The University of Tokyo, School of Engineering
Small group seminar
2010/10/20 to 2010/01/19

Guillaume Lopez
Department of Mechanical Engineering

Understanding Heart and Body Status from a Smartphone

– Introduction on how to collect, display, measure, and communicate sensor signal –

Smartphone first application programming tutorial

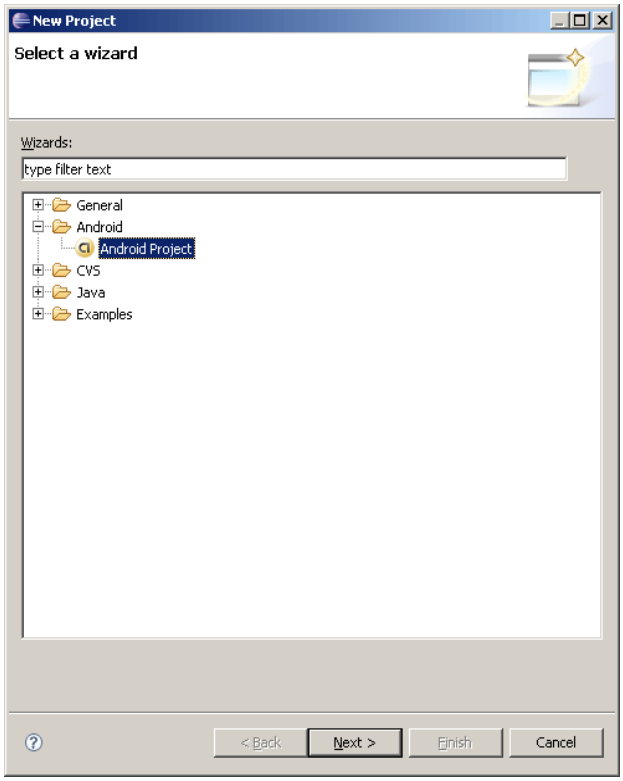
1 Create a first Android application

1.1 Hello Android

Now that the working environment is set, let's create a basic Android application to test it and understand Eclipse development environment.

See http://developer.android.com/resources/tutorials/hello-world.htmlfor detailed explanations.

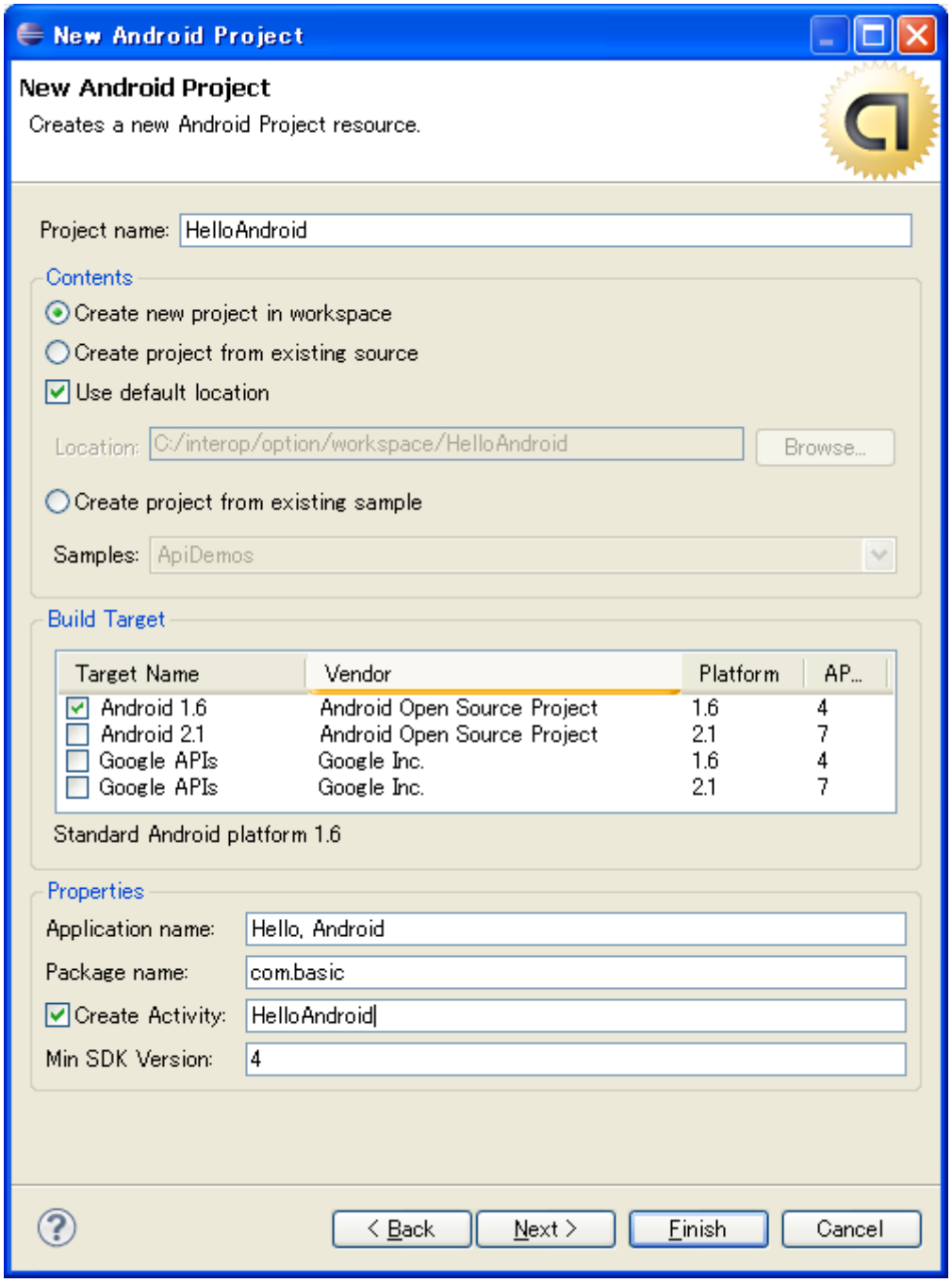
i. Create a new Project: From Eclipse, select **File** >**New** >**Project**. If the ADT Plugin for Eclipse has been successfully installed, the resulting dialog should have a folder labeled "**Android**" which should contain "**Android Project**". Select "**Android Project**" and click **Next**.

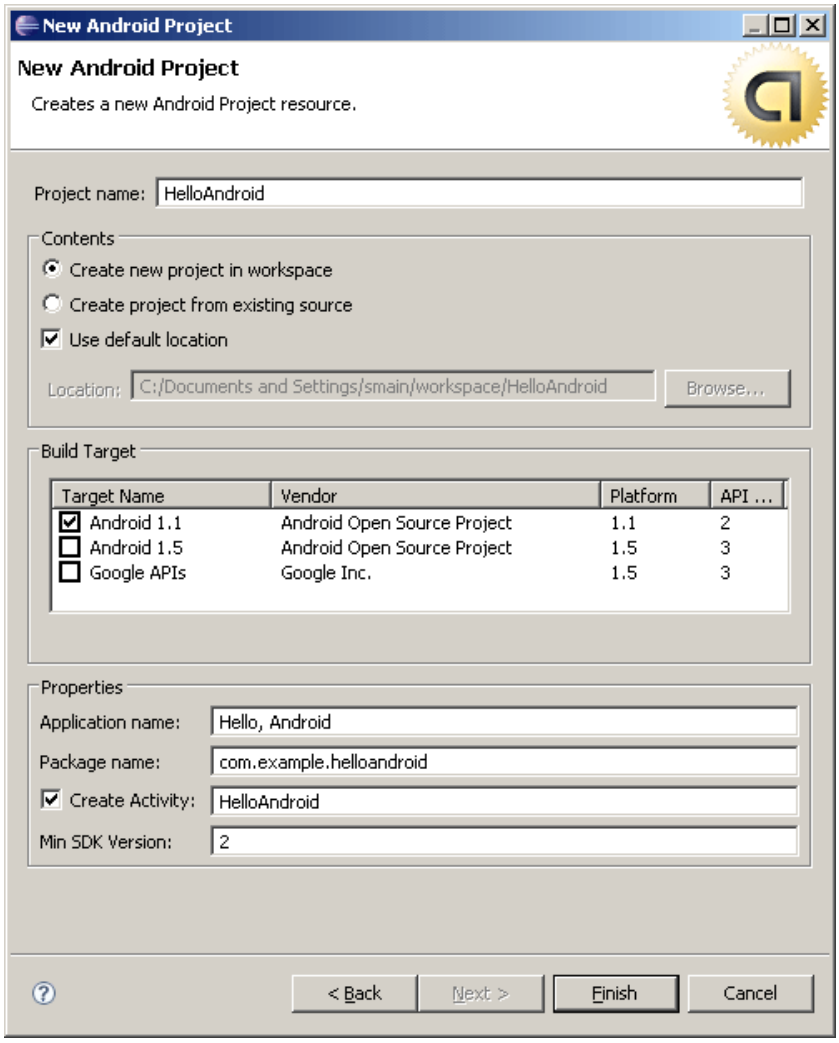


Fill in the project details with the following values:

- *Project name:* HelloAndroid
- *Application name:* Hello, Android
- *Package name:* com.example.helloandroid (or your own private namespace)
- *Create Activity:* HelloAndroid
- *Min SDK Version:* 2

Click **Finish**.





The Android project is now ready. It should be visible in the Package Explorer on the left. Open the"**HelloAndroid.java**" file, located inside **HelloAndroid**>**src**>**com.basic**. It should look like this:

```
Package com.basic;

import android.app.Activity;
import android.os.Bundle;

public class HelloAndroid extends Activity{
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState){
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Notice that the class is based on the Activity class. An Activity is a single application entity that is used to perform actions. An application may have many separate activities, but the user interacts with them one at a time. The onCreate() method will be called by the Android system when your Activity starts — it is where you should perform all initialization and UI setup. An activity is not required to have a user interface, but usually will.
Now let's modify some code!

Guillaume Lopez
Department of Mechanical Engineering

ii. Construct the basic UI of your application

Take a look at the revised code below and then make the same changes to your HelloAndroid class. The bold items are lines that have been added.

```
Package com.basic;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HelloAndroid extends Activity{
   /** Called when the activity is first created. */
   @Override
   public void onCreate(Bundle savedInstanceState){
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("Hello, Android");
        setContentView(tv);
   }
}
```

An Android user interface is composed of hierarchies of objects called Views. A View is a drawable object used as an element in your UI layout, such as a button, image, or (in this case) a text label. Each of these objects is a subclass of the View class and the subclass that handles text is TextView.

In this change, you create a TextView with the class constructor, which accepts an Android Context instance as its parameter. A Context is a handle to the system; it provides services like resolving resources, obtaining access to databases and preferences, and so on. The Activity class inherits from Context, and because your HelloAndroid class is a subclass of Activity, it is also a Context. So, you can pass this as your Context reference to the TextView.

Next, you define the text content with setText(CharSequence) setText().
Finally, you pass the TextView to setContentView() in order to display it as the content for the Activity UI. If your Activity doesn't call this method, then no UI is present and the system will display a blank screen.
There it is — "Hello, World" in Android! The next step, of course, is to see it running.

iii. Run the Application

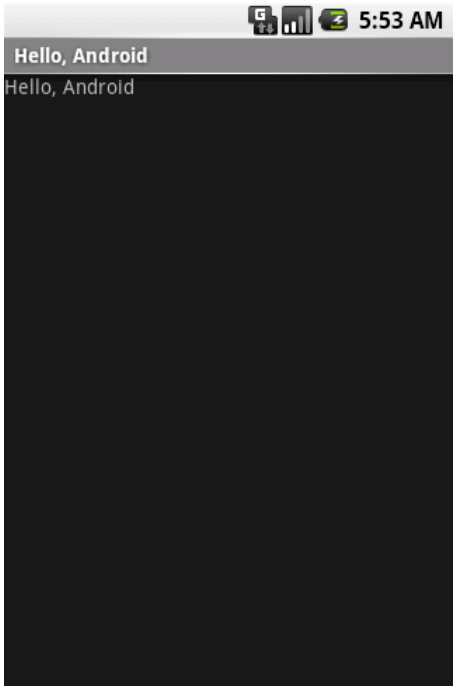
The Eclipse plugin makes it very easy to run your applications:

Select **Run >Run**.

Select "**Android Application**".

The Eclipse ADT will automatically create a new run configuration for your project and the Android Emulator will automatically launch. Once the emulator is booted up, your application will appear after a moment. You should now see something like this:

Guillaume Lopez
Department of Mechanical Engineering



The "Hello, Android" you see in the grey bar is actually the application title. The Eclipse plugin creates this automatically (the string is defined in the res/values/strings.xml file and referenced by your AndroidManifest.xml file). The text below the title is the actual text that you have created in the TextView object.

That concludes the basic "Hello Android" tutorial