PubNub

The Definitive Guide to Creating a Reliable Mobile App

The mobile app market is estimated to reach $57 billion by 2020. All but two of Y Combinator's Top 10 Companies list are either built around a mobile app or a service for building mobile apps. In a nutshell, products are going mobile.

But building mobile apps at scale isn't easy. There are a ton of pitfalls and risks that stand between you and setting your app apart from the rest. One key component of separating from the pack is reliability: delivering blazing fast experiences, which are completely secure at scale.

Ensuring performance and security is easy in the lab, but not in the wild. Reliability is your mobile app or product working as intended, no matter the network environment, device, operating system, or load of concurrent users. It's providing a persistent, high-speed and predictable experience for your end users. And it's protecting the valuable and sensitive data your app and users are streaming.

In this guide, you'll hear from Product Managers across a variety of industries on what they see as most important for ensuring mobile reliability; and gain a better understanding of what it takes to build scalable and secure mobile apps.

The Guide at a Glance

Migrate to Microservices
Break apart the mobile app and embrace interoperability.

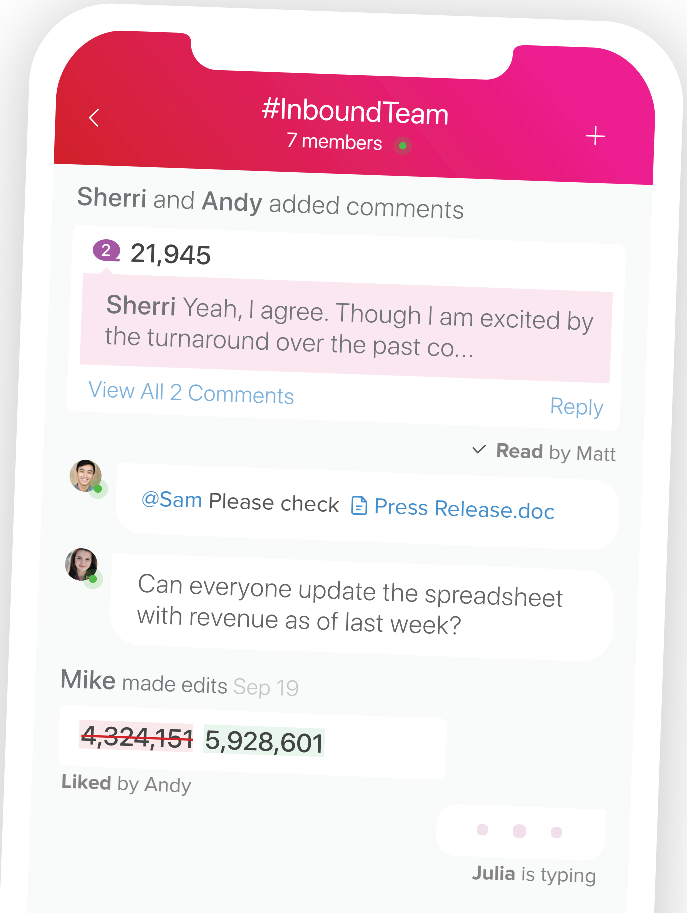
Reduce Bandwidth Consumption
And think about the worst of Internet conditions.

Reduce Battery Consumption
If your app is power hungry, you're bound for negative reviews. Or worse.

End-to-End Security
Assume all devices are insecure and apps can be compromised.

Global Redundancy
Bring your backend infrastructure as close to your end users as possible.

Move to the Edge
Go beyond the centralized data center.



2 21,945

Sherri Yeah, I agree. Though I am excited by the turnaround over the past co...

@Sam Please check Press Release.doc

Can everyone update the spreadsheet with revenue as of last week?

Mike made edits Sep 19

~~4,324,151~~ 5,928,601

1. Migrate to Microservices

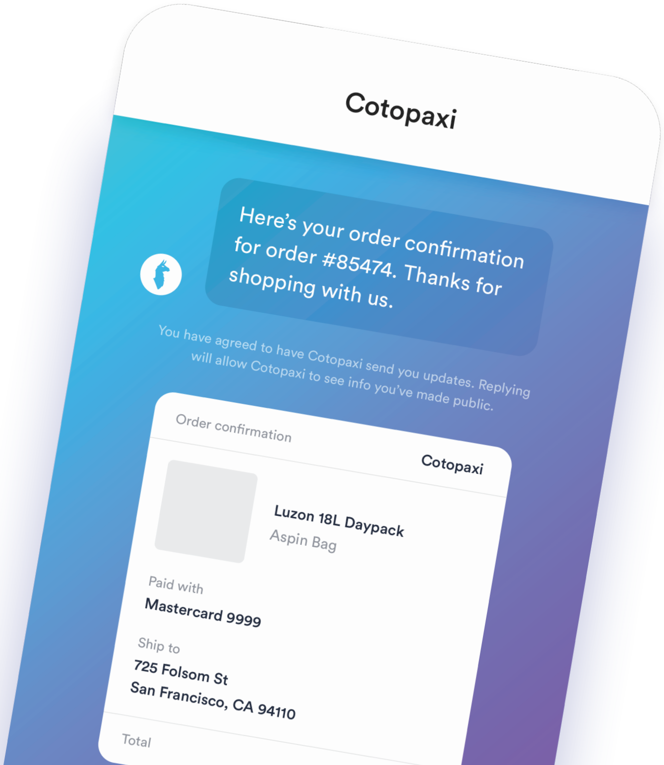
> *We no longer believe that one single point to access information or make changes is the right way to go. Instead, put the two to three most crucial features at users' fingertips.*
>
> *Behind the scenes, most technology companies have completed the transition to RESTful APIs and are embracing microservices, the next step toward breaking apart the mobile app and embracing interoperability of different systems. This process has allowed companies to build rich, segmented experiences, and to layer in additional information and data points from other services.*
>
> *Today it's common to use five to ten outside services (AKA microservices) during app construction, instead of wasting time reinventing the wheel.*

Gal Oppenheimer
Senior Product Manager @ Built.io

Microservices are a large part of the move from monolithic to distributed application design. Decoupled and only responsible for discrete tasks, microservices allow teams to build more scalable and extensible applications, more quickly.

From a reliability standpoint, a key benefit of microservices is that failure of a single service doesn't impact the other services. Fault isolation means your mobile app itself is more resilient. If, say, your push notifications service has an outage, the rest of your application will continue to operate normally, and because microservices themselves are individual components, you could easily fall back to a different push notification service until your primary one is back to being operational.

Microservices are also highly receptive to playing nicely with third-party services. In building mobile apps, developers often don't have time (or, in some cases, the expertise) to build every feature and function from scratch, and instead choose to integrate best-in-class services that are more reliable and trustworthy. Microservices let you quickly and confidently deploy challenging features like artificial intelligence, or bolster security with developer tools to build feature-rich, secure apps.



Cotopaxi

Here's your order confirmation for order #85474. Thanks for shopping with us.

You have agreed to have Cotopaxi send you updates. Replying will allow Cotopaxi to see info you've made public.

Order confirmation

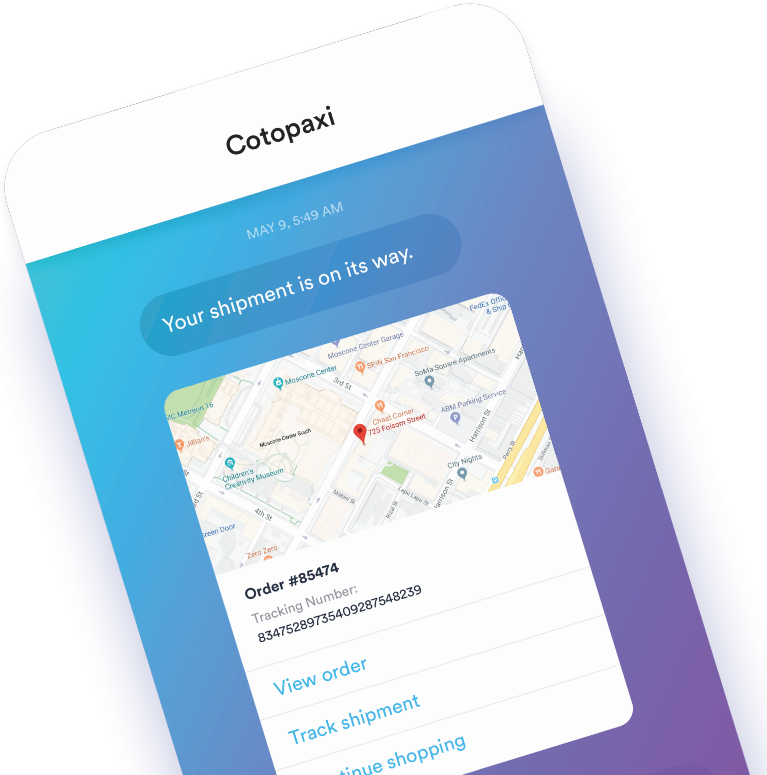
Cotopaxi

Luzon 18L Daypack
Aspin Bag

Paid with
Mastercard 9999

Ship to
725 Folsom St
San Francisco, CA 94110

Total



Cotopaxi

MAY 9, 5:49 AM

Your shipment is on its way.

Order #85474
Tracking Number:
83475289735409287548239

View order

Track shipment

tinue shopping

2. Reduce Bandwidth Consumption

> *It is important to consider the web scenes in those regions where your target audience lives, such as fluctuations in internet speed, which have a negative impact on user experience. Your client may not be able to connect to the internet all the time. So, when creating a program, you should be thinking about the worst of internet conditions.*
>
> *Think about power consumption and speed when choosing software and hardware protocols. Also create a caching, state management and data access mechanism that can be tuned to slow and intermittent web connections.*

Max Summers
Product Manager @ Magora Systems

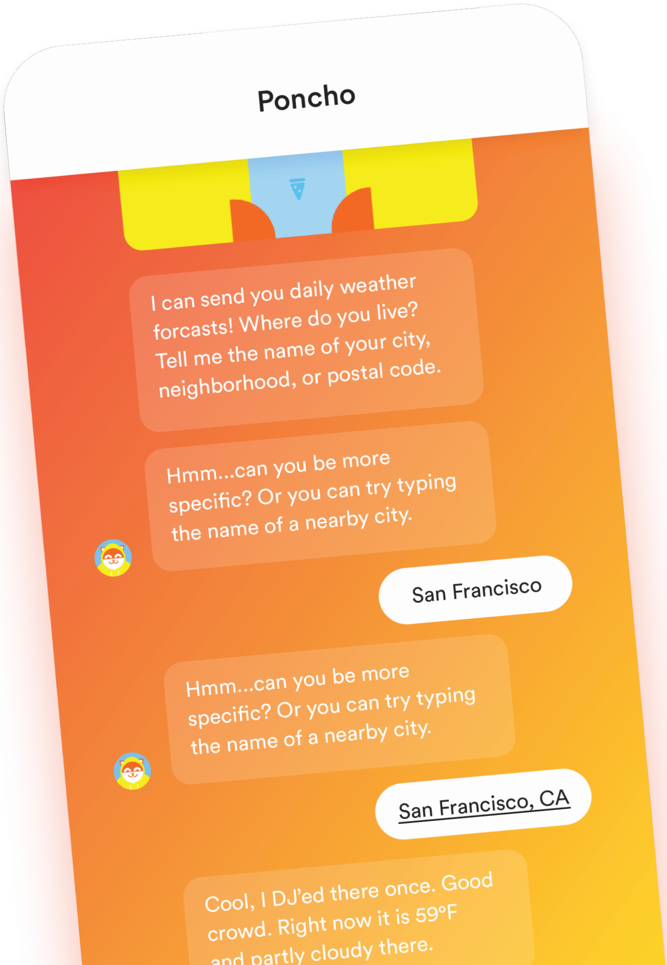
Never assume that your users will have a 4G or WiFi connection on their smartphone. Internet speeds fluctuate and networks are unreliable, so making development decisions with bandwidth consumption in mind is fundamental. If your application is built to perform only on the strongest internet connections, the app won't be reliable on anything worse.

One way to reduce bandwidth consumption is choosing the right frameworks and protocols for a mobile app. WebSock-ets, MQTT and HTTP long polling are all efficient realtime protocols that are built for low bandwidth consumption, largely due to their open socket bidirectional data streaming capabilities.

Traditional request/response models, by contrast, are constantly emitting messages that in most cases aren't necessary, eating up mobile bandwidth. More efficient realtime protocols only use bandwidth when they have to, delivering faster and more efficient mobile apps.

Another way to reduce bandwidth consumption is to ensure your mobile app has features to make the app more reliable in unreliable mobile environments, like sub-3G network areas, or even places with no Internet connection (like tunnels and high elevation).

Message catchup and message caching are two examples - both ensure that messages reach their intended recipient after they reconnect, guaranteeing that data gets where it needs to go.



Poncho

I can send you daily weather forcasts! Where do you live? Tell me the name of your city, neighborhood, or postal code.

Hmm...can you be more specific? Or you can try typing the name of a nearby city.

San Francisco

Hmm...can you be more specific? Or you can try typing the name of a nearby city.

San Francisco, CA

Cool, I DJ'ed there once. Good crowd. Right now it is 59°F and partly cloudy there.

3. Reduce Battery Consumption

> *While smartphones offer a tremendous opportunity to app developers, the constrained resources of mobile also pose certain challenges for developers. One of these challenges is power consumption.*
>
> *If your app is power-hungry, you are bound to get negative reviews from your customers. In some cases, they may end up deleting your app altogether. Hence, traditionally concerned with performance and memory usage optimization, app developers are increasingly looking to address the challenges of optimizing energy consumption.*

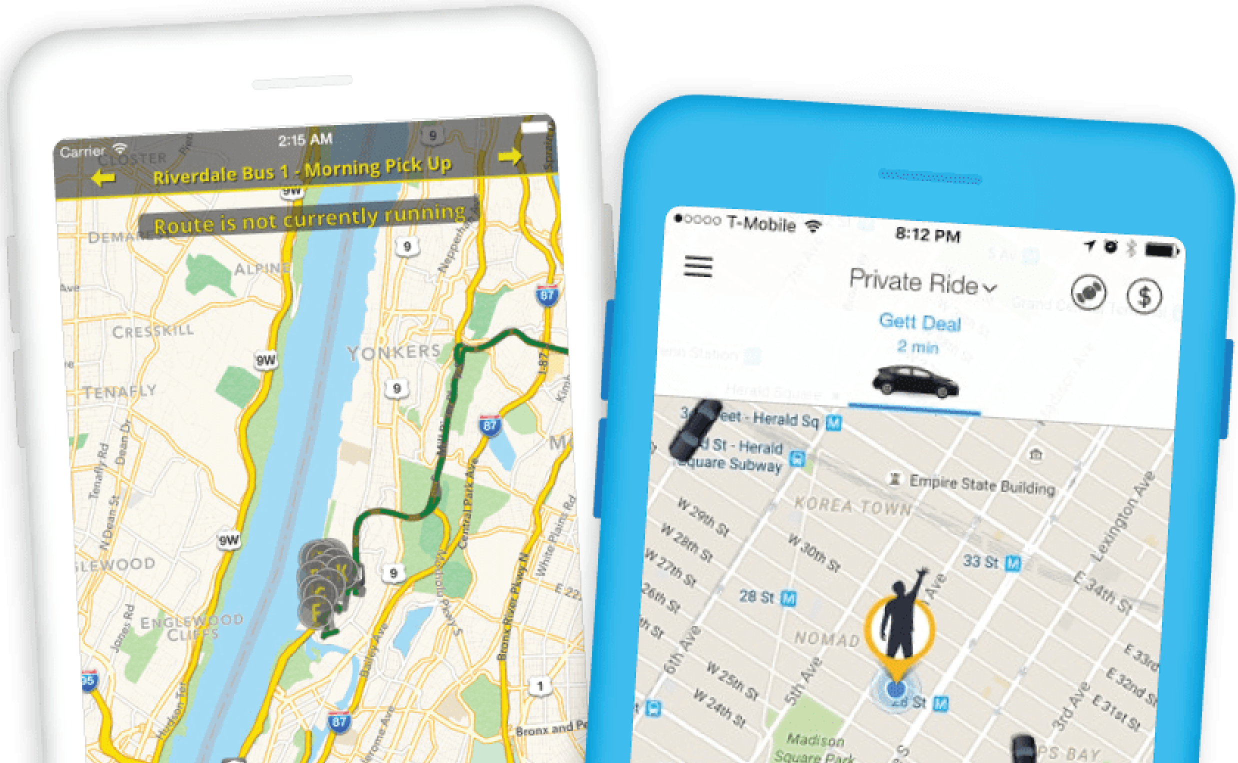
Rajat Harlalka
Product Manager @ Bellurbis Technologies

How can users rely on a mobile app if their phone is out of juice? A good battery consumption management strategy demands that mobile development teams understand where and how energy is used. Some common battery drainers include unnecessary background activity, inefficient refreshing and location-heavy apps.

There are two ways that you can better manage power consumption: make your apps lazy first and take advantage of platform and infrastructure features to manage battery usage.

Lazy first is an approach to app development in which you're looking for ways to reduce and optimize operations that are battery intensive, and doing everything you can to reduce, defer, or coalesce operations and functions in the app. Some examples include cutting out redundant operations, performing actions when efficient (not in the moment), and batching streams of data.

When choosing any infrastructure or platform to power your mobile app, keep battery consumption in mind and see what features your service provider has to reduce it. Does the platform utilize efficient data transfer protocols? Does it include things like message caching? Does it allow you to use a microservices-oriented architecture? How that platform provider architects their product determines how well your product will perform.



4. End-to-End Security

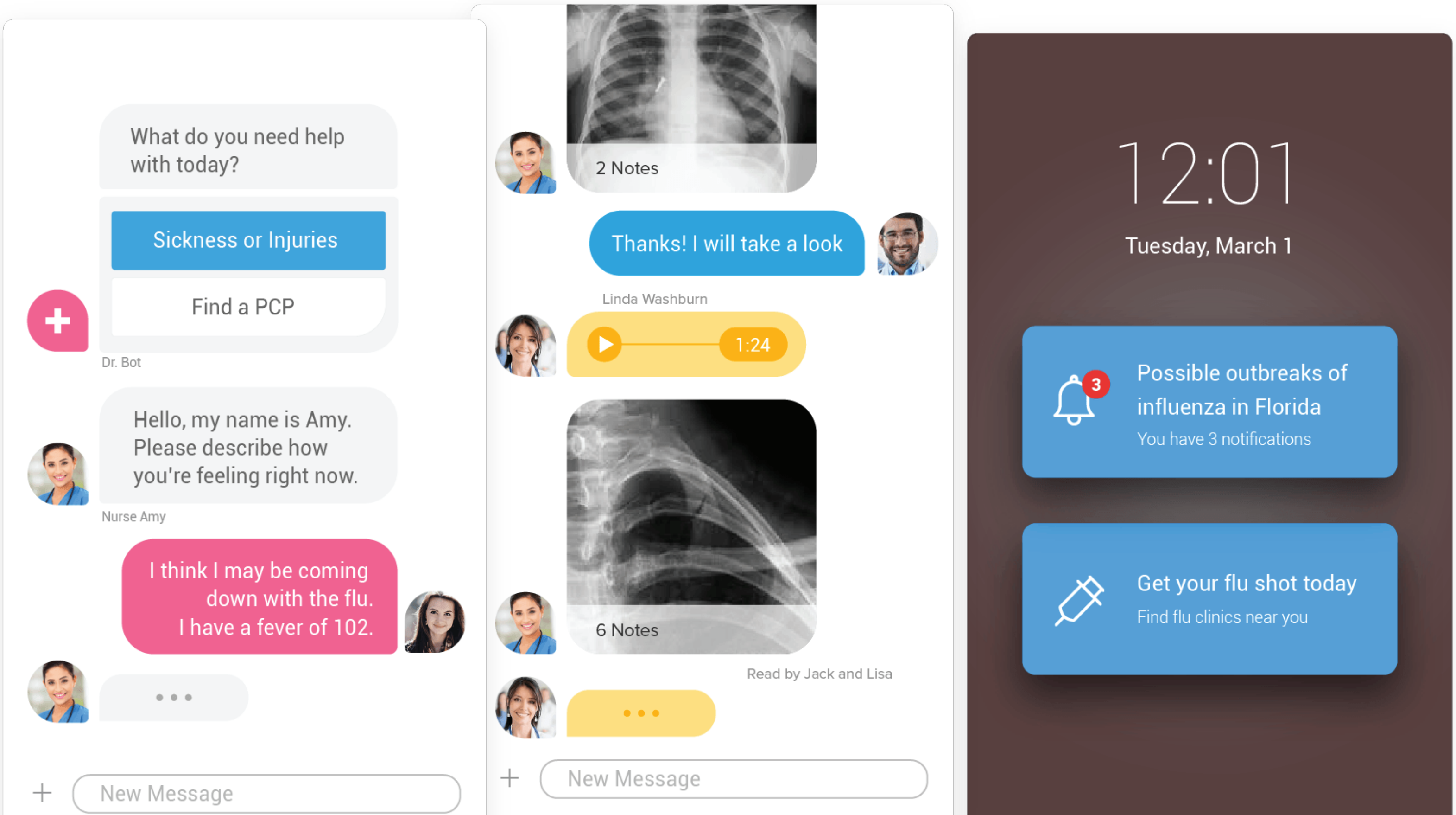
> *When it comes to addressing application security, straight out assume that all mobile devices are insecure, all of our applications can be compromised, and people can capture the data moving to and from our apps. We're not paranoid. We just make these central security assumptions for all of our mobile apps. I think of it as insurance in a mobile world.*
>
> *Given those assumptions, we're always looking for new ways to harden the security of our mobile apps against the most common security failures, and you should too. To do this, we focus on five key areas - the mobile device, the application, authentication, development, and data at rest and in transit.*

Matthew David
Innovation Product Manager @ QBE North America

There's exhaustive list after exhaustive list of security considerations for mobile app development, so we won't try to cover it all here. In this guide, we'll focus specifically on securing data in transit.

Data in transit is any message flowing over a realtime data stream. This could be a chat message, an IoT reading, a financial stock price, or a push notification. There are a couple of 'table stakes' security requirements for data in transit: AES/SSL/TLS encryption on every message must be included in any mobile app, big or small.

In addition to encryption, fine-grained access control absolutely must be provided, with a special focus on token-based authorization systems. These allow you to segment permissions by individual data channel or even down to the individual device. With full control over who can read and who can write data in your mobile app, you greatly improve your ability to prevent unauthorized access to the application.



What do you need help with today?

Dr. Bot

Hello, my name is Amy. Please describe how you're feeling right now.

Nurse Amy

I think I may be coming down with the flu. I have a fever of 102.

Thanks! I will take a look

Linda Washburn

Read by Jack and Lisa

12:01
Tuesday, March 1

Possible outbreaks of influenza in Florida
You have 3 notifications

Get your flu shot today
Find flu clinics near you

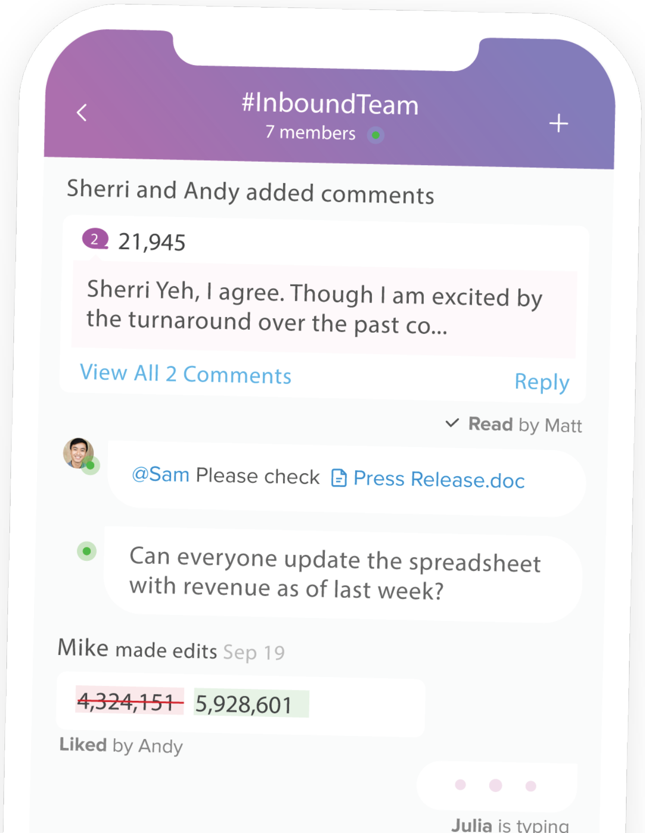
5. Global Redundancy

> *As usage of applications and websites scaled into the millions (or billions) of users over the past several years, previous eras of application architecture began to break down under the sheer volume of traffic and user data. More recently, a desire to make greater use of incredible amounts of available digital data has sparked interest in an entirely new class of technologies aimed at that end alone.*
>
> *The solution to these problems has often involved the creation of new storage, processing and database frameworks designed to easily scale across clusters of servers, and to simplify the process and speed of moving information between the pieces.*

Derrick Harris
Product Marketing @ Pivotal Software

To deliver mobile apps at scale, you need global coverage. Multiple points-of-presence ensure that mobile users are connected to the closest data center based on their geolocation, meaning users in Tokyo get the same seamless mobile experience as users in Sao Paulo. Meanwhile, a sophisticated orchestration system underlying the distributed network will be required to ensure constant, consistent, and persistent replication of data.

To deliver a reliable mobile app experience, you need global redundancy - in other words, you need multiple data centers that back up each piece of data and send it if messages drop. From a mobile app development standpoint, this is a core component of ease at scale - infrastructure reliability staying stable as your user base grows.



Sherri and Andy added comments

2 21,945

Sherri Yeh, I agree. Though I am excited by the turnaround over the past co...

✓ **Read** by Matt

@Sam Please check Press Release.doc

Can everyone update the spreadsheet with revenue as of last week?

Mike made edits Sep 19

~~4,324,151~~ 5,928,601

Liked by Andy

Julia is typing

6. Moving to the Edge

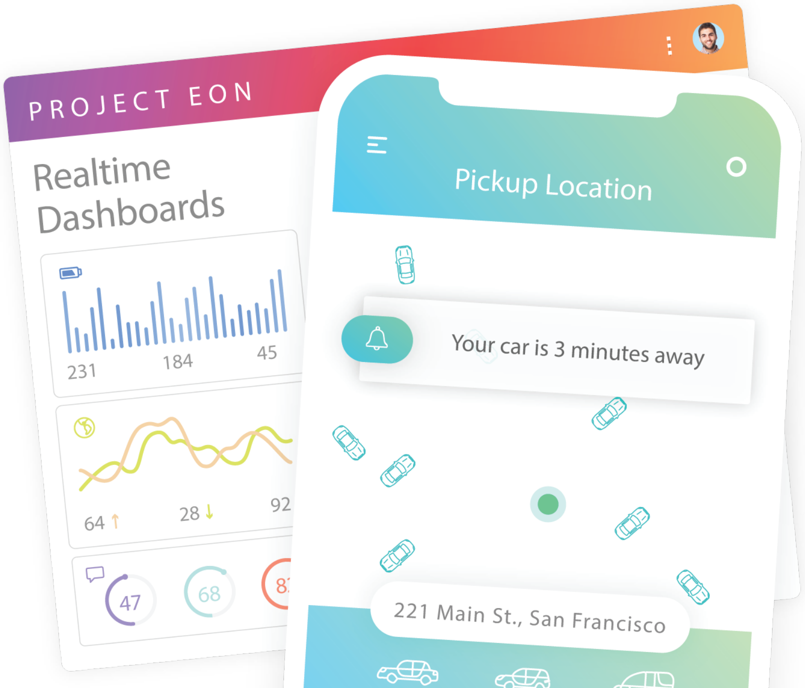
> *There are two mega-trends driving the need for new computing infrastructure beyond the centralized datacenter, commonly referred to as “edge computing”.*
>
> *The first is the explosion of data stream-based applications – applications and experiences relying on huge amounts of data points being sent and received by devices. This broad category consists of everything from mobile games like Pokemon Go to IoT sensors, and everything in between.*
>
> *The second mega-trend is the expectation of realtime experiences from end users. Users expect to see realtime updates on their screens, mirroring what’s happening in the real world. Like a Hollywood sci-fi movie, we now expect to control devices instantly around the world, and see our screens flash when something happens on another continent.*

Todd Greene
CEO @ PubNub

Edge computing will play a huge role in ensuring mobile reliability and will continue to replace traditional mobile app infrastructures (most commonly request/response architecture), that we’ve relied heavily on for the last decade. This architecture has worked well for applications up until now, but modern users have come to expect more interactivity, more speed, and more data in their applications.

Request/response architecture are both too slow and too resource-intensive to meet the needs of low-latency apps that are globally scaled, with devices and end users constantly emitting and consuming large streams of data.

Edge computing promises to solve many of these challenges, with the ability for business logic to be executed closer to the devices, that is, closer to the “edge” of the network. In this way, a significant share of the computation that otherwise would happen on a central server can instead take place on the phone itself. This not only increases the speed of computation but also requires less bandwidth consumption, creating more reliable mobile apps at scale.



PubNub

PubNub operates the world's leading Data Stream Network, which connects, delivers, and controls the data and logic used to power realtime applications at global scale for thousands of companies around the world including Peloton, Atlassian, and athenahealth.

Combining over 70 SDKs, a global network, and blazing fast serverless edge computing, PubNub delivers twice as many messages each day as Twitter, Snapchat, and the global SMS network combined. Founded in 2010, PubNub is a Silicon Valley technology company backed by leading investors including Relay Ventures, Sapphire Ventures, Scale Venture Partners, Cisco Investments, and Ericsson.

www.pubnub.com

(415) 223-7552

@pubnub

Contact Us