

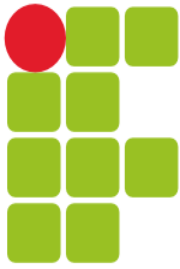
INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE
Campus Parnamirim

Introdução ao Android

André Gustavo Duarte de Almeida
docente.ifrn.edu.br/andrealmeida

Parte 1 – Conhecendo o Sistema e Primeiro Programa

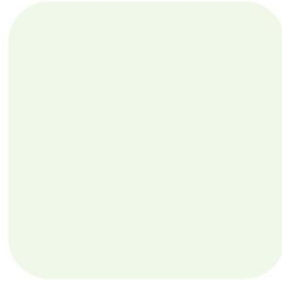




INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE
Campus Parnamirim

Roteiro

- Pré-requisitos
- Conceitos Básicos
- Configurando o Ambiente
- Primeiro programa

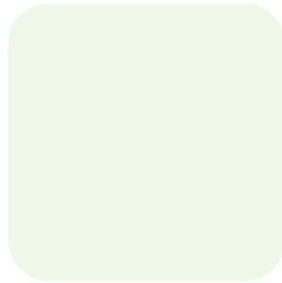


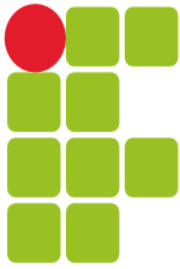


INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE
Campus Parnamirim

Pré-requisitos

- Conhecimentos
 - Noções de Java
 - Noções de Orientação a Objetos





Pré-requisitos

- Software
 - Java Development Kit
 - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
 - Ubuntu: `sudo apt-get install`
 - Se você já tem o Eclipse, instalar através do plugin ADT Tools
 - Senão baixar o ADT Bundle, que inclui todos os pré-requisitos(exceto) necessários para o desenvolvimento de aplicativos Android
 - Demonstraremos os dois processos

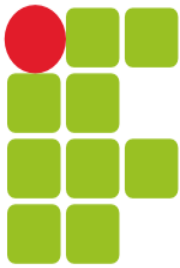


Conceitos Básicos

- O que é Android?!
- Sistema Operacional para plataformas móveis

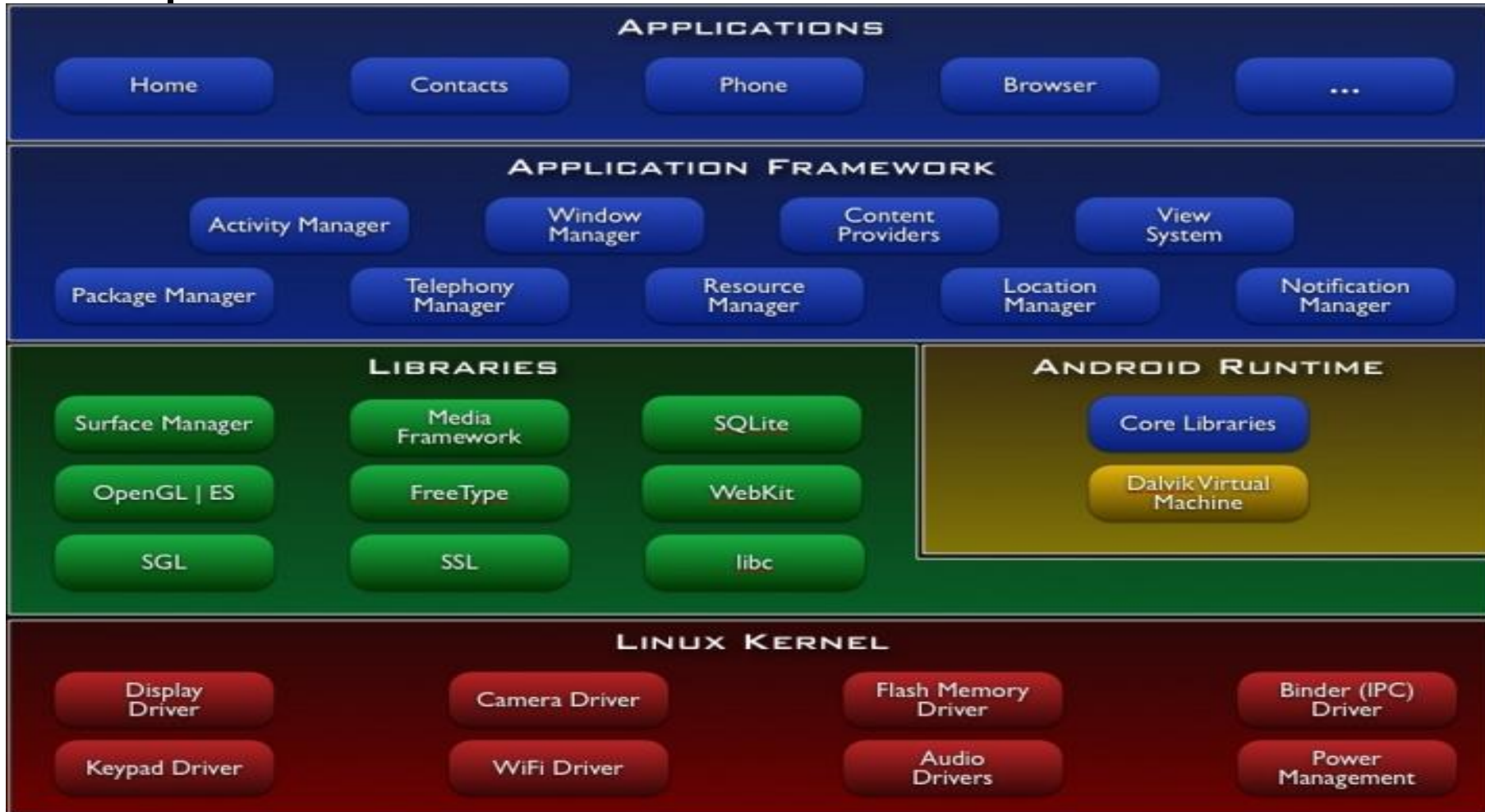
Considerado hoje a plataforma móvel mais utilizado no mundo, em especial pela sua capacidade de funcionar em diferentes dispositivos





Conceitos Básicos

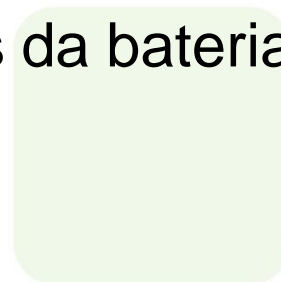
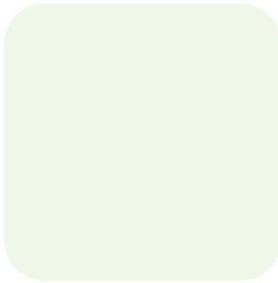
- Arquitetura

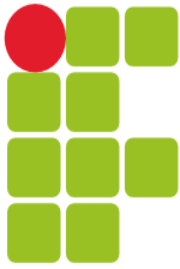




Conceitos Básicos

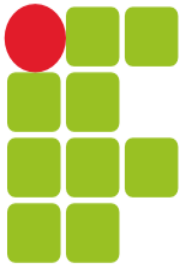
- Fundamentos
 - Aplicações escritas em Java
 - Funcionamento em sistema *SandBox*
- Componentes da Aplicação
 - Atividades
 - Serviços
 - Provedores de Conteúdo
 - Receptores de Mensagens BroadCast
 - Ex: Notificação do status da bateria





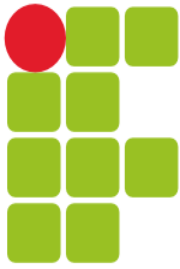
Conceitos Básicos

- Ativação de Componentes
 - Uso do Intent
 - Permite que um componente estabeleça uma relação com outro componentes para uso futuro
 - Ex: Permite uma atividade(tela), chamar uma segunda tela(atividade) baseado em algum evento realizado pelo usuário



Conceitos Básicos

- Manifest file
 - Arquivo utilizado em diversas aplicações Java, para seu empacotamento
 - Declara os componentes da aplicação
 - Define as permissões requeridas pelo aplicativo
 - Define a API mínima a ser requerida
 - Define os recursos de hardware e software necessários
 - Bibliotecas necessárias e etc....



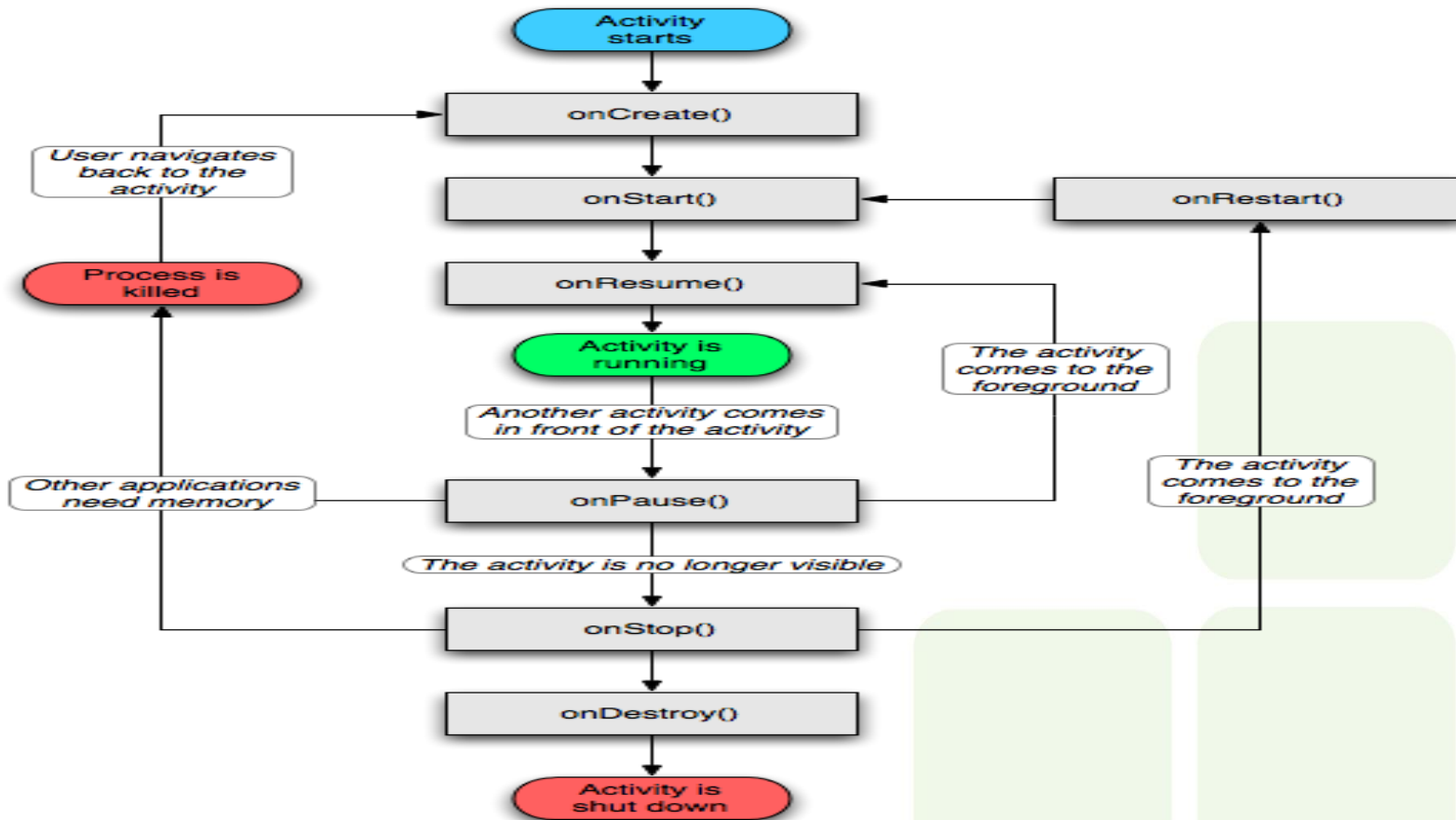
Conceitos Básicos

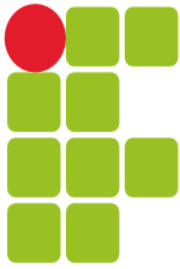
- Recursos
 - Aplicações Android não são apenas código Java
 - Exigem, figuras, vídeos, arquivos (configuração ou não)
 - Permite uma maior separação entre o código e a configuração da aplicação
 - Evita que o código precise ser modificado, quando for necessário modificar o alvo (dispositivo) na qual a aplicação será implantada



Conceitos Básicos

- Ciclo de Vida de uma Aplicação Android





Configurando o ambiente

- Usando uma versão pré-instalada do Eclipse
- Help -> Install new Software
- Adicionar novo site, com o nome Android e o repositório <https://dli-ssl.google.com/android/eclipse/>
- Ao concluir a instalação, será iniciado o download do SDK necessário para o desenvolvimento das aplicações



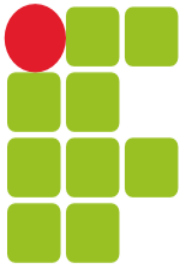
Configurando o Ambiente

- Android SDK Bundle
 - Acessar o site
<http://developer.android.com/sdk/index.html>
 - Baixar o arquivo de acordo com sua plataforma
 - Descompactar o arquivo em local apropriado
 - Abrir o arquivo eclipse **dentro da pasta eclipse**
 - Pronto, ambiente pronto para desenvolvimento!!



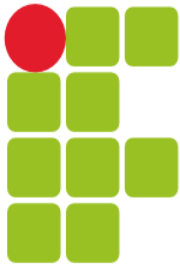
Primeiro Programa

- Vamos desenvolver um simples calculadora
- Etapas
 - Criar o projeto, especificando a plataforma de destino
 - Testando o projeto
 - Executar em um dispositivo móvel
 - Criar um Android Virtual Device(AVD)
 - Projetar a interface do usuário
 - Programar a interface
 - Executar o programa



Primeiro Programa

- Criar o Projeto
 - Abrir o Eclipse que possui a instalação do ADT
 - File -> New -> Android -> Android Application Project
 - Informar o nome da aplicação, nome do projeto(Eclipse), pacote inicial
 - Configuração do Android
 - Minimum Required SDK
 - Target SDK
 - Compile With
 - Theme



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE
Campus Parnamirim

Primeiro Programa

- Criar o Projeto

Application Name: Calculadora
Project Name: Calculadora
Package: br.edu.ifrn.calculadora
Theme: Holo Dark

Ao concluir, clique em next. Na tela seguinte aceite as configurações padrões e clique em next

New Android Application

New Android Application
Creates a new Android Application

Application Name:

Project Name:

Package Name:

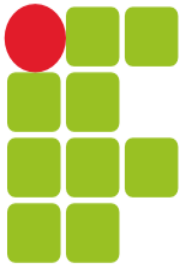
Minimum Required SDK:

Target SDK:

Compile With:

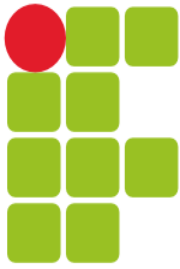
Theme:

Choose the base theme to use for the application



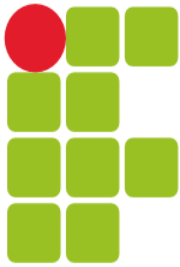
Primeiro Programa

- Criar o Projeto
 - Na tela seguinte podemos criar um ícone que será utilizado para lançar a aplicação
 - Vamos utilizar o padrão, podendo ser customizado de acordo com a necessidade do desenvolvedor.
 - No slide seguinte escolheremos o perfil da atividade a ser utilizada
 - Retomaremos o conceito de atividade mais adiante. Selecione o template Blank Activity



Primeiro Programa

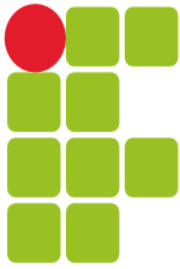
- Criar o Projeto
 - Informe o nome da atividade como MainCalculadora
 - Automaticamente o nome do layout(interface) é redefinido. Usaremos esse padrão. O tipo de navegação será NONE
 - Na tela seguinte, caso alguma dependência esteja faltando será fornecida a opção para instalação dessa dependência.



Primeiro Programa

- Criar o Projeto

The screenshot displays the Android Studio interface. On the left, the Project Explorer shows the project structure for 'Calculadora', including the 'src' directory with 'br.edu.ifrn.calculadora' and 'MainCalculadora.java', and the 'res' directory with 'layout' containing 'activity_main_calculadora.xml'. The main editor area shows the graphical layout of the activity, titled 'Calculadora', with a 'Hello world!' text view. The Palette on the left lists various widgets like TextView, Button, Spinner, and Text Fields. The top toolbar shows the device configuration as Nexus One, the theme as AppTheme, and the activity as MainCalculadora.



Primeiro Programa

- Testando o Projeto
 - Para executar qualquer programa Android, precisamos informar em qual dispositivo o mesmo será executado, seja um dispositivo virtual ou um dispositivo físico
 - Executaremos o teste das duas maneiras



Primeiro Programa

- Executar em um dispositivo físico
- Habilitando debug no dispositivo
 - Android < 3.2
 - Configurações -> Aplicativos -> Desenvolvimento -> Depuração USB
 - Android 4.0 – 4.1
 - Configurações -> Opções do Desenvolvedor
 - Android 4.2
 - Configurações -> Sobre o Telefone, pressione Build Number 7 vezes. Isso habilita a opção de Desenvolvimento



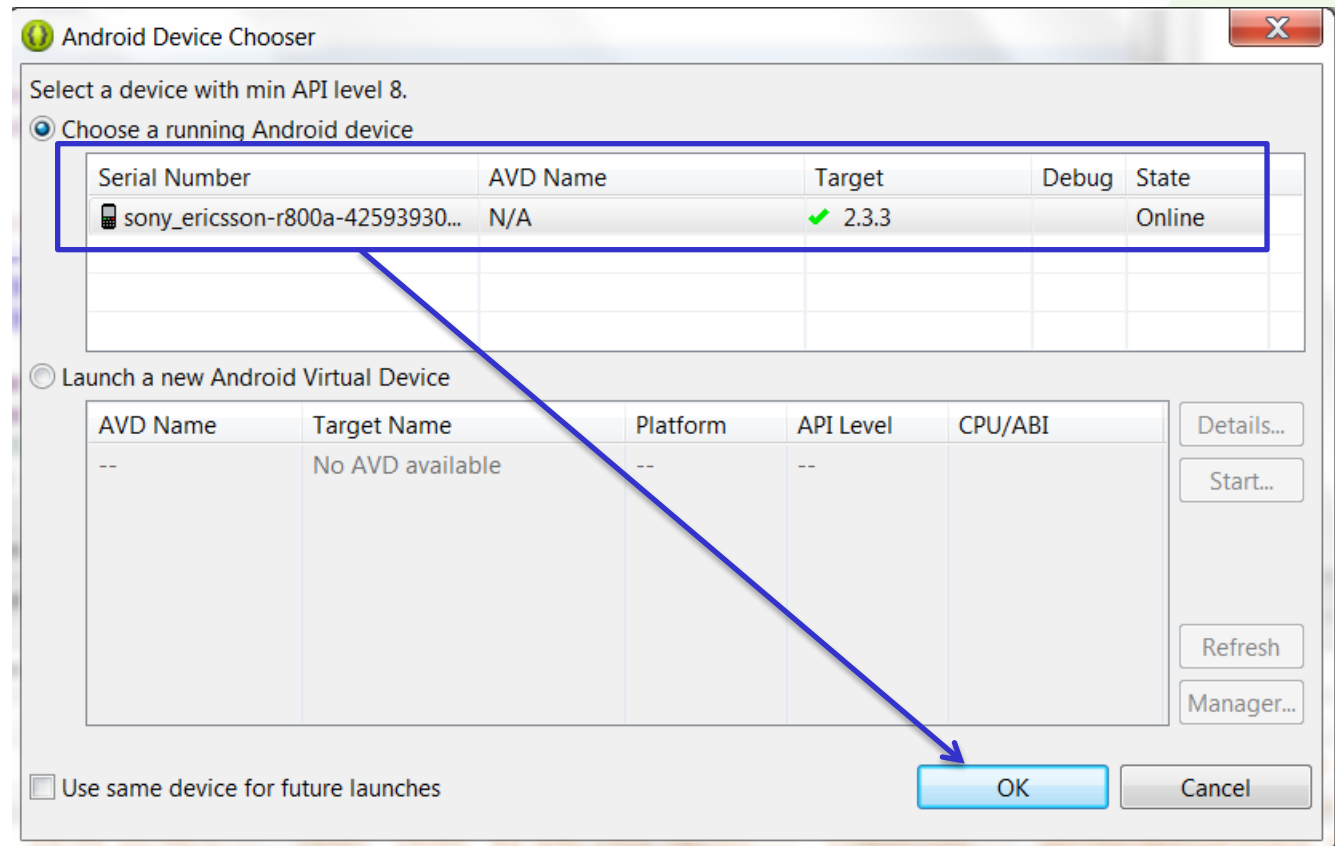
Primeiro Programa

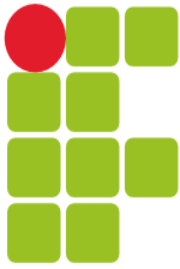
- Conecte o telefone/tablet ao computador
- Pode ser necessário instalar os drives do dispositivo para que o reconhecimento seja concluído
- Após conectado, selecione o projeto Calculadora, clique com o botão direito, Run As-> Android Application



Primeiro Programa

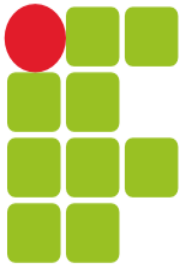
- Se tudo correr bem, o dispositivo será mostrado no *Android Device Chooser*





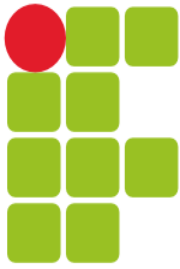
Primeiro Programa

- Executando em um dispositivo virtual
 - Usaremos o Android Virtual Device Manager, para criar um dispositivo virtual
 - Para criar um dispositivo virtual, precisamos selecionar um padrão de dispositivo fornecido pelo SDK ou criar um padrão próprio
 - Para usar criar um padrão próprio, devemos configurar as informações relativas ao hardware do dispositivo, tais como tela, presença de câmera, acelerômetro e etc...



Primeiro Programa

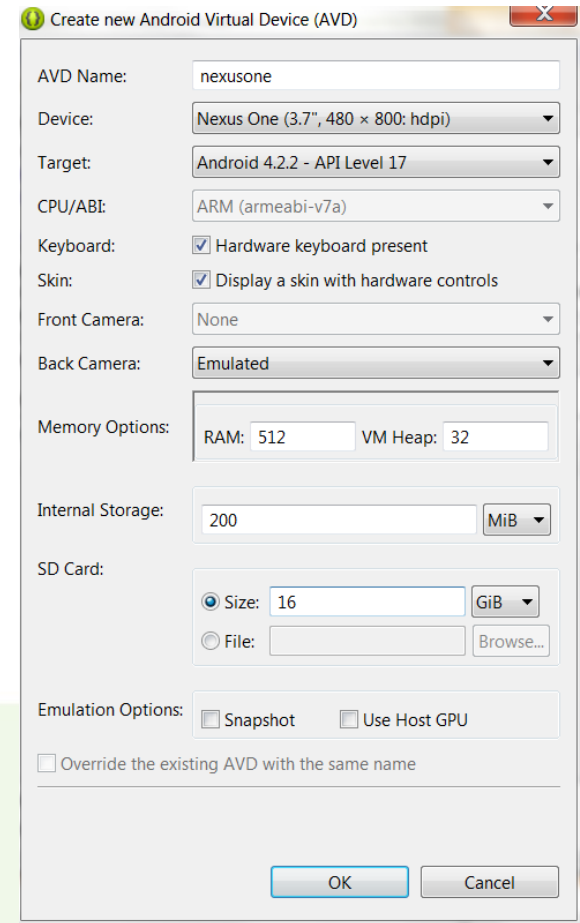
- Executando em um dispositivo virtual
 - Para configurar as propriedades, use o manual do seu celular o consulte http://en.wikipedia.org/wiki/List_of_Android_devices
 - Usaremos nesse primeiro momento um dos dispositivos virtuais fornecidos, o Nexus One da Google
 - Vamos executar a aplicação novamente, dessa vez criando um dispositivo virtual
 - Clique em **Launch a new Android Device**, em seguida clique em Manager, Na primeira aba clique new

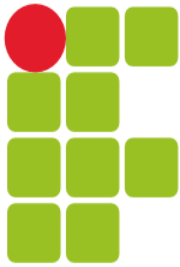


Primeiro Programa

- Executando em um dispositivo virtual

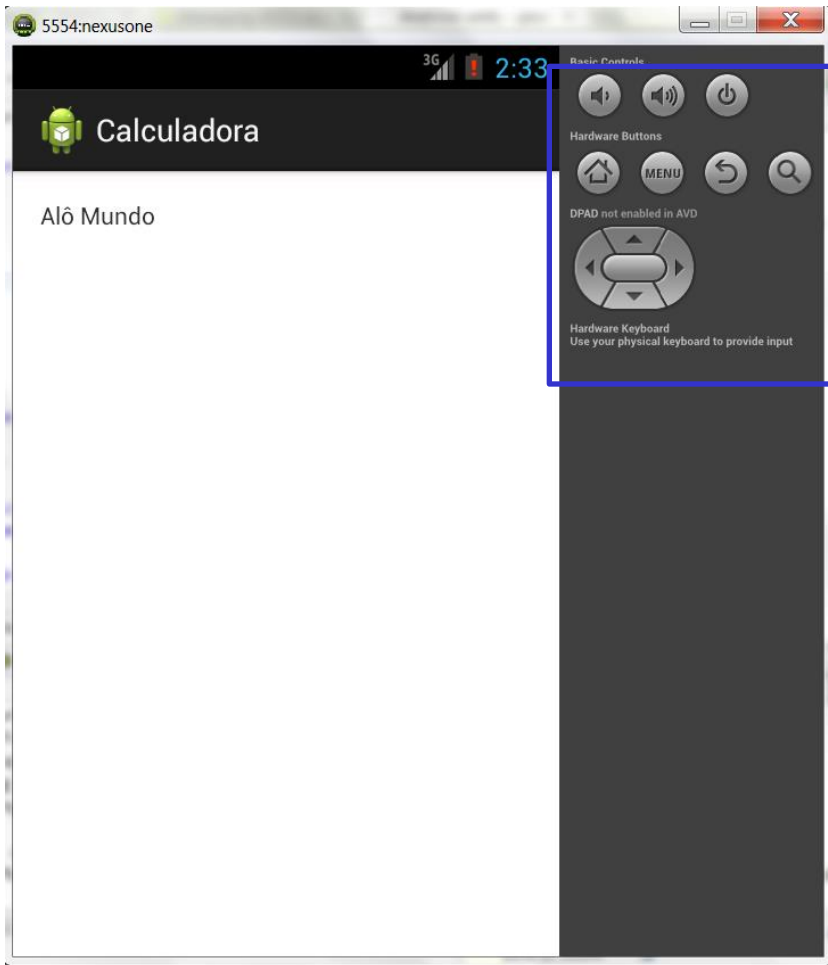
Propriedade	Valor
AVD Name	Nexusone
Device	Nexus One
Target	Android 4.2.2
Keyboard	Padrão
Skin	Padrão
Back Camera	Emulated
Memory	Padrão
Internal	200MB
SDCard	0



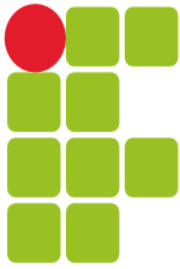


Primeiro Programa

- Executando em um dispositivo virtual



Use os botões ao lado para navegar pelo dispositivo



Projetando a Interface

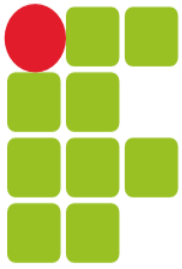
- A interface gráfica de uma aplicação Android é construída através de arquivos XML
- Utiliza os conceitos de View e ViewGroup, onde os componentes View são os widgets (botões, caixas de texto) e ViewGroups, funcionam como containers onde esses componentes são adicionados, organizando a estrutura da interface
- A definição da interface fica no arquivo `layout/activity_layout_calculadora`



Projetando a Interface

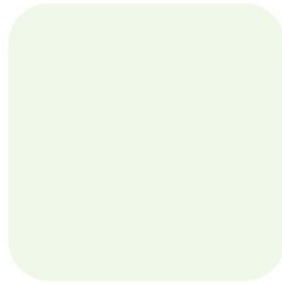
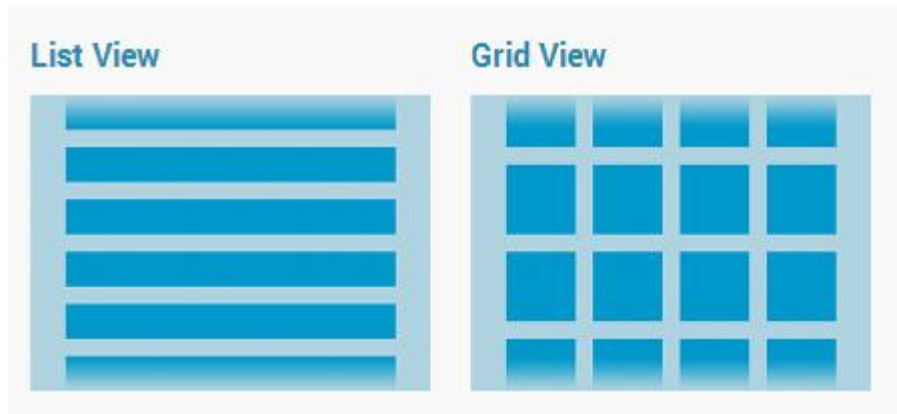
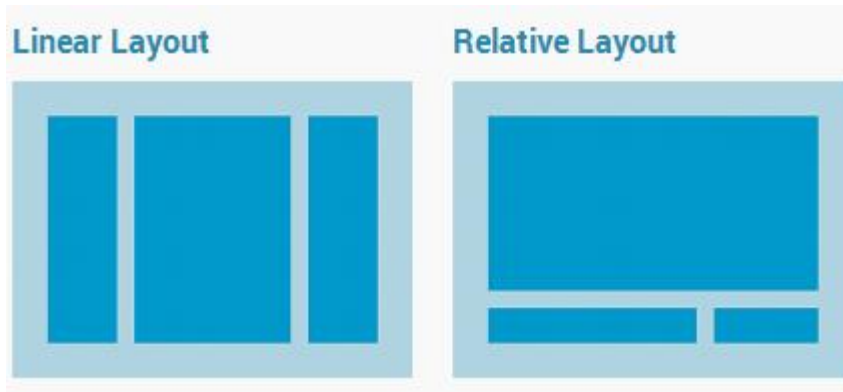
- Layouts
 - Determinam o sistema de posicionamento dos elementos da interface
 - Layouts comuns
 - Linear Layout
 - Relative Layout
 - Layouts Dinâmicos
 - List View
 - Grid View





Projetando a Interface

- Layouts



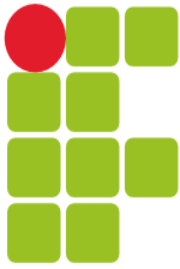


Projetando a Interface

- Componentes de Interface
- Controles de Entrada de Dados/Exibição

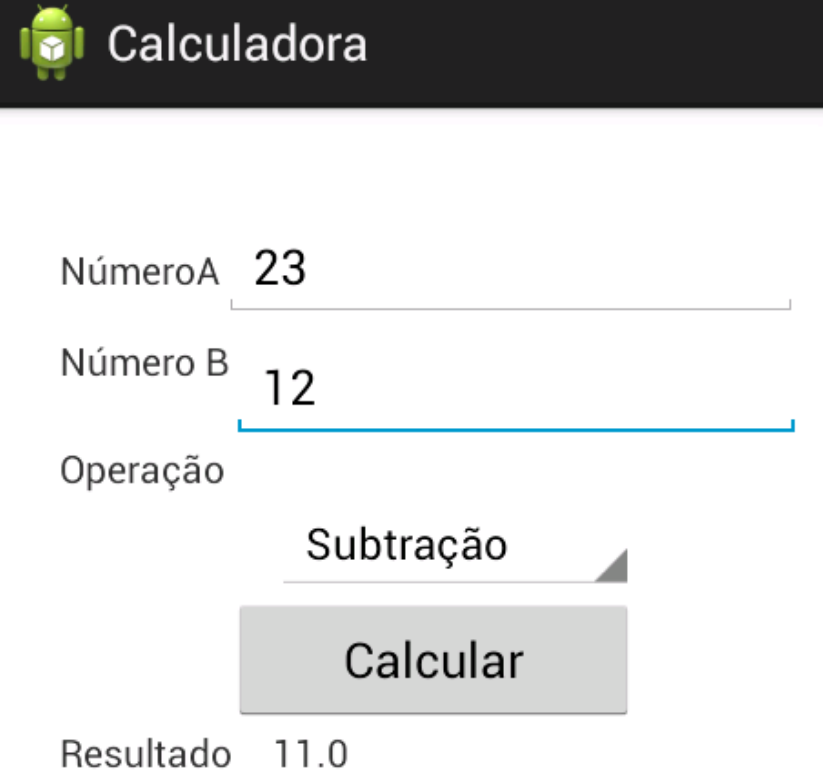
Componente	Funcionalidade
Button	Botão para cliques do usuário
Text field	Caixa de texto
Checkboxes	Caixas de marcação(Múltipla Seleção)
RadioButton	Botão de rádio(Seleção única)
Toggle Button	Botão que se assemelha ao ligar/desligar
Spinner	Caixa do tipo combo, que o usuário pode selecionar uma opção
Pickers	Caixa para seleção de datas(DatePickers) e hora(TimePickers)
TextView	Funciona com rótulos

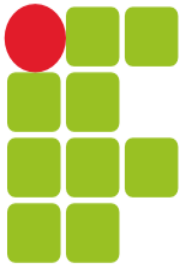




Projetando a Interface

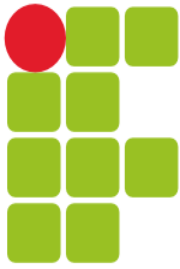
- Vamos construir uma interface que se assemelha a figura abaixo





Programar a Interface

- Com a interface desenhada, podemos avançar para a programação
- Os componentes da interface gráfica são acessados através da API do Android
- Para recuperar/enviar informações da interface, precisamos localizar os componentes
- O Android usa o conceito de recursos(resources), que refere-se a todos os arquivos de suporte ao programa



Programar a Interface

- Para o nosso programa os passos:
 - Localizar a referência a nossos componentes
 - Implementar método associado ao clique do botão
- A programação fica associada a atividade
- Devemos antes declarar os widgets como atributos privados da nossa classe
- Dentro do método onCreate, na classe MainCalculadora



Programar a Interface

- Método onCreate

```
super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main_calculadora);  
  
    txtNumeroA=(EditText)findViewById(R.id.txtNumeroA);  
  
    txtNumeroB=(EditText)findViewById(R.id.txtNumeroB);  
  
    txtResultado=(TextView)findViewById(R.id.txtResultado);  
  
    spinnerOperacao=(Spinner)findViewById(R.id.spinnerOperacao);
```



Programar a Interface

- Associando um método a um evento
 - Vamos associar um método ao evento onClick do nosso botão calcular
 - Para isso devemos informar o nome do método na propriedade onClick
 - Usaremos o nome calcular
 - Seguindo a regra do Android a assinatura do método deve ser calcular(View v)
 - Vejamos o código



Programar a Interface

- Método calcular

```
double numeroA=Double.parseDouble(txtNumeroA.getText().toString());
```

```
double numeroB=Double.parseDouble(txtNumeroB.getText().toString());
```

```
double resultado=0;
```

```
int op=spinnerOperacao.getSelectedItemPosition();  
//realizar a o cálculo baseado na operação
```

```
txtResultado.setText(Double.toString(resultado));
```



Conclusão

- Tivemos uma visão geral do Android
- Questões gerais sobre desenvolvimento
- Na próxima etapa veremos como usar recursos específicos o hardware do dispositivo.