Functional Testing

• ActivityInstrumentationTestCase2
  – Allows us to create/start an Activity
  – Get Views from the Activity (e.g. Buttons)
  – Run things on the UI thread (e.g. click Buttons)
  – Perform asserts in JUnit

• Other options
  – http://code.google.com/p/autoandroid/
  – Formerly Positron
  • Android + Selenium = Positron
public class FunctionalTest extends ActivityInstrumentationTestCase2<AdvancedJokeList> {

    public FunctionalTest() {
        super("edu.calpoly.android.lab2", AdvancedJokeList.class);
    }

    protected void setUp() throws Exception {
        super.setUp();
    }

    public void testAddJoke() {
        ArrayList<Joke> m_arrJokeList = null;
        m_arrJokeList = this.retrieveHiddenMember("m_arrJokeList",
                m_arrJokeList, getActivity());

        assertEquals("Should be 3 default jokes", m_arrJokeList.size(), 3);
        getActivity().runOnUiThread(new Runnable() {
            public void run() {
                AdvancedJokeList theActivity = (AdvancedJokeList) getActivity();
                EditText et = (EditText) theActivity.findViewById(edu.calpoly.android.lab2.R.id.
                        newJokeEditText);
                Button bt = (Button) theActivity.findViewById(edu.calpoly.android.lab2.R.id.
                        addJokeButton);
                et.setText("This is a test joke");
                bt.performClick();
            }
        });

        getInstrumentation().waitForIdleSync(); // wait for the request to go through
        assertEquals("Should be 4 jokes now", m_arrJokeList.size(), 4);
        assertEquals("Ensure the joke we added is really there",
                m_arrJokeList.get(3).getJoke(), "This is a test joke");
    }
}
@SuppressWarnings("unchecked")
public <T> T retrieveHiddenMember(String memberName, T type, Object sourceObj) {
    Field field = null;
    T returnVal = null;
    try {//Test for proper existence
        field = sourceObj.getClass().getDeclaredField(memberName);
    } catch (NoSuchFieldException e) {
        fail("The field " + memberName + " was renamed or removed. Do not rename or remove this member variable.");
    }
    field.setAccessible(true);
    try {//Test for proper type
        returnVal = (T)field.get(sourceObj);
    } catch (ClassCastException exc) {
        fail("The field " + memberName + " had its type changed. Do not change the type on this member variable.");
    }
}
// Boiler Plate Exception Checking. If any of these Exceptions are thrown it was because this method was called improperly.
catch (IllegalArgumentException e) {
    fail("This is an Error caused by the UnitTest!
    Improper user of retrieveHiddenMember(...) -- IllegalArgumentException:
    Passed in the wrong object to Field.get(...)"当他);
} catch (IllegalAccessException e) {
    fail("This is an Error caused by the UnitTest!
    Improper user of retrieveHiddenMember(...) -- IllegalAccessException:
    Field.setAccessible(true) should be called.");
}
return returnVal;
Monkey

• Random stress testing
  – From http://d.android.com/guide/developing/tools/monkey.html
  – When the Monkey runs, it generates events and sends them to the system. It also watches the system under test and looks for three conditions, which it treats specially:
    • If you have constrained the Monkey to run in one or more specific packages, it watches for attempts to navigate to any other packages, and blocks them.
    • If your application crashes or receives any sort of unhandled exception, the Monkey will stop and report the error.
    • If your application generates an application not responding error, the Monkey will stop and report the error.

adb shell monkey -p edu.calpoly.lab2 -v 500
TDD and Android Resources

• Android SDK documentation

• Tutorial:

• Blogs:
  – http://dtmilano.blogspot.com/search/label/test%20drive%n%20development