

Exercícios PDP Plataforma Android - 2013-1

Exercício 1:

Dado um servidor de Socket, que efetua a soma de dois números inteiros baseado no protocolo na tabela abaixo, faça um cliente de Socket em Android e uma interface com dois EditText que conterão os números a serem somados e submetidos e exibir o resultado com um Toast ou em um TextView. A aplicação deve permitir fazer a soma de dois números diversas vezes. Ao sair deve-se enviar uma mensagem ao servidor afirmando que o cliente vai se desconectar.

Para facilitar a programação, abaixo da tabela é possível encontrar o código do Servidor (Em funcionamento) e o código de um cliente se socket (também funcionando, em java SE).

Palavra	Descrição	Origem -> Destino
HELLO	Pede para conectar-se com o servidor	Cliente -> Servidor
WELCOME	Conexão Aceita	Servidor -> Cliente
REJECT	Conexão Negada	Servidor -> Cliente
SUM;num1;num2	Soma os valores de num1 com num2	Cliente -> Servidor
RESULT;numero	Envia o resultado da soma em numero	Servidor -> Cliente
EXIT	Fecha a conexão no servidor	Cliente -> Servidor

Protocolo das Mensagens

Endereço IP e Porta do servidor de Socket no laboratório:

IP do Servidor: xxx.xxx.xxx.xxx

Porta do Servidor: 8513

OBS.: Não esquecer de fornecer a permissão de uso da internet.

Código do Servidor:

- <https://docs.google.com/file/d/0B746pjy4jQVAVnppaXIXMTJfMIU/edit?usp=sharing>

Código do Cliente Socket SE:

- <https://docs.google.com/file/d/0B746pjy4jQVAYnlKcnZPZHFGTEE/edit?usp=sharing>

Documentação Java da Classe:

- <http://docs.oracle.com/javase/1.4.2/docs/api/java/net/Socket.html>

Exercício 2:

Usar a API de Geo localização pela classe LocationManager e obter a latitude, longitude e altura atual, exibindo essas informações na tela (Seja com Toast ou com TextView).

OBS.: O GPS do emulador obviamente é emulado, então deve-se adicionar Coordenadas GPS remotamente (Desnecessário para um dispositivo Real):

- telnet <ipEmulador> <portaEmulador>
- telnet localhost 5554
- geo fix <longitude> <latitude> [<altura>]
- geo fix -82.411629 28.054553 44
- GEO nmea (Opcional - Não determinante para o funcionamento desta aplicação)
- geo nmea \$GPGGA,001431.092,0118.2653,N,10351.1359,E,0,00,,-19.6,M,4.1,M,,0000*5B

Também, não devem esquecer de dar as permissões no AndroidManifest.xml:

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

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

```
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
```

Usar de base a aplicação exemplo de localização da K19 (Capítulo 10):

- <http://www.k19.com.br/downloads/apostilas/java/k19-k41-desenvolvimento-mobile-com-android>

Exercício 3:

Deve-se implementar um sistema em que busque a localização atual (latitude e longitude) através e exiba um ponto em no mapa com a latitude e a longitude atual. Utilizar a biblioteca Google Maps v2 para Android. E a classe LocationManager para retornar as coordenadas, idem exercício 2.

Dica: Faça funcionar primeiro somente o MAPA utilizando coordenadas fixas, depois adicionar o recurso para buscar a posição por Geo Localização.

Para tanto é necessário entrar na SDK e baixar o pacote Google APIs 17. Após criar uma Máquina virtual com essa API, caso não apareça no momento da criação, clique em refresh.

Link referência exercício em 2011:

- <https://saloon.inf.ufrgs.br/wiki/view/Disciplinas/INF01008/LabAndroid2011>

Link referência biblioteca (pagina 78):

- <http://www.k19.com.br/downloads/apostilas/java/k19-k41-desenvolvimento-mobile-com-android>

Configurar Máquina Virtual para aceitar mapas:

- <https://developers.google.com/maps/documentation/android/?hl=pt-BR#avdsetup>

Gerar API Key:

Na home do usuário: keytool -list -v -keystore .android/debug.keystore

OBS.: Para comando funcionar em alguns computadores o idioma do sistema deve

estar em inglês (se for em português ocorre erros).

Depois -> Enter keystore password: android

- Copiar código SHA1 e criar uma chave para Maps Android v2 no console da Google (<https://code.google.com/apis/console/>).

Links:

- developers.google.com/maps/documentation/android/start#the_google_maps_api_key
- <https://code.google.com/apis/console/>

Também, não devem esquecer de dar as permissões no AndroidManifest.xml:

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

Gabarito dos Exercícios:

- 1) <https://docs.google.com/file/d/0B746pjy4jQVAVUJ6dWR1Vi1tbGM/edit?usp=sharing>
- 2) <https://docs.google.com/file/d/0B746pjy4jQVANKh2ZTVVnFiZ1E/edit?usp=sharing>
- 3) <https://docs.google.com/file/d/0B746pjy4jQVAOFdmak85Q2o3NjQ/edit?usp=sharing>

Link dos Slides Conceituais:

<https://docs.google.com/file/d/0B746pjy4jQVAS0tjNzQzTS1VT2c/edit?usp=sharing>

Link dos Slides Implementação

<https://docs.google.com/file/d/0B746pjy4jQVATnJpRTlwTDA2SUE/edit?usp=sharing>