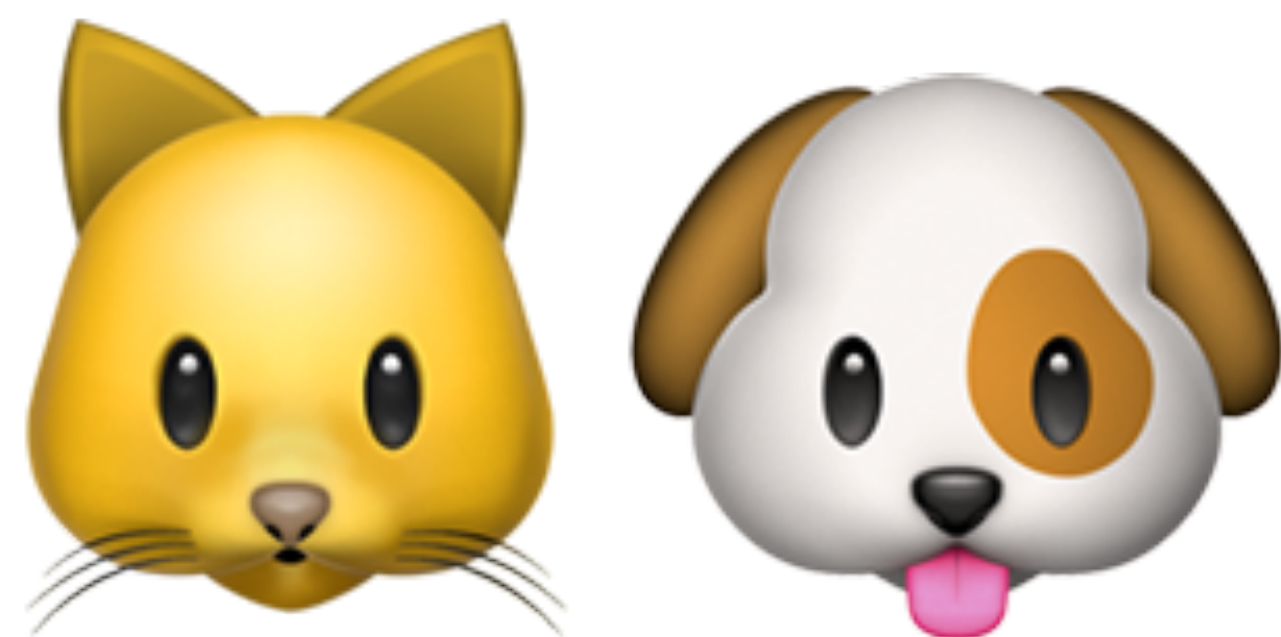




Cats and Dogs

(lessons learned from working as an iOS and Android team)



Romain Piel
Deliveroo
[@_rpiel](#)



Michael May
BBC
[@codermay](#)





Think different.

android

be together. not the same.





#courage

Google unveils Pixel 2 and Pixel 2 XL with no headphone jack

EMIL PROTALINSKI @EPRO OCTOBER 4, 2017 10:04 AM



After rumors and leaks, Google product chief Mario Queiroz today unveiled two new Android Oreo smartphones at the company's annual hardware event: the Pixel 2 and the Pixel 2 XL. Available for preorder starting today, the two phones

#future

George

Ringo

John

Paul

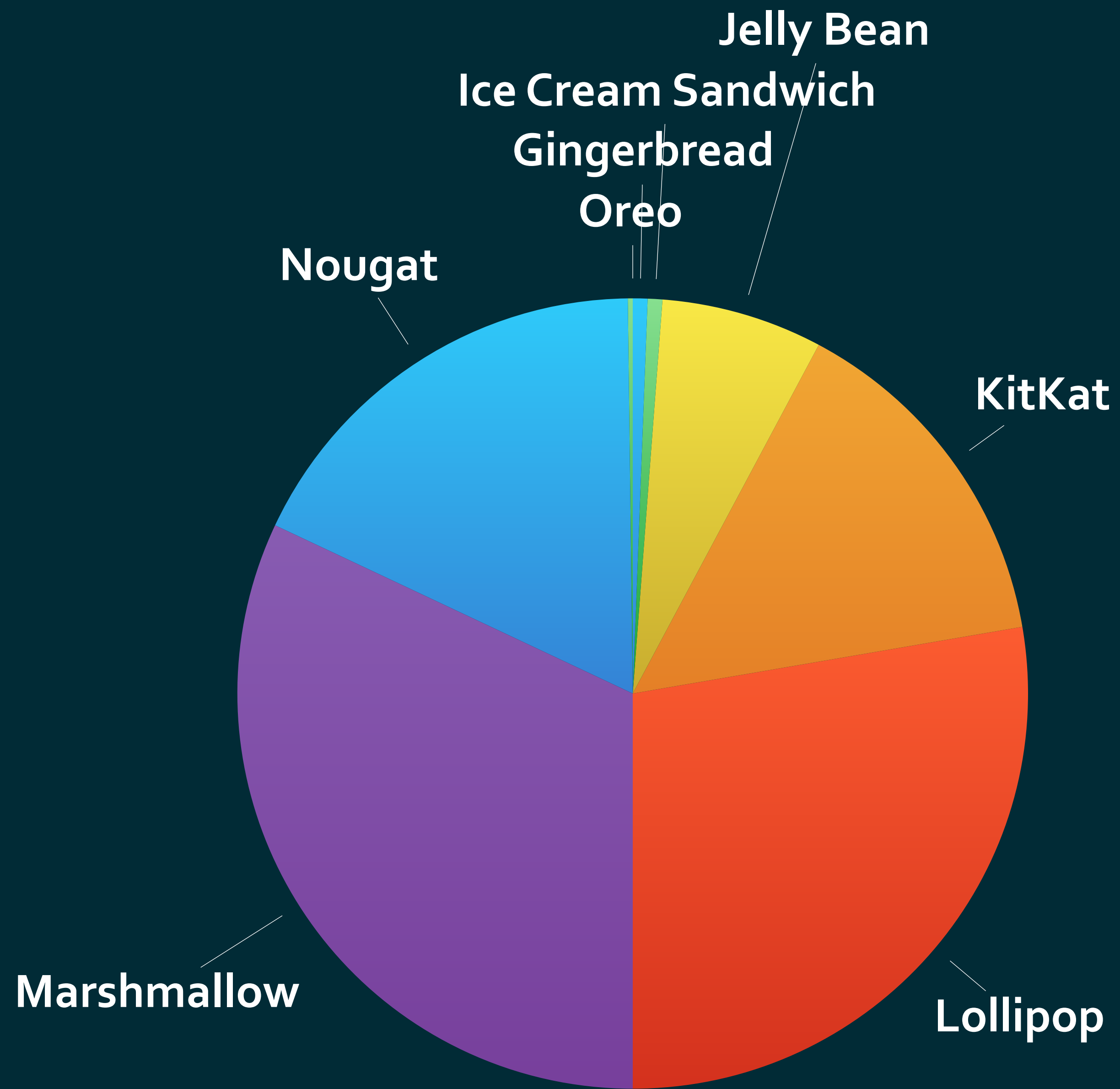
George

Ringo

John

Paul

George




```
- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {  
    NSString *name = [[self products] objectAtIndex:[[[self tableView]  
                                                         indexPathForSelectedRow] row]];  
    [[[segue destinationViewController] navigationItem] setTitle:name];  
}
```



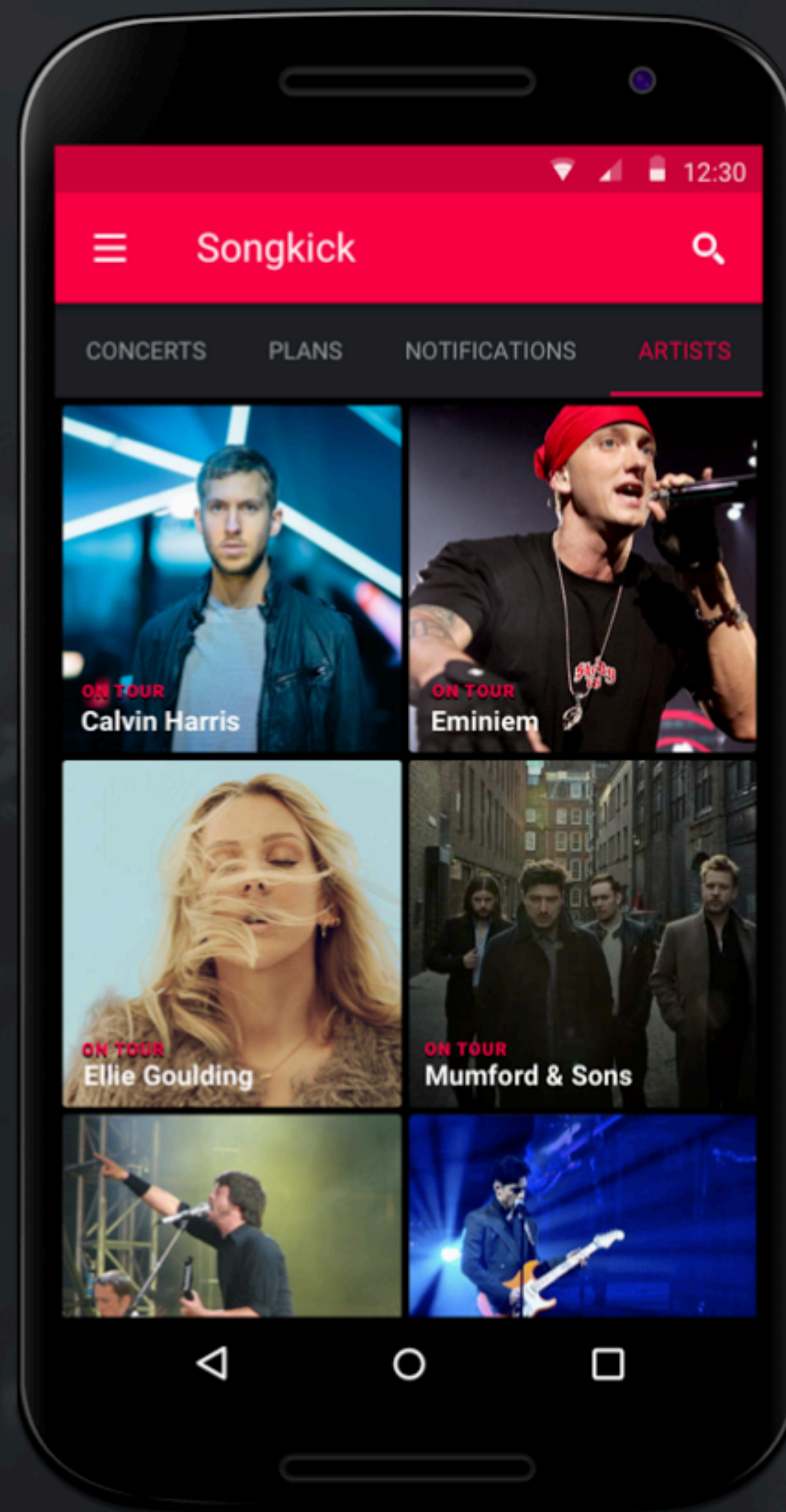

```
var conference = Conference(name: "KotlinConf", date: "11/03/2017")
```



```
var conference = Conference(name = "KotlinConf", date = "11/03/2017")
```




Never miss
another show.





Are you a mobile developer?

Are you a backend developer?

Mobile and backend developer?

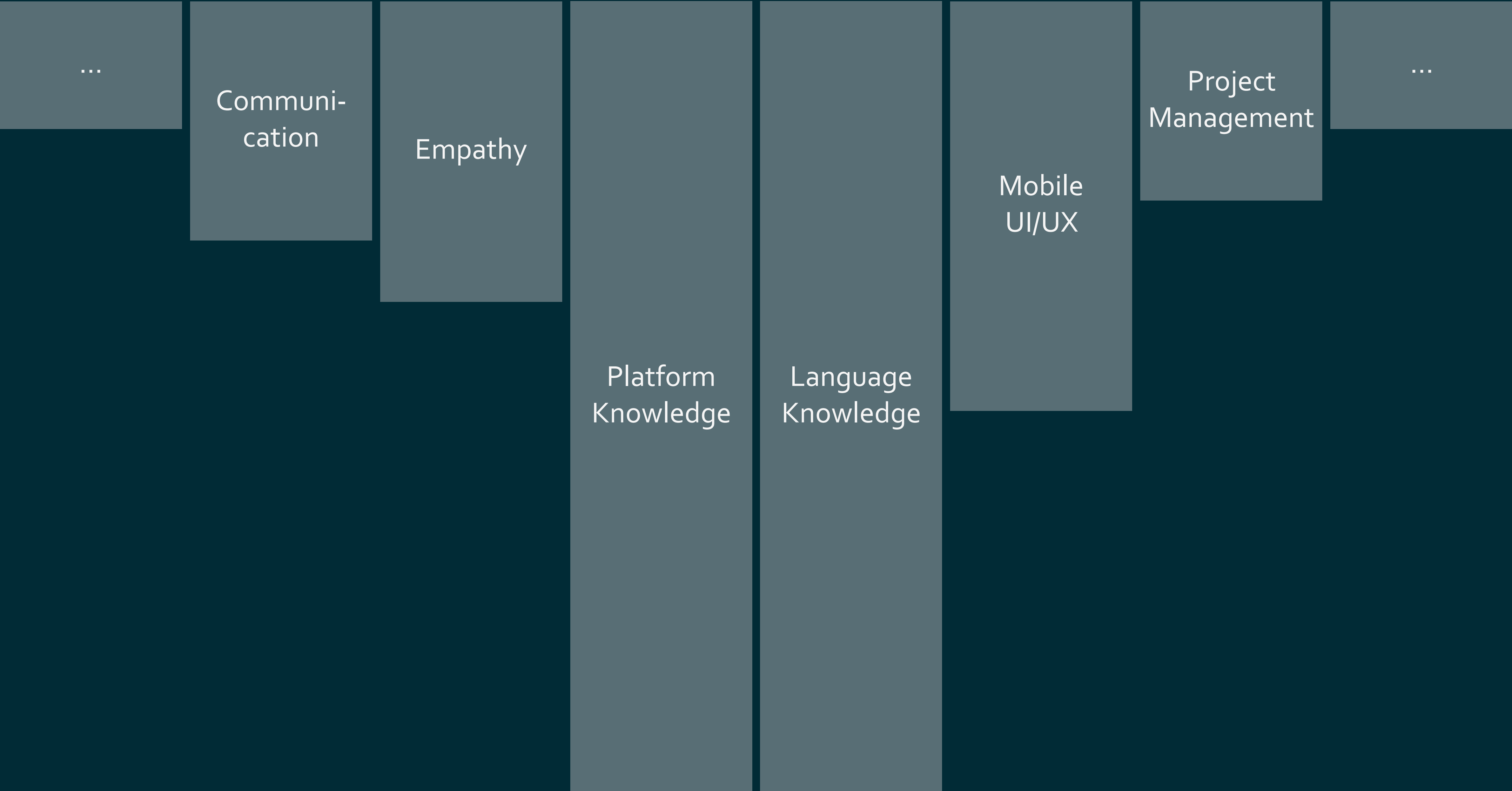
Neither mobile nor backend developer?



Are you mainly a Kotlin developer?

Are you mainly a Swift developer?

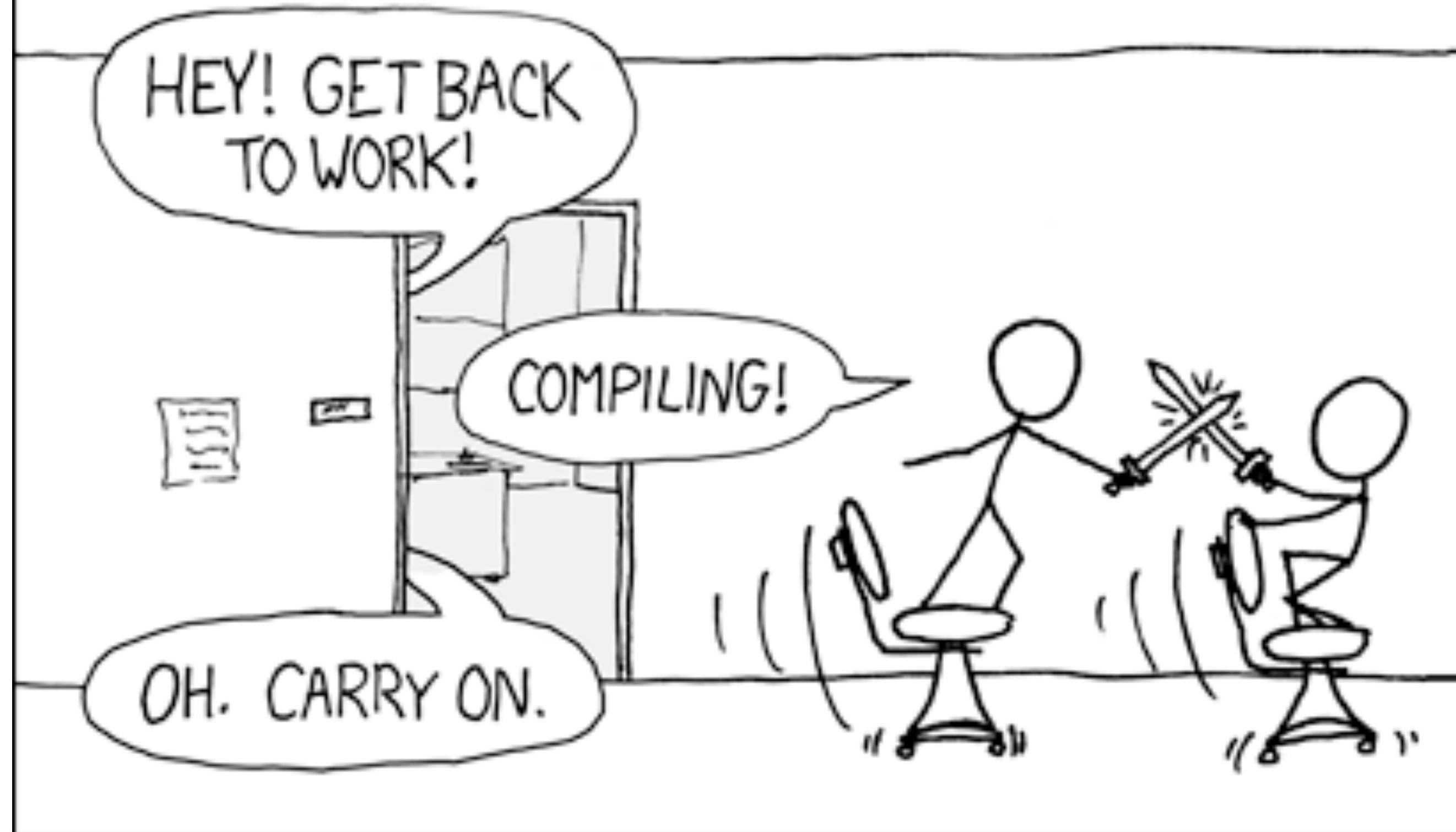
- Kick-off and design together
- Maintain the API as a team
- Pair whenever possible
- QA together
- Share knowledge







THE #1 PROGRAMMER EXCUSE
FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."



- Build a simple schedule app for the conference, on iOS and Android
- Pair on it as much as possible
- Avoid focusing on the same architecture
- Respect the platforms
- Write the iOS app in Swift
- Write the Android app in Kotlin

View

Safe Area

Table View

Activity Indicator

Constraints

Table View.trailing = Safe Ar...

Table View.leading = Safe A...

Table View.bottom = bottom

Table View.top = Safe Area.t...

Activity Indicator.centerY = ...

Activity Indicator.centerX = ...

Prototype Cens

Time

Label

Relation

Equal

Second Item

Superview.Bottom

+

Constant

0

Priority

1000

Multiplier

1

Identifier

Identifier

Placeholder

☐ Remove at build time

+

☒ Installed

CatsAndDogs

Project

CatsAndDogs

gradle

idea

app

build

libs

src

androidTest

main

kotlin

com.romainpiel.catsanddogs

CatsAndDogsApp

Item

ItemAdapter

ItemViewHolder

MainActivity

ScheduleRepository

ScheduleService

res

layout

activity_main.xml

item.xml

menu

main.xml

mipmap-hdpi

mipmap-mdpi

mipmap-xhdpi

mipmap-xxhdpi

mipmap-xxxhdpi

values

AndroidManifest.xml

test

.gitignore

app.iml

build.gradle

proguard-rules.pro

build

gradle

.gitignore

build.gradle

CatsAndDogs.iml

gradle.properties

gradlew

gradlew.bat

local.properties

README.md

settings.gradle

External Libraries

1

<?xml version="1.0" encoding="utf-8"?>

2

<android.support.v7.widget.RecyclerView

3

xmlns:android="http://schemas.android.com/apk/res/android"

4

xmlns:tools="http://schemas.android.com/tools"

5

android:id="@+id/recycler_view_main"

6

android:layout_width="match_parent"

7

android:layout_height="match_parent"

8

tools:context="com.romainpiel.catsanddogs.MainActivity"

9

tools:listitem="@layout/item"

10

tools:menu="main"/>

11

12

Preview

Nexus 4

26

AppTheme

Language

90%

CatsAndDogs

Poniedziałek, 8 maja 2017

16:00

Cats and Dogs (lessons learned from working together as an iOS and Android team)

Michael May, Romain Piel

Poniedziałek, 8 maja 2017

16:00

Cats and Dogs (lessons learned from working together as an iOS and Android team)

Michael May, Romain Piel

Poniedziałek, 8 maja 2017

16:00

Cats and Dogs (lessons learned from working together as an iOS and Android team)

Text

Design

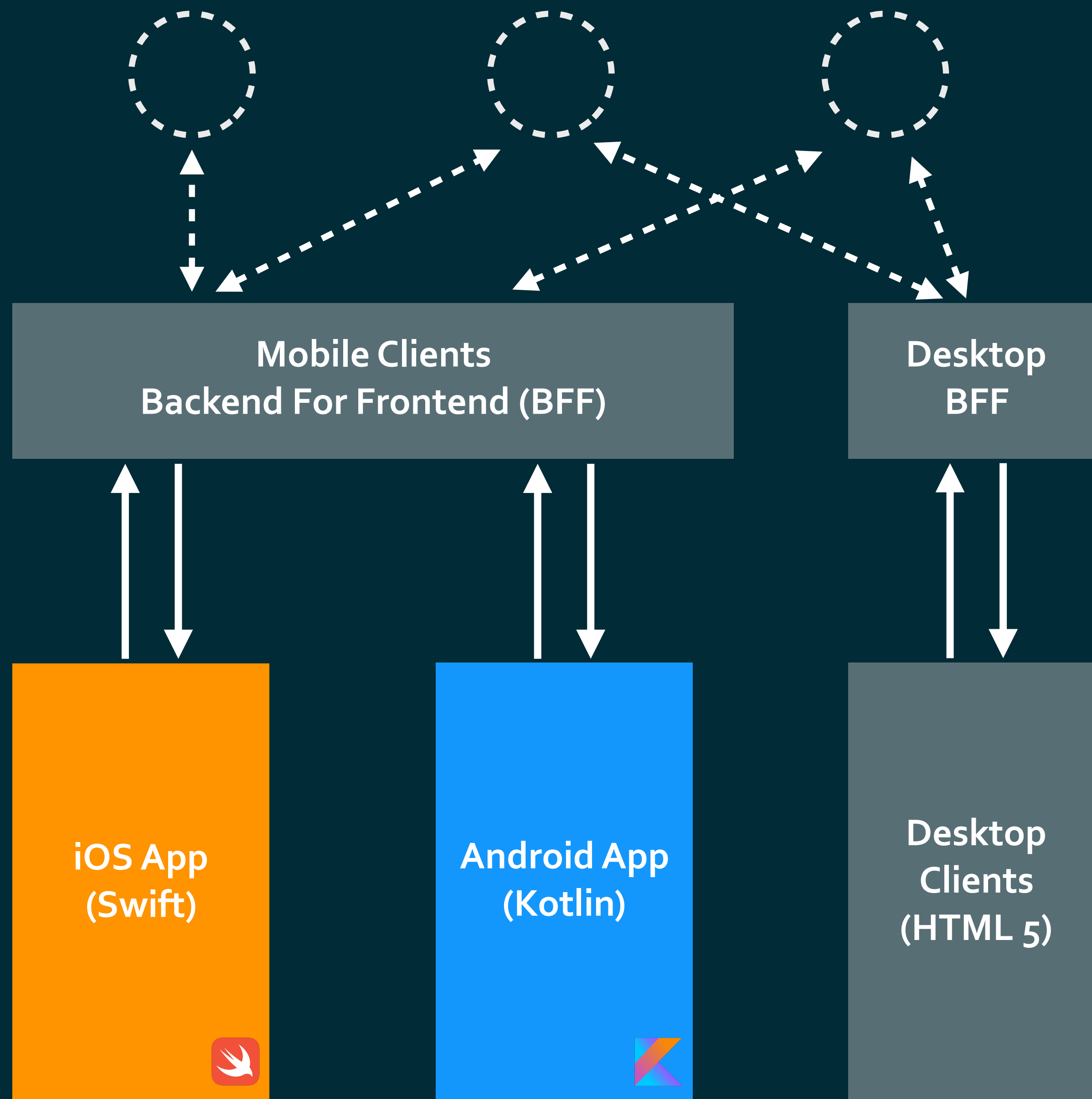
“In 2016, both iOS and Android have first-party layout systems based on Cassowary.”

bignerdranch.com/blog/constraintlayout-vs-auto-layout-how-do-they-compare

michael as? KotlinDeveloper



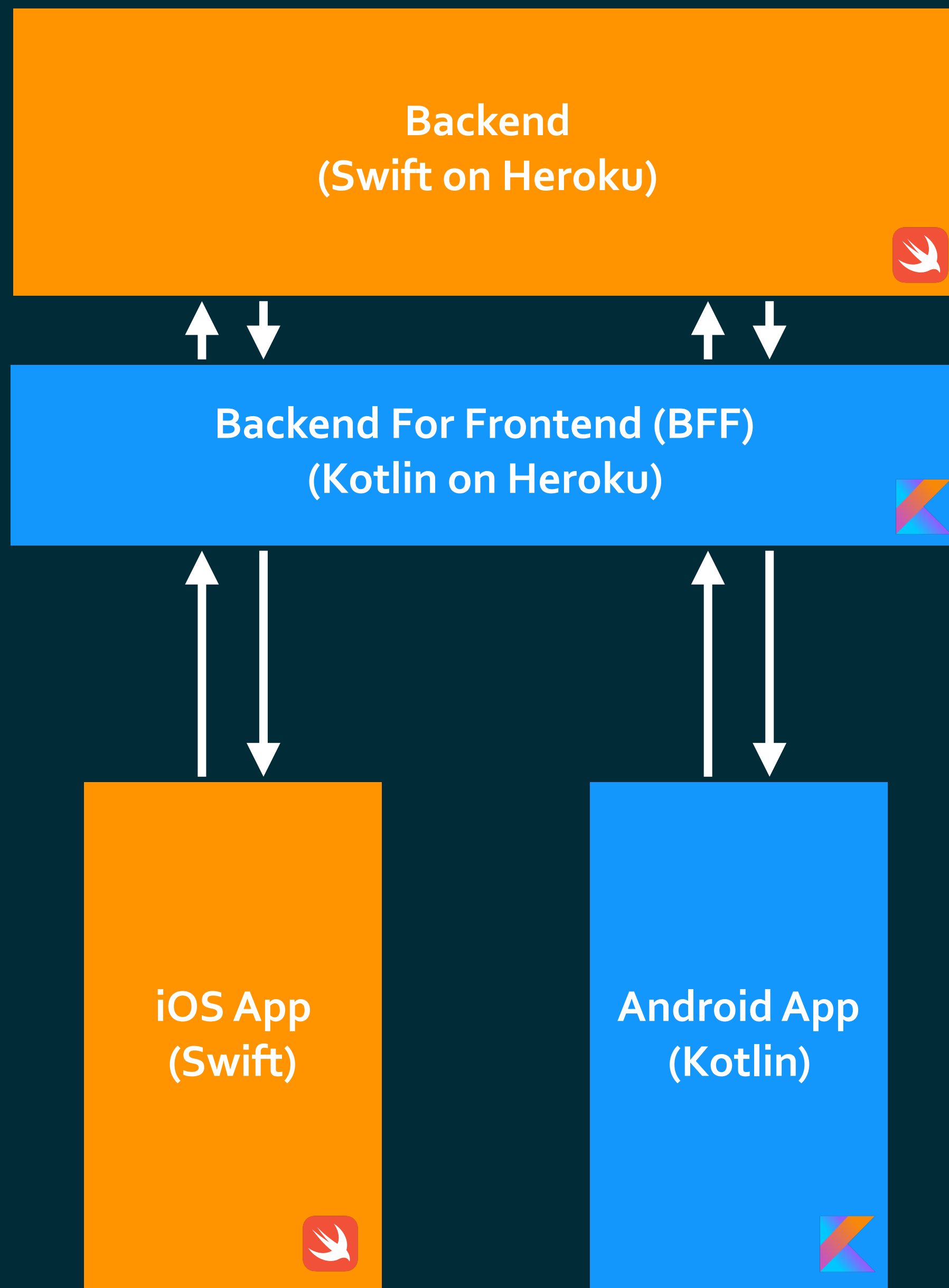




General API (Microservices)

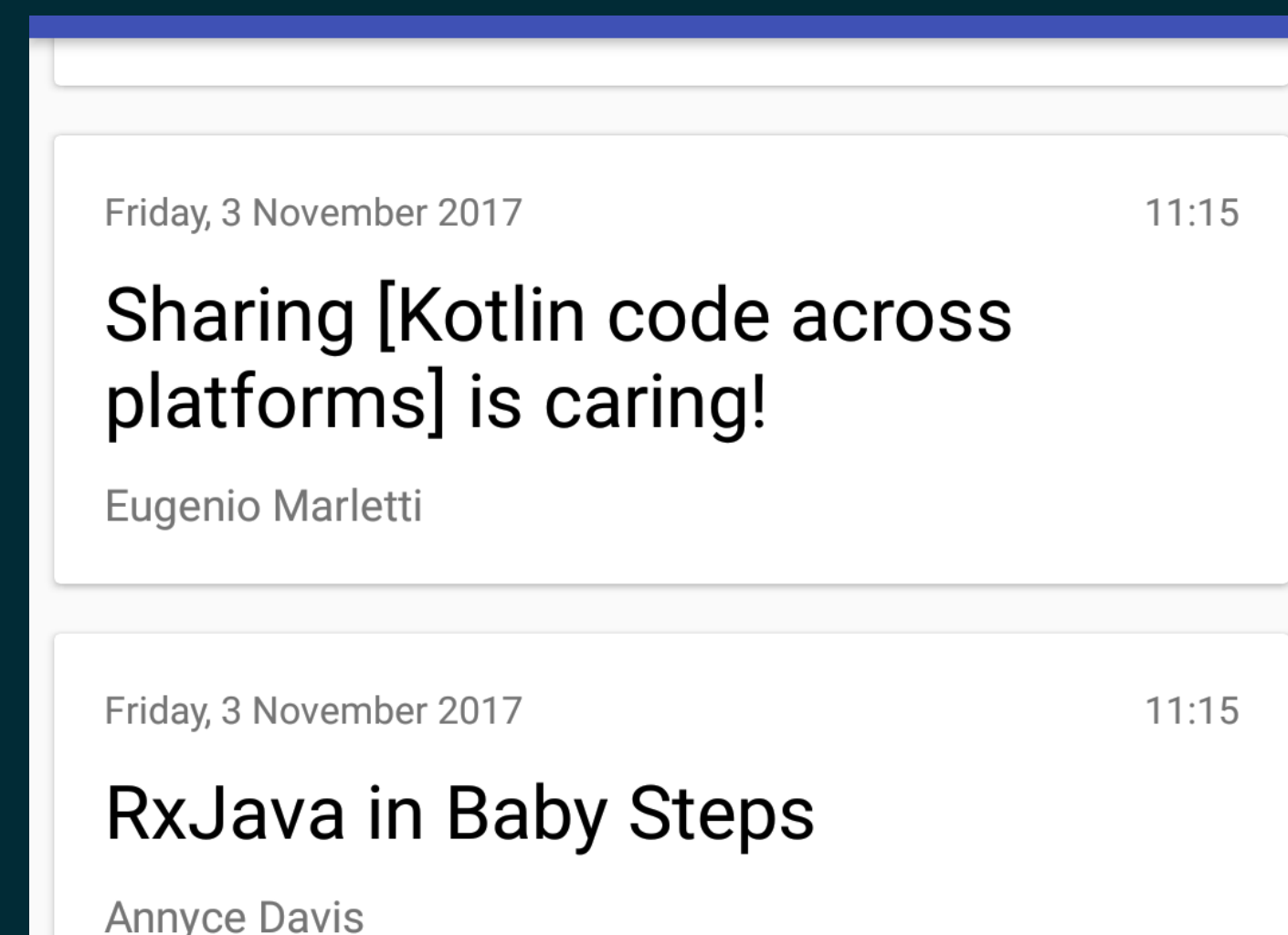
Backend For Frontend (BFF)
(Microservices)

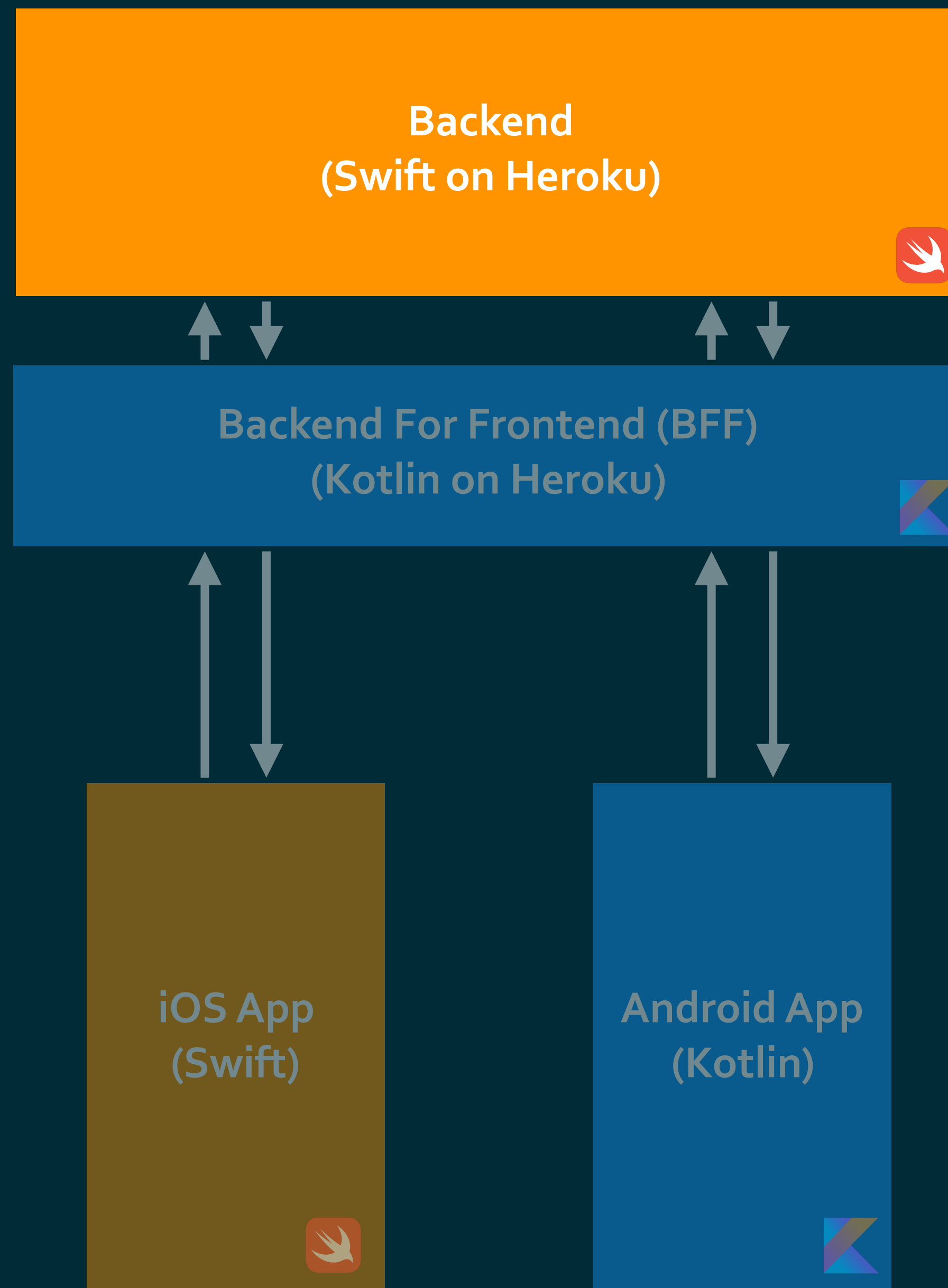
Platform Clients



```
[{
  "datestamp": "2017-11-03T11:15:00-07:00",
  "title": "Sharing [Kotlin code across
platforms] is caring!",
  "speaker": ["Eugenio Marietti"],
  "track": "Room 1"
}...]
```

```
[{
  "title": "Sharing [Kotlin code across
platforms] is caring!",
  "subtitle": "Eugenio Marietti",
  "date": "Friday, 3 November 2017",
  "time": "11:15"
}...}]
```





kitura.io



vapour.codes



perfect.org

```
protocol Event : JSONSerializable {  
    var timezone: TimeZone { get }  
    var locale: Locale { get }  
    var title: String { get }  
    var schedule: [Talk] { get }  
}  
  
struct Talk {  
    let timestamp: Date  
    let duration: Int  
    let title: String  
    let speaker: [String]  
    let track: CustomStringConvertible?  
}
```



```
struct KotlinConf : Event {
    enum Tracks : String, CustomStringConvertible {
        case room1 = "Room 1"
        // ..
    }

    var timezone: TimeZone { return TimeZone.California! }
    var locale: Locale { return Locale.US }
    var title : String { return "KotlinConf" }

    var schedule: [Talk] {
        return [
            Talk(datestamp: dateFormatter.date(from: "2017-11-02T08:00:00-08:00"),
                duration: 60,
                title: "Registration",
                speaker: [],
                track: nil),
            // ..
        ]
    }
}
```

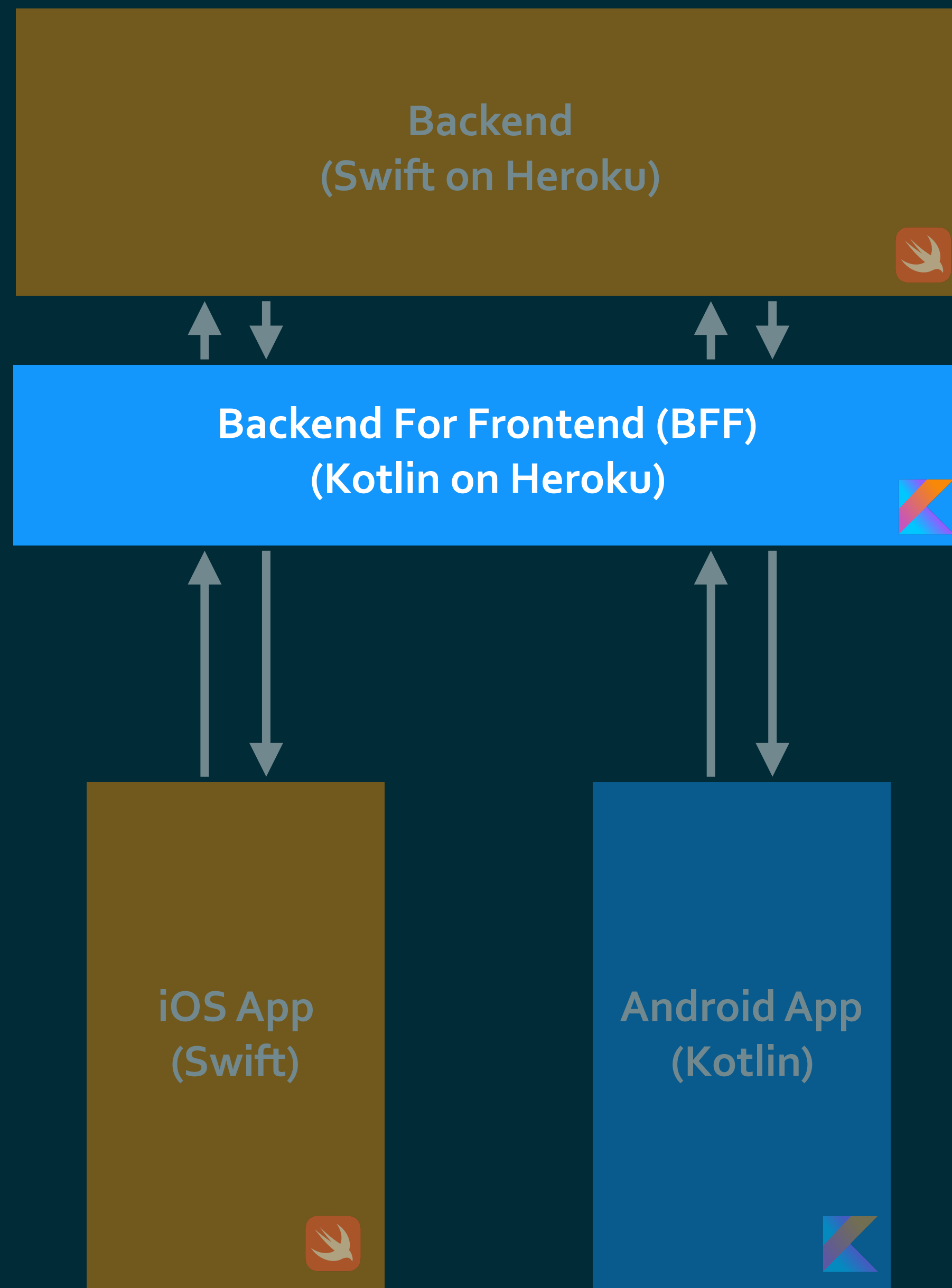
```
import Kitura
import Foundation

let router = Router()

router.get("/kotlinconf/schedule") { _, response, next in
    response.send(KotlinConf().toJSON())
    next()
}

Kitura.addHTTPServer(onPort: 8080, with: router)

Kitura.run()
```

ktor 
ktor.io

 
sparkjava.com

```
data class Item(  
    val title: String,  
    val subtitle: String,  
    val date: String,  
    val time: String  
)
```

```
fun scheduleList(from: OffsetDateTime, locale: Locale, conference:
Conference)
    = service.getSchedule(locale.language, conference.rawValue)
        .map { it.schedule }
        .flatMapIterable { it }
        .filter { /* after selected date */ }
        .map { /* convert to view models */ }
        .toList()

fun schedule(from: OffsetDateTime, locale: Locale, conference:
Conference)
    = scheduleList(from, locale, conference).toJson(gson)
```



```
embeddedServer(Netty, port) {  
    routing {  
        get("/{conference}/schedule.json") {  
            val conference =  
                Conference.instance(call.parameters["conference"])  
  
            val from = call.request.queryParameters.getDate("from")  
            val acceptLanguage = call.request.headers["Accept-Language"]  
            val locale = Locale.forLanguageTag(acceptLanguage)  
  
            val json = scheduleRepository.schedule(from, locale,  
                conference).blockingGet()  
  
            call.respondText(json, ContentType.Application.Json)  
        }  
    }  
}
```



CatsAndDogs		DATE	TIME
Friday, 3 November 2017			11:15
Sharing [Kotlin code across platforms] is caring!			
Eugenio Marletti			
Friday, 3 November 2017			11:15
RxJava in Baby Steps			
Annyce Davis			
Friday, 3 November 2017			11:15
kscript - Scripting enhancements for Kotlin			
Holger Brandl			
Friday, 3 November 2017			12:00
Lunch 🍴🍷			

Carrier

8:04 PM

Friday, 3 November 2017

11:15

Sharing [Kotlin code across platforms] is caring!

Eugenio Marletti

Friday, 3 November 2017

11:15

RxJava in Baby Steps

Annyce Davis

Friday, 3 November 2017

11:15

kscript - Scripting enhancements for Kotlin

Holger Brandl

Friday, 3 November 2017

12:00

Lunch 🍴🍷

Friday, 3 November 2017

13:00

TBA

Andrey Breslav

Friday, 3 November 2017

13:00

Asynchronous programming with Kotlin coroutines in Spring

Konrad Kamiński

Friday, 3 November 2017

13:00

Kickstarting Kotlin Culture: The Journey from Java to Kotlin



```
struct Item {  
    let title: String  
    let subtitle: String  
    let date: String  
    let time: String  
}
```

iOS App

```
data class Item (  
    val title: String,  
    val subtitle: String,  
    val date: String,  
    val time: String  
)
```

Android App

```
data class Item (  
    val title: String,  
    val subtitle: String,  
    val date: String,  
    val time: String  
)
```

BFF

<https://github.com/michael-r-may/CatsAndDogsiOS/blob/master/CatsAndDogs/Item.swift>

<https://github.com/RomainPiel/CatsAndDogs-kotlin-bff/blob/master/src/main/kotlin/com/romainpiel/catsanddogs/model/Item.kt>

<https://github.com/RomainPiel/CatsAndDogs-Android/blob/master/app/src/main/kotlin/com/romainpiel/catsanddogs/Item.kt>


```
class ItemCell: UITableViewCell {
    @IBOutlet fileprivate var title: UILabel!

    func bound(with item: Item) -> ItemCell {
        self.title?.text = item.title
        // ..
        return self
    }
}
```

```
class ItemViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
    val titleTextView: TextView by bindView(R.id.title)

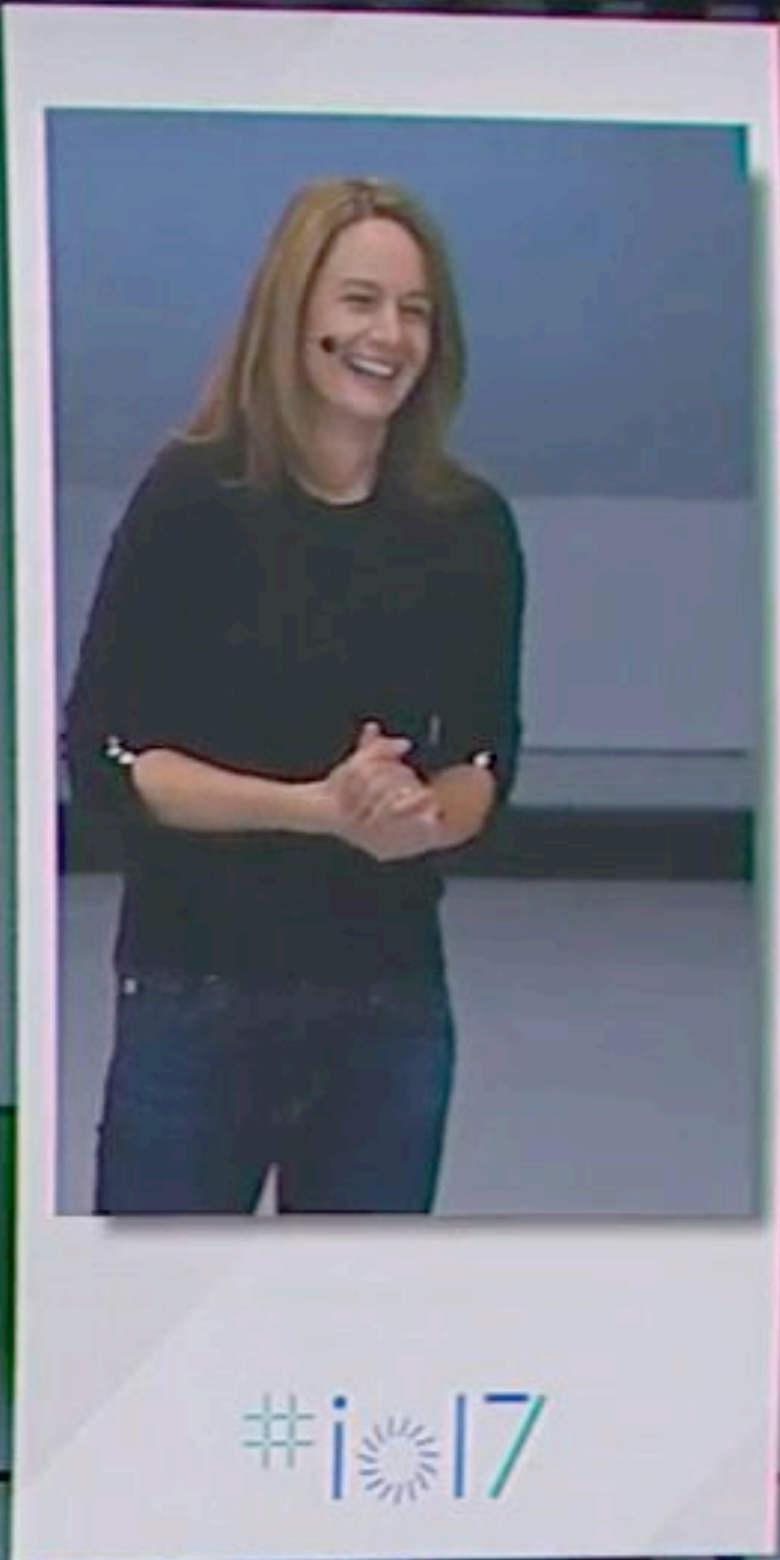
    fun bind(item: Item) {
        titleTextView.text = item.title
        // ..
    }
}
```

<https://github.com/RomainPiel/CatsAndDogs-Android/blob/master/app/src/main/kotlin/com/romainpiel/catsanddogs/ItemViewHolder.kt>

<https://github.com/michael-r-may/CatsAndDogsiOS/blob/master/CatsAndDogs/ItemCell.swift>

```
func allItems(completion: @escaping ([Item])->()){  
    self.downloadSchedule() { jsonData in  
        let items = jsonData?.jsonArray?.items ?? []  
  
        completion(items)  
    }  
}
```

```
fun schedule(from: String?) = service.getSchedule(from)  
    .flatMapIterable { it }  
    .toList()
```


```
UITableViewCell *cell = [[UITableViewCell alloc]  
initWithStyle:UITableViewCellStyle.Subtitle  
reuseIdentifier:@"MyTestCell"]
```



```
let cell = UITableViewCell(style:.Subtitle, reuseIdentifier: "MyTestCell")
```

```
let on = ["", "first", "second", "third", "fourth", "fifth", "sixth",  
"seventh", "eighth", "ninth", "tenth", "eleventh", "twelfth"]  
let amt = ["", "one", "two", "three", "four", "five", "six", "seven",  
"eight", "nine", "ten", "eleven", "twelve"]  
let what = ["", "partridge in a pear tree", "turtle doves", "french  
hens", "calling birds", "gold rings", "lords a leaping", "drummers  
drumming", ...]  
  
for ds in 0...11 {  
    print("On the \ (on[ds+1]) day of Christmas,")  
    print("My true love gave to me,")  
    for d in (0...ds).reversed() {  
        print("\ (d+1==1 ? (ds+1>1 ? "and a" : "a") : "\ (amt[d+1])") \  
(what[d+1])\ (d+1==1 ? "" : ",")")  
    }  
    print("\n")  
}
```

```
override fun onCreateOptionsMenu(menu: Menu): Boolean {  
    if (showMenu) {  
        fill(menu)  
        return true  
    }  
  
    return false  
}
```

```
override fun onCreateOptionsMenu(menu: Menu) =  
    takeIf { showMenu }?.also { fill(menu) } != null
```


<http://bit.ly/kotlinconf-catsanddogs>

This is a quick reference for Swift developers looking to get started playing with Kotlin.

🔄 6 commits

🌿 1 branch

🏷 0 releases

👤 1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

👤 michael-r-may Update README.md		Latest commit bc57ff9 7 days ago
📁 LICENSE.md	Update LICENSE.md	7 days ago
📄 README.md	Update README.md	7 days ago

📖 README.md

Swift Kotlin

A Quick Reference for The Other Side

This is a quick reference for Swift developers looking to get started playing with Kotlin. I began writing it as I started learning Kotlin so that it would stick in my memory better, so that I could refer to it myself, and so that others (like you, dear reader) might benefit and contribute to it.

Real World Example #1

Deliveroo





6:38

×

Order help

LIVE

Estimated arrival

1 minute

John is nearby

✓

Rue Pernelle

Tour Saint-Jacques

Hôtel de Ville

Hôtel de V

Quai de Gesvres

Rue du Renai

Rue du Temple

Rue Lobau

Google

17°C Partly cloudy

John

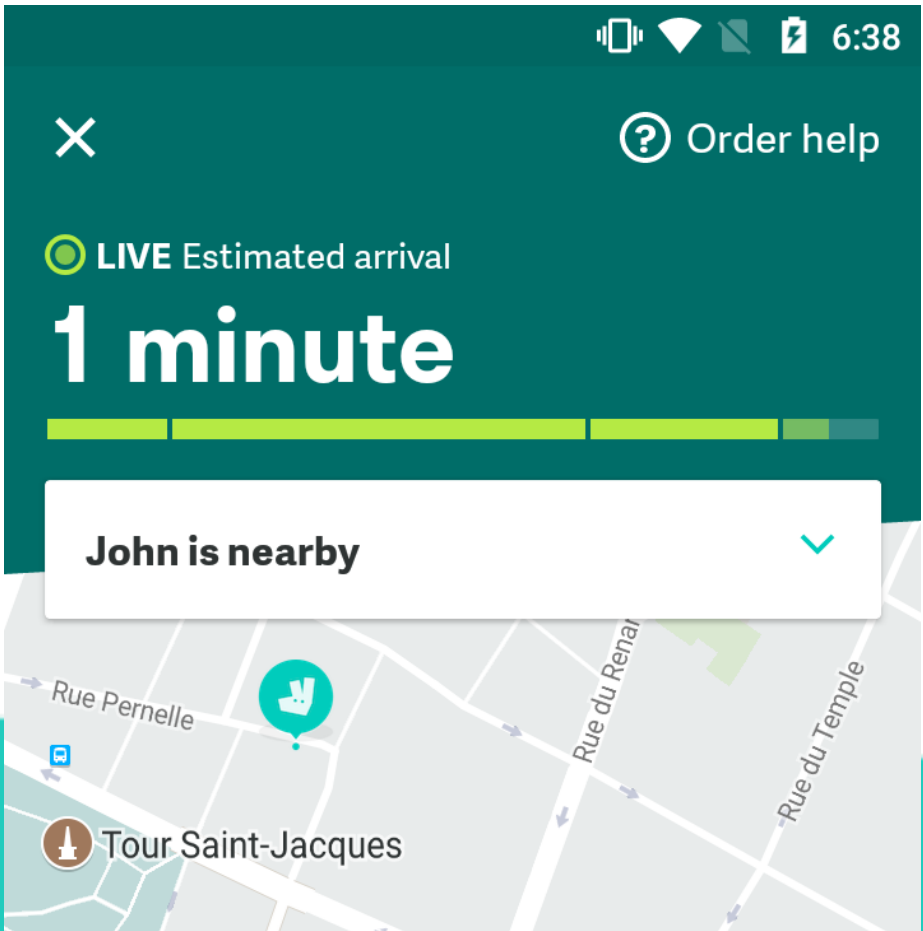
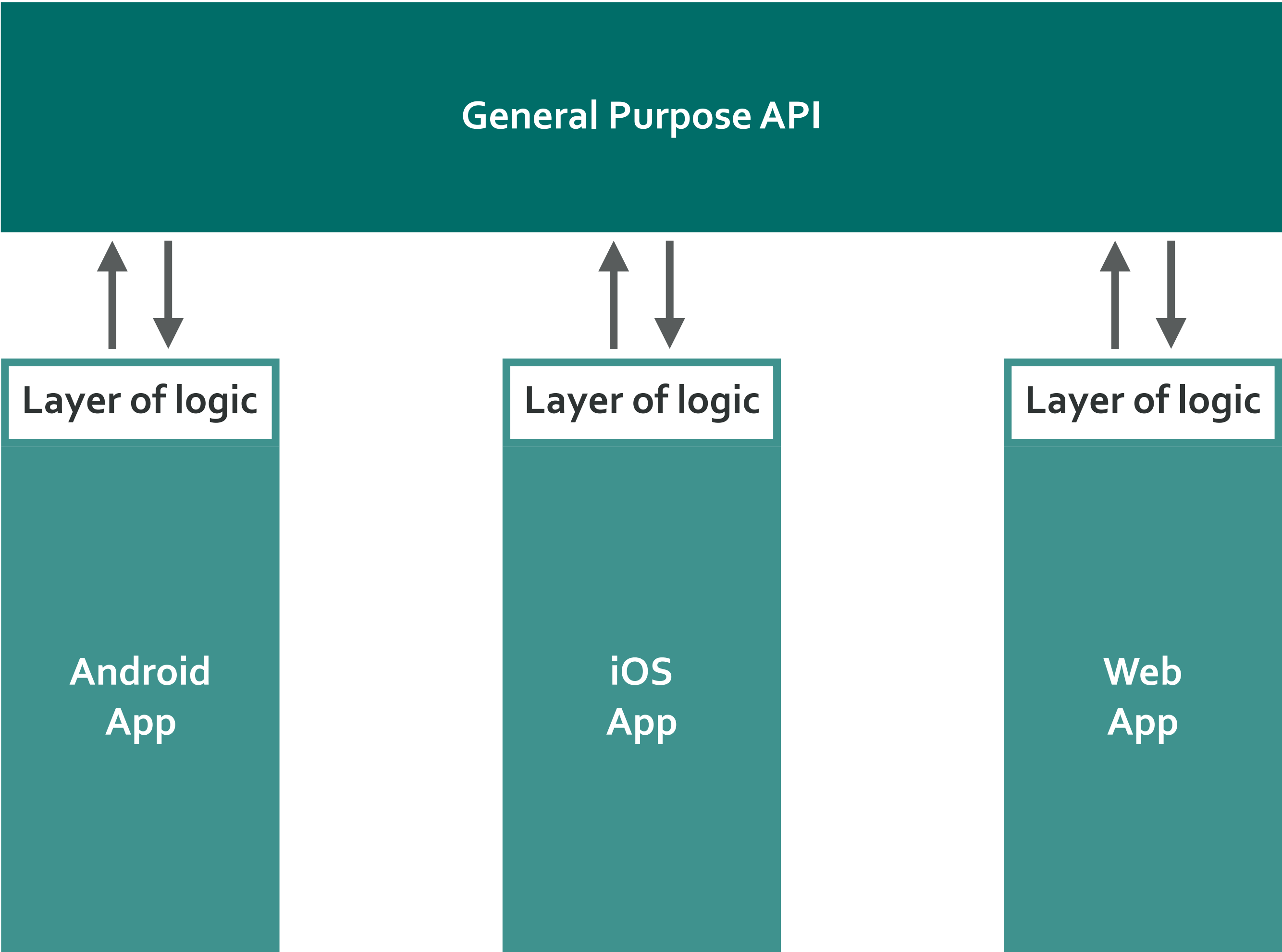
Bicycle

Call

Old Architecture



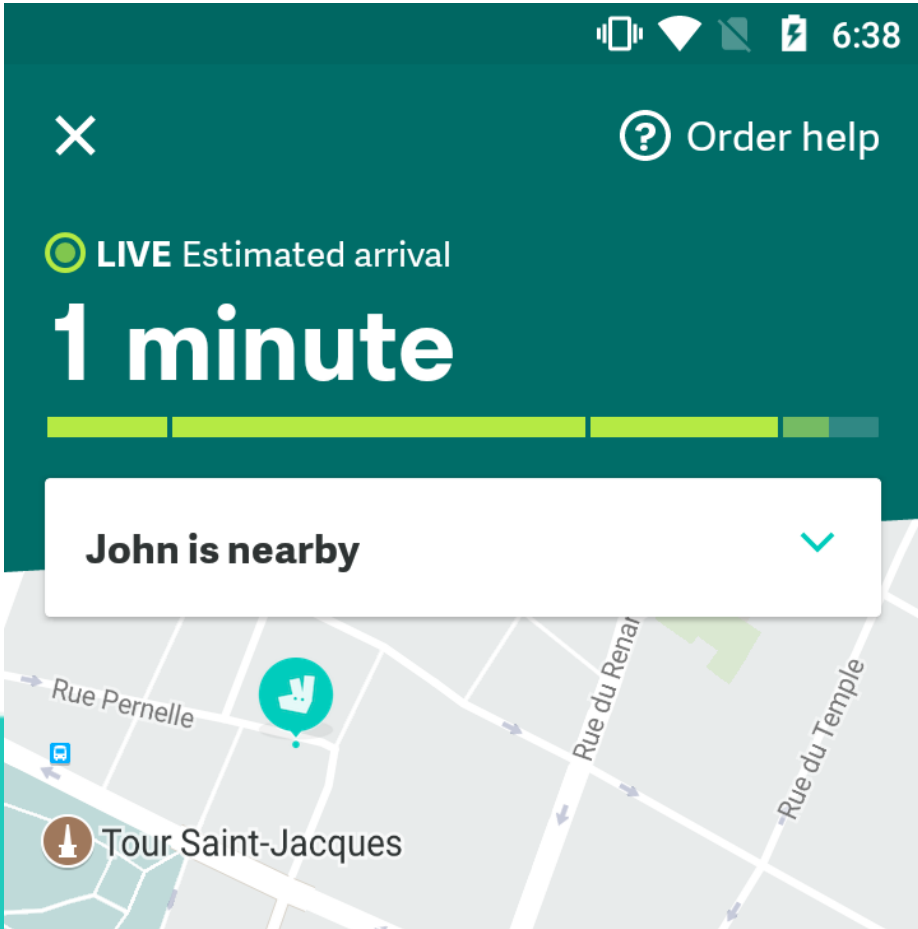
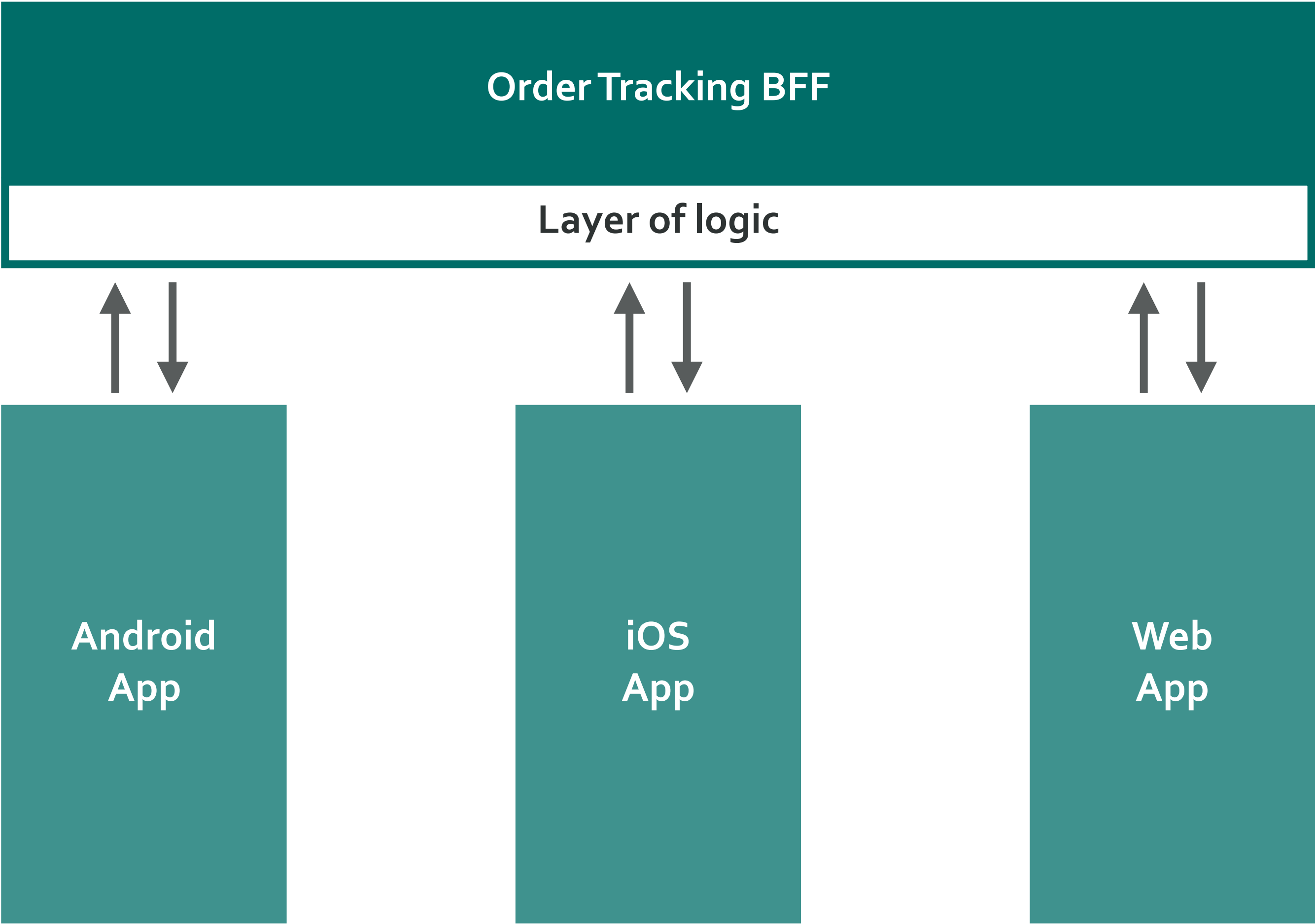
```
{
  "order": {
    "id": 123,
    "status": DELIVERING,
    "estimated_delivery_at":
    "2017-06-01T13:55:28Z"
    ...
  }
}
```



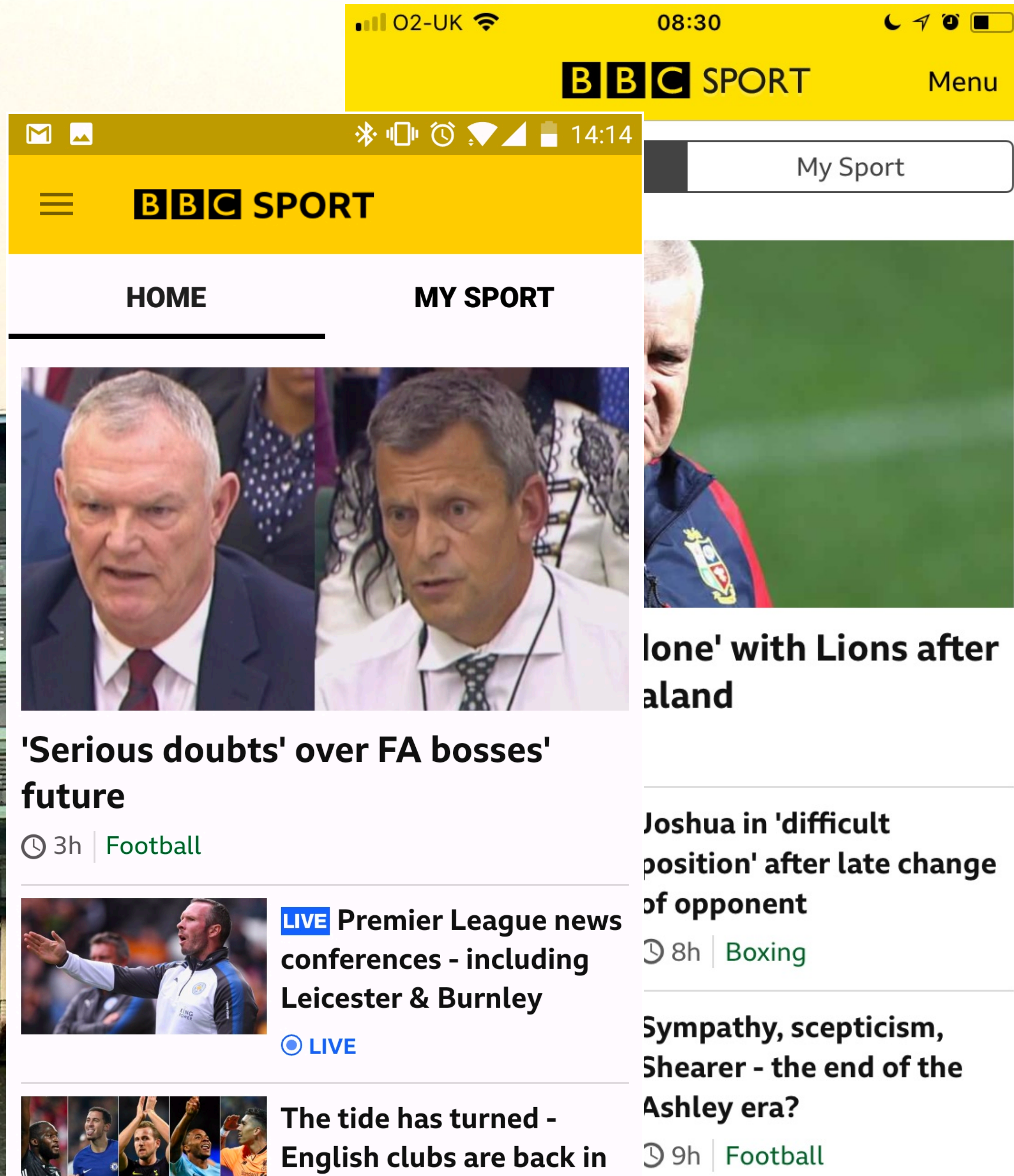
New Architecture



```
{
  "eta_message": "1 minute",
  "message": "John is nearby",
  "show_map": true,
  "progress": {
    "from": 88,
    "to": 100
  }
  ...
}
```



Real World Example #2



images: michael may / bbc sports

“90% of our codebase used to be square brackets!!!”

“[Kotlin,] it’s alright yeah”

“Swift is OK but Kotlin has more fun!”

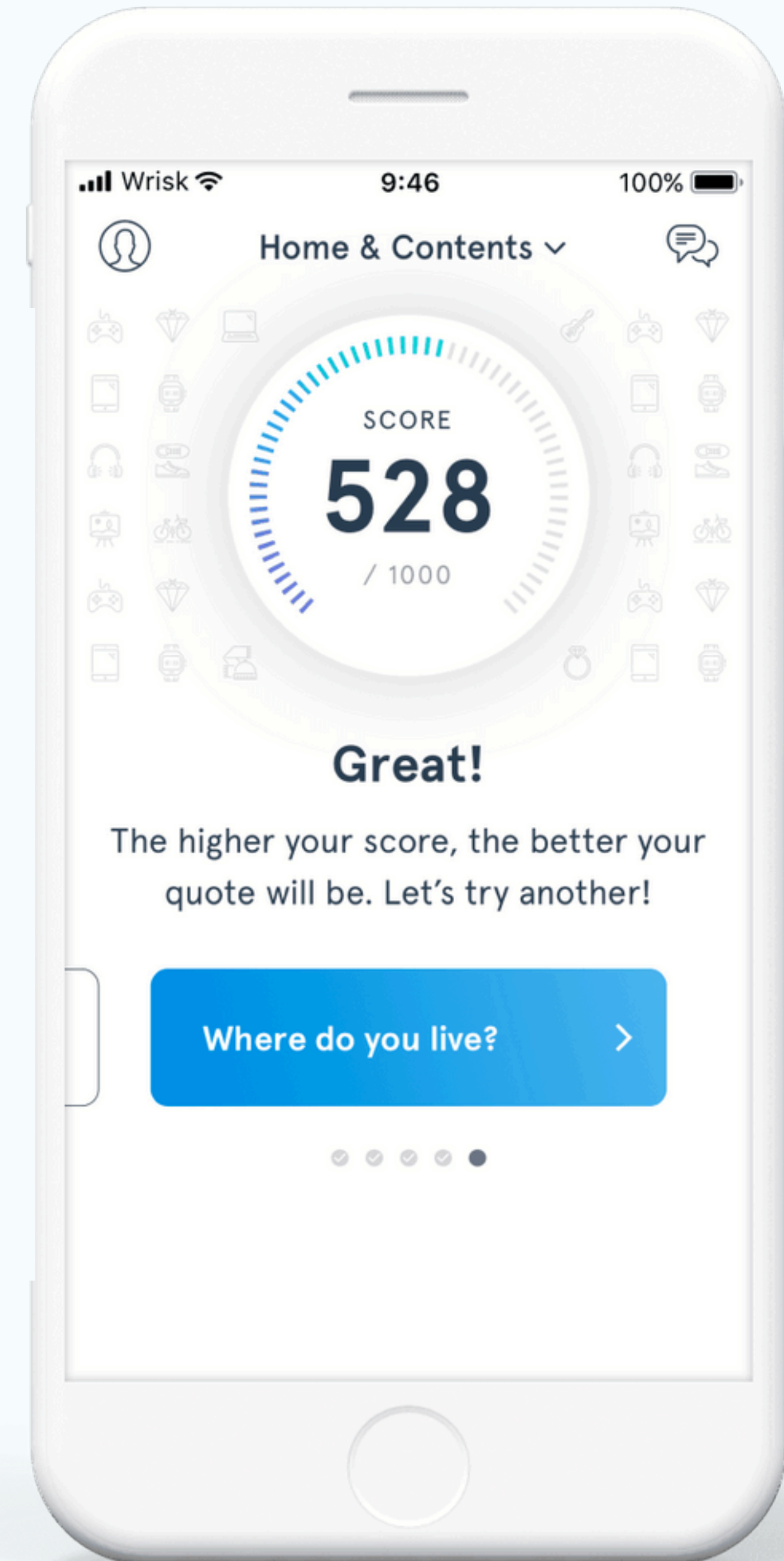
“Yellow card! Just mention something about garbage collection ... :grenade:”

“We’re about 20% Swift and 40% Kotlin now. It’s enabled us to write more concise, readable code with increased safety.”

Real World Example #3



Insurance so simple it's almost unrecognisable



images: wrisk

```
“premium”: {  
  “weekly”: {  
    “country_code”: “UK”  
    “currency_code” : “GBP”  
    “amount: “12.45”  
  }  
}
```

```
let countryCode= json["countryCode"].stringValue
let currencyCode = json["currencyCode"].stringValue
let amount = json["premium"].stringValue

let formatter = NumberFormatter()
formatter.locale = Locale(identifier: countryCode)
formatter.numberStyle = NumberFormatter.Style.currency
formatter.currencyCode = currencyCode

let amountDecimal = NSDecimalNumber(string: amount)

let premium = formatter.string(from: amountDecimal)
```


“Your premium this week will be £12.45”

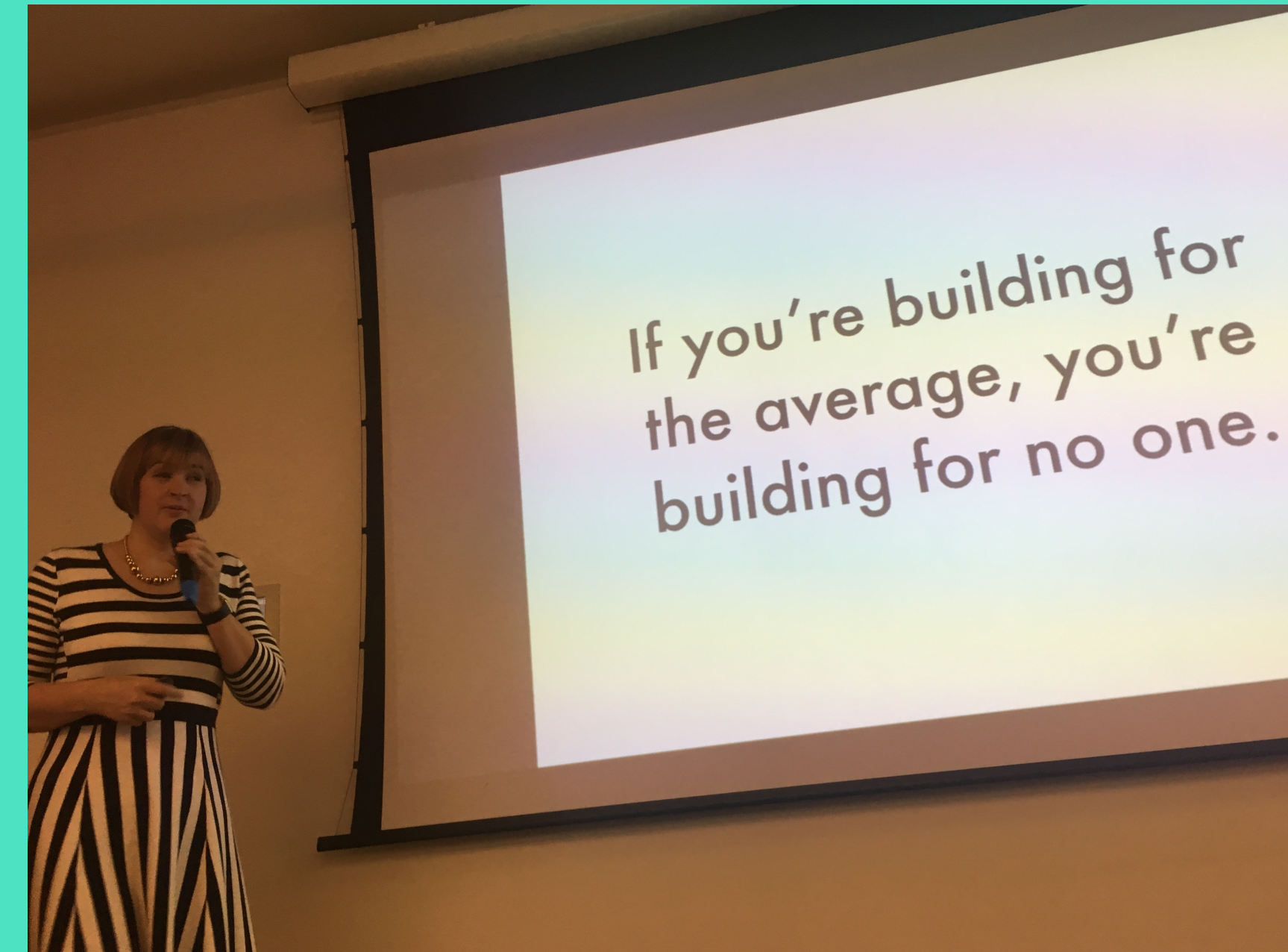
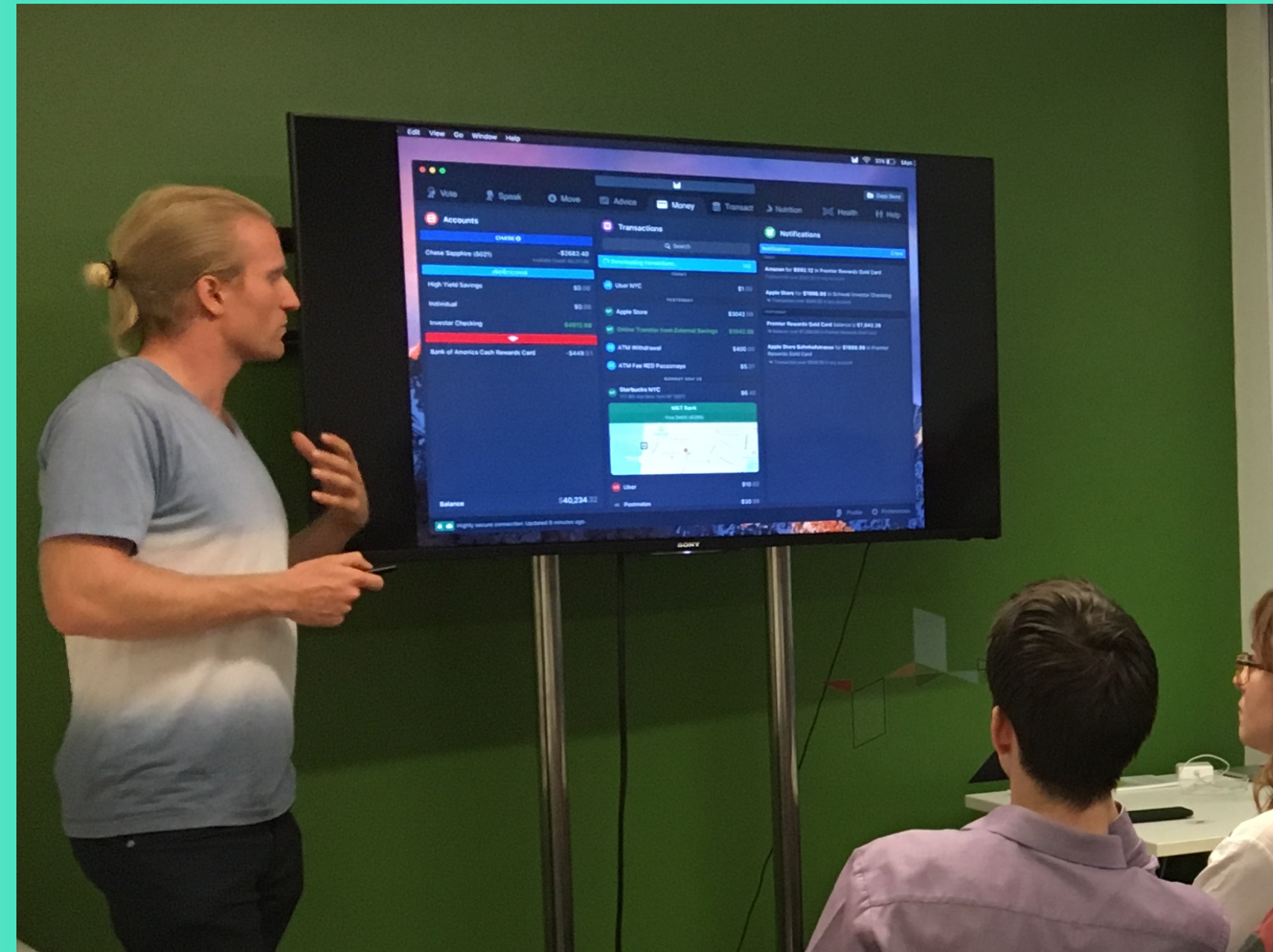
```
“premium” : {  
  “weekly” {  
    “text”: “Your premium this week will be £12.45”,  
    “country_code”: “UK”  
    “currency_code” : “GBP”  
    “amount: “12.45”  
  }  
}
```


“Now we’ve just finished setting up the one way flows of data through the app via the BFF layer [everything] is so much simpler now.”

- Alex, CTO, Wrisk
(October 2017)



// TODO London



From Anime Passion Project to Tech Career

Down The Crypto Rabbit Hole

Communities Of Practice

Never Stop Learning

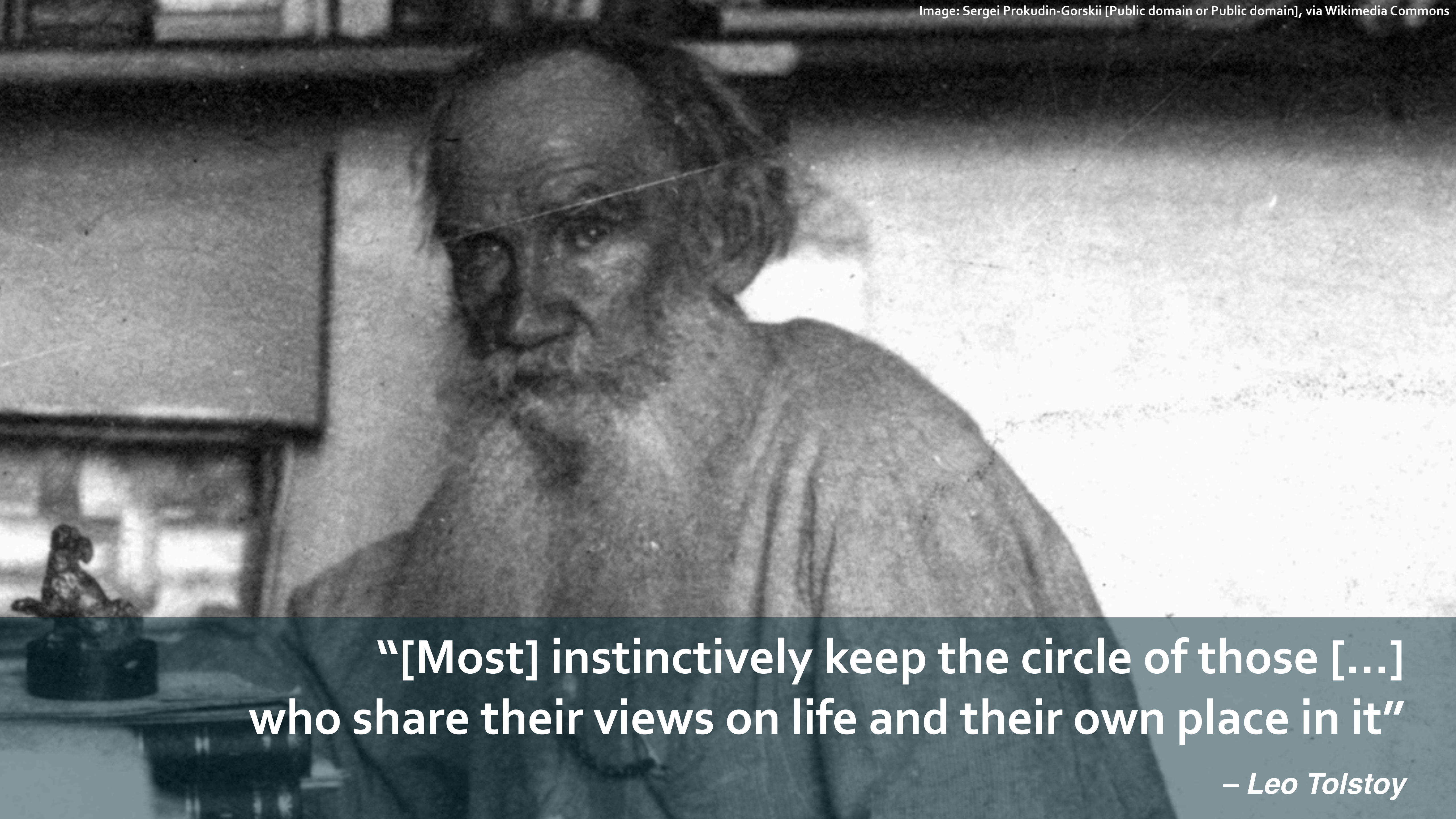
From Product Management to Denim Tailor

```
expect fun soon()
```

```
actual fun soon() {
```

```
    ...
```

```
}
```

“[Most] instinctively keep the circle of those [...] who share their views on life and their own place in it”

– *Leo Tolstoy*

- 
- A group of five people are gathered around a white circular table in an office environment. They are focused on their work, with some looking at laptops and others at smartphones. The office has a modern feel with glass partitions and other desks visible in the background. The text is overlaid on the left side of the image.
- You build better products by collaborating with everyone
 - Collaboration requires shared ways of understanding and communication
 - Which develops empathy between people
 - That allows you to see what you have in common (not just what you don't)
 - And helps you to find the best solutions for everyone
 - Generating a positive feedback cycle

BBC News Apps Team (iOS & Android)
Regression Testing

Getting Started

- Kick off stories together
- Pair whenever you can
- Share your work
- Always QA together
- If you're new to Kotlin or Swift, start with tests
- Talk about the BFF approach with your team
- Be the change you want to see

“Think different, be different, but on the same team”



samnewman.io/patterns/architectural/bff

kitura.io / ktor.io

swift.org / kotlinlang.org

github.com/RomainPiel/CatsAndDogs-Android

github.com/michael-r-may/CatsAndDogsiOS

@_rpiel @codermay

With thanks to Aaron, Alexey, Ikhsan, Anastasia and everyone else at Songkick as well as Deliveroo, the BBC and Wrisk, for letting us talk about our/their work and allowing us to attend **(and, yes, they are all hiring, of course)**.