

Android Tips & Tricks

Joe Lallouz, Hashable

Agenda

- Preparing
 - Naming conventions
- Designing
- Developing
 - App Design Platform
 - App Design UI
- Testing
 - Reporting errors and user tracks
 - Tools
- Deploying
- Tips & Tricks
- Q & A

Introduction

Joe Lallouz - Android Developer @ Hashable

- Twitter: @JoeLallouz
- Email: joe@hashable.com

Intro

Hashable

- mobile, lightweight CRM
- featured app July & August 2011



Naming conventions

- make resources easier to find
- help organize a cluttered file structure
- avoid circular references

example:

icon_<iconname>

fragment_<classname>

activity_<classname>

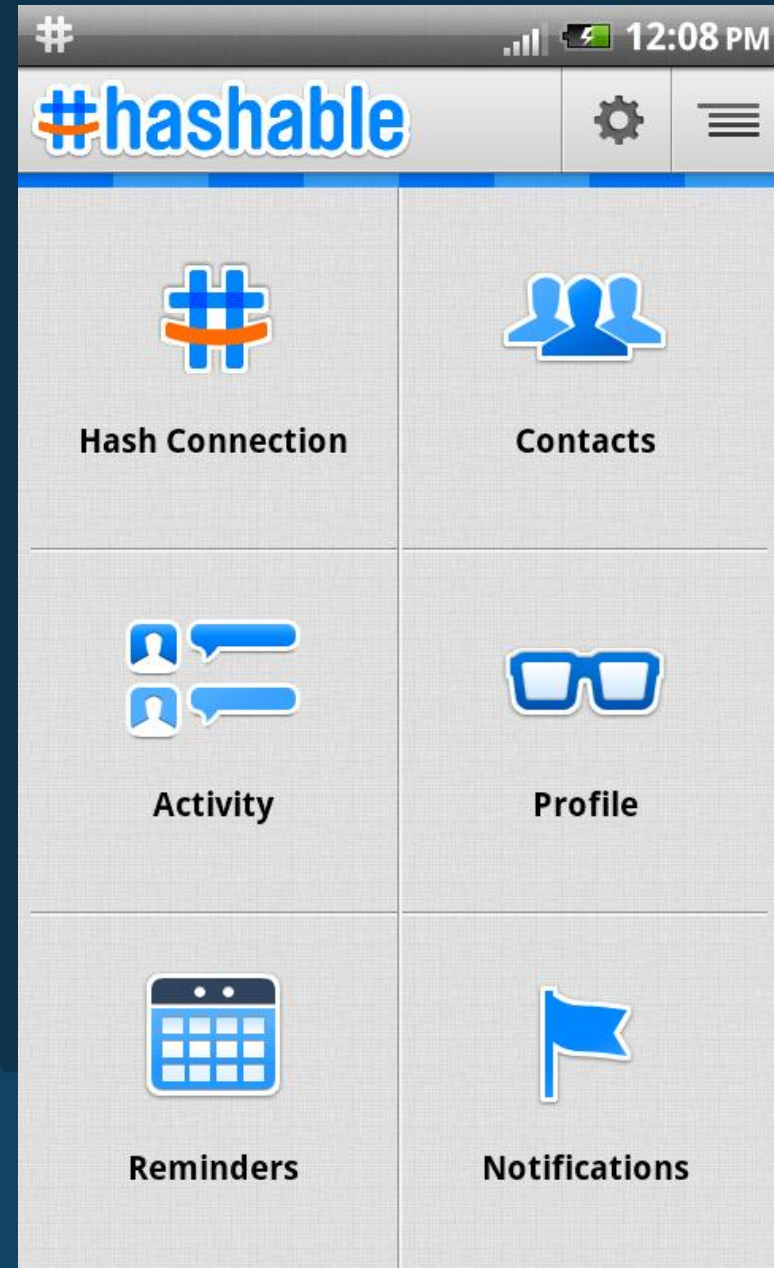
Application Design - UI

- Design for:
 - *Screen Densities & multiple sizes*
 - *Orientation switching*
 - *Touch input*

App Design - Dashboard

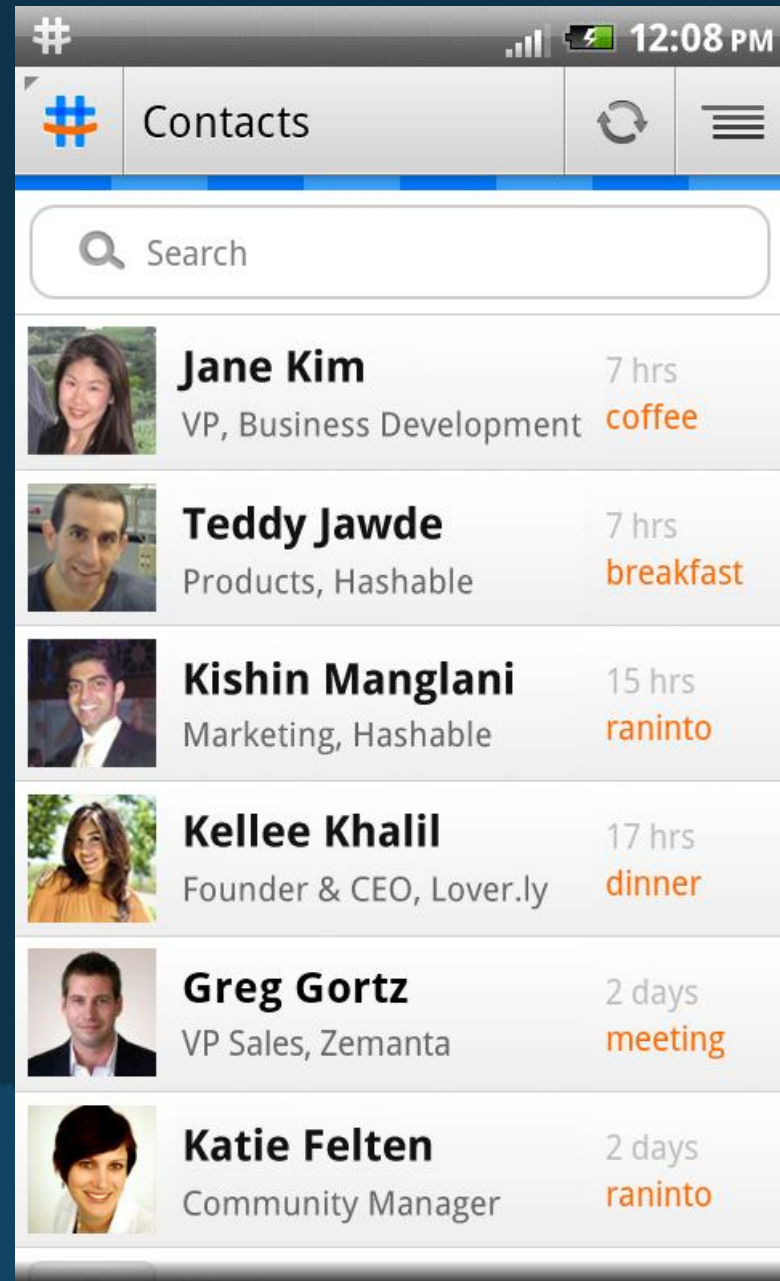
Dashboard

- Multiple entry points
- easy to navigate



App Design - Action Bar

- Important actions
- back functionality
- users awareness
- overflow actions
- tabs



App Design - Themes & Styles

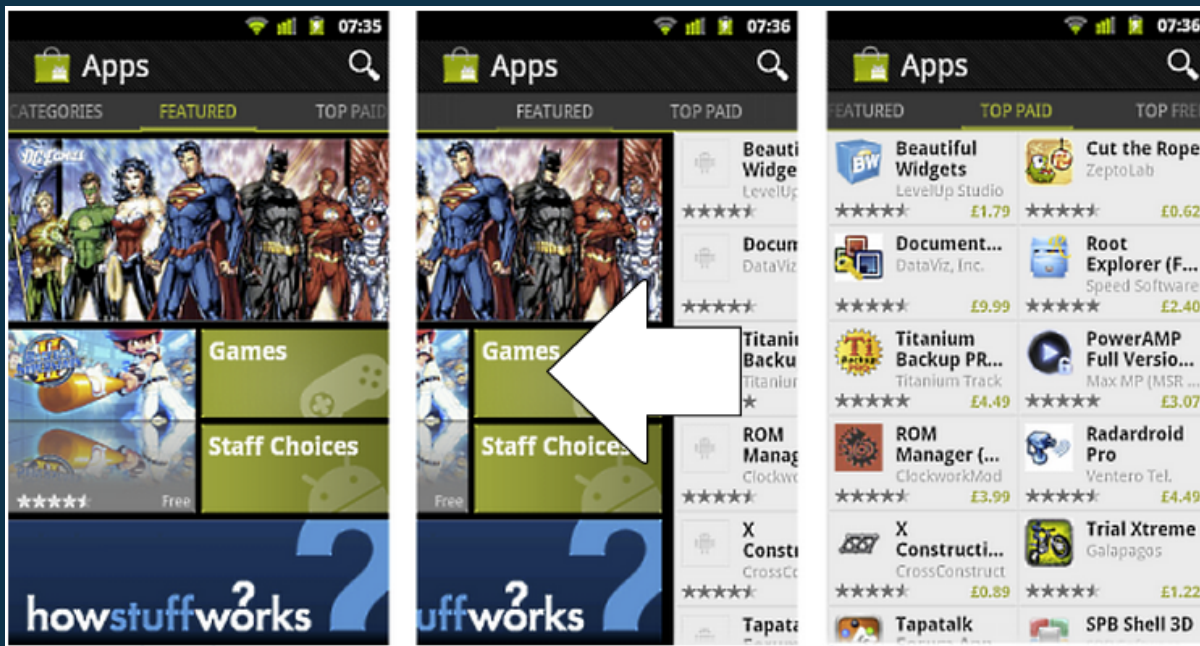
Themes & Styles

- extend parent themes
- reduce redundant XML

```
<style name="RequiredText.EditUser" parent="@style/RequiredText">
  <item name="android:singleLine">true</item>
  <item name="android:ellipsize">end</item>
  <item name="android:layout_centerVertical">true</item>
  <item name="android:textColor">#000</item>
  <item name="android:textSize">@dimen/text_size_smaller</item>
  <item name="android:gravity">center</item>
</style>
<style name="HeaderText.EditUser" parent="@style/RequiredText.EditUser">
  <item name="android:textSize">@dimen/text_size_mediumsm</item>
  <item name="android:textStyle">bold</item>
</style>
<style name="HeaderTextSmall.EditUser" parent="@style/RequiredText.EditUser">
  <item name="android:textSize">@dimen/text_size_smaller</item>
  <item name="android:textStyle">normal</item>
</style>
```

App Design - UI

- ViewPager



- ViewPagerPropertyAnimator

```
myView.animate().x(500).y(500);
```

Application Design - Platform

- Intents

- *Beautiful design pattern*
- *allows seamless app integration*
- *Open Intents*
- *register intent.ACTION_SEND and watch the usage*

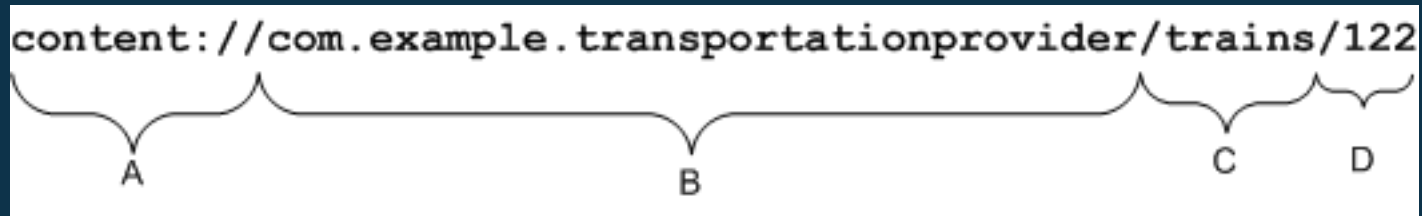
- *Register Intent Filters for your apps!*

Application Design - Platform

Content Providers

- Allow controlled access to your application's data
- Create maintainable api for other apps
- Use managed cursor loaders

```
content://com.example.transportationprovider/trains/122
```

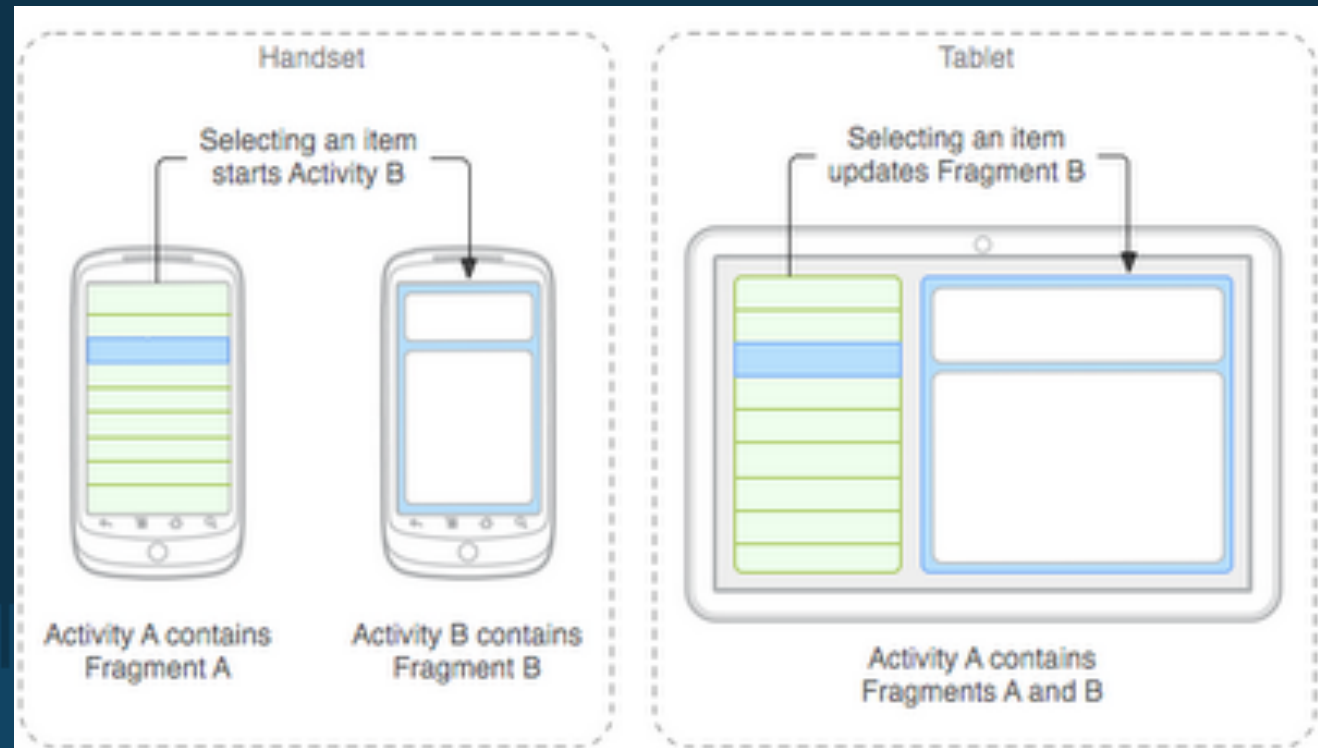


The diagram shows a content URI: `content://com.example.transportationprovider/trains/122`. Below the URI, four curly braces indicate segments: 'A' covers the scheme and authority, 'B' covers the path, 'C' covers the first path segment, and 'D' covers the final path segment.

Application Design - platform con't

- Fragments

- *separate out UI components*
- *portable*
- *re-usable*



Reporting with Flurry + Acra

- Test new features
- Test user paths
- Automatic error reporting
 - *listen to users*

- Android Market team is working on ways to interact with reviewers/raters.

Tools

Use the SDK tools to help create better apps that work across all devices. Test on actual devices!

- logcat - meow
- layoutopt
- hierarchyviewer
- Monkey
- Strict Mode

Tips & Tricks

- Develop and test on worst case devices
- Design for offline
 - Local (un-synced) -> API -> Local (synced)
- Design for orientation switching
- Design using touch, swipe and tap gestures
 - keep in mind d-pad and other input -- GoogleTV
 - multiple APKs
- Leave the UI thread alone

Tips & Tricks cont'd

- Beta test on the market
 - Easy update, Easy communication
- Be stingy with permissions
- Register company package -- `com.hashable.android`
- Use `Strings.xml` and translate
- Use Public API's
 - calendar content providers in ICS

Tips & Tricks cont'd

- Always assume your task can be interrupted
- Battery life
- Have fun

Q & A

@JoeLallouz
joe@hashable.com

Thanks

Roman Nurik (goog) * Jassim Latif (goog) * Adam Koch (goog)
Mike Novak (GroupMe) * Geoff Hacket (GroupMe)
Chris Haseman (Tumblr/DoubleTwist)
Aaron Henshaw (Hashable) * Nick Burton (4Sq)