

Android Development

Larva Labs

Background

- ✦ First game written on Nokia 3100 in 64k
- ✦ Switched to Sidekick with music apps
- ✦ Now on iPhone and Android



Some of our Apps



Demo

Our Android Experience

- ✦ Started with SDK when in beta
- ✦ Now have 3 games and 1 app
- ✦ Retro Defense was number 1 in the paid games, currently 2
- ✦ Need to do a “lite” or free version
- ✦ Definitely not iPhone money

Android Market Issues

- ✦ No screenshots
- ✦ Tiny description text (325 chars)
- ✦ Hard to find sorting options
- ✦ Fairly static rankings
- ✦ Credit card / Google checkout billing
- ✦ Refunds
- ✦ Copy Protection

Common Support Topics

- ✦ Problems due to phone customization
- ✦ Credit card problems
- ✦ Refund requests
- ✦ Game suggestions

Android Pros

- ✦ Background processes
- ✦ Instant approvals
- ✦ Touch + Keyboard + Trackball
- ✦ Replaceable system components
- ✦ Java
- ✦ Less competition than iPhone
- ✦ Many devices
- ✦ Open source

Android Cons

- ✦ Background processes
- ✦ Instant approvals
- ✦ Touch + Keyboard + Trackball
- ✦ Replaceable system components
- ✦ Java
- ✦ Less competition than iPhone
- ✦ Many devices
- ✦ Open source

Android vs. iPhone

- ✦ iPhone SDK specific to one device type, Android more general
- ✦ iPhone UI builder is great, Android has no UI builder
- ✦ UI more attractive and more responsive on iPhone
- ✦ Objective-C vs. Java
- ✦ iPhone has single app model, limited notifications
- ✦ iPhone has low level audio and video access, Android more limited

A Few Tech Tips

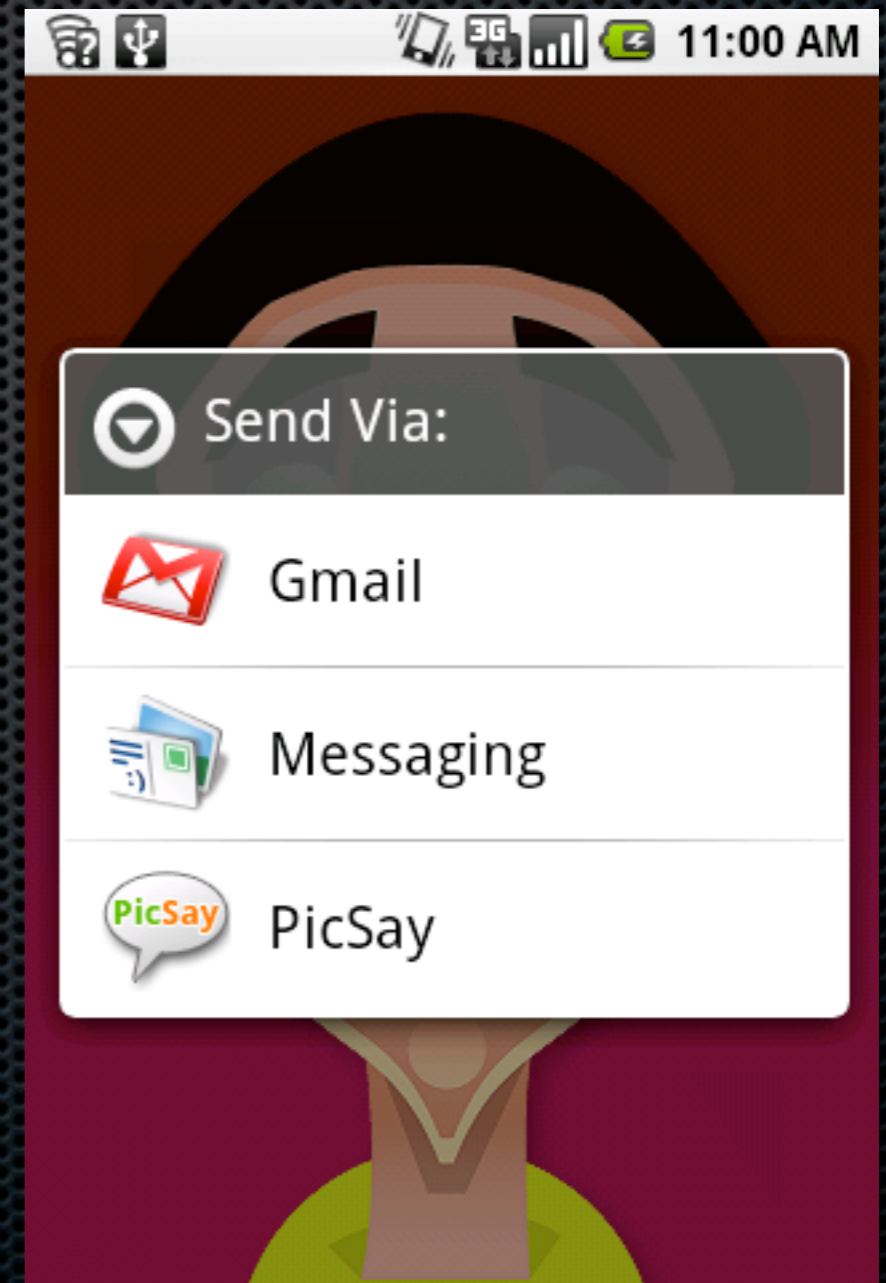
- ✦ There is always some hard-won knowledge when trying to get software to market on a new platform.
- ✦ Here are some tips that we've picked up along the way.

Game Audio

- ✦ There is a class called SoundPool for high-performance game audio, but it is unstable, can't use it.
- ✦ The MediaPlayer class can play audio, but it is slow to initialize, results in audio latency.
- ✦ We had to create a pool of MediaPlayer objects, one for each game sound.
- ✦ Also, only Ogg Vorbis files have good performance! Everything else has high latency.

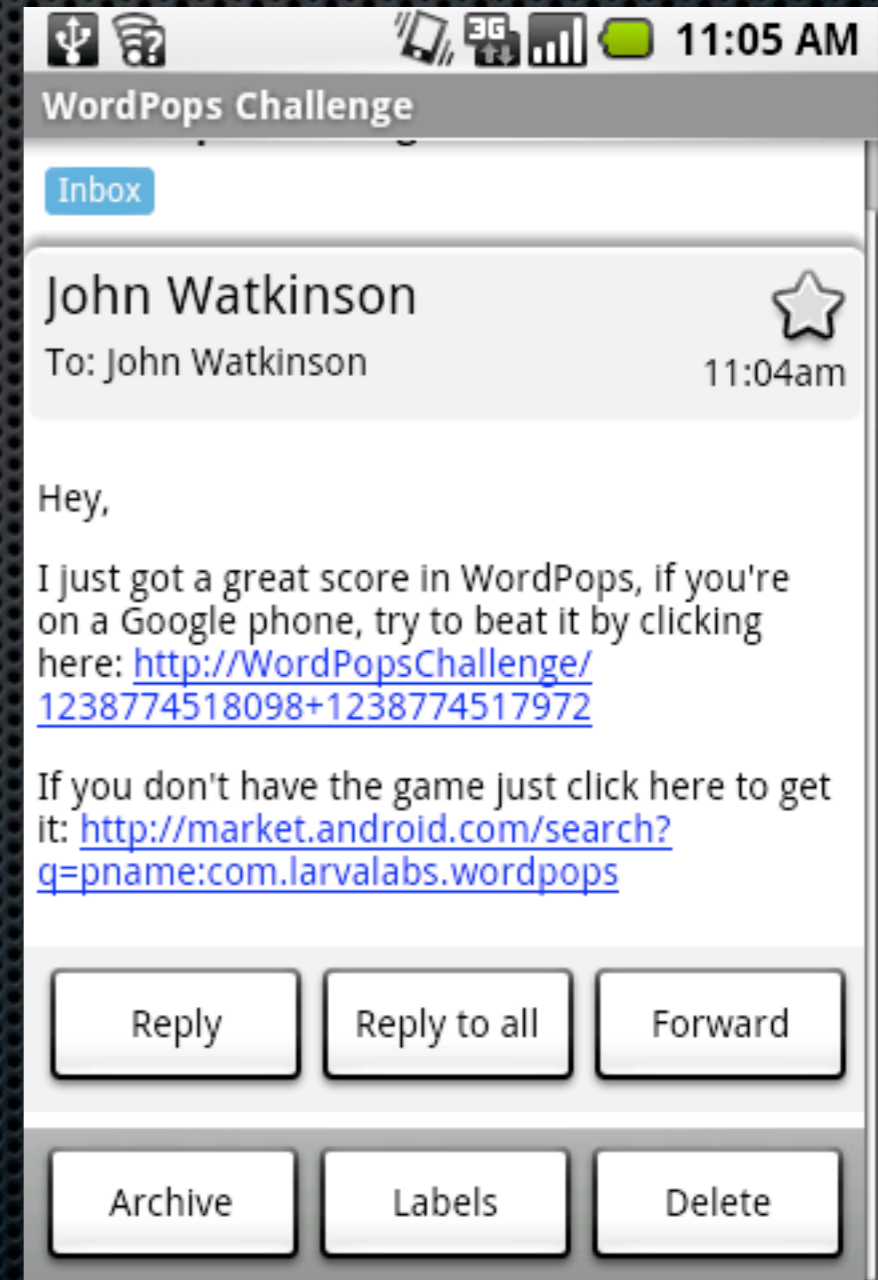
Sending a Photo

- ✦ We create an ACTION_SEND event, which can be handled by Gmail, Email, MMS, etc.
- ✦ The action includes an EXTRA_STREAM URL. This URL points back to photo content in our app.
- ✦ The sender (eg. Gmail) then requests the content. Our app must behave like a database, replying to content queries.



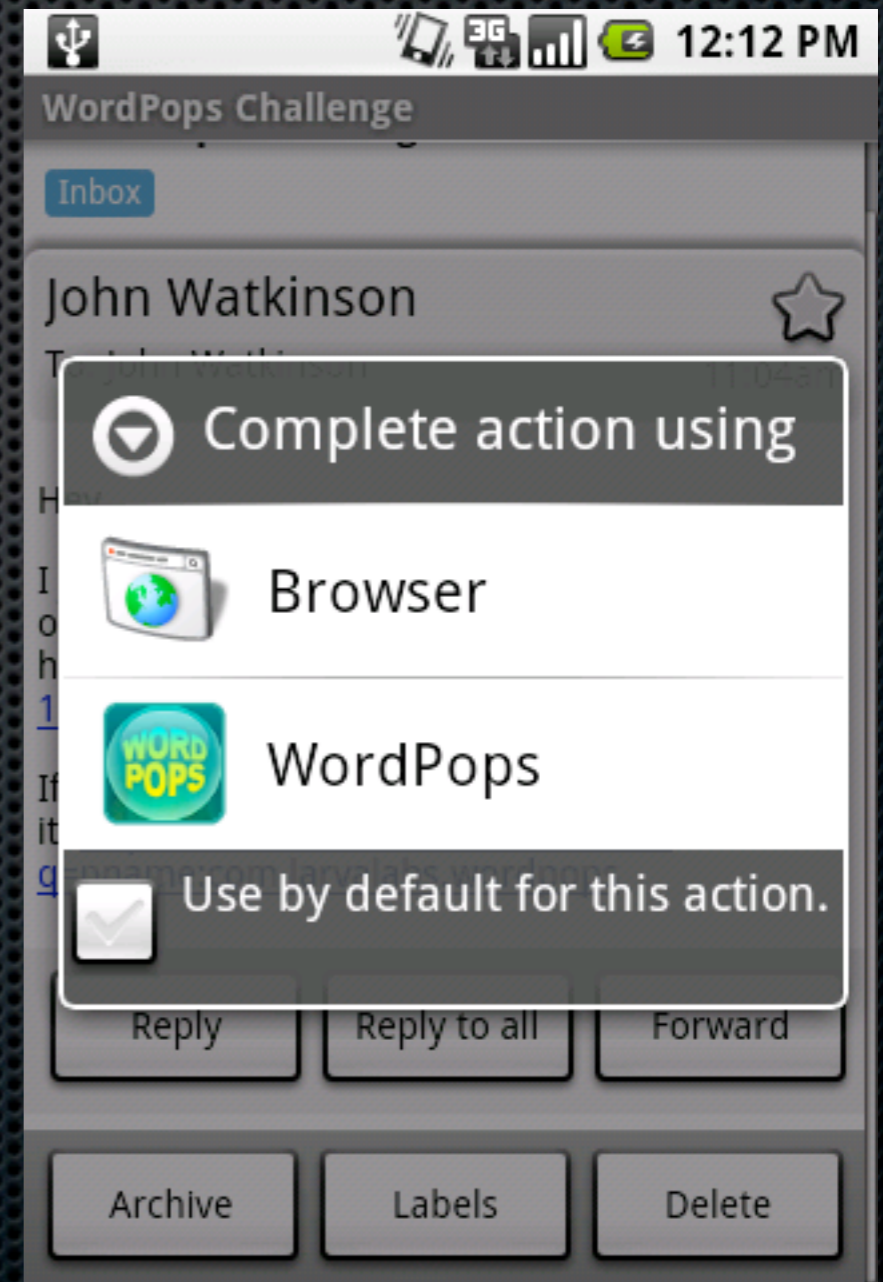
URL Intent Intercept

- ✦ We wanted players of Word Pops to be able to challenge each other via email.
- ✦ We were able to intercept certain URLs with our app and handle them ourselves.



URL Intent Intercept

- ✦ We wanted players of Word Pops to be able to challenge each other via email.
- ✦ We were able to intercept certain URLs with our app and handle them ourselves.



Orientation Issues

- ✦ Whenever the G1 keyboard is opened/closed, the default behavior is to **restart** the activity (app) entirely!
- ✦ There is debate about whether this is insane or not.
- ✦ We usually override it and do nothing in our games.
- ✦ In Word Pops we allow the user to enter a name for a high score, which requires an orientation change. We do the high score entry in a separate activity.

Gesture Detection

- ✦ A very useful tool that can detect long-presses, flings, etc.
- ✦ These are not easy to detect on your own, and having standardized detection will result in a more consistent experience for the user.

App Size Concerns

- ✦ There are not very strict limits on app size.
- ✦ However, most devices have limited space for apps, especially copy-protected apps.
- ✦ We try to keep our apps fairly small, mostly by using PNG optimizers.