

Android Development Without Java

Android Development With Scala

Scala

Scala is

Scala is

A Functional

Scala is

A Functional
Object Oriented

Scala is

A Functional
Object Oriented
Statically Typed

Scala is

A Functional
Object Oriented
Statically Typed
Scalable

Scala is

A Functional
Object Oriented
Statically Typed
Scalable
Language

Scala is

A Functional
Object Oriented
Statically Typed
Scalable
Language

Running on JVM

Scala is

A Functional
Object Oriented
Statically Typed
Scalable
Language

Running on **JVM** and **DalvikVM**

Why Scala

Java

```
public class MyActivity extends Activity {
```

```
    ListView comments;
```

```
    Button newComment;
```

```
    @Override
```

```
    void onCreate(Bundle bundle) {
```

```
        super.onCreate(bundle);
```

```
        comments = (ListView) findViewById(R.id.comments);
```

```
        newComment = (Button) findViewById(R.id.new_comment);
```

```
        //...
```


```
    }
```

```
    public void sayHello() {
```

```
        Toast.makeText(this, "Hello", Toast.LENGTH_LONG).show();
```

```
    }
```

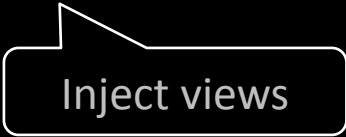
```
}
```



Initialize context and views

Java with RoboGuice

```
public class MyActivity extends RoboActivity {  
    @Inject Application context;  
    @Inject SessionManager sessionManager;  
  
    @Inject(R.id.comments) ListView comments;  
    @Inject(R.id.new_comment) Button newComment;
```



Inject views

```
public void sayHello() {  
    Toast.makeText(this, "Hello", Toast.LENGTH_LONG).show();  
}  
}
```



GO HOME ANDROID. YOU ARE DRUNK.

Scala

```
class MyActivity extends ScalaActivity {  
  [Implicit Context]  
  lazy val SessionManager  
  
  lazy val comments = findView(TR.comments)  
  lazy val newComment = findView(TR.new_comment)
```

Lazily load views

Toast using implicit conversion

```
def sayHello() = "Hello".toast()
```

```
}
```


A Sample Project



shadowsocks

- [github/shadowsocks/shadowsocks-android](https://github.com/shadowsocks/shadowsocks-android)
- written in C/C++ and Scala
- modules
 - [shadowsocks-libev](#): secured socks5 proxy in pure C
 - [shadowsocks-android](#): UI/Android interface in Scala
 - [lwIP/tun2socks](#): user space TCP/IP stack in C++
 - [pdnsd/iptables/redsocks](#): native porting for android

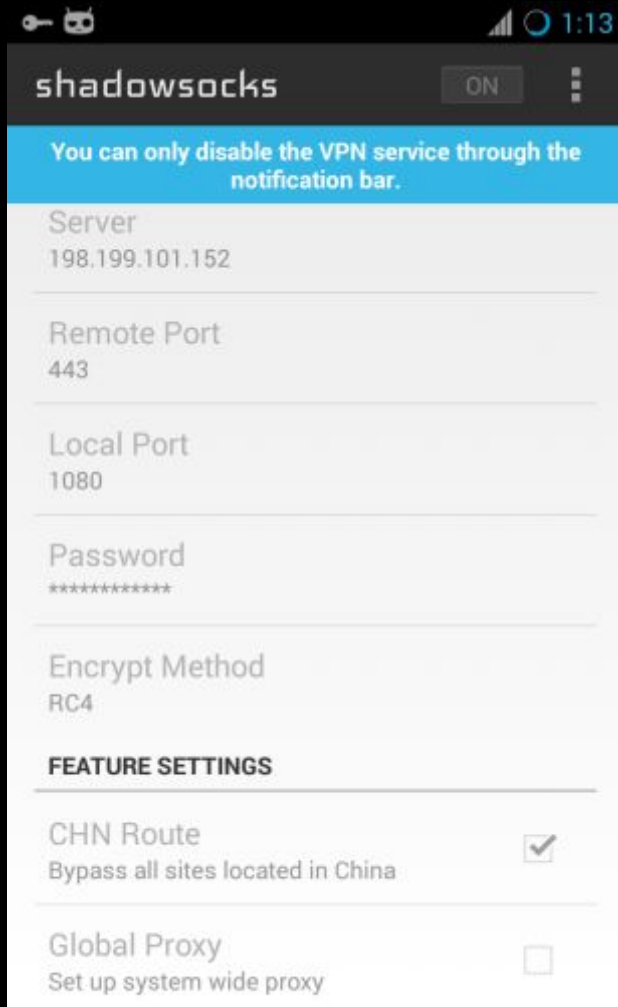
Project Structure

```
project/  
  Build.scala          <SBT makefile>  
  plugins.sbt         <SBT plugin config>  
src/  
  main/  
    res/              <resource files>  
    assets/          <asset files>  
    jni/             <native codes>  
    libs/            <jar and native libraries>  
    scala/           <main Scala sources>  
    java/            <main Java sources>  
    AndroidManifest.xml <manifest template>  
    test/            <test sources>
```

SBT

- Simple Build Tool for Scala / Java
 - Ant, Maven, Gradle like tool in Scala
- [github/jberkel/android-plugin](https://github.com/jberkel/android-plugin)
 - SBT plugin for Android
 - Dependency management, unit test, instrumentation, auto tasks, etc

Components



- UI
 - ActionBarSherlock
 - SwitchBackport
 - Crouton
 - UnifiedPreference
 - UniversalImageLoader
- Libraries
 - DNSJava
 - Analytics
 - AdMob

Components

```
4
5 libraryDependencies += Seq(
6   "com.google.android" % "support-v4" % "r12",
7   "com.google.android.analytics" % "analytics" % "2.0beta4",
8   "dnsjava" % "dnsjava" % "2.1.5",
9   "com.nostra13.universalimageloader" % "universal-image-loader" % "1.8.4",
10  "com.google.android.admob" % "admob" % "6.3.1"
11 )
12
13 libraryDependencies += Seq(
14  "com.actionbarsherlock" % "actionbarsherlock" % "4.4.0" artifacts(Artifact
15  "net.saik0.android.unifiedpreference" % "unifiedpreference" % "0.0.2" arti
16  "org.jraf" % "android-switch-backport" % "1.0" artifacts(Artifact("android
17  "de.keyboardsurfer.android.widget" % "crouton" % "1.7"
18 )
19
```

Best Practices

- Proguard
 - Aggressive default rules
- IntelliJ IDEA
 - Best IDE for Scala and Android
- Continuous Integration
 - Travis is best, Jenkins works well

Questions?

Thanks!

About Me

- Max Lv
 - Member of Tangcha Project
 - 字节社 for Android, <http://zijieshe.com>
 - Major Contributor of GoAgent Project
 - [github/madeye/gaeproxy](https://github.com/madeye/gaeproxy)
 - Core maintainer of Shadowsocks Project
 - [github/madeye/shadowsocks-libev](https://github.com/madeye/shadowsocks-libev)
 - [github/shadowsocks/shadowsocks-android](https://github.com/shadowsocks/shadowsocks-android)