CS 696 Emerging Web and Mobile Technologies
Spring Semester, 2011
Doc 21 Testing and Some Tools
Apr 14, 2011

Testing References

JUnit Cookbook http://junit.sourceforge.net/doc/cookbook/cookbook.htm

JUnit Test Infected: Programmers Love Writing Tests http://junit.sourceforge.net/doc/testinfected/testing.htm

JUnit Javadoc: http://www.junit.org/junit/javadoc/3.8/index.htm, http://junit.org/junit/javadoc/4.5/

JUnit FAQ, http://junit.sourceforge.net/doc/faq/faq.htm

Testing for Programmers, Brian Marick, Available at: http://www.exampler.com/testing-com/writings.html

Android Documentation, http://developer.android.com/reference/android/test/ActivityTestCase.html

Testing

Testing

Johnson's Law

If it is not tested it does not work

The more time between coding and testing

- More effort is needed to write tests
- More effort is needed to find bugs
- Fewer bugs are found
- Time is wasted working with buggy code
- Development time increases
- Quality decreases

Unit Testing

Tests individual code segments

Automated tests

When to Write Tests

First write the tests

Then write the code to be tested

Writing tests first saves time

- Makes you clear of the interface & functionality of the code
- Removes temptation to skip tests

What to Test

Everything that could possibly break

Test values

- Inside valid range
- Outside valid range
- On the boundary between valid/invalid

GUIs are very hard to test

- Keep GUI layer very thin
- Unit test program behind the GUI, not the GUI

Common Things Programs Handle Incorrectly

Adapted with permission from “A Short Catalog of Test Ideas” by Brian Marick,

http://www.testing.com/writings.html

Strings

Empty String

Collections

Empty Collection

Collection with one element

Collection with duplicate elements

Collections with maximum possible size

Numbers

Zero

The smallest number

Just below the smallest number

The largest number

Just above the largest number

XUnit

Free frameworks for Unit testing

SUnit originally written by Kent Beck 1994

JUnit written by Kent Beck & Erich Gamma

Available at: http://www.junit.org/

Ports to many languages at:
    http://www.xprogramming.com/software.htm

Sample JUnit 4.x Example

```
import static org.junit.Assert.*;
import java.util.ArrayList;
import org.junit.Before;
import org.junit.Test;

public class HelloWorldTest {
	int testValue;
	@Test
	public void testMe() {
		assertEquals(1, testValue);
	}

	@Test
	public void foo() {
		assertTrue(2 == testValue);
	}
```

```
	@Test
(expected=IndexOutOfBoundsException.class)
    public void testIndexOutOfBoundsException() {
        ArrayList emptyList = new ArrayList();
        Object notValid = emptyList.get(0);
    }

	@Before
	public void initialize(){
		testValue = 1;
	}
}
```

Assert Methods

assertTrue()
assertFalse()
assertEquals()
assertNotEquals()
assertSame()
assertNotSame()
assertNull()
assertNotNull()
fail()

For a complete list see

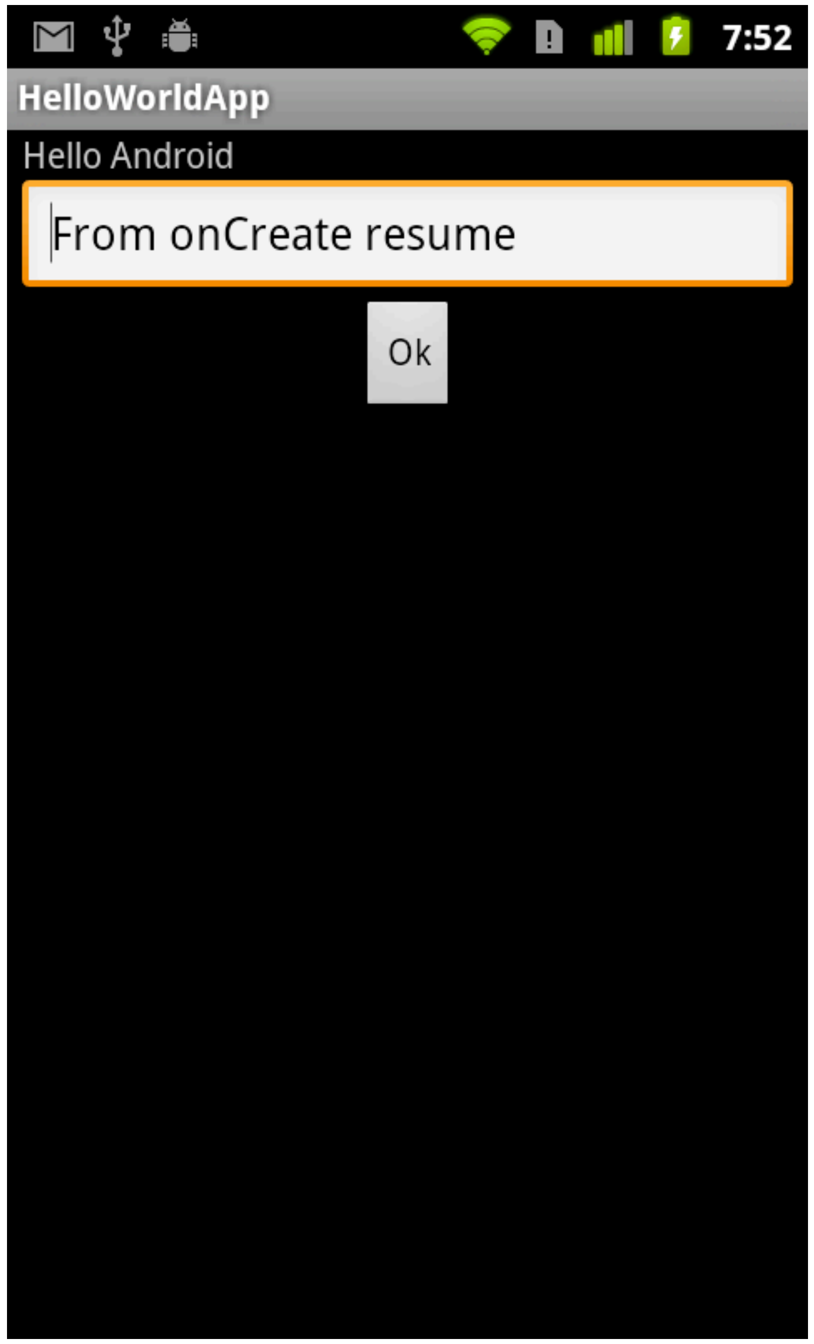
http://www.junit.org/junit/javadoc/3.8/index.htm/allclasses-frame.html/junit/junit/framework/Assert.html/Assert.html

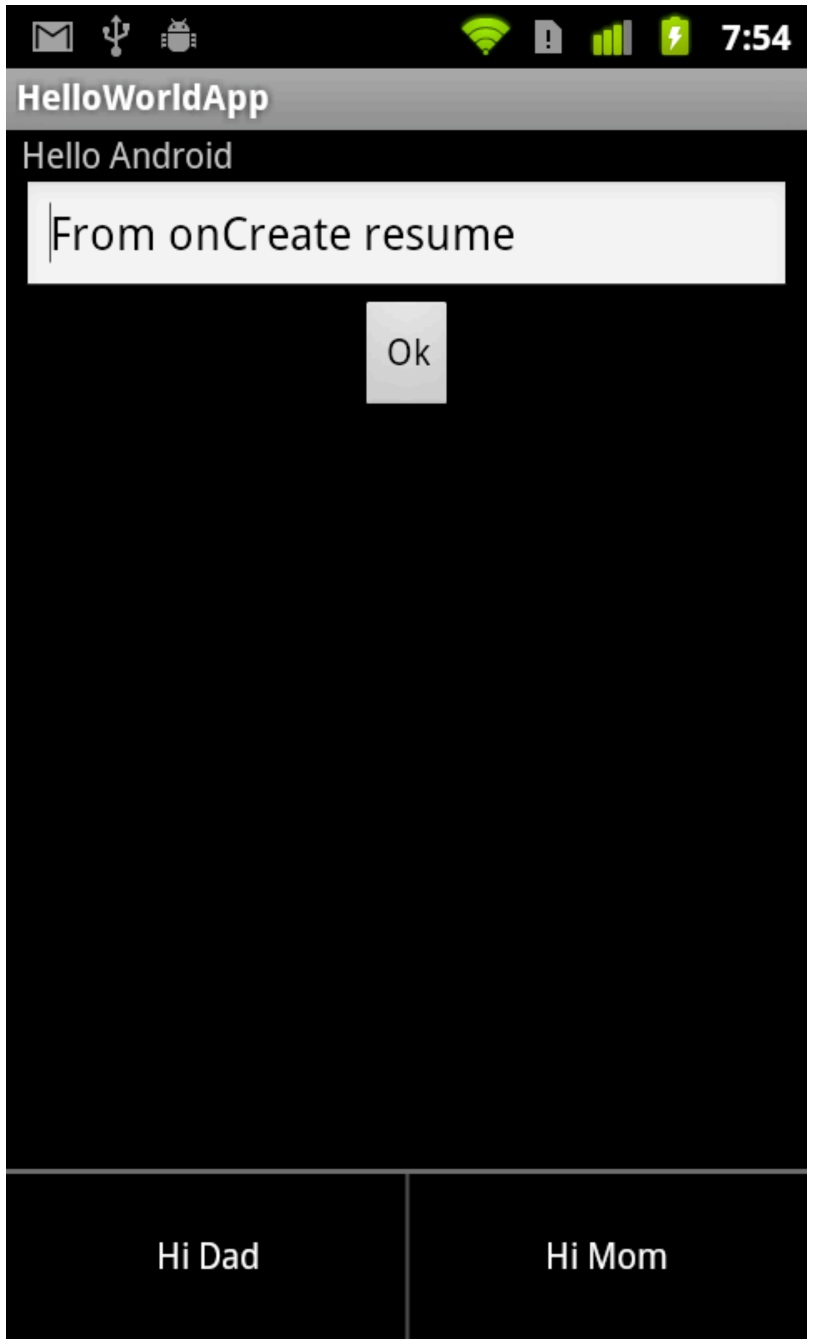
Android Testing

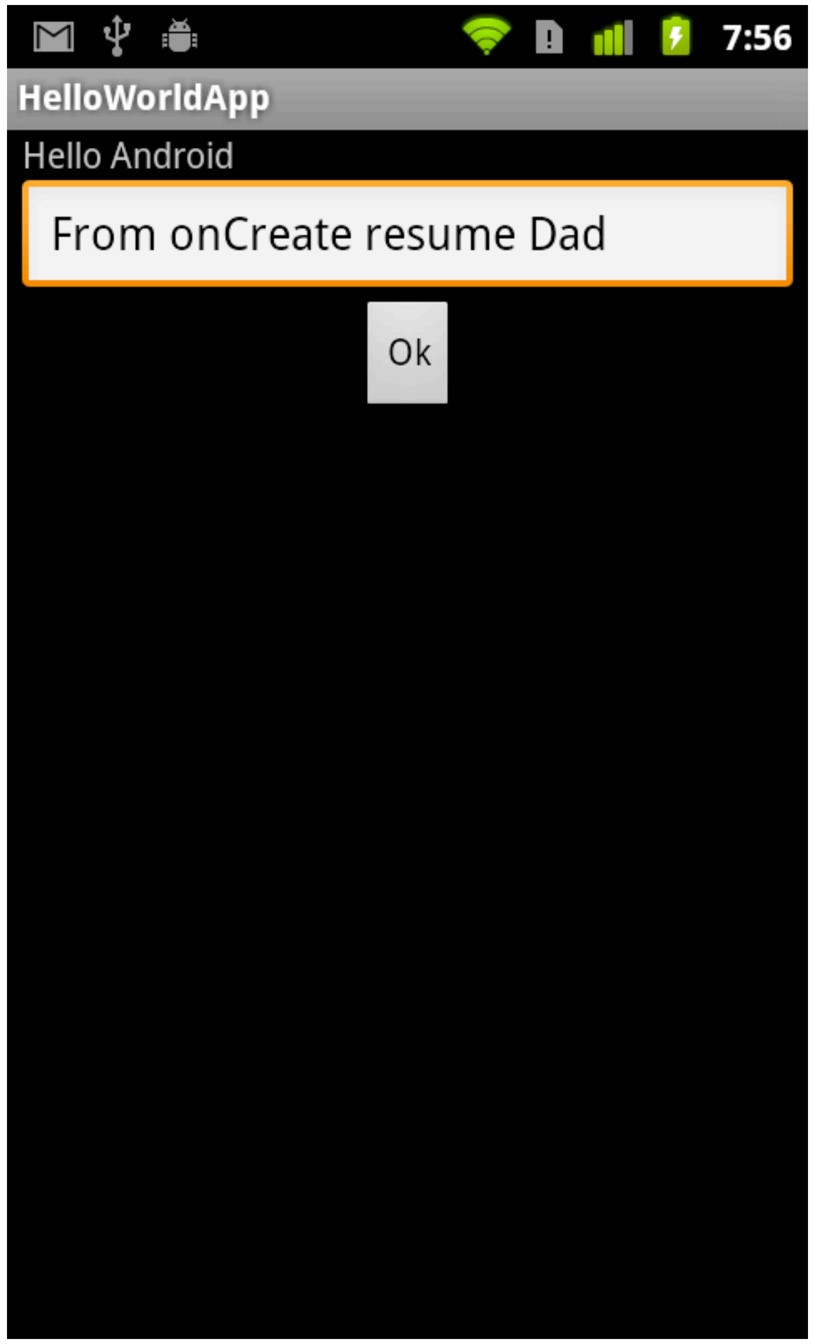
Unit Testing

- Application logic independent of UI/OS events
    - Normal JUnit tests
- Logic dependent on UI/OS events
    - Require special environment

Application to test







Application to test

```
package sdsu.cs696;

//imports not listed

public class HelloAndroid extends Activity implements View.OnClickListener {
	private EditText messageText;
	private static final int DAD_ID = Menu.FIRST;
	private static final int MOM_ID = Menu.FIRST + 1;

	public void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.main);
		messageText = (EditText) this.findViewById(R.id.message);
		messageText.setText("From onCreate");
		Button ok = (Button) findViewById(R.id.ok);
		ok.setOnClickListener(this);
	}
}
```

Application to test

```
public void onClick(View v) {
	messageText.setText(messageText.getText() + " click");
}

protected void onPause() {
	messageText.setText(messageText.getText() + " pause");
	super.onPause();
}

protected void onResume() {
	super.onResume();
	messageText.setText(messageText.getText() + " resume");
}
```

Application to test

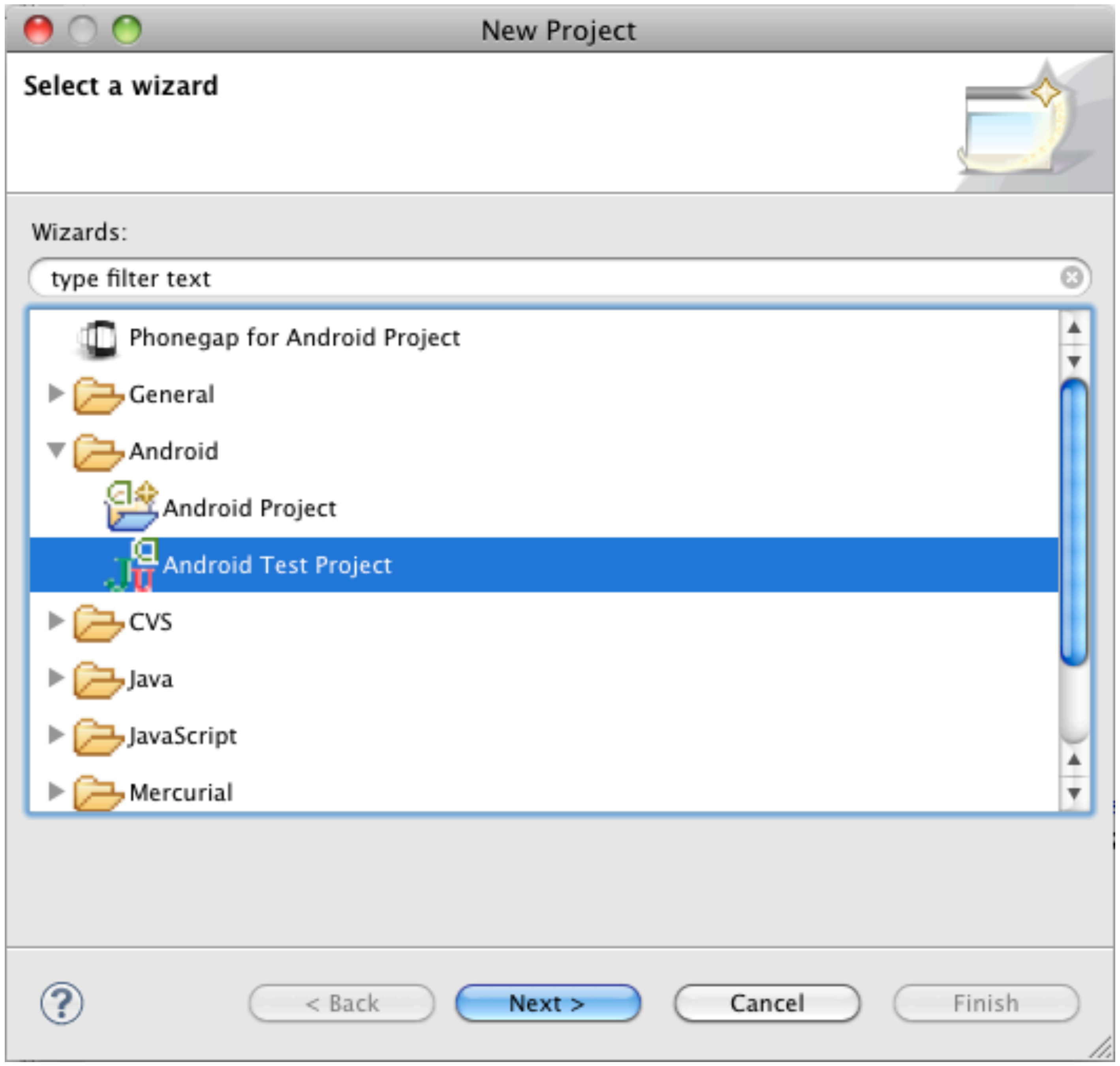
```
public boolean onCreateOptionsMenu(Menu menu) {
    super.onCreateOptionsMenu(menu);

    menu.add(0, DAD_ID, 0, R.string.menu_dad).setShortcut('0', 'd');
    menu.add(0, MOM_ID, 0, R.string.menu_mom);
    return true;
}

public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
    case DAD_ID:
        messageText.setText(messageText.getText() + " Dad");
        return true;
    case MOM_ID:
        messageText.setText(messageText.getText() + " Mom");
        return true;
    }
    return super.onOptionsItemSelected(item);
}
```

Test Setup

http://developer.android.com/resources/tutorials/testing/helloandroid_test.html



Start of test

```
package sdsu.cs696.test;

// imports not shown

public class HelloAndroidTest extends
        ActivityInstrumentationTestCase2<HelloAndroid> {
    private HelloAndroid mActivity;
    private EditText mView;
    private Button mButton;
    private Instrumentation mInstrumentation;

    public HelloAndroidTest() {
        super("sdsu.cs696", HelloAndroid.class);
    }
```

Set up

```
private String getText() {
    String resultText = mView.getText().toString();
    Log.i("rew", resultText);
    return resultText;
}

protected void setUp() throws Exception {
    super.setUp();
    mInstrumentation = getInstrumentation();
    mActivity = this.getActivity();
    mView = (EditText) mActivity.findViewById(sdsu.cs696.R.id.message);
    mButton = (Button) mActivity.findViewById(sdsu.cs696.R.id.ok);
}
```

Tests

```
public void testPreconditions() {
        assertNotNull(mView);
        assertNotNull(mButton);
        assertEquals(getText(), "From onCreate resume");
}

public void testButton() {
        mActivity.runOnUiThread(new Runnable() {
                public void run() {
                        mButton.performClick();
                }
        });
        mInstrumentation.waitForIdleSync();
        assertEquals(getText(), "From onCreate resume click");
}
```

Tests

```
public void testPause() {
    mActivity.runOnUiThread(new Runnable() {
        public void run() {
            mInstrumentation.callActivityOnPause(mActivity);
        }
    });
    mInstrumentation.waitForIdleSync();
    assertEquals(getText(), "From onCreate resume pause");
}

public void testMenu() {

    final boolean didMenuRun = mInstrumentation.invokeMenuActionSync(
                mActivity, Menu.FIRST, 1);
    assertTrue(didMenuRun);

    assertEquals(getText(), "From onCreate resume Dad");
}
```

Debugging

Debugging with Toast

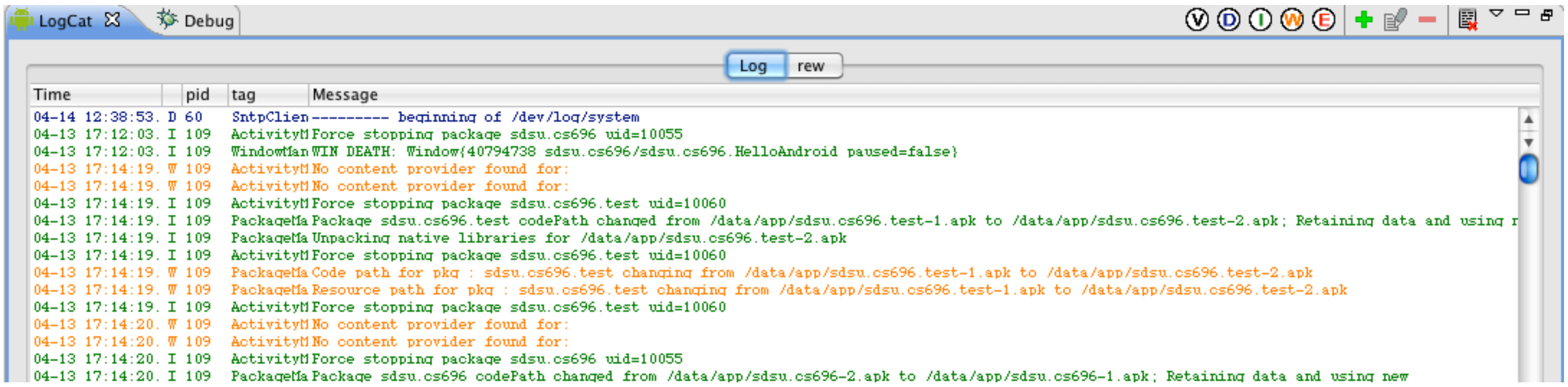
```
Toast.makeText(this,"Message",LENGTH_LONG).show();
```

Log statements

Log sends messages to LogCat

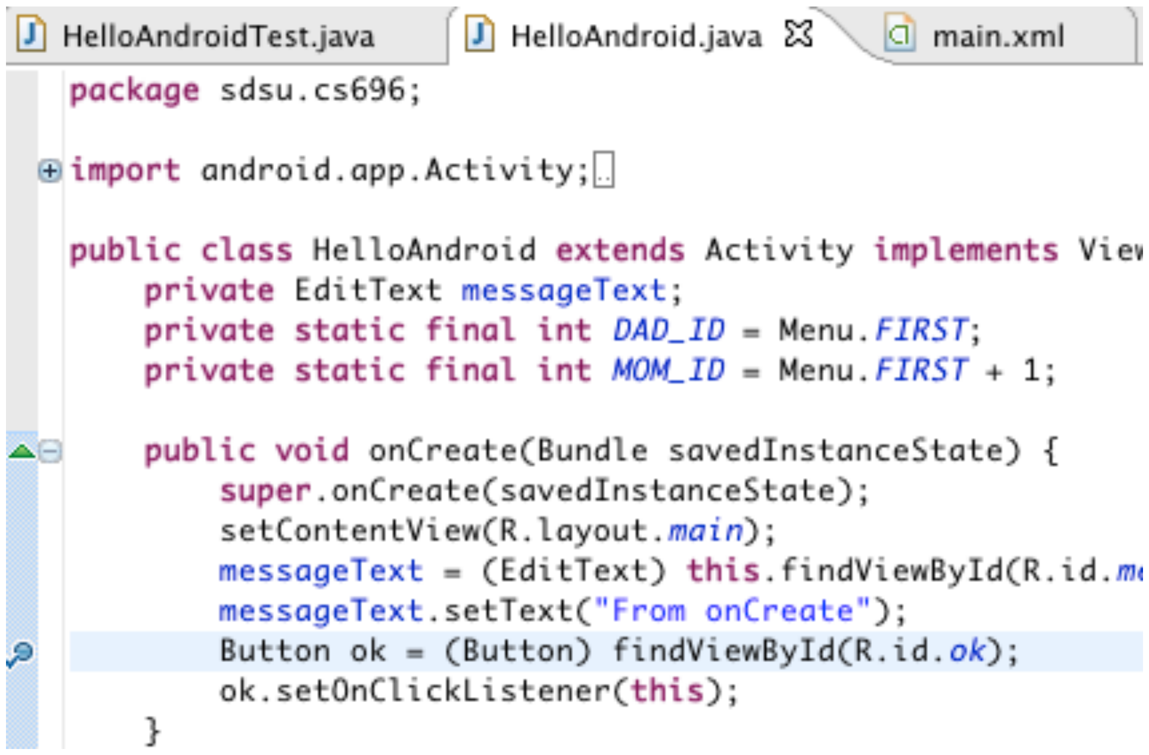
Log.v(String, String) (verbose)
Log.d(String, String) (debug)
Log.i(String, String) (information)
Log.w(String, String) (warning)
Log.e(String, String) (error)

```
Log.i("rew", "Made it to line 12");
```



```
Time              pid  tag        Message
04-14 12:38:53. D 60   SntpClien --------- beginning of /dev/log/system
04-13 17:12:03. I 109  ActivityM Force stopping package sdsu.cs696 uid=10055
04-13 17:12:03. I 109  WindowMan WIN DEATH: Window{40794738 sdsu.cs696/sdsu.cs696.HelloAndroid paused=false}
04-13 17:14:19. W 109  ActivityM No content provider found for:
04-13 17:14:19. W 109  ActivityM No content provider found for:
04-13 17:14:19. I 109  ActivityM Force stopping package sdsu.cs696.test uid=10060
04-13 17:14:19. I 109  PackageMa Package sdsu.cs696.test codePath changed from /data/app/sdsu.cs696.test-1.apk to /data/app/sdsu.cs696.test-2.apk; Retaining data and using r
04-13 17:14:19. I 109  PackageMa Unpacking native libraries for /data/app/sdsu.cs696.test-2.apk
04-13 17:14:19. I 109  ActivityM Force stopping package sdsu.cs696.test uid=10060
04-13 17:14:19. W 109  PackageMa Code path for pkg : sdsu.cs696.test changing from /data/app/sdsu.cs696.test-1.apk to /data/app/sdsu.cs696.test-2.apk
04-13 17:14:19. W 109  PackageMa Resource path for pkg : sdsu.cs696.test changing from /data/app/sdsu.cs696.test-1.apk to /data/app/sdsu.cs696.test-2.apk
04-13 17:14:19. I 109  ActivityM Force stopping package sdsu.cs696.test uid=10060
04-13 17:14:20. W 109  ActivityM No content provider found for:
04-13 17:14:20. W 109  ActivityM No content provider found for:
04-13 17:14:20. I 109  ActivityM Force stopping package sdsu.cs696 uid=10055
04-13 17:14:20. I 109  PackageMa Package sdsu.cs696 codePath changed from /data/app/sdsu.cs696-2.apk to /data/app/sdsu.cs696-1.apk; Retaining data and using new
```

Debugger



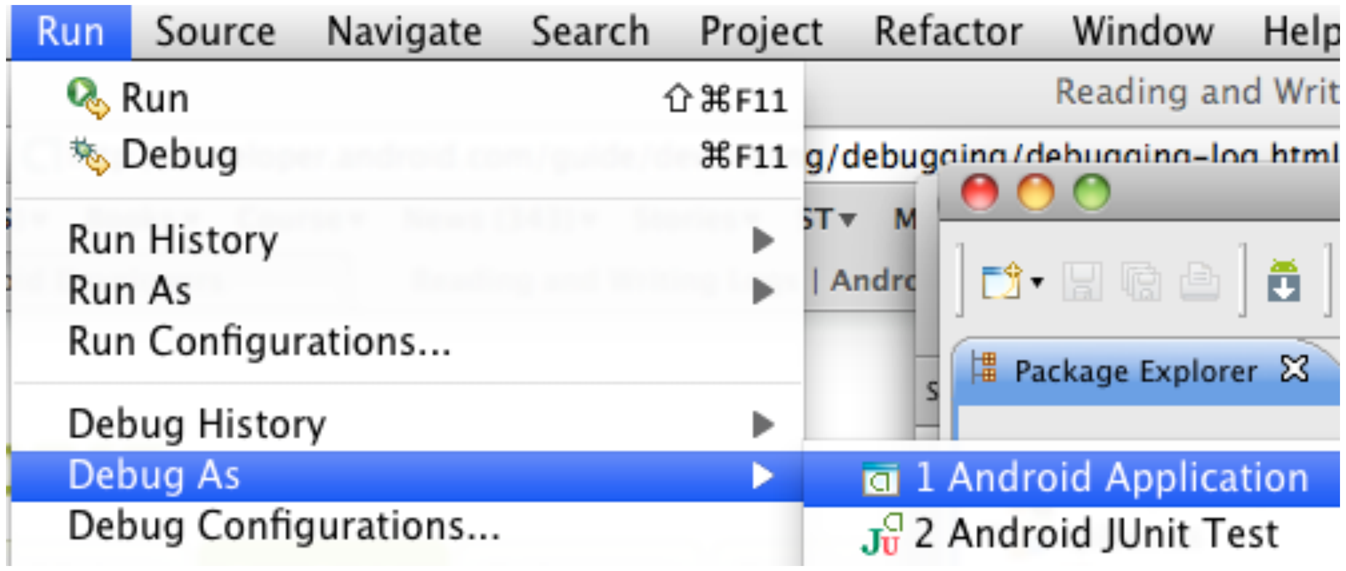
HelloAndroidTest.java | HelloAndroid.java | main.xml

```
package sdsu.cs696;

import android.app.Activity;

public class HelloAndroid extends Activity implements View
    private EditText messageText;
    private static final int DAD_ID = Menu.FIRST;
    private static final int MOM_ID = Menu.FIRST + 1;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        messageText = (EditText) this.findViewById(R.id.me
        messageText.setText("From onCreate");
        Button ok = (Button) findViewById(R.id.ok);
        ok.setOnClickListener(this);
    }
```



Monkey Testing

Generates random events for your activity

Enters text

Click buttons

Selects menus

Rotates screen

etc.

Sample Run

```
Al pro 23->adb shell monkey -p sdsu.cs696 500
Events injected: 500
## Network stats: elapsed time=7681ms (0ms mobile, 7681ms wifi, 0ms not connected)
```

Verbose Mode

```
Al pro 24->adb shell monkey -p sdsu.cs696 -v 10
:Monkey: seed=0 count=10
:AllowPackage: sdsu.cs696
:IncludeCategory: android.intent.category.LAUNCHER
:IncludeCategory: android.intent.category.MONKEY
// Event percentages:
//   0: 15.0%
//   1: 10.0%
//   2: 15.0%
//   3: 25.0%
//   4: 15.0%
//   5: 2.0%
//   6: 2.0%
//   7: 1.0%
//   8: 15.0%
:Switch: 
#Intent;action=android.intent.action.MAIN;category=android.intent.category.LAUNCHER;
Flags=0x10000000;component=sdsu.cs696/.HelloAndroid;end
    // Allowing start of Intent { act=android.intent.action.MAIN cat=
[android.intent.category.LAUNCHER] cmp=sdsu.cs696/.HelloAndroid } in package sdsu.c
```

Interface Testing

Hierarchy Viewer

http://developer.android.com/guide/developing/debugging/debugging-ui.html

located in <sdk>/tools

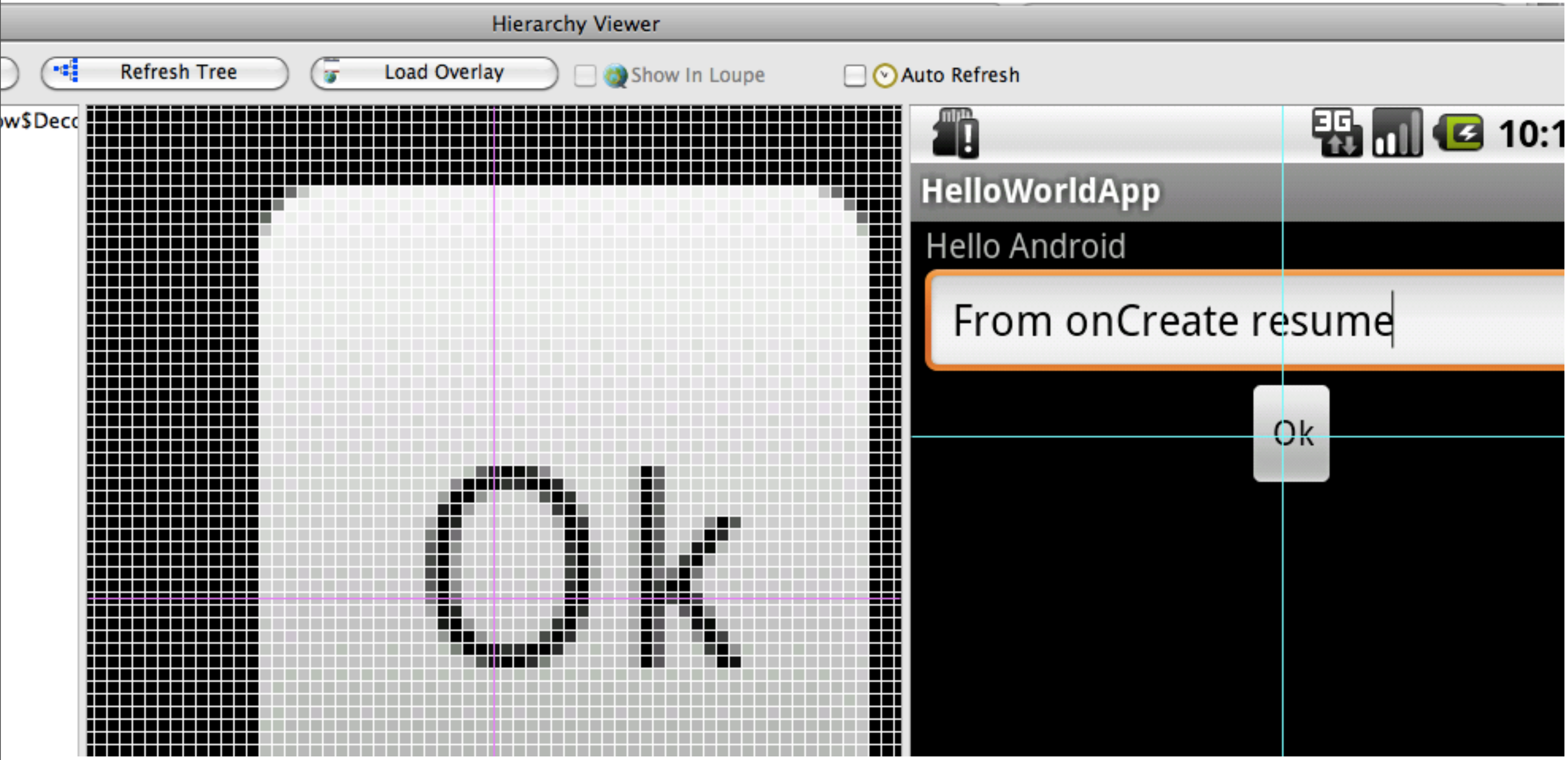
Pixel Perfect Window

- View UI at pixel level

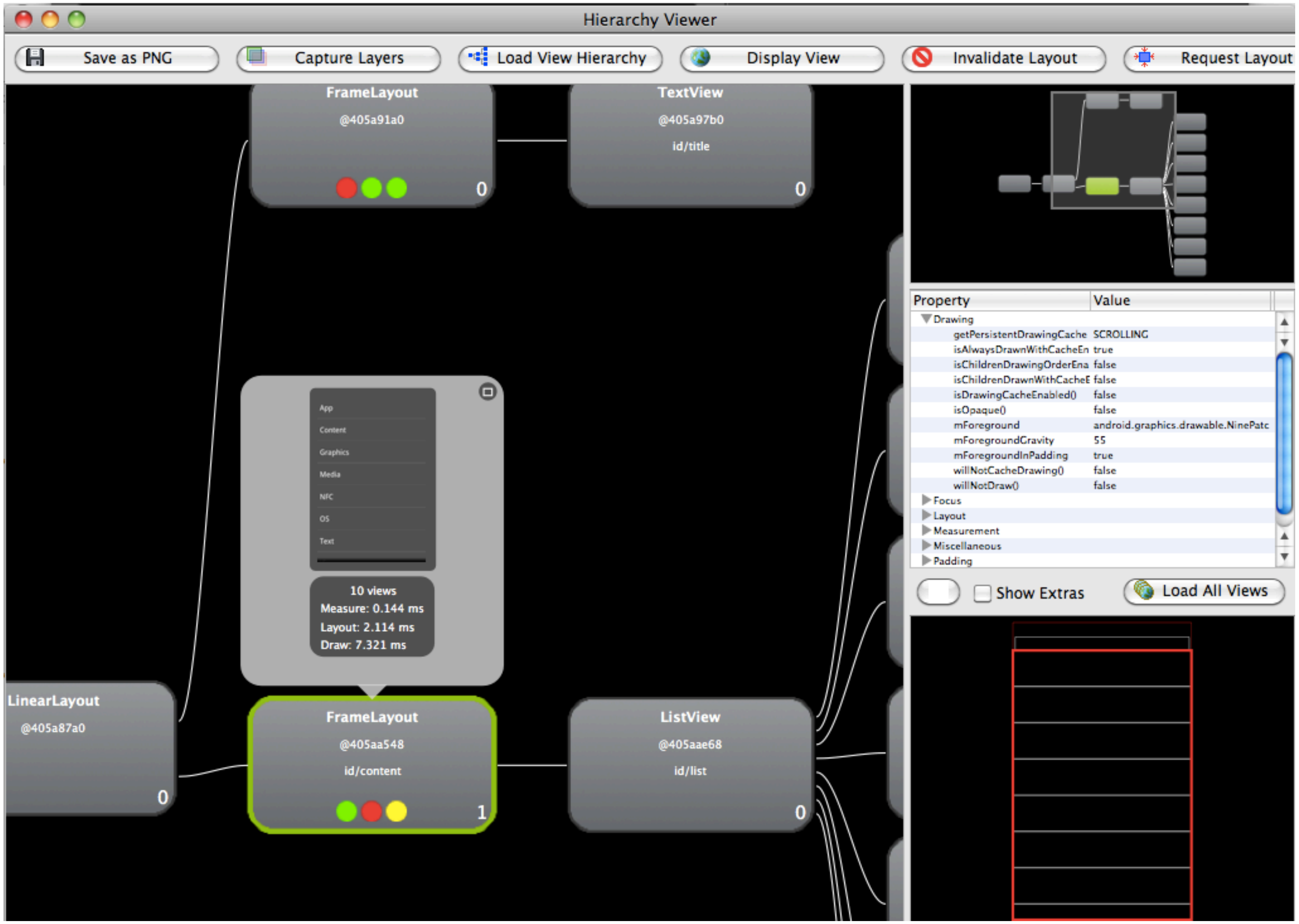
View Hierarchy Window

- View hierarchy structure of UI
- See all view properties
- Measure render time of each screen

Pixel Perfect Window

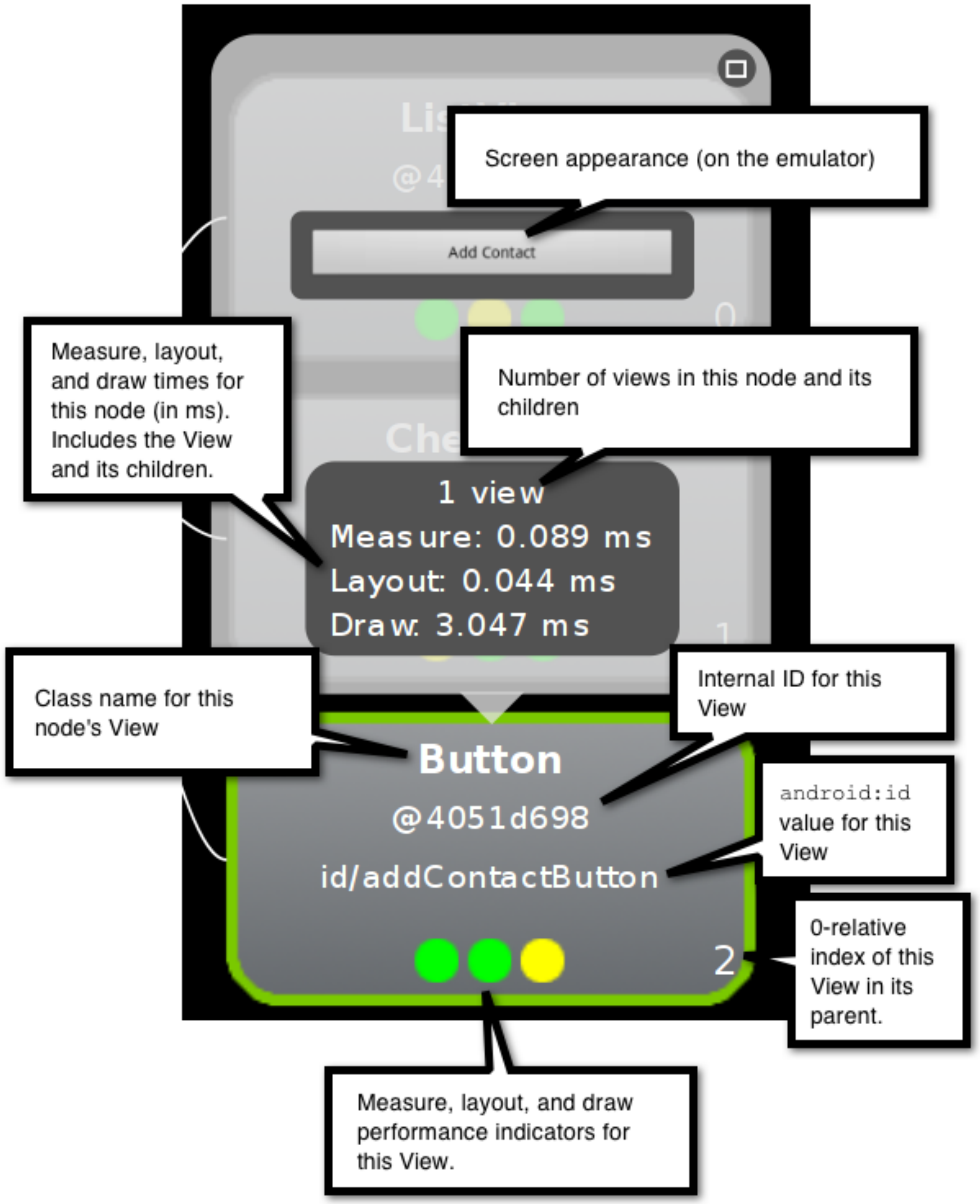


View Hierarchy Window



Property	Value
▼ Drawing	
getPersistentDrawingCache	SCROLLING
isAlwaysDrawnWithCacheEn	true
isChildrenDrawingOrderEna	false
isChildrenDrawnWithCacheE	false
isDrawingCacheEnabled()	false
isOpaque()	false
mForeground	android.graphics.drawable.NinePatc
mForegroundGravity	55
mForegroundInPadding	true
willNotCacheDrawing()	false
willNotDraw()	false
▶ Focus	
▶ Layout	
▶ Measurement	
▶ Miscellaneous	
▶ Padding	

Info Key


Add Contact
Screen appearance (on the emulator)
Measure, layout, and draw times for this node (in ms). Includes the View and its children.
Number of views in this node and its children
1 view
Measure: 0.089 ms
Layout: 0.044 ms
Draw: 3.047 ms
Class name for this node's View
Internal ID for this View
Button
@4051d698
id/addContactButton
android:id value for this View
2
0-relative index of this View in its parent.
Measure, layout, and draw performance indicators for this View.

layoutopt

Finds inefficiencies in the view hierarchy in xml layout files

<sdk>/tools/layoutopt <xmlFiles>

```
Al pro 33->layoutopt /Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/*
/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/activity_animation.xml
	28:28 Use an android:layout_height of 0dip instead of wrap_content for better performance
/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/alarm_controller.xml
	28:28 Use an android:layout_height of 0dip instead of wrap_content for better performance
/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/alarm_service.xml
	28:28 Use an android:layout_height of 0dip instead of wrap_content for better performance
/Java/android-sdk-mac_x86/samples/android-10/ApiDemos/res/layout/alert_dialog.xml
	25:50 This LinearLayout tag should use android:layout_height="wrap_content"
```