



Android on tablet devices - lessons learned

Saša Slavnić

Agenda

- A little bit about project background
- Tablet specifics
- Porting an app
- Performance
- Q&A

Project background

- Relatively complex app
- TV program guide
- Companion device (app as a remote)
- Video playback
- Video on demand catalog (sport, vod)
- Personal recordings

Our task

- At the time we had Android phone app, iPhone app and iPad app developed
- Task was to bring Android experience on same level as iOS, by providing tablet support
- We had prior experience in porting iPhone to iPad (which proved to be more trouble than benefit)

Tablet specifics (1)

- Screen orientation
 - Unlike phones, tablets are supposed to support all screen orientations
 - Obvious consequence is that number of XML layouts in the app doubles
 - Less obvious consequence is that activities get re-created all the time, which can lead to serious performance degradation
- Screen sizes
 - UX on 7" device is not the same as on 11" device
 - So more XML layouts are needed (dimensions help a lot)

Tablet specifics (2)

- Handling phablets
 - Decision to support phablets can lead to some more XML layouts, so be aware when you make decision
- Why is number of layouts so important?
 - Implementation cost
 - Maintenance cost
 - They are very prone to errors (e.g. forgetting to wire up one UI element in one of layouts leads to NPE)
 - Testing cost (a lot of regression defects are introduced)

Tablet specifics (3)

- More things on screen at once
 - Shifting mindset from “one task per screen”
- More content
 - More screen real estate means more images

Porting an app (1)

- Moving from activities to fragments
 - Fragments are recommended way of doing things anyway
 - If you do not deal with legacy code, you maybe do not have this problem
- Remove onResume() data initialisation
 - Usual place where apps are initialised with data
 - It creates problem when you present fragment later in the activity, that changes data or app state

Porting an app (2)

- Removing `onActivityResult()`
 - Obviously, if you had activities in your project you will need to get rid of activity centric navigation
- Callbacks for data retrieval
 - Having callback methods for data retrieval does not work that well when you have multiple things going on in one screen
- Broadcasting on data change
 - Both things above can be sorted out with making broadcasting mechanism

Performance (1)

- Wide range of devices (performance-wise)
 - There is some really exotic hardware out there
 - Good thing is that Samsung is considered as baseline device
 - Bad thing is that there are lot of issues with Samsung devices
- Newer devices are not always faster
 - We had some really quirky behaviour with scrolling on Nexus7
 - Galaxy Tab 10.1 outperformed Galaxy Tab 10.2 in many cases

Performance (2)

- Orientation performance
 - Long story short - its bad
 - Responsiveness of orientation sensor is not consistent across devices
 - Keep in mind that in many cases layout needs to be re-created, so be careful what you do onResume
- Memory consumption
 - More content on screen means more memory consumption
 - Tablet apps often display more images

Conclusion

- Moving app from “activity centric” to “fragment centric” was quite straightforward
- A lot of time was spent on performance tuning
- And a lot more on memory related issues
- We underestimated testing effort, coming out of so many different layouts



Questions?

Saša Slavnić

001@retrocode.com

rs.linkedin.com/in/sasaslavnic/