



Cross (Multi) Platform Mobile App Development

A Myth or Reality

IBM

Vidyasagar Machupalli

- Developer Advocate - IBM Mobile & Cloud
- Polyglot & Pragmatic Programmer
- @VidyasagarMSC
- <http://vidyasagarmsc.com>



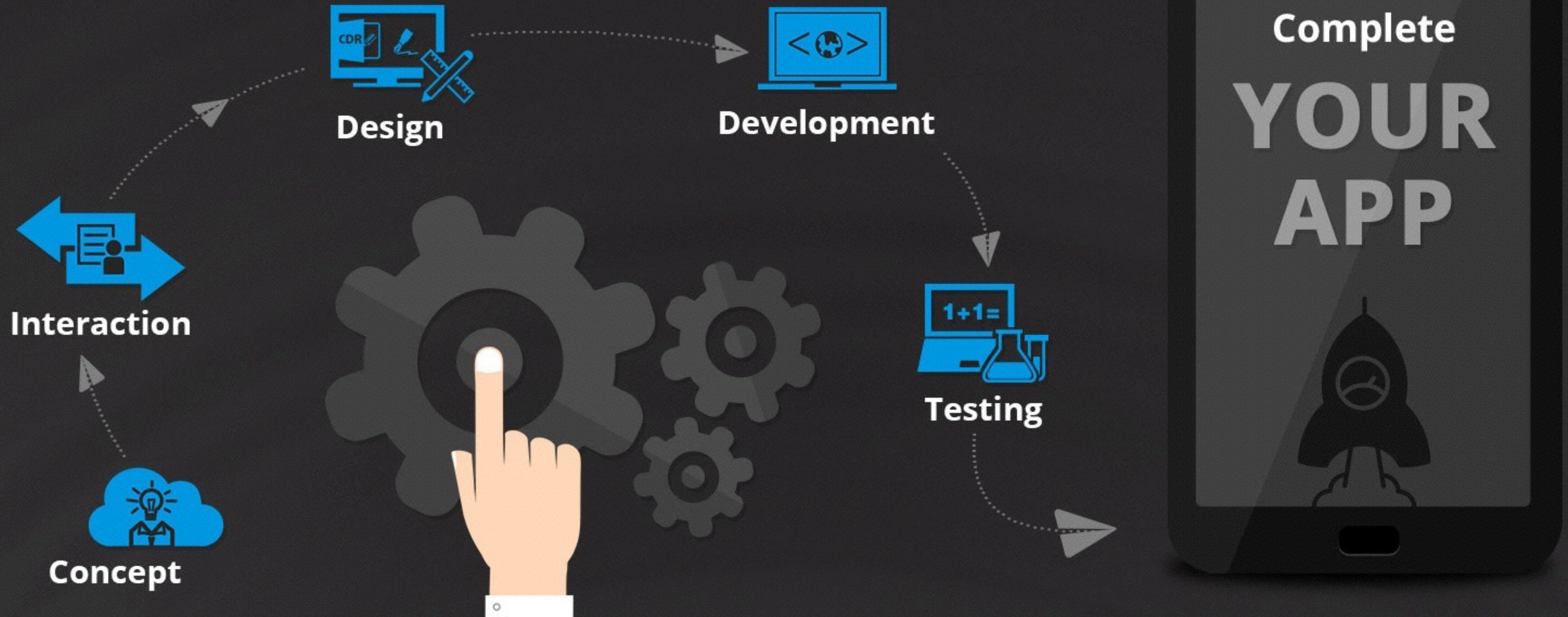
Mobile Phones > Human Population



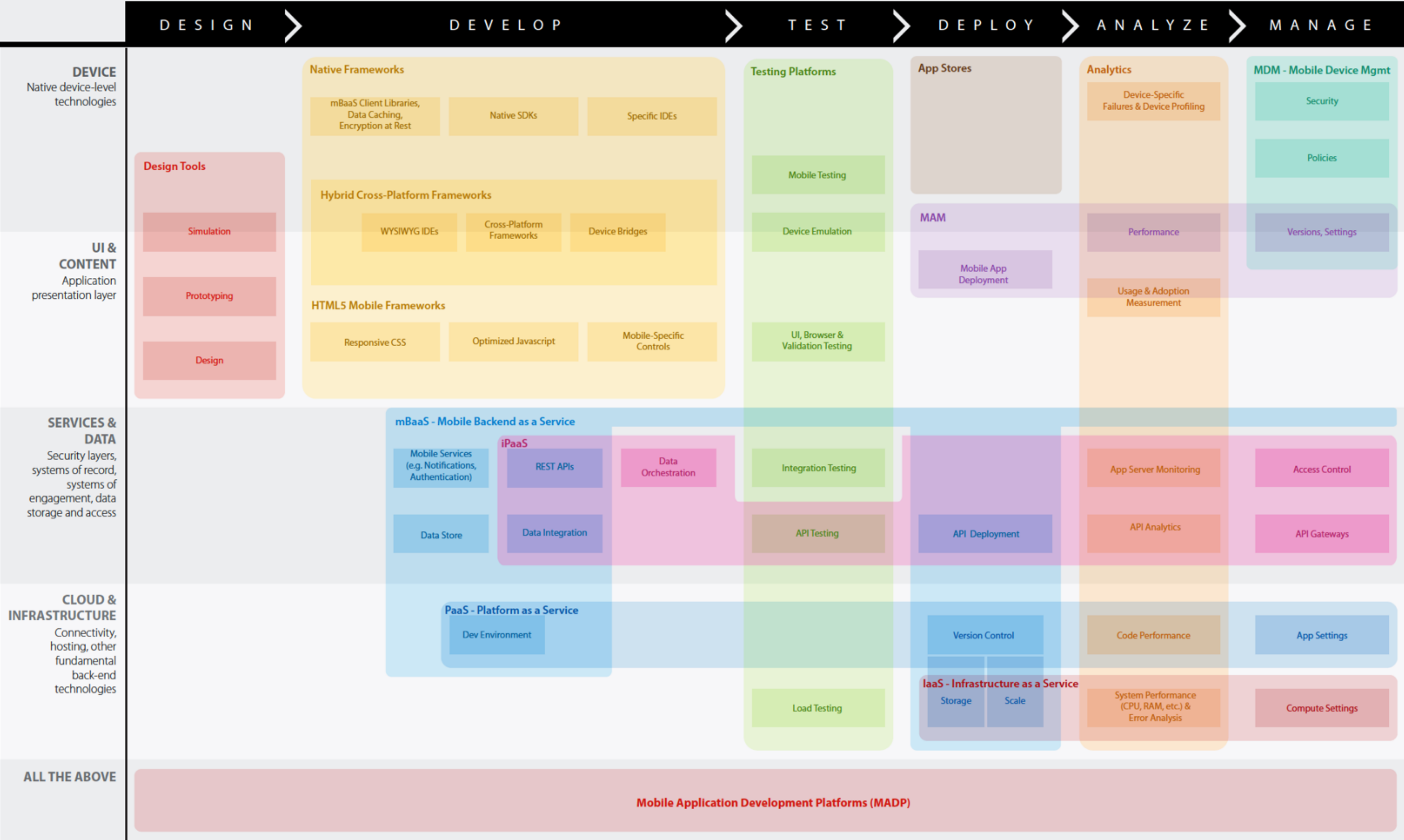
What are Mobile APPs?



Mobile App Development
PROCESS



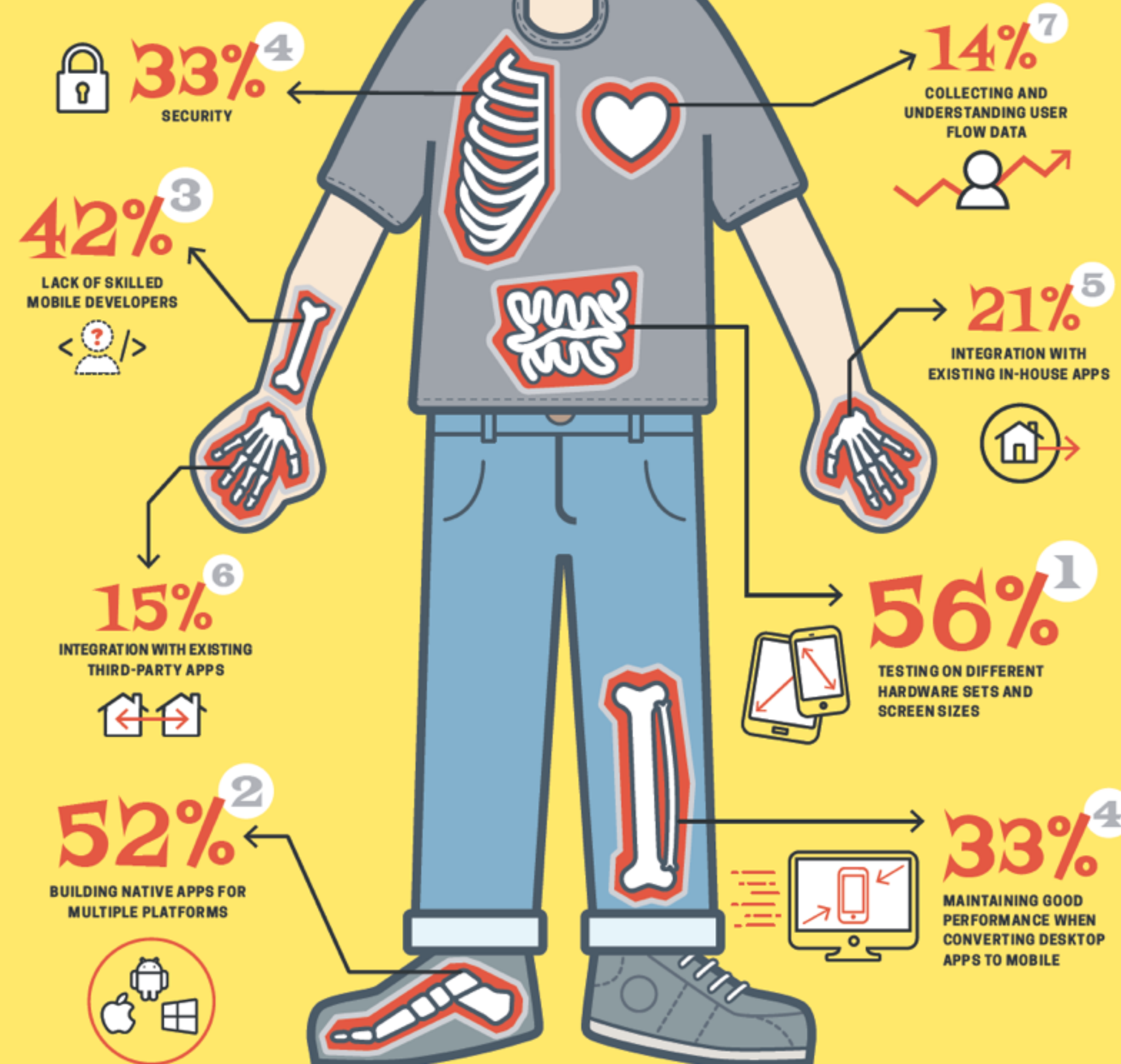
A View of the Mobile App Development Landscape



MOBILE DEVELOPMENT PAIN POINTS

Let's face it — mobile development is tough. Developers encounter several problems, often unforeseen, throughout the development process. Facing these problems can set back development by days, weeks, or even months. Not anticipating these issues can cause a lot of frustration and can lose you money.

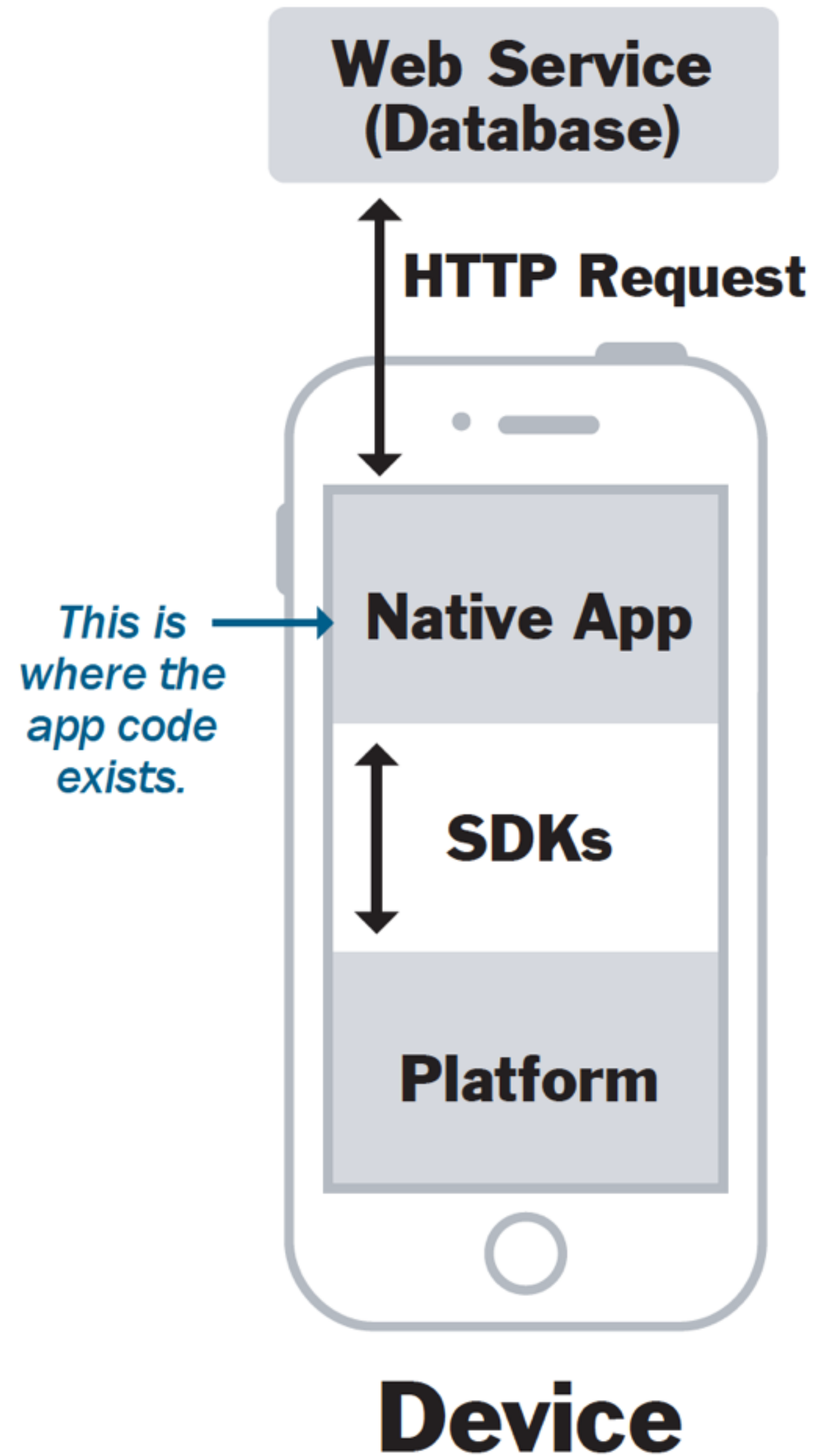
We asked more than 500 software developers and architects to tell us what causes them pain when they develop mobile applications. Multiple answer selections were allowed. The results are summarized below. The number next to each icon/label indicates the percent of respondents who indicated that they encounter this pain point during mobile development.



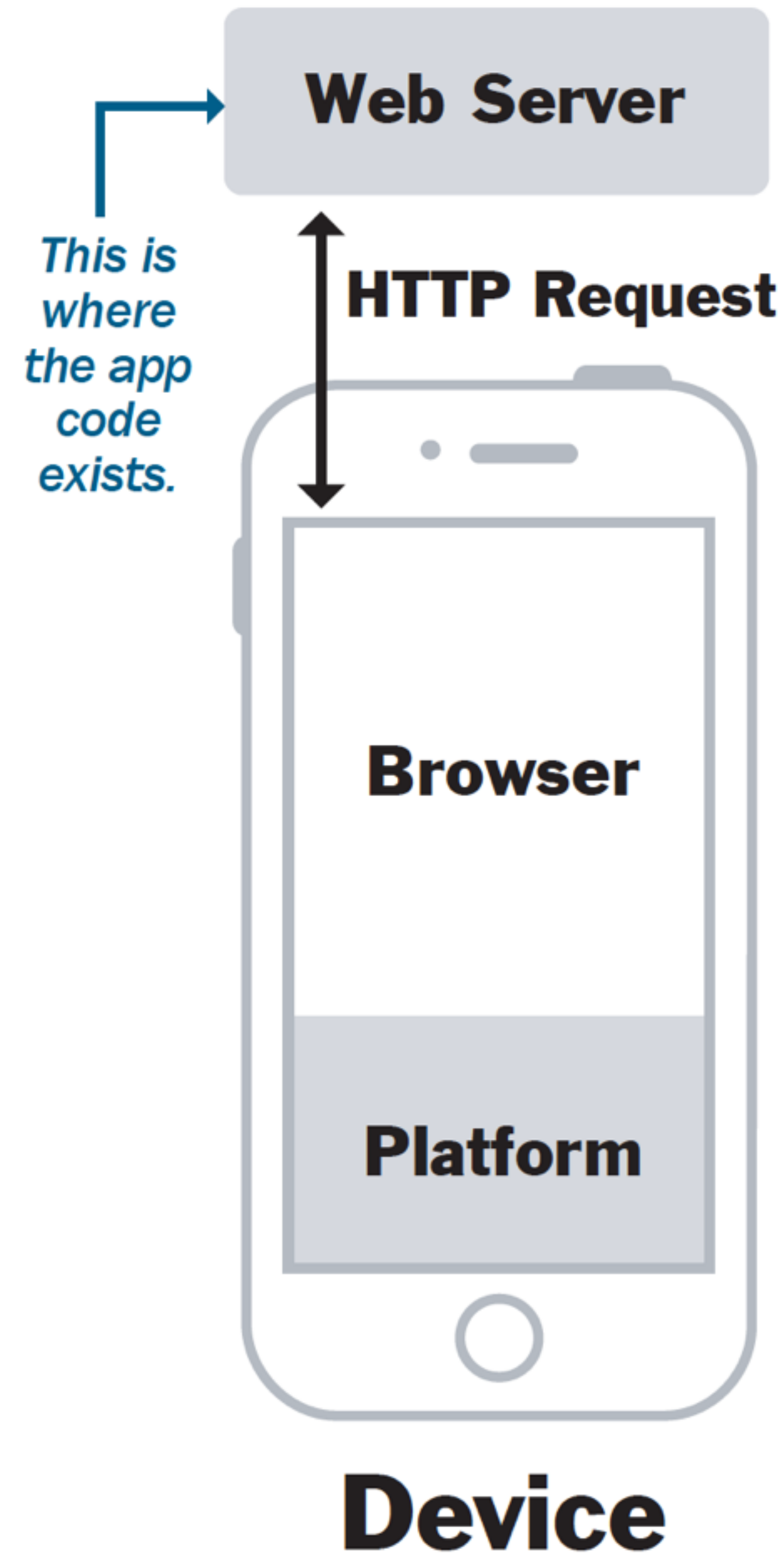
How are these Apps Build?

Several ways to build applications for mobile devices

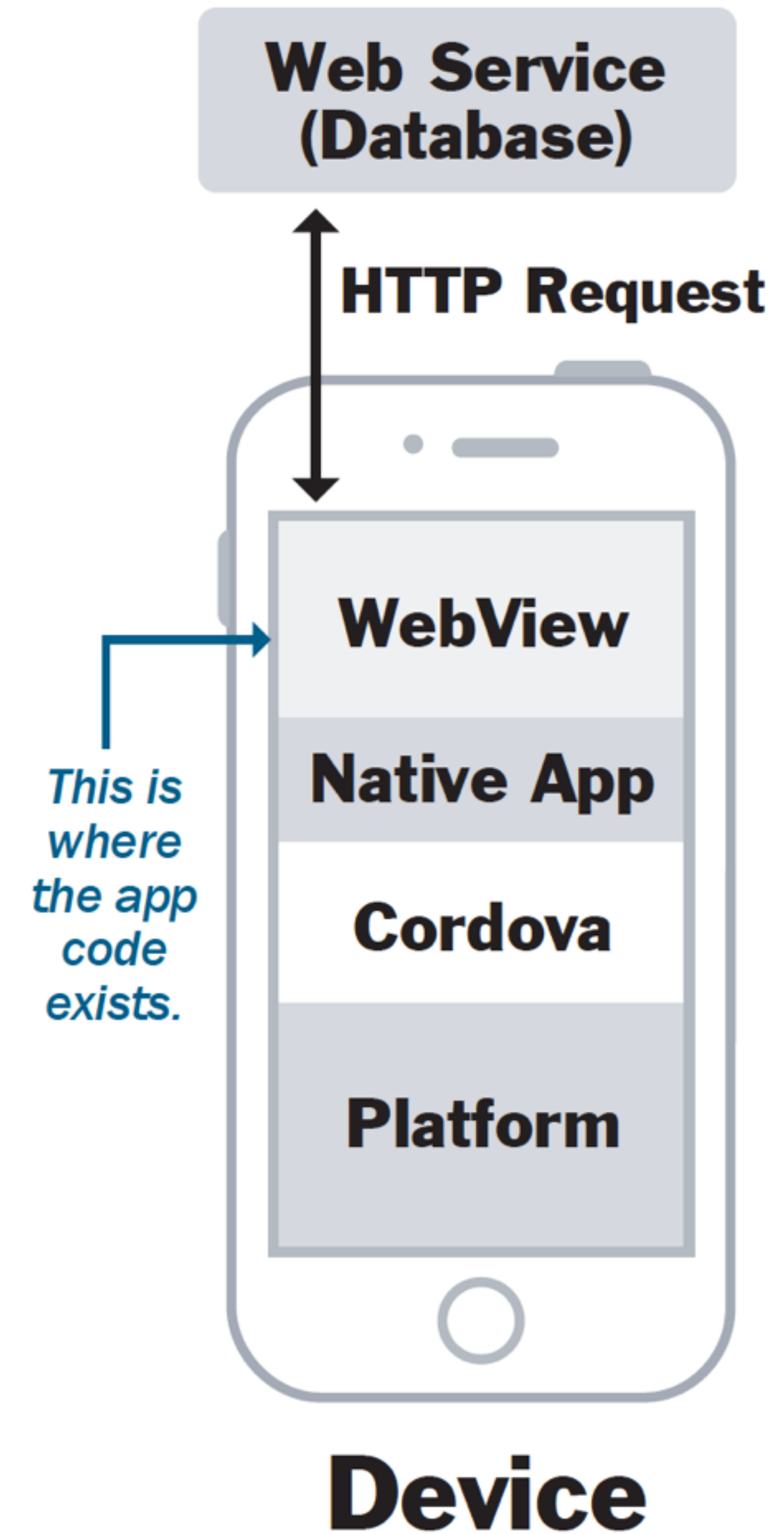
NATIVE APP



MOBILE WEBSITE



HYBRID APP



What are NATIVE mobile Apps?

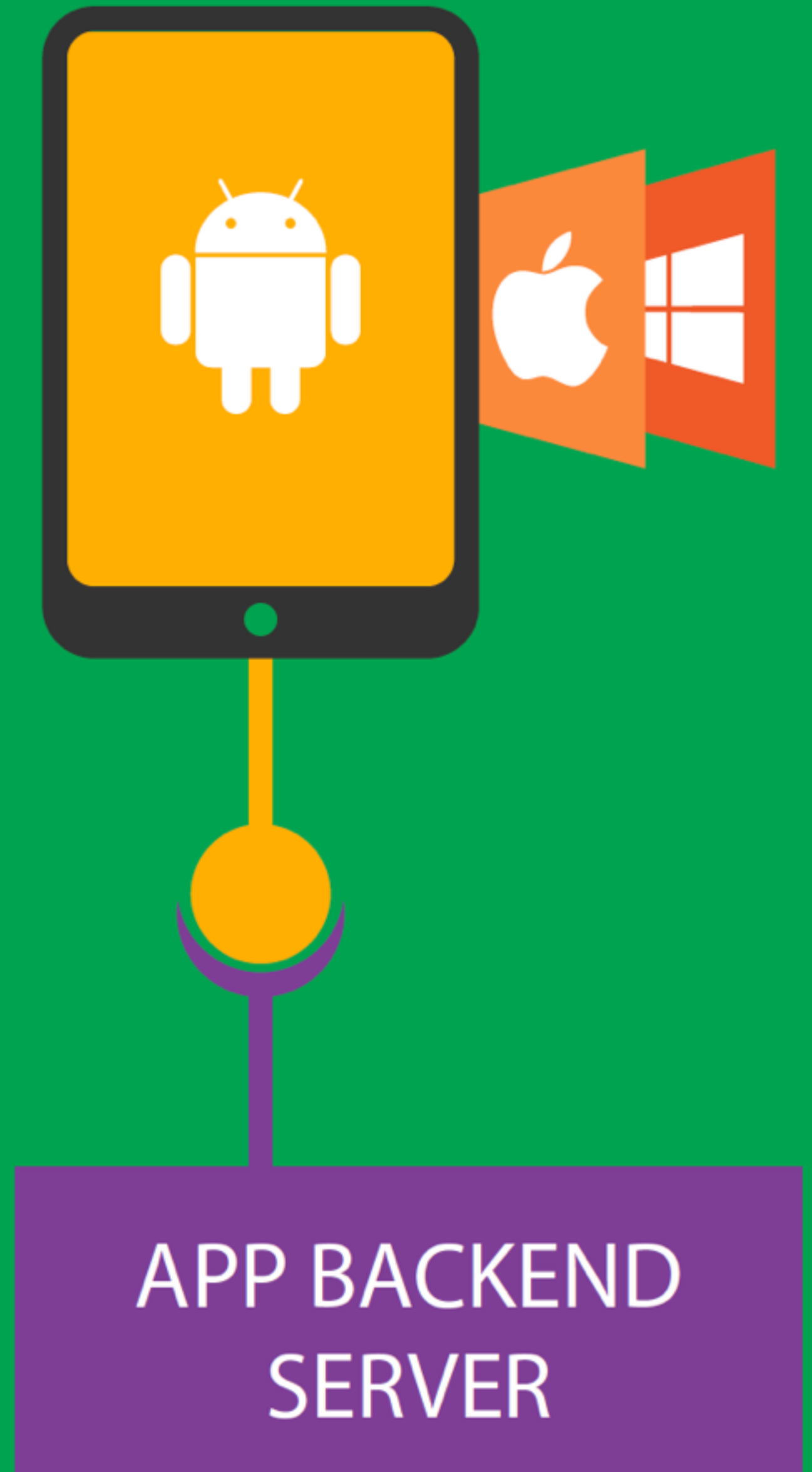
Native apps

- Run on a specific device and mobile operating system
- Built using native code (e.g. Objective-C or Swift)
- Downloaded from an app store
- Live on the device

EXCEPTION - ONLY IF REQUIRED

Bottom line

Build a native app when you want to deliver an extraordinary user experience at any cost. Games, like Angry Birds, are a good candidate for native apps.



Mobile Web Apps

Mobile web apps

Run on any device with a web browser

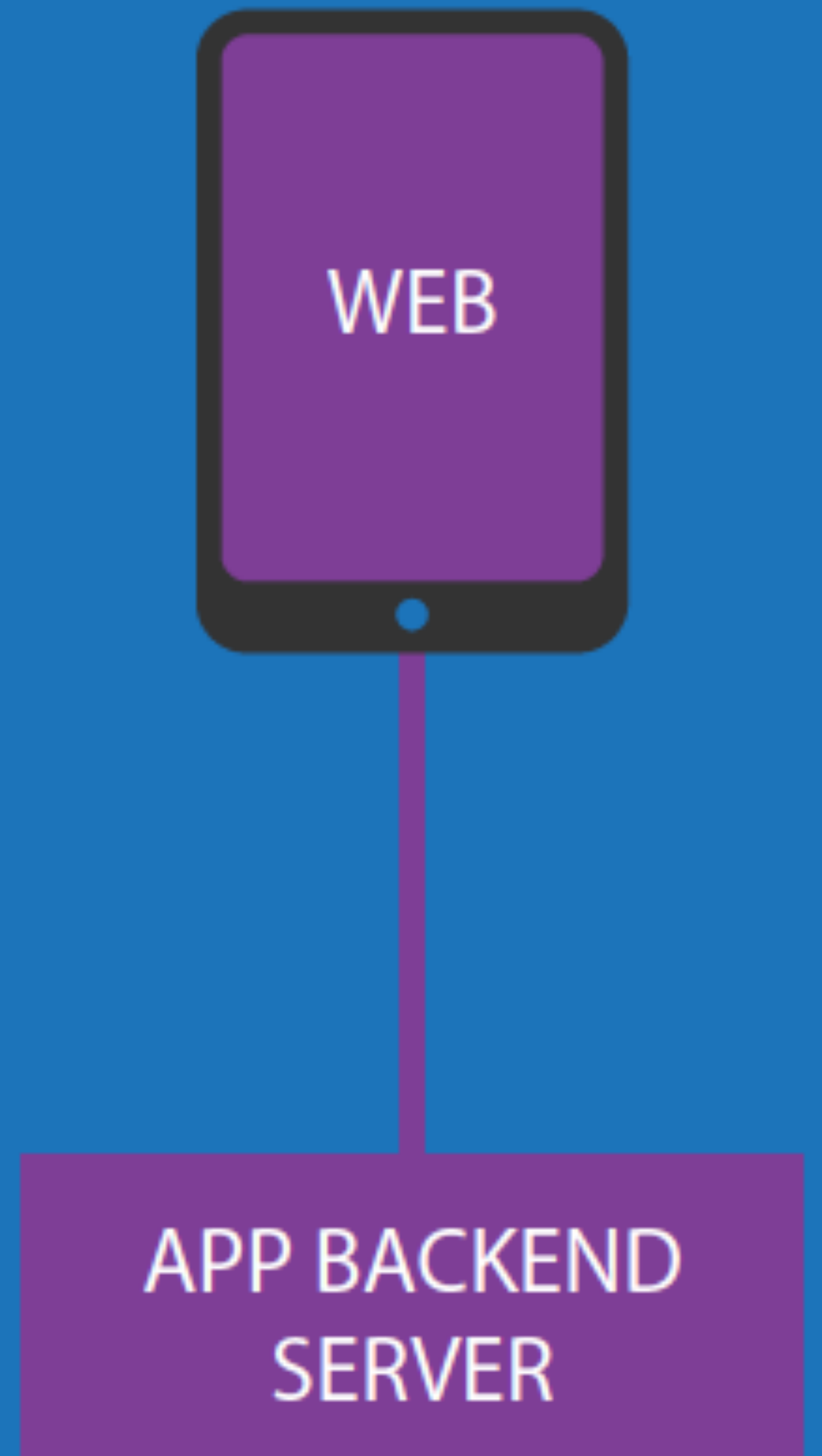
Built using standard web technologies

Served from an application server

DEFAULT - MOST COMMON CHOICE

Bottom line

Build a mobile web app when you want your application to be accessible from any device, without the need to install an application from an app store. An example of a mobile web application is the Financial Times web app.



Hybrid Apps

Hybrid Web Apps

Hybrid Mixed Apps

Packaged Hybrid Apps

- ▶ Run on multiple mobile operating systems (after fine-tuning for each operating system)
- ▶ Built using a cross-platform framework
- ▶ Downloaded from an app store
- ▶ Part lives on the device and part served from an application server

ALTERNATIVE - MOBILE WEB+

Hybrid - The Best of Both Worlds

The hybrid app approach is the fastest and most efficient way to deliver “real,” device-savvy mobile applications to users with frequency and low development cost and overhead. After several years of painful and costly missteps with pure native apps, the emergent hybrid approach has recently swelled in popularity and continues to gain momentum. The hybrid approach minimizes the amount of custom code required for each supported operating system, while still giving developers the ability to incorporate native features and functionality.

Additional options within the hybrid category allow organizations to cost effectively fulfill application requirements while optimizing in-house resources.

Hybrid Mixed

Hybrid Web

Packaged Hybrid

So, Which is the best approach?

NATIVE vs. WEB vs. HYBRID: 7 FACTORS OF COMPARISON

KEY

CON

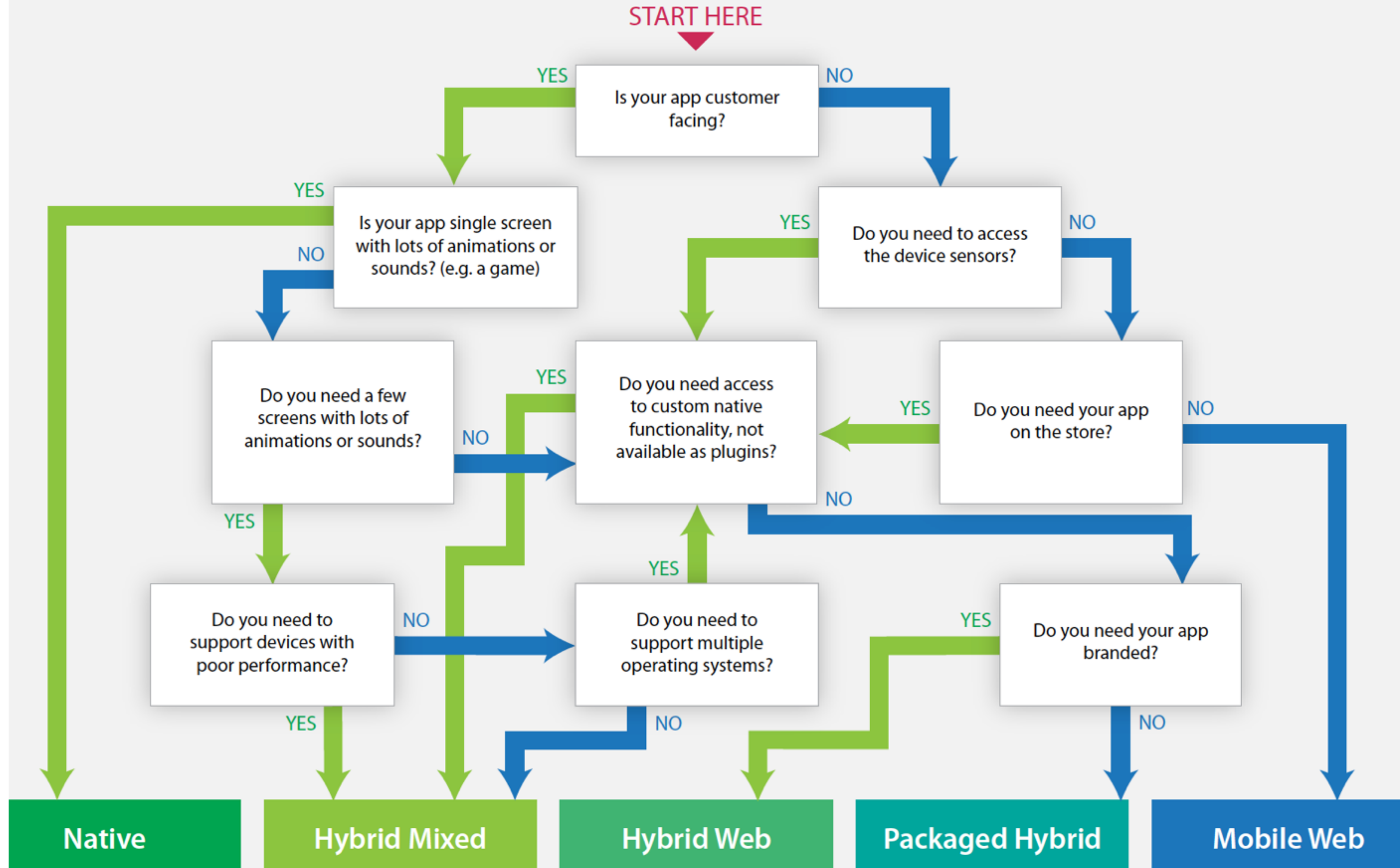
PRO

NEUTRAL

	NATIVE	HYBRID	WEB
COST	Commonly the highest of the three choices if developing for multiple platforms	Similar to pure web costs, but extra skills are required for hybrid tools	Lowest cost due to single codebase and common skillset
CODE REUSABILITY/ PORTABILITY	Code for one platform only works for that platform	Most hybrid tools will enable portability of a single codebase to the major mobile platforms	Browser compatibility and performance are the only concerns
DEVICE ACCESS	Platform SDK enables access to all device APIs	Many device APIs closed to web apps can be accessed, depending on the tool	Only a few device APIs like geolocation can be accessed, but the number is growing
UI CONSISTENCY	Platform comes with familiar, original UI components	UI frameworks can achieve a fairly native look	UI frameworks can achieve a fairly native look
DISTRIBUTION	App stores provide marketing benefits, but also have requirements and restrictions	App stores provide marketing benefits, but also have requirements and restrictions	No restrictions to launch, but there are no app store benefits
PERFORMANCE	Native code has direct access to platform functionality, resulting in better performance	For complex apps, the abstraction layers often prevent native-like performance	Performance is based on browser and network connection
MONETIZATION	More monetization opportunities, but stores take a percentage	More monetization opportunities, but stores take a percentage	No store commissions or setup costs, but there are few monetization methods

Decision Tree

NATIVE vs WEB vs HYBRID



NATIVE vs WEB vs HYBRID

Which **mobile architecture** is right for **your app**?



CHOOSE YOUR MOBILE DEVELOPMENT PATH



We know consumers spend more time using apps than mobile Web...



but, both hold intrinsic value — which is the right development path for you?

MOBILE WEB

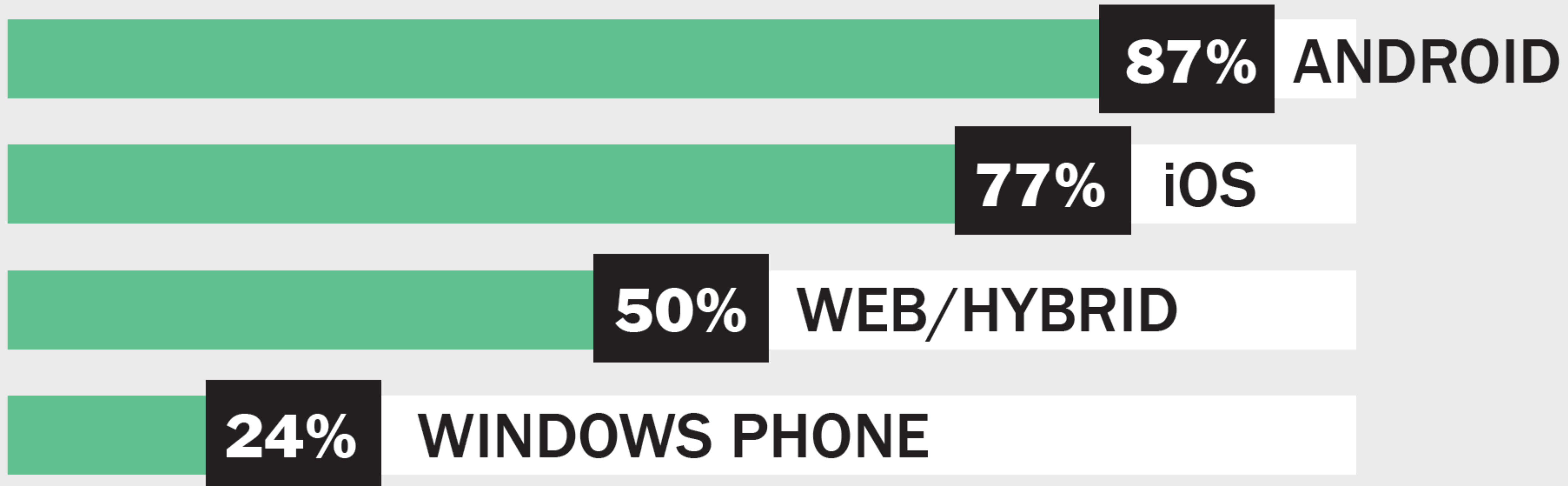


HYBRID NATIVE



PURE NATIVE

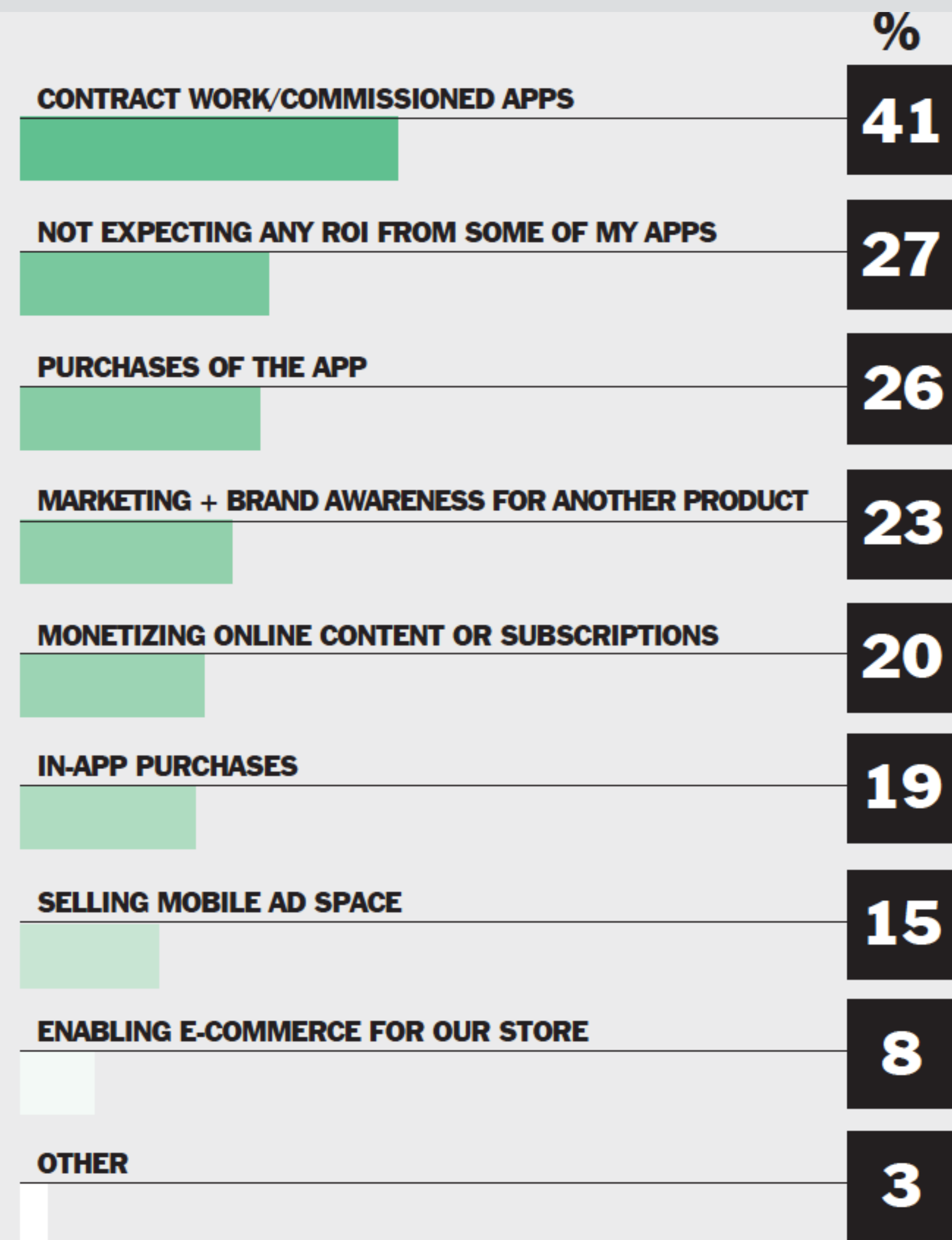




DZone 2015 Survey*

IBM

Revenue generation



What's from IBM?

- IBM MobileFirst Foundation - <https://mobilefirstplatform.ibmcloud.com>
- IBM Cloud (PaaS) - <https://Bluemix.net>

Thank YOU

Q&A

