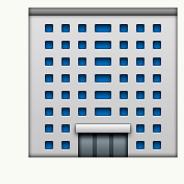


Tasting tests at Cookpad

@Kazu_cocoa



Kazuaki Matsuo(@Kazu_cocoa)

 : Cookpad Inc.

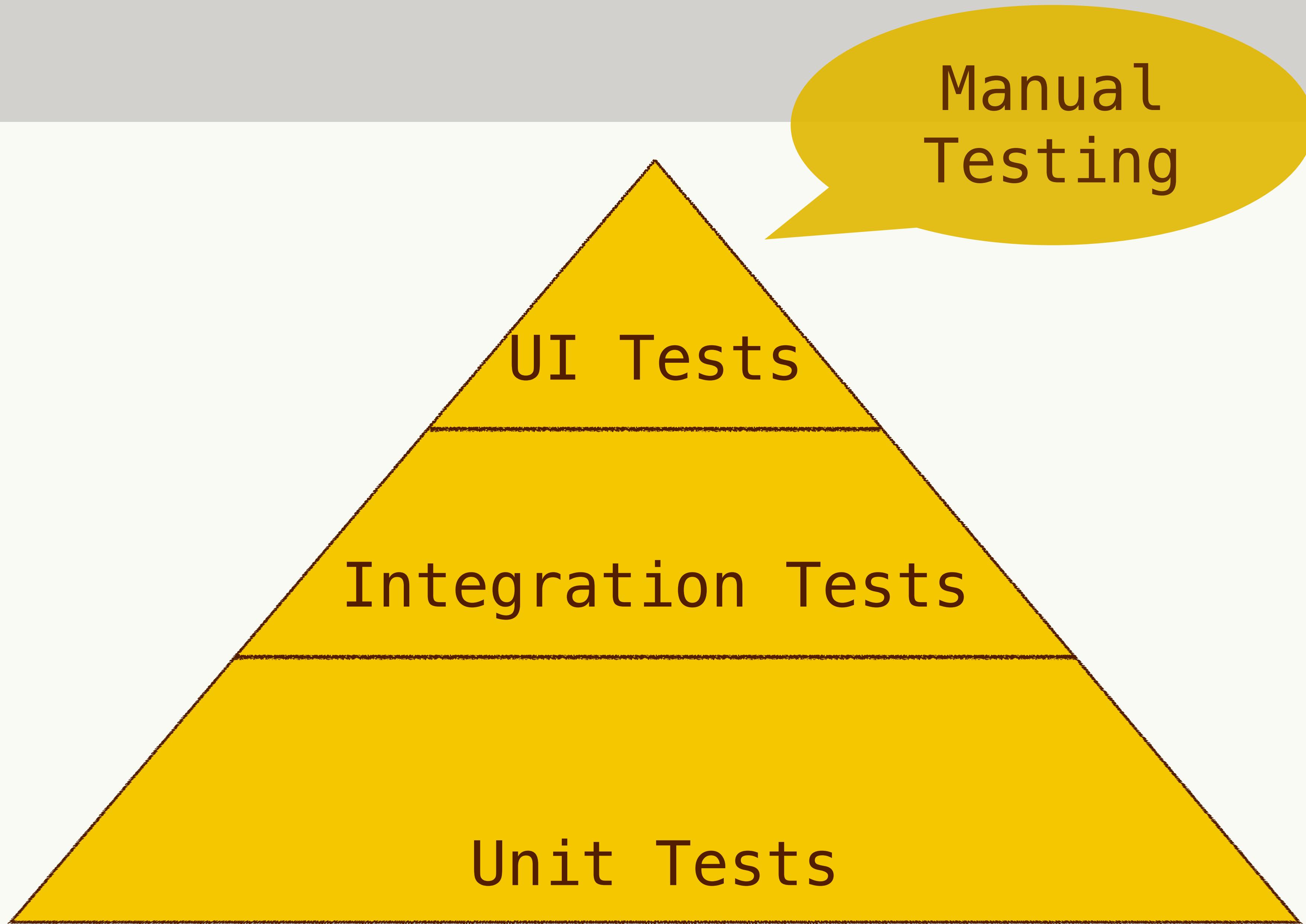
Test Engineer / Software Engineer in Quality

Lang: Swift/Ruby/Java(for Android)/Elixir(Erlang)

maintainer: Ruby Appium binding 



A bunch of themes in tests



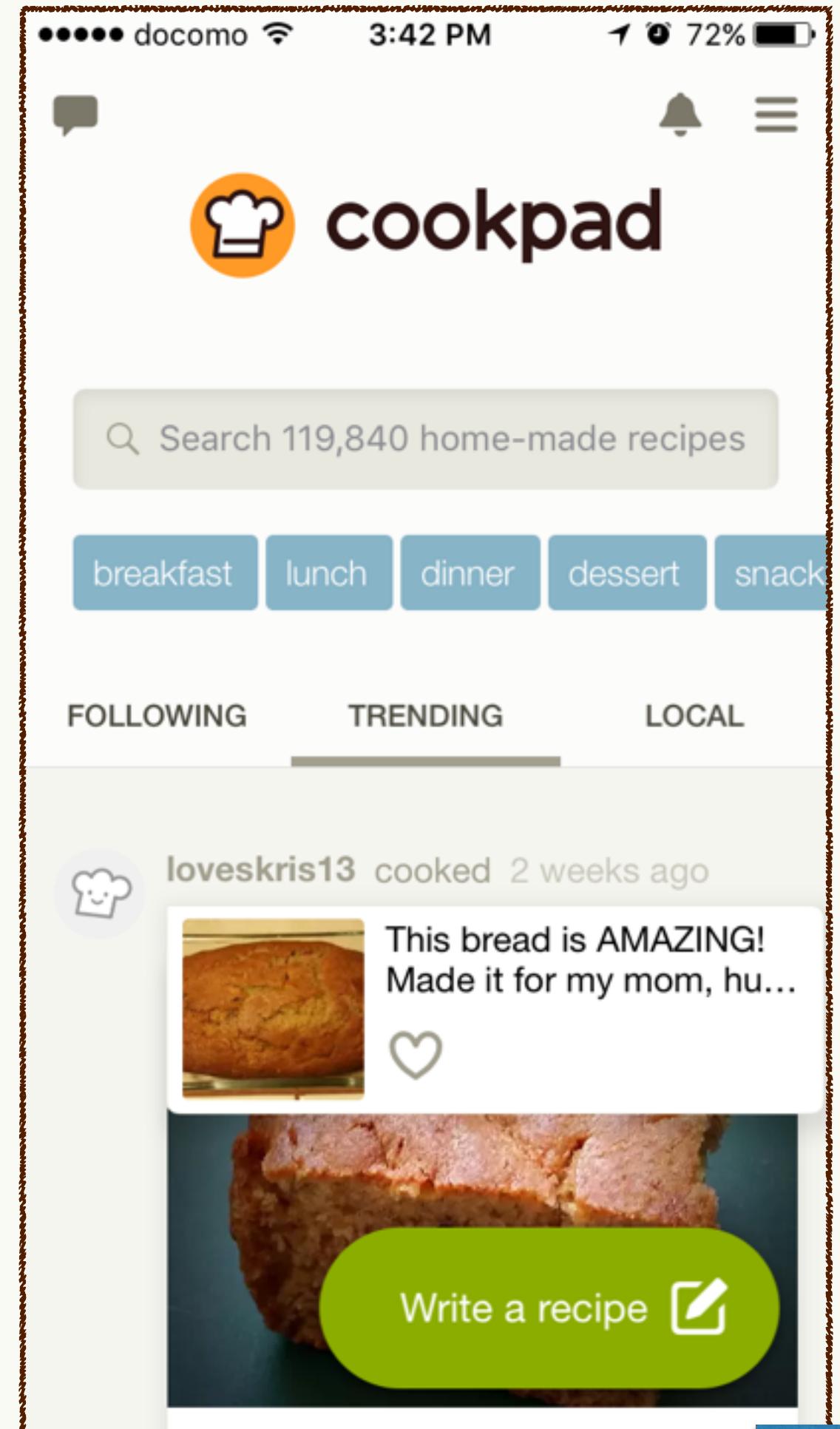
How UI Tests support our development

We should know about the test target
if we try to tests stuff

what is Cookpad?

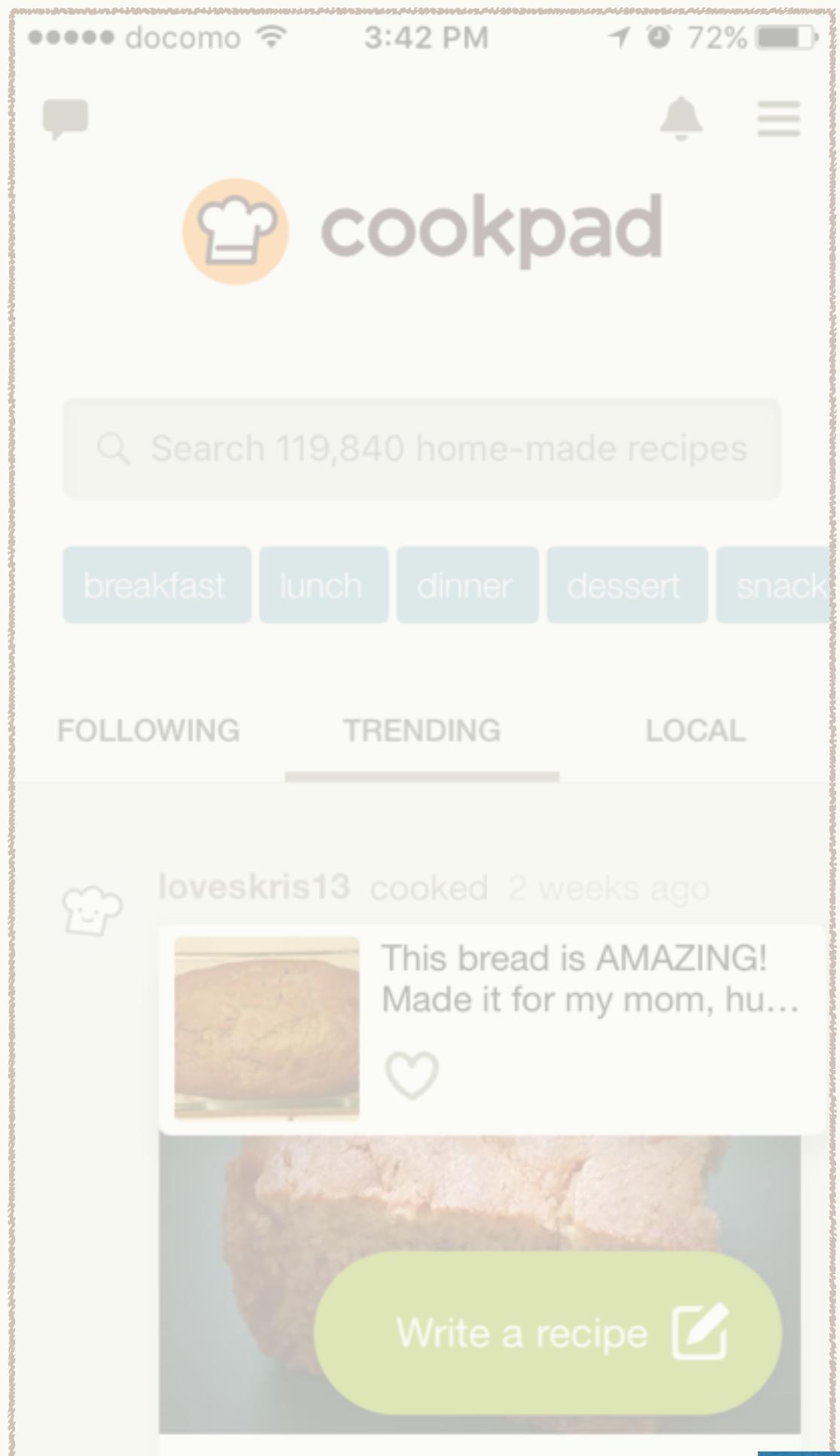


Cookpad for iOS(Japan and Global)



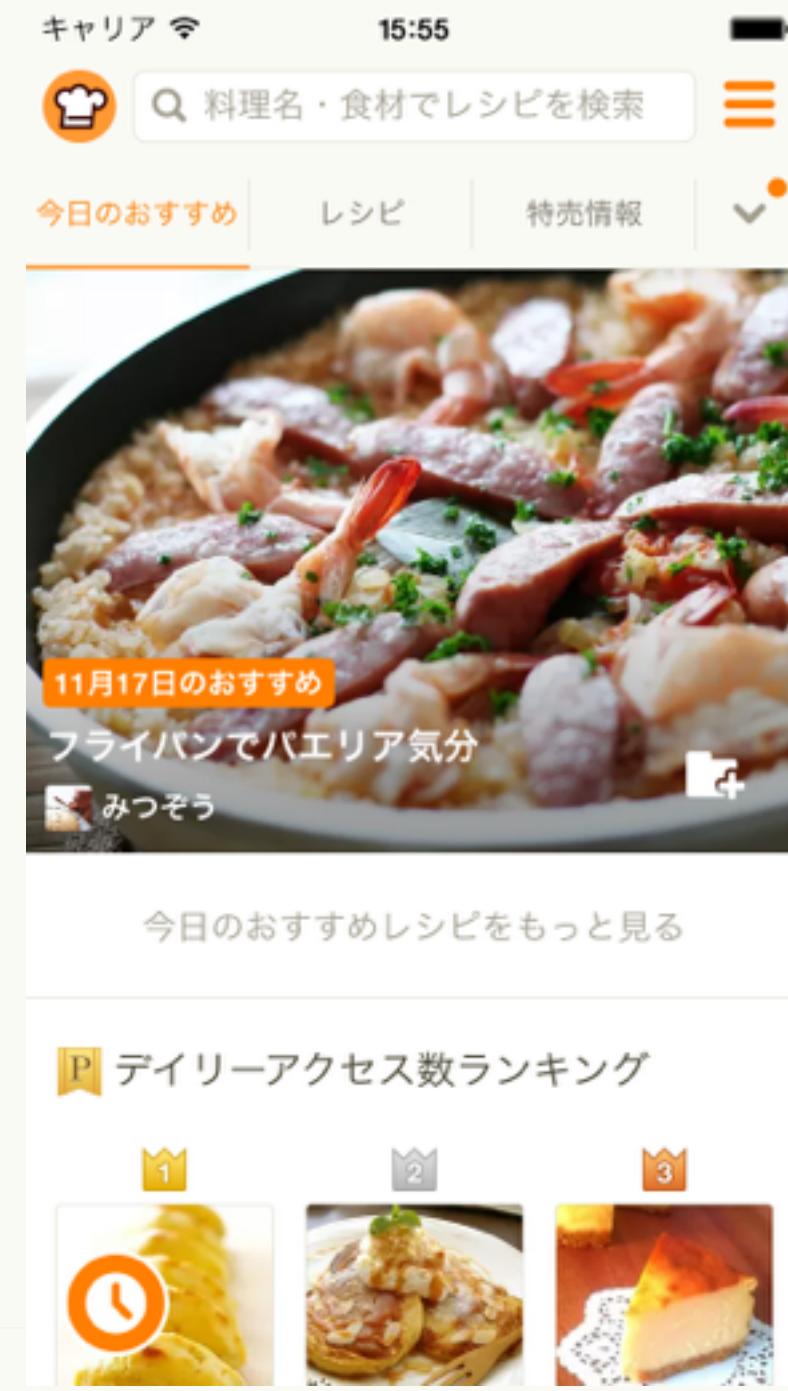
iPhoneの側をかぶせる

Cookpad for iOS (Japan)



iPhoneの側をかぶせる

A History for Cookpad iOS App



2012年1月

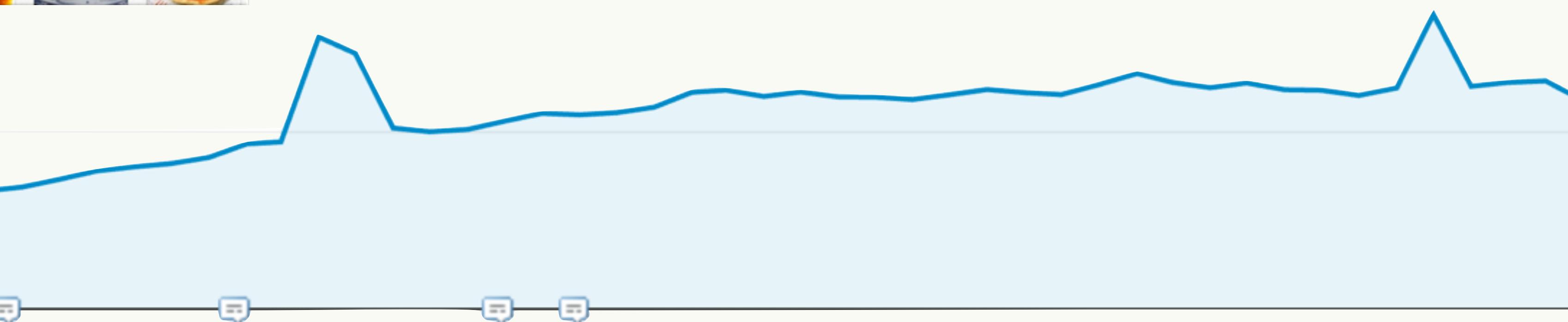
2013年1月

2014年1月

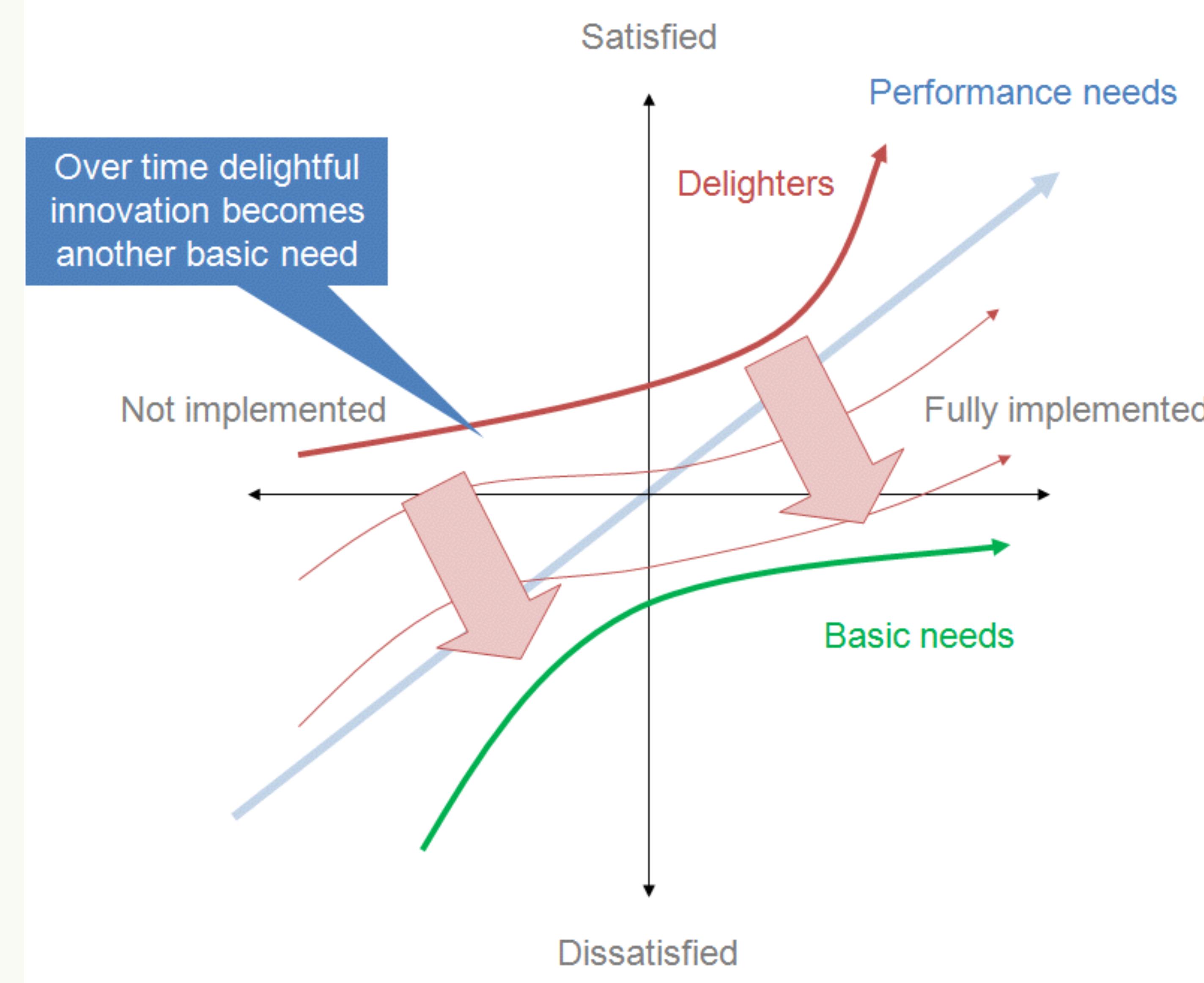
2015年1月

2016年1月

2017年

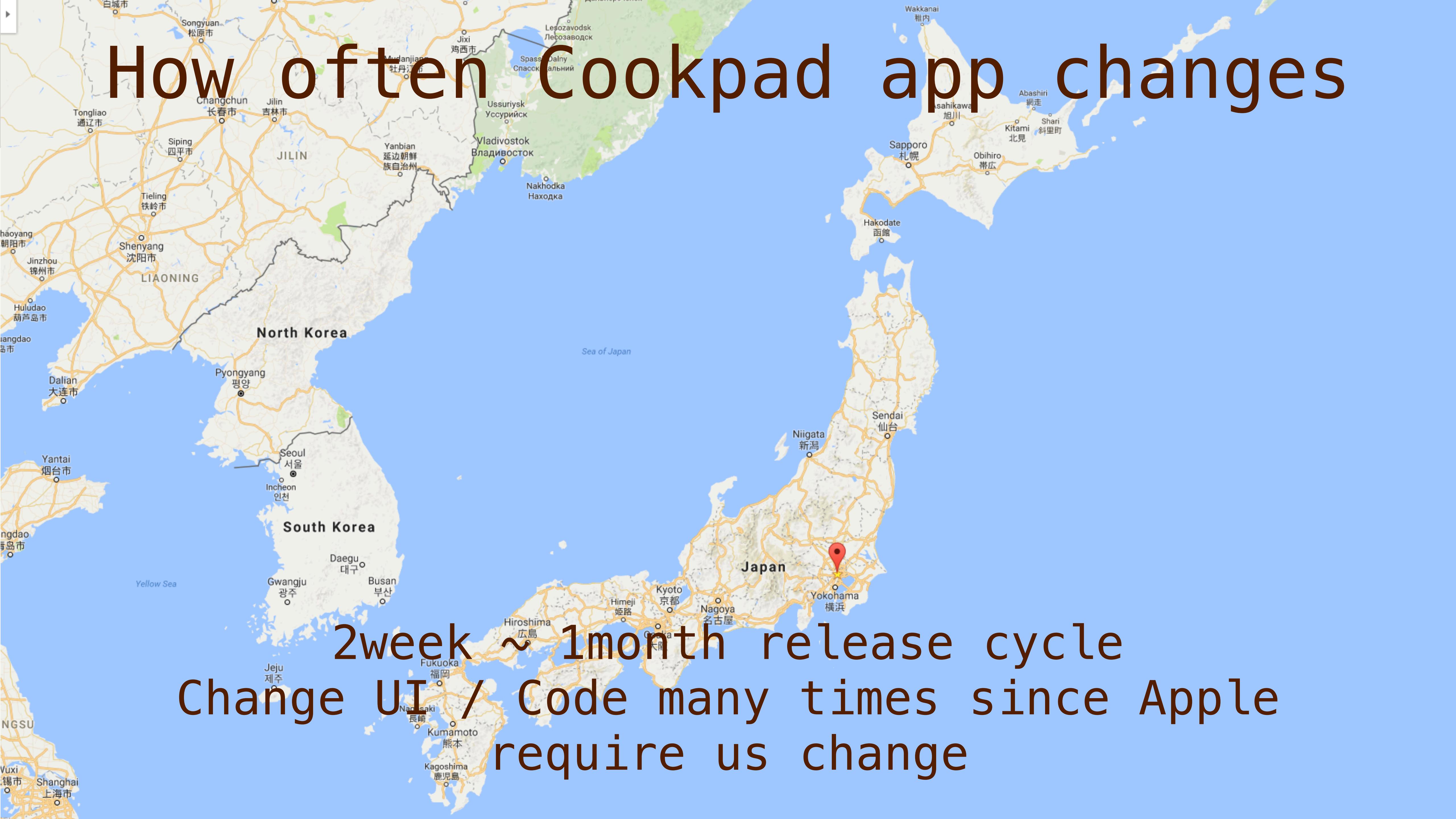


kano-model and Japan market



Diachronic Quality for Mobile

How often Cookpad app changes



2 week ~ 1 month release cycle
Change UI / Code many times since Apple
require us change

take a break



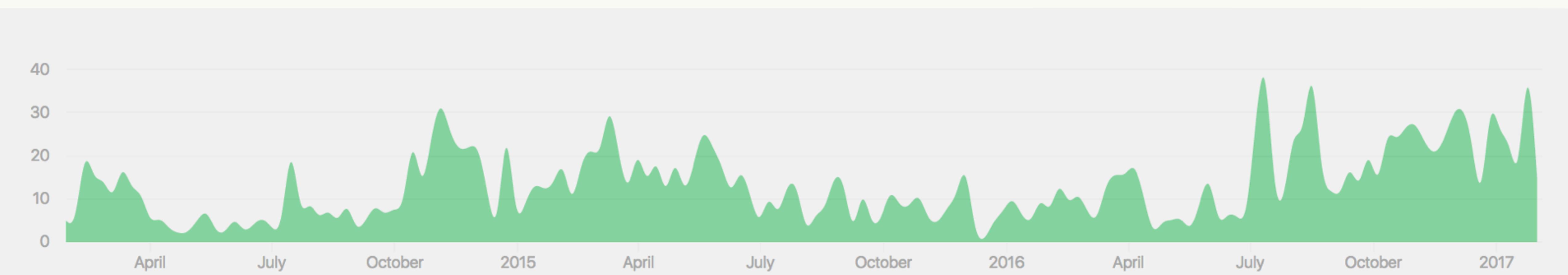
Tasting tests 😊

A History for UI Tests for Cookpad Apps

Feb 9, 2014 – Feb 16, 2017

Contributions to master, excluding merge commits

Contributions: Commits ▾



Why have we implemented
this UI tests? 🤔

Should we taste from?

Writing unit tests before refactoring is sometimes impossible and often pointless.

Basic strategy

図にする

(internal) production code => |app| <=
(external) users

Make checkable from external to internal

図を用意する

check from external \Leftrightarrow re-write/refactor/
implement from internal (and add unit tests)

Unit tests for Re-Engineering

*Most developers would agree that unit test
should be fully automated,*

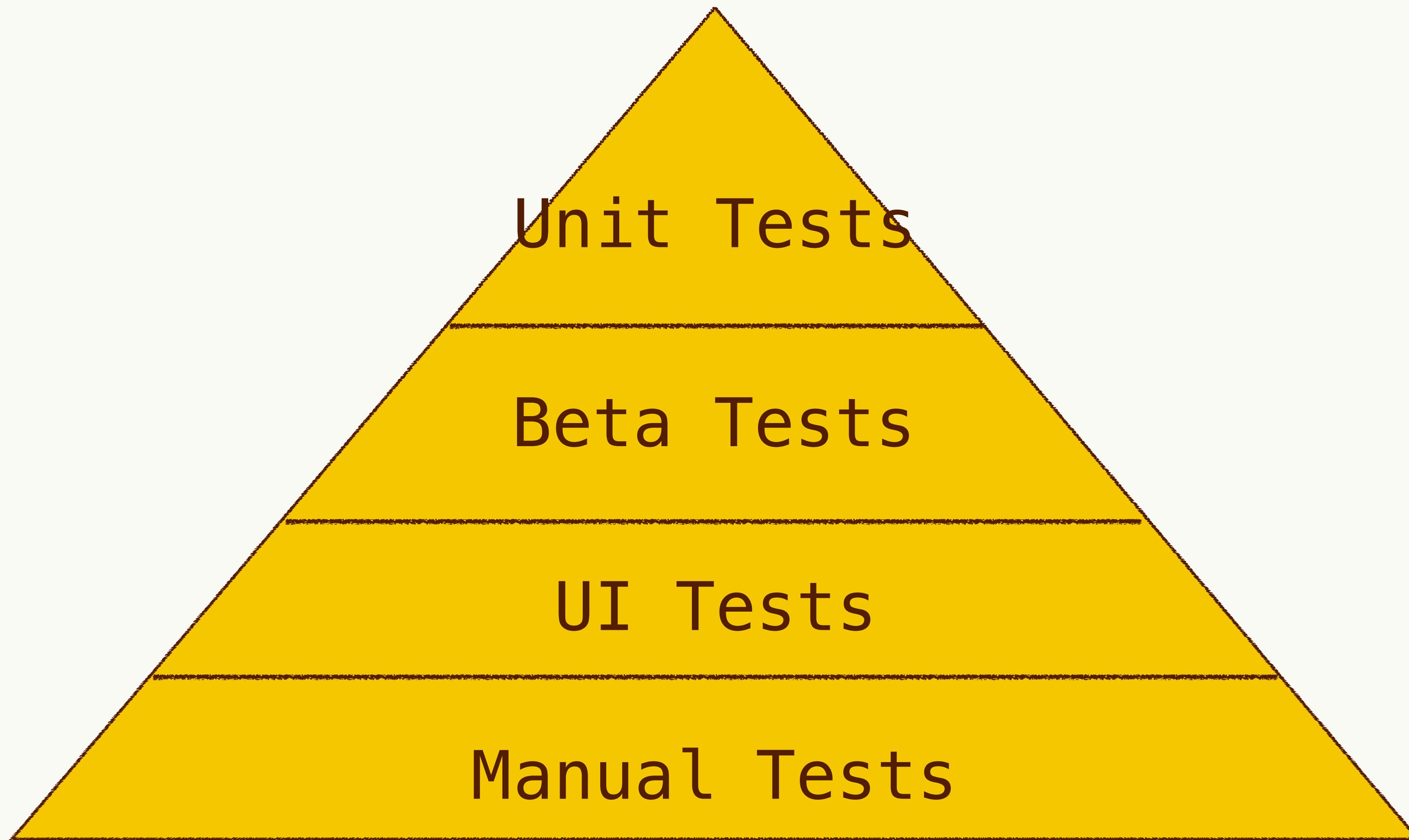
Unit tests are not a silver bullet

but the level of automation for other kind of tests(such as integration tests) is often much lower.

UI Test should be automated

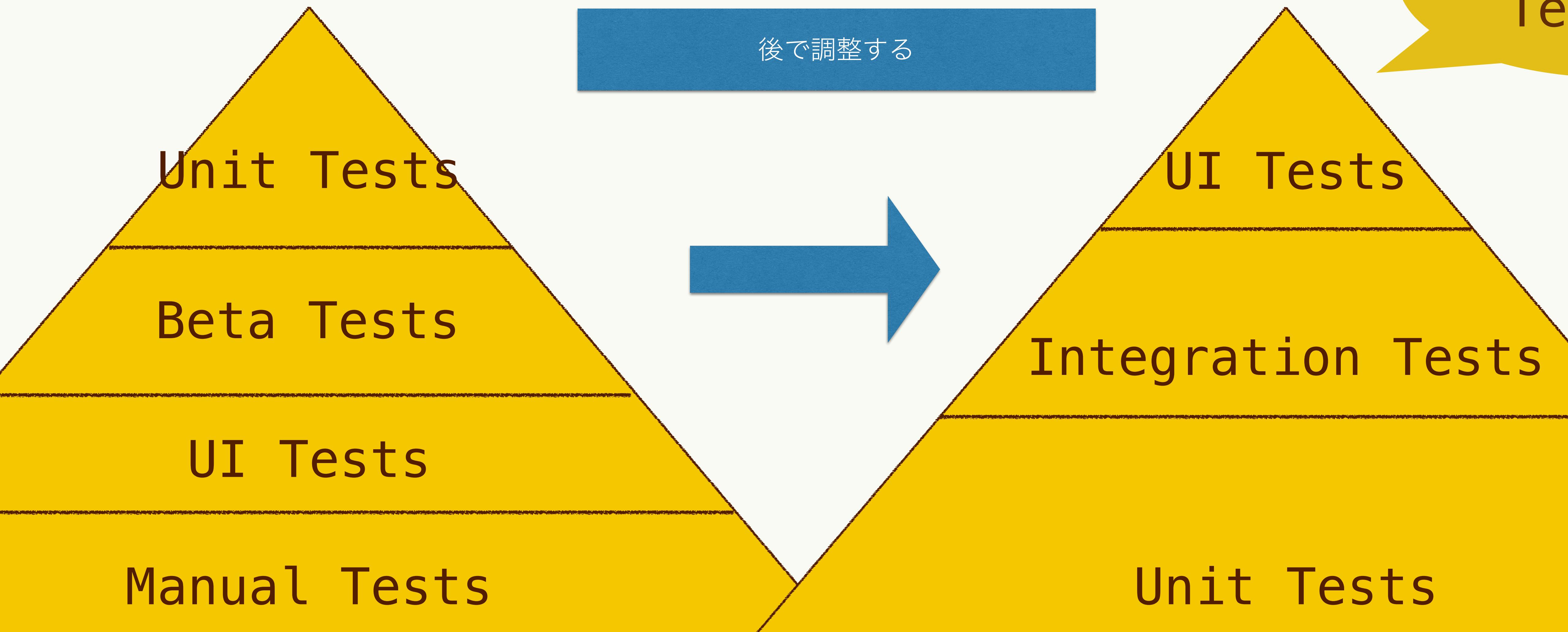
One area that cries out for automation is UI testing.

Flipped pyramid make development cycle slow



- <http://www.utest.com/articles/mobile-test-pyramid>

UI Test should be automated(again)

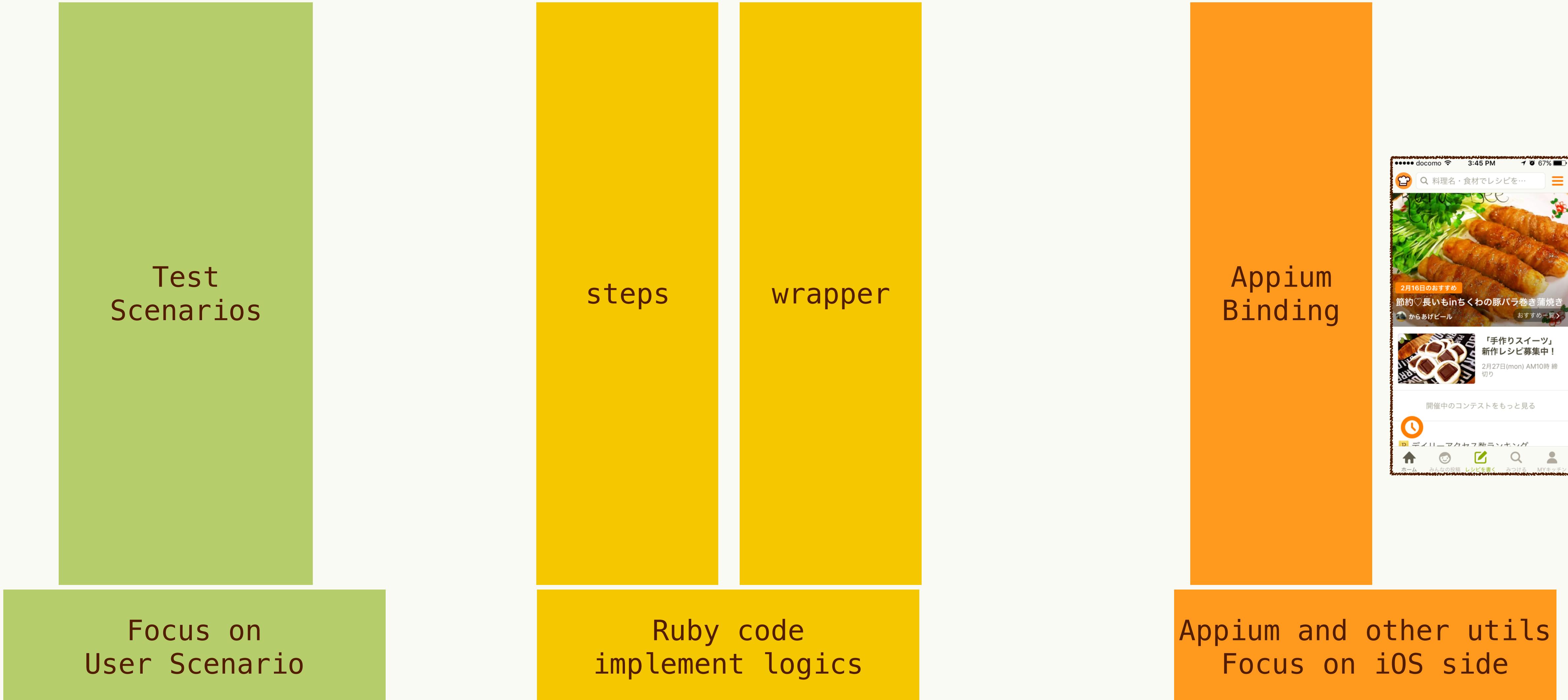


Ma
Te

implement the strategy

Automated UI Test with Appium from 2014

Architecture for UI Tests



Scenarios(英語にする)

feature: A user can search several words via search field

background:

given 'iPhone' で試験を行う

scenario_outline: ユーザは自分のログイン状態によって変化する検索結果を見ることができる

when <user_status> ユーザでログインする

when <search_words> と検索欄から検索する

then 私は '3' 回下側にスクロールする

then 画面に 'xxx' が表示されている

example:

user_status	search_words
'ps'	'ヤシガニ'
'non-ps'	'月食'
'guests'	'カワエビ'
'guest'	'テナガエビ'

Seasoning(味付け)

reduce dependency from internal product code

✗

```
find_element :xpath,  
    “//UIAApplication[1]/UIAWindow[1]/UIATableView[1]/UIATableCell[1]”
```

○

```
find_element :accessibility_id, “an arbitrary identifier”
```

Don't conduct tests for all
boundaries

more

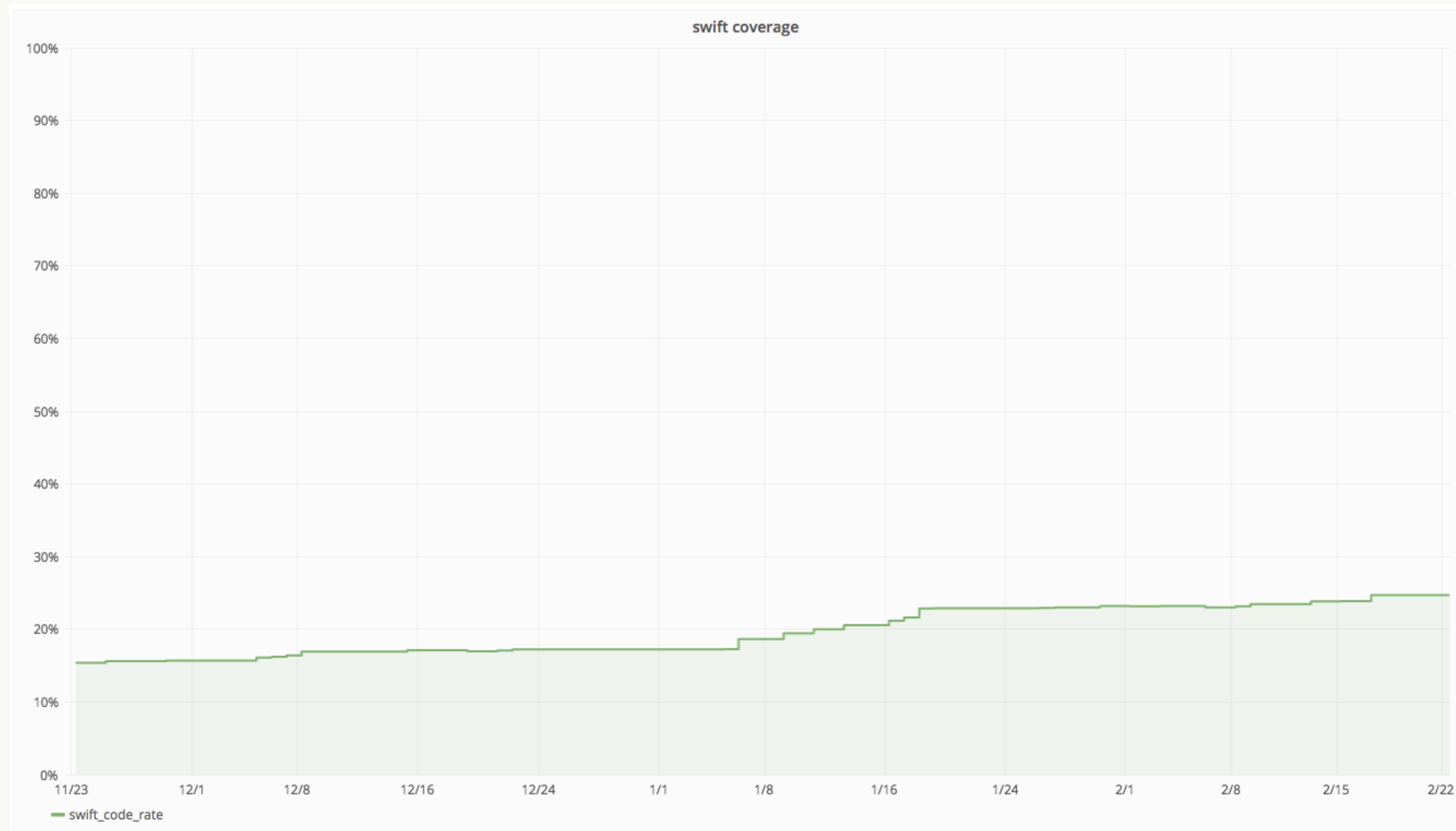


image diff



Re-Engineering - re-write / re-factor without fear for developers

introduce Swift



faster and more stable



Conclusion

Quality, Environment, Services change frequently
in mobile

How automated UI Tests support re-engineering at
Cookpad

What we implement to enhance the automation in
addition

Do you get motivations to
implement automated UI Tests?

Thanks

同僚のicons + Thanks