External Display Support Using DisplayManager

What's an External Display?

- Auxiliary screen ("secondary display")

- Inconsistently available

- Display-only (no touchscreen)

- Not:

  – Simply being on a TV (think OUYA)

  – Devices with dual primary displays (SONY Tablet P)

Getting To Second: Sending

- HDMI

- Mobile High-Definition Link (MHL)

- SlimPort

- Proprietary Cable

- Miracast/AllShare Cast

  - WifiDirect-based display protocol

  - Available on many modern devices

Getting to Second: Receiving

- HDMI

  - Direct

  - Via MHL adapter

  - Via SlimPort adapter

- Miracast/AllShare Cast

  - Direct (e.g., most of Samsung Smart TVs)

  - Via HDMI adapter

Secondary Display: What Is Shown

- Android 1.0 through 2.3

  – Generally nothing

  – Sporadic device-specific support (e.g., HTC DROID Incredible and composite output)

- Android 3.0 through 4.1

  – Mirroring

  – Some limited developer control (video players)

- Android 4.2+: Developer Control

  – At least for devices that shipped with 4.2+

Not All Screens Are Created Equal

- Key Differences

  - Size (720p, 1080p, etc.)

  - Density (`tvdpi`, `hdpi`, etc.)

- Net: Different `Context` for Different Display

  - Inflate layouts, load resources, etc. using proper `Context` to get the right ones for the targeted display

Exerting Your Control

- Presentation
  - Subclass of `Dialog`, added in API Level 17
  - Override `onCreate()`, call `setContentView()`
  - Supplies `Context` suitable for use with designated `Display`
  - When displayed using `show()`, appears on designated `Display`

Exerting Your Control

- CWAC-Presentation and PresentationHelper

  - Lets you know when a secondary display becomes available or goes away

  - Usage

    - Create instance

    - Supply a Context and a Listener

    - Forward `onPause()`/`onResume()`

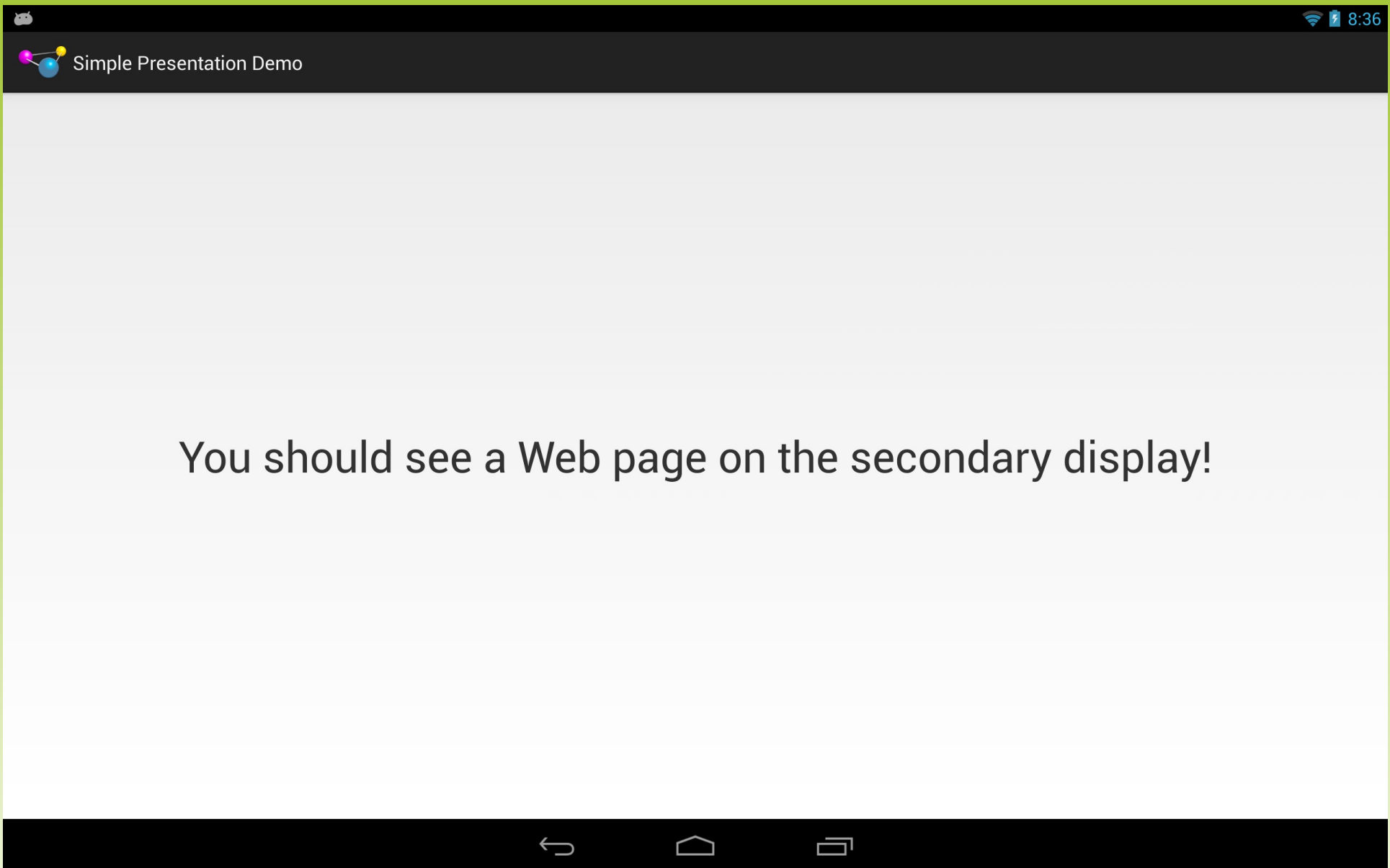   - Implement `showPreso()`/ `clearPreso()`
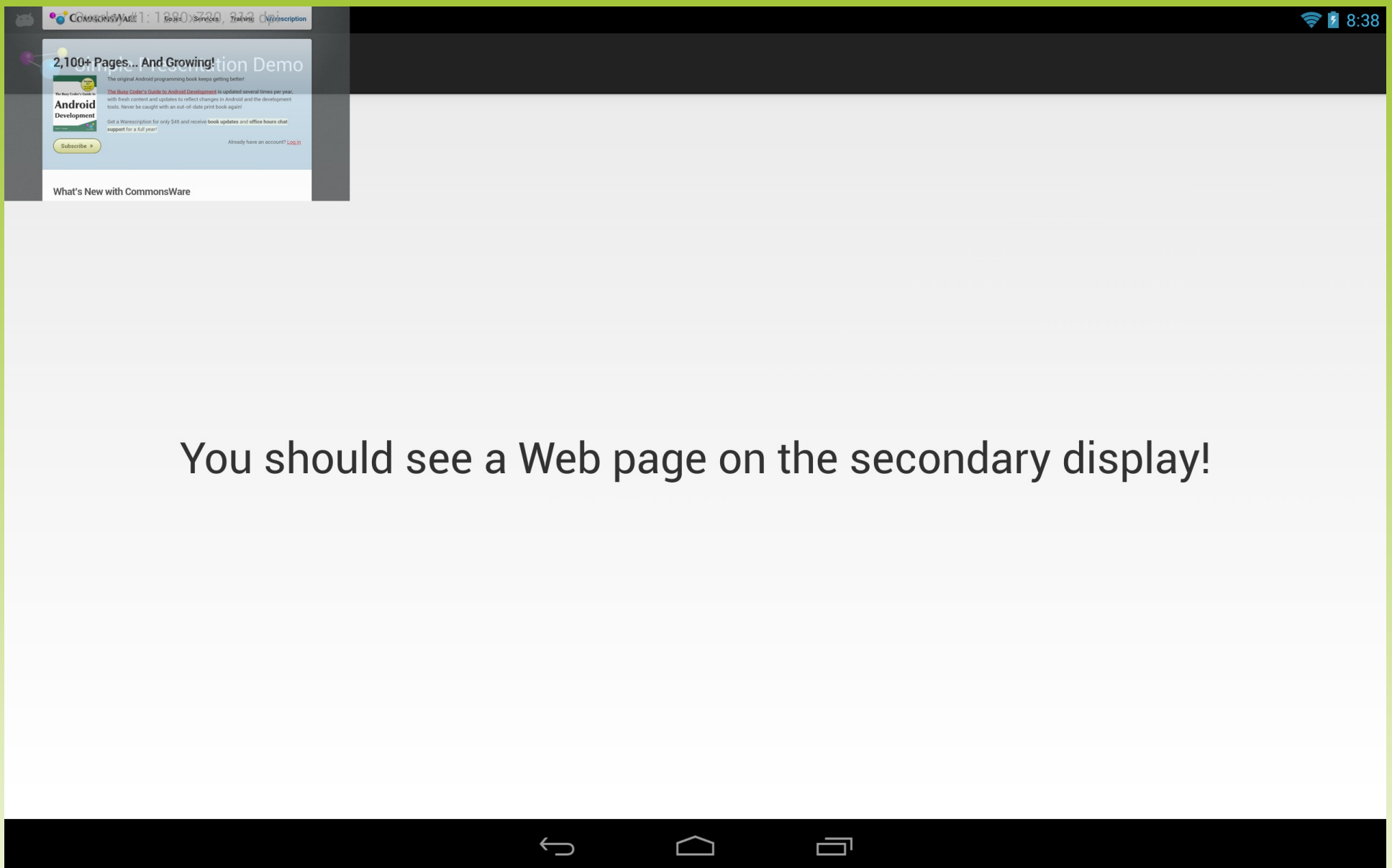
Trying It Out

- Testing Options

  - Actual "secondary screen"

  - Simulated secondary display

    - Developer Options → Simulate secondary displays

    - Works well with hardware, less so with x86 emulator

You should see a Web page on the secondary display!

You should see a Web page on the secondary display!

Low-Level Exertion

- `DisplayManager`

  – System service (`DISPLAY_SERVICE`)

  – `getDisplays()` (all or those in a category, like `DISPLAY_CATEGORY_PRESENTATION`)

  – `registerDisplayListener()` to find out about changes in mix of displays

  – New to API Level 17

Low-Level Exertion

- MediaRouter

  – System service (MEDIA_ROUTER_SERVICE)

  – More general: find preferred "route" for audio or video

  – getSelectedRoute() to find the route to be used right now (ROUTE_TYPE_LIVE_VIDEO)

  – addCallback() to find out route changes

  – Added in API Level 16

Fragments of a Presentation

- Good News!

  - `Presentation` extends `Dialog`, so `DialogFragment` works

  - `DialogFragment` supports either show-as-dialog or show-as-regular-fragment

- Bad News!

  - Since different `Contexts`, cannot use the same `DialogFragment` instance for each

Secondary Screen Strategies

- Secondary Screen: Not a Touchscreen

    - Whatever shows on secondary screen is driven and controlled by what is on the primary screen

- Example: Video Player

    - Second screen shows the video playback

    - Primary screen has play/pause, fast-forward, rewind, SeekBar, IMDB content, chat screen, etc.

Secondary Screen Strategies

- Must Have Secondary Screen

  – Presentation-specific app

- Optional, Dedicated UX

  – Game using second screen for primary output, device screen for control surface and secondary output

- Optional, Using Fragments

  – Push stuff to second screen or show "inline"

Secondary Screen Tactics

- Separate Fragments/Views
  - One for what is shown
  - One for controlling what is shown

- Mirroring Content
  - Same fragment/view, shown on both displays, controlled by primary screen

Limitations and Annoyances

- Activity-Based

  - Presentation extends `Dialog`, must have an activity to show a `Dialog`

  - Cannot have second screen driven by service, with primary screen running something independent

  - Samsung, multi-window apps, and secondary screens?

Limitations and Annoyances

- Activity-Based

  - Must `dismiss()` `Presentation` as part of switching to a new activity → flicker

    - Briefly shows mirroring before new `Presentation` appears

  - Similar flicker if you try switching from one `Presentation` to another

  - General rule: one activity and one Presentation, just updating content