvimento de um aplicativo para dispositivos móveis com sistema operacional *Android*, tendo como finalidade a para realização de testes de conexão de forma prática e de acesso fácil.

O aplicativo pode ser utilizado por usuários domésticos com pouco ou praticamente nenhum conhecimento mais avançado.

É importante enfatizar o caráter científico do projeto, onde a tecnologia é utilizada para otimização e obtenção de resultados em uma área que difere do meio tecnológico.

Por fim, pode-se afirmar que os objetivos propostos foram atingidos e os conhecimentos adquiridos no decorrer do curso foram utilizados em todo percurso que resultou na entrega final deste projeto.

6 REFERÊNCIAS

QUERINO FILHO, L. C. **Desenvolvendo seu primeiro aplicativo android.** São Paulo: Novatec, 2004.

GOOGLE. What is Android. Disponível em:
<http://developer.android.com/guide/basics/what-is-android.html>

TOSIN, Carlos. Conhecendo o Android. Disponível em:
<http://www.softblue.com.br/blog/home/postid/11/CONHECENDO+O+ANDROID>