Mobile Apps 2010
iPhone and Android

March 9, 2010

Norman McEntire, Founder
Servin Corporation - http://servin.com
Technology Training for Technology Professionals™
norman.mcentire@servin.com

Legal Info

- Android is a trademark of Google Inc.
- iPhone is a trademark of Apple Inc.
- Servin is a trademark of Servin Corporation.

About Servin Corporation

- Servin On-Site Training
  - “Get Your Development Team Up To Speed Fast!”
- Servin Web-Based Training
  - “Watch Code Written Before Your Eyes!”
- Servin Books
  - Books available on Amazon Kindle and Amazon Kindle Reader (iPhone, PC, Blackberry)
- Servin Apps
  - 25+ iPhone Apps in various categories

Servin On-Site Training

- "Get Your Development Team Up To Speed Fast!"
- iPhone Programming Courses
  - Intro and Advanced
- Android Programming Courses
  - Intro and Advanced
- Linux Programming Courses
  - Embedded Linux
  - Device Drivers
  - Services/Daemons

Servin Web-Based Training

- "Watch Code Written Before Your Eyes!"
  - Uses Cisco WebEx Player
  - Cicso WebEx Player supported on
    - Linux
    - Mac
    - Windows
- This demo/presentation is also available as Web-Based Training
  - http://servin.com

Servin Books

- Books available on Amazon Kindle and Amazon Kindle Reader (iPhone, PC, Blackberry)
  - iPhone Objective-C 2.0 Programming Exercises
  - iPhone Database Programming Exercises: SQLite
  - iPhone Camera Programming Exercises
  - iPhone External Accessory Programming Exercises
  - iPhone Flashlight Programming Tutorial
  - iPhone 3 Programming Templates Explained
  - Red Hat Enterprise Linux 5 Admin Skills
  - SUSE Linux Enterprise Server 11 Admin Skills

Servin Apps
25+ iPhone Apps

- Technology Professionals
  - Device Info Plus
    - Device ID Info, Font Info, CPU Info, Network Info, and more!
  - Accelerometer Fun
    - Display, record, and export accelerometer info
  - Process Log
    - Display list of processes
- Health/Fitness
  - Weight Log Simple – keep log of your weight
  - Sleep Log Simple – keep log of your sleep
- Games and Entertainment
  - Touch Fast – how fast can you touch the screen?
  - Simon Says Face Up – can you do as simon says?

UCSD Extension Courses

- These courses are taught by Norman McEntire and fill up quickly so register early
  - iPhone Programming: Touch, Sound, and More!
    - Spring 2010 Session: April 7 – June 2
    - Wednesdays, 6PM - 9PM
  - Introduction to Objective-C
    - Spring 2010 Session: April 8 – June 3
    - Thursdays, 7pm – 9:50pm
  - iPhone Programming: Advanced
    - Summer 2010 Session: Watch Catalog for Dates/Times

Opening Remarks

- Welcome!
- Thank you!
- My Promise To You
  - Show you Mobile App Development 2010!
    - iPhone OS SDK 3.2 Beta 3 (includes iPad)
    - Android SDK 2.1
- My style
    - Show a few slides
    - Do lots of live demos
    - Questions at anytime are great!

PDF Slides and WebEx Recording Will Be Posted on http://servin.com

- My goal is to show you, ***faster than any method on planet earth***, how to use both the iPhone SDK and the Android SDK
  - The tools. The Methods. The Source Code.
- This demo is fast paced but PDF slides and WebEx recording will be posted on http://servin.com
  - The PDF Slides include the source code
  - The WebEx Recording can be paused/replayed

Demo is very focused. Here are the Guideposts

- Guidepost #1
  - Use only the latest native SDKs downloaded March 2010
    - iPhone OS SDK 3.2 Beta 3 (iPhone, iPod Touch, iPad)
    - Android SDK 2.1
- Guidepost #2.
  - Make zero changes or modifications to the native SDKs – use the defaults as they come
- Guidepost #3.
  - Use no 3rd party tools or add-ons – just the SDKs

Briefly...because the question always comes up...

- Question: Are there alternatives to using the native SDKs?
- Answer: Yes. Many alternatives, but that is not the focus of this demo/presentation.
- Some alternatives to native SDKs – great topics for other SDSIC speakers
  - Rhomobile.com - “HTML + Ruby” Native Apps
  - PhoneGap.com - “HTML + JavaScript” Native Apps
  - MonoTouch.net - “C#” Native Apps

Introducing the SDKs

iPhone SDK 3.2 Beta 3

- Hardware Requirements
  - Mac (running Mac OS 10.6 Snow Leopard)
- Membership Requirements
  - To download SDK
    - Register as **Apple Developer**. Free registration.
  - To run on device and submit to store
    - Join iPhone Developer Program. Yearly fee.
- Software Requirements
  - iPhone SDK. Includes everything.

Android SDK 2.1

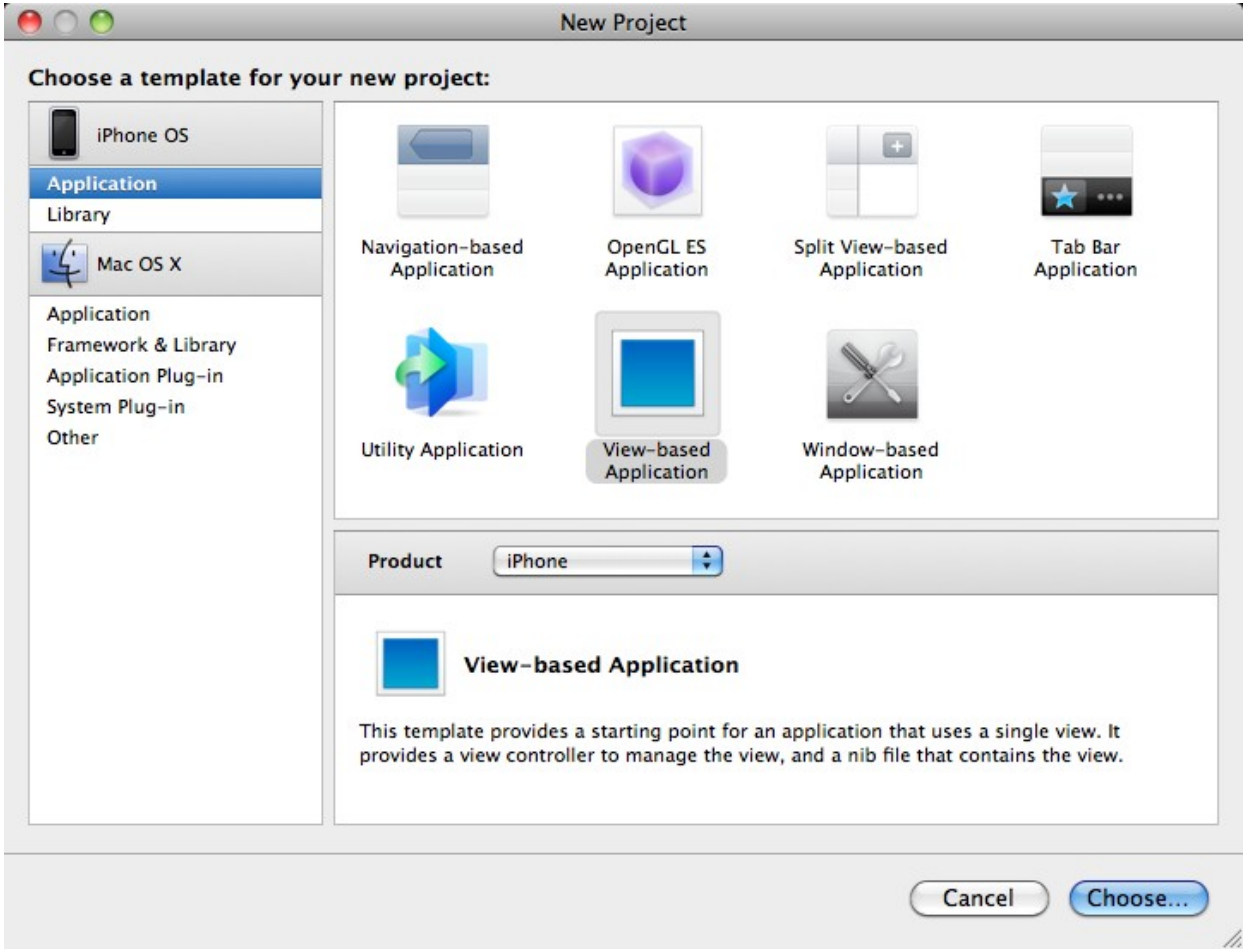
- Hardware Requirements
  - Any Hardware running Linux, Mac, or Windows
- Membership Requirements
  - To download SDK: None
  - To run on device: None
  - To submit to store: Small One-Time Fee
- Software Requirements
  - Java SDK + Java IDE for Eclipse + Android SDK + Android Plugin for Eclipse

Time for the Demos!

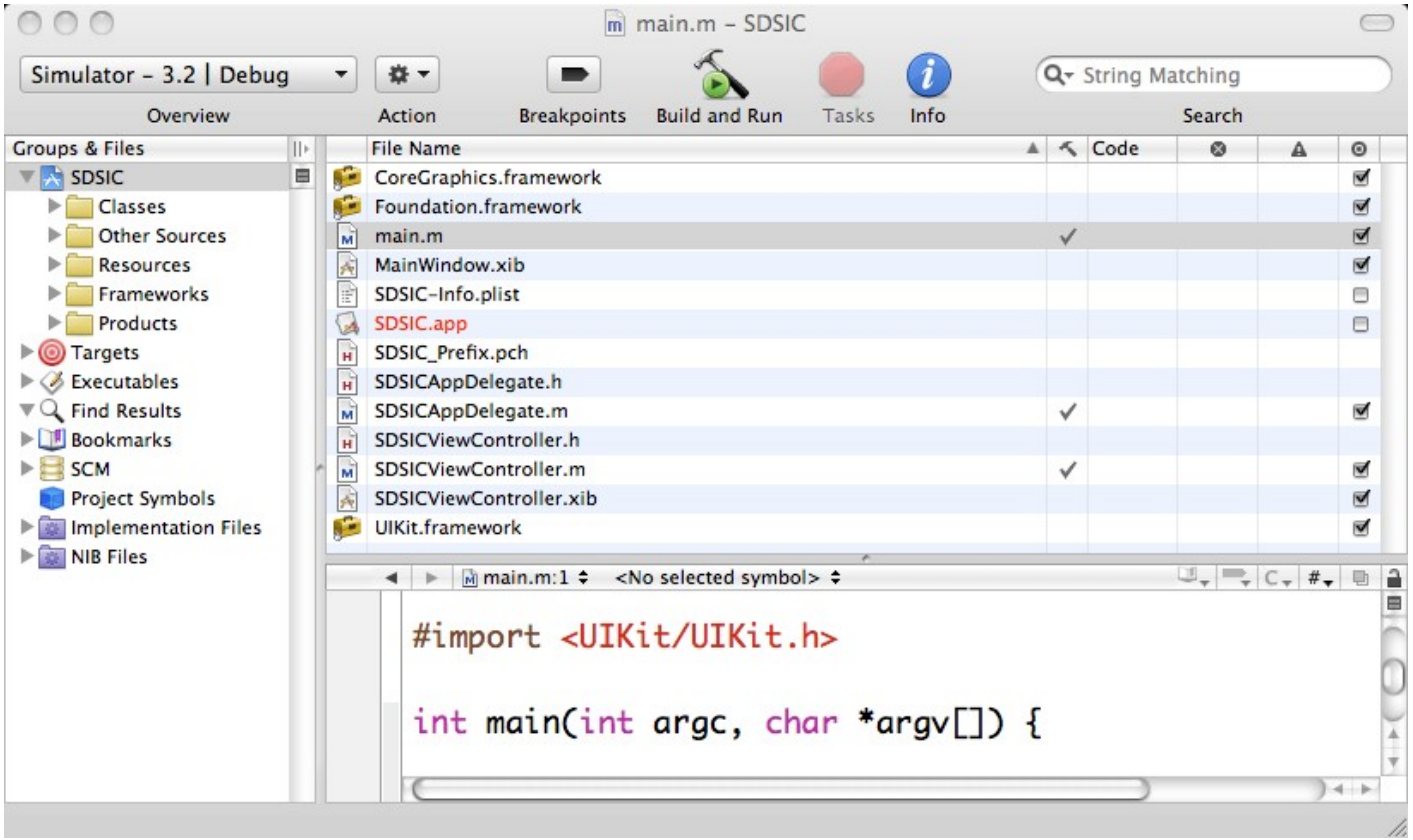
Demo 1

Building the Default App

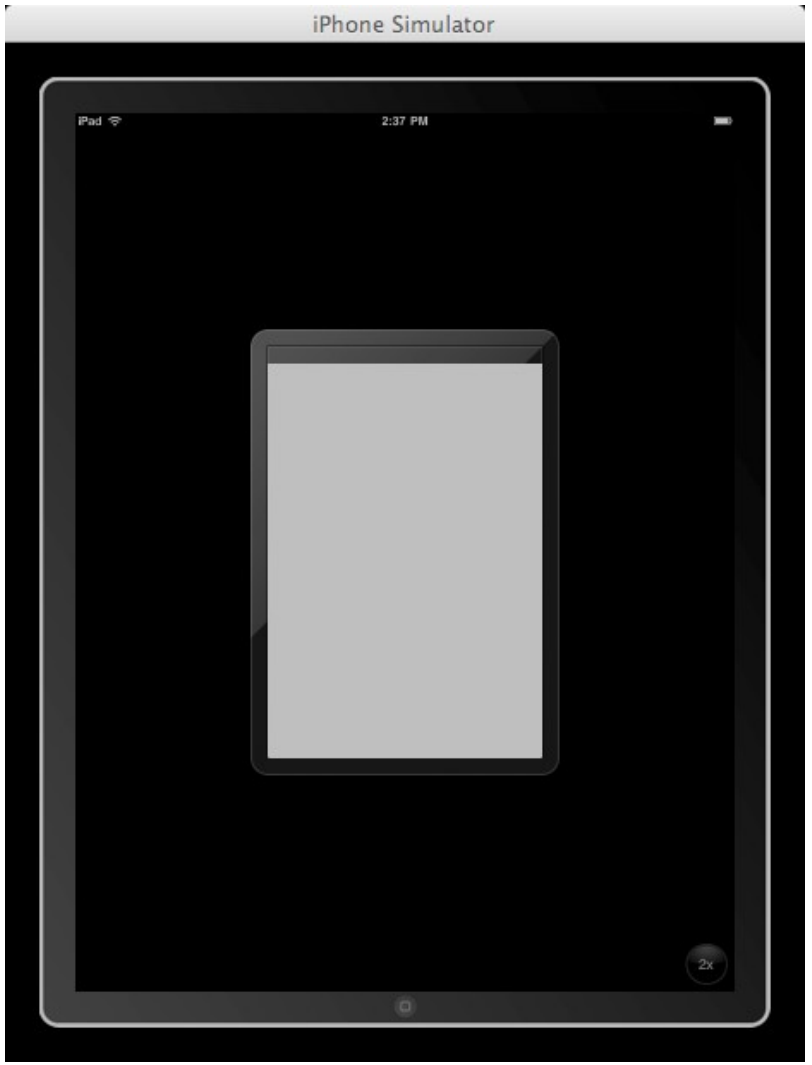
Demo 1: iPhone SDK
Xcode: File > New Project

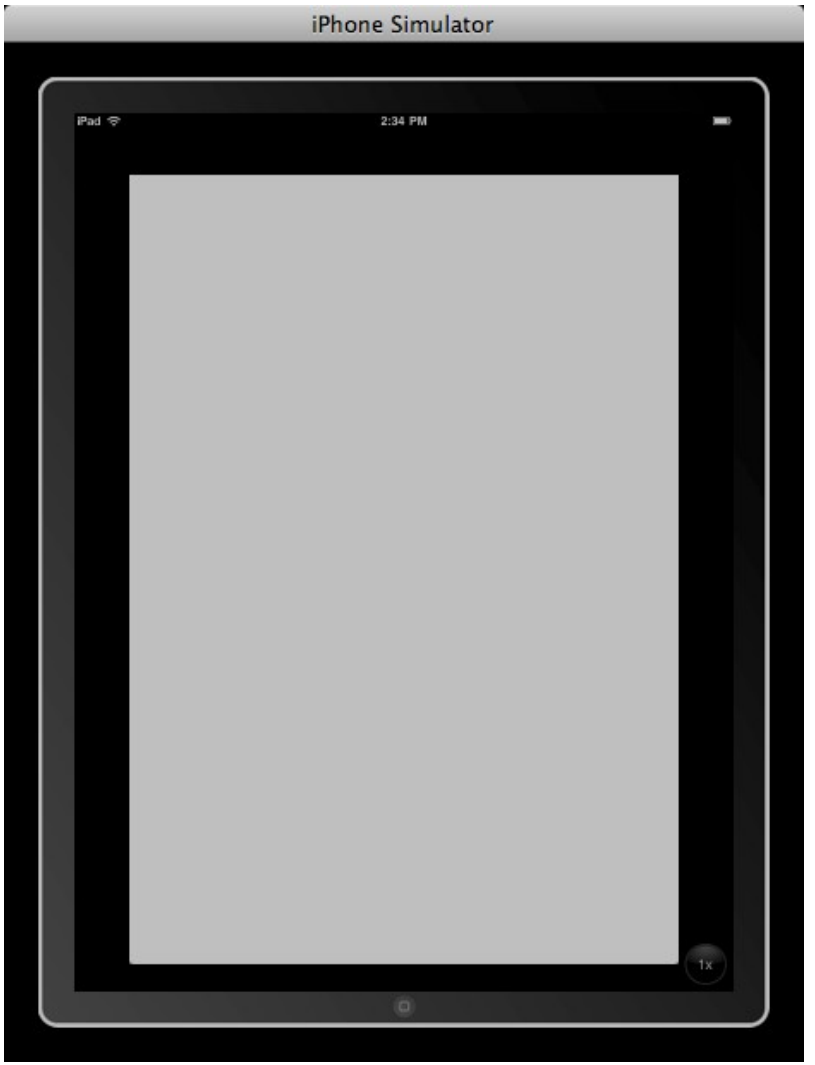


Demo 1: iPhone SDK Xcode IDE



Demo 1: iPhone SDK
iPhone Simulator – iPad Mode





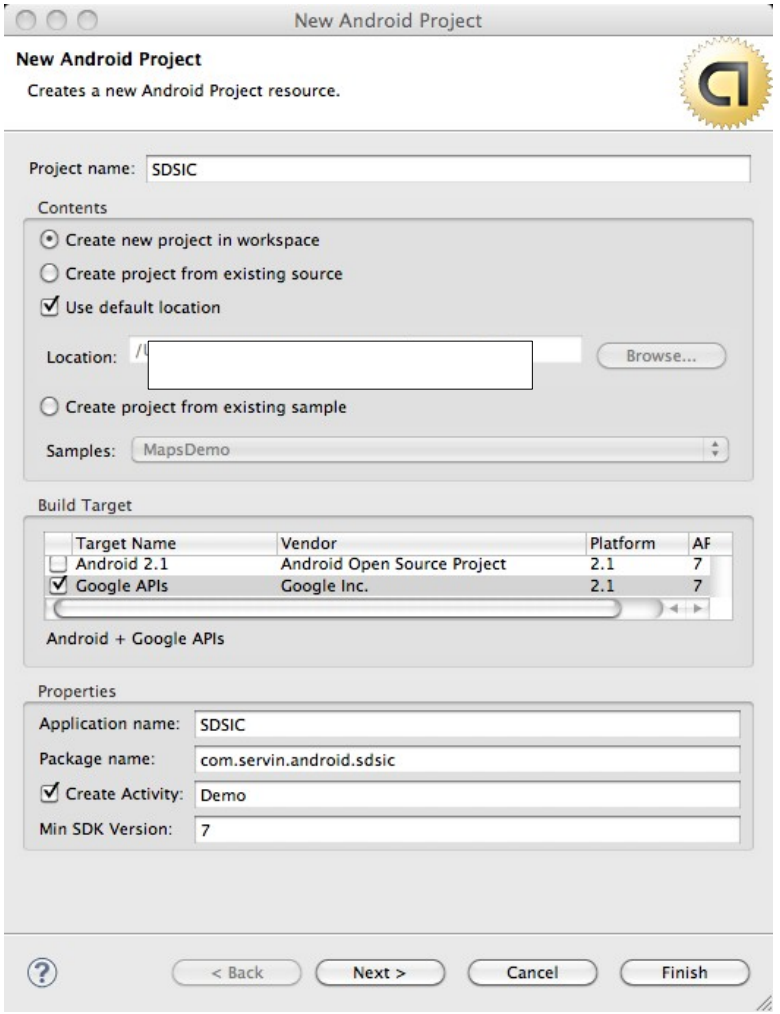
Demo 1: iPhone SDK
iPhone Simulator – iPhone Mode

User Interface Observations

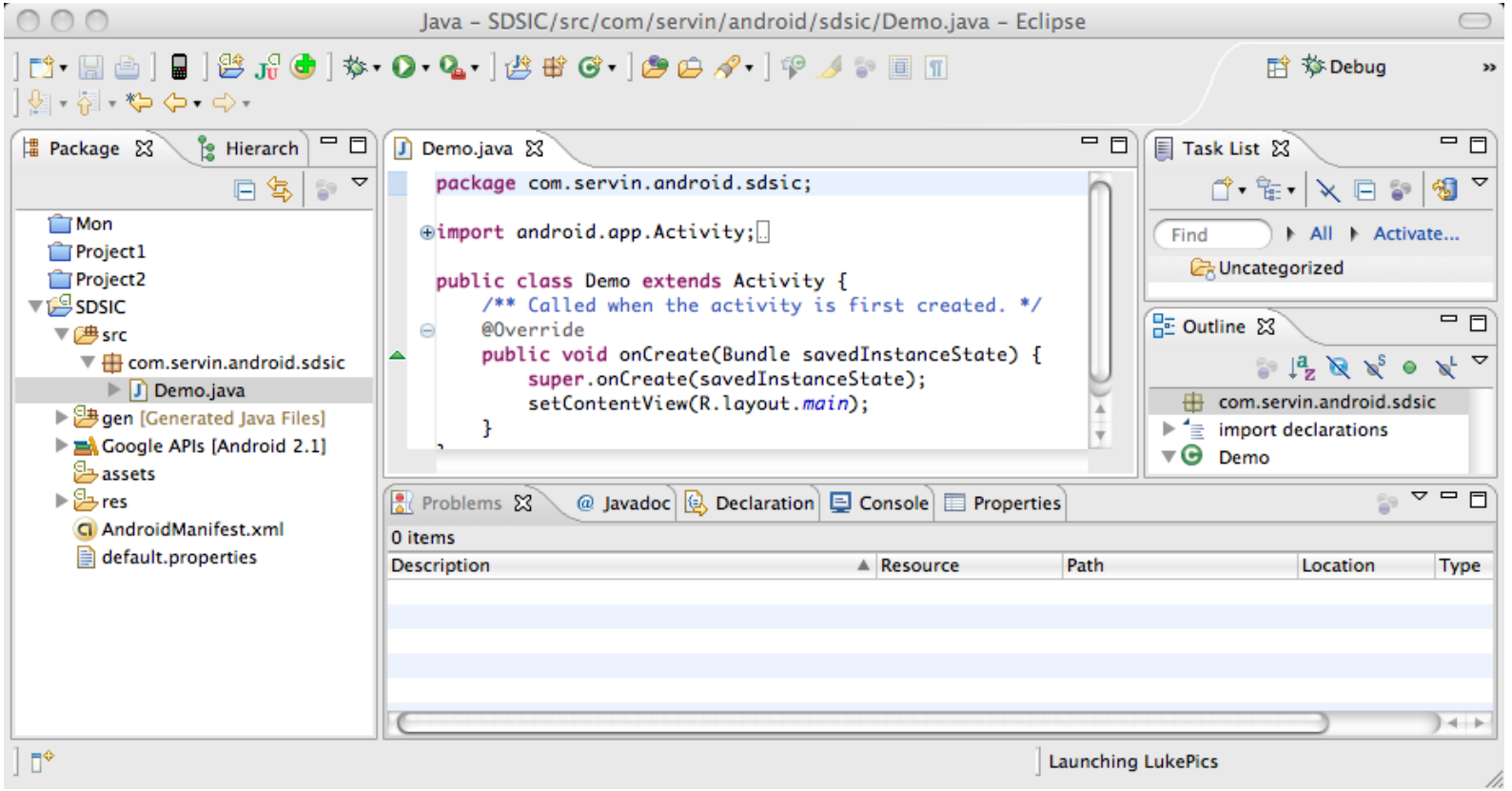
1. Status bar at top

2. Single Home Button at bottom

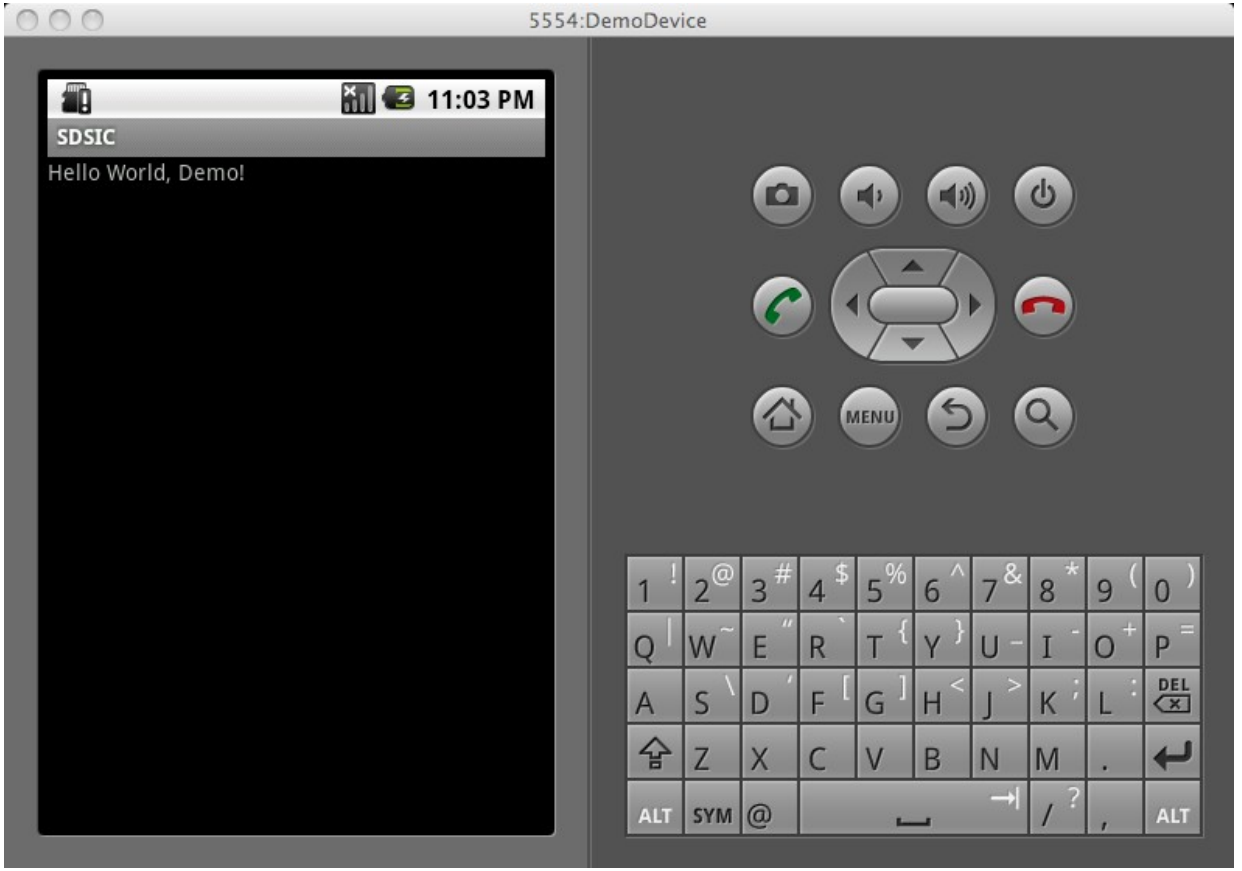
Demo 1: Android SDK
Eclipse: New Android Project



Demo 1: Android SDK Eclipse IDE



Demo 1: Android SDK Android Emulator



User Interface Observations

1. Status bar at top

2. Title Bar

3. Multiple Buttons
- Home
- Menu
- Return
- Search
- D-Pad (Directional Pad)
- Others

4. Keyboard

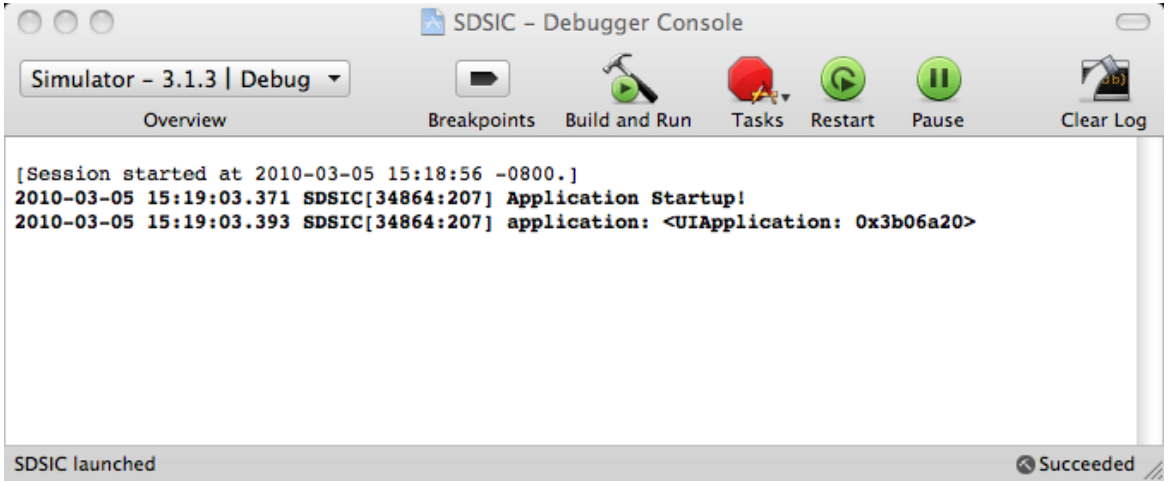
Demo 2

Application Startup and Logging

Demo 2: iPhone SDK Logging: NSLog() - C Function

```
//  SDSICAppDelegate.m

...code not shown...

- (BOOL)application:(UIApplication *)application
    didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

    // Log a startup message
    NSLog(@"Application Startup!");
    NSLog(@"application: %@", [application description]);
    NSLog(@"launchOptions: %@", [launchOptions description]);

    [window addSubview:viewController.view];
    [window makeKeyAndVisible];

    return YES;
}
```

Demo 2: iPhone SDK
Xcode: Run > Console

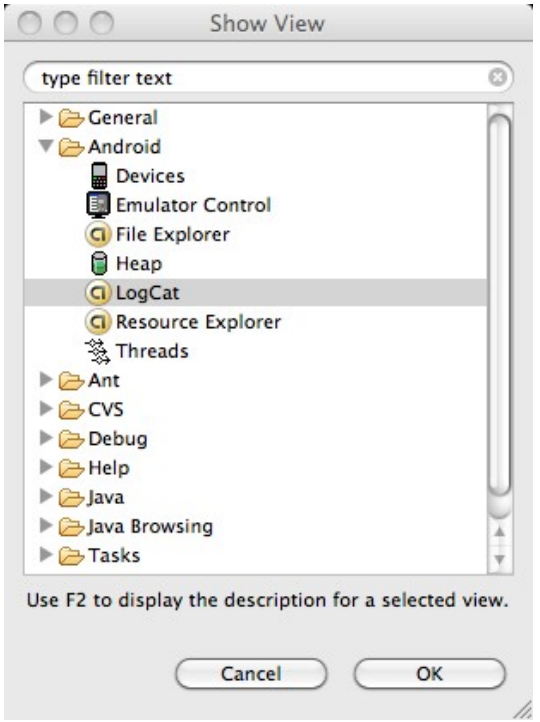


Demo 2: Android SDK Logging: Log.v() - Java Class

This code crashes if savedInstanceState is NULL. That is part of the demo...

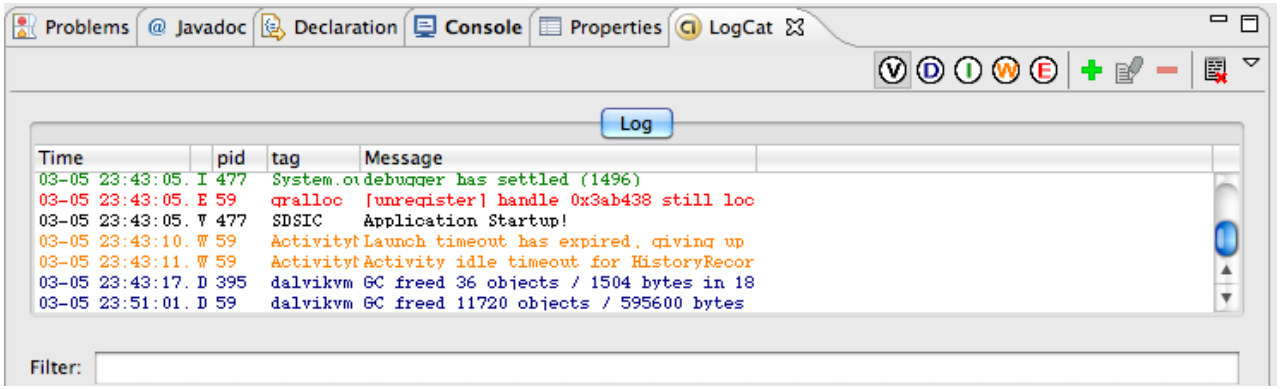
```
// Demo.java

...code not shown...

public class Demo extends Activity {
    private static final String TAG = "SDSIC";

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Log a startup message
        Log.v(TAG,"Application Startup!");
        Log.v(TAG, "savedInstanceState: " +
             savedInstanceState.toString());
        setContentView(R.layout.main);
    }
}
```

Demo 2: Android SDK
Eclipse: Window > Show View > Other



The LogCat view shows messages output from the Log class

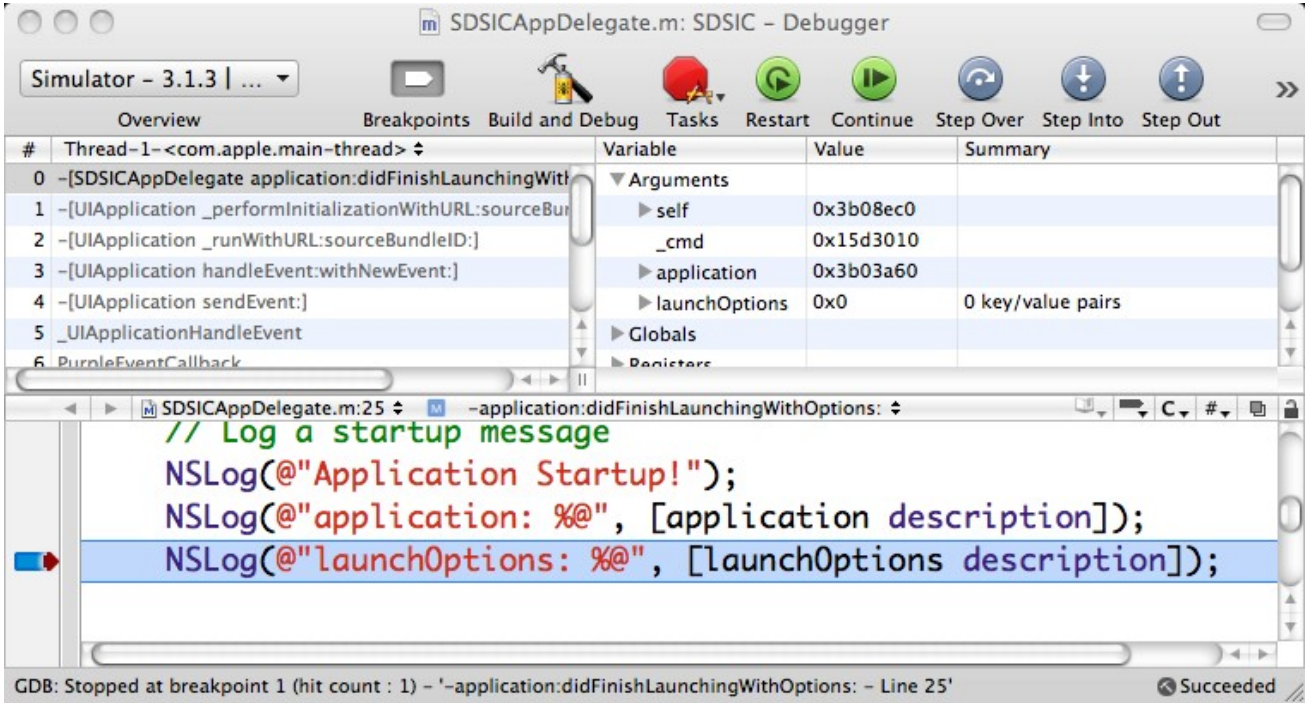
Demo 2: Android SDK
Eclipse: LogCat After Running



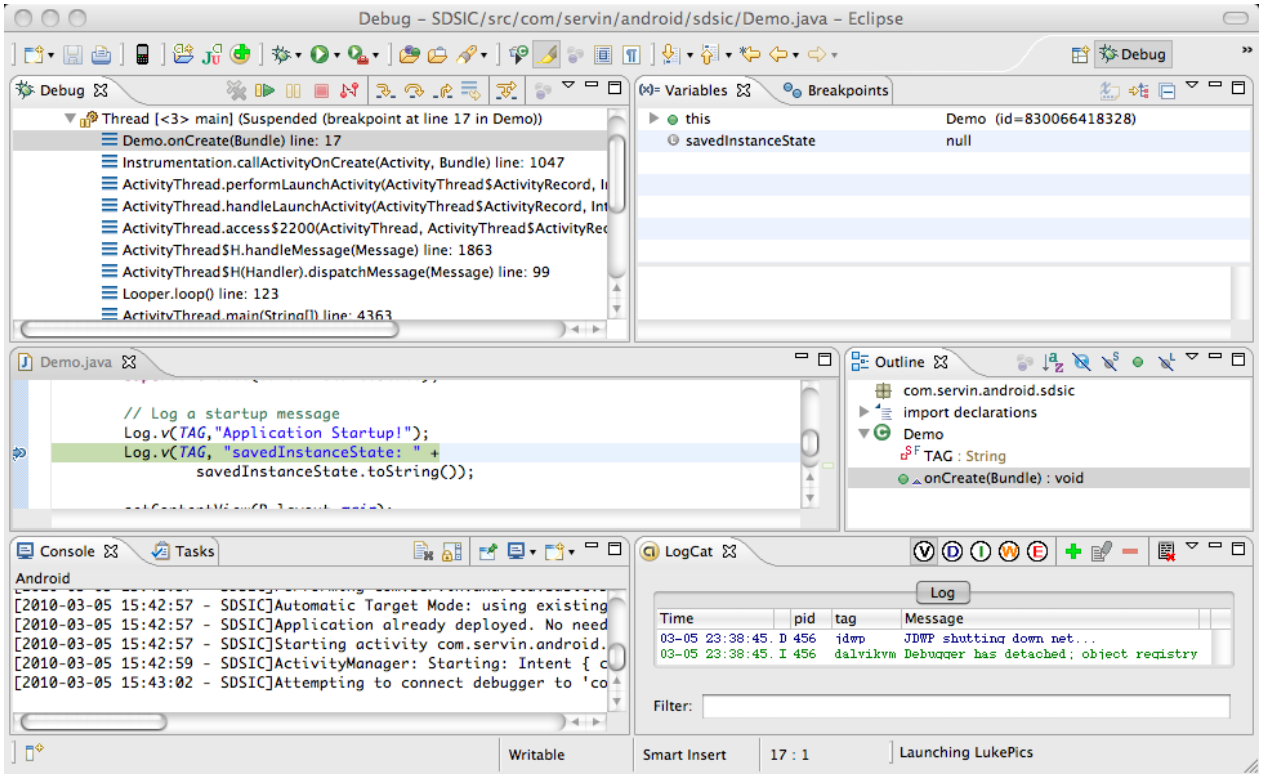
Demo 3

Debugging

Demo 3: iPhone SDK Xcode: Debugger



Demo 3: Android SDK Eclipse: Debug Perspective



Demo 4

Create UITextView/TextView at Runtime

iPhone: UITextView
Android: TextView

Demo 4: iPhone SDK UITextView at Runtime

```
- (BOOL)application:(UIApplication *)application
    didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

    ...code not shown...

    CGRect frame = viewController.view.bounds;
    UITextView *myTextView =
                  [[UITextView alloc] initWithFrame:frame];
    myTextView.text = @"SDSIC Demo\n"
          @"This is a test\nBe at your best";
    myTextView.editable = NO;
    [viewController.view addSubview:myTextView];

    [window addSubview:viewController.view];
    [window makeKeyAndVisible];

    return YES;
}
```

Demo 4: iPhone SDK UITextView at Runtime

Demo 4: Android SDK TextView at Runtime

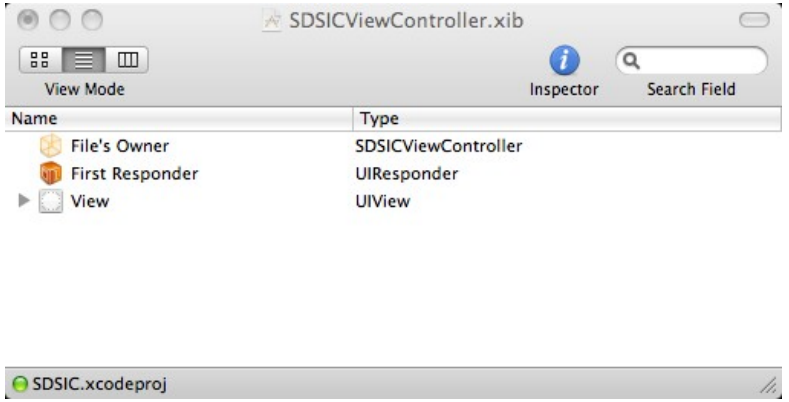
```
// Demo.java

...code not shown...

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

         ...code not shown

            TextView myTextView = new TextView(this);
            myTextView.setText("SDIC Demo\n" +
                "This is a test\nBe at your best.");
            this.setContentView(myTextView);

    }
```

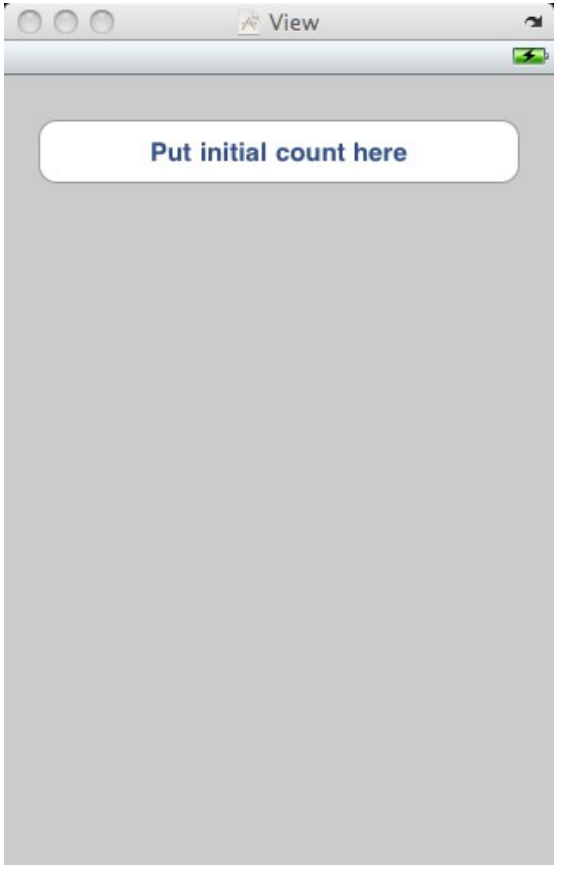
Demo 4: Android SDK TextView at Runtime

Demo 5

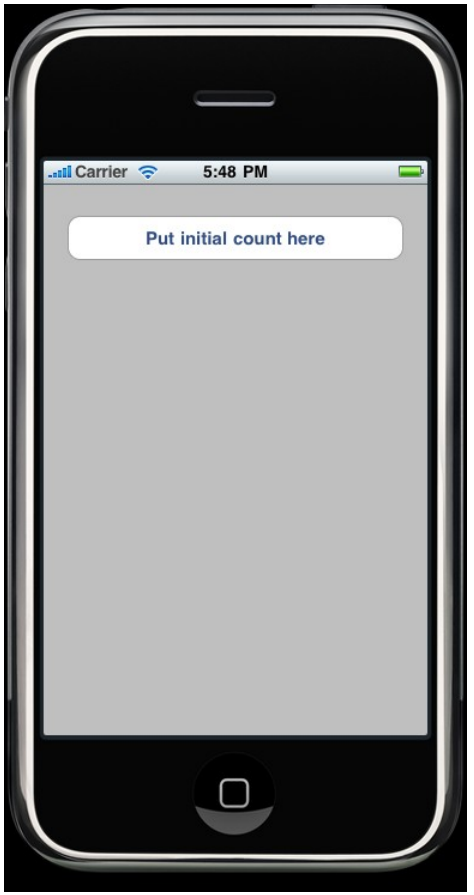
Create UIButton/Button using Layout Tools

Demo 5: iPhone SDK Interface Builder and UIButton

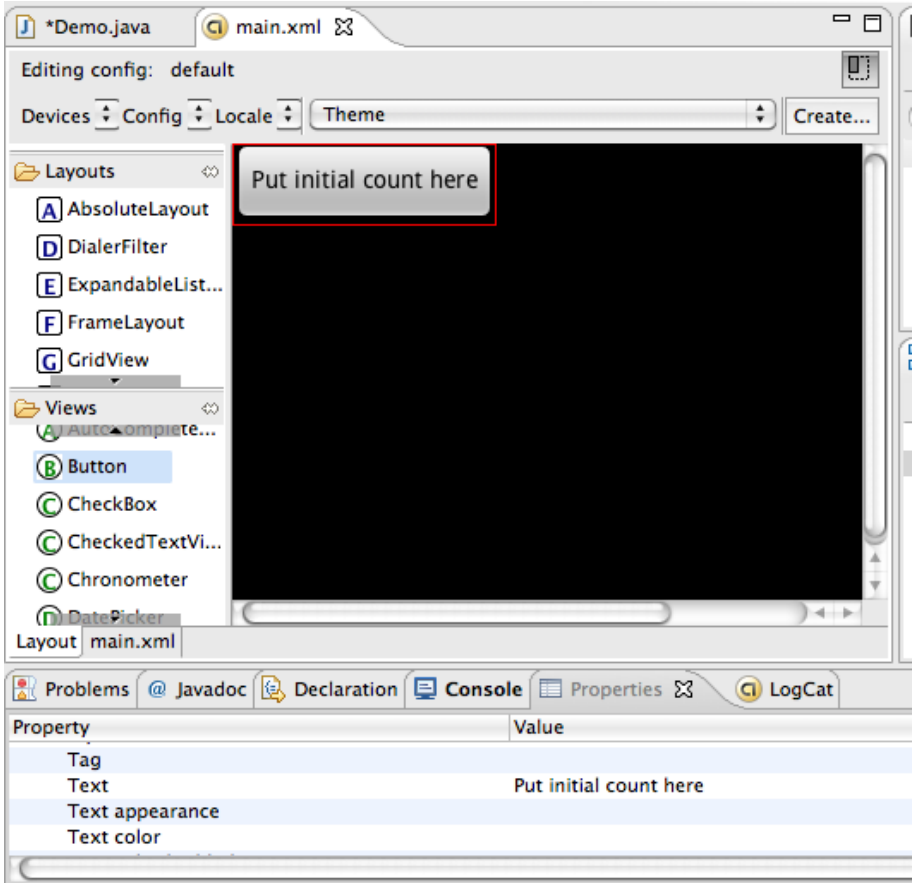




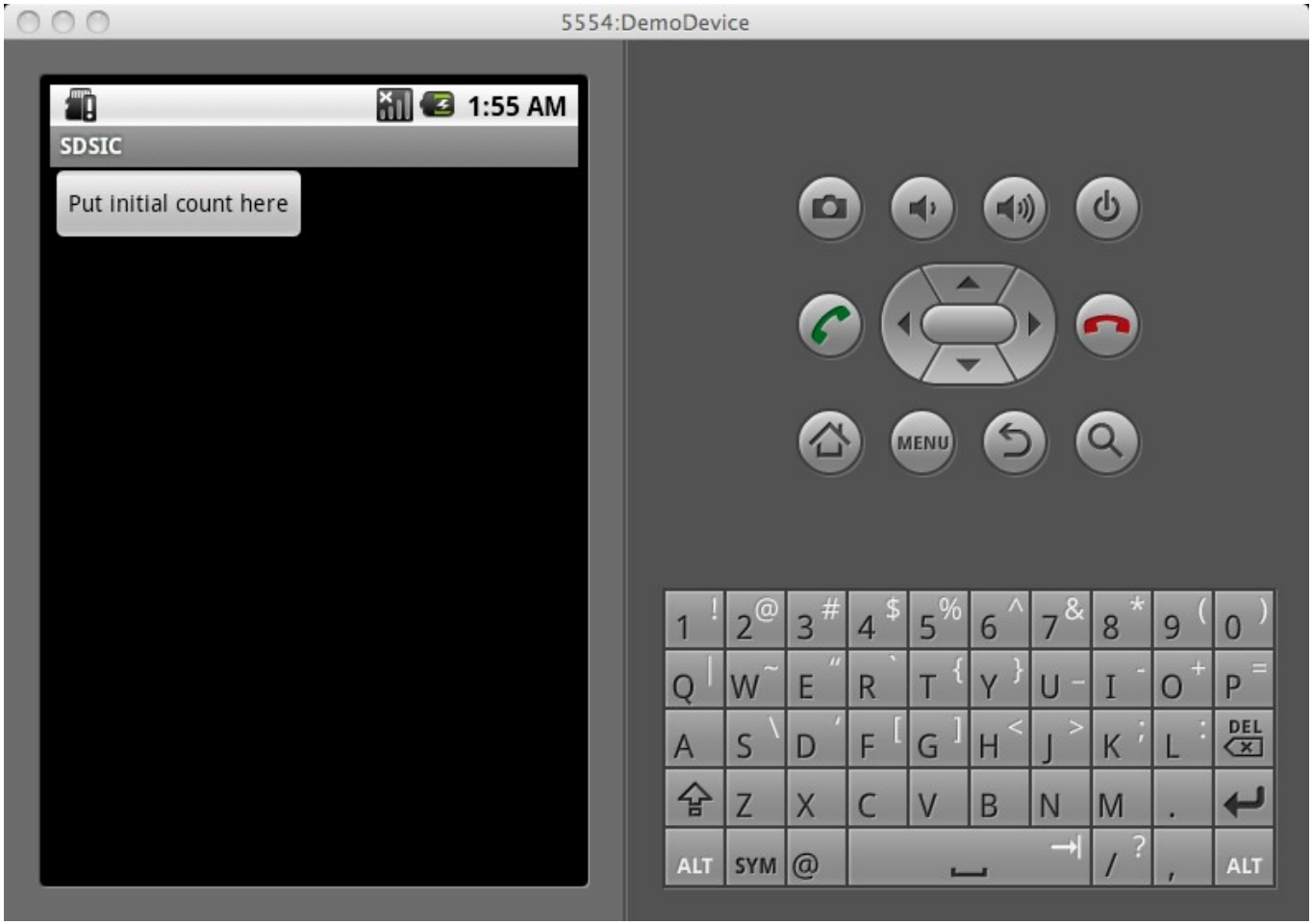
Demo 5: iPhone SDK Running the App



Demo 5: Andriod SDK Eclipse: XML Layout



Demo 5: Android SDK Running the App



Demo 6

Change Value of UIButton/Button At Runtime

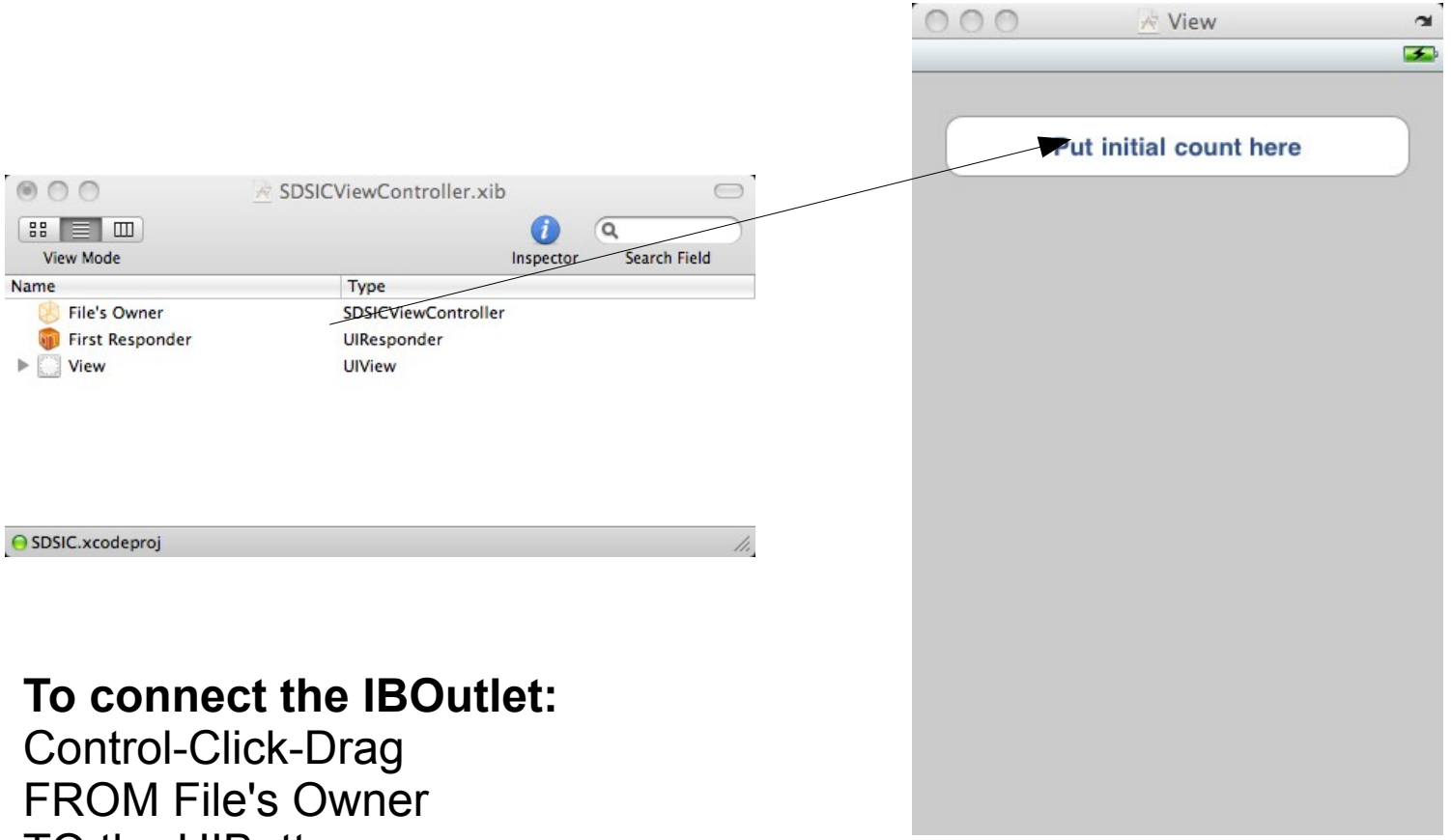
Demo 6: iPhone SDK IBOutlet - SDSICViewController.h

```
//  SDSICViewController.h

#import <UIKit/UIKit.h>

@interface SDSICViewController : UIViewController {

    IBOutlet UIButton *myButton;

}
```

Demo 6: iPhone SDK SDSICViewController.m

```
//  SDSICViewController.m

#import "SDSICViewController.h"

@implementation SDSICViewController

- (void)viewDidLoad {
    [super viewDidLoad];

    [myButton setTitle:@"0" forState:UIControlStateNormal];

}
@end
```

Demo 6: iPhone SDK
Interface Builder - Control-Click-Drag



To connect the IBOutlet:
Control-Click-Drag
FROM File's Owner
TO the UIButton.

Demo 6: iPhone SDK Running

Demo 6: Android SDK Button As Runtime

```
package com.servin.android.mon2;

...code not shown...

public class Demo extends Activity {

    private Button myButton;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        myButton = (Button) this.findViewById(R.id.Button01);
        myButton.setText("0");
    }
}
```

Demo 6: Android SDK Button As Runtime

Demo 7

Handle UIButton/Button Events

Demo 7: iPhone SDK IBOutlet - SDSICViewController.h

```
//  SDSICViewController.h

#import <UIKit/UIKit.h>

@interface SDSICViewController : UIViewController {

    IBOutlet UIButton *myButton;

    int counter;

}

-(IBAction) doButton;

@end
```

Demo 7: iPhone SDK SDSICViewController.m

```
//  SDSICViewController.m

...code not shown...
```

```
-(IBAction) doButton {
    counter++;
    NSString *s = [NSString stringWithFormat:@"%d", counter];
    [myButton setTitle:s forState:UIControlStateNormal];
}
```

Demo 7: iPhone SDK
Interface Builder - Control-Click-Drag

To connect the IBAction:
Control-Click-Drag
FROM UIButton
TO File's Owner.

Demo 7: iPhone SDK Running

Demo 7: Android SDK
Demo.java - 1

```
public class Demo extends Activity
                        implements OnClickListener {
    private Button button;
    private int counter;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);
        button = (Button) this.findViewById(R.id.Button01);
        button.setText("0");

        button.setOnClickListener(this);

    }
```

Demo 7: Android SDK Demo.java - 2

```
@Override
public void onClick(View v) {
    Counter++;
    // Either method will work
    //myButton.setText(String.format("%d", counter));
    myButton.setText("" + counter);

}
```

Demo 7: Android SDK Button As Runtime

We Are Done!

Thank You!