References


Android Documentation


Reading Assignments
Some Background
Optimizing a Screen for Mobile Use

Jakob Nielson - Usability Expert

http://www.useit.com/alertbox/mobile-redesign.html

Redesigned site as exercise

http://www.useit.com/alertbox/mobile-redesign.html
Allkpop mobile Site

The Good

- Separate desktop & mobile versions
- Large touch targets
- Content-carrying keywords

Tuesday, April 12, 2011

http://www.useit.com/alertbox/mobile-redesign.html
Custom
Text summaries

Thumbnail images
Iteration 2

Added
Search
Date

Combined less used items into one menu
Iteration 3

Menu becomes "Categories"

Moved Search and Categories apart
Final Version

One screen

Required 10 design changes to meet usability guidelines
Psychopathology of Things

Affordance
Conceptual Model
Make Things Visible
Mapping
Feedback

From The Design of Everyday Things, Norman, Basic Books, 2002
Affordance
Affordance
Affordance
Conceptual Model
Make things Visible

Start

Stop
Make things Visible
Mapping
Mapping
Feedback

0.1 seconds

0.1 to 1 seconds

More than one second
Why is Angry Birds so Popular

Information Architecture

"The truly successful mobile products always have a well-thought-out information architecture"

"information architecture stands apart as being the most crucial element of your product"

Brian Fling
Information Architecture

How the user will get to information or perform tasks within an application

How the user will interact with the application to accomplish her goals

Visual layout of information

The design of the visual paradigms used to create action or understanding

http://mobiledesign.org/what_is_information_architecture
Information Architecture

Kim Hyung Jun releases “Gi…”
By ashpodel
Categories: Music
Tags: girl, kim hyung jun, mv

2AM’s Seulong tries to take …
By chloejn
Categories: Internet
Tags: 2am, aegyo, jo kwon, seulong, twitter

Lee Seung Gi for ‘Pizza Hut’
By ramham424
Categories: Endorsements
Tags: lee seung gi, pizza hut

B2ST to hold their first fan m…
By VITALSIGN
Categories: Concerts / Events

Another cover for SNSD’s Japanese single “Mr. Taxi” revealed
The rumors are true! There will be an original single of “Mr. Taxi” for the Japanese market.

Jang Geun Suk is shaking up Japan
The showcase celebrating the release of his Japanese single is now sold out.

MBLAQ’s Mir rages against Jang Ja Yeon’s offenders
He is expressing his anger towards what is called the “Jang Ja Yeon Case.”

Won Bin reveals second “C’mon Girl” teaser for Japan
The CD release for “C’mon Girl” is slated for March 14th with a total of 4 tracks.
Some Design Tools
Clickstreams
Prototypes

Source http://mobiledesign.org/prototyping
Android UI
Examples

http://developer.android.com/resources/browser.html

Source
androidInstallation/platforms/android-10/samples/ApiDemos
Activity

Code that does some work

Single, focused thing that a user can do

Usually each screen(View) has its own activity

An application may have multiple screens, hence multiple activities

An application runs in its own Linux process

Activities can be viewless
Activity starts

onCreate()

onStart()

onResume()

Activity is running

Another activity comes in front of the activity

onPause()

The activity comes to the foreground

onStop()

The activity is no longer visible

onDestroy()

Activity is shut down

User navigates back to the activity

Process is killed

Other applications need memory
Views

View
Displays content in rectangular area of screen
Handles
   Layout, focus, scrolling
   Keyboard events
   Gestures

ViewGroups
Manages set of views and view groups
Composite pattern
View Documentation

Common Layouts

FrameLayout - displays one child

LinearLayout
  Items stacked vertically or horizontally
Common Layouts

TableLayout

RelativeLayout
Layout Documentation with Examples

Intents - Calling Activities

Android application consists of multiple activities

Activity represents one screen or view

Going from one screen to another
  Requires calling activity

Can't call new activity directly
  Use intent to indicate activity to start
Intent Resolution - Explicit Intents

Specify the component (class) an intent is to run

Common way to call your own code

Intent go = new Intent();
go.setClassName("edu.sdsu.cs696", "edu.sdsu.cs696.Hello");
startActivity(go);
Intent Resolution - Implicit Intents

Provide information about activity you want to run

display web page

System determines which component to run

If more than one activity can handle request
User is asked to select

Each activity declares in manifest what it can handle

<intent-filter>
  <action android:name="android.intent.action.MAIN" />
  <category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
Intent - Passing Data

IntentExample

Displays/Edits age

Go button
  Calls PersonEditor
  Passes data
    Name
    Age

PersonEditor

Displays/Edits Name and age

Done button
  Returns edited data back
  Age = 0 cancels edit
When we want a reply back from an Intent request we supply a request number. The request number is return with the answer. That way it is possible to know where the request originated from.
IntentExample.Java continued

Sending the data to PersonEditor

```java
public void onClick(View v) {
    Intent go;
    go = new Intent();
    go.setAction("android.intent.action.EDIT");
    go.addCategory("person_editor");
    String newAge = numberText.getText().toString();
    go.putExtra("age", newAge);
    go.putExtra("name", "Roger");
    startActivityForResult(go, INTENT_EXAMPLE_REQUEST);
}
```

The name was sent just to show we can send multiple items. They can be of any base type or serializable. See the putExtra methods in the Intent class.
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode != INTENT_EXAMPLE_REQUEST) {
        numberText.setText("Not from me");
        return;
    }
    switch (resultCode) {
    case RESULT_OK:
        String editedAge = data.getStringExtra("age");
        numberText.setText(editedAge);
        break;
    case RESULT_CANCELED:
        numberText.setText("Cancelled");
        break;
    }
}
PersonEditor.java

public class PersonEditor extends Activity implements View.OnClickListener {
    private EditText ageText;
    private EditText nameText;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.person_editor);
        Button done = (Button) findViewById(R.id.edit_done);
        done.setOnClickListener(this);
        ageText = (EditText) this.findViewById(R.id.edit_age);
        nameText = (EditText) this.findViewById(R.id.edit_name);
        Bundle personData = getIntent().getExtras();
        String age = personData.getString("age");
        String name = personData.getString("name");
        if ((age != null) && (name != null)) {
            ageText.setText(age);
            nameText.setText(name);
        }
    }
}

Showing how to access the intent that started the activity and extracting the extras from the intent. If the key-value pair was not set in the intent the value will be returned as null.
public void onClick(View v) {
    String newAge = ageText.getText().toString();
    Intent result = getIntent();
    result.putExtra("age", newAge);
    if (newAge.equals("0"))
        setResult(RESULT_CANCELED, result);
    else
        setResult(RESULT_OK, result);
    finish();
}
AndroidManifest.xml

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="edu.sdsu.cs683.example" android:versionCode="1"
  android:versionName="1.0.0">
  <application android:icon="@drawable/icon" android:label="@string/app_name">
    <activity android:name=".IntentExample" android:label="@string/app_name">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>

    <activity android:label="PersonEditor" android:name="PersonEditor">
      <intent-filter>
        <action android:name="android.intent.action.EDIT"></action>
        <category android:name="person_editor"></category>
        <category android:name="android.intent.category.DEFAULT" />
      </intent-filter>
    </activity>
  </application>
</manifest>

The intent filter for the activity must contain all the categories used by the intent to select the activity. It can contain more categories. The example does not work without the default category.