Product Management

Peter K. Shultz

Program Manager, Microsoft

(Somebody still trying to figure it out)

About me

- Graduated from UM in 2018
    - Studied CS and economics
        - Took EECS 482 and ECON 401 at the same time
            - What a disaster 🤦‍♂️
- After graduation, took four months to hike the 2,190.9 mile Appalachian Trail

What this talk is

- An overview of what being a PM is like
- What you may enjoy about being a PM, what you may dislike
- Traits of good PMs
- My opinions

What this talk isn’t

- Waxing poetic about “Back in my day, campus looked like…”
  - Why does everyone do this?
- Telling you what type of computer I first learned to program on
  - Again, why does everyone do this?
- Telling you that everybody should become a PM
  - I will try and maintain a semblance of objectivity.
- My employer’s opinions

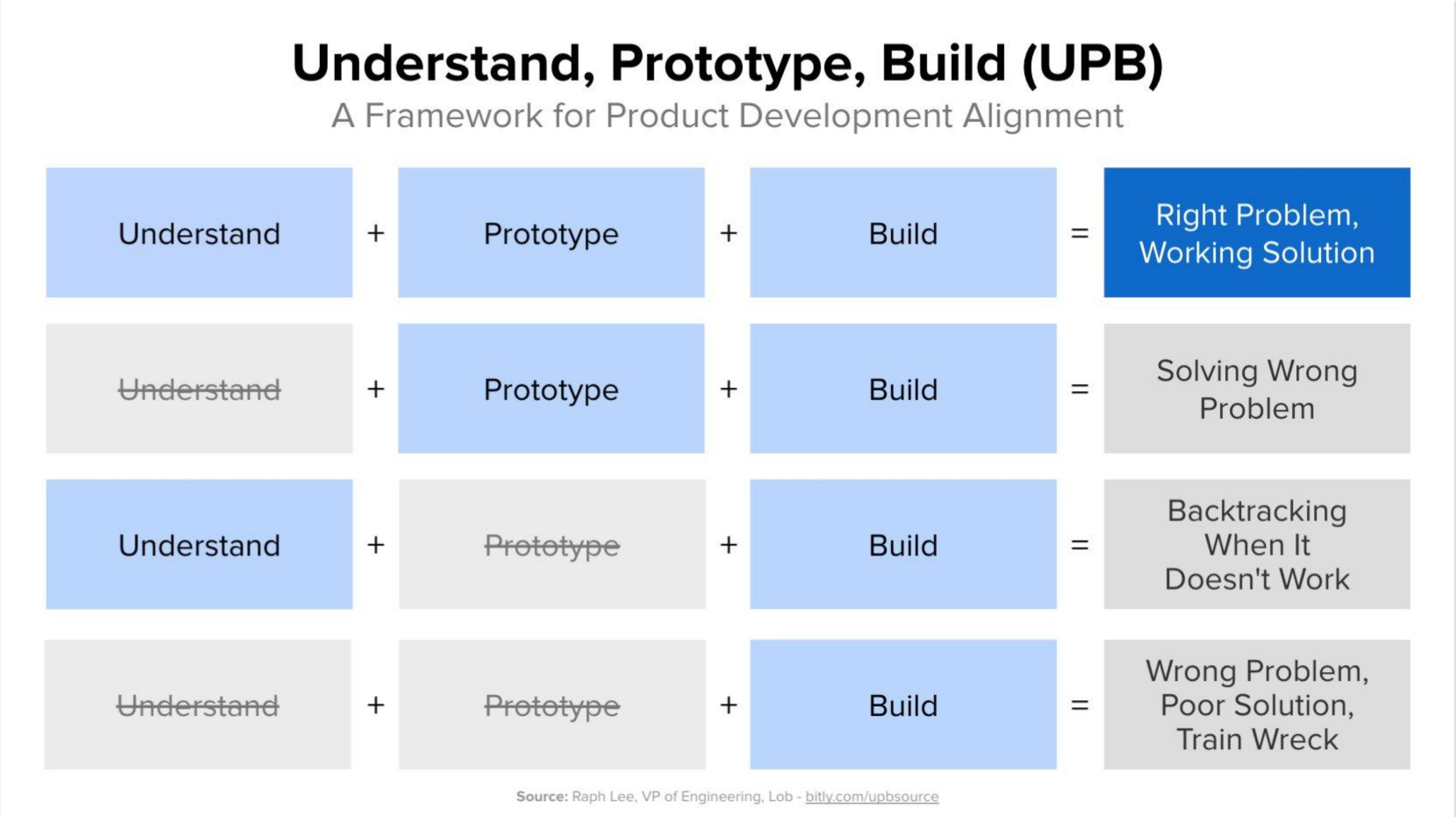
What does PM stand for?

- Product Manager
- [Technical] Program Manager
- Project Manager

- All of these job titles have subtle differences between them, but for sake of simplicity you can think of them as all being the same thing.

What is a PM?

- “PMs make sure companies build the right things, and software engineers make sure the things are built right”
- “PMs deal with things so that software engineers don’t have to”
- “PMs are the voice of the customer for product development”
- “PMs make things happen”



Understand, Prototype, Build (UPB)

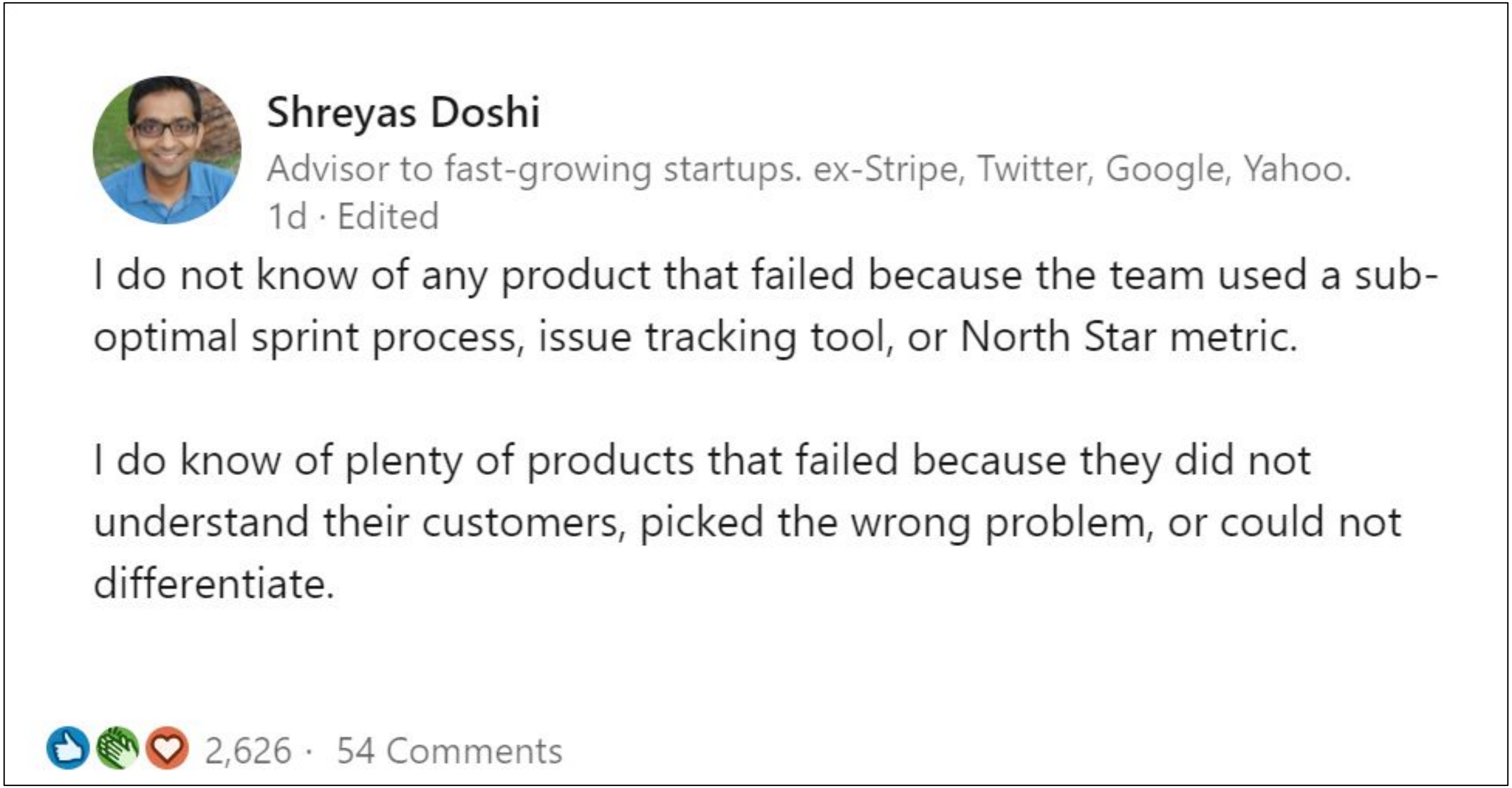
A Framework for Product Development Alignment

Understand	+	Prototype	+	Build	=	Right Problem, Working Solution
~~Understand~~	+	Prototype	+	Build	=	Solving Wrong Problem
Understand	+	~~Prototype~~	+	Build	=	Backtracking When It Doesn't Work
~~Understand~~	+	~~Prototype~~	+	Build	=	Wrong Problem, Poor Solution, Train Wreck

Source: Raph Lee, VP of Engineering, Lob - bitly.com/upbsource

What is a PM?

- “PMs make sure companies build the right things, and software engineers make sure the things are built right”
- “PMs deal with things so that software engineers don’t have to”
- “PMs are the voice of the customer for product development”
- “PMs make things happen”
- “PMs help understand and prototype”



Shreyas Doshi

Advisor to fast-growing startups. ex-Stripe, Twitter, Google, Yahoo.
1d · Edited

I do not know of any product that failed because the team used a sub-optimal sprint process, issue tracking tool, or North Star metric.

I do know of plenty of products that failed because they did not understand their customers, picked the wrong problem, or could not differentiate.

2,626 · 54 Comments

Source

Why would I want to be a PM?

- To ensure no day is the same
- So that you can work in and around technology without needing to code every day
- To learn about other disciplines:
  - Data science
  - User experience
  - Design
  - Finance
  - Marketing
- To have maximum impact on a company or a business
  - “The best PMs at the company are hands-down the most influential people”

    -Technical fellow at Microsoft

Why would I *not* want to be a PM?

- Ambiguity everywhere
  - Sometimes software engineers don’t know what PMs do
    - Hopefully by the end of this lecture you’ll feel like you know!
- Depending on the role, may not do a lot of coding
  - Programming skills could atrophy, though you can always take preventative steps so that this doesn’t happen
- Lack of direct control over product
  - “Influence without authority”
    - No matter how smart/useful/awesome/amazing you think your idea is, you don’t have the time to implement it yourself—you need to convince somebody else to do that for you.

What does the day of a PM look like?

- 9:00 AM – meet with UX researcher to go over qualitative study plan
    - Qualitative study: interviewing people and asking them lots of questions to validate or invalidate a hypothesis you have
- 10:00 AM – call with high-priority customer who is unhappy with your product for reasons X, Y, and Z. They'll ask you for an ETA on a feature that isn't possible to deliver/not prioritized for the short-term roadmap
- 11:00 AM – meet with doc writer, go over list of documentation fixes that need to get done. Triage what we can do this week and what we'll need to punt for later
- 12:00 PM – lunch with coworkers who claim Kendrick Lamar isn't a good artist despite winning over a dozen Grammy Awards
- 1:00 PM – work on SQL queries to track usage patterns of largest customers. Write report with insights to share with broader team. Automate queries so that you don't have to spend so much time doing this later. Help other people write queries
- 2:00 PM – Write spec + build prototype for new logging events. Painstakingly draw out a scenario and then manually write JSON versions of logs. Ingest logs into data source to show team what querying the logs would be like.
- 3:00 PM – meet with a designer and ask for their help in turning your ugly wireframes into high-fidelity mock-ups
- 4:00 PM – meet with business planning to discuss a price change you launched—hear if it was useful
- 5:00 PM – review 4-6 support cases for your product, some of which are in a different language and require a generous reading of a translation

What does another day of a PM look like?

- 9:00 AM – meet with internal team who is using your product to do amazing things. Work with them and marketing colleagues to launch a cool case study
- 10:00 AM – chase down and babysit a bug fix for a high-priority customer. Apologize profusely to customer.
- 11:00 AM – Record a video demo of your product for a special subset of customers.
- 12:00 PM – grab lunch, find a quiet spot in the cafeteria, and catch up on some personal reading
- 1:00 PM – suggest UI and copy changes to your SWE team. Find links to files in codebase where changes need to happen to help the team implement the changes faster
- 2:00 PM – call up a friendly SWE to learn about a feature or technology you don’t understand
- 3:00 PM – go into a review with your division’s CVP (corporate vice president) for a data model you’ve created. Discuss business insights derived from the data model.
- 4:00 PM – write a lengthy SQL query to do some complex data analysis. Ask a teammate to peer review it so that you can verify its accuracy
- 5:00 PM – use an open-source tool to generate really cool visualizations for some data you’ve been working with

How can I find out if I’d be a good PM?

- Evaluate:
  - Your communication skills
    - A majority of the job is communicating your ideas and persuading others. Make sure you’re comfortable writing and speaking frequently.
    - Pro tip: communication is fun and easy when you put things in lists and/or tables.
  - Your conflict resolution and coordination skills
    - You need help from teams A and B to launch feature X. How do you work with both teams (whose priorities are very different from yours) to make things happen?
  - Your comfort in and around ambiguity
    - “Users say we don’t have good observability features. Can you go fix that?”
      - If this sort of prompt gets you curious, you might be interested in PM-like problems!

What does impact look like for a PM?

- Think up a good way to improve a product or process

  → ship the improvement

  → show a tangible difference in an important metric

  → get promoted

  - FYI: this is what impact looks like in pretty much any job in any field, and your resume should ideally be a collection of stories like this

- Example: review 800 support cases to see common issues

  → think up and implement diagnostics to prevent support cases

  → see support case resolution time for common issues go down

What does career trajectory look like for a PM?

Career stage	**Given a problem?**	**Given a solution?**	**Need to execute?**
Junior PM	Yes	Yes	Yes
Mid-level PM	Yes	No	Yes
Lead PM	No	No	Yes

Source: Dare Obasanjo in this Twitter Space, time mark 9:40
(Note: this link won’t be viewable after 2022-04-24, so give it a listen soon!)

What can be frustrating about being a PM?

- Reasonable amount of authority over product vision, but not necessarily product development day-to-day
  - You don’t code—you convince others what to code
    - I’ll say it again: “Influence without authority”
- Trying to get teams to align on an idea/vision
  - Like getting all your friends to agree on where to go for dinner, but way harder

How do I land a PM role?

- Option 1: Internships
    - Feasible, though do be aware that openings are limited
        - Look for any company you admire with a PM internship role. Apply! Reach out to a recruiter!
- Option 2: first job out of college without prior internship
    - Lots of companies have APM (associate product manager) programs.
        - These look really cool IMO.
        - Plan to:
            - Show technical and business aptitude
            - Use products critically—how can they be better? What would you improve?
- Option 3: Start as a software engineer, move over to PM
    - “Try to find a role in your current team where you have domain knowledge & credibility. It’s much harder to learn new team/tech & new discipline at the same time.” (source)
        - Credibility is super important. Having a great relationship with your engineers makes being a PM 10x easier.
            - Some people think the best PMs are those with prior software engineering experience

What should I read to learn more about being a PM?

Books	Blogs	Twitter accounts
Making Things Happen	Sachin Rekhi	Dare Obasanjo
The Mom Test	Bring the Donuts	Jackie Bavaro
Working Backwards		Shreyas Doshi
Cracking the PM Interview		Gergely Orosz*
Cracking the PM Career		Aparna Chugh
Debugging Teams*		
The Effective Engineer*		

*even if you have zero interest in being a PM, you should read this

How does any of this relate to what I’m learning in class?

- Elicitation, validation, and risk
  - When talking with customers, you have to ask questions to ensure you understand the root cause of their pain.
    - Sometimes, customers will say pain point A is their biggest issue, and that if you created feature A*, their life would be better. As a PM, you should have the domain expertise to understand whether their problem is really A, or if A is just a symptom of pain point Z (in which case the best solution is actually Z*).
      - “Customers can't tell us everything. They are experts in pain, not solutions.” – April Dunford
  - The book *The Mom Test* talks about how you can phrase questions such that customers don’t lie to you about how good your idea is (so that they don’t hurt your feelings).
- System design mock interview
  - Mapping user needs into a technical design
    - As a PM, you may sit in on technical design discussions to validate that a user’s needs are 100% met from the design. SWEs *should* capture everything, but it’s good to double-check.

How do you resolve conflicts with a teammate, colleague, or manager?

- Options:
  - Communicate your needs as directly as possible
    - Excellent book recommendation on this topic: Nonviolent Communication
      - Feels a little corny at first but is ultimately worth your time
        - I have a hypothesis that all problems boil down to communication problems
  - Put your best foot forward, work earnestly to understand the other person's needs—see if you can reconcile with them
    - "You catch more flies with honey than vinegar"
  - Escalate—tell other people in authority what problems you're facing
    - "People in authority" can often mean somebody's manager, though it can also be peers/colleagues who are close with or influential to the person you're dealing with
- If a situation is bad, you don't have to try and resolve it.
  - Sometimes the juice isn't worth the squeeze—you're in the driver's seat of your career, and you should feel empowered to find something that meets your needs

How can I reach out and learn more?

- Email me! I want to help you.
  - pshultz@umich.edu
  - Peter K. Shultz on LinkedIn

Unsolicited life advice

- Do something fun after graduation and before you start a job/go to grad school
  - Take a trip, go on a long hike, act in a play, anything!
- Don’t be afraid to negotiate your first job offer—you have nothing to lose and everything to gain
  - Good blog posts on negotiation by Haseeb Qureshi and Patrick McKenzie
- Invest money early. Time is on your side!
- Wear sunscreen
  - It’s not lost on me that I’m saying this to a bunch of Michigan students in April
- If you move to a city with decent transit options, try living without a car for one month
  - Regardless of how you move around, try living close to a grocery store
- Nobody knows anything—we’re all making things up as we go along. There’s very little separating you from <Person you admire> besides a little bit of gumption and luck.

Things I wish I read a long time ago

- Do things, tell people
- Do the real thing
- Validating product ideas before building them
- Your product is not their problem
- There’s no speed limit