Android App Development
for Google TV

Jiho Park
jihopark@skku.edu
Contents

1. Google TV Overview
2. Setting Google TV SDK
3. Google TV App Development
4. User Interface Design Guidelines
5. Migrating Existing Applications
1. Overview of Google TV
Google TV Roadmap

2010  May 2011  Late Summer, 2011  Q4 2011

Android 3.1 (Honeycomb)
- Tablet
- TV

Google TV
- Android compatibility
- Android Market for TV
- Emulator
- Adb
- USB Host
- Samsung, Vizio

Android 3.2 (Ice cream Sandwich)
- Mobile
- Tablet
- TV
- TV specific functionality
- 3D

Chrome
- Sony, Logitech
Google TV Features

- Android 3.1 (Honeycomb)
- Chrome browser
- Not a cable replacement
- App friendly
- Full web experience on the TV
Google TV Features

Screen Shot: Search Box

Android Market for Google TV
Google TV H/W Specifications

- Intel Atom CE4100 chip (Sony)
- ARM-based dual core (?)
- HDMI
- Bluetooth
- USB port
- Streaming 1080p
Classification of Google TV Apps

1. Universal application
   - Apps with full functionality on all Android devices

2. Limited functionality application
   - Apps with limited hardware or features on Google TV
   - Still useful

3. Google TV-only applications
   - Apps working only on Google TV
   - To install only on Google TV
     - <uses-library> in AndroidManifest.xml
     - <uses-library android:name="com.google.android.tv" android:required="true"/>

4. Unsuitable Apps on Google TV
   - Not working or useless
   - To Prevent installing on Google TV
     - <uses-feature> in AndroidManifest.xml
     - <uses-feature android:name="android.hardware.touchscreen" android:required=true"/>
2. Setting Google TV SDK
Setting Up a Development Environment

- Google TV App with Android 2.3 or 3.1

- Installing the Android SDK
  - Android Honeycomb SDK

- Settings for Google TV AVD
  - AVD target of Android 3.0 (API Level 11) or later.
  - Screen resolution of 1152 x 648 (corresponding to a 720p TV screen)
  - No 1080p; too large emulator screen
  - Hardware Properties
    - hw.lcd.density: 213 (hdpi density)
    - hw.dPad: yes
    - hw.keyboard.lid: no
    - vm.heapSize: 48
    - hw.ramSize: 256
    - hw.touchScreen: no
3. Google TV App Development

3.1 Guidelines
3.2 UI Controls Guidelines
3.3 Feature Support in Google TV
Goggle TV Principal

1. No Conventional Display
   – No own display Included.
     • detecting screen resolution and density during initialization.
   – Large displays viewed from a distance
     • Large Application UI required
     • Clear Boundaries between user-selectable regions required

2. No Conventional UI controls
   – Better keyboard
   – Directional pad (D-Pad)
   – No touchscreen

3. Different H/W & Features
   – No GPS
   – No Sensors
   – No Telephony
1) Google TV screen size and density
2) Font guidelines
3) Widget guidelines
4) Layout guidelines
5) Display resolution guidelines

3.1 Display Guidelines
### Screen Size & Pixel Density

- **Google TV Screen Size**
  - Large, not xLarge
- **Google TV Screen Pixel Density**
  - 1080p TV: xhdpi
  - 720p TV: hdpi

<table>
<thead>
<tr>
<th>Screen Size</th>
<th>LDPI (Low Density) ~120 DPI</th>
<th>MDPI (Medium Density) ~160 DPI</th>
<th>HDPI (High Density) ~240 DPI</th>
<th>XHDPI (Extra High Density) ~320 DPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small Screen</td>
<td>QVGA (240x320)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Normal Screen</td>
<td>WQVGA400 (240x400)</td>
<td>HVGA(320x480) HTC-G1</td>
<td>WVGA800 (480x800) WVGA854 (480x854)</td>
<td></td>
</tr>
<tr>
<td>Large Screen</td>
<td>WQVGA432 (240x432)</td>
<td>WVGA800 (480x800) WVGA854 (480x854)</td>
<td>TV 720p (1280x720) 213 DPI</td>
<td>TV 1080p (1920x1080) 320 DPI</td>
</tr>
<tr>
<td>Extra Large Screen</td>
<td></td>
<td>WXGA (1280x800) Xoom Tablet</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Screen Size & Pixel Density

- Alternative Resources for Screen size and pixel density

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Qualifier Values</th>
<th>Description</th>
<th>Example directory name</th>
</tr>
</thead>
</table>
| Screen size           | large            | • Use with layout.  
  • Layouts in this directory are used on Google TV;  
  • all Google TV devices are classified large | res/layout-large       |
| Screen pixel density  | hdpi, xhdpi, nodpi | • Use with drawable or bitmap resources.  
  • Use hdpi for 720p, xhdpi for 1080p.  
  • Use nodpi to prevent Android from scaling any of the resources. | res/drawable-hdpi     |
| Touchscreen type      | notouch          | • Use with any resource.  
  • Contents are used on any no touchscreen device. (Currently, only Google TV.)  
  • This qualifier can be used with other qualifiers. | res/drawable-xhdpi-notouch |
Font & Widget Guidelines

- **Font Guidelines**
  - Use *scale-independent pixels* (*sp* unit) for the font size.

- **Widget Guidelines**
  - Use the "10-foot (3-meter) UI" rule
    - widgets (buttons, text, ...) to be clearly visible from ten feet away from the TV
    - greater for very large screens
  - Use *layout-relative sizing* (e.g. *wrap_content*)
  - Use *density-independent pixel values* (*dp* units)
  - No hard pixels (*px* units).
Layout Guidelines

- **Layout Guidelines**
  - Use `RelativeLayout`
  - Align child View objects to their parent Layout
    - e.g. `android:alignParentLeft` for left-align
    - `android:alignParentRight`, ..... 
  - with `RelativeLayout`, set **minimum margins** for each View object, using **dp units**.
    - layout not to be crowded.
    - use **margins**, not **padding**;
    - (In general, padding controls spacing *inside* a View object, while margins control the spacing *between* View objects.)
Display Resolution Guidelines

- **Display Resolution Guidelines**
  - Consider **Overscan**
    - Reduced TV's usable screen margins
    - No standard for the amount of reduction, because of various factors.
    - E.g. 1920 x 1080, 5% reduction on all four edges \( \rightarrow \) 1728 x 972.
  - Use Dynamic Layouts
Display Guidelines: App Test for correct Display

- Smaller Display Size Test for overscan
  - Start Test by **setting an actual display size that is smaller than the display size:**
    1. Go to **Settings > Picture and Sound > Picture Size > Overscan** dialog.
    2. Select **No, start with default.**
    3. On the screen, a black box entitled **Maximize your screen area** appears. The box is surrounded by a light blue edge.
    4. Click **Next.**
    5. At each succeeding screen, click **Next** but do not adjust the edges of the box.
    6. At the last screen, click **Save and restart** to reboot the Google TV device.
    7. After your device reboots, examine the screen closely. You should see that the edges of the Home screen have a large black border around them.
    8. Now test your application. The entire application with all of its UI controls should be visible and should appear undistorted.
Display Guidelines: App Test for correct Display

• Larger Display Size Test for overscan
  – Next, set the actual display size to be slightly larger than the TV’s viewable area:
    1. Go to Settings > Picture and Sound > Picture Size > Overscan dialog.
    2. Click Yes, start with current.
    3. On the screen, a black box entitled Maximize your screen area appears. The box is surrounded by a light blue edge.
    4. Click Next.
    5. On the next screen, press Up to move the black box edge one line up towards the top edge of the entire screen. When the box edge reaches the screen edge, and the blue border is no longer visible, press Up 10 times more to adjust the screen area beyond the actual viewable area.
    6. Repeat step 3 to adjust the right, bottom, and left edges of the screen.
    7. At the last screen, click Save and restart to reboot the Google TV device.
    8. Now test your application. The entire application with all of its UI controls should be visible and should appear undistorted.
1) Controlling Navigation & Focus
2) Media Keys

3.2 UI Controls Guidelines
Controlling Navigation and Focus

• The standard input hardware for a Google TV
  – Directional pad (D-Pad)
  – Media transport controls keys (play, pause, rewind, …)
  – Alphanumeric keyboard
  – Mouse or equivalent
  – No full touch screen

• Interacted by D-Pad keys
  – Up, Down, Left, and Right arrow keys for navigation (sends KeyEvent.KEYCODE_DPAD_UP)
  – Center key for action (sends KeyEvent.KEYCODE_DPAD_CENTER)
  – ENTER key (sends KeyEvent.KEYCODE_ENTER)
  – Test Apps with the D-Pad for easy navigating Activity.

• More designing UIs Info for non-touchscreen devices
  – Designing for Accessibility
    (http://developer.android.com/guide/practices/design/accessibility.html)
UI Controls Guidelines: Media keys

- Seven media-control keys on Google TV keyboards
- Handle control keys in the `android.view.View.onKeyDown()` callback for the media controlling view

Google TV Media Keys, KeyEvent Mappings, and Symbols

<table>
<thead>
<tr>
<th>Action</th>
<th>KeyEvent keycode constant</th>
<th>Keyboard Symbol(^1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start playback</td>
<td><code>KeyEvent.KEYCODE_MEDIA_PLAY</code></td>
<td>🎥</td>
</tr>
<tr>
<td>Pause playback</td>
<td><code>KeyEvent.KEYCODE_PAUSE</code></td>
<td>🕹️</td>
</tr>
<tr>
<td>Stop playback</td>
<td><code>KeyEvent.KEYCODE_STOP</code></td>
<td>🚪</td>
</tr>
<tr>
<td>Next</td>
<td><code>KeyEvent.KEYCODE_NEXT</code></td>
<td>🎥</td>
</tr>
<tr>
<td>Fast-forward</td>
<td><code>KeyEvent.KEYCODE_MEDIA_FAST_FORWARD</code></td>
<td>🎥</td>
</tr>
<tr>
<td>Previous</td>
<td><code>KeyEvent.KEYCODE_MEDIA_PREVIOUS</code></td>
<td>🎥</td>
</tr>
<tr>
<td>Rewind</td>
<td><code>KeyEvent.KEYCODE_MEDIA_REWIND</code></td>
<td>🎥</td>
</tr>
</tbody>
</table>

\(^1\)The symbol usually associated with the key. Some keyboards may have two symbols on the same key; in this case, the scan code sent out by the key is controlled by a separate modifier key.
Example of handling media key events:

```java
@Override
/** Overrides the callback method for a UI control handling a slideshow **/
public boolean onKeyDown(int keyCode, KeyEvent event) {

    /* Chooses the action to take, based on the incoming keycode */
    switch (keyCode) {
        case KeyEvent.KEYCODE_PAUSE:
            Log.d("MediaPlayer", "Pausing the slideshow");
            if (slideshow.pause()) {
                showStatusToast(R.string.slideshow_paused);
            }
            return true;
        case KeyEvent.KEYCODE_MEDIA_PLAY:
            Log.d(TAG, "Resuming the slideshow");
            if (slideshow.resume()) {
                showStatusToast(R.string.slideshow_resumed);
            }
            return true;
        /* Keycodes should always be passed upwards in the chain for handling. */
        default:
            return super.onKeyDown(keyCode, event);
    }
}
```
3.2 Feature Support in Google TV

1) Unsupported Hardware
2) Unsupported Software
3) Supported Media Formats
## Unsupported Hardware

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Feature(s)</th>
<th>Feature descriptor*</th>
</tr>
</thead>
</table>
| Bluetooth                 | • A2DP  
• wireless keyboard**                                                  | android.hardware.bluetooth |
| Camera                    |                                                                           | android.hardware.camera  
android.hardware.camera.autofocus  
android.hardware.camera.flash  
android.hardware.camera.front |
| Location (GPS)            | Geolocation is less useful for a Google TV. For an example of handling the two preferences in a single application, see the topic [Migrating Existing Applications](#). | android.hardware.location.gps |
| Microphone                |                                                                           | android.hardware.microphone |
| NFC (Near-Field Communications) |                                                                           | android.hardware.nfc |
| Sensors                   | • Orientation  
• Movement  
• Automatic screen brightness                                           | android.hardware.sensor.accelerometer  
android.hardware.sensor.barometer  
android.hardware.sensor.compass  
android.hardware.sensor.gyroscope |
| Telephony                 | • Phone dialing  
• SMS  
• Cell-based data (for example, 3G)                                    | android.hardware.telephony  
android.hardware.telephony.cdma  
android.hardware.telephony.gsm |
| Touchscreen               | The android.hardware.faketouch feature *is* supported.                    | android.hardware.touchscreen  
android.hardware.touchscreen.multitouch |
Unsupported Software

• Not Supported S/W Features
  – Live Wallpaper (android.software.live_wallpaper).
    • **Note:** Regular wallpaper is supported, but it is not usually visible because live TV is playing when an application is not running.
  – SIP/VOIP (android.software.sip).

• Supported S/W Features, not in previous versions
  – OpenGL for Java.
  – Application Widgets.
**Supported Media Formats**

- Same media and Container formats in Android 3.x
- Additional support for DD/AC3 encoding and decoding

<table>
<thead>
<tr>
<th>Format/Codec</th>
<th>Encoder</th>
<th>Decoder</th>
<th>Details</th>
<th>Supported file types/container formats</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Honeycomb</td>
<td>TV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JPEG</td>
<td>•</td>
<td>•</td>
<td>Base+progressive</td>
<td>JPEG(.jpg)</td>
</tr>
<tr>
<td>GIF</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PNG</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BMP</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Supported Media Formats (Video)

<table>
<thead>
<tr>
<th>Format/Codec</th>
<th>Encoder</th>
<th>Decoder</th>
<th>Details</th>
<th>Supported file types/container formats</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Honeycomb</td>
<td>Honeycomb</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TV</td>
<td>TV</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H.263</td>
<td>•</td>
<td>•</td>
<td>Opt</td>
<td>Sorenson</td>
</tr>
<tr>
<td>H.264 AVC</td>
<td>•</td>
<td>•</td>
<td>Yes</td>
<td>Baseline Profile (BP), H.264 in AVI has limited B-frame support</td>
</tr>
<tr>
<td>MPEG-4 SP</td>
<td>•</td>
<td>•</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>VP6</td>
<td>•</td>
<td>(Yes)</td>
<td>Optional HW Accel on CE4100 SOC (Sodaville)</td>
<td>Flash (.flv)</td>
</tr>
<tr>
<td>VP8</td>
<td>•</td>
<td>•</td>
<td>(Yes)</td>
<td>Optional HW Accel on CE4100 SOC (Sodaville)</td>
</tr>
<tr>
<td>WMV9/VC-1</td>
<td>•</td>
<td>Yes</td>
<td></td>
<td>ASF (.wmv, .asf)</td>
</tr>
<tr>
<td>MPEG-2</td>
<td>•</td>
<td>Yes</td>
<td></td>
<td>MPEG (.mpg, .mpeg)</td>
</tr>
<tr>
<td>H.263</td>
<td>•</td>
<td>•</td>
<td>Opt</td>
<td>Sorenson</td>
</tr>
<tr>
<td>H.264 AVC</td>
<td>•</td>
<td>•</td>
<td>Yes</td>
<td>Baseline Profile (BP), H.264 in AVI has limited B-frame support</td>
</tr>
</tbody>
</table>
## Supported Media Formats (Audio)

<table>
<thead>
<tr>
<th>Format/Codec</th>
<th>Encoder</th>
<th>Decoder</th>
<th>Details</th>
<th>Supported file types/container formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAC LC/LTP</td>
<td>Honeycomb TV</td>
<td>Honeycomb TV HW</td>
<td>Yes</td>
<td>Mono/Stereo content in any combination of standard bit rates up to 160 kbps and sampling rates from 8 to 48kHz</td>
</tr>
<tr>
<td>HE-AACv1(AAC+)</td>
<td></td>
<td></td>
<td></td>
<td>3GPP (.3gp) and MPEG-4 (.mp4, .m4a). No support for raw AAC (.aac)</td>
</tr>
<tr>
<td>HE-AACv2(enhanced AAC+)</td>
<td></td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>AMR-NB</td>
<td></td>
<td></td>
<td></td>
<td>4.75 to 12.2 kbps sampled @ 8kHz</td>
</tr>
<tr>
<td>AMR-WB</td>
<td></td>
<td></td>
<td></td>
<td>9 rates from 6.60 kbit/s to 23.85 kbit/s sampled @ 16kHz</td>
</tr>
<tr>
<td>MP3</td>
<td></td>
<td></td>
<td>Yes</td>
<td>Mono/Stereo 8-320Kbps constant (CBR) or variable bit-rate (VBR)</td>
</tr>
<tr>
<td>MIDI</td>
<td></td>
<td></td>
<td></td>
<td>MIDI Type 0 and 1. DLS Version 1 and 2. XMF and Mobile XMF. Support for ringtone formats RTTTL/RTX, OTA, and iMelody</td>
</tr>
<tr>
<td>Ogg Vorbis</td>
<td></td>
<td></td>
<td></td>
<td>Google TV also supports the WebM format</td>
</tr>
<tr>
<td>PCM/WAVE</td>
<td></td>
<td></td>
<td>8- and 16-bit linear PCM (rates up to limit of hardware)</td>
<td></td>
</tr>
<tr>
<td>DD/AC3</td>
<td></td>
<td></td>
<td>Yes</td>
<td>DIVX(.divx), MP4 (.mp4), AVCHD (.mts)</td>
</tr>
<tr>
<td>WMA</td>
<td></td>
<td></td>
<td>Yes</td>
<td>ASF (.asf, .wma, .wmv)</td>
</tr>
<tr>
<td>DTS</td>
<td></td>
<td></td>
<td>Yes</td>
<td>No support for raw DTS</td>
</tr>
</tbody>
</table>
4. User Interface Design Guidelines
UI Design Guideline: Input Device

Touch optimized UI

D-Pad optimized UI
UI Design Guideline: Design for D-Pad Use

Touch optimized UI

D-Pad optimized UI

http://tv.clicker.com/
UI Design Guideline: CNBC Real Time

Mobile Phone UI

Web UI

Google TV UI
5. Migrating Existing Applications: Panoramio
Migrating Existing Applications: Panoramio

Feature: WiFi, 3G, GPS, Location.

Mostly Portrait, Touch screen, keypad
Migrating Existing Applications: Panoramio

Google TV App

No Mapview ➔ Search Box

Gridview

Feature: LAN,
Always Landscape, Key board

To do
• Handling unsupported features
• Migrating UI layout
• Migrating UI controls
• Migrating Media Keys
Understand that content is king.
- Get users to the content as quickly and easily as possible.
- Don't interrupt when users are watching TV. Instead, make the viewing experience better.

Respect the living room context.
- Think about what users will and won't want to do when viewing TV with their family and friends.

Remember that TV is social.
- Consider how groups might use your website or application.
- Offer ways for individuals to use your site or apps in social settings.

Learn the pros and cons of TV screens and audio.
- TV screens are wider and colors look different.
- Text must be readable from a distance.
- Sound is now a viable interface element.

Make it easy.
- Offer simple choices and make actions obvious and easy to select.
- Provide navigation that is simple enough for a remote control.