Master Detail Flow

Erwin Sutanto, S.T., M.Sc.

Daftar Isi

1. Definition

2. The Process

3. Realisasi Design dengan UML
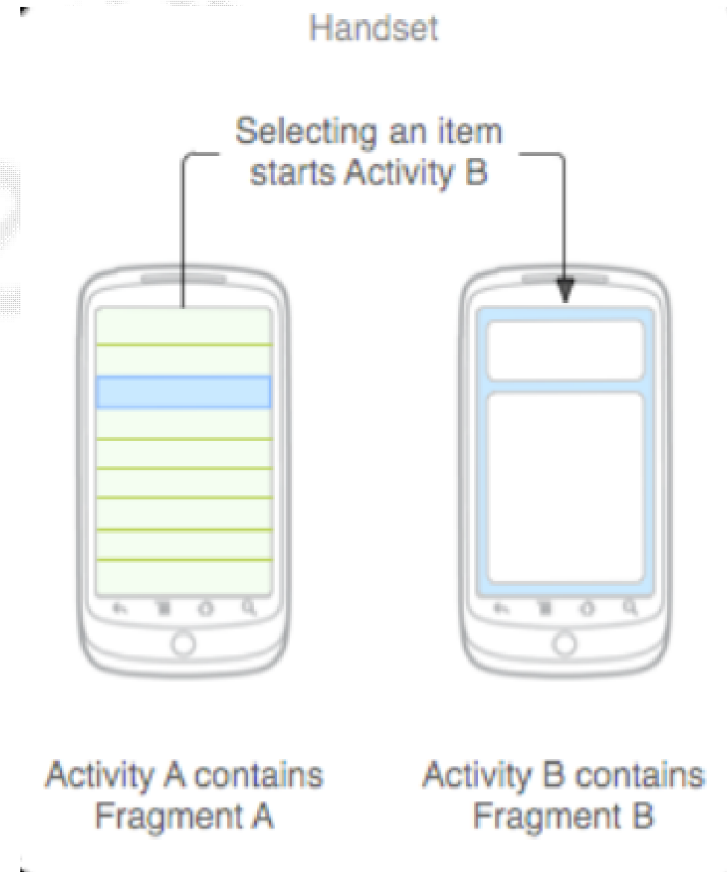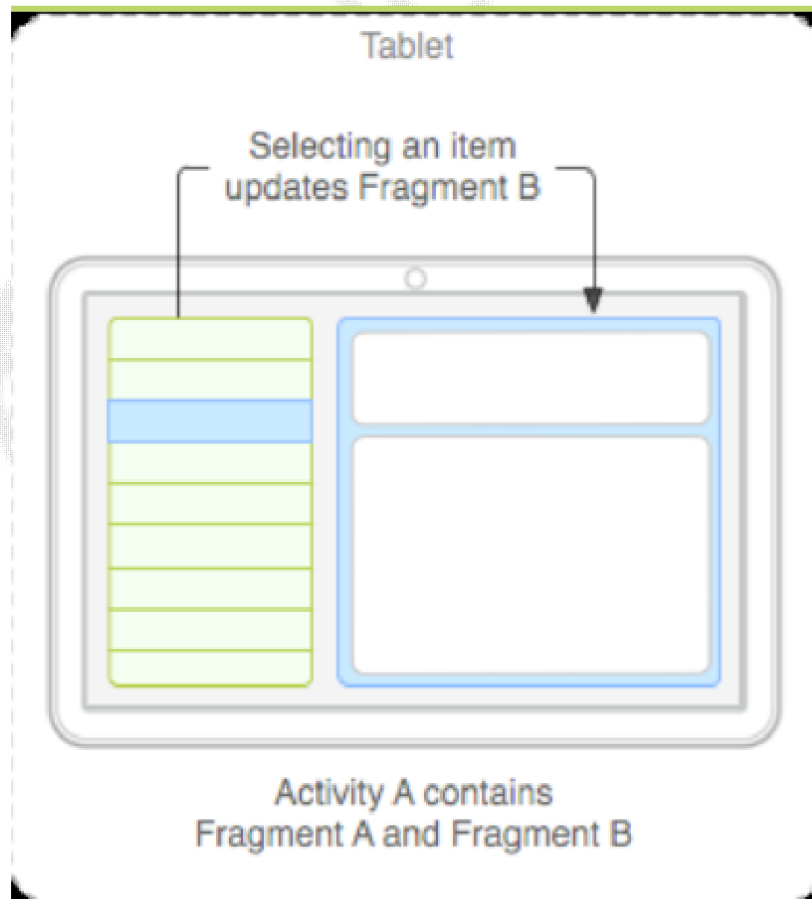
4. Other Possibility

1. Definition

- **Master/Detail Pattern**
  - Master = a collection of content
    - Title/display name
    - Other metadata
  - Detail = specifics about one piece of content
- **Pattern**
  - Show master or detail on smaller screens
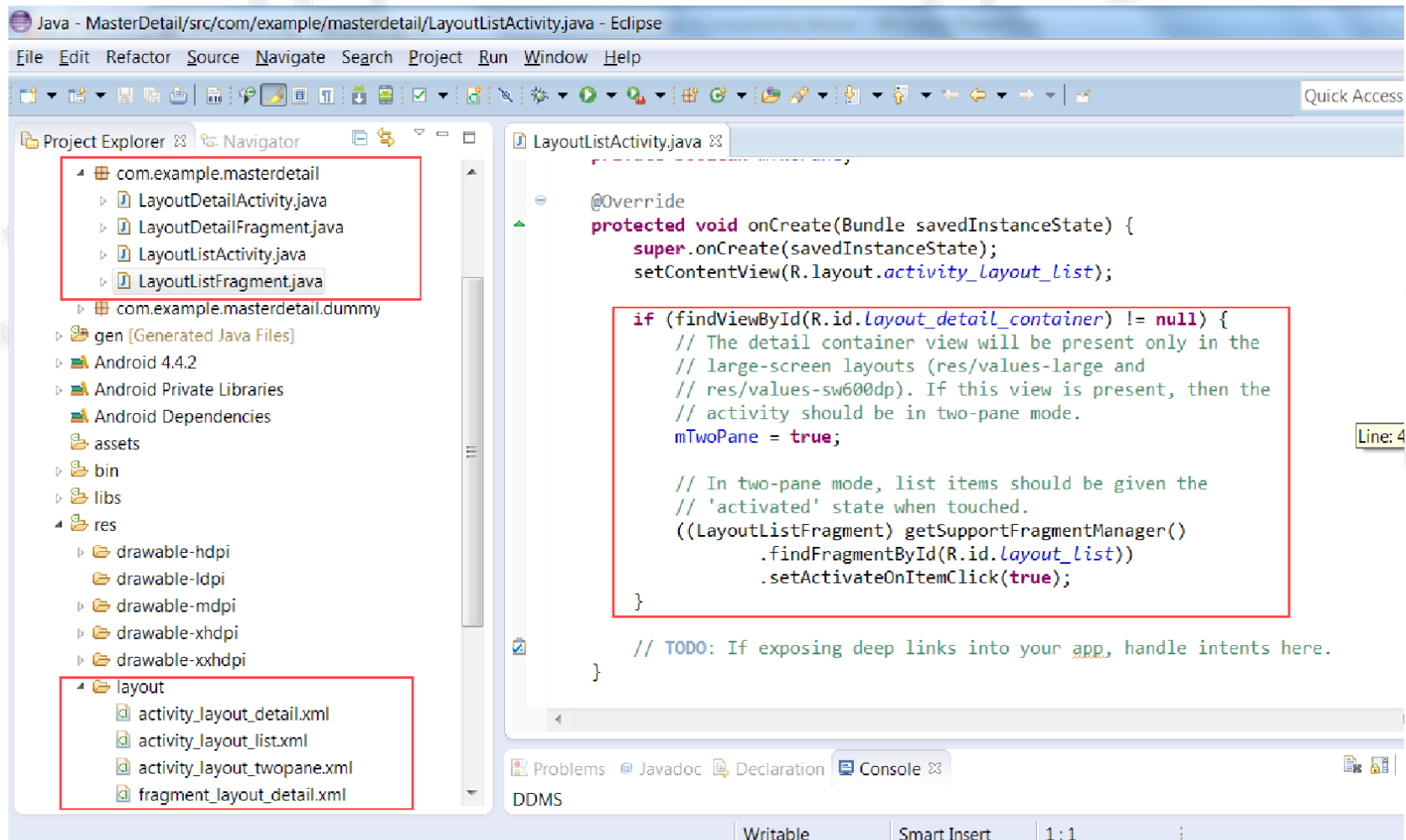 - Show master and detail on larger screens

Pattern

Tablet — Selecting an item updates Fragment B — Activity A contains Fragment A and Fragment B

Handset — Selecting an item starts Activity B — Activity A contains Fragment A — Activity B contains Fragment B

Project Template

Master/Detail Flow App Template

- **Option in Eclipse when creating a new project**

- **Results**

  ❑ Two activities (master/detail and detail-only)

  ❑ Two fragments (master and detail)

  ❑ Four layouts (plus a layout alias)

  ❑ Always single-pane or dual-pane, based on screen size

Project Files

2. The Process

- **One Activity/Two Fragments**
  - ✓ In single-pane mode, Activity executes a FragmentTransaction to replace list with detail
    - ❖ Adds to back stack to allow BACK to reverse the transaction and return to master
  - ✓ In dual-pane mode, just shows both fragments
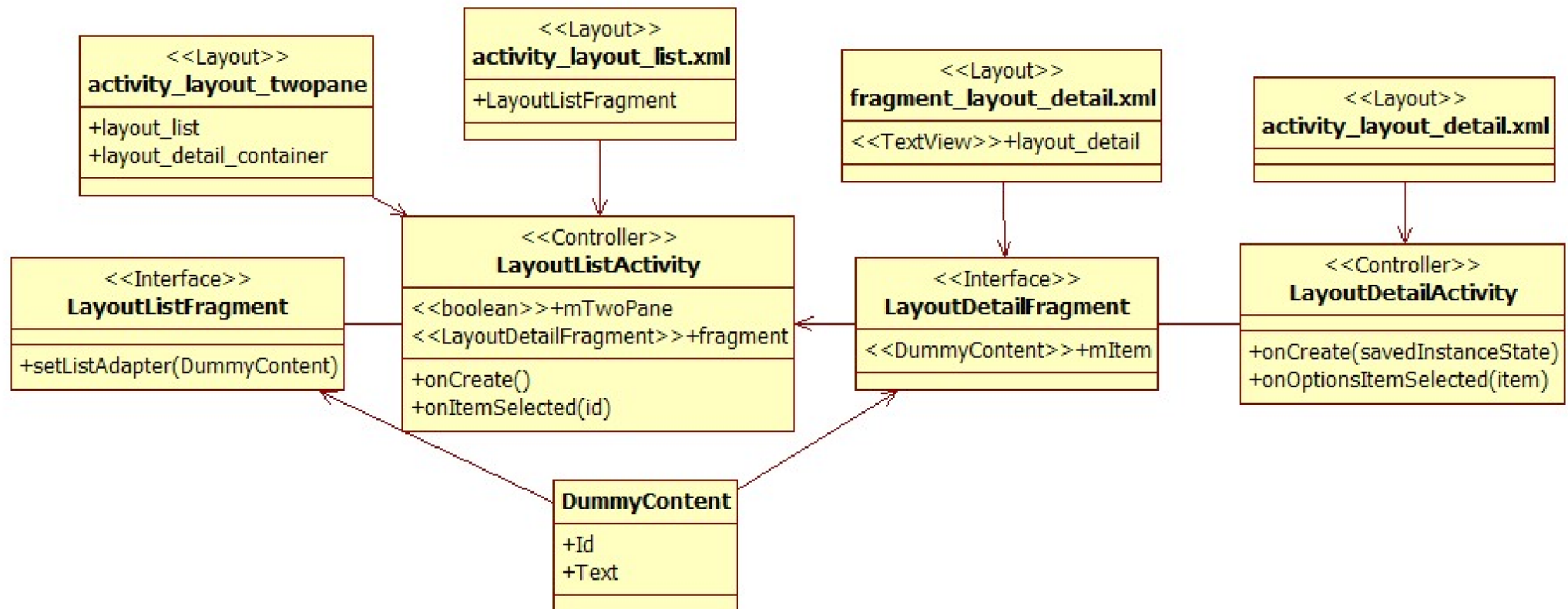
Pros & Cons

- **Pros**

  - ❑ One less activity (class, manifest, etc.)

  - ❑ Not limited by Intent extras
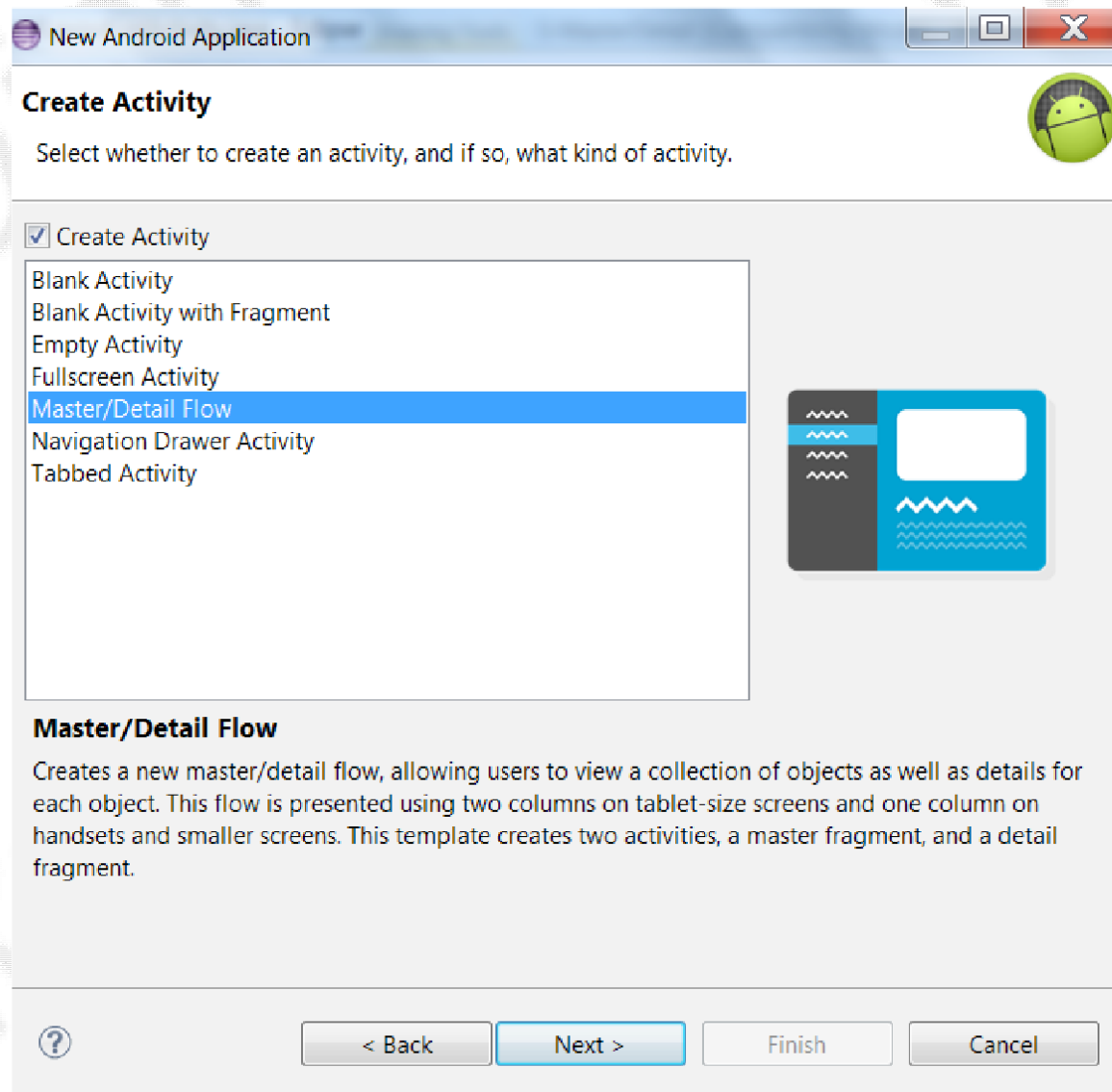
- **Cons: Too Many Scenarios**

  - ❑ Do we have a master fragment in the FragmentManager?

    - ✓ If yes, is it visible? Should it be?

  - ❑ Do we have a detail fragment in the FragmentManager?

    - ✓ If yes, is it visible? Should it be?

  - ❑ What happens when the user presses BACK?

  - ❑ What happens if we rotate the screen and change strategies?

3. Realisasi dengan UML

4. Other Possibility

4.1 Navigate Drawer Activity

- **SlidingPaneLayout**
  - ❑ Android Support package
  - ❑ Handles complexity for you
    - ✓ Single-pane vs. dual-pane
    - ✓ Swipe gesture to show, hide master in single-pane mode
    - ✓ All the configuration change idiosyncrasies

Pros & Cons

- **SlidingPaneLayout Pros**

  ❑ Simplicity

- **SlidingPaneLayout Cons**

  ❑ Confusion over swipe gesture

  ❑ Detail-first single-pane flow

  ✓ Can be fixed, requires a bit of extra code

4.2 Tabbed Activity

- **ViewPager**

  ❑ ViewGroup, attached to a PagerAdapter

  ✓ FragmentPagerAdapter, FragmentStatePagerAdapter, or custom implementation

  ❑ Horizontal swipes move between pages

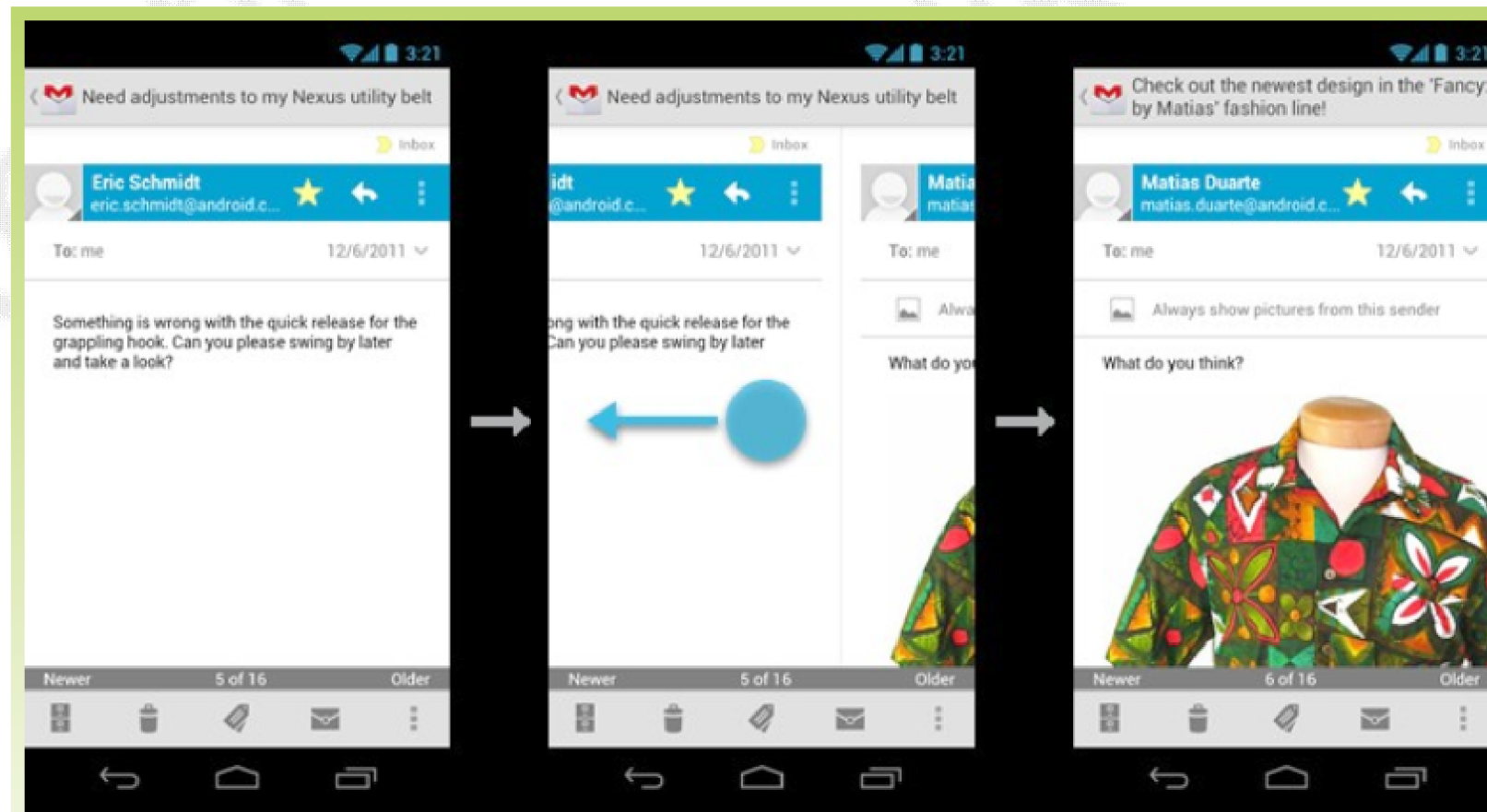 ❑ Optional "indicators" to show where the user is in the sea of pages

Design

ViewPager and Design

"On a phone, since the master and detail are on separate screens, this typically requires the user to jump back and forth between the list and the detail view, aka "pogo-sticking"."

"In cases where users will want to view multiple detail items in succession, avoid pogo-sticking by using the swipe gesture to navigate to the next/previous detail view."

View

4.3 Other Possibilities

- Switching Between Single-Pane/Dual-Pane

- Replaces regular action bar for contextual actions

- Action Modes and Master/Detail

- Multiple Flavors of Master List Highlighting

- Detail in Multiple-Selection?

- Retaining Action Mode and Selections

- Many more to explore....

- Thank You -

Referensi:

Erwin S., Pemrograman android dengan menggunakan eclipse & staruml, 2018, bab 9.