Minimize Your Circus Factor

Building resilient teams

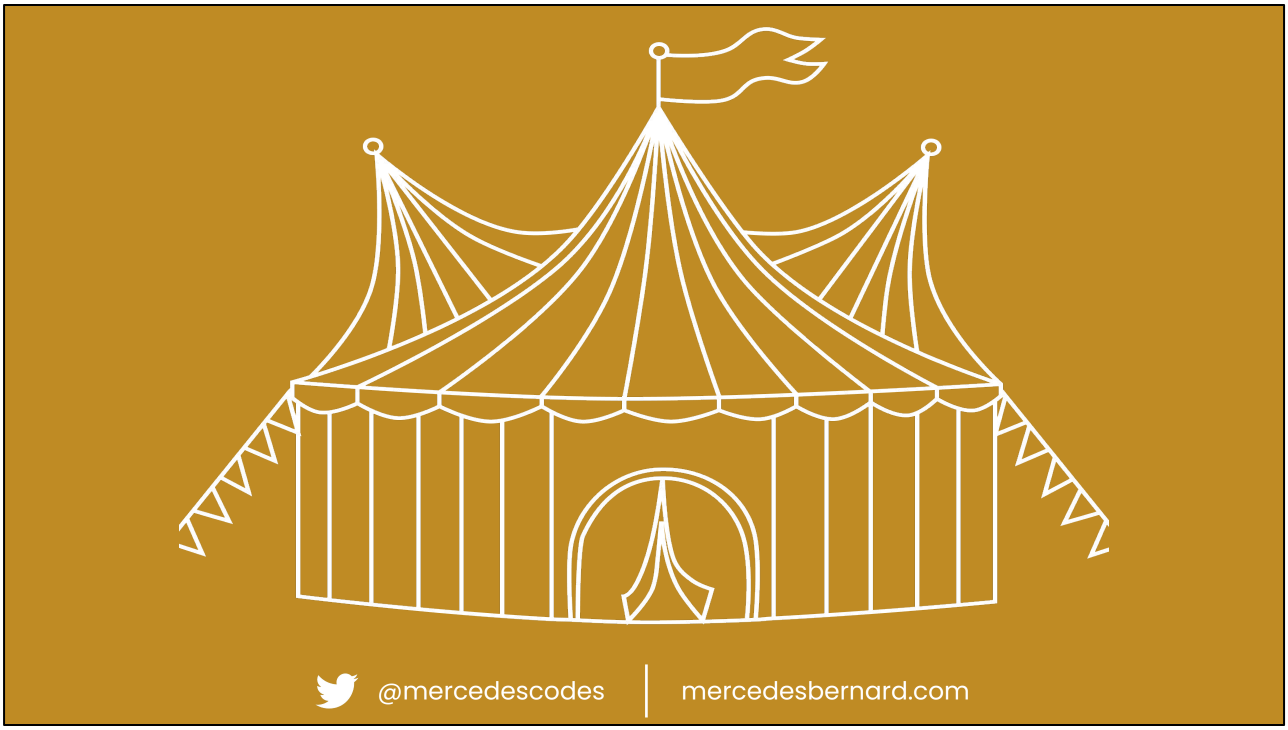
@mercedescodes | mercedesbernard.com

Hello! Welcome to Minimize Your Circus Factor: Building resilient teams. My name is Mercedes Bernard and I’m a principal engineer with Cloud City, a software consulting firm in the Bay area.

mercedesbernard.com/speaking/minimize-circus-factor

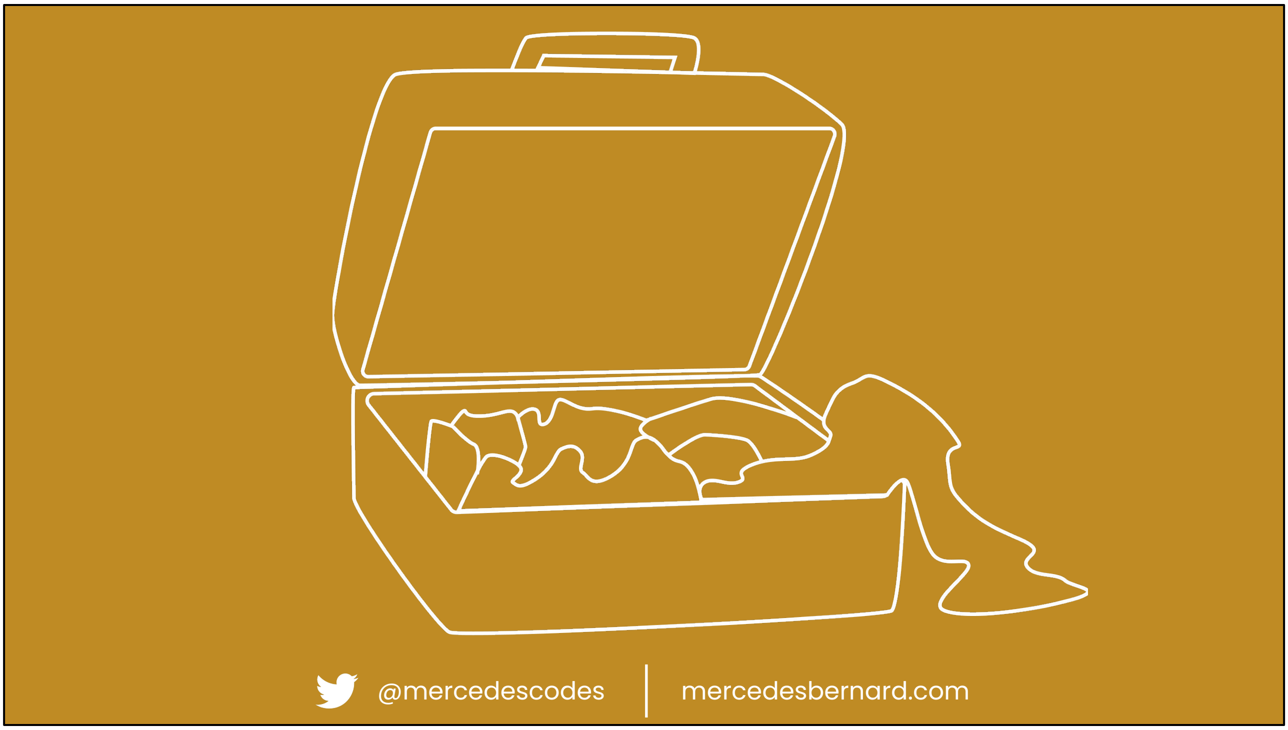
@mercedescodes | mercedesbernard.com

If you are someone who likes to follow along with slides or if you want the reference links for some of the stats I mention in this talk, I've published the slides on my website.



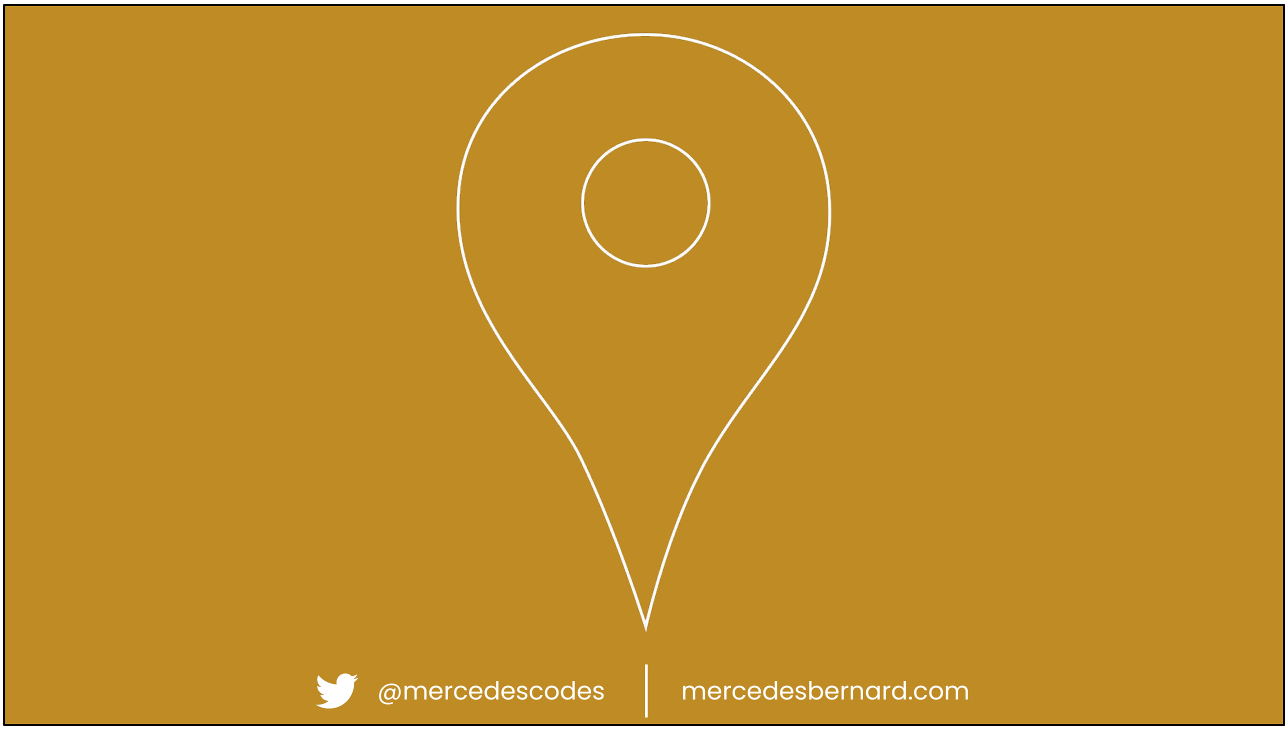
@mercedescodes | mercedesbernard.com

You may have heard of the “bus factor” which is a bit morbid of a metaphor for illustrating the impact on your team if you were to leave suddenly. I prefer to think about it as your circus factor. If you were to run away and join the circus, what impact would that have on your team? Are they set up to succeed?



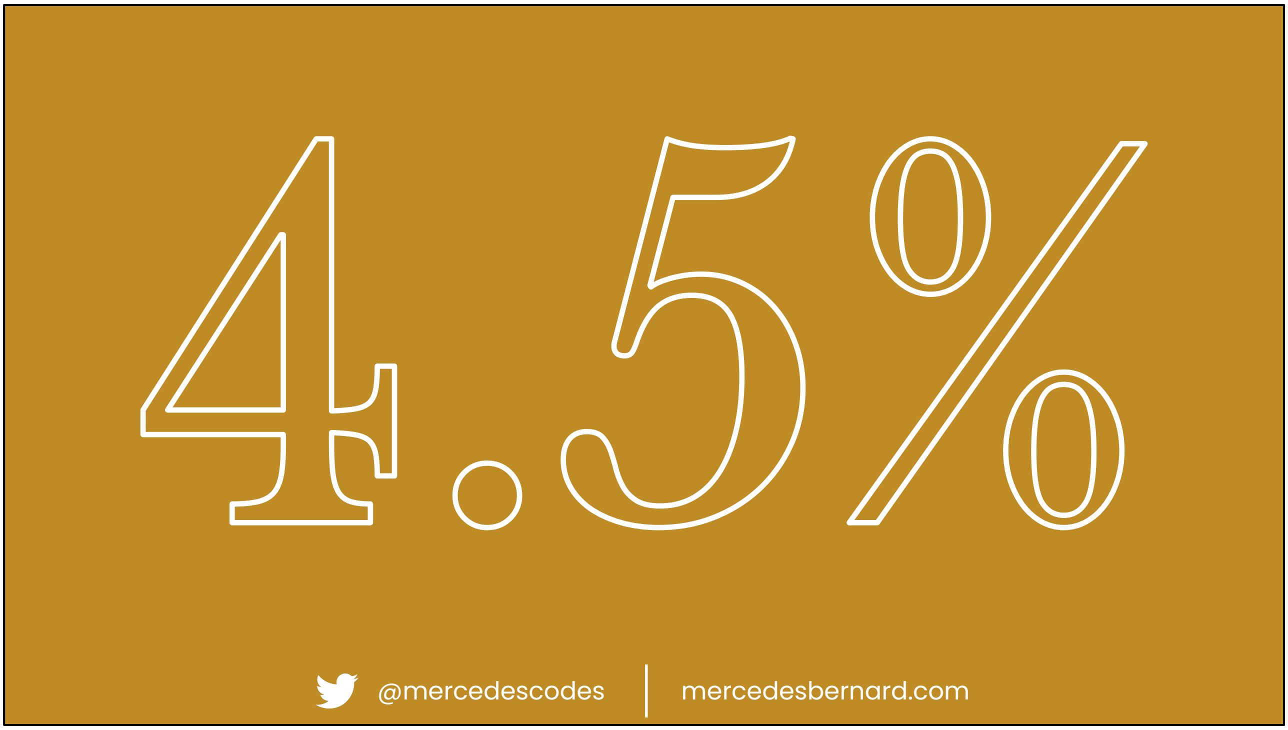
@mercedescodes | mercedesbernard.com

I recently left the company I had been with for 3 ½ years, the last 2 of which I held a leadership role within the organization. I was planning to leave for about 6 months prior to my job search and wanted to do what I could to make the experience as unstressful as I could for my team. I had a wonderful group of humans that I still care about very much so this was really important to me. Did I get it 100% right? Nope. But I did learn a lot and I was able to ease the transition and find strategies that helped them continue to deliver high-quality work with as little disruption as possible.



@mercedescodes | mercedesbernard.com

I turned in my resignation in April. I was part of the record 4 million people who quit their jobs in the US that month. We are currently in the midst of what is being called The Great Resignation. And the tech industry in particular is seeing higher than normal turnover and unfilled jobs.



4.5%

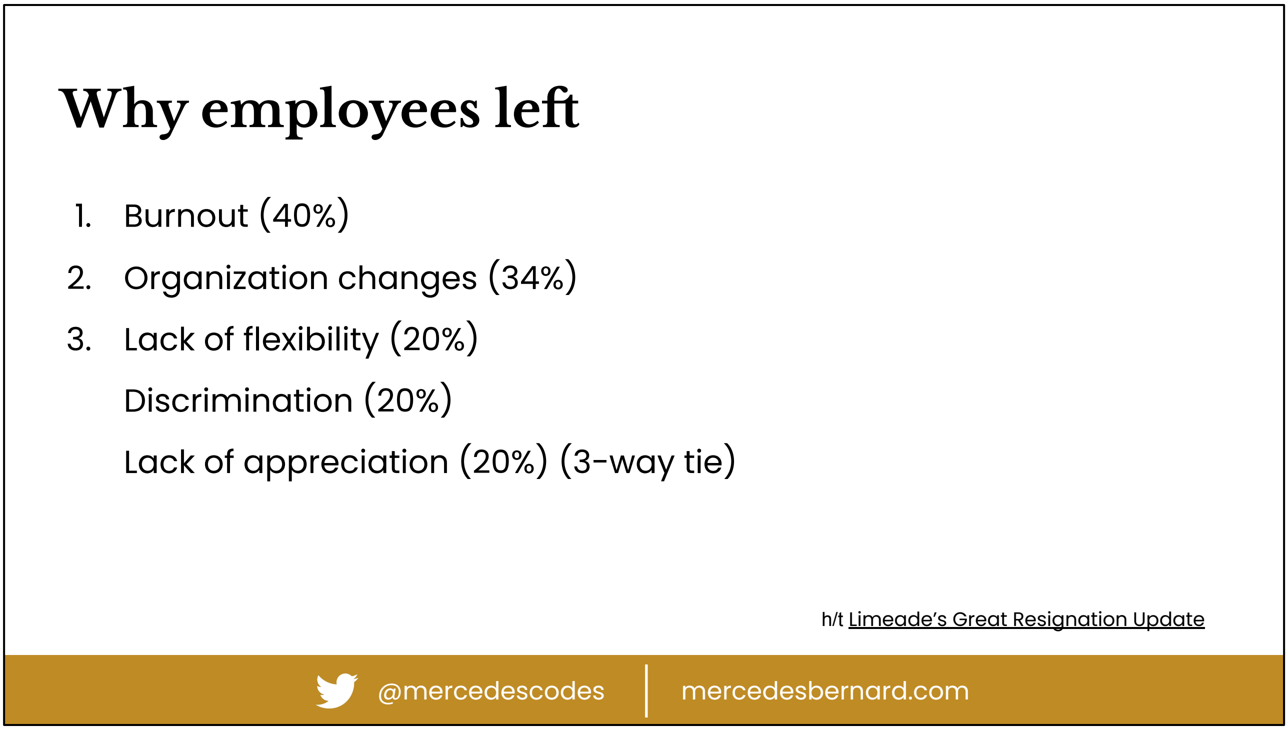
@mercedescodes | mercedesbernard.com

According to a report released by Visier, tech industry resignations are up 4.5% in 2021[1] [2]. This is on top of the fact that tech historically already has the highest turnover rates of any industry with stats over the last few years being cited anywhere between 13 and 21% depending on how specifically the report defined “tech worker”.[3] For reference the average turnover rate is somewhere around 3.5 - 4% depending on the month.[4]

700,000

@mercedescodes | mercedesbernard.com

It is currently projected that there will be about 700k unfilled tech jobs this year.[5] Even during the height of the pandemic when the economy looked bleakest, that number only shrunk to 500k. (Side note: there are also between 70k-80k people entering the industry every year looking for their first software development jobs. It seems that a 10 year investment into early career folks could have helped prevent our current situation)



Why employees left

1. Burnout (40%)
2. Organization changes (34%)
3. Lack of flexibility (20%)

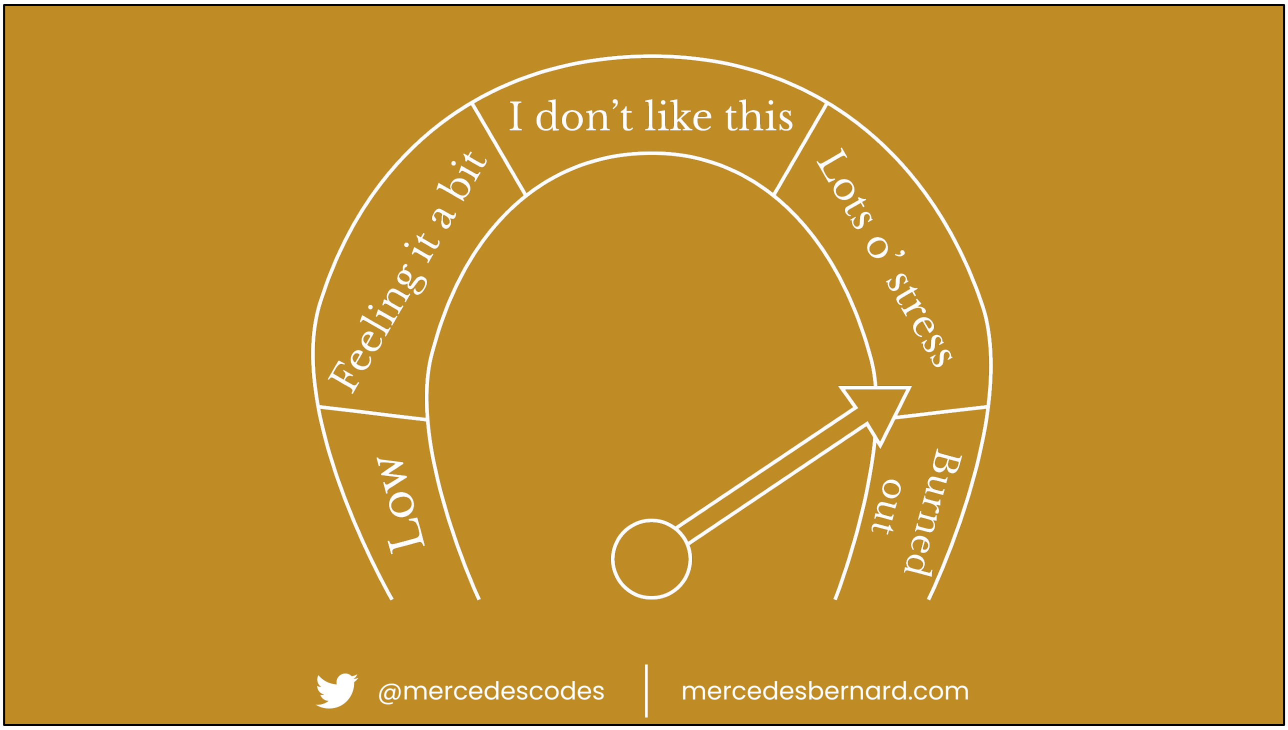
   Discrimination (20%)

   Lack of appreciation (20%) (3-way tie)

h/t Limeade's Great Resignation Update

@mercedescodes | mercedesbernard.com

High demand + high stress and low satisfaction due to companies continuing to operate as normal during a global pandemic and failing to recognize their employees as full humans leads to high attrition. As companies try to get back to "business as usual," burnout, org changes, lack of flexibility, discrimination, and lack of appreciation are some of the currently top-ranked reasons why workers are looking to switch jobs.[6] [7] Given that most knowledge workers prefer flexible remote working arrangements, many folks are looking for remote-friendly jobs that will value their contributions.



Low

Feeling it a bit

I don’t like this

Lots o’ stress

Burned out

@mercedescodes | mercedesbernard.com

Attrition is expensive. And stressful for those remaining on the team. With the current turnover and hiring trends, it’s going to take a while for anyone to backfill a role which means those who stay with a team may be dealing with the stress of being understaffed for a while.

Care for humans, not companies

@mercedescodes | mercedesbernard.com

This is the unfortunate reality so if we are in a senior, lead, or manager role, it is our responsibility to enable and empower our teams to do their best work both while we are working with them and after we leave.

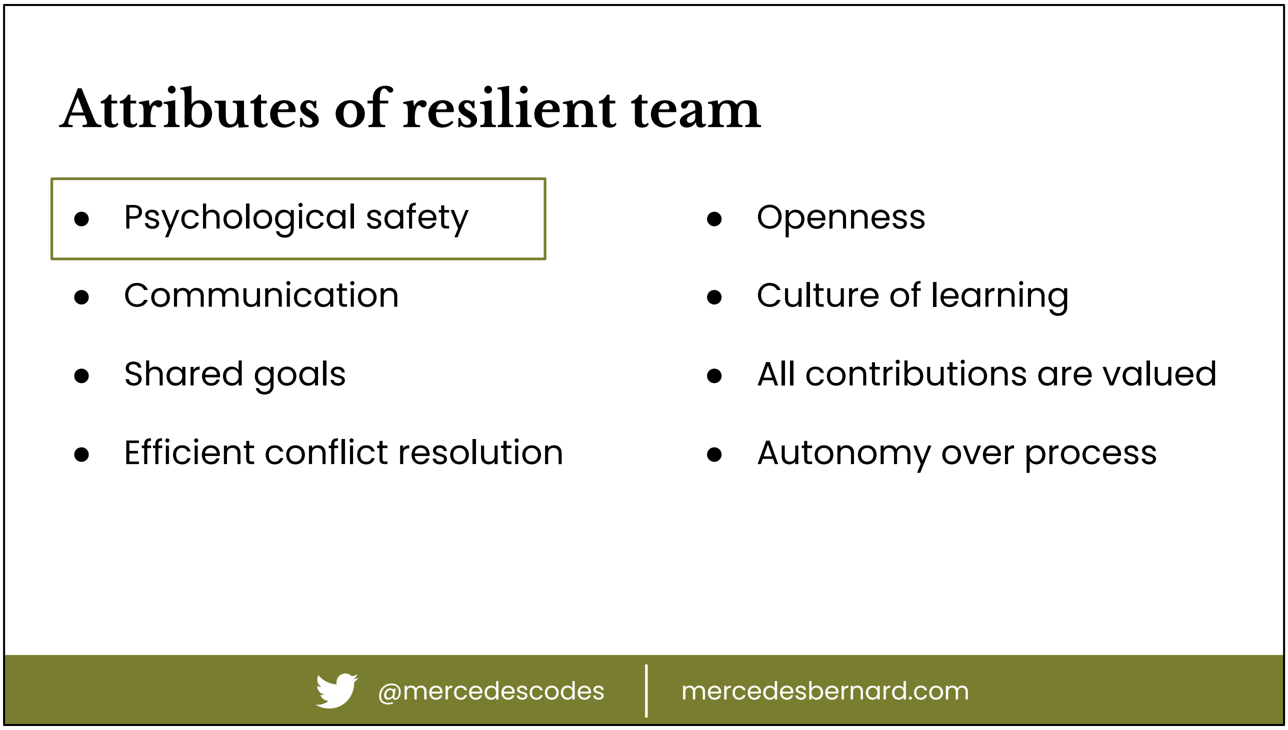
I want to make it clear that this talk is about individual interventions aimed at enabling and empowering your team members. At no point during this talk am I advocating for you to take on extra responsibilities for the good of the company or to bend over backward without additional compensation. Unfortunately under capitalism, companies are not incentivized to care about you. Your role with a company is a business relationship. If you are doing what is outline in your job description (and no more!), then you are fulfilling your job duties. My goal is to give you low-effort strategies to incorporate into your daily work that will empower those around you without making more work for yourself. And if these strategies are successful, they may actually help share responsibilities and create more work/life balance for everyone on the team.

The perspective that I'm taking in this talk and that I hope you'll walk away with is that these are strategies to care for the humans around you and lower their stress levels without doing uncompensated or disproportionate labor for the company.

Resilient teams

@mercedescodes | mercedesbernard.com

Resilient teams are important because we work in an industry that is constantly changing and there are always new challenges. If we have a team that is highly cohesive and resilient, they'll be able to adapt to changes and respond to challenges without overwhelming stress. But what does that look like in practice?

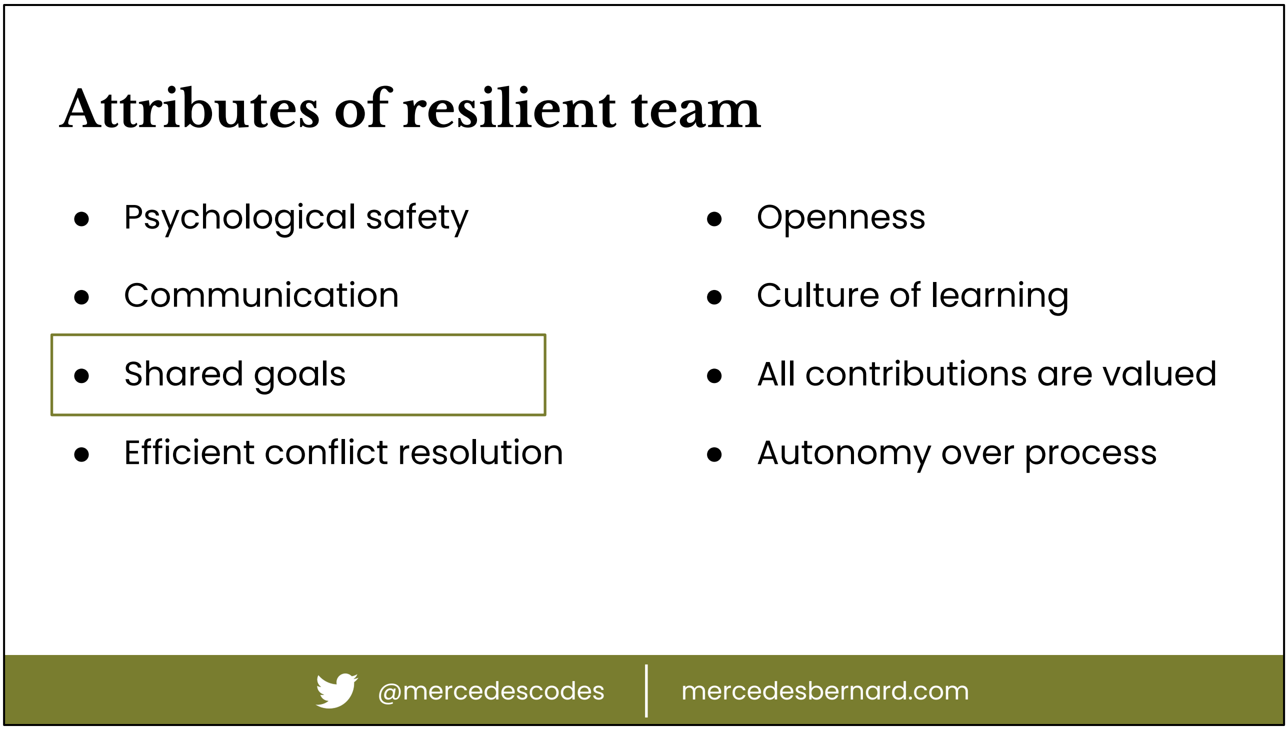


Attributes of resilient team

- Psychological safety
- Communication
- Shared goals
- Efficient conflict resolution
- Openness
- Culture of learning
- All contributions are valued
- Autonomy over process

@mercedescodes | mercedesbernard.com

Resilient teams are those with high levels of trust and psychological safety that can adapt to change while maintaining an enjoyable working culture.

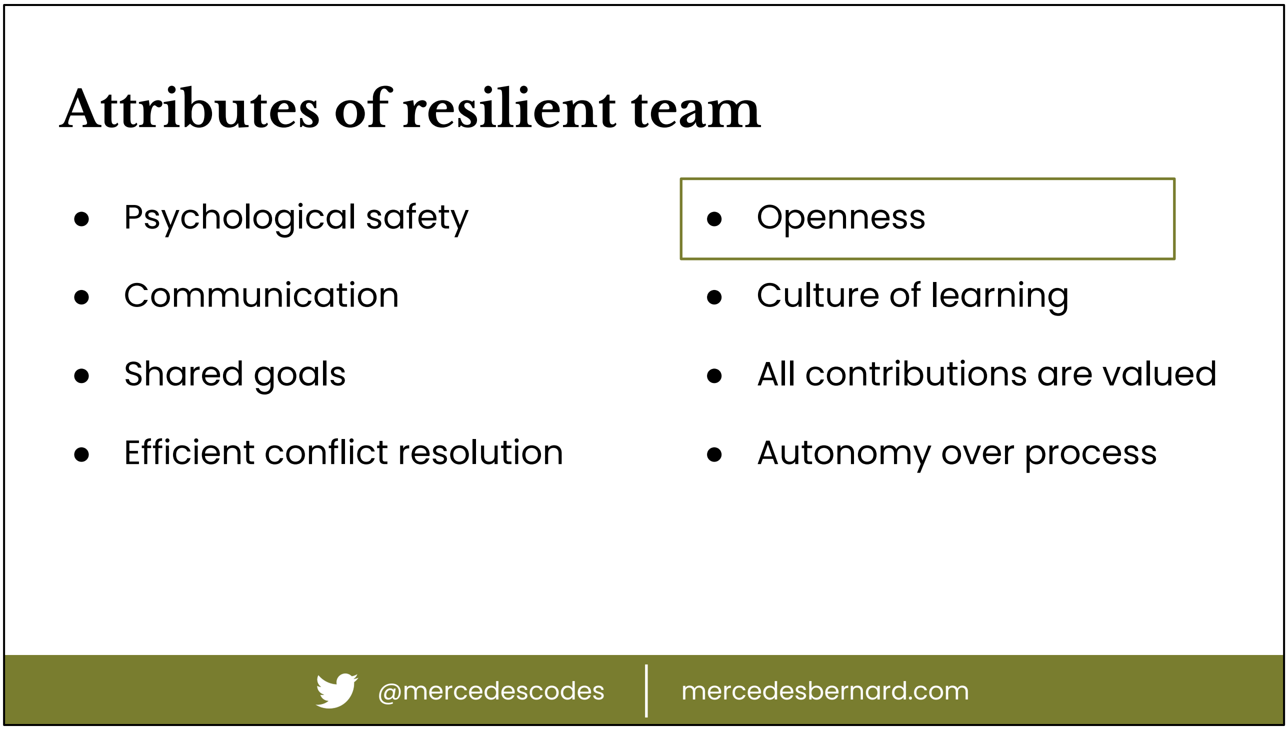


Attributes of resilient team

- Psychological safety
- Communication
- Shared goals
- Efficient conflict resolution
- Openness
- Culture of learning
- All contributions are valued
- Autonomy over process

@mercedescodes | mercedesbernard.com

They are those in which no one member of the team carries an outsized burden, everyone shares a vision and common goals, and everyone is committed to each other's individual success as well as the team's success.

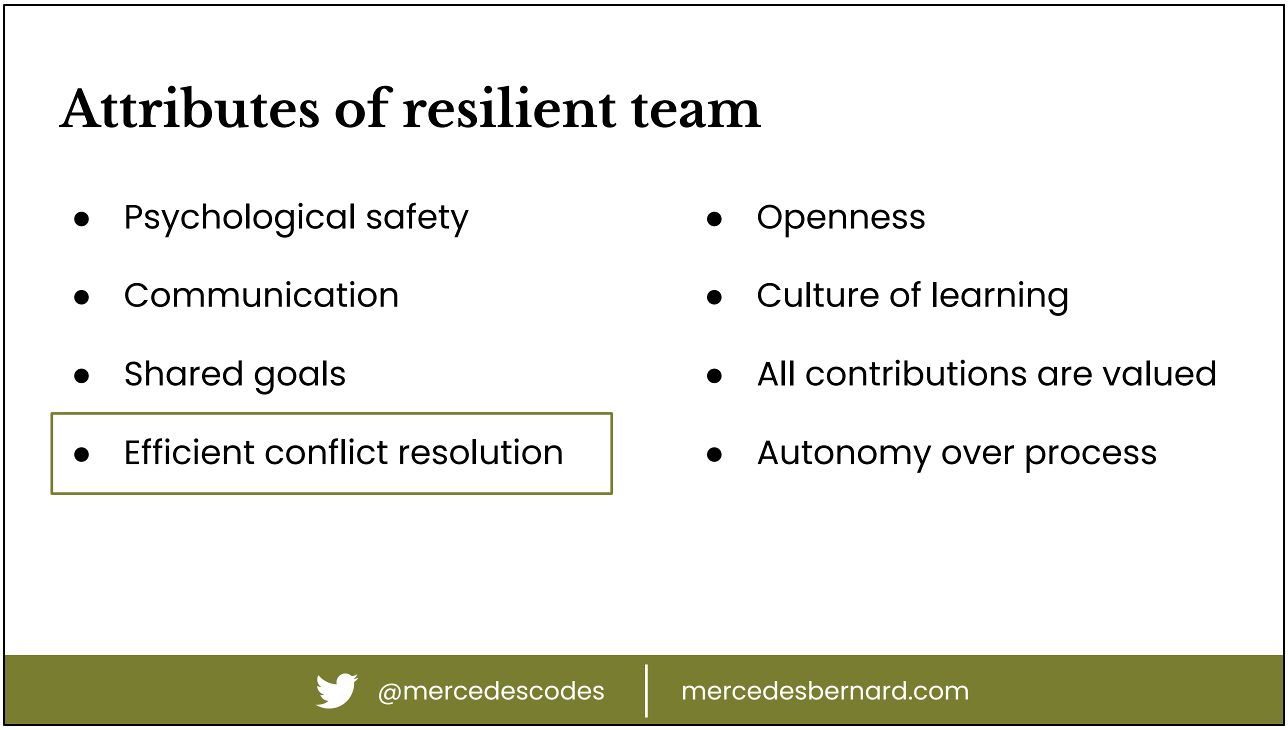


Attributes of resilient team

- Psychological safety
- Communication
- Shared goals
- Efficient conflict resolution
- Openness
- Culture of learning
- All contributions are valued
- Autonomy over process

@mercedescodes | mercedesbernard.com

To build a resilient team, you need to create a culture in which the team and the individuals on the team will thrive. You need to create a culture where open and transparent communication is valued and modeled from the top down.

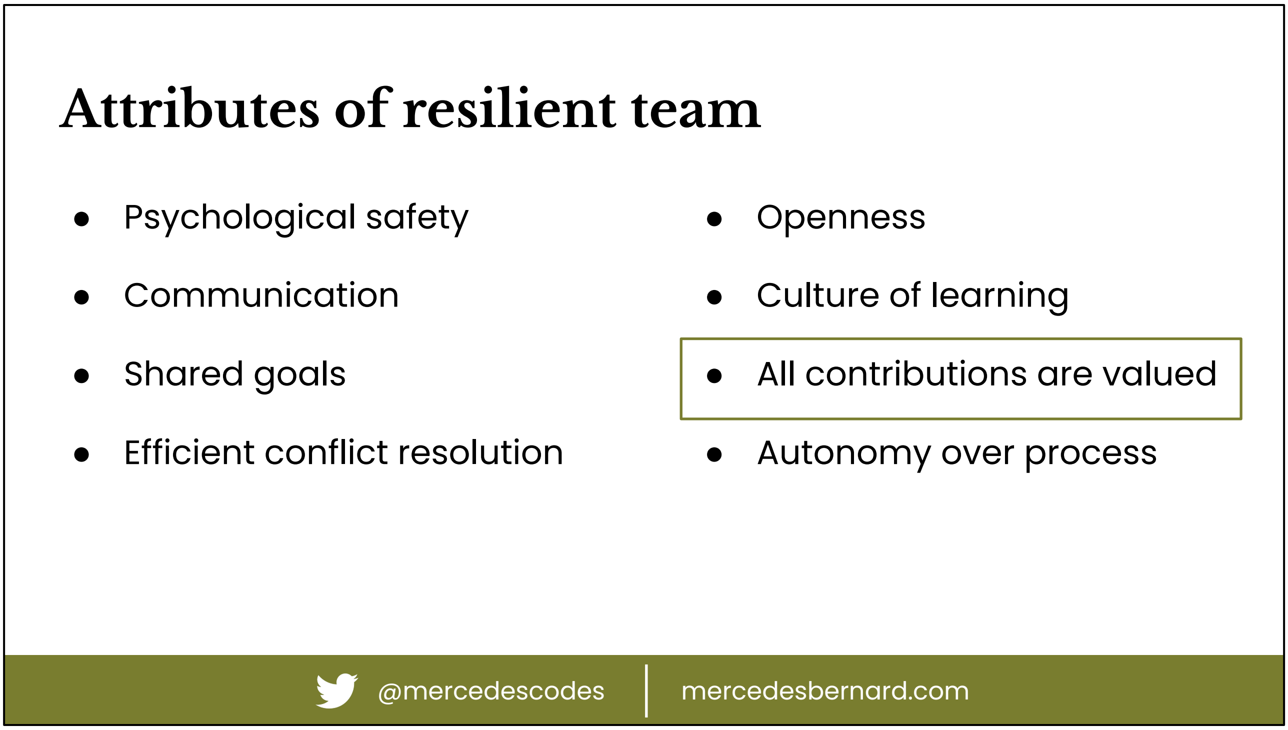


Attributes of resilient team

- Psychological safety
- Communication
- Shared goals
- Efficient conflict resolution
- Openness
- Culture of learning
- All contributions are valued
- Autonomy over process

@mercedescodes | mercedesbernard.com

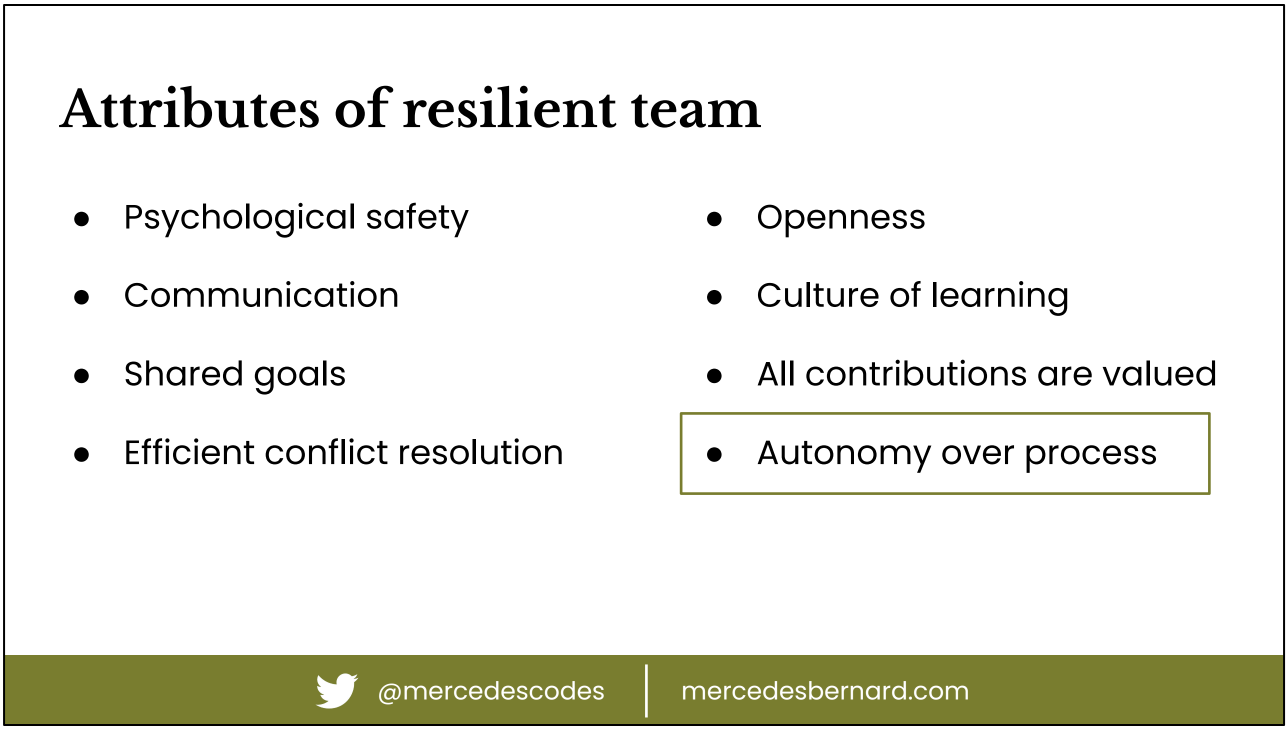
You need to be willing to approach conflict with curiosity.



Attributes of resilient team

- Psychological safety
- Communication
- Shared goals
- Efficient conflict resolution
- Openness
- Culture of learning
- All contributions are valued
- Autonomy over process

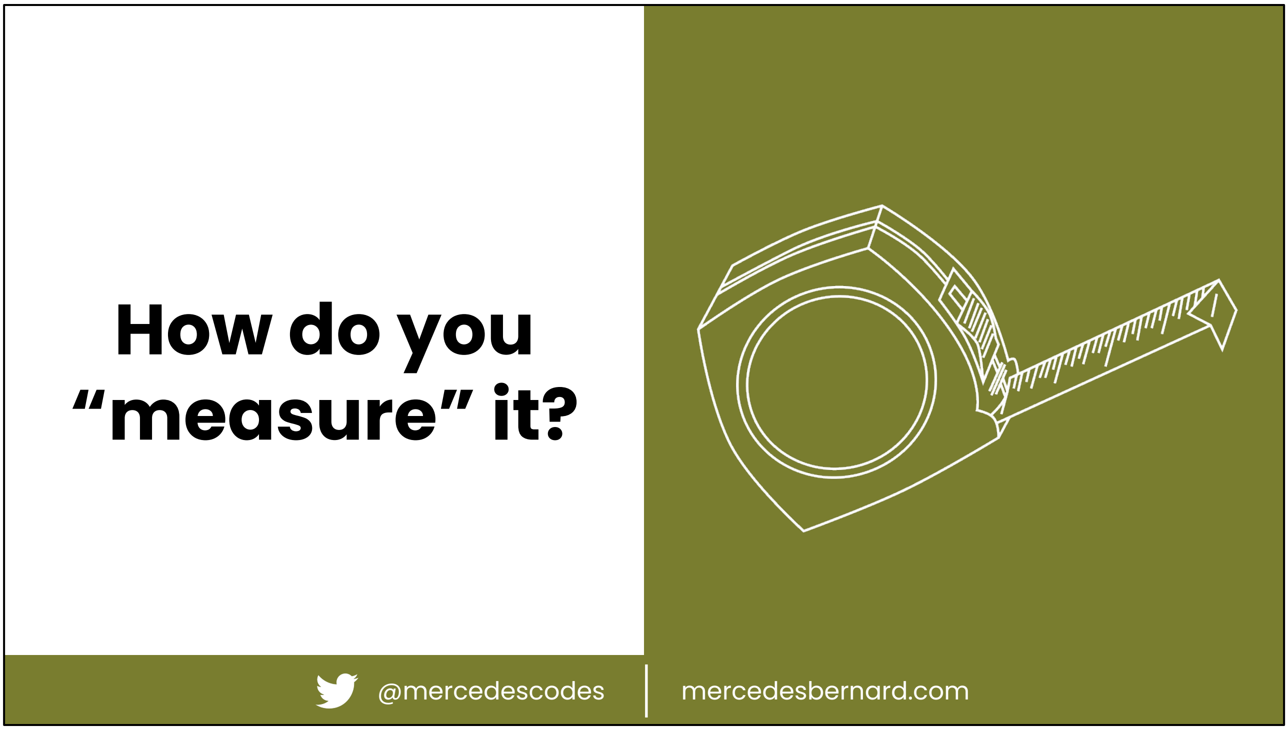
You need to show outward appreciation and gratitude towards all members of the team and share that gratitude with others in the organization.



Attributes of resilient team

- Psychological safety
- Communication
- Shared goals
- Efficient conflict resolution
- Openness
- Culture of learning
- All contributions are valued
- Autonomy over process

And you need to let your team do what they do best. Give them control and latitude over how they get their work done so that they can do it in whatever way they'll be most successful.



How do you “measure” it?

@mercedescodes | mercedesbernard.com

This is all well and good, but how do you know if you’re on your way to being part of a resilient team? How in the world do you measure something like that? Well, I can’t give you anything quantitative. There’s no such thing as a psychological safety unit (PSU). But if you pay attention to how the team responds to challenges, you can start to gauge where there is high resilience and where we may need to invest in some of those attributes from the previous slides.

“I don’t know”

@mercedescodes | mercedesbernard.com

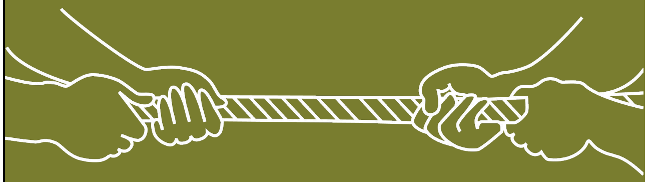
How often do you hear someone on the team admit that they don’t know something? When they admit that, is there shame or a genuine desire to learn?

When they do admit they don’t know something, how does the rest of the team react? What offers of support are provided by other members of the team?

Shifting priorities

@mercedescodes | mercedesbernard.com

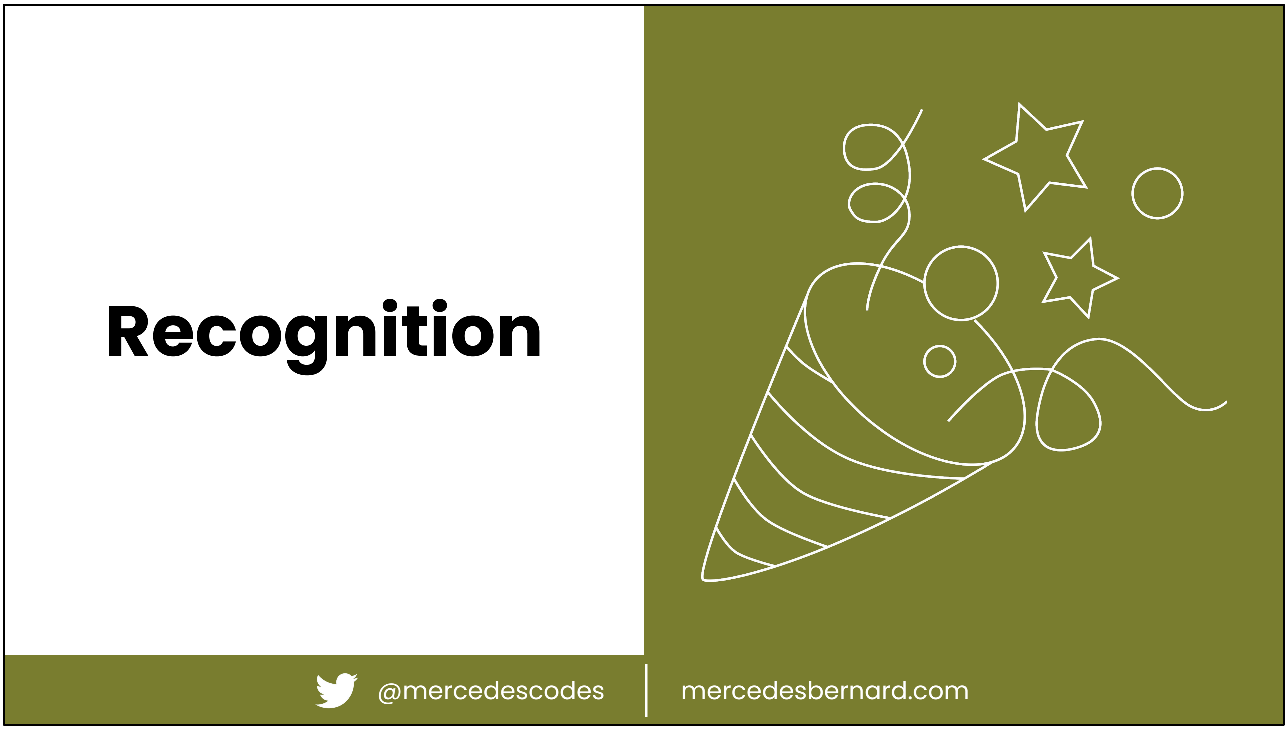
When priorities shift, how are they communicated to the team? Does the team feel well-informed after an announcement? Do they ask questions? Do they ask open-ended questions? Are they included in the implementation planning for delivering on the new priorities?



Conflict

@mercedescodes | mercedesbernard.com

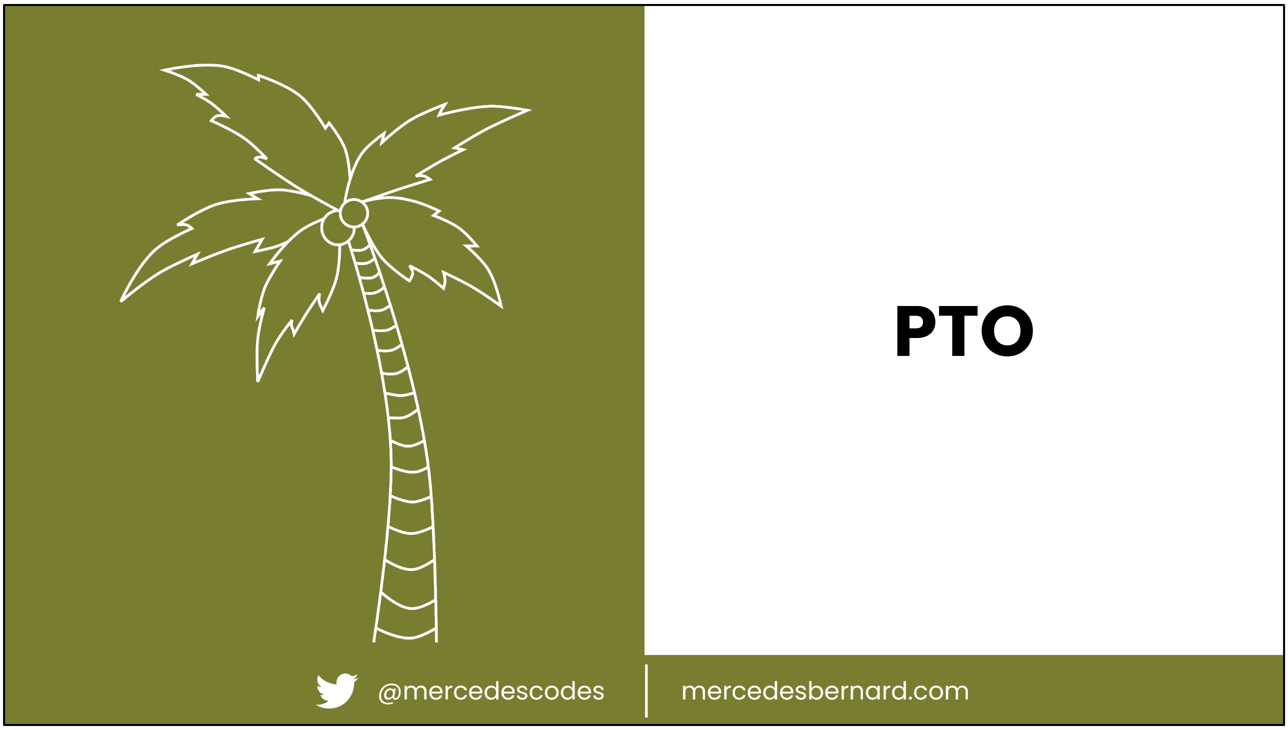
When a conflict or disagreement arises, how long does it fester? How well does the team navigate the discussion? How open are folks to seeking help to resolve disagreements? Is everyone able to stay respectful and continue to maintain trusting working relationships after a conflict?



Recognition

@mercedescodes | mercedesbernard.com

How openly and freely are folks on the team recognized for their contributions? How often do members of the team give each other shout-outs? Do team members share credit with each other? Are early-career team members’ contributions shared as widely as the senior members’ of the team?



PTO

@mercedescodes | mercedesbernard.com

And finally, one of my personal most important measurements for team resilience.

When folks take PTO, actually let’s back up. Do people take PTO? Do they feel comfortable taking PTO? When they take PTO how much planning and prep needs to take place beforehand? How many meetings get scheduled on the calendar to take advantage of their time and knowledge sharing before they leave?

How often is PTO interrupted by emails or Slack?

Highly resilient teams can manage change and other sources of friction with little disruption and can give each team member the space they need to enjoy their time away.

Support delivery

Support people

Support process

@mercedescodes | mercedesbernard.com

The best way to minimize your circus factor is to consistently support delivery, support people, and support process.

We'll talk about each of these and strategies you can use every day to make it as easy and low-stress as possible should you ever want extended PTO, unexpected leave, or to pursue a new opportunity with a different company.

Improving delivery resilience

What we do

@mercedescodes | mercedesbernard.com

Delivery resilience refers to the actual work product that we produce. Does everyone on the team have the knowledge and tools to continue to deliver quality, consistent work even when there is churn?

Improving people resilience

Who we do it with

@mercedescodes | mercedesbernard.com

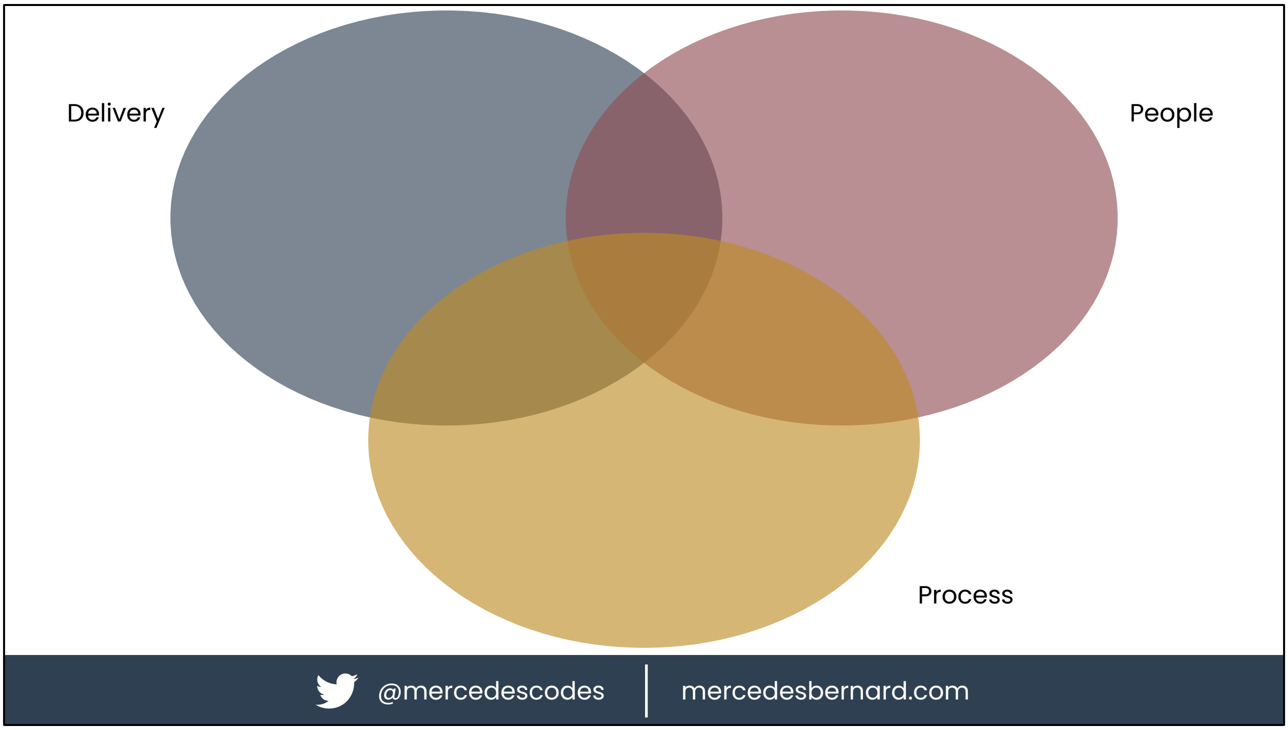
Very closely related to improving technical ability to consistently deliver quality work is investing in the people around you and their people skills. Invest time in their careers and support them in the non-technical skills they need to continue to succeed beyond this current team and current role.

Improving process resilience

How we do it

@mercedescodes | mercedesbernard.com

Finally, process resilience refers to simplifying and strengthening the processes we have that facilitate running our teams and our business. Strong, simple processes are not dependent on any one individual for them to be able to be followed and to keep things in good working order.



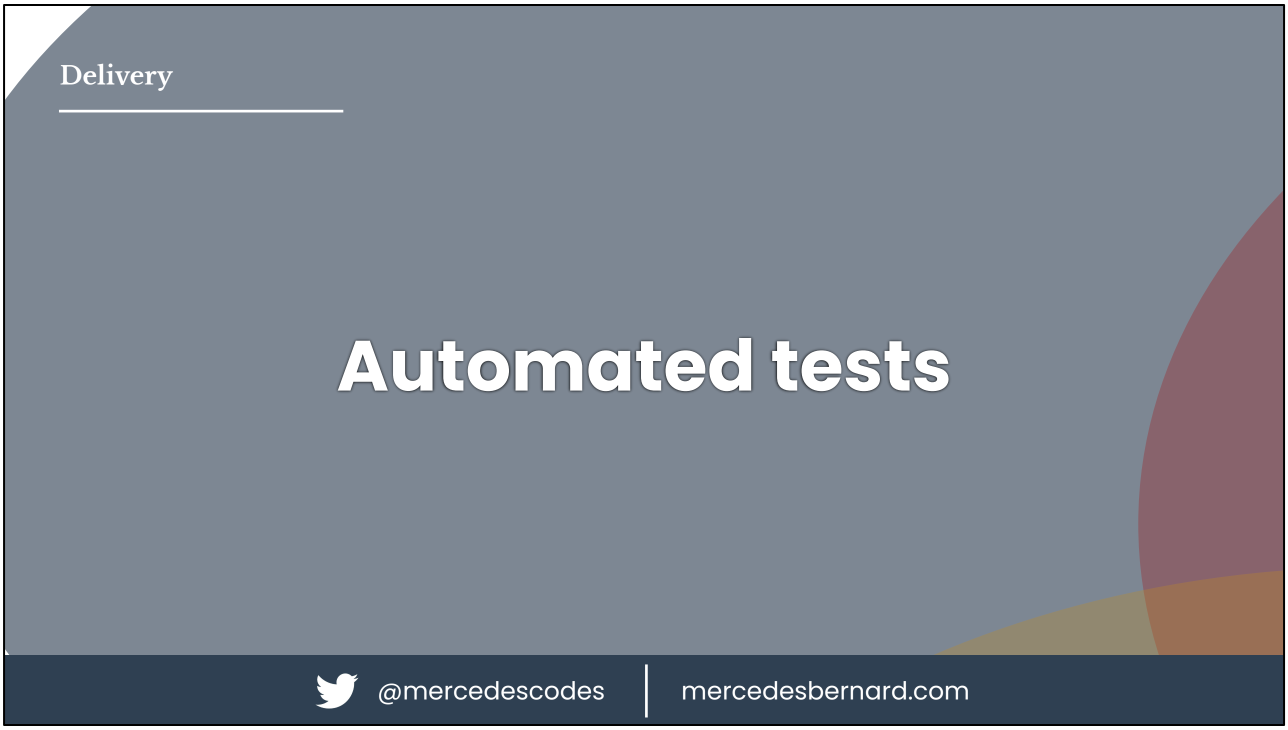
Delivery

People

Process

@mercedescodes | mercedesbernard.com

A lot of the strategies that we’re going to talk about today for minimizing our circus factor don’t fall cleanly into one of these 3 categories. Often things that support our internal processes also support the people we work with. As we go through some ways to reduce friction and create a sustainable environment for your team to continue to be successful, I’ll try to call out where in this venn diagram the strategies fall and why,



Delivery

Automated tests

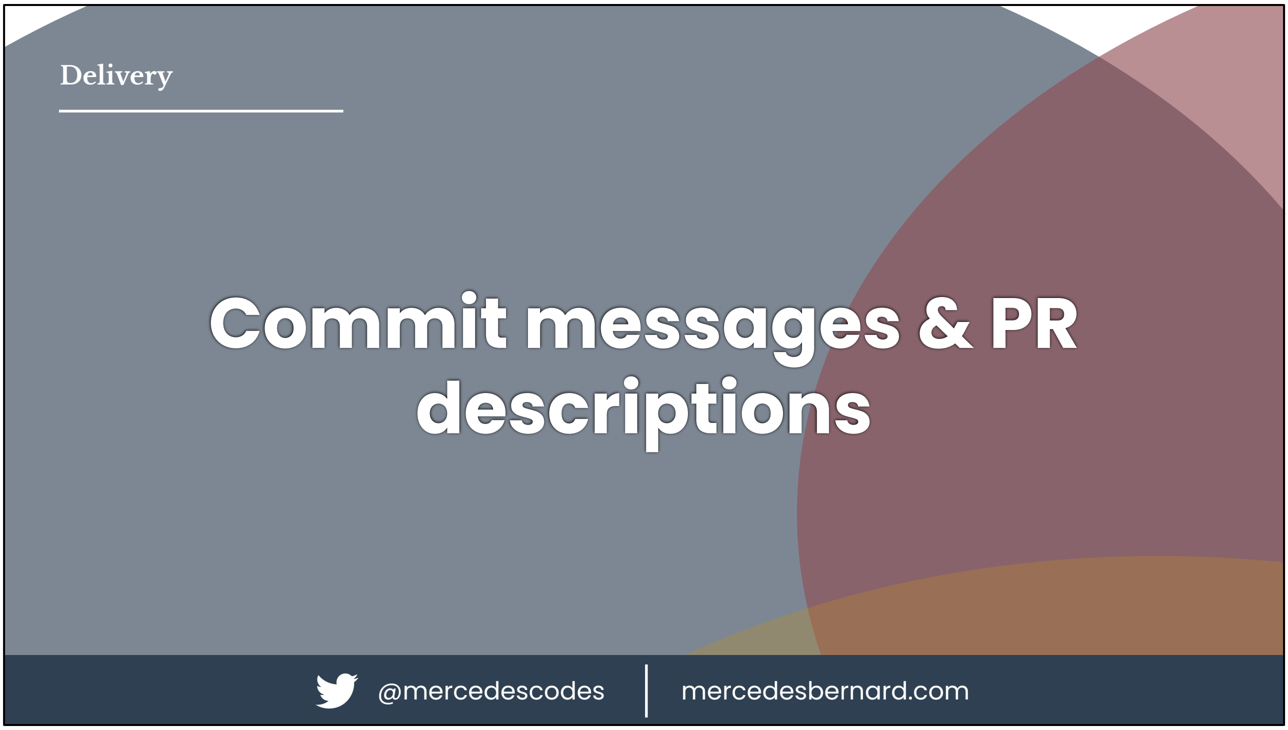
@mercedescodes | mercedesbernard.com

The first strategy is automated testing. This is pretty clearly a delivery strategy.

I’ve seen teams that don’t test at all and I’ve seen teams that enforce 100% test coverage. In both of those cases, the tests (or lack of tests) were not useful to current and future team members. Tests are a great tool for catching regressions but more than that, they are a main source of developer documentation. They communicate expected behavior, expected side effects, and how to use components within the code to other devs. If you don’t have any tests or if you require such extreme test coverage to the point that your tests aren’t valuable and are just copy pasta in order to cover lines of code, then the documentation they provide isn’t helpful.

Support your team members by writing valuable tests. Make sure that the descriptions of the tests cases are accurate and describe actual behavior or context and not just the state of variables. Write your tests so that someone could read the descriptions and assertions and understand the responsibility of the tested code.

Avoid writing brittle tests that care too much about implementation details so your team doesn’t have to invest time in keeping them green. You want your team members to understand and enjoy living with your code even after you’re gone.



Delivery

Commit messages & PR descriptions

@mercedescodes | mercedesbernard.com

Next we have another form of developer documentation but this one moves a little into supporting people, not just delivery, because we can provide a lot of extra info for our team to learn from.

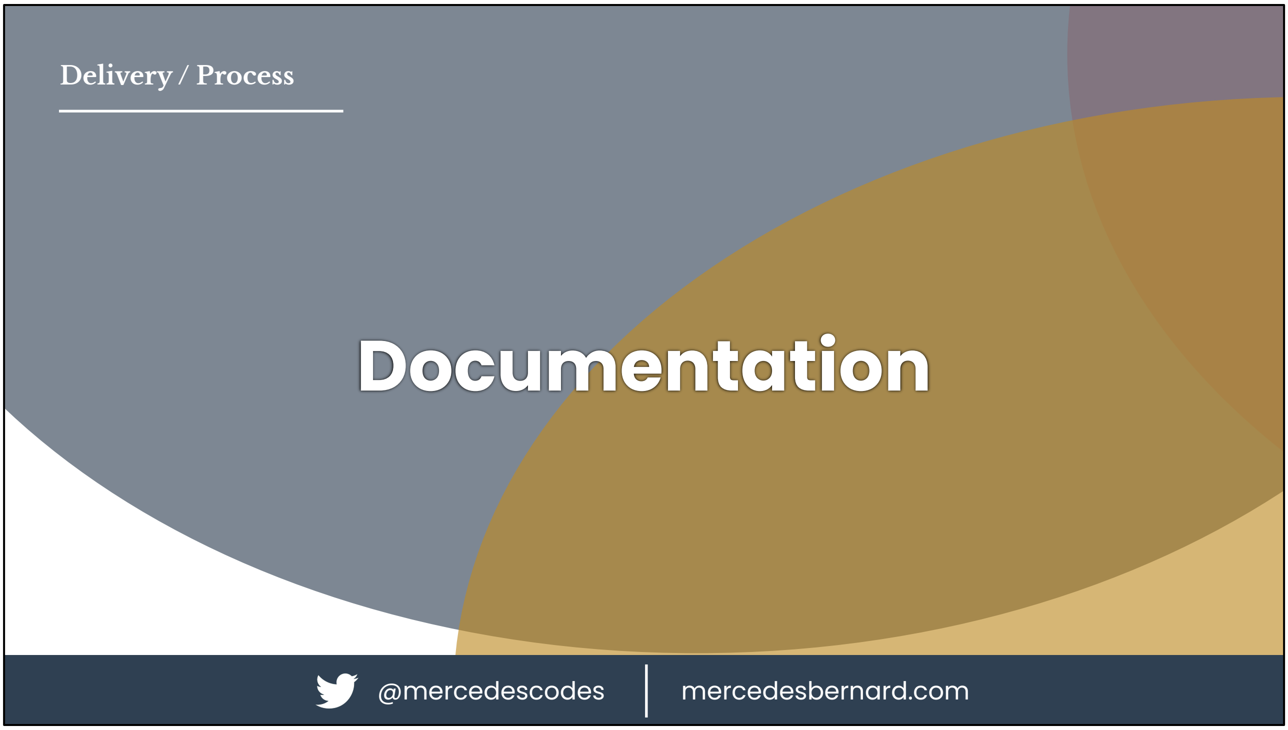
Lately, I've been finding myself in situations where I have to dig through code that's 6 months to a year old but the authors have left the company and the commit messages are just the JIRA ticket number. When I look for the PR, the description is pretty sparse, just reiterating the Jira description or describing exactly what's in the code. These situations feel like an archaeological dig where I'm looking for clues as to why something was written or what it's intent might be and trying to figure out what behavior is intentional and explainable and what might be a bug or an unintended side effect without any of the context or the ability to ask questions.

In these cases, a good commit message or PR description could save so much time and provide helpful information.

To prevent situations like this for your future teams and developers, write descriptive commit messages and helpful PR descriptions. Explain the context and need for the change, who asked for it and why, rather than just describing the code change. Hopefully the code is straight-forward but if not, all the other info helps just as much as explaining the tricky parts of the code itself. And be sure to include information on how to test it. Information about preconditions and various use cases and their outcomes is valuable for future developers if they have to debug something and need to quickly familiarize themselves with expected behavior.

In the case of tricky code, links to documentation are also a big time-saver. If you found a bug in open source code and had to implement a non-obvious work around, include a link to the Github issue in your commit description.

Similar to automated tests, commit messages and PRs are often the first place we look for answers and the first thing we ignore in the interest of time.



Delivery / Process

Documentation

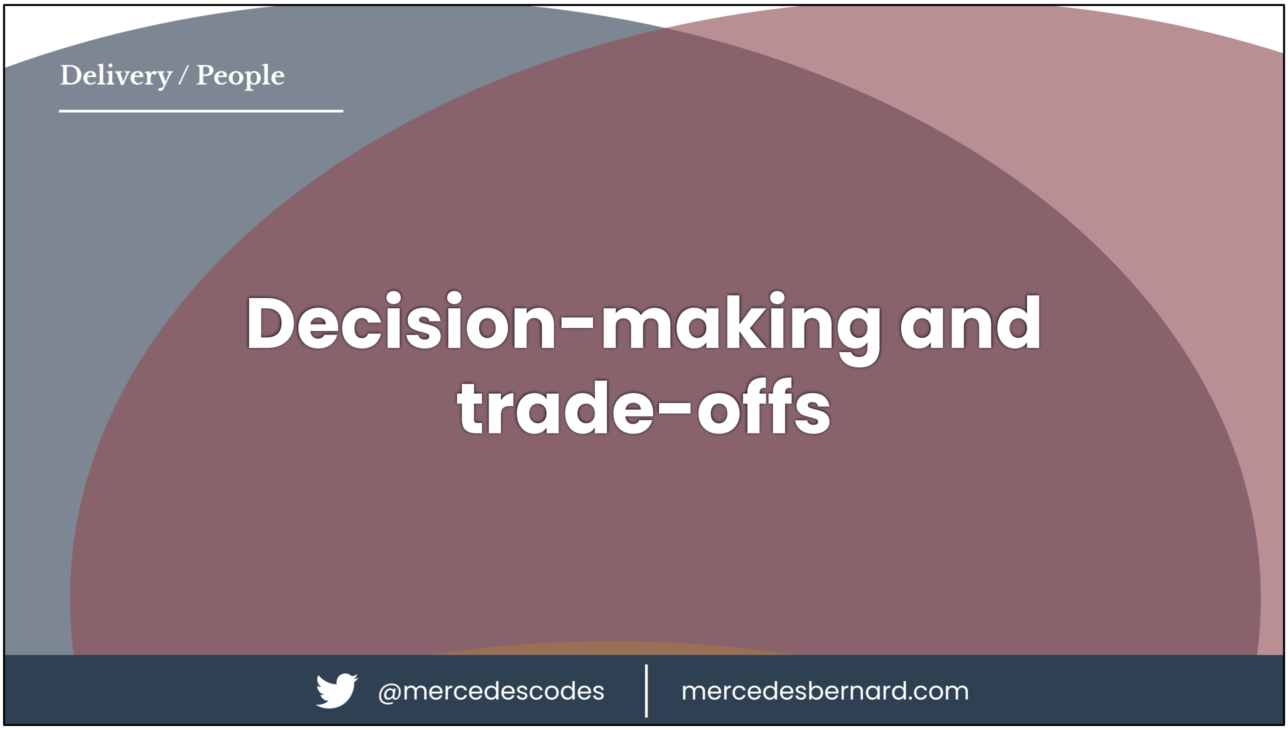
@mercedescodes | mercedesbernard.com

We have a theme :) Documentation is important, who knew? But here I'm talking about more traditional forms of documentation which is why we're going to put this in between delivery and process.

This is the most obvious strategy in this talk but I see it consistently deprioritized and it leads to long-term consequences. If nothing is documented, organizational context lives in the heads of individuals rather than teams. If only a select few people, usually those with the longest tenure and as-such those who are the biggest attrition risk, are the ones who know why things are the way they are or know how all the pieces of a system work together. If they leave the team suddenly, all of their context is lost.

Typically, documentation is only prioritized after someone has given notice or right before an extended vacation. And no matter how hard anyone tries, there are always things that are forgotten or deemed to be not that important when there are other, bigger things that need to be documented. To combat this, get in the habit of documenting everything in a convenient place for your team. Technical documentation should live as close to the code as possible. Please don't leave breadcrumbs all over random Jira comments because even though something is technically "documented," very few people will be able to find it and get use out of it.

A recent example I encountered was at a client who had a single sign-on (SSO) option set up for users. The engineers knew how it worked in prod but never used it during local development or testing because they had multiple accounts to test with for different features so they always logged in with their various email addresses. But this meant that when a bug was reported, no one knew how to use SSO in local dev to troubleshoot and fix it. The engineers who had originally worked on this feature were long gone. So it required myself and another engineer 2 weeks to find all the pieces of the local infrastructure and the correct configuration in order to set it up for testing. If this had been documented, we could’ve spent an hour or two figuring out how to test it so that most of our time could be spent fixing the bug.



Delivery / People

Decision-making and trade-offs

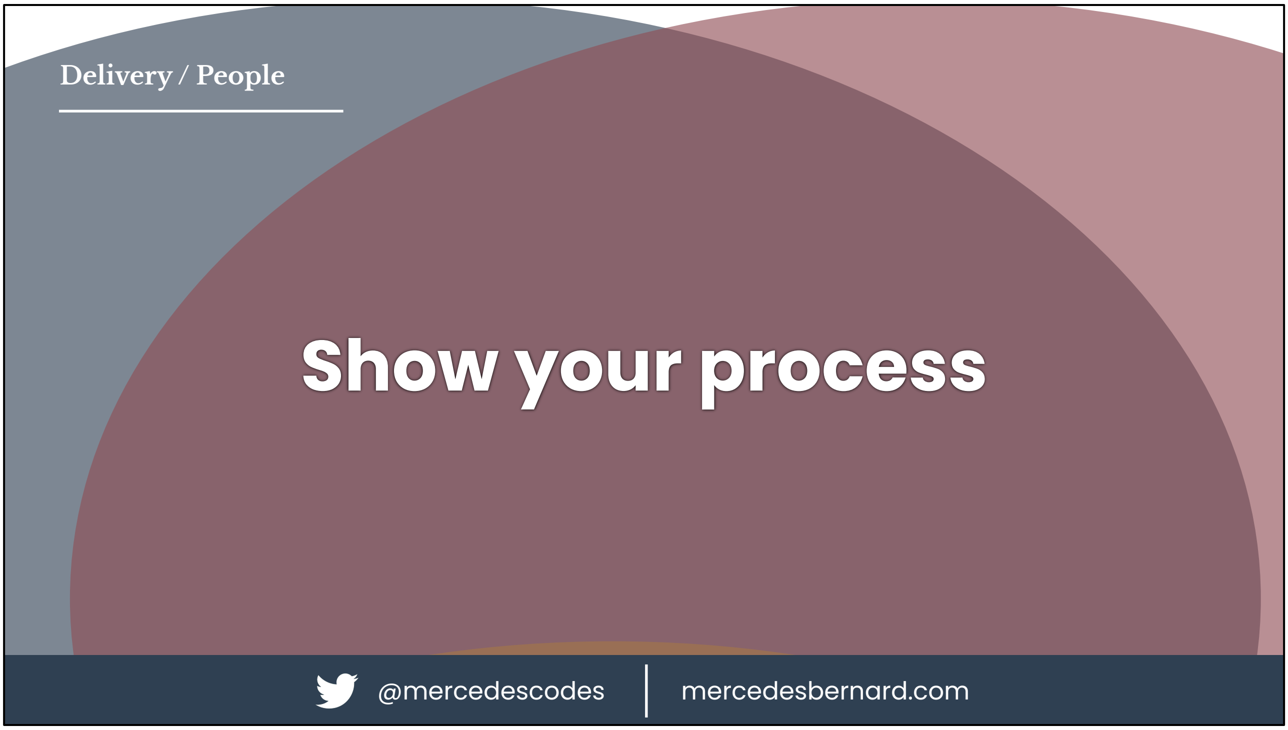
@mercedescodes | mercedesbernard.com

Now we’re going to move a bit more into supporting the people around us.

I’ve talked about this in previous talks I’ve given but one area that I think many early career folks don’t receive enough mentorship in is how to make decisions and evaluate trade-offs. We’re often so focused on becoming technically competent and being able to write good code that we don’t spend as much time learning how to make technical decisions. If you are a senior member of your team, one of the ways you can support the folks you work with to be empowered after you leave and to continue delivering great work is to coach them on how to make decisions and evaluate trade-offs so that they feel confident doing that independently after you’ve left.

How do you do that?

Well first, always make sure that you explain the “why” for every decision you make. This can be any kind of decision. If you are deciding on the team’s upcoming pair rotation schedule, explain how you took PTO into consideration for the schedule or how you balanced skill levels and unique strengths for a specific upcoming feature (for example if you have an early career member of the team who kicks butt at styling but wants to be more full-stack, maybe you paired them with a senior engineer who has more of a back-end focus). Or if your decision is technical in nature, explain the trade-offs in performance and timeline that you considered before deciding on the solution you did. Make sure your team understands the context behind decisions so they can integrate that into future decisions they may have to make without you.



Delivery / People

Show your process

@mercedescodes | mercedesbernard.com

Related to decision making and sharing context, make sure to show your process to the team.

When deciding between possible solution architectures, share the diagrams and strategies you didn't choose and why they didn't make sense for the team's current goals and constraints.

If you had to create an estimate for an upcoming project, show the roadmap iterations to the team and explain how stakeholder priority overlaps with engineering priority and how you have to balance both in order to come up with a timeline and project plan that works for all parties involved.

And if you are involved in setting team or organizational strategy and goals, make sure you're transparent with the team and explain where these things came from. Are they revenue based, customer experience based, do they support your hiring strategy? How do they fit into the long-term vision? What is the plan to achieve the goals? Be as detailed as possible. Paint a picture for the future so that your team members feel connected and ready to go on that ride with you and the company. And if you do choose to leave, they won't feel lost because they know the purpose and the plan.

People

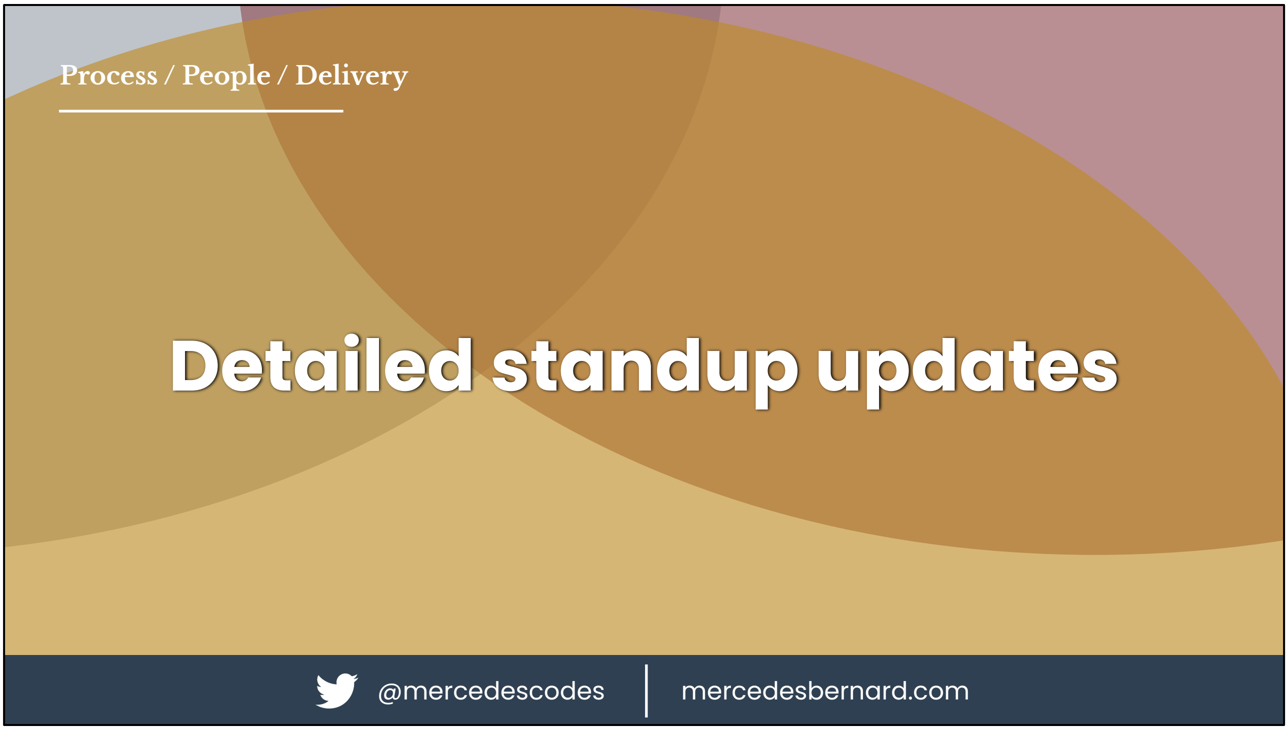
Building team credibility and stakeholder trust

@mercedescodes | mercedesbernard.com

Firmly in supporting people, we have building team credibility and stakeholder trust.

It can be tempting to want to take credit for good work and we all enjoy a shout out from a stakeholder about how they appreciate our contributions. But this can create dependence on you as an individual. As a consultant, I have a lot of experience with clients valuing my individual work and feeling as though I'm needed for a project to succeed. This is obviously very flattering but if I'm doing my job right, I'm enabling the client team to be successful with or without me. I'm giving them tools they may not have and helping them solve problems so that when I leave, they can continue to succeed. If you are a senior+ engineer or in any leadership role within your team, you should have the same goal.

When you're talking to stakeholders, make sure to shout out other individuals on the team for their vital and important work. Recognize and identify their strengths so that stakeholders know what everyone is good at. Share credit. Because of your position on the team you are granted a level of trust and credibility from your stakeholders by default. The best thing you can do is lend that credibility to your team members and help build that trust so that when you leave, your stakeholders don't feel like the rug got pulled out but instead know that they're in good hands and the team is going to continue to deliver the same work they have been. By doing this, you set your team up to be able to continue to make decisions without being challenged by stakeholders because they'll already have confidence in the team which is a better experience for everyone.

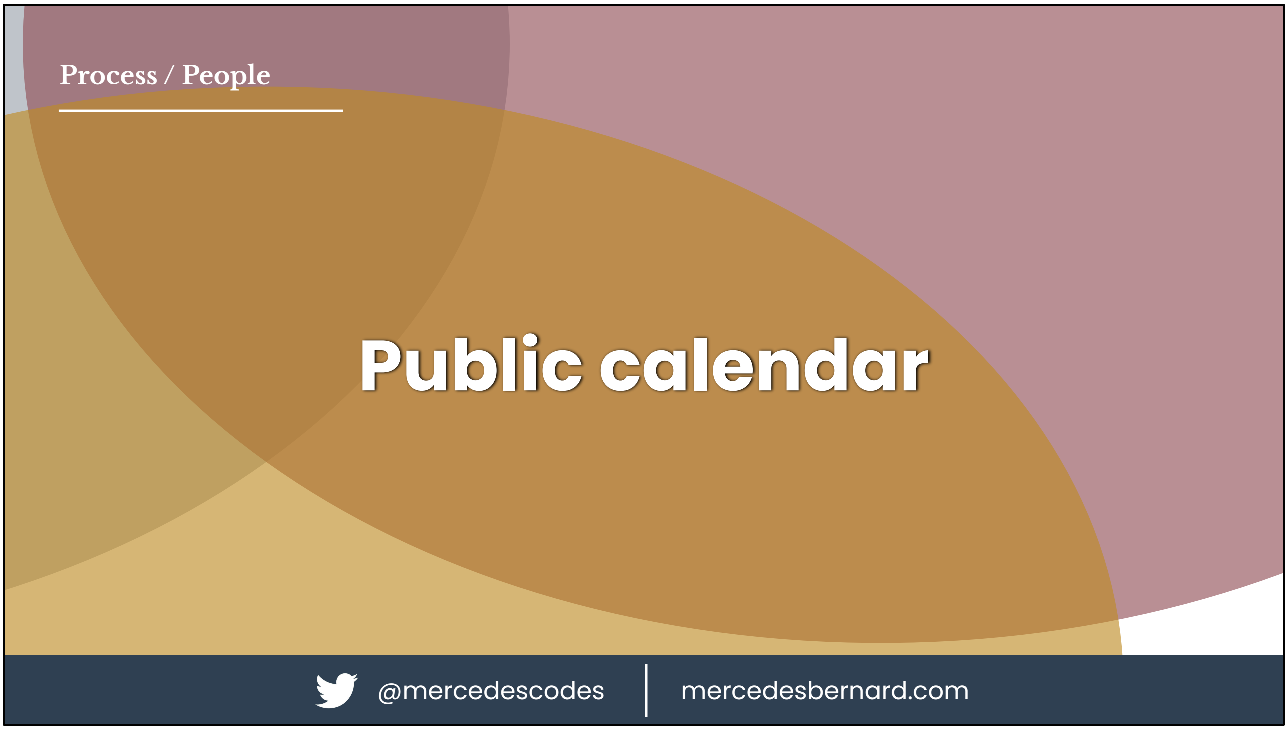


Process / People / Delivery

Detailed standup updates

@mercedescodes | mercedesbernard.com

Now we're getting into the individual interventions you can take in the day-to-day processes on your team. One low-hanging fruit is to start giving more detailed standup updates. I don't mean you should takeover the whole meeting. You should still keep your update concise but with the goal of increasing awareness of what you're working on. Instead of just saying what ticket you're working on, describe what you've implemented so far, what you got stuck on, what you've investigated and learned, the types of problems you're solving. The goal is to increase team context as much as possible. Who knows? 2 weeks after you leave, there may be a bug in your code and someone will remember something you mentioned in standup that leads them to a fix.



Process / People

Public calendar

@mercedescodes | mercedesbernard.com

Politics is the set of activities that are associated with making decisions in groups, or other forms of power relations between individuals, such as the distribution of resources or status. And calendars are a simple way to share political information with your team. Who is making the decisions, who is being consulted, what is being decided, etc.

I coach all my team members and direct reports that one of the best ways to know what's going on within a company is to keep an eye on calendars. Knowing what people in leadership are spending their time on gives you a lot of information about the state of the company and company goals. So as a senior member of the team, I encourage you to open your calendar as much as possible. Go ahead and keep your personal appointments private but make as much public as you possibly can and encourage people to check your calendar for what's going on and ask questions if there's anything they'd like to know more about.

By being open with your team, you are increasing their political awareness and creating a psychologically safe environment for them to ask questions and be better informed. Well-informed teams are resilient teams.

You also want to increase visibility of how you spend your day as much as possible. The more folks know what you're doing, the more prepared they will be for what may change or need to get picked up if you leave. And the more they know, the more they'll be able to express interest if there's something they'd like to get involved with giving you the opportunity to sponsor someone as well.

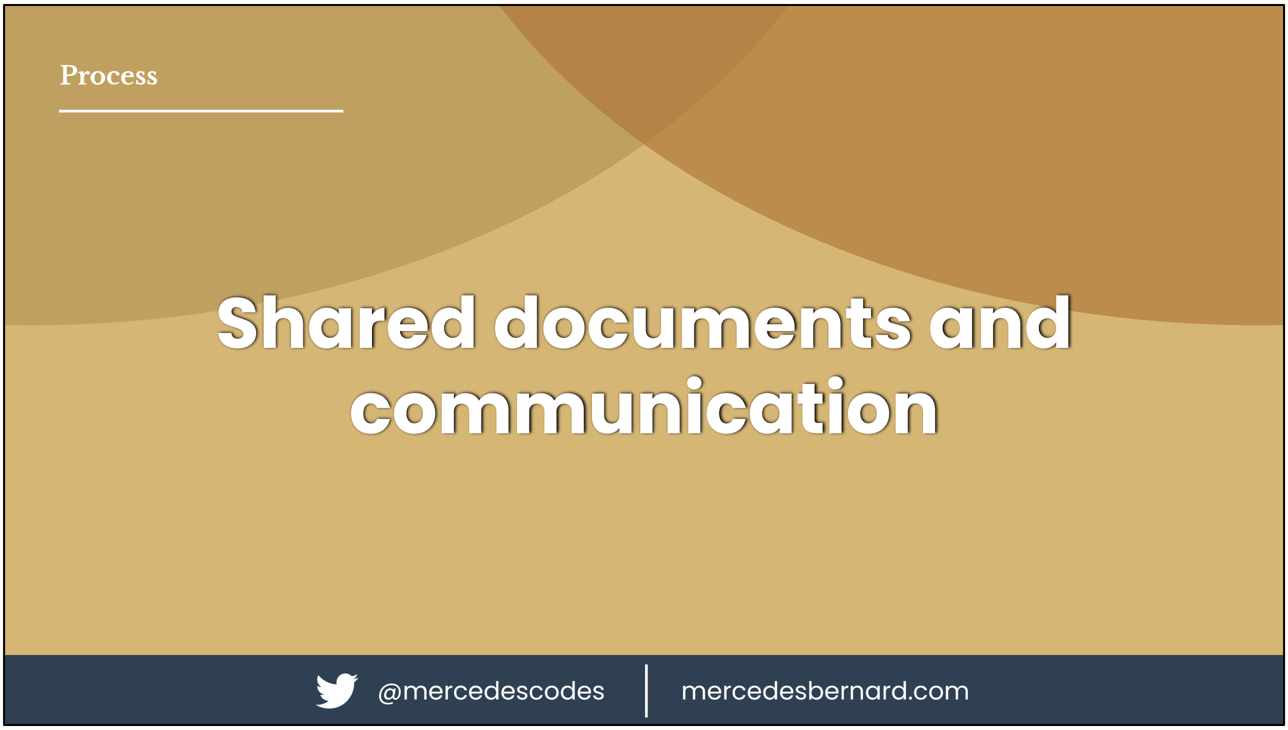
Process

Shared credentials and config

@mercedescodes | mercedesbernard.com

These next couple strategies are straightforward but often forgot about until it’s too late. Make sure that you are not the sole owner of any company accounts or tools. Use 1Password (or some other password management tool) liberally to share passwords and make sure to use company email addresses for everything so that if you leave suddenly, the team is not locked out of any vital tools they need for their workflow.

Make sure critical env vars are documented in your 1Password as a backup so that they can be shared among the team (this is especially helpful if your CI tool obfuscates env vars). Your env vars shouldn’t be checked into source control anyway so even if there’s no turnover, this is still a good idea so you have a source of truth in case everyone’s laptops brick over night.



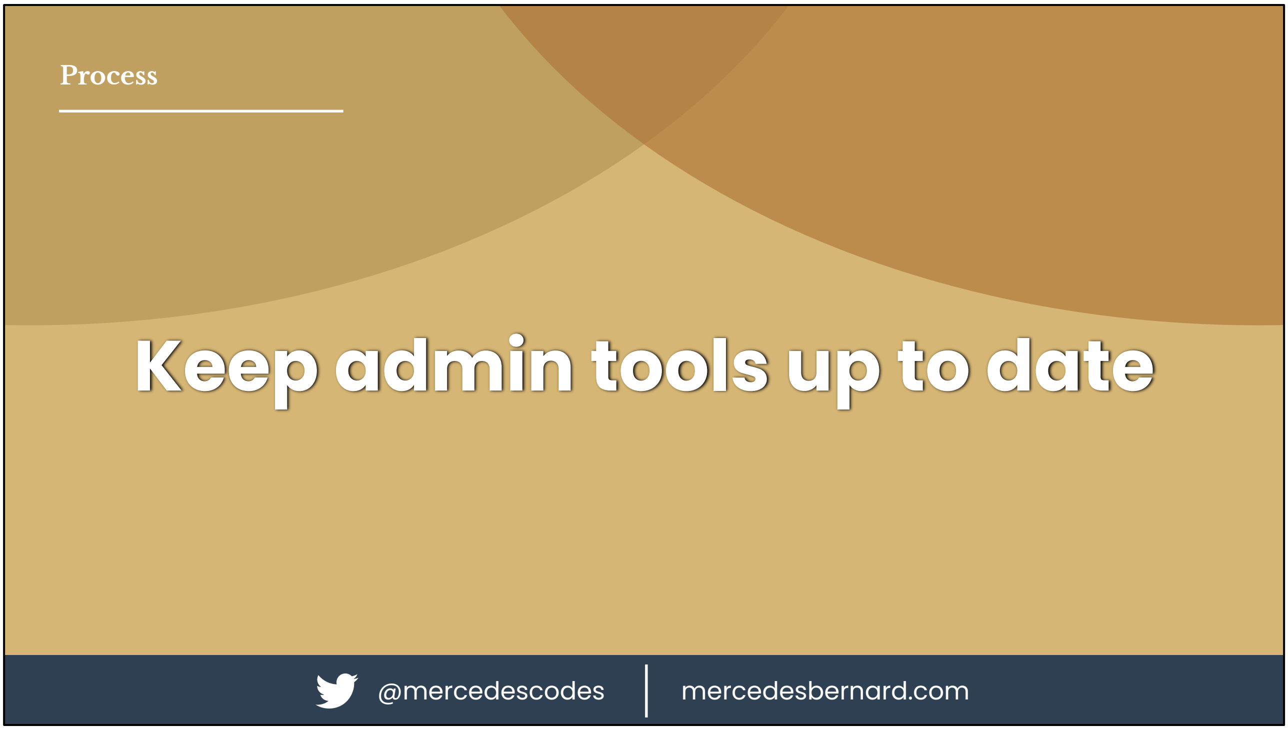
Process

Shared documents and communication

@mercedescodes | mercedesbernard.com

Similarly, clean out your personal document files. If you’re using Google Drive or Dropbox or some other file sharing tool, try to keep files out of your personal folders as much as possible and in shared spaces where everyone on the team has access. This can include things like technical docs such as architecture diagrams or requirements docs or less “official” things like past lunch & learn presentations and team bonding activities.

Try to have as many conversations in public Slack channels as possible. Include your team members in emails with stakeholders. After you leave, at least there will be a searchable record of your past decisions and conversations that may help others who have to step into your shoes.



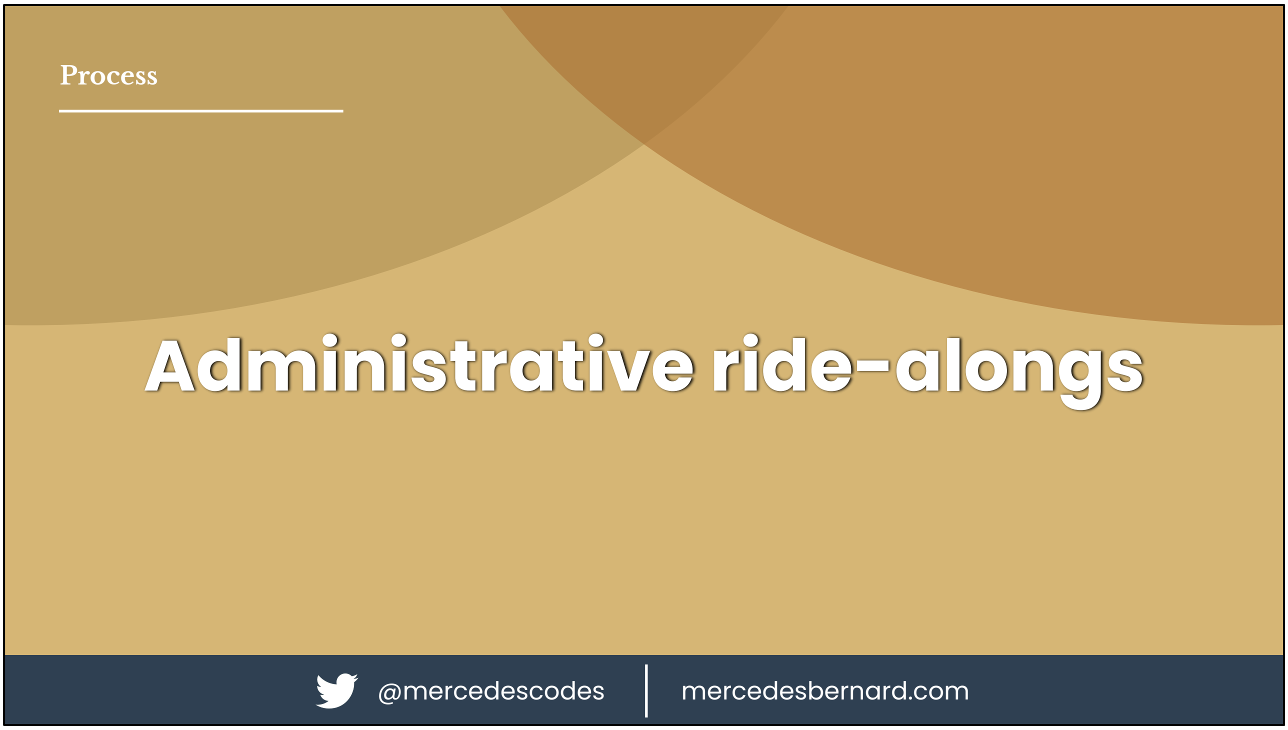
Process

Keep admin tools up to date

@mercedescodes | mercedesbernard.com

You can also help support them after you leave by keeping all of your tools up to date. This will vary depending on your role and responsibilities but could include your applicant tracking system and job postings, onboarding resources for new team members who may join, performance management tools, incident response tooling, project management tools like Jira, etc.

The less your team will have to stress about and figure out when you leave, the more successful they can be at continuing to deliver and achieve their goals. For example, if you keep the job postings updated during your time with the company (if that falls under your purview, i.e. you’re an EM) then it won’t fall on someone to figure out what the team needs, write a job posting, and publish it to backfill your position. They can just click the publish button and continue on with their work.



Process

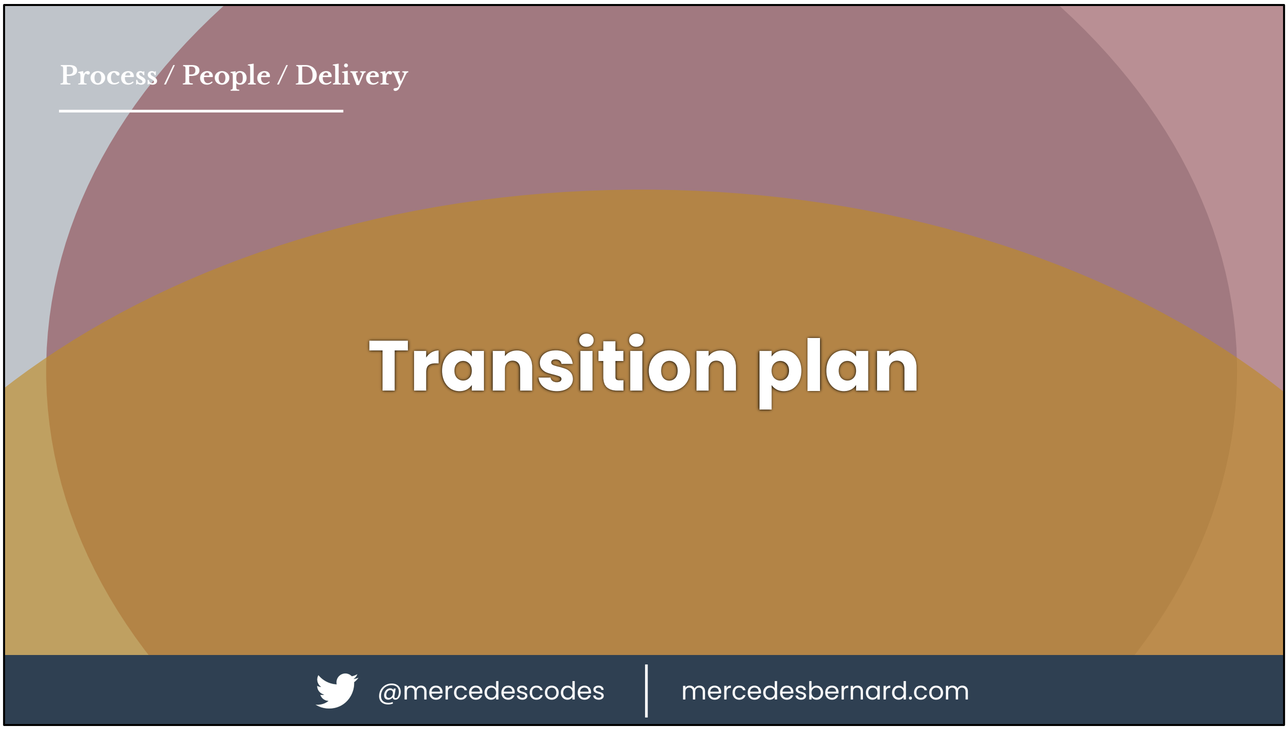
Administrative ride-alongs

@mercedescodes | mercedesbernard.com

Related to keeping your admin tools updated is increasing visibility into your administrative tasks and making sure that folks on your team know how to do them.

If you have to do something non-technical but that is vital to the team functioning, such as doing an initial phone screen for a candidate, sending out a project update to stakeholders, checking PTO calendars to make the upcoming pair rotation schedule, or whatever it may be, see if you can find anyone interested in pairing or doing a ride-along so you can show them what goes into these tasks. Often, senior engineers have some administrative responsibilities that the rest of the team doesn't have but that means the team may also lack visibility into how much work goes into those tasks. When you leave, the team may be caught by surprise by how much time these things take or without the awareness that they even need to be done. If they know about them, they can plan for them.

There may even be the ability to spread the responsibility around. Something like candidate phone screens is a great responsibility to share so that everyone gets to share their perspective in hiring discussions and team building. But even something that is "busy work" is good for the team to be aware of even if it's not a shared responsibility.



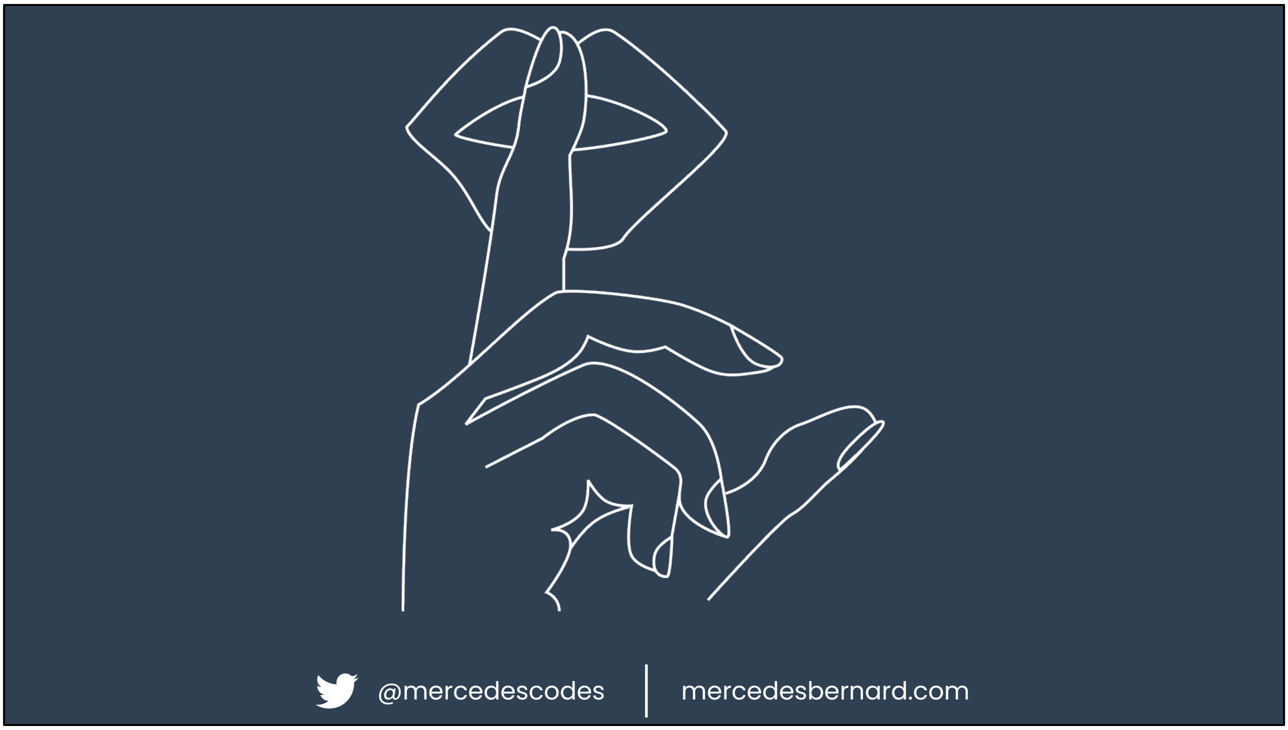
Process / People / Delivery

Transition plan

@mercedescodes | mercedesbernard.com

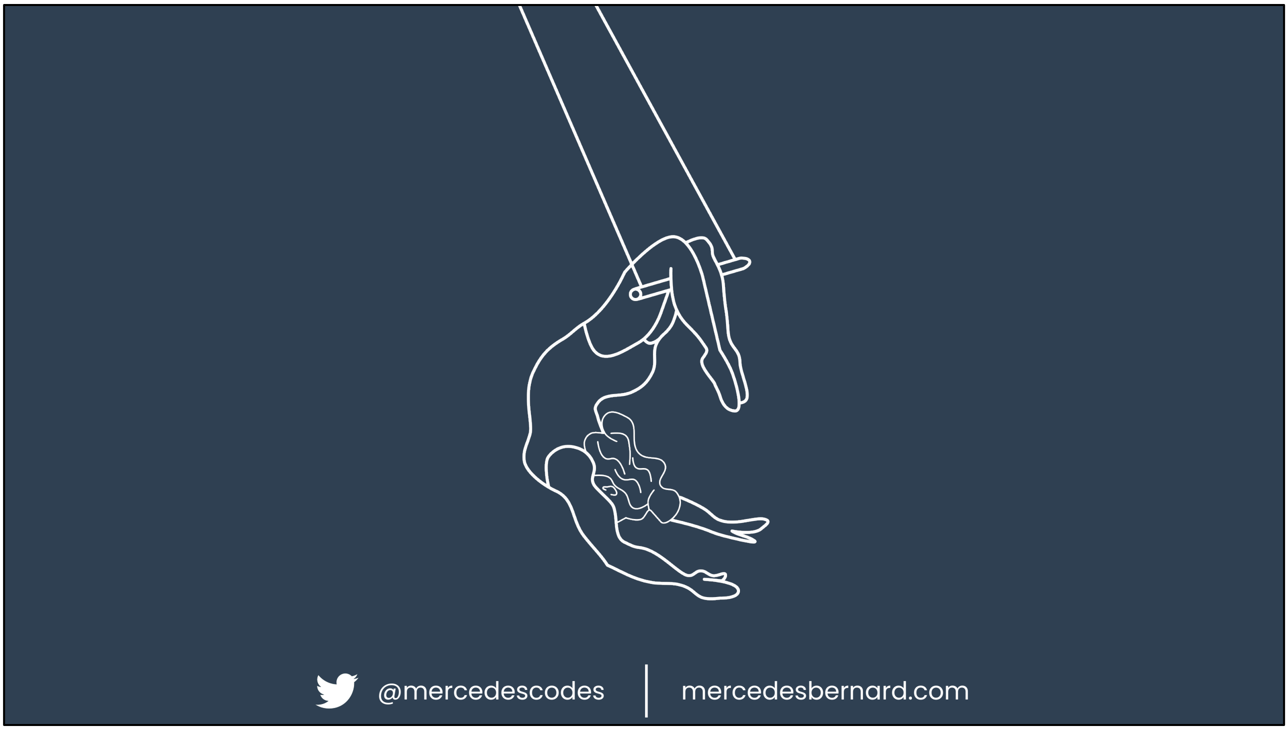
The last thing you can try encompasses delivery, people, and process. That's to think about a transition plan. Unfortunately, incompetent leaders will dump your workload on to someone else and leave them to figure it out without much support. But you can help support your team by thinking through what the next steps should be if and when you decide to leave.

- Who could and should be promoted to fill your shoes? Who has expressed interest in the role or is ready for the opportunity? Start having conversations with their manager early to give the feedback that they're ready for a promotion and should be given more opportunity
- What role/roles should be opened up to backfill your position? Make sure those job postings are up to date ;) (again, if that is already a responsibility of yours)
- Are there any internal initiatives that you're leading that you should share or pass on to someone else? Do you know anyone interested in helping you with those initiatives? How can you get them involved before you leave so that it's an easier transition?
- And finally, is there any valuable feedback you want to provide for someone's upcoming performance review to help support them in pursuit of a promotion? Talk to their manager. See if you can give that feedback now while it's fresh or if you can be included in their peer reviews.



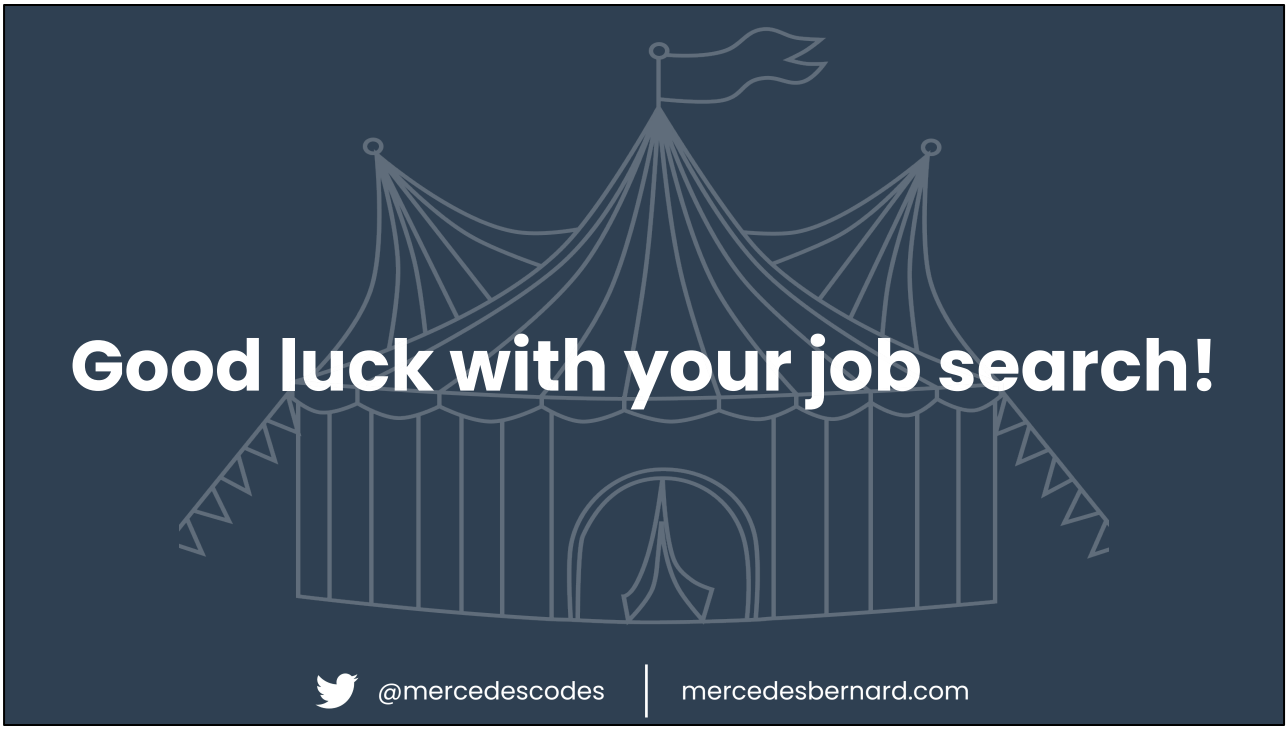
@mercedescodes | mercedesbernard.com

Unfortunately, it is still taboo in most companies and on most teams to talk about wanting to leave a job. There is still an unequal power dynamic that makes many people afraid to mention that they’re job-hunting for fear that it will jeopardize their current role before they have another lined up. This is starting to change, especially in tech, but is far from going away. So while you may not be able to be open with everyone you work with, there are lots of low-friction, non-obvious ways you can start to ease your transition off a team and out of a company.



@mercedescodes | mercedesbernard.com

If you have dreams of flying trapeze or swallowing swords under a big top, you can minimize the stress on the humans you care about with small, individual interventions. And if you are content in your current role, you can still make these investments in your team culture and improve resilience because turnover is a fact of life in the tech industry regardless if you’re leaving or someone else is. A resilient team is always an enjoyable one to work on.



Good luck with your job search!

@mercedescodes | mercedesbernard.com

And if you are planning on leaving, I wish you all the best with your job search!!