



---

## 5 LITTLE KNOWN REASONS YOUR APP IS UNDERPERFORMING



# Contents

---

<b>Introduction</b>	<b>1</b>
<b>Blockers of Success</b>	<b>2</b>
The App Stores Make Everything Slow	
You Can't Rollback or Fix Mistakes Quickly	
Underlying Web Assumptions	
Following Others' Best Practices	
Learning is Expensive	
<b>How Top Apps Succeed Anyway</b>	<b>8</b>
Test All Their Assumptions	
Build Small and Build Iteratively	
Invest in Learning and Testing Infrastructure	
<b>Applying These Methods to Your App</b>	<b>14</b>

# Introduction

---

If you don't have a strong mobile strategy, you're creating a huge opportunity for competitors to swoop in and take your customers. Almost 50% of traffic now comes through mobile, making it a tremendous opportunity to generate deeper customer insights and engage and retain customers. If your app isn't connecting with users, they're going to start looking to competitors for other solutions that better serve their mobile needs.

Most companies have made strides in the past few years to create better mobile experiences for their customers. However, while 87% of C-Level executives say they have a formal mobile strategy, "only 2 of 5 companies have made good progress on their mobility efforts." So why is it that so many mobile strategies have produced only mediocre results? In this paper, we're going to dig into the major blockers for mobile success, how the top apps have overcome those blockers, and how you can do the same.

# Blockers of Success

---

“Releasing on mobile is like reverting back to the days of shipping boxed software.”

- Product development has come a long way in the last 10 years, so why is it
- that on mobile it’s like we’ve reverted back to the days of shipping boxed
- software? On web, we’ve developed agile methodologies to be nimble,
- adapt to changing markets, and focus on users. There, we use learnings
- from incremental improvements to push and drive the product roadmap,
- with great success.
- 
- Yet somehow on mobile, we’ve reverted back to antiquated waterfall
- methodologies of designing, building, and releasing features over the
- course of months. Even after release, we cross our fingers and hope
- things turn out well. **There’s a huge investment of time and money**
- **early on, without ever having tested or validated the changes.**
- 
- That’s a big gamble to take, especially when we know there’s a better
- way. When you look at the way mobile is set up, it quickly becomes clear
- why: the mobile ecosystem wasn’t designed to encourage modern day
- development techniques.

# 1. The App Stores Make Everything Slow

**Long App Store reviews are one of the biggest blockers to mobile success.** The problem is the way distribution is set up. With an average of 9 days to complete a review process and get your new version in the hands of consumers, teams can't be agile in the same way they've learned to do on web. That makes it difficult to pivot or adapt to an ever changing market. Any change, no matter how large or small, takes an inordinate amount of time to process and deploy to users.

- “In mobile what do I do? I push a
- button, it goes to The App Store,
- the black hole that is Apple, and
- out comes, in some indeterminate
- amount of time, my binary. If I've
- made a mistake or if there's a fatal, or
- something silly in that app, it's gone.
- That bullet has left the barrel, and
- I'm screwed.”
- 

## **Chuck Rossi**

Release Engineering  
Director at Facebook



## 2. You Can't Rollback or Fix Mistakes Quickly

**App stores don't let you rollback to a previous state.** Instead, you have to deploy a new version (even if it's the same as the previous one). This means you have to be much more careful when deploying a new release. It makes engineers scared of making any mistake and decreases confidence.



- “The less you release, the less confident you are as a company
- that whatever you’re pushing out is not going to crash and
- you’re not going to be able to fix a major bug that affects all
- your users.”

**David Dryjanski**  
Product Manager at Lyft

### 3. Underlying Web Assumptions

Another common pitfall is that companies fail to realize the key differences between mobile and web user goals. Each of these user groups are seeking to accomplish different tasks, yet many teams have underlying assumptions that users are looking for similar experiences in both. Web and mobile are totally different platforms, and should be treated as such.

Mobile users are fundamentally different in two ways:

- 1. They're highly task oriented**
- 2. They're easily disengaged**

Using a web strategy to serve mobile customers just isn't going to work. On web, users are much more open to exploring and learning about a brand. On mobile, they're trying to accomplish a task quickly and easily get frustrated when they're inundated with unrelated options. Additionally, any small distraction in a user's environment can derail the entire process. You have to test and research to learn what it is that your customers actually want and expect from your app, instead of cramming all your web features onto a mobile app.

## 4. Following Others' Best Practices

"We strongly believe that there is no such thing as the perfect solution that you can learn from somewhere else...You can't just go in and say this hey this worked somewhere else it must work here," says Sergei Sorokin of Yahoo!'s growth team. Truly serving your users comes from developing a deep understanding of them. **Assuming you know what works best for your users without testing is just poor practice.**



**Luke Wroblewski**  
@lukew

[Follow](#)

"best practice audits" where you collect how other products do things like on-boarding, checkout, etc.

are hazardous to your design [1/3]

2:28 PM - 5 Aug 2015

↩️ ↻ 70 ★ 75



**Luke Wroblewski**  
@lukew

[Follow](#)

rather than coming to conclusions through iteration use & testing you assume others did & got to a best practice. usually they didn't [2/3]

2:29 PM - 5 Aug 2015

↩️ ↻ 62 ★ 60



**Luke Wroblewski**  
@lukew

[Follow](#)

and once you see solutions, it's very hard to un-see them so they end up strongly influencing your design. [3/3]

2:31 PM - 5 Aug 2015

↩️ ↻ 68 ★ 86

## 5. Learning Is Expensive

Current mobile development methods **force you to wait weeks or months for feedback and data.** Most teams today release every 1-2 months, too slowly to improve and learn quickly. They spend lots of time and effort spec'ing out a feature, building it, waiting for it to release, and then learning from the results. By the time you see the analytics, you'll already have invested a lot of time and money into something before you even know how it'll affect user behavior.

**While you're trying to determine the impact of a mobile release, those changes keep affecting your users.** In the app stores, bad reviews can cost you future downloads. You could also negatively impact engagement and retention if there's a bad change. As mentioned above, fixing those changes requires another slow app review while you quickly bleed users.

**Engineering is also a limited resource that you need to allocate accordingly.** Wasting it on features or changes that nobody cares about is going to waste both time and money. You want to spend your time on things that give you the best ROI and best serve your customers. Without data to get insights about your customers, it's hard to know where to allocate engineering resources most effectively.

# How Top Apps Succeed Anyway

---

- The top apps rise above the competition because they've created
- entirely new strategies for mobile success. Instead of assuming things
- work similarly on web and mobile, they have a fundamentally different
- approach for serving their mobile users' needs. Additionally, they invest
- in the infrastructure to build small, iterative changes that maximize their
- learnings. These keys allow them to ensure that they're providing users
- with the best possible experience, even within the constraints of mobile.

# 1. They Test All Their Assumptions

**In hindsight, everything seems obvious.** “The things that ended up working were the things that in retrospect were like ‘oh yeah well that’s obvious. If I were the person receiving this I would totally respond better. But when you’re in the mode and when you’re coming at it from the other side it’s really hard to think about,” says Yahoo’s Sorokin. **Lots of times we’ll think we have a great idea that’ll just work, most of the time it doesn’t.**

- “It’s not just good enough for
- somebody to come up and say ‘I have
- this perfect idea. It has to work.’ Well
- guess what, we tested 122 of them
- and most of them did not.”

**Sergei Sorokin**

Product Manager at Yahoo! Growth



**There is no perfect solution that you can learn from others.** Each app is different, so figuring out how to best serve your users is key. However, **traditional methods of interviewing end users isn’t an effective way to learn.** According to the Research VP at Gartner, “Users find it challenging to effectively describe what a mobile app needs to do.” This means that it’s unlikely you’ll gain valuable feedback just by talking to or surveying your users.

That’s why testing is so incredibly vital to product development. **The top apps are constantly running small tests on their users to figure out what works best for their unique apps.** This helps them see how users actually behave when presented with a new change, rather than just taking their word on it. They do this by creating small iterative changes, then validating them through A/B testing.

## 2. They Build Small and Build Iteratively



- “Smart people really like to over complicate the problem
- because you can see four steps ahead, so it’s very tempting
- to say like, ‘Hmm let me go build that final version.’ But when
- you do that...you don’t get something that actually increases
- your users metrics.”

**Waseem Daher**  
Product Manager at Dropbox

In a typical large release, it’s hard to tell exactly what changes are affecting user behavior and how due to a multitude of confounding factors. Teams have a tendency to build out full features without really understanding how each component is going to affect and drive user behavior.

The top teams don’t invest unnecessary time and effort into untested changes. **Instead, they make simple versions (minimum viable products or MVPs) of a feature and then test them early on to see their effects.**

**Making small iterative changes and A/B testing them will reveal a direct cause and effect relationship** between your change and user behavior, something that’s incredibly difficult to see in a large release. This allows you to validate your ideas by seeing whether users respond positively or negatively, as well as help avoid assumptions. Just keep the scope simple. However, doing so can sometimes be difficult given the constraints on mobile. That’s why teams are investing in the infrastructure to support this methodology.

### 3. They Invest in Learning and Testing Infrastructure

Tightening up the release cycle is critical if you want to build iteratively and use testing to improve your app. We've found that **the top apps iterate 10x faster than the competition**. Doing so allows them to **learn more quickly and be more flexible in perpetually changing market**. Instead of spending months developing a large release, teams can make incremental and compounding changes that build off of the learnings they get from each short release. These teams have made it a priority to tighten their release cycles so that they're releasing ever 1-2 weeks instead of months.

- "If you put your features out there sooner to
- other users, you'll learn faster, and you'll iterate
- more quickly. And product and engineering
- now becomes the company leader in getting
- things done."

**David Dryjanski**  
Product Manager at Lyft

#### **Top teams also invest in feature flagging to minimize risk.**

"You gotta build your features behind feature flags. This allows you to minimize risk when releasing a feature. You can do either do a gradual rollout or turn it on to 1% of your users and scale up," says Dryjanski. Utilizing feature flagging to do gradual (staged) rollouts help ensures everything is working as it should as you scale up and identify potential problems before they become widespread. Feature flagging also allows you to instantly rollback if something goes wrong, something that's not possible to do with the app stores. Combined, these benefits allow you and your team to push forward with confidence that you can deal with any situation swiftly.

# Applying These Methods to Your App

---

Mobile and web differ in quite a few significant ways, so it's crucial for us to figure out how to overcome the challenges presented. Simply porting over a mobile strategy just doesn't work. Teams have to recognize the obstacles and design their workflow to account for those differences. If you're serious about mobile and utilizing to its fullest potential, use the same methodology that top apps use to get the best results.

**Get data and insights about your customers rather than making assumptions.** Release small, iterative changes at a faster rate and A/B test them to figure out what's working and what isn't. Once you know that, you can further flesh out ideas to build products and features that delight users. Using data collected from your customers will give you far better results and ROI than traditional waterfall methods.

Create the Infrastructure to support agility. “ Make sure you have the right set of tools that allow you to operate at this speed,” says Lyft’s Dryjanski. While building out these tools yourself is possible, most teams don’t have the near limitless engineering resources of Lyft, Facebook, or Dropbox to do so. Using a solution such as Apptimize will give you the tools you need for mobile success, without a huge investment and specialized knowledge. Doing so will make A/B testing, releasing quickly, and feature flagging a breeze so you can focus on what you do best: serving your users.

Apptimize empowers teams learn faster and create better apps. We provide you with everything you need to iterate intelligently and find success in the mobile space.

**To try Apptimize for free on your app,  
sign up today at [www.apptimize.com](http://www.apptimize.com).**