HOW TO DESIGN YOUR MOBILE APPS

HP HELSINKI

11 APRIL 2013

Rev: 15 April 2013

JULIAN HARTY
Contact me: julianharty@gmail.com

INTRODUCTION

SETTING THE CONTEXT

OUR CONTEXT

Over 1,300,000 apps available for Android and iOS

Selling apps is like a fickle lottery

$$$ Billions in revenue

Devices are ubiquitous

- from babyhood to dotage

Rich interfaces and capabilities

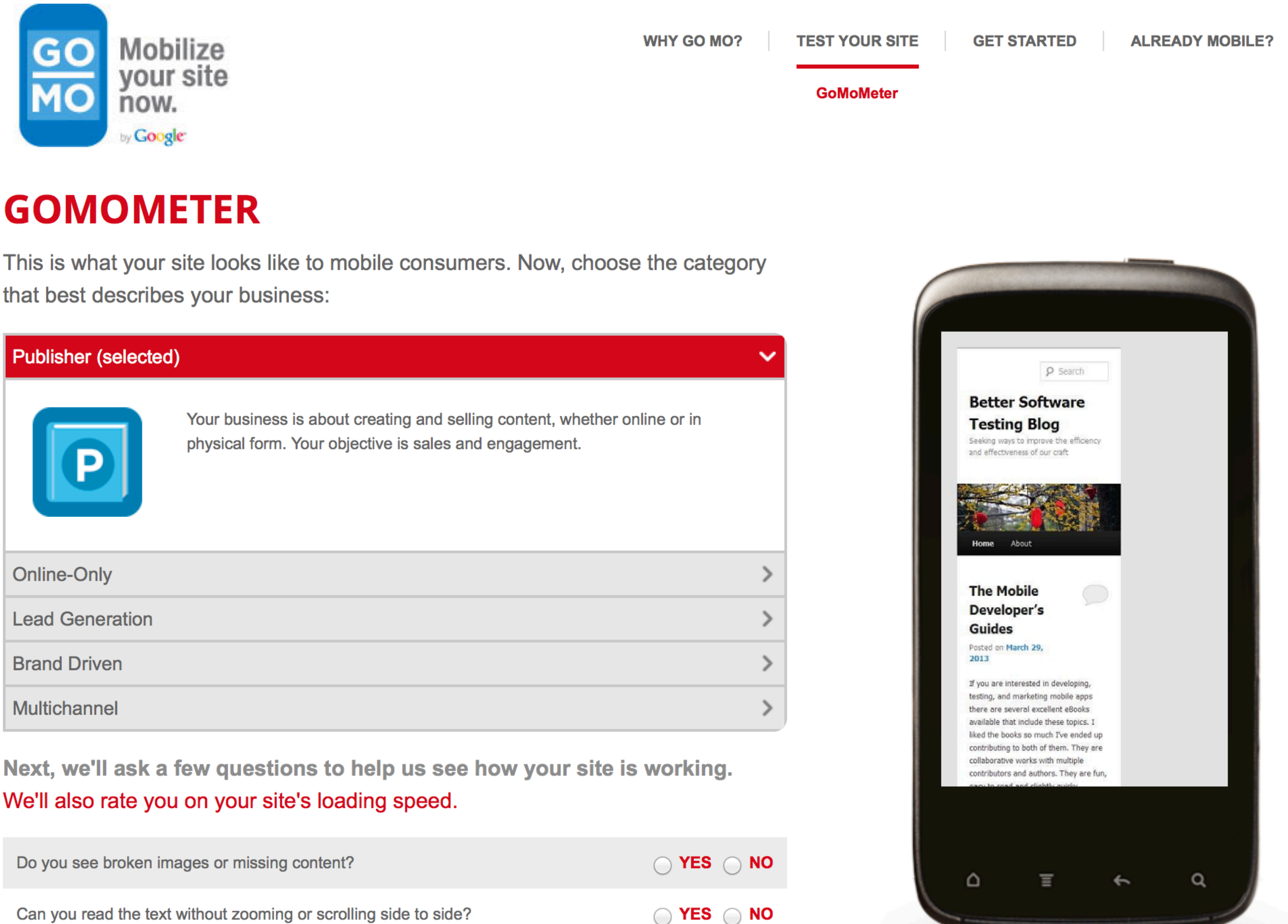
10+ smartphone platforms in use

GOALS FOR OUR APPS

What are the aims and purposes of the mobile apps?

- Branding?
- Presence?
- Satisfy current users, who now use mobile devices?
- New & Additional revenues?
- New business models?
- New forms of usage?

http://www.howtogomo.com/en/d/test-your-site/#gomo-meter



GOMOMETER

This is what your site looks like to mobile consumers. Now, choose the category that best describes your business:

Publisher (selected)

Your business is about creating and selling content, whether online or in physical form. Your objective is sales and engagement.

- Online-Only
- Lead Generation
- Brand Driven
- Multichannel

Next, we'll ask a few questions to help us see how your site is working.

We'll also rate you on your site's loading speed.

Do you see broken images or missing content? ○ YES ○ NO

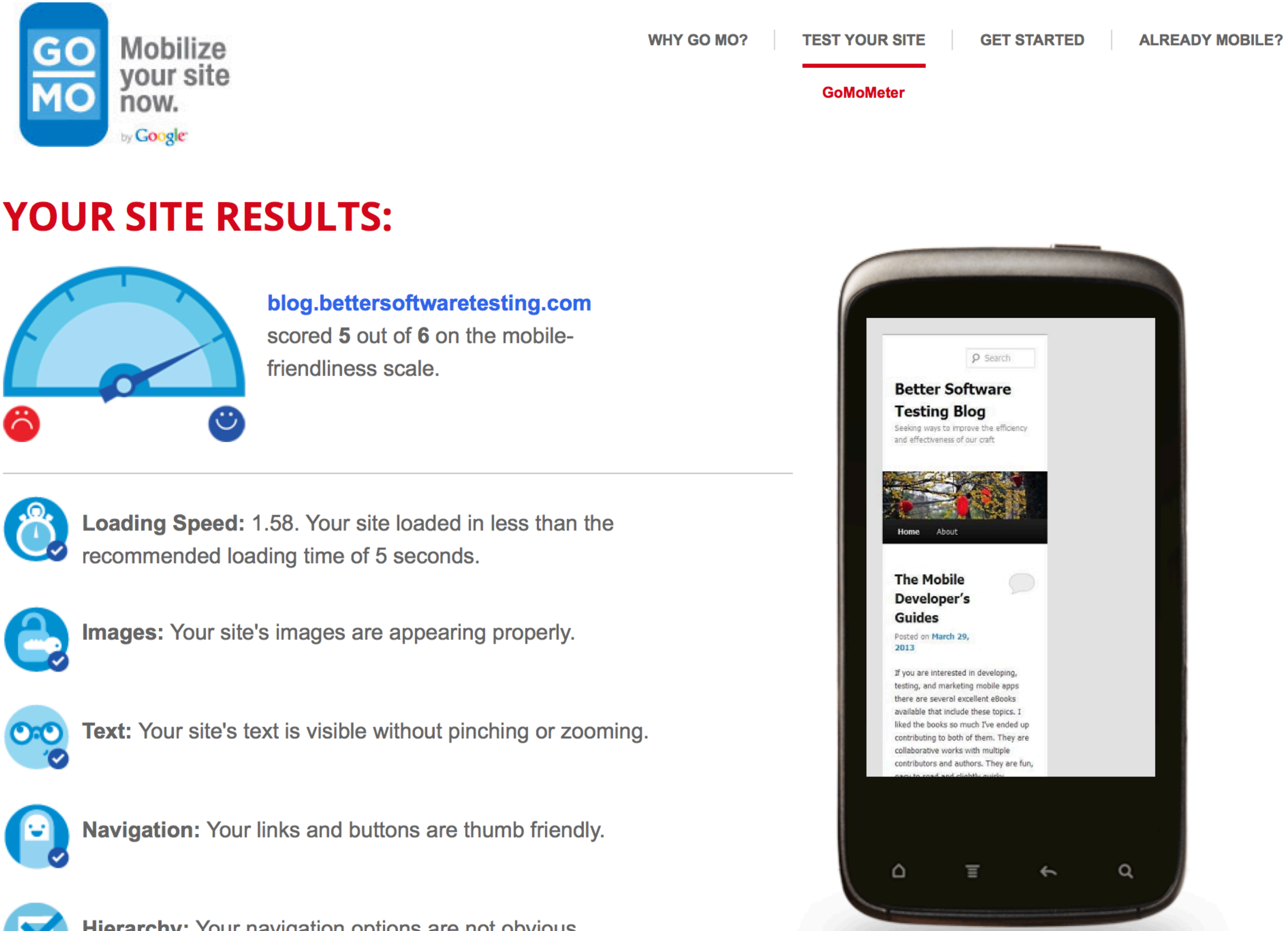
Can you read the text without zooming or scrolling side to side? ○ YES ○ NO

Search
Better Software Testing Blog
Seeking ways to improve the efficiency and effectiveness of our craft
Home
About

The Mobile Developer's Guides

Posted on March 29, 2013

If you are interested in developing, testing, and marketing mobile apps there are several excellent eBooks available that include these topics. I liked the books so much I've ended up contributing to both of them. They are collaborative works with multiple contributors and authors. They are fun, http://www.howtogomo.com/en/d/test-your-site/#gomo-meter



YOUR SITE RESULTS:

blog.bettersoftwaretesting.com scored **5** out of **6** on the mobile-friendliness scale.

Loading Speed: 1.58. Your site loaded in less than the recommended loading time of 5 seconds.

Images: Your site's images are appearing properly.

Text: Your site's text is visible without pinching or zooming.

Navigation: Your links and buttons are thumb friendly.

Hierarchy: Your navigation options are not obvious

Search

Better Software Testing Blog

Seeking ways to improve the efficiency and effectiveness of our craft

Home About

The Mobile Developer's Guides

Posted on March 29, 2013

If you are interested in developing, testing, and marketing mobile apps there are several excellent eBooks available that include these topics. I liked the books so much I've ended up contributing to both of them. They are collaborative works with multiple contributors and authors. They are fun, easy to read and slightly quirky

PLATFORMS AND TECHNOLOGIES

- **iOS**
- **Android**
- **HTML5 & Web Apps**
- **Others?** (Windows Phone 8, BlackBerry 10, Nokia Ashi, etc)

Do we want cross-platform or separate apps?

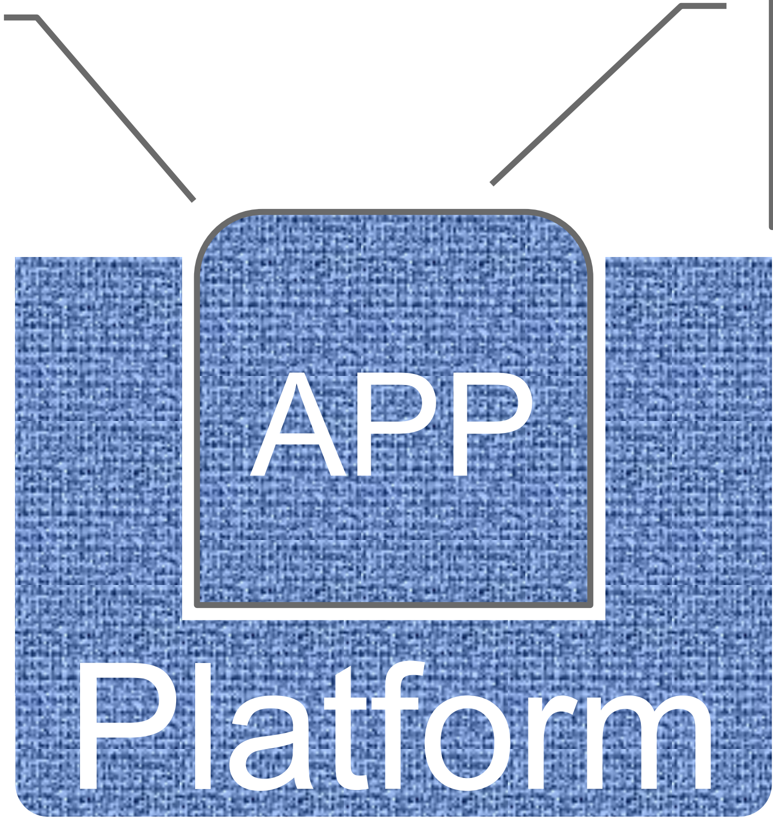
DESIGN CONSIDERATIONS

TYPE OF APP

NATIVE APP

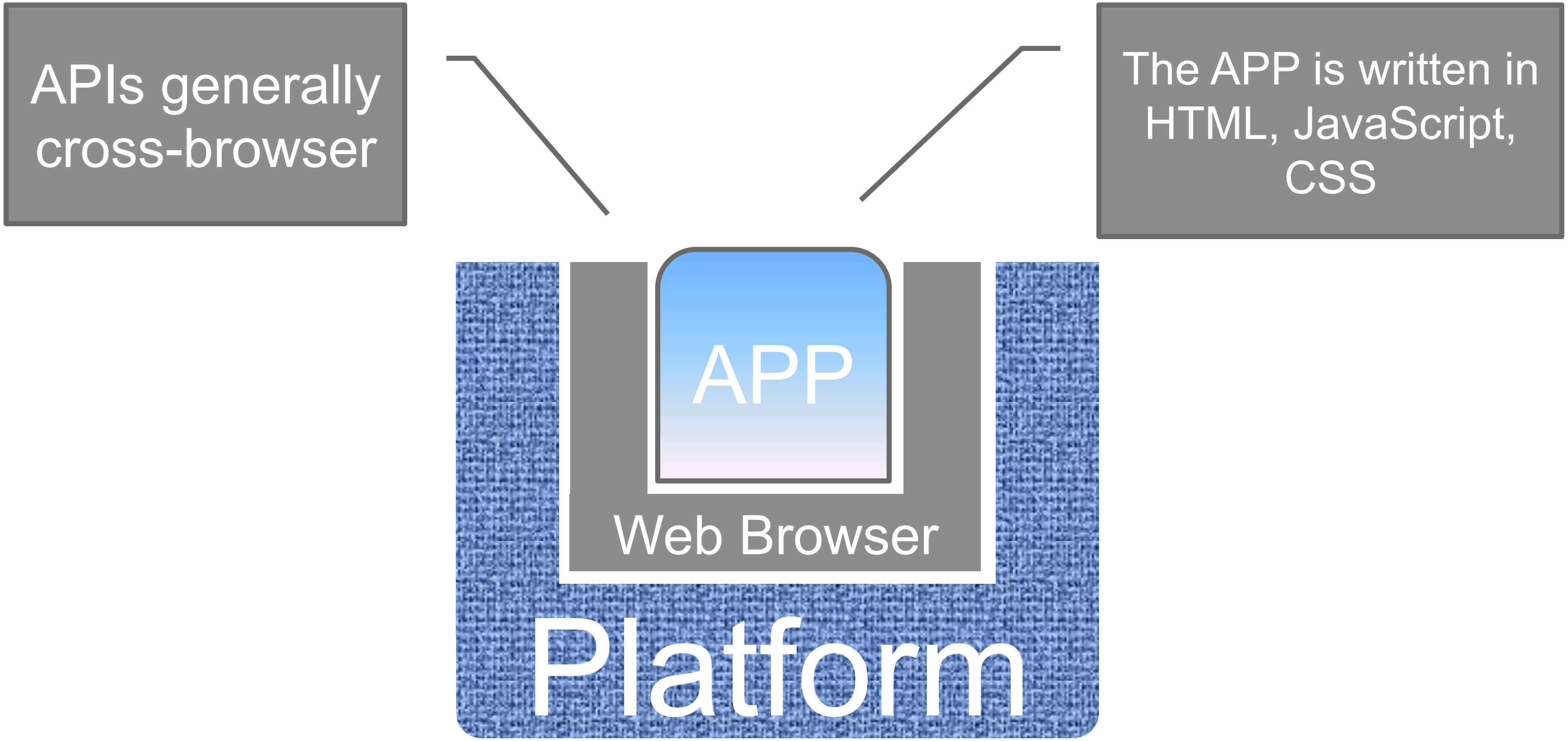
APIs are specific to the platform

The OS and the APP share a common programming language



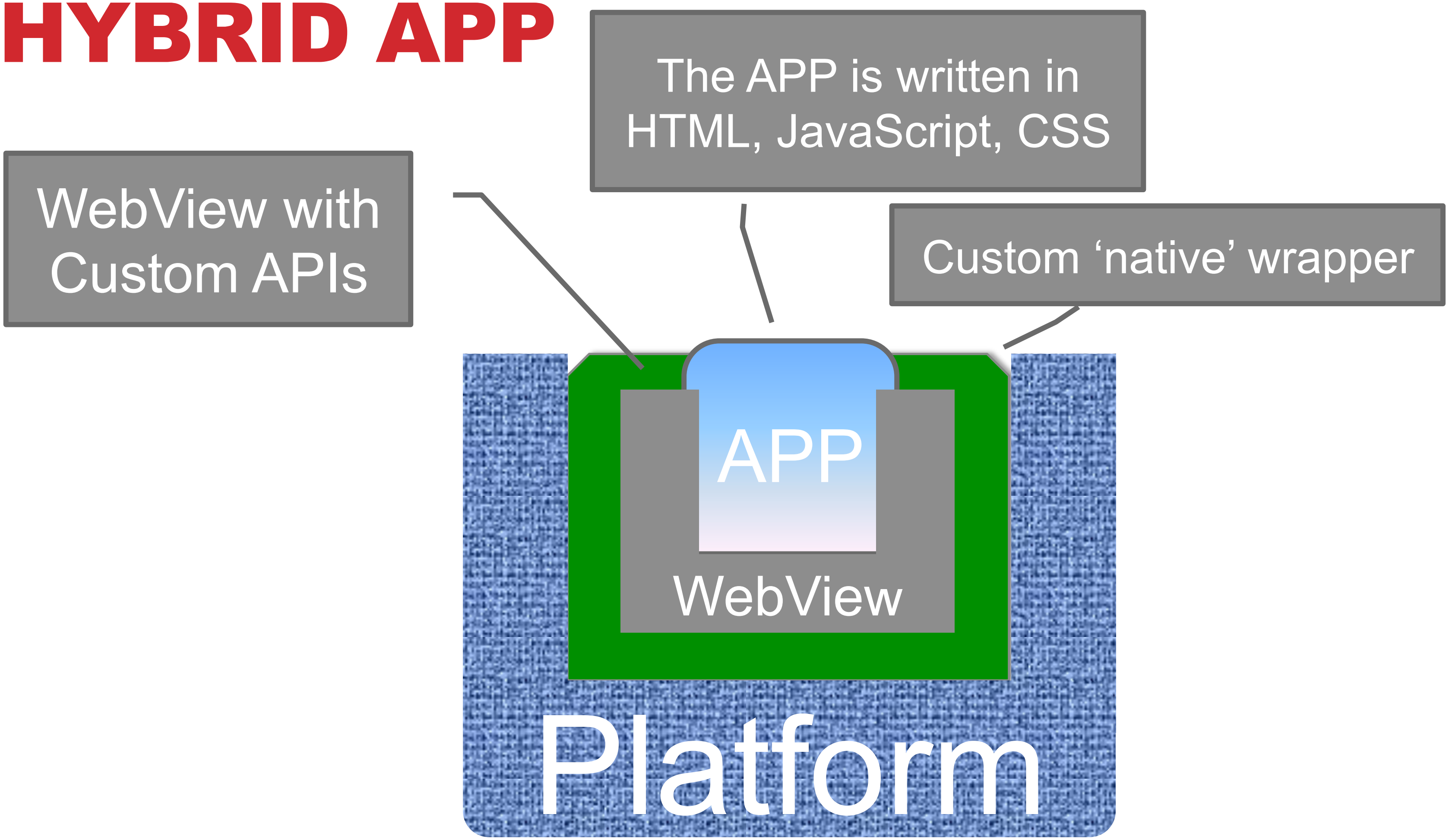
Native App

WEB APP


APIs generally cross-browser
The APP is written in HTML, JavaScript, CSS
APP
Web Browser
Platform

Web App

HYBRID APP



WebView with Custom APIs

The APP is written in HTML, JavaScript, CSS

Custom ‘native’ wrapper

Hybrid App

REMEMBER

USER-CENTRIC DESIGN

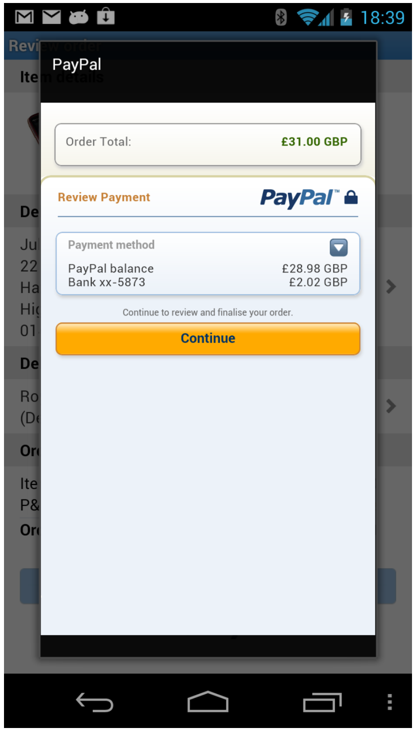
USER-CENTRIC UI & UX

Design apps that *delight your users*

With good engineering foundations

Beauty

User-journeys



Order Total: £31.00 GBP

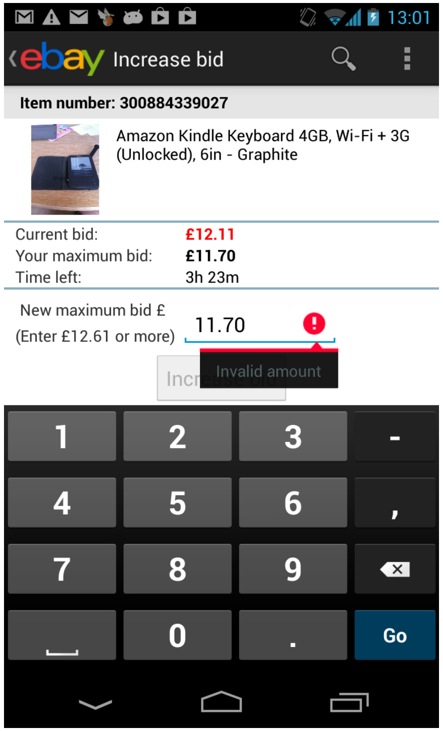
Review Payment

Payment method

PayPal balance £28.98 GBP
Bank xx-5873 £2.02 GBP

Continue to review and finalise your order.

Cannot continue…



Item number: 300884339027

Amazon Kindle Keyboard 4GB, Wi-Fi + 3G (Unlocked), 6in - Graphite

Current bid: £12.11
Your maximum bid: £11.70
Time left: 3h 23m

New maximum bid £ 11.70
(Enter £12.61 or more)

Invalid amount

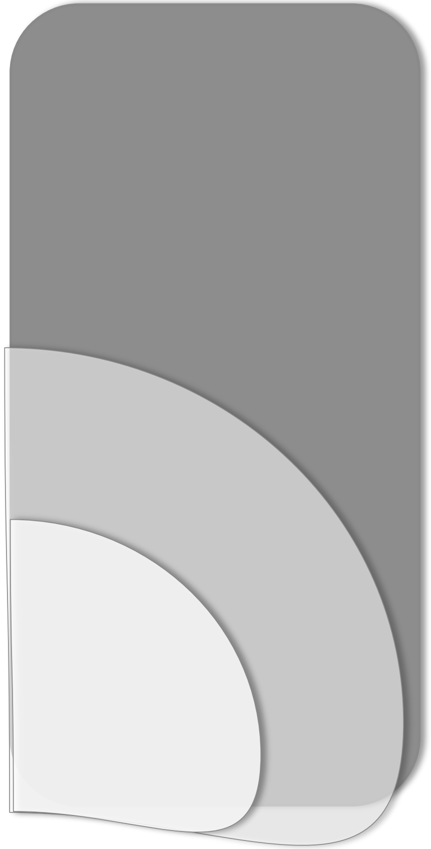
Confusing 'error' message

https://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html

USER-CENTRIC UI & UX

Consider context-sensitive uses

- ***On-the-move: walking, cycling, driving***
- ***Varying connectivity***

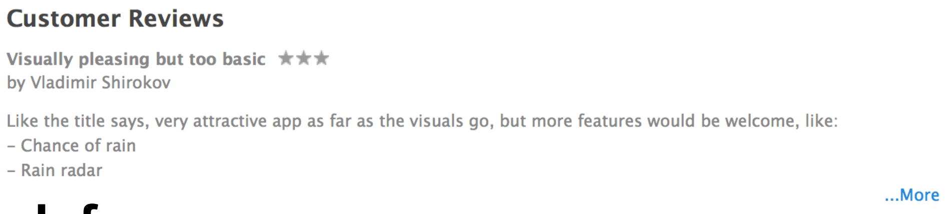


Single-Thumb
Reach (left hand)

DATA DRIVEN

USER-CENTRIC FEEDBACK

USER-CENTRIC FEEDBACK



Customer Reviews

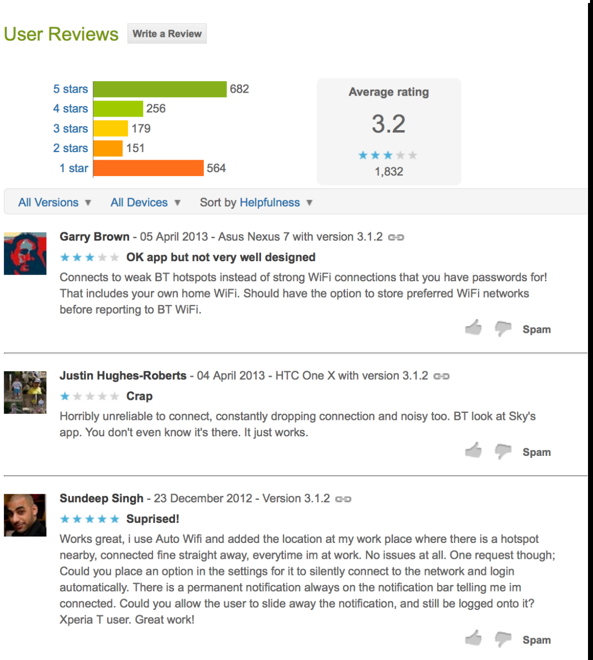
Visually pleasing but too basic ★★★
by Vladimir Shirokov

Like the title says, very attractive app as far as the visuals go, but more features would be welcome, like:
– Chance of rain
– Rain radar

...More

Our primary conduits for feedback from users are:

- Comments in App Stores
- Social media



User Reviews

5 stars 682
4 stars 256
3 stars 179
2 stars 151
1 star 564

Average rating
3.2
★★★☆☆
1,832

Garry Brown - 05 April 2013 - Asus Nexus 7 with version 3.1.2
★★★☆☆ **OK app but not very well designed**
Connects to weak BT hotspots instead of strong WiFi connections that you have passwords for! That includes your own home WiFi. Should have the option to store preferred WiFi networks before reporting to BT WiFi.

Justin Hughes-Roberts - 04 April 2013 - HTC One X with version 3.1.2
★☆☆☆☆ **Crap**
Horribly unreliable to connect, constantly dropping connection and noisy too. BT look at Sky's app. You don't even know it's there. It just works.

Sundeep Singh - 23 December 2012 - Version 3.1.2
★★★★★ **Suprised!**
Works great, i use Auto Wifi and added the location at my work place where there is a hotspot nearby, connected fine straight away, everytime im at work. No issues at all. One request though; Could you place an option in the settings for it to silently connect to the network and login automatically. There is a permanent notification always on the notification bar telling me im connected. Could you allow the user to slide away the notification, and still be logged onto it? Xperia T user. Great work!

Bad news hangs around…

Plan to respond effectively to user-feedback

- Nip problems ‘in-the-bud’
- Solicit feedback in a positive setting

DATA DRIVEN

QUALITY-IN-USE FEEDBACK

QUALITY-IN-USE

What qualities are relevant?

- **Visible qualities**
  - Performance
  - Usability
  - Reliability
- **Invisible qualities**
  - Security
  - Privacy
  - Robustness
  - etc

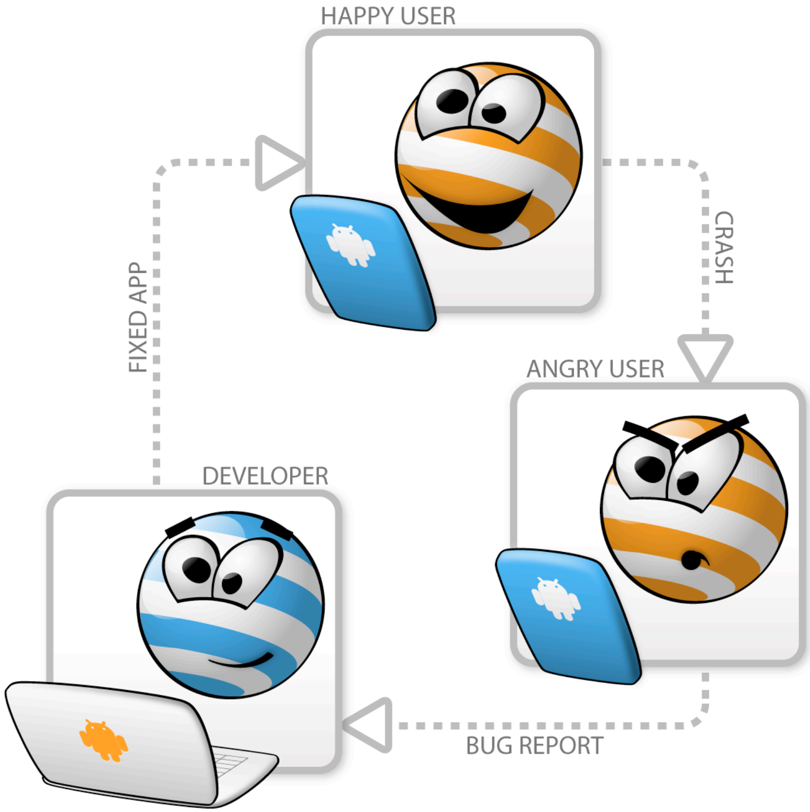
http://bits.blogs.nytimes.com/2012/06/27/facebook-plans-to-speedup-its-iphone-app/

QUALITY-IN-USE FEEDBACK

Design-in:

- **Crash reporting**
- **In-app feedback?**
- **Mobile Analytics**

Consider the privacy implications



http://android-developers.blogspot.co.uk/2010/05/google-feedback-for-android.html

CRASH REPORTING SERVICES

Options include:

- Platform / App Store: integrated
- Commercial offerings: from third-parties
- Homebrew: most control & flexibility

THE MOBILE LANDSCAPE

Reliability of the platforms

- iOS vs. Android vs. other platforms?

Ways to diagnose reported crashes

- Look for correlation and causation
- Clustering error reports e.g. By:
  - Platform version
  - Device Model
  - Versions of our app
  - Other software installed?

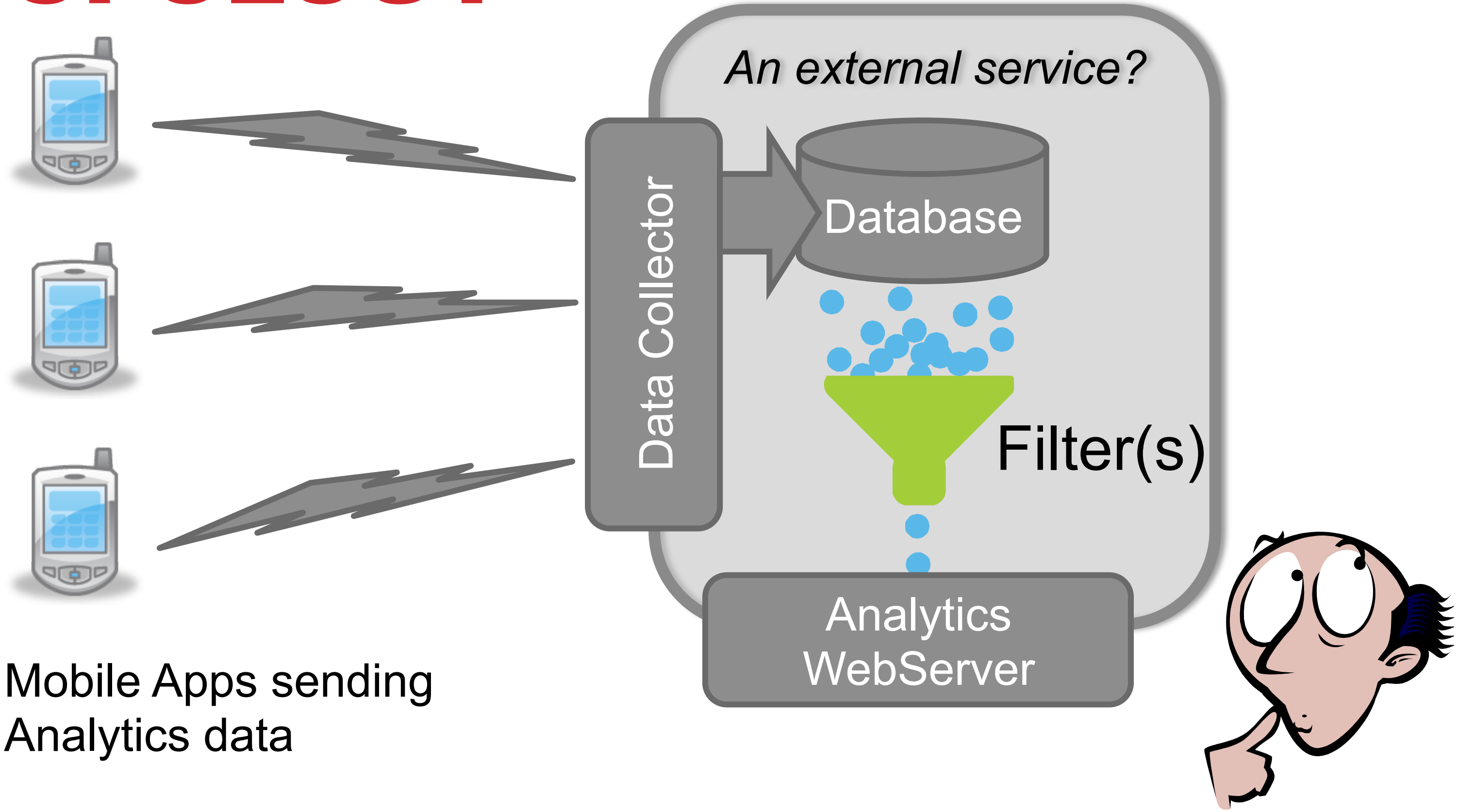
http://www.pcworld.com/article/249356/whos_the_smartphone_os_crash_champion.html

http://www.crittercism.com/

DATA DRIVEN

USE MOBILE ANALYTICS

TOPOLOGY

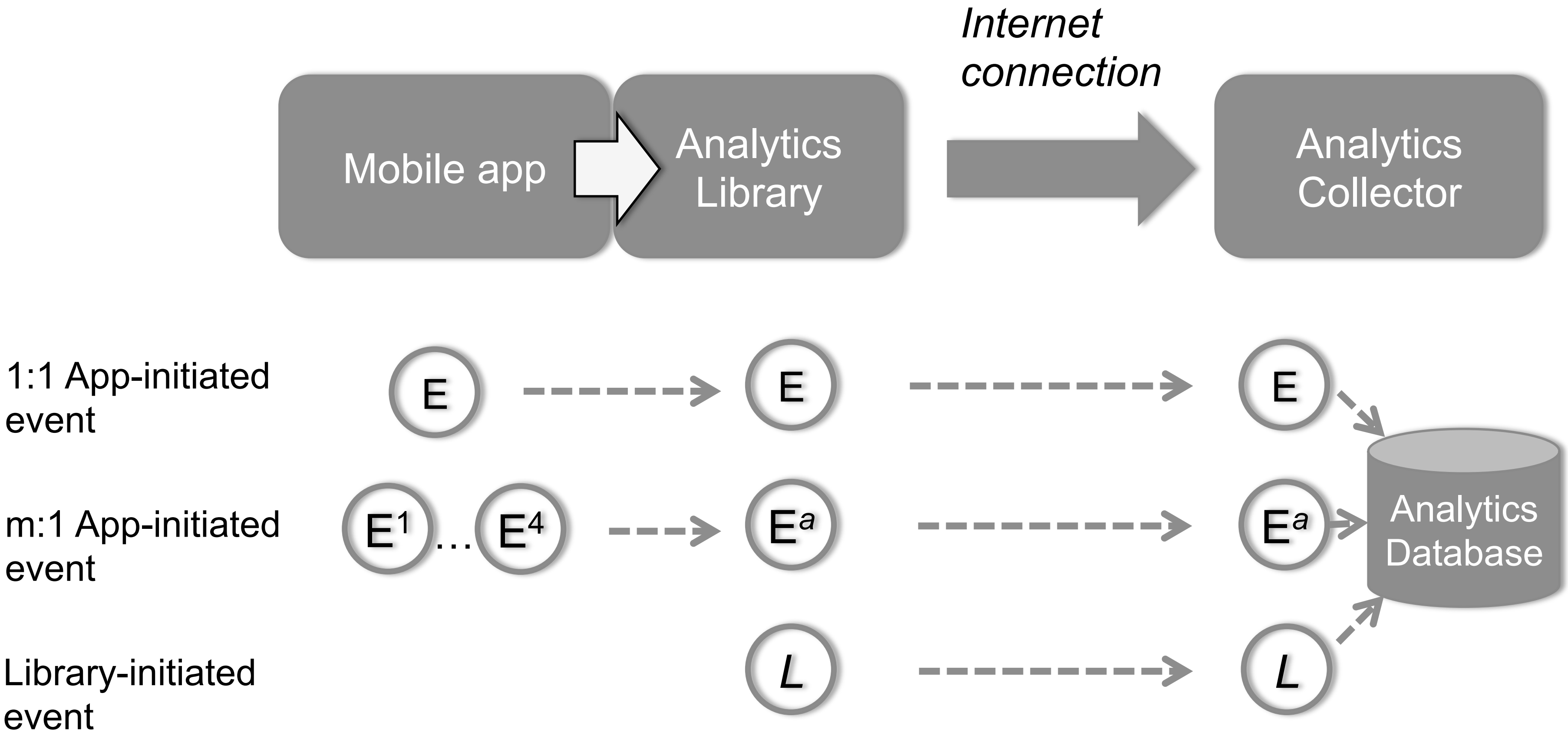


Mobile Apps sending
Analytics data

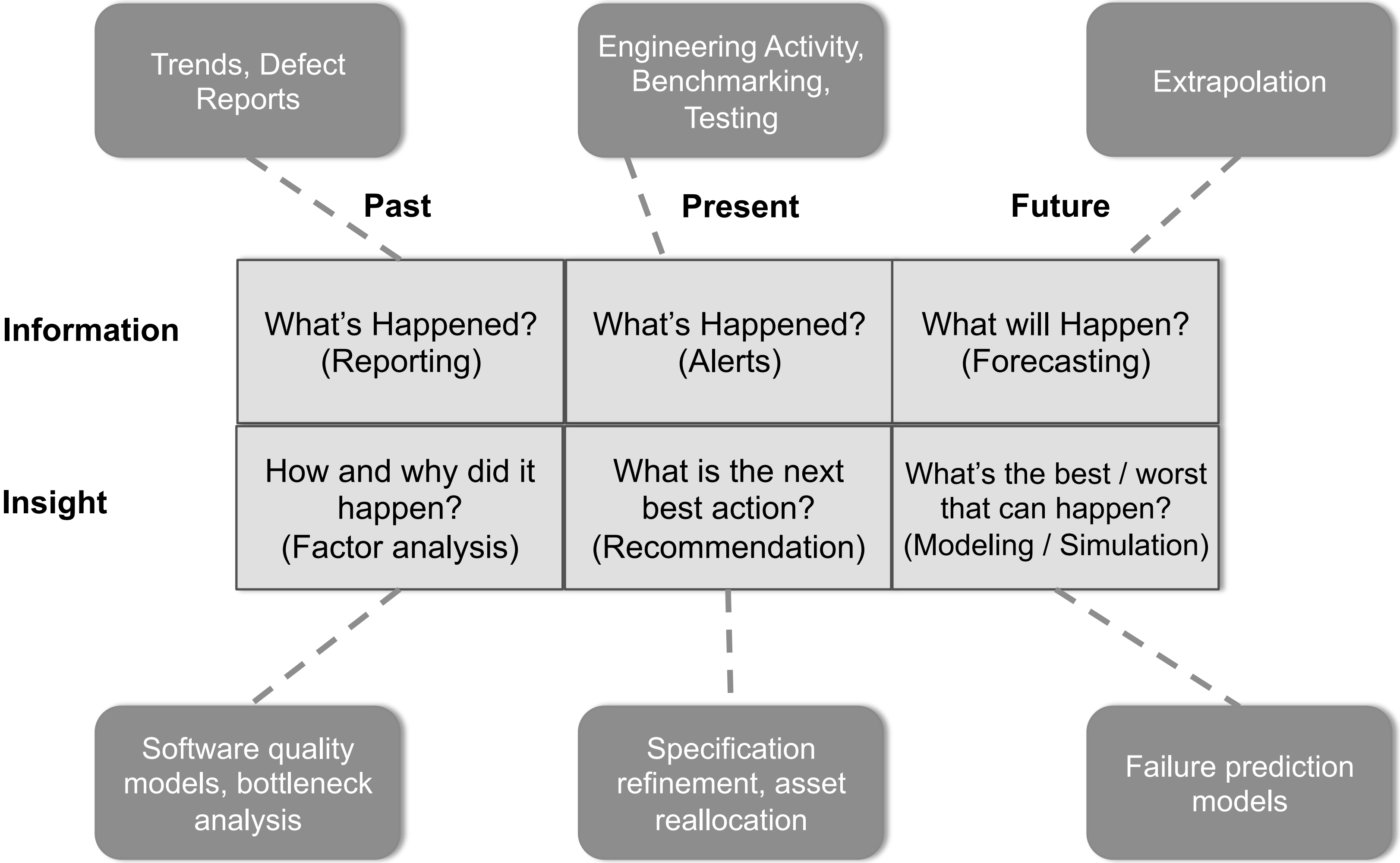
Business
view

Overview of Mobile Analytics
Each step may be delayed

TYPES OF EVENTS



ANALYTICAL QUESTIONS



	Past	Present	Future
Information	What's Happened? (Reporting)	What's Happened? (Alerts)	What will Happen? (Forecasting)
Insight	How and why did it happen? (Factor analysis)	What is the next best action? (Recommendation)	What's the best / worst that can happen? (Modeling / Simulation)

Trends, Defect Reports

Engineering Activity, Benchmarking, Testing

Extrapolation

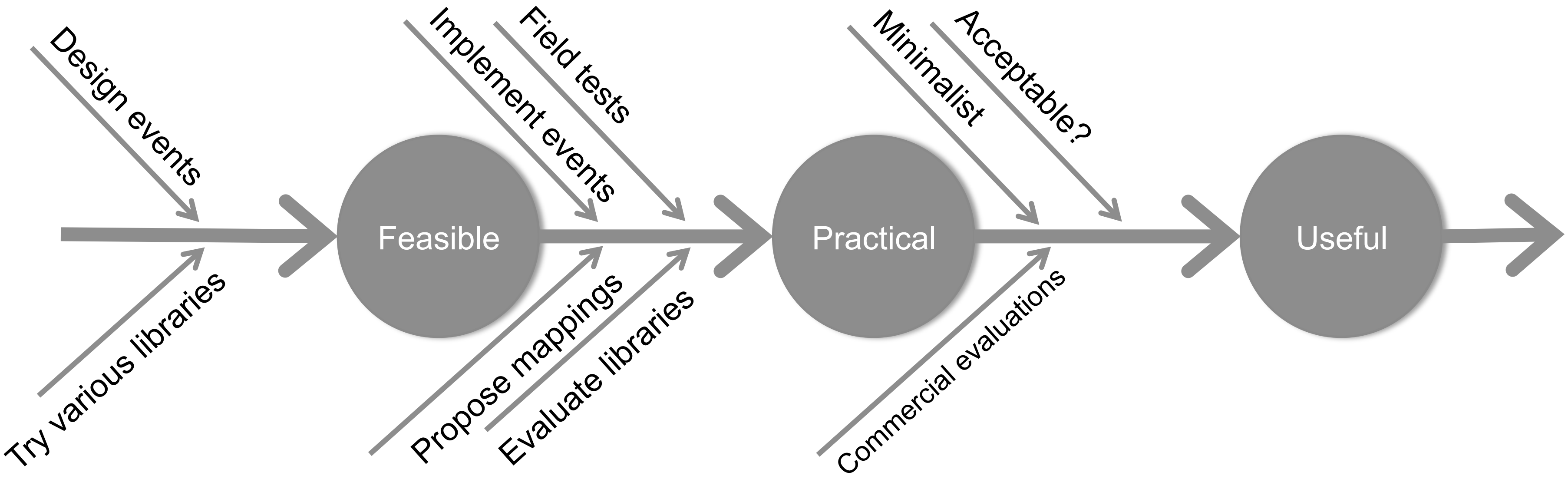
Software quality models, bottleneck analysis

Specification refinement, asset reallocation

Failure prediction models

http://research.microsoft.com/pubs/136974/foser-2010-buse.pdf

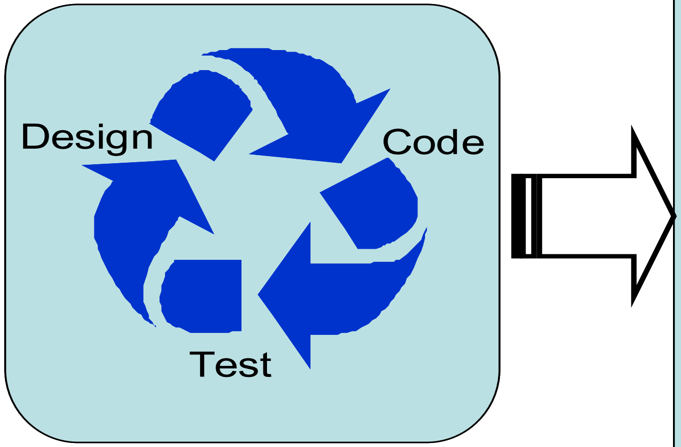
FISHBONES


Design events
Try various libraries
Feasible
Implement events
Field tests
Propose mappings
Evaluate libraries
Practical
Minimalist
Acceptable?
Commercial evaluations
Useful

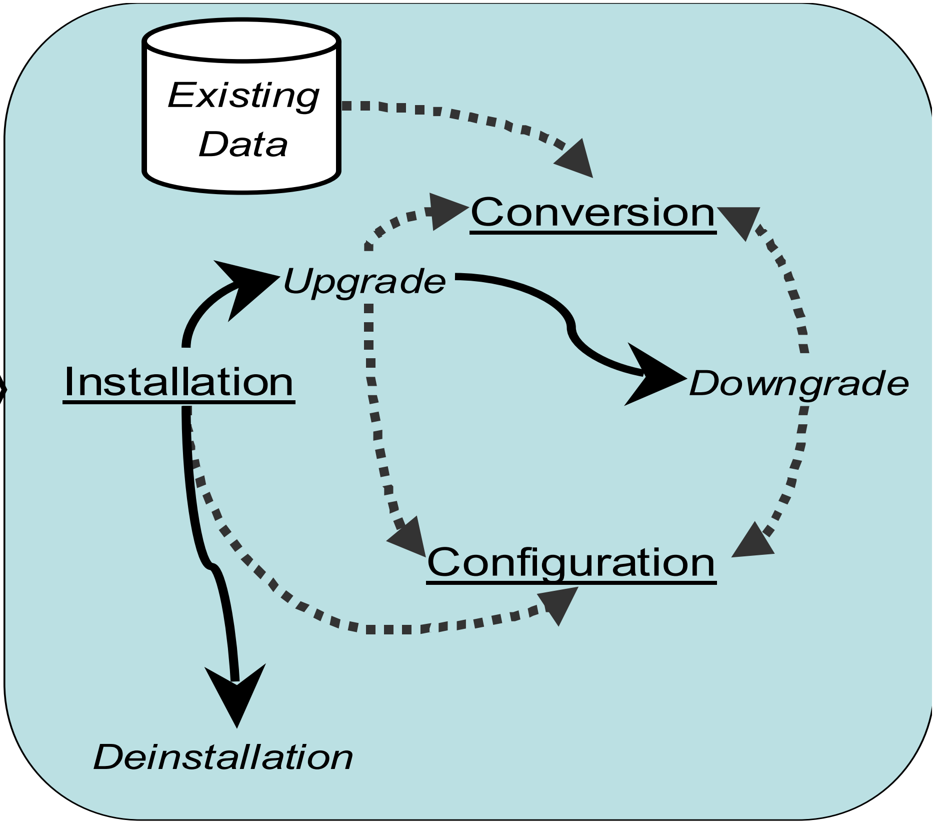
DEVELOPING AN APP

SOFTWARE DEVELOPMENT LIFECYCLES

LIFECYCLES OF TRADITIONAL SOFTWARE

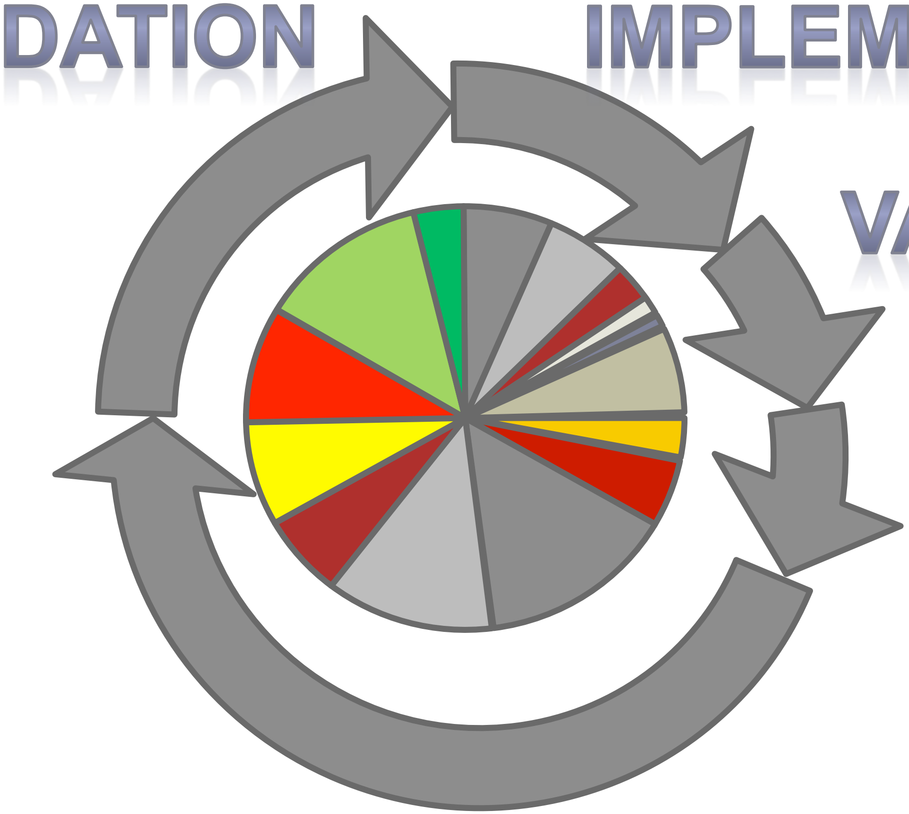


Software Development Life Cycle



Software Usage Life Cycle

FROM *CREATION* TO *USE* PIE CHART



VALIDATION

IMPLEMENTATION

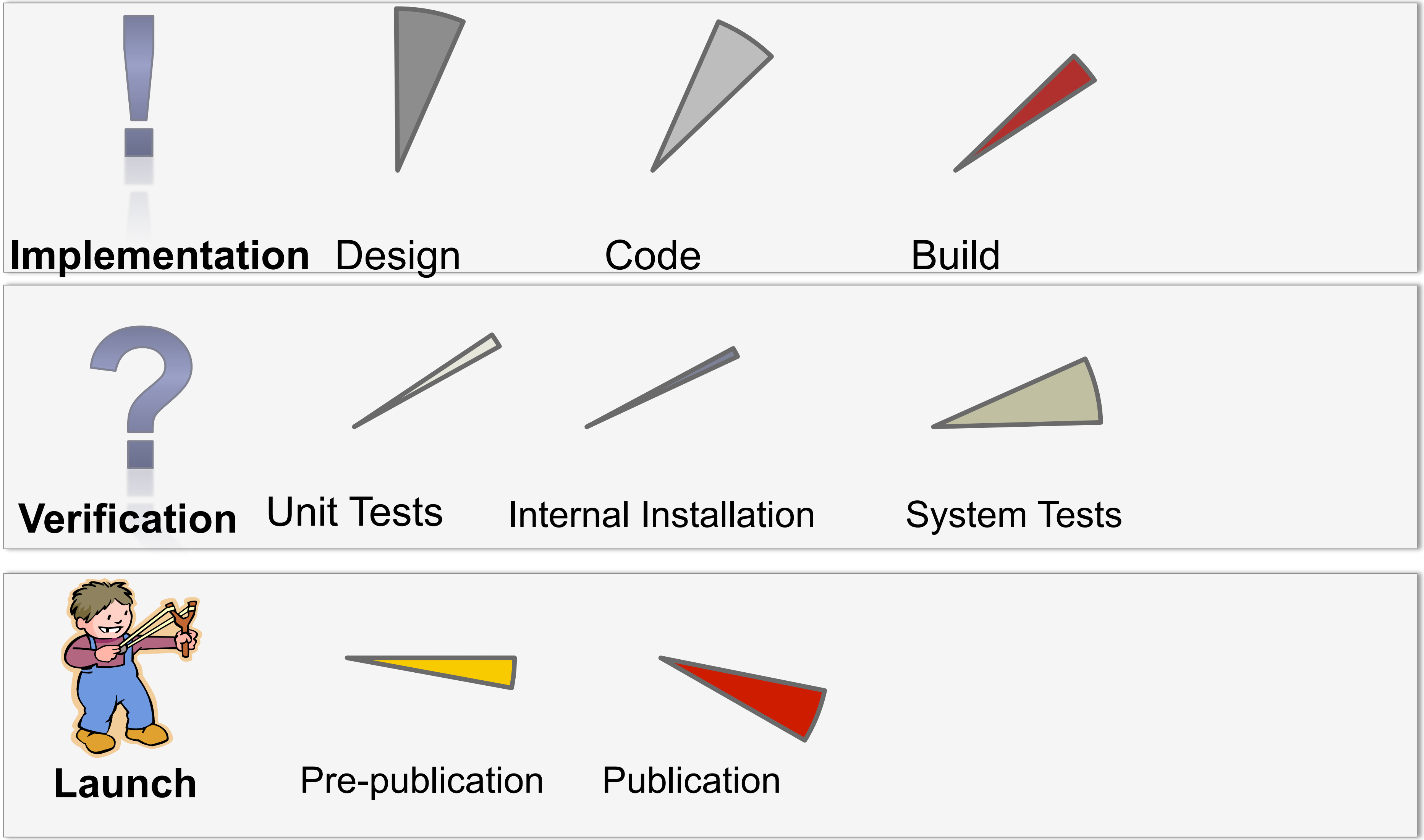
VALIDATION

LAUNCH

ENGAGEMENT

Note: The dimensions are indicative, rather than realistic

MOBILE DEVELOPMENT FROM *CREATION* TO *USE*$^{(I)}$

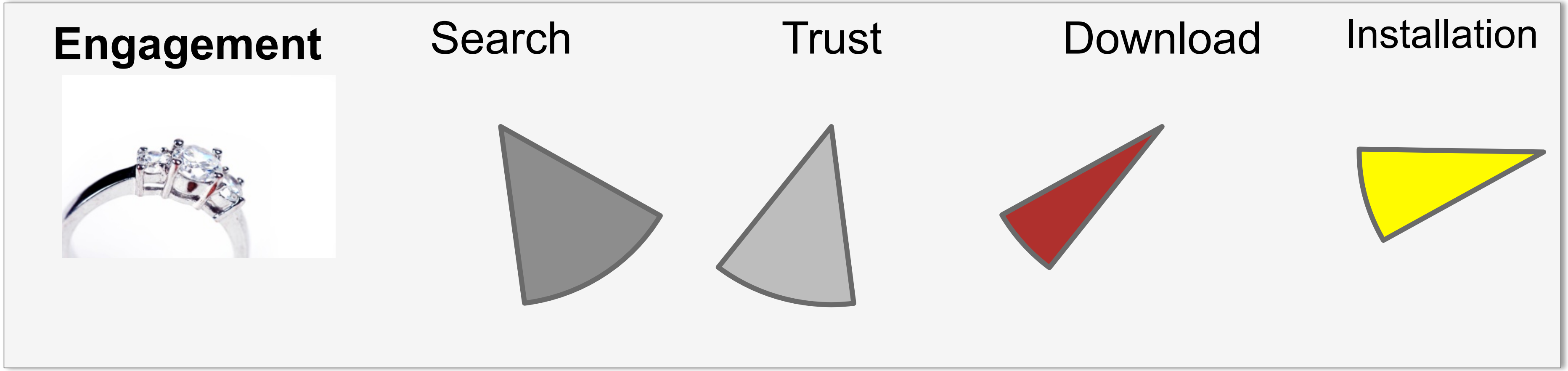


Implementation Design Code Build

Verification Unit Tests Internal Installation System Tests

Launch Pre-publication Publication

MOBILE DEVELOPMENT FROM *CREATION* TO *USE*(II)



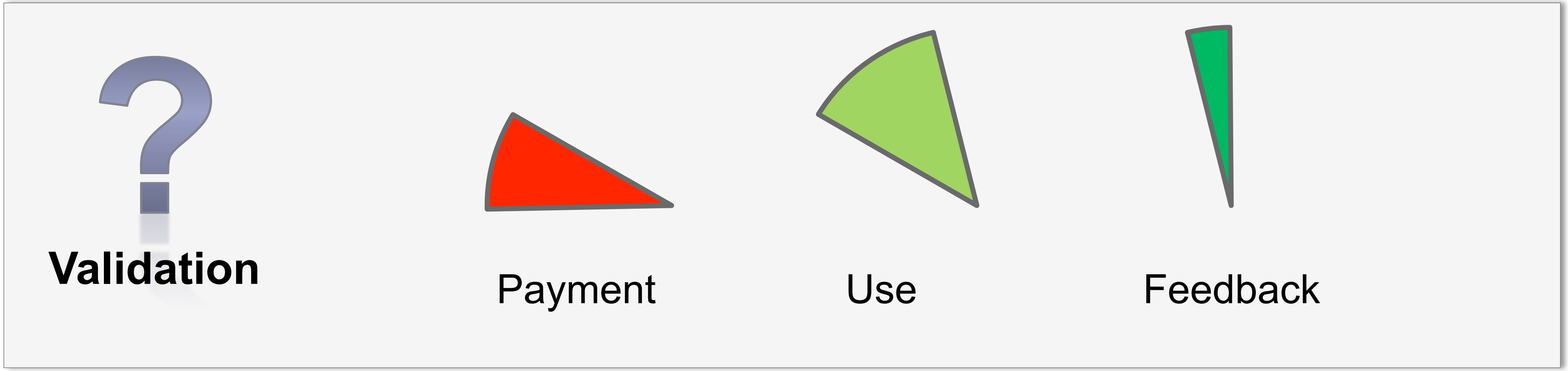
Engagement

Search

Trust

Download

Installation



Validation

Payment

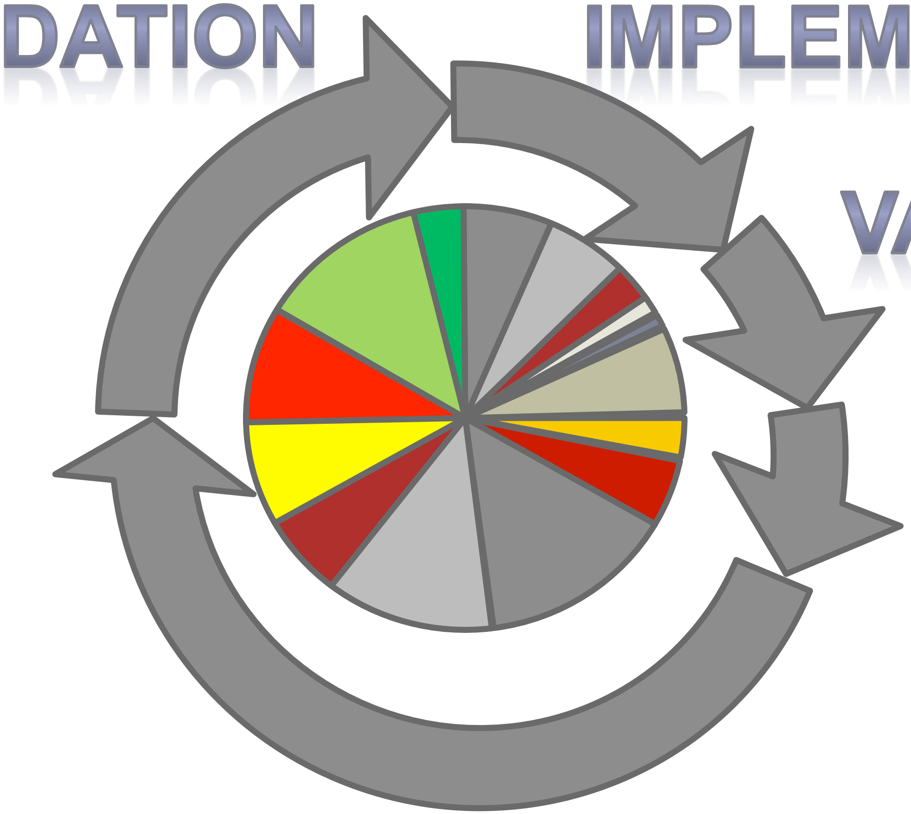
Use

Feedback

FROM *CREATION* TO *USE* PIE CHART

VALIDATION

IMPLEMENTATION



VALIDATION

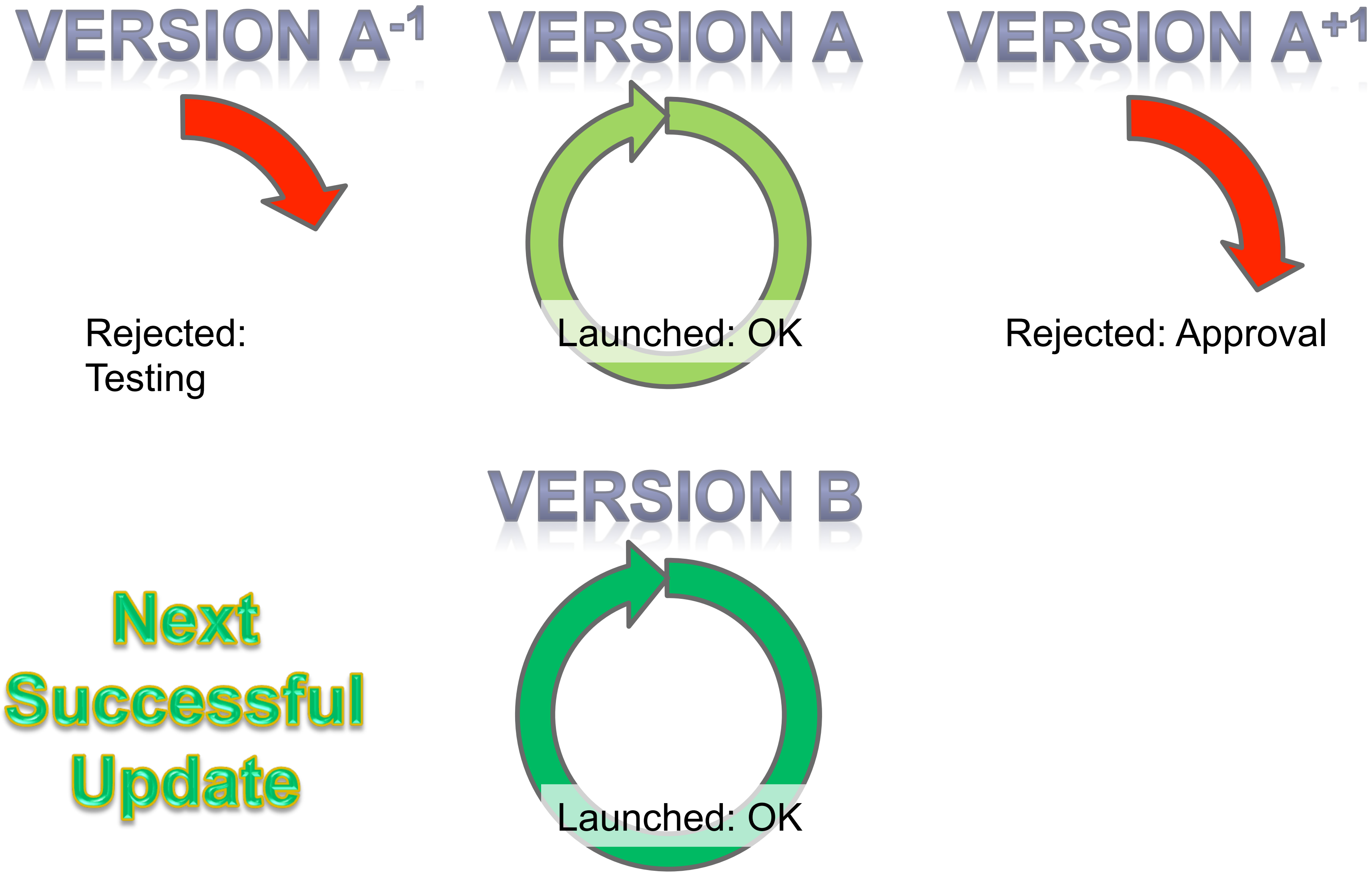
LAUNCH

5 phases

ENGAGEMENT

Note: The dimensions are indicative, rather than realistic

ITERATIONS & UPDATES



VERSION A^{-1}

Rejected: Testing

VERSION A

Launched: OK

VERSION A^{+1}

Rejected: Approval

VERSION B

Launched: OK

Next Successful Update

WHO DOES WHAT?

TEAM COMPOSITION

TEAM COMPOSITION

- **In-house**
- **Outsourced**
  - Development
  - Design
  - Testing
- **Hybrid arrangements**
- **Where are the people?**
  - Co-located
  - In-phase (same timezone)
  - Out-of-phase (amount of overlap may vary)

SCALING YOUR APP

PLANS FOR GROWTH

PLANS FOR GROWTH

Design considerations

- **Globalization**
  - Internationalization
  - Localization

- **Integration with third-parties**
  - Payment processing
  - Social networking
  - *These may be specific to a country or region*

m-pesa

Orkut, Nimbuzz[1]

[1] http://www.forbes.com/sites/parmyolson/2013/04/03/facebook-phone-faces-an-uphill-battle-in-emerging-markets/

DESIGNING OUR APPS

CLEAN CODE?

TANGLED SOURCE CODE?

PI
UAP
U A
P U A
I U
U A
UA
I A
A I
AI U
U
UAPI
PAP
P
PP
P
I
U

U – User Interface

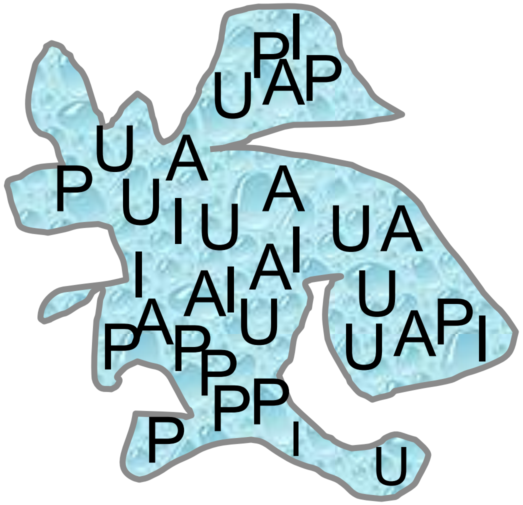
A – Application

I – Input & Output

P – Platform-Specific

*Is this how **your** application code appears to be constructed?*

DISENTANGLING IS…



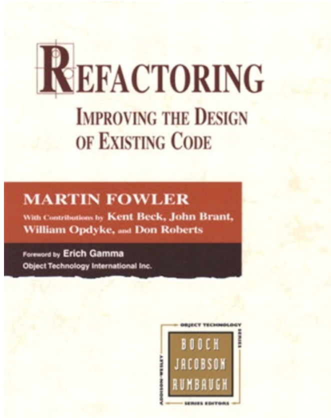
U – User Interface

A – Application

I – Input & Output

P – Platform-Specific

- Hard
- Expensive
- Error-prone
- Unlikely


REFACTORING

IMPROVING THE DESIGN OF EXISTING CODE

MARTIN FOWLER

With Contributions by Kent Beck, John Brant, William Opdyke, and Don Roberts

Foreword by Erich Gamma
Object Technology International Inc.

BOOCH
JACOBSON
RUMBAUGH

http://martinfowler.com/books.html

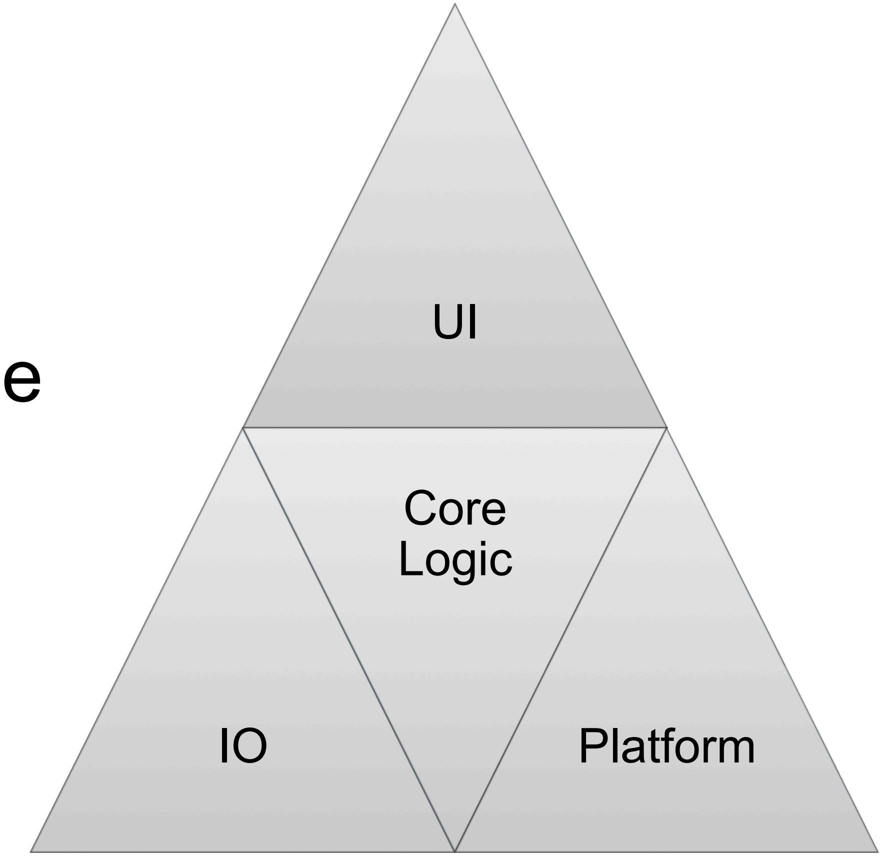
http://en.wikipedia.org/wiki/Code_refactoring

So let's aim to avoid this from the outset!

SEGMENTED DESIGN

Apps include:

- User Interface code
- Input & Output code
- Platform-specific code
- Core Logic code



Can we group and segment them?

DESIGN CONSIDERATIONS

APIS

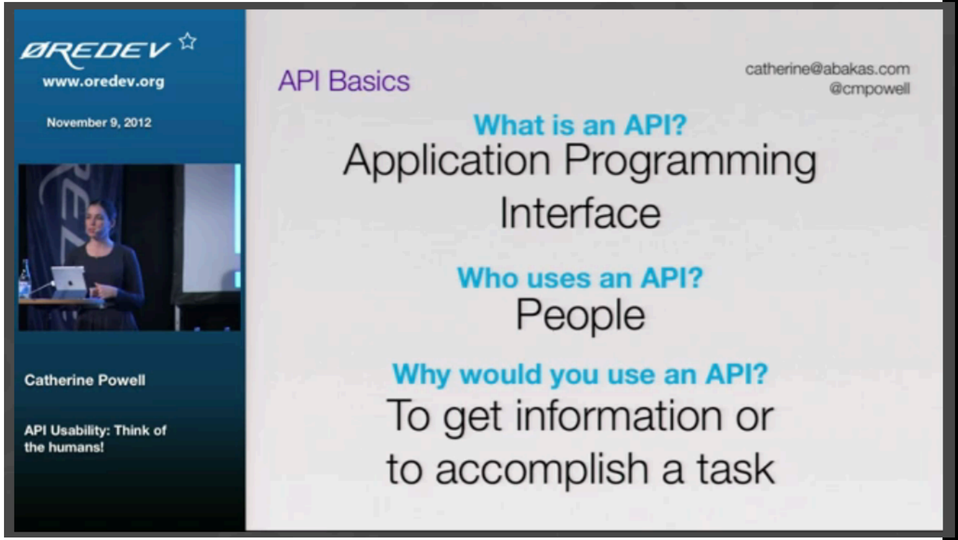
APIS

APIs – bi-directional conduits of information & services

- APIs for Servers & Services
- Client-side APIs and Interfaces

Key design considerations

- Clean APIs
- Robustness
- Scalability
- Privacy of data across APIs



ØREDEV
www.oredev.org
November 9, 2012

Catherine Powell

API Usability: Think of the humans!

API Basics

catherine@abakas.com
@cmpowell

What is an API?
Application Programming Interface

Who uses an API?
People

Why would you use an API?
To get information or to accomplish a task

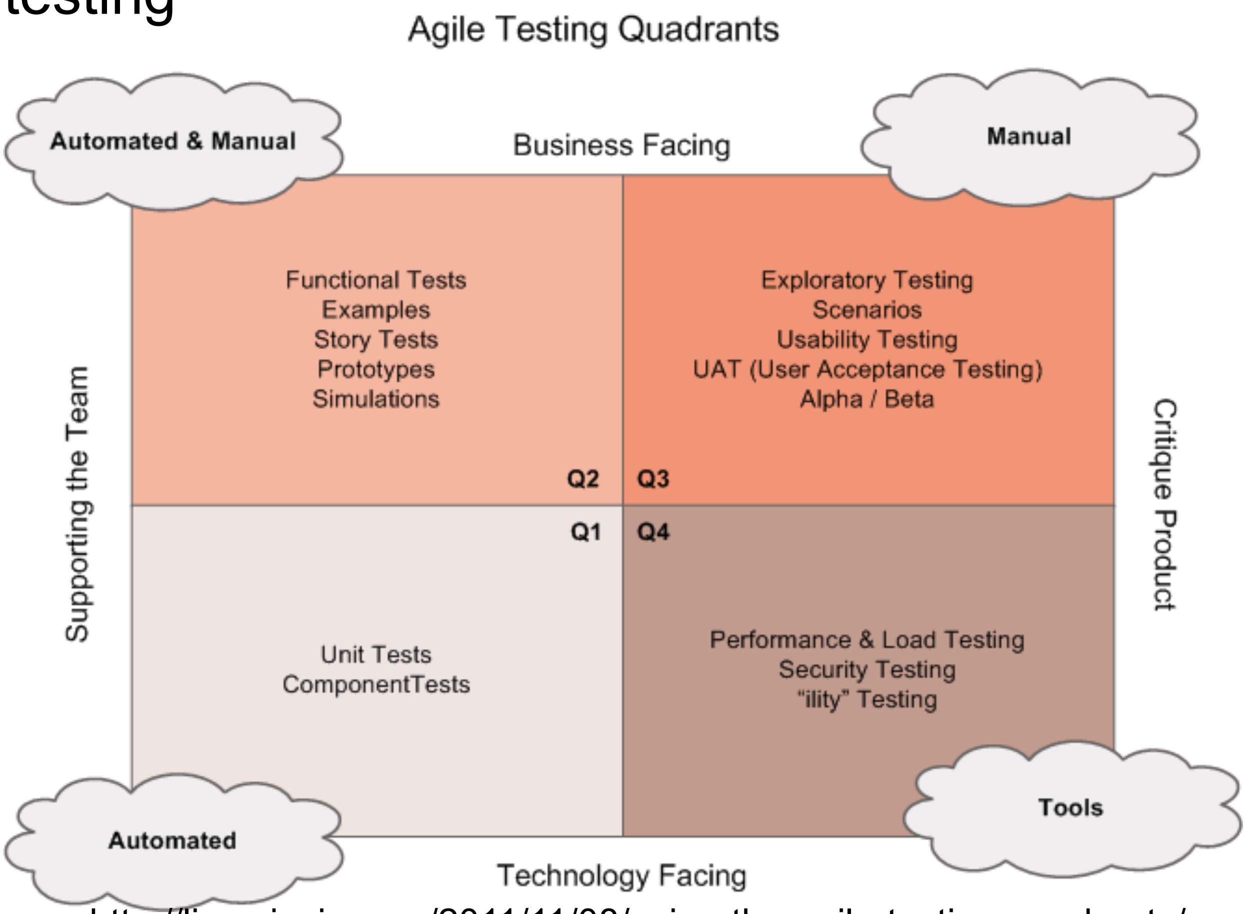
http://oredev.org/2012/sessions/api-usability-think-of-the-humans

TESTING OUR MOBILE APPS

VERIFICATION & VALIDATION

PERSPECTIVES OF VERIFICATION

- Developer- & Technology- centric automated tests
- Business-centric testing



Agile Testing Quadrants

Automated & Manual

Business Facing

Manual

Q2
Functional Tests
Examples
Story Tests
Prototypes
Simulations

Q3
Exploratory Testing
Scenarios
Usability Testing
UAT (User Acceptance Testing)
Alpha / Beta

Q1
Unit Tests
ComponentTests

Q4
Performance & Load Testing
Security Testing
"ility" Testing

Supporting the Team

Critique Product

Automated

Tools

Technology Facing

http://lisacrispin.com/2011/11/08/using-the-agile-testing-quadrants/

PERSPECTIVES OF VALIDATION

Are we providing the right product?

- Are we meeting the business goals?
- Is this app one that people are happy with?

- ***Remember the human aspects***

INTERACTIVE TESTING

Learn to do it well

- Test things that will affect the application e.g.
  - Rotation, network connectivity, locales

Learn utilities and tools e.g.

- `adb logcat`

REAL & VIRTUAL DEVICES

How do we test rich interactive apps unless you have the device in your hand?

Testing on devices is essential

- Capture the rich interactions and behaviours
- Find bugs related to devices bugs, performance, usability
- More realistic representation of reality

Testing on virtual devices

- Early prototyping e.g. can test new screen sizes quickly
- Sometimes necessary for business, practical & other reasons

“In theory, theory and practice are the same.
In practice, they are not.”

A device in your hand is worth 2 in the cloud, and 100 virtual devices.

EXTENDING YOUR REACH

Remote Devices

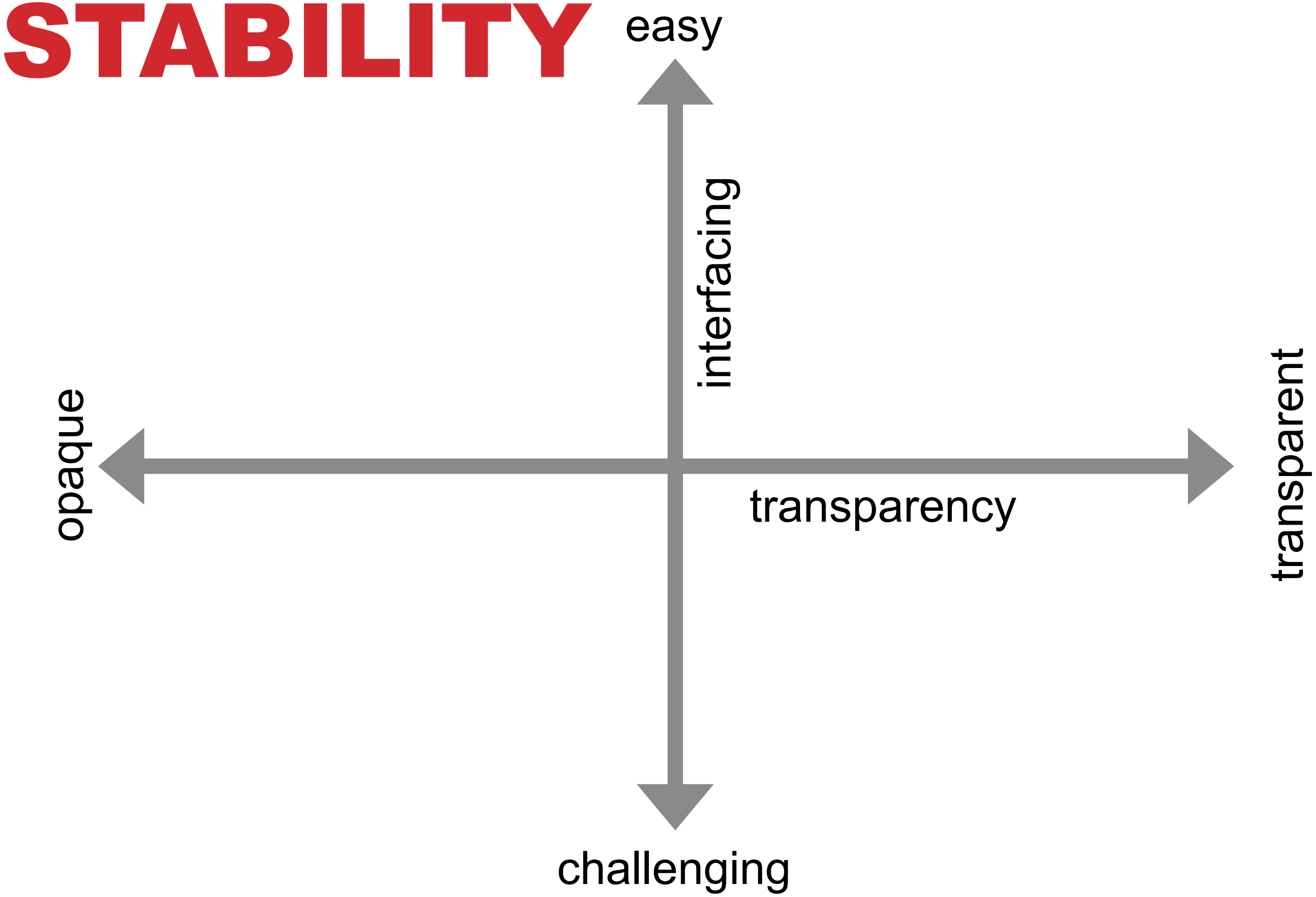
Remote Test Execution

Lunch out: testing a mobile video app

GOOD DESIGN

TESTABILITY & MAINTAINABILITY

SCALES OF TESTABILITY



There are at least 2 dimensions of Testability:

- **ease of interfacing**
- **transparency into the state & behaviour of the software being tested.**

MAINTAINABILITY

- **Corrective:** fix problems & bugs
- **Adaptive:** work better with other software and devices
- **Perfective:** improving the app
- **Preventative:** address potential problems pre-emptively

http://www.csse.monash.edu.au/~jonmc/CSE2305/Topics/13.25.SWEng4/html/text.html

DESIGNING FOR TESTABILITY: HOOKS

Programmatic Hooks

To connect test automation easily

Consider whether to leave them in situ

DESIGNING FOR TESTABILITY: VISIBILITY

"Eyes into the Soul of the machine..."

Expose internal data and state

- Makes some checks easier to confirm
- e.g. Error recovery mechanisms cleaned up the app's internal state

Beware:

- Non-test code might start using the data
- If so, consider formalising the access in an API

GOOD DESIGN: LAYERING OF CODE

(Already covered some aspects in the Segmented Design topic)

Ideal to be able to automate the testing of each layer or component independently

Then testing of the composite software can focus on testing the composite aspects

Beware of emergent behaviour

- Test the qualities: non-functional-testing (NFT)

GOOD DESIGN: SEPARATION OF CONCERNS

Separate generic and platform-specific code

Generic code:

- Application logic: What the app does, functionality

Platform-specific code:

- User Interface
- Threading
- Calls to platform-specific APIs

GOOD DESIGN: ISOLATE COMPLEX CODE

Try encapsulating & isolating complex code

- Provide an interface*
- Have excellent automated tests exercise it
- Warn casual developers (and testers) not to tamper with it
- Now the rest of our code is easier to understand & manage

In parallel consider ways to replace complex code with simpler code

* e.g. See the Facade design pattern

SPENDING WISELY?

FULL LIFECYCLE COSTS

WHERE AND WHEN TO

SPEND MONEY ON TESTING?

NOVODA

Costs 60% more money to 'add' test automation to Android projects

- Who's willing to sign off on it?

Where and when does the ROI start?

- After 2nd Release: 1/3rd elapsed testing effort

GOOD DESIGN: THINGS TO CONSIDER

How long do your code bases ‘last’?

Who pays for ‘maintenance’?

Where is the expertise to maintain the code?

Active apps need ongoing nurture & investments *even if you’re not changing the functionality*

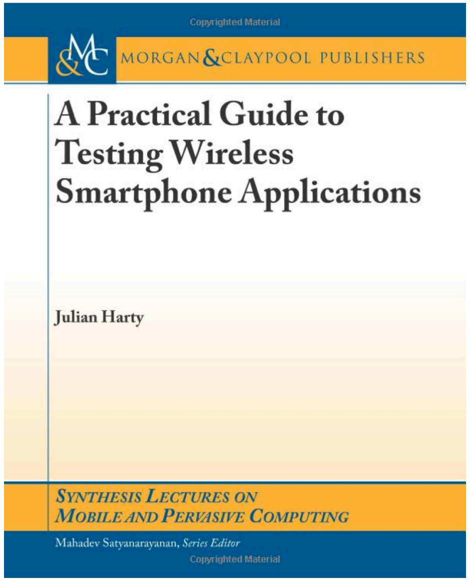
FURTHER CONSIDERATIONS

Intellectual Property (IP)

- **Copyright**
- **Licenses**
  - Open Source and Free Software
- **IP assignments**
- **Copy & Paste development practices?**

SOME FURTHER READINGS

MOBILE DEVELOPER'S GUIDE TO THE PARALLEL UNIVERSE
WARNING 3RD EDITION Marketing Stuff
The Companion Guide of the MOBILE DEVELOPER'S GUIDE TO THE GALAXY


Copyrighted Material

MORGAN & CLAYPOOL PUBLISHERS

A Practical Guide to Testing Wireless Smartphone Applications

Julian Harty

Synthesis Lectures on Mobile and Pervasive Computing

Mahadev Satyanarayanan, *Series Editor*

Copyrighted Material

Testing and Test Automation for Mobile Apps
Julian Harty

Summer 2013
CRC Press

Development **Marketing** **Testing$^{(2009)}$** **Testing$^{(2013)}$**

http://www.wipconnector.com/download/GuideToTheParallelUniverse_3rdEdition.pdf

http://www.enough.de/fileadmin/uploads/dev_guide_pdfs/Guide_12thEdition_WEB.pdf

IF YOU WANT THE DRAFT BOOK

Draft book available for review & feedback

Email me **julianharty@gmail.com**

- Agree not to share or distribute
- Comment to get the next available draft

Q & A ?