STAR WEST
September 30–October 5, 2012
Anaheim, CA • Disneyland Hotel
SOFTWARE TESTING ANALYSIS & REVIEW

Innovative Tools for Your Mobile Testing Toolkit

Eing Ong

Staff Software Engineer in Quality, Intuit, Inc.

Session outline

- Test automation technologies
- Innovative technologies
  - User behavior – Sikuli
  - Device SDK tool - MonkeyRunner
  - Image processing – ImageMagick
  - Design patterns – MOET
- Demo
- Q & A

Native mobile apps automation

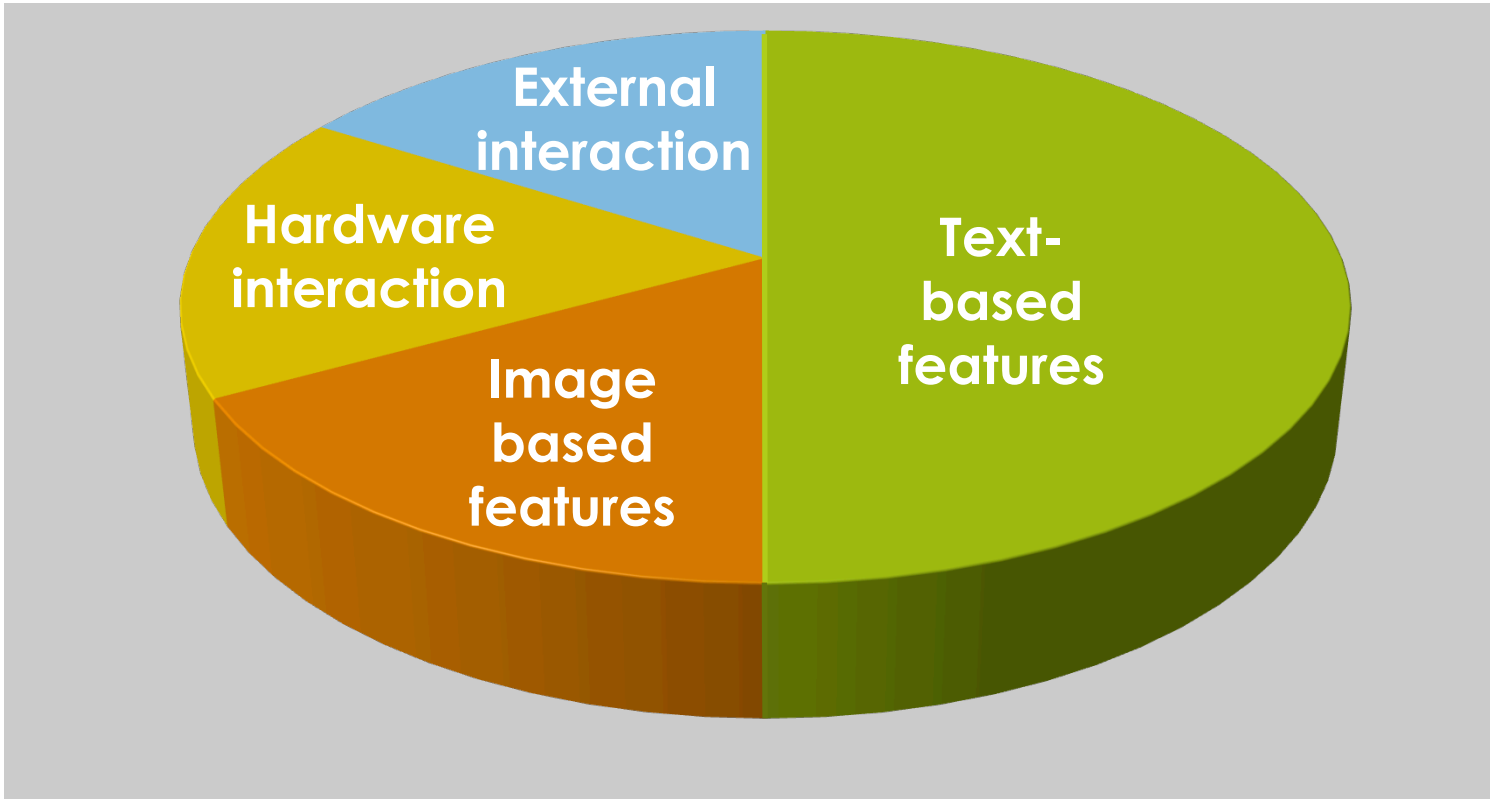
- **Two categories**
  - Instrumented technique
  - Non-instrumented technique
- **What is instrumentation**
  - Test project is compiled with the app
  - Tests are installed and launched with the app
  - Source code is required and may be modified
  - Only one application can be tested at a time
  - White box approach

Advantages of both techniques

Non-instrumentation	Instrumentation
• Device platform agnostic • Test code reuse • Test language and test harness autonomy • Support for *Multi-applications testing* *Custom UI elements* *Database/API assertions* *Use of external libraries (e.g. image manipulation)*	• Elements can be accessed • Debugging ease • Test verification ease • Reduce tools dependencies • Support for *Installing application* *Launching application* *Kill application* *Test execution on device* *Code coverage*

Which technique should I use ?

Non-instrumentation **Instrumentation**



Open sourced tools

Mobile OS	Non-instrumentation	Instrumentation
Android	Sikuli, MOET, MonkeyRunner	Robotium, MonkeyTalk, Calabash
iOS	Sikuli, MOET	KIF, UISpec, Frank, Calabash, iCuke, Zucchini, Bwoken, MonkeyTalk

Sikuli : User behavior innovation

- Visual technology to automate and test GUI using images – *sikuli.org*
- Platform and OS agnostic
  - Controls on desktop
  - Controls mobile simulators and devices (via VNC)
- Actions
  - Gestures, keystrokes
  - Captures screenshots
  - Detects screen changes
  - Finds image, image OCR

Customizing Sikuli for iPhone

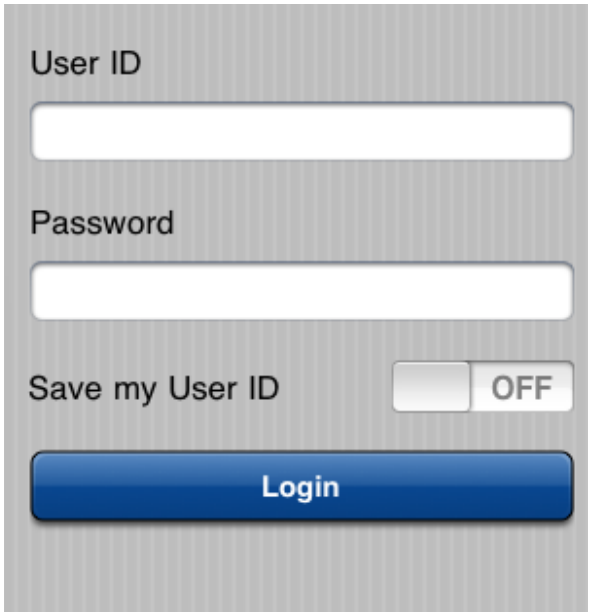
Customized API	Sikuli API
touch(int x, int y)	click(new Location(x,y), 0)
enter(String str)	type(null, str, 0)
touchImage(String im)	click("homebutton.png", 0)
screenshot(String file)	capture(region.getRect())
getText()	region.text();

home()	keyDown(Key.SHIFT);
	keyDown(Key.META);
	keyDown("h");
	keyUp(Key.META);
	keyUp(Key.SHIFT);
	keyUp("h");

Example

```
import com.intuit.moet.iPhone;

void login (username, password)
    touch ("30%", "40%");
    backspaces (20);
    enter(username);
    touchImage(iPhone.KeyNext);
    enter(password);
    touchImage(iPhone.KeyGo);
```



Customized APIs from *com.intuit.moet.iPhone.java (github.com/eing/moet)*

MonkeyRunner : Device SDK tool innovation

- API for controlling an Android device/emulator outside of Android code - *developer.android.com*
- Actions
  - Multiple emulators & devices interaction
  - Captures screenshots
  - Send keystrokes, gestures
  - Start/stop/reconnect emulator
  - Configure logging

Using MonkeyRunner

Customized API	MonkeyRunner API
touch(int x, int y)	touch(x, y, TouchPressType.DOWN_AND_UP);
backspaces(int num)	while (num > 0) device.press(KEYCODE_DEL, …); num--;
screenshot(String file)	image = takeSnapshot(); image.writeToFile(filename, "png");
home()	press(KEYCODE_HOME, TouchPressType.DOWN_AND_UP);
launch(String activity)	shell(" am start -n " + activity);

Customized APIs from *com.intuit.moet.Android.java (github.com/eing/moet)*

ImageMagick ® : Image processing innovation

- Software suite to create, edit, compose, or convert bitmap images – *imagemagick.org*
- Actions
  - Resize, flip, mirror, rotate, distort
  - Transform images, adjust image colors
  - Format conversion, draw, captions
- Supports
  - Windows, Unix, Mac, and iOS
  - C, C++, Perl APIs

Customizing ImageMagick for mobile

- crop (String crop, int resX, int resY)
  crop("200x100+10+10", 480, 800)
  crop("50%x100%)
  crop("50%x50%+10%+10%", 320, 480)
- boolean compare(String image, int tolerance)
  compare("HomeScreen.png", 200)
- boolean waitForScreenChange(String image)
  waitForScreenChange("CurrentImage.png")

Customized APIs from com.intuit.moet.ImageKit.java

MOET : Design patterns innovation

- *Think Design*
  - Interfaces
  - Creational pattern
- *Think Reuse*
  - Device independent tests
- *Think One*
  - Test language & harness

Test

login('user1','passwd1')

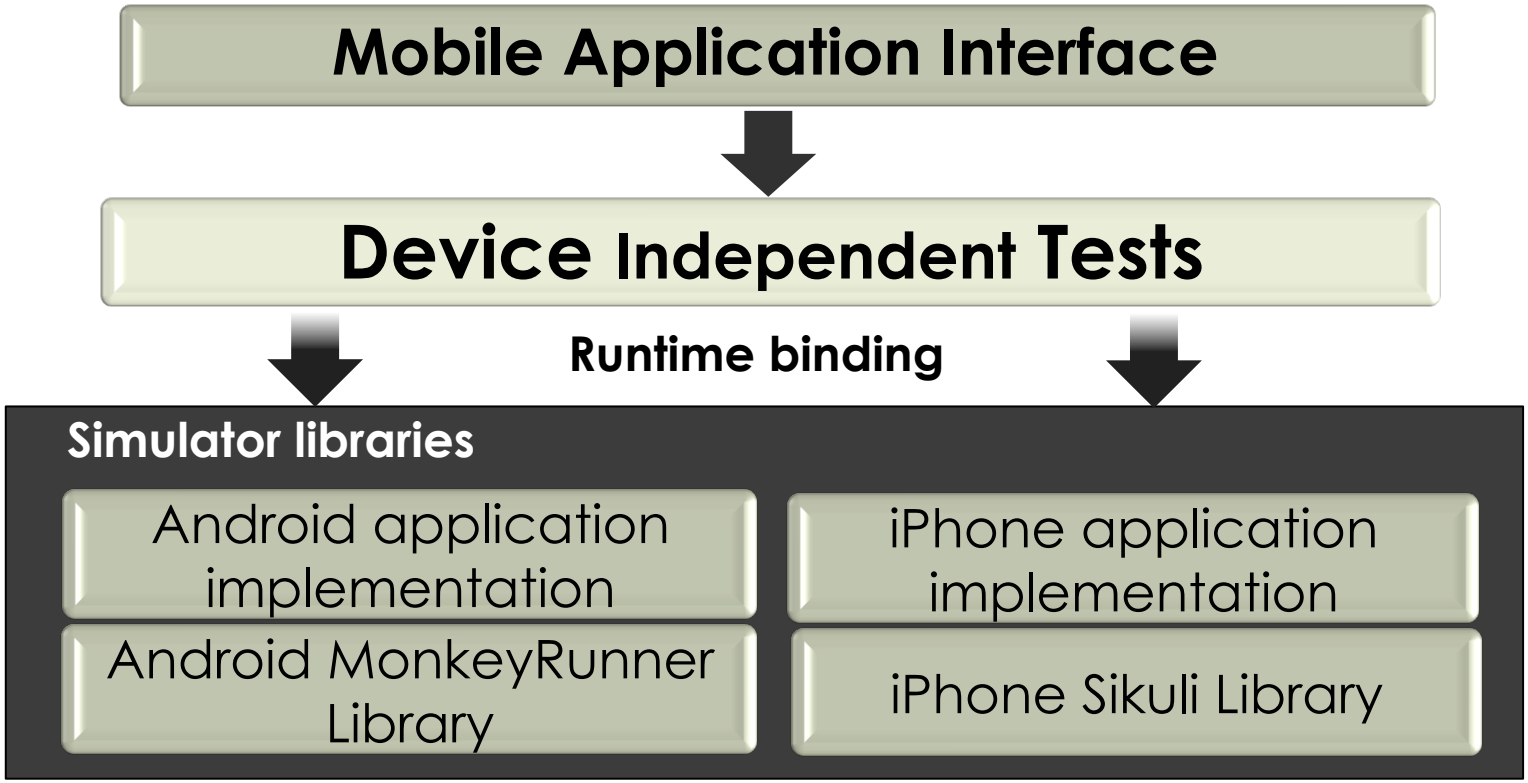
iPhone implementation

touch(100,100)
enter(username)
touch(100,200)
enter(password)
touch(150, 300)

iPhone library

void enter()
void touch(x,y)

Test architecture using MOET



Add contact test

```
class AddContactTest()  :
    public void addContactWithOnlyFirstnameTest() {
        Contact contact = new Contact(firstname, null);
        assertTrue(device.addContact(contact));
     }
    public void addContactWithOnlyLastnameTest() {
        Contact contact = new Contact(null, lastname);
        assertTrue(device.addContact(contact));
     }
```

Android implementation

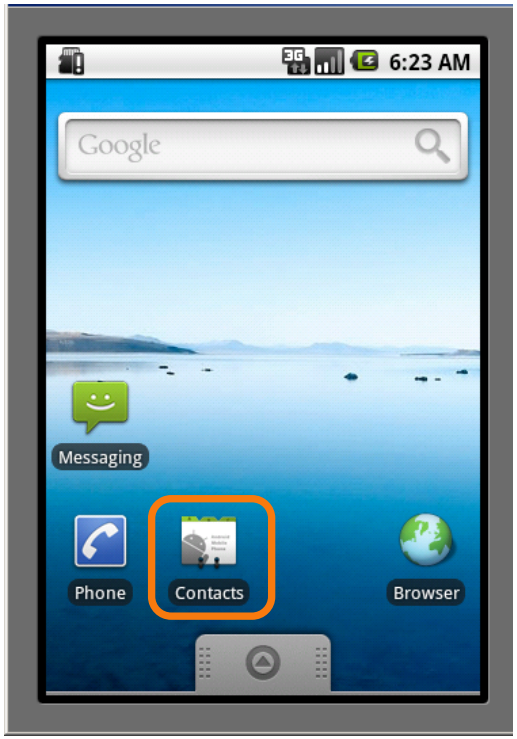
```
public class AndroidImpl implements IAddressBookApp  {
        private Android device;
        public boolean addContact(Contact contact) {
                device.menu();
                device.scroll("down");
                device.enter(contact.getFirstname());
                device.scroll("down");
                device.enter(contact.getLastname());
                … enter other fields…
                device.enter();
```

iPhone implementation

```
public class iPhoneImpl implements IAddressBookApp  {
        private iPhone device;
        public boolean addContact(Contact contact) {
                device.touch("First");
                device.enter(contact.getFirstname())
                device.touch("Last");
                device.enter(contact.getLastname());
                device.scroll("down")
                … enter other fields …
                device.touchImage("ButtonDone.png");z
```

Demo

- Test reused and executed on Android & iOS
- Using Sikuli, ImageMagick, MOET and MonkeyRunner on Android
- Automate address book app
  - Add contact
  - Find contact
  - Delete contact



Resources

- **Sikuli** http://sikuli.org
- **MonkeyRunner** http://developer.android.com/tools/help/index.html
- **ImageMagick** http://www.imagemagick.org
- **MOET** https://github.com/eing/moet
- **Other open sourced tools grouped by language**
  - ObjectiveC - KIF
  - Java - Robotium, MonkeyRunner
  - Ruby - UISpec
  - Coffeescript - Zucchini, Bwoken
  - Cucumber - Calabash, iCuke, Frank
  - MonkeyTalk - MonkeyTalk

Q & A

Thank you

- Thank you for submitting your feedback !
- For more details on the presentation, please contact @eingong / eing.ong@intuit.com