Localizing phone and tablet apps: Why and How

Jose Palomares, OpenMints
A time of revolution

- The revolution of the “where”
  - Boom of the Internet. From local to global.
- The revolution of the “who”
  - The web 2.0. From spectators to creators.
- A revolution of the “when”
  - Just anytime.
First lesson: Apps are not applications
Why ‘apps’ are different?

Typical desktop applications
• Built on specific frameworks
• Customized workflow
• Long life
• Deep testing needed
• Slow turnaround
• Higher overhead
• Big money investment

Mobile apps
• Common framework
• Standardized workflow
• Short life
• Need less testing
• Fast turnaround
• Low overhead
• Moderate money investment
Inside the client’s mind

Localizing our app would allow us to...

– Sell more
– Sell to new countries
– Sell to new audiences
– Grow our brand
– Improve user experience
Inside the vendor’s mind

Knowing how to localize apps…
– Allows us to sell more
– Creates new business opportunities
– Differentiates us from our competitors
– Grows our brand
– Improves user experience
– Eventually, it might even help us to make money
Mobile OSs

Windows Phone

palm

symbian OS

Blackberry

Android

webOS

bada

iOS

Windows Mobile
Mobile OSs

Windows Phone
Palm
Symbian OS
Blackberry
Android
webOS
Bada
iOS
Windows Mobile
What to localize?

- Graphic User Interface files
  - Resource files
    - .RRC, .XIB, .NIB, .AFK, script/xml-based...
  - Strings files
    - .STRINGS, .TXT, .XML...
- Images
- Content files
  - Databases, external repositories, public data, etc.
- Help/UA
BlackBerry

• .RRH, .RRC files

• Very limited space

• Absolute / relative X-Y

• Limited number of screen sizes
Android

- More space, tactile elements
- Variable layout
- Editable attributes
  - Singular / plural strings
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="vertical"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
>
<TextView
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:text="Hello World"
/>
</LinearLayout>
iOS

- .NIB, .XIB files
  - To get it right, you’ll need a Mac (but you might be able to resize)
- .STRING files
  - “TextMessageBlocked”=“A text message has been blocked”;
- .PLIST and .XML files
  - It’s all about XML
Android & Blackberry: “One UI can’t rule them all”

• Too many devices = Tough decisions
  • Portrait / Landscape
  • Fixed XY position / Relative XY position
  • Images:
    – BB: pre-defined or one per screen size
    – Android: unique or “nine-patch” images
The iOS advantage:
Tailor-made user experience
Tools of the trade

- No magic solution needed (Notepad++?)
- Any CAT / TEnT that allows to customize filters
  - Okapi Tools / OmegaT
- Localization suites
  - Alchemy Catalyst: Android, iOS
  - Multilizer: Android
- SDL Passolo: Symbian, Palm, iOS, Android
As easy as 1-2-3

- Localizing an Android application with Alchemy Catalyst 10
Considerations: Language

- String length
  - Bear in mind all potential target devices
  - One fits all?
- Beware of abbreviations
- Style and terminology
- Consistency vs. User Experience
- Tone - Who is the intended audience?
- Age, gender, affiliation, wealth/social position
Considerations: Testing

- Simulators
  - Included in the SDK, free of cost
  - Restricted number of hardware configurations
  - Behavior is not genuine
  - Allow some automation
- Real devices
  - Not free
  - Limited number of hardware configurations
  - Real use case
Developer’s “black list”

- “The application is only 430 words, but we need 8 languages, so please consider a volume discount”
- “We will be sending you an update every other Thursday, that needs to be back by Friday EOB”
- “We will create an export containing only the relevant strings, as not all of the text needs to be localized”
- “Any additional text will be sent on a separate e-mail”
- “You will get a fully working testing build on Monday”
- “If you need any context, just ask”
Dear Developer,

- “Please send your native files, we will process them, leverage text, extract terminology and send you a quote”
- “Please comment your strings. Comments provide invaluable context and save you a lot of time and money
- “Let’s work together to define a handoff/delivery schedule that is feasible on both ends
- “Please try to deliver fully functional builds if possible and appoint someone who can answer queries fast”

Yours sincerely,

Your loyal localization partner
Why bother?

- 400M+ Android devices, 400M+ iOS devices worldwide
  Only in Europe, 860M+ smartphones
- Expansion of low-end tablets = increase of content consumers
- Nokia and Microsoft Windows Phone and Windows 8 to rise in 2013
- User experience depends on it
Why bother?

- 700,000+ apps in Apple’s App Store
- 675,000+ apps in Google Play
- All categorized by type and success.
- Less than 10% are localized. Many are Google Translated.
- 40+ languages, powerful monetization
- 60,000 M+ downloads,
  = a HUGE business case.
- Easy to do. No client too small.
How to make it profitable

- Forget about wordcounts, offer per hour or fixed rates
- Forget about resizing
- Automate text processing
- Focus on testing
- Focus on “collaterals”
  - Marketplace metadata
  - Help, documentation, web
- Enhance user experience
Thanks for your attention!

Jose Palomares
jose@openmints.com
twitter: @localizing