The Role of Conceptual Knowledge in API Usability

Andrew J. Ko  
Yann Riche
one of the first steps in engineering a great idea is deciding whether it’s feasible

assess feasibility, code, code, code, code, code, code,
for example...

image an Android app that said “Hi!” to every nearby Android phone...

“Hi!”

is it possible?
is it possible?

what would you do to decide?
what would you need to know to succeed?

search for similar apps online?
find example code?
read API documentation?
WHAT WE KNOW

when developers *write* code, they

they search for **code examples** on the web

**reformulate queries** to find the right keywords

seek **architectural details**
WHAT WE DON’T KNOW

do developers do the same things when assessing feasibility?

what else do developers need to know?

is API documentation with classes, methods, fields, and architecture detail sufficient?
WE RECRUITED...

2 pilots + 7 developers

median self-reported fluency of 10 languages

most common were C, C++, C#, Java, JavaScript, PHP
WE ASKED THEM TO...

assess 2 requirements

RQ1 The application must be able to obtain a list of discoverable Bluetooth-enabled devices in its proximity.

RQ2 The application must be able to determine whether the wifi network the phone is connected to is using a WPA or WEP secure connection.

for 2 mobile platforms for 20 minutes each
WE ASKED PARTICIPANTS...

to think aloud

to assess each requirement/platform pair independently, in a random order

to provide a **verbal judgments of the feasibility** of requirement during tasks

we also recorded screen capture videos of their work
WE ANALYZED THEIR WORDS...

we identified declarative judgments in participants’ utterances (2,633 utterances total)

performed open coding on judgments ➔ 5 categories (82% agreement)

relevance judgments (about info)
usability judgments (about site)
audience judgments (about site)
proximity judgments (about task)
metacognitive judgments (about self)
RESULTS

some participants struggled to even find the API reference online

\[
\text{discoverability} \propto \text{API usability}
\]

some treated the design of a site as a predictor of the API’s usability

\[
\text{visual clarity} \propto \text{API usability}
\]

[Dorn & Guzdial 2010, Myers et al. 2010]
RESULTS

based on relevance judgments, developers sought similar information as when writing code

searching for code examples on the web

reformulating queries to find the right keywords

seeking architectural details


Brandt et al. 2009, Myers et al. 2010, Robillard & DeLine 2010

Myers et al. 2010, Nykaza et al. 2002, Robillard & DeLine 2010
RESULTS

based on metacognitive judgments, developers also needed knowledge about the standards and concepts used in an API

“encryption,” “pairing,” “discoverable”, etc.

they needed this knowledge to

(1) know what queries to write
(2) make relevance judgments
(3) understand examples
NEEDED CONCEPTS FOR...

knowing what queries to write

“I’d like a general overview of how Bluetooth connections work... Maybe an index into the page that would help to give me more abstract terms for how to think about it.”

Andrew J. Ko — 14
NEEDED CONCEPTS FOR...

knowing what queries to write

“I’d like a general overview of how Bluetooth connections work... Maybe an index into the page that would help to give me more abstract terms for how to think about it.”
NEEDED CONCEPTS FOR...

making relevance judgments

“Since I don't have this kind of knowledge, it's kind of hard for me... I'll spend a lot of time on the wrong page, keep searching wrong information, or maybe the right page but I didn't get it.”
NEEDED CONCEPTS FOR...

making relevance judgments

“Since I don't have this kind of knowledge, it's kind of hard for me... I'll spend a lot of time on the wrong page, keep searching wrong information, or maybe the right page but I didn't get it.”
NEEDED CONCEPTS FOR...

understanding examples and documentation text

“‘put radio into discoverable mode’. What is radio? I'm not quite sure here what radio means.”
NEEDED CONCEPTS FOR...

understanding examples and documentation text

“ ‘put radio into discoverable mode‘. What is radio? I'm not quite sure here what radio means.”
DESIGN IMPLICATIONS

API resources need more than just classes, functions, and data structures. Resources need to teach developers about the concepts used in an API.

- internet
- wireless network
- wifi
- 802.11, WPA, WEP
- WifiConfiguration

familiar

unfamiliar

teach conceptual knowledge to bridge general concepts and API concepts
DESIGN IMPLICATIONS

this teaching will help developers translate requirements...

say “hi” every to every nearby Android phone

into API-specific terminology

poll Bluetooth discovery module in background thread and extract new discoverable devices relative to previous poll, filtering by device_id, then invoke “Hi” sample for each
QUESTIONS?

developers need conceptual knowledge that links general concepts to API concepts

API documentation should teach general concepts and link them to API concepts