
Avoid burn out

Quality of life

- Quality of life discussions are a thing in the industry for a reason
- There have been quite a few terrible examples in the industry's history
 - EA: The Human Story (a spouse) (2004)

“The current mandatory hours are 9am to 10pm-- seven days a week-- with the occasional Saturday evening off for good behavior (at 6:30pm). This averages out to an eight and a half hour work week. Complaints that these once more extended hours combined with the team's existing fatigue would result in a greater number of mistakes made and an even greater amount of wasted energy were ignored.”

EA: The Human Story

My significant other works for Electronic Arts, and I'm what you might call a disgruntled spouse.

Burn out

- Hours can be high in the games industry
 - During regular hours
 - 54% worked 40-44 hours per week
 - 14% worked 45-49 hours per week
 - 5% worked 50-59 hours per week
 - During crunch
 - 38% worked 50-59 hours
 - 19% 60-69 hours per week
 - 13% more the 70 per week
 - 41% said crunch time was expected

*IGDA Developer Satisfaction Survey 2019

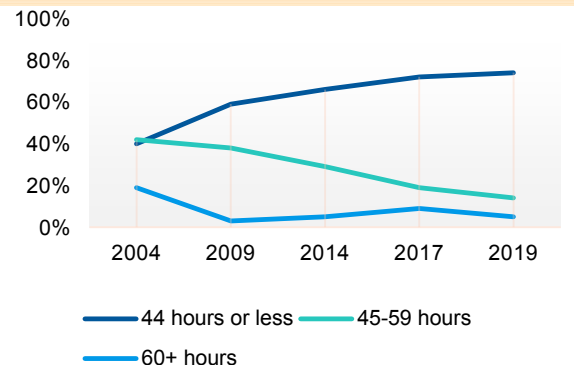


Burn out

- Hours have gotten better over time

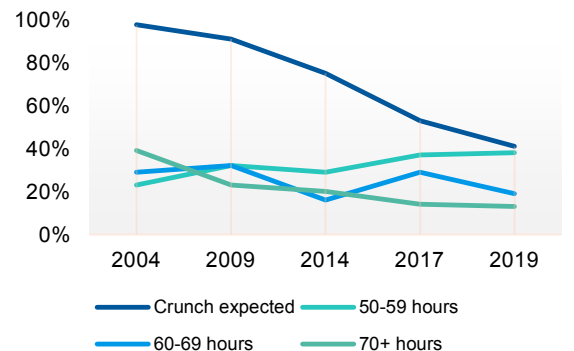
Regular Hours

2004	2009	2014	2017	2019	
40%	59%	66%	72%	74%	44 hours or less
42%	38%	29%	19%	14%	45-59 hours
19%	3%	5%	9%	5%	60+ hours



Crunch Hours

2004	2009	2014	2017	2019	
97.6%	91%	75%	53%	41%	Crunch expected
23%	32%	29%	37%	38%	50-59 hours
29%	32%	16%	29%	19%	60-69 hours
39%	23%	20%	14%	13%	70+ hours



Be in a state to work

- Stay in a state where you are able to work
 - Get sleep
 - Eat
 - Take breaks when you need them
 - Maintain humanity outside of development



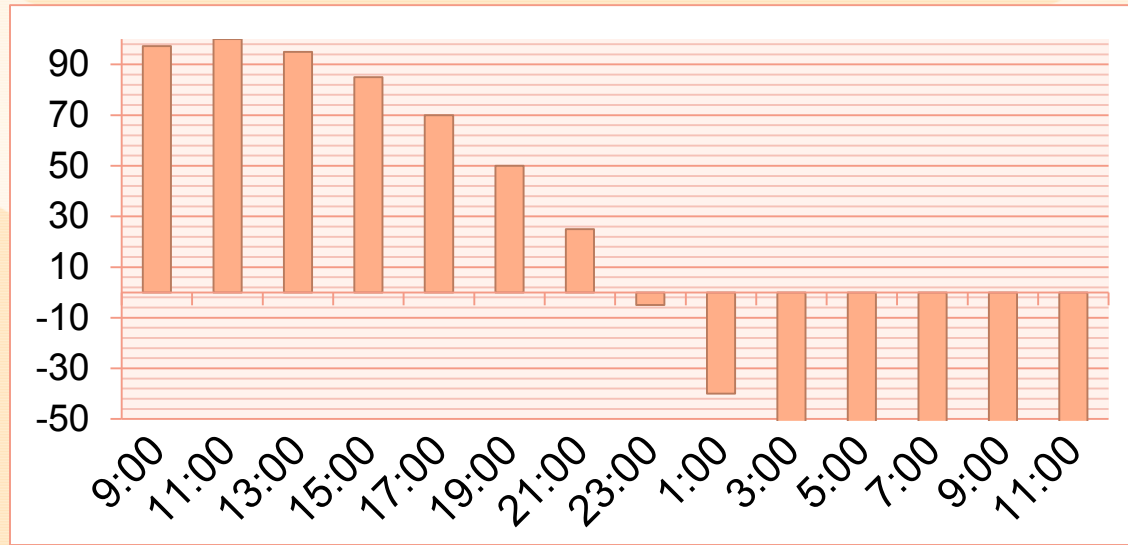
Avoid burn out

- You need more than just programming and red bull



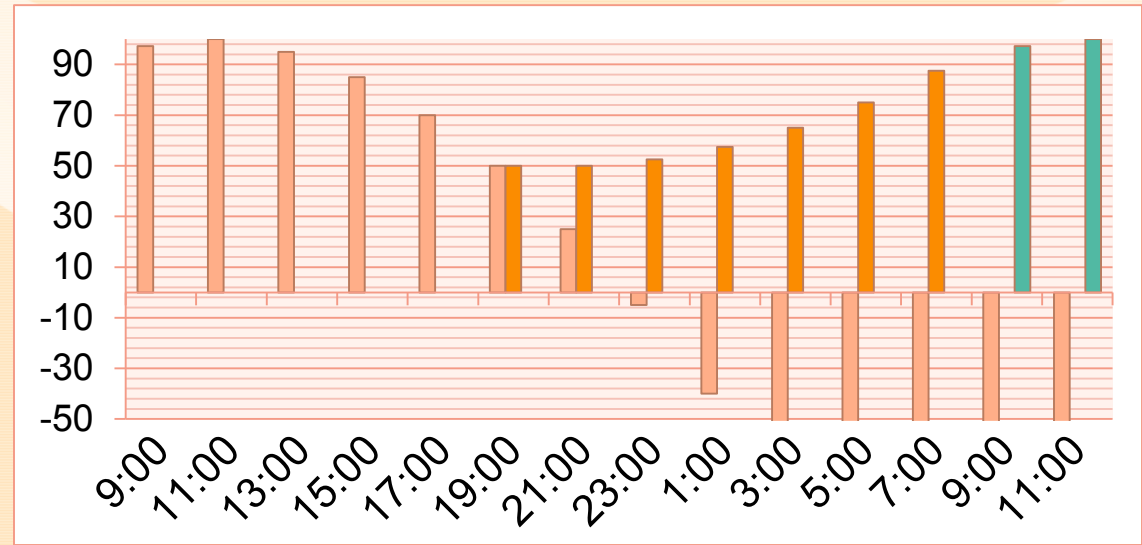
Daily productivity

- Productivity varies over the course of the day
 - The greatest productivity occurs in the first 4 - 6 hours
 - After enough time working, productivity approaches zero
 - Eventually productivity becomes negative



Daily productivity

- You will be more productive if you allow yourself to take breaks
 - Let your mind rest periodically though the day
 - Stop working at the end of the day
 - Sleep!



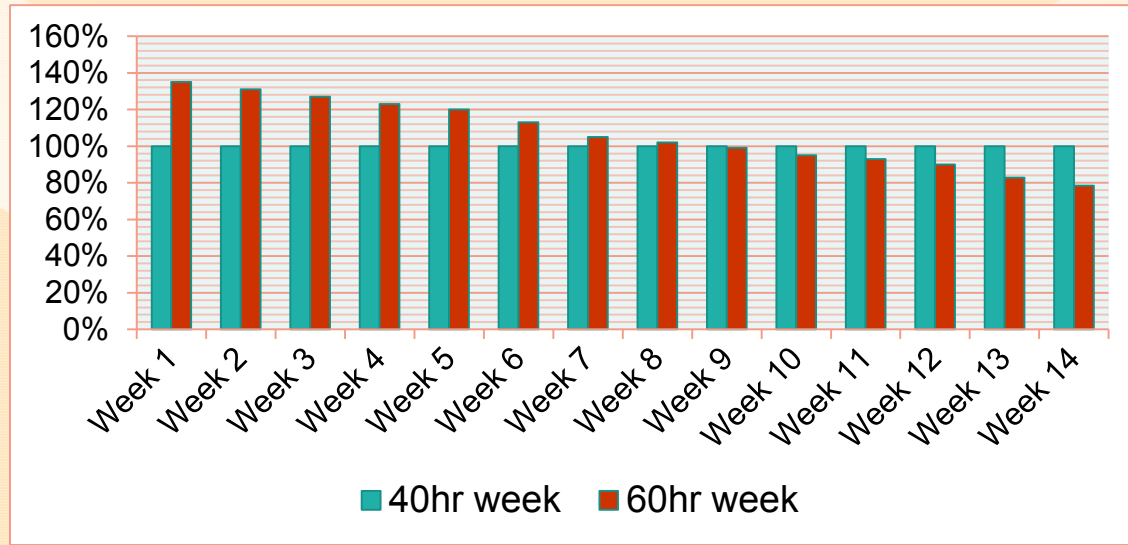
When to take a break and come back

- There is a significant difference between being in the groove and continuing work and pushing through exhaustion.
- Sticking with a Groove is valuable.
- Exhaustion is tanking your quality of work.

Long term productivity

So why is crunch so common?

- Crunch can lead to short term gains
- However continued crunch produces less than a standard 40 week in a couple months



Additional Resources

- Evan Robinson: Why Crunch Modes Doesn't Work —Six Lessons
 - <http://www.igda.org/?page=crunchsixlessons>
- EA: The Human Story (a spouse) (2004)
 - <http://ea-spouse.livejournal.com/274.html>
- IGDA Developer Satisfaction Survey (DSS)
 - <https://igda.org/dss/>



home



about



news



events



community



advocacy



resources



get involved

Why Crunch Modes Doesn't Work: Six Lessons

There's a bottom-line reason most industries gave up crunch mode over 75 years ago:
It's the single most expensive way there is to get the work done.

by Evan Robinson

Executive Summary

When used long-term, Crunch Mode slows development and creates more bugs when compared with 40-hour weeks.

More than a century of studies show that long-term useful worker output is maximized near a five-day, 40-hour workweek. Productivity drops immediately upon starting overtime and

Prepping for Sprint Review

(end of lecture portion)

Alpha Sprint Goals

- The game world is populated with at least one example of each game object
 - All game object types
- Game progress can be shown
 - Multiple levels, multiple objectives, persistent data saved between levels...
 - Ability to progress. Not final level designs
- The game contains its win/loss conditions
- Cleared Technical debt
- Feature complete
 - Enough asset creation complete to prove the use of features
- Any feature not completed by the end of this sprint must be cut from the product

Code Freeze

Code Freeze

A time where no changes are permitted to the codebase to allow for integration and ensure the quality and integrity of the build

Code freeze To Dos

- Did the entire team meet for final integration?
- Does each team member have the same version of the game on their computers?
 - All work committed and merged
 - Each local copy fully updated
- All work has been confirmed to have been integrated
 - Peer checks on userstories completed
- Was a build exported and shared with the team ?
 - In the "Milestone build" folder
- Did each team member complete a play through of the build and signed off that the build is ready to present?

Hold each other accountable

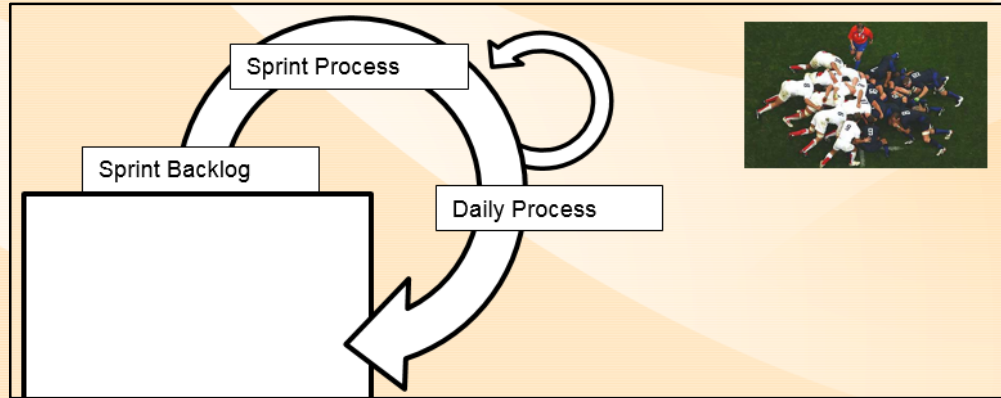
- Code freeze no later than 5pm Tuesday.
- If someone is not done with their work we make the build without that work.
- Everyone needs to be able to come into class at 9am and be productive.
 - Milestone review
 - Sprint planning

Daily Stand Up meeting

Stand up meeting

Keep everyone informed

- What did you do?
- What are you about to do?
- What stands in your way?

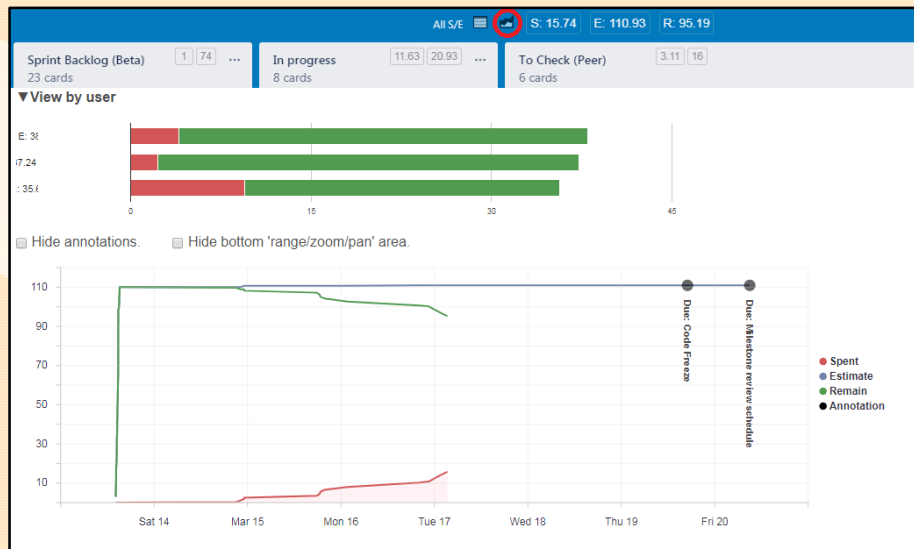


Stand up meeting

Keep everything up to date

- Update the task board (trello) while working
- Enter hours spent on any userstory worked on
- The burn down chart will be updated automatically as hours spent get entered

This team is behind and they should know it so they can respond to it



Sprint 4: Alpha

By the end of the day

- At least 35 hours logged per team member
- Burn down charts ~4/5 complete

