

A dev's guide to burnout



Contents

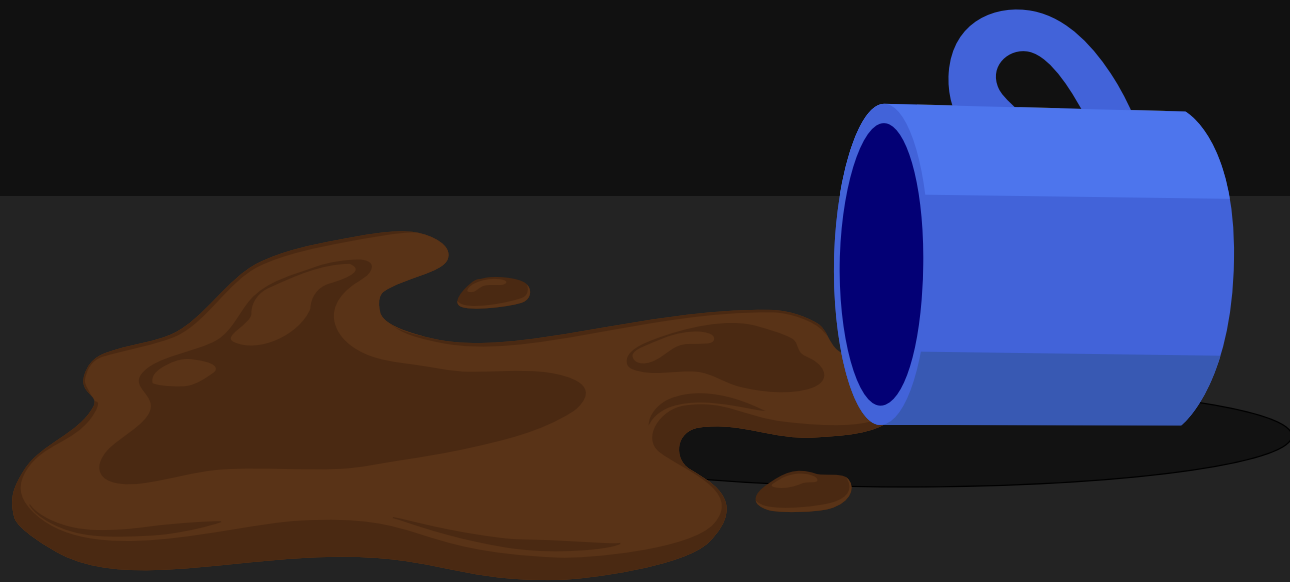
What is burnout? 3

Common causes of developer burnout 7

How to respond to burnout 15

Conclusion 23

What is burnout?



Some days, your heart isn't in your work. You find yourself staring out the window, returning to the same line of code several times before accepting that it's just not happening today.

If a good night's sleep and a cup of coffee can restore your focus, that's just Wednesday. If the struggle continues for weeks, that might be something deeper: **burnout**.

Burnout feels like a total disconnect from your work. It is where fatigue meets apathy—and it takes intentional effort to recover. To find a silver lining, there's a saying that **"you can't burn out if you aren't on fire."** Burnout is possible because you care. The goal is to keep the fire burning, without pushing beyond your limits.

While burnout is often felt by an **individual**, it results from decisions by the **organization**. In this e-book, we'll consider common causes of burnout and ways that developers can effectively respond at the personal level, plus additional strategies to support effective conversations with your manager.



World Health Organization definition of burnout



Burn-out is a syndrome conceptualized as resulting from **chronic workplace stress that has not been successfully managed**. It is characterized by three dimensions:

- feelings of energy depletion or exhaustion
- increased mental distance from one's job, or feelings of negativism or cynicism related to one's job
- reduced professional efficacy



Three stages of burnout

The typical burnout experience can reach three stages, getting progressively more difficult to reverse at each stage.



1 Exhaustion

First, you reach a deep level of depletion—while still managing your workload.

➤ *Response is most effective when started at this early stage.*

Sounds like: A weary sigh as you open Jira. And Slack. And Zoom.

2 Cynicism

Next, you might create distance from the burnout, which appears as apathy.

Sounds like: “I don’t care what we pick. This doesn’t matter anyway.”

3 Inefficacy

Finally, this is when you will simply detach.

➤ *Burnout is hardest to remedy once it has reached this third stage.*

Sounds like: “That’s not my job.” [Opens LinkedIn.]





Common causes of developer burnout

When we talk to software developers about burnout, these issues are frequently contributing factors.

Low Deep Work

We describe Deep Work as 2+ hours of uninterrupted working time (and a cornerstone of engineering effectiveness)

In today's "ping culture" of frequent Slack messages and impromptu huddles, uninterrupted focus time can seem like an elusive dream, rather than a typical workday.

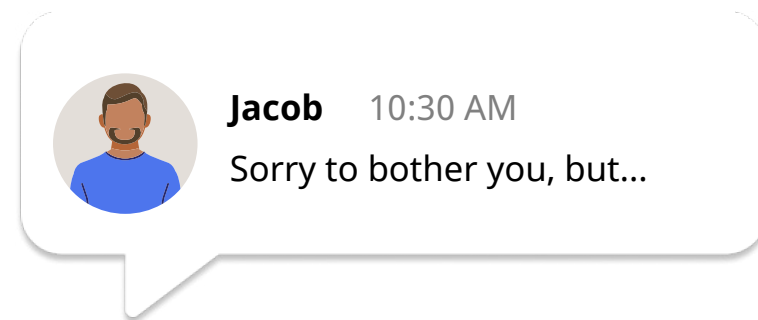
We often hear developers say they get caught up in meetings and short work blocks all day, then need to catch up on pull requests and DMs in the evening or weekend to unblock their teammates.

Uplevel data shows that when engineers have sufficient time to focus, they have less of a need to work overtime.



Frequent Interruptions

There's a unique relief to a meeting-free day with a single task at hand. Finally, you have time to focus. Right as the code starts flowing, you receive an urgent Slack message.



You complete the requested task and return to your code. It takes a few minutes to get back in the groove. What seemed like a minor interruption is now responsible for stealing twenty minutes of your Deep Work time.

This is a classic example of the energy drain that occurs when changing your focus from one thing to another.

How can I reduce interruptions?

Work with your manager and team to adjust scheduling and work allocation.

- Run a calendar audit to remove low-value meetings
- Consider a standard time or day for most meetings
- Implement a meeting-free day
- Schedule biweekly Deep Work blocks for the whole team
- Set Slack to "Do Not Disturb" during Deep Work or pre-set daily windows
- Evenly distribute bug work or assign on-call days to reduce randomization
- Reduce the number of Jira epics assigned to each person
- Reduce the number of PR repos assigned to each person



Too many meetings

As helpful as they can be, meetings often break up the workday into unhelpful pieces. It can take 20 minutes to shift from Deep Work into meeting mode, and another 20 minutes to really sink back into focus afterward.

Oh yeah, and then there's Zoom fatigue

Back-to-back meetings are tough, and moving from one video call to the next can be uniquely draining.



What you can do

- When screen sharing isn't necessary, try a good old fashioned phone call to take your eyes off the screen or opt for a walking meeting.
- Tough love: stop multitasking. Answering emails and Slack messages on the side can accelerate fatigue, so put on Do Not Disturb and full-screen mode to avoid distractions.
- Refresh your desk with anything that sparks joy: plants, better lighting, action figures, a Nespresso...

What your team can do

- Truncate 60-minute meetings to 50, and 30-minute meetings to 25. This frees time in the schedule to get water or stretch.
- Try asynchronous meetings, which break up screen time by meeting for 15-minute to set an intention, breaking away to do solo work, then meeting up again for a 30-minute collaboration. Repeat the cycle as many times as needed.
- Establish a meeting-free day. Simply said, distance makes the heart grow fonder.



Ineffective sprint retros

Retros are designed to continuously improve sprint health. When change comes painfully slow—or not at all—it’s time for your team to work together on a refreshed retrospective agenda.

How can I improve sprint retros?

These tips give more depth and breadth to your retrospective for stronger outcomes.



What you can do

- Collect your feedback during the sprint, not just after. Record notes on specific Jira tickets, process failures, and red flags as they happen.
- Highlight items that were added mid-sprint. Any additional work—no matter how “simple”—can derail a sprint, making this review critical to inform future sprint planning.

What your team can do

- Look at Project Health: How was your cycle time? Were any PRs merged before being approved? Do you have the right reviewers assigned? Are people remembering to include story points and descriptions on Jira tickets?
- Look at People Health: Did the team complete the sprint at the expense of high burnout? Did devs work long hours and on weekends? Were they context-switching at unsustainable rates?



How to respond to burnout

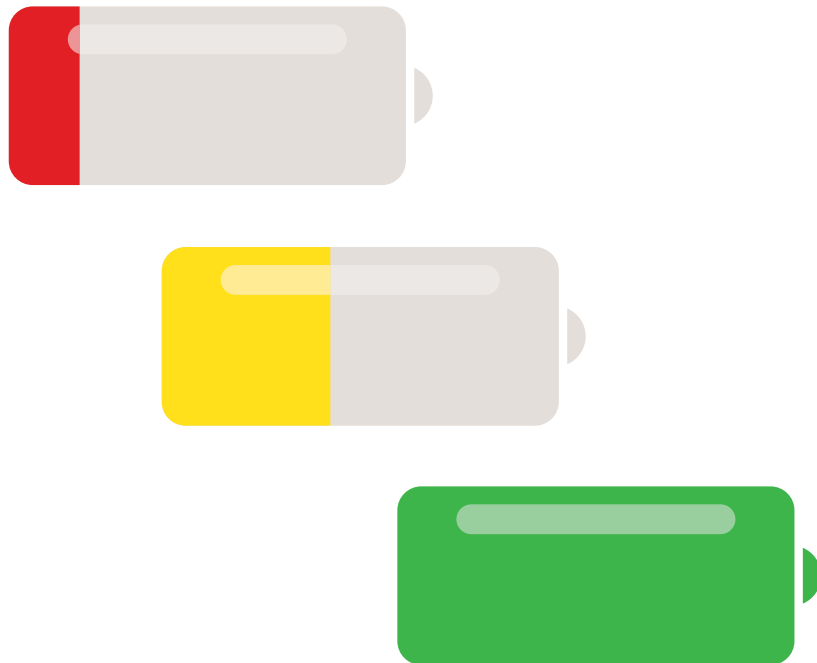


What should you do if you are nearing burnout?

First up, schedule one-on-one time with your manager. The priority is to discover the “why” behind the burnout. Are you tired of unrealistic deadlines? Has this quarter held an unusual amount of work? Too many epics?

The next key is to understand which stage of burnout you’re in. If you’re feeling exhaustion, you might benefit from a mental health day or load-balancing work with your teammates. When you’re already feeling inefficacy, you might want a bigger conversation about your role and purpose.





How can you recover from burnout?

First, reflect on needs: What energizes you? Depletes you? What do you need to feel recognized? How is your role—and how are you, personally—connected to the mission?

Think beyond work. Is your free time lacking creative projects? Break out the crafting supplies. Spending all your time inside? Add a lunchtime walk. Always on your feet? Cue Netflix. Understanding what's missing can help you find balance.

Then, reflect on your mindset. Shifting the thought process from “I have so much work to do” to “we have a problem to solve together” can bring a sense of purpose to your workload.

Three types of recovery

Different forms of rest can counterbalance different feelings of disengagement.

Detachment: Take a genuine break. Have you ever planned a vacation, only to stress up until the moment you leave the office and answer “urgent” emails from the Airbnb? Detachment requires true separation from the source of stress. This could also be achieved by logging off early this Friday and setting your phone to Do Not Disturb mode.

Relaxation: Do a leisurely, low-effort activity, like playing cards or cooking a meal. Engage your brain just enough to forget about work, but not so much that you're challenged.

Mastery: When work is draining, it can feel revitalizing to reconnect with your other skills. A fantastic way to recover is to do something you're great at. Bake your famous sourdough bread, work on your macrame, or get gaming.

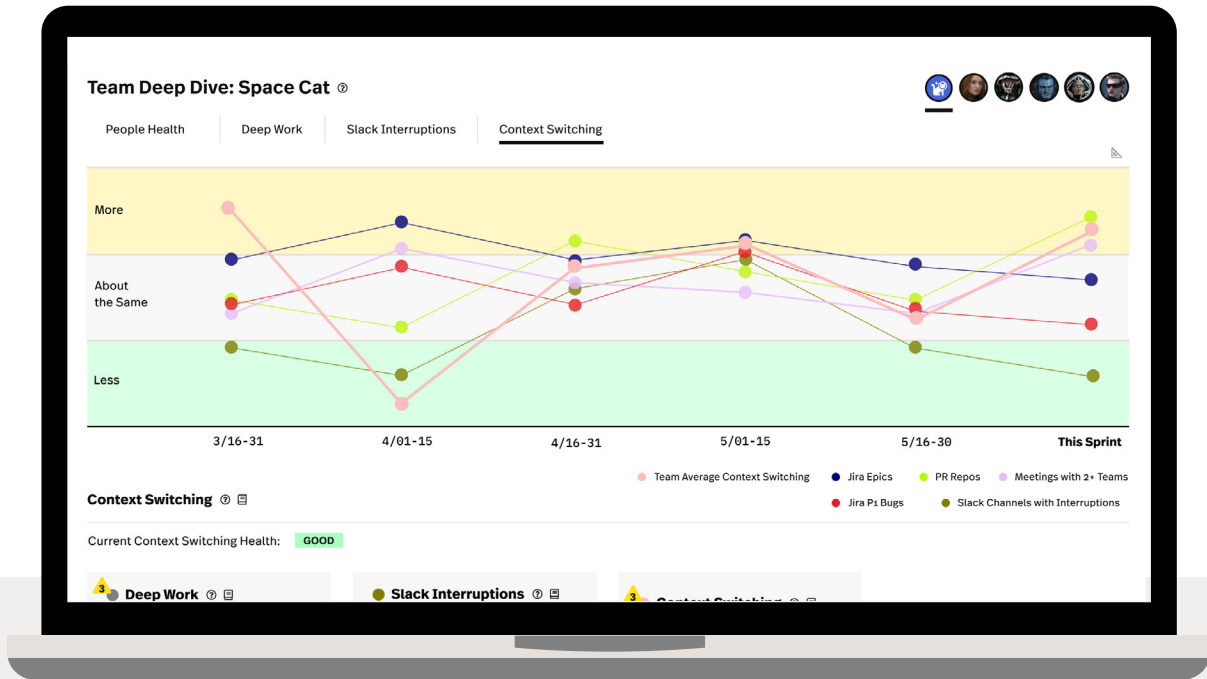


How to share concerns with your manager using data support from Uplevel

When you're facing inconveniences, you can provide data to show their impact. A tool like Uplevel can give structure to your feedback and experience.

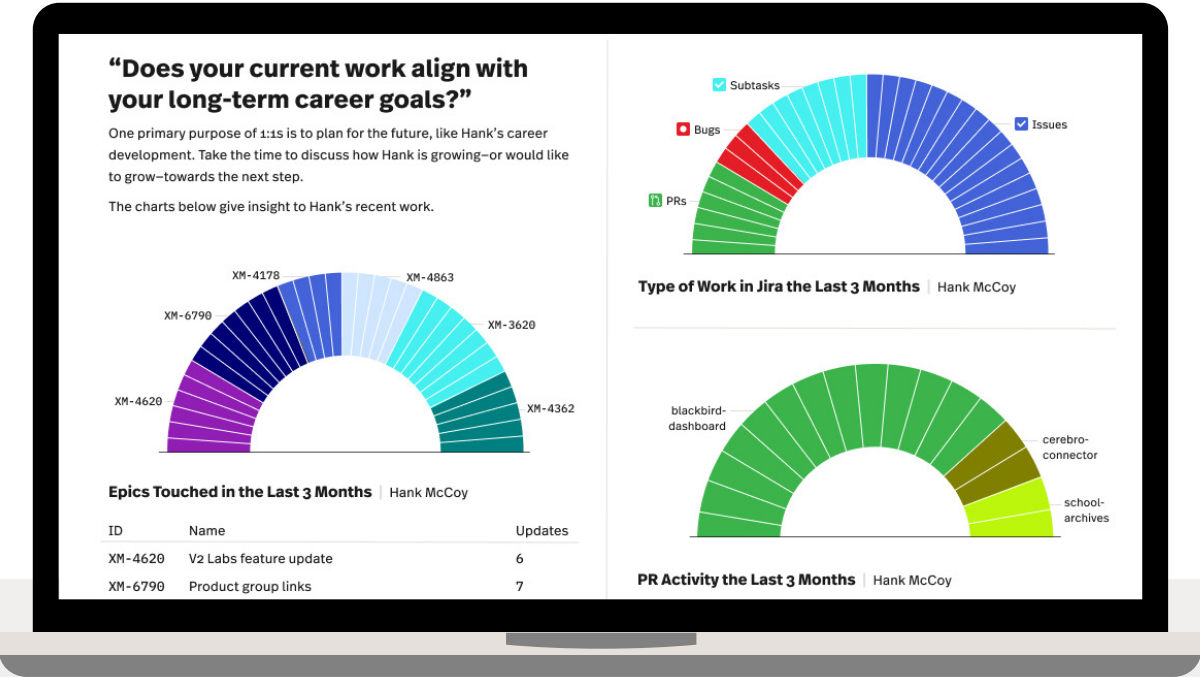
“I have too much work.”

This is a common concern and a hard one to communicate—it's helpful to drill down into what exactly makes it “too much.” Too many tasks? Consider how your Context Switching has trended over time. Juggling simultaneous complex tasks? Look at your Deep Work metrics, which indicate whether you have sufficient time to dig below the surface.



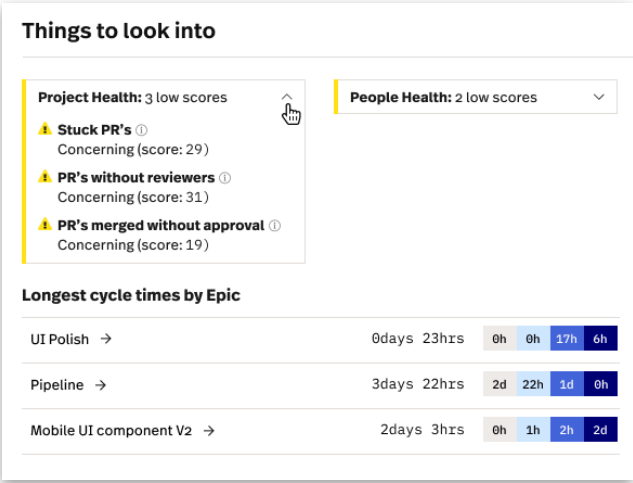
“I want to build [x] skills so I can grow my career.”

When you’re ready to expand into a new skill set, bring out your work allocation from the past three months. (Uplevel categorizes it by epic, work type, or pull request, for clarity.) This shows where you’ve been spending time and energy, so you can discuss where things might change.

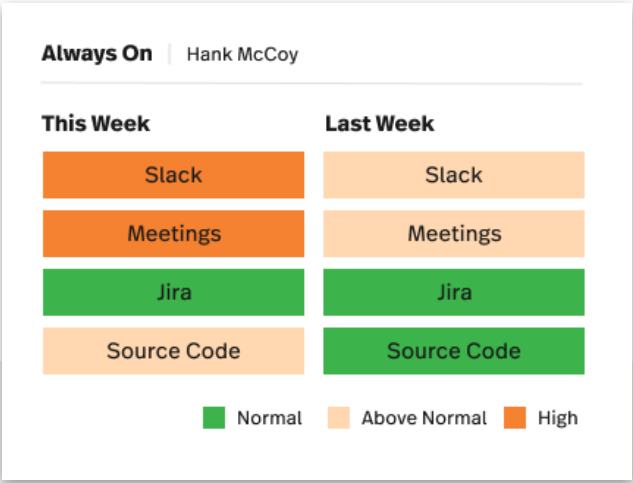


“This sprint was messy.”

When the sprint retro comes around, you might struggle to identify the specific aspects that made it difficult. (Sleep debt is real.) Bring in data, like the percentage of goals completed, prevalence of agile best practices, PR cycle time, and qualitative feedback from the team.



“Work-life balance feels fictional.”



Given the blurred lines between work and home, you might feel “always on.” Bring in indicators of overtime, like how frequently you’ve had to check in after-hours or during weekends. (We don’t glorify long hours.)





Conclusion

Burnout is a natural, yet unfortunate response to extended periods of unmanaged stress.

Purposeful awareness and the right tools can help you remedy work-related fatigue before your flame goes out entirely.



A quick guide to burnout recovery

- Check in with yourself about burnout regularly. Are you at the stage of exhaustion, cynicism, or inefficacy?
- Explore potential causes. Are you experiencing low Deep Work, frequent interruptions, too many meetings, ineffective sprint retros, or other issues?
- Rest and respond as needed. Consider activities that offer detachment, relaxation, or mastery.
- Bring data to support a discussion with your manager and team. Look at work allocation, desired skill set expansion, problematic sprint issues, work-life balance, and other concerns.
- Collaborate across the organization to recover and put new prevention measures in place.

Further resources

Ready to refresh your work life? Sharpen your approach to [weekly 1:1s](#), [career development conversations](#), and [addressing common pain points](#).

Uplevel is your engineering insights solution that leverages data from everyday developer tools to help you reliably meet your sprint goals without burning out your team.

Schedule a demo today.

uplevelteam.com

