

Moving Quickly Inside a Larger Organization

Pablo Jablonski

VP Engineering, UnitedMasters

Moving Quickly Inside a Larger Organization

Pablo Jablonski

VP Engineering, UnitedMasters

A little about me

- I truly believe you can't build the right technology if you don't first really understand the customer problem you are solving
- For the past 18 years I've been building tech to solve problems, from very small startups to large companies
- I joined Twitter in 2015 working on Periscope
- I worked on multiple 0 to 1 projects at Twitter, and became very acquainted with the many things that caused Twitter to move slowly

A little about me

- I believe you can't build the right technology if you don't first really understand the customer problem you are solving
- For the past 18 years I've been building tech to solve problems, from very small startups to large companies
- I joined Twitter in 2015 working on Periscope
- I worked on multiple 0 to 1 projects at Twitter, and became very acquainted with the many things that caused Twitter to move slowly

Context for today's talk

- In mid 2020, I was tasked with building a team to take a social audio prototype and launch it as quickly as possible - Spaces
- We started building the team in November, and launched our first beta on iOS in December.
- With a team of about 40 engineers, we GA'ed the product across all platforms in the beginning of May
- This talk will cover some of the key learnings about how we were able to move quickly, and how it may be useful to you

Where do we start?

Answer the *why*

Getting to the *why*

- You need to understand your customer, and you need to understand (and define!) the problem you are solving
- How is solving this problem going to materially help the business achieve its goals?

Getting to the *why*

- You need to understand your customer, and you need to understand (and define!) the problem you are solving
- How is solving this problem going to materially help the business achieve its goals?
- Why is it important to do *now*?
- Create real (and ambitious) measurable goals that measure if you are making the right progress towards solving that problem

Build the team

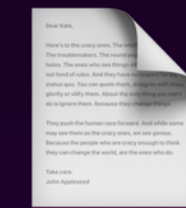
Building the team

- Find people who are passionate about the problem you are solving, and are excited about the work to come.
- Fine tune your hiring to look for the skills your team needs
- Establish and assign key roles and responsibilities

Building the team

- Find people who are passionate about the problem you are solving, and are excited about the work to come.
- Fine tune your hiring to look for the skills your team needs
- Establish and assign key roles and responsibilities
- Make it clear that the requirements for this team are *different*
- Establish how the team works together as early as possible, and make that a team activity

How we collaborate



- Collaboration is absolutely critical to finding the best ideas, and you must make it safe and open for everyone to contribute

**• Ideas come from everyone, but we never get
too attached to them**

- In your area of expertise, you have the final word. All decisions have exactly one approver.
- Always seek collaboration, but not consensus

- When working in a small and nimble team, it is critical for every member to internalize their responsibility in building the product

If something doesn't seem right or fully thought out, speak up; draw attention to it and help figure it out.

- Everyone should be proud of what is being built, and if they are not, do something about it

- This applies to all functions!
 - This goes for internal sharing of work to get feedback, but also public releases to get real learning as quickly as possible
- Publish your work in the smallest possible increments**
- If we are not testing the simplest version as soon as possible, we won't know if we are actually on the right path to solve the problem

Publish your work in the smallest possible increments

- This applies to all functions!
- This goes for internal sharing of work to get feedback, but also public releases to get real learnings as quickly as possible
- If we are not testing the simplest version as soon as possible, we won't know if we are actually on the right path to solve the problem

We are self-reliant

- Reduce dependencies on other teams whenever possible
- Don't be afraid to make changes in other team's codebases if necessary. Communicate why you are doing it!
- Give the team the power to understand and fix as much of the system they rely on

We are self-reliant

- Being actual users of our product was the fastest way to find issues and make improvements
- Make it part of everyone's work process to use both new features, as well as what is in production, to discover pain points
- Speak to real users if possible!

Use our products

Use our products

- Being actual users of our product was the fastest way to find issues and make improvements
- Make it part of everyone's work process to use both new features, as well as what is in production, to discover pain points
- Speak to real users if possible!

- I am a firm believer in the power of having the smallest team size possible to complete the task

- When the team grows, make sure to encourage them to, keep breaking up into the smaller groups that can be extra effective
and self-organize

- Let the overall collaboration rules serve as a guide, but let people organize how works best for them
- Have established channels for groups to share updates and needs

1. Take care of yourself first.

1. Take care of yourself first.

Yes, that means you, too

Every challenge is different

Every challenge is different

Best of luck!