Software Engineering

Week 11: Other Aspects in Software Engineering

Dr. Sridhar Iyer, IIT Bombay
Dr. Prajish Prasad, FLAME University

Software Engineering

Software Organizations

Dr. Sridhar Iyer, IIT Bombay
Dr. Prajish Prasad, FLAME University

Purpose of a Software Company

Provide value to potential users of the software system

Reflection Spot

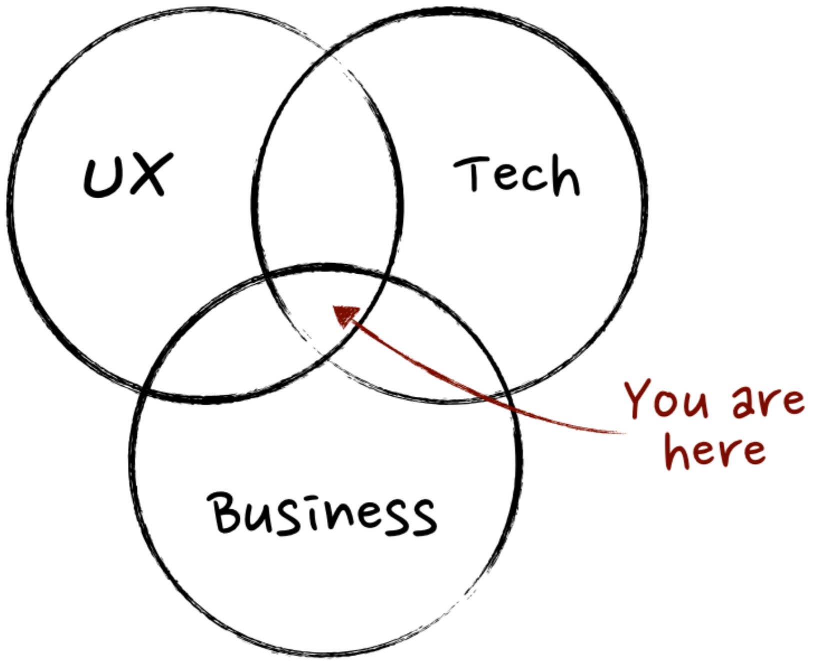
Imagine that you are a software developer working in a software company that wants to build the Amazon Seller Portal. What are other types of roles in a software company?

Please pause the video and written down your responses

Marketing Team

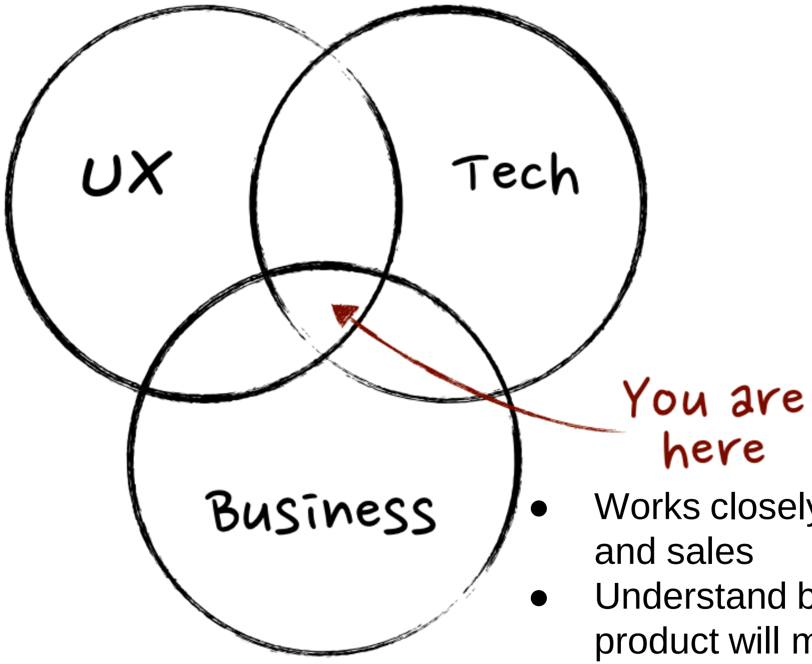
- Look for opportunities in the market - to provide value
- Conduct market research, identify audiences

Product Managers



https://medium.com/@bfgmartin/what-is-a-product-manager-ce0efdcf114c

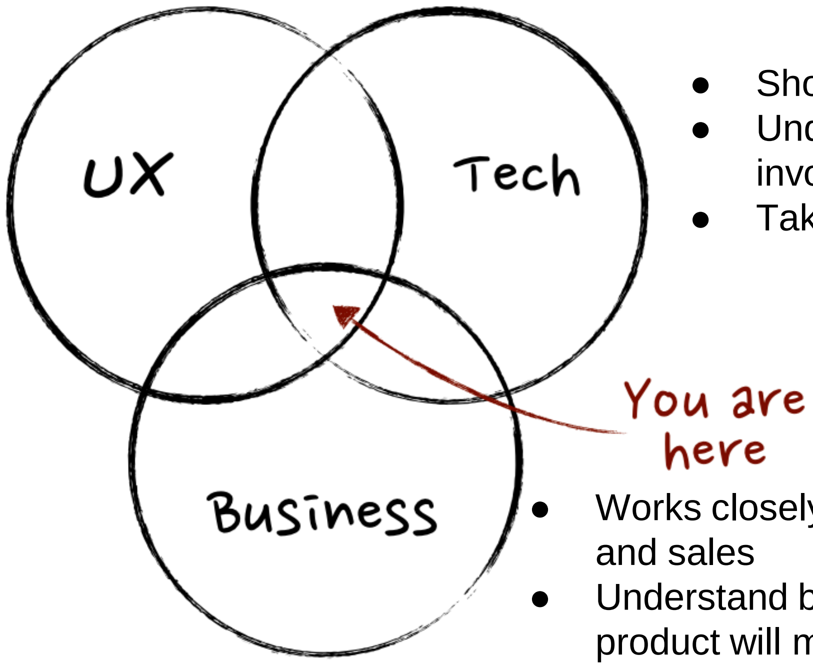
Product Managers



- Works closely with marketing teams and sales
- Understand business goals - how the product will maximise return on investment

https://medium.com/@bfgmartin/what-is-a-product-manager-ce0efdcf114c

Product Managers



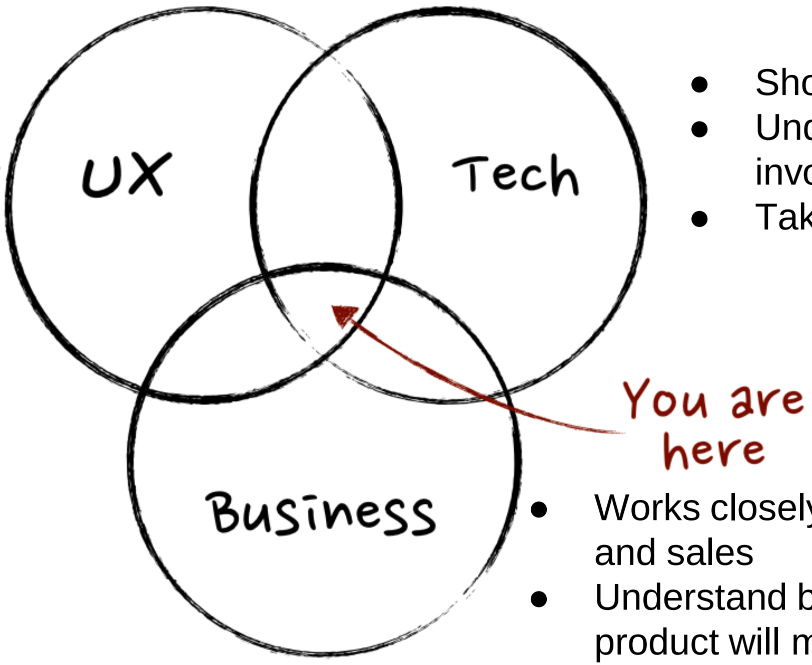
- Should know the technology stack
- Understand the level of effort involved
- Take important technical decisions.

- Works closely with marketing teams and sales
- Understand business goals - how the product will maximise return on investment

https://medium.com/@bfgmartin/what-is-a-product-manager-ce0efdcf114c

Product Managers

- Passionate about the user
- What does the user want



- Should know the technology stack
- Understand the level of effort involved
- Take important technical decisions.

- Works closely with marketing teams and sales
- Understand business goals - how the product will maximise return on investment

https://medium.com/@bfgmartin/what-is-a-product-manager-ce0efdcf114c

Designers

- User Experience (UX) roles
- Transform requirements to solutions
- Talk to users, create prototypes

Software Engineers

- Write code with others in the team to implement requirements

Software Engineers

- Write code with others in the team to implement requirements

- Software engineers don’t get to decide what product is made, or what problems the product solves (most of the time)

Engineering Managers

- In large organizations - transmitting information between higher and lower parts
- Also known as project manager
    - Organizing and prioritizing work
    - Coordinating between different teams
    - Resolving interpersonal conflict between engineers

Sales Team

- Sell the product it to users that marketing team has identified

- Provide feedback to marketing, product, and design teams regarding the product, which engineers then address.

Support Team

- Resolve problems that clients have

Support Team

- Resolve problems that clients have
- Provide feedback to product, design, and engineering about the product and it’s defects/shortcomings

Data Scientists

- Analyze data generated from different teams, users
- Help organization make better decisions
- E.g. -
    - Help marketing team analyze business data
    - Track sales targets
    - Help engineers understand patterns on app usage

Ethics and Policy Specialists

- People with background in law, social science, policy
- Shape the terms of service of the software product, software licenses, privacy policy etc.
- Important for any company that works with data

References

Material for this video taken from
Cooperative Software Development by
Dr. Amy J. Ko, University of Washington
Chapter 2 - Organizations
https://faculty.washington.edu/ajko/books/cooperative-software-development/organizations

Software Engineering

Communication, Collaboration and Productivity

Dr. Sridhar Iyer, IIT Bombay
Dr. Prajish Prasad, FLAME University

Conceptual Integrity

Everyone on a team has the same understanding of what is being built and why

Conceptual Integrity

Everyone on a team has the same understanding of what is being built and why

Effective communication ensures conceptual integrity

Reflection Spot

How do you think software engineers communicate with each other? What tools do they use?

Please pause the video and written down your responses

Knowledge Sharing Tools

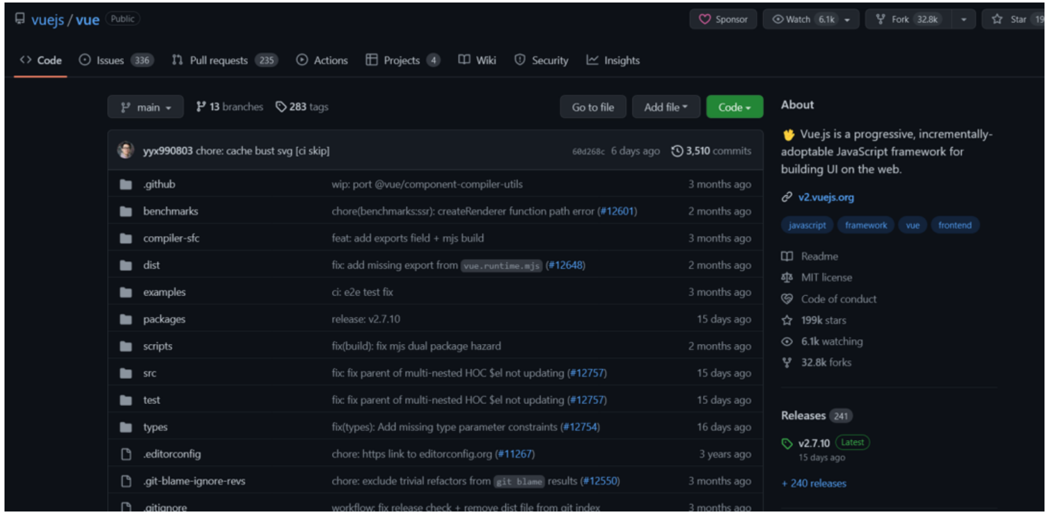
- Main Purpose - used for sharing documents and archiving decisions

Issue Tracker - Knowledge Sharing Tools

- JIRA, Pivotal Tracker
- Track different issues
- History of who all worked on these issues

GitHub Pages - Knowledge Sharing Tools

- Many libraries, frameworks are hosted on Github - E.g. - Vue.js



yyx990803 chore: cache bust svg [ci skip] 60d268c 6 days ago 3,510 commits

.github	wip: port @vue/component-compiler-utils	3 months ago
benchmarks	chore(benchmarks:ssr): createRenderer function path error (#12601)	2 months ago
compiler-sfc	feat: add exports field + mjs build	3 months ago
dist	fix: add missing export from `vue.runtime.mjs` (#12648)	2 months ago
examples	ci: e2e test fix	3 months ago
packages	release: v2.7.10	15 days ago
scripts	fix(build): fix mjs dual package hazard	2 months ago
src	fix: fix parent of multi-nested HOC $el not updating (#12757)	15 days ago
test	fix: fix parent of multi-nested HOC $el not updating (#12757)	15 days ago
types	fix(types): Add missing type parameter constraints (#12754)	16 days ago
.editorconfig	chore: https link to editorconfig.org (#11267)	3 years ago
.git-blame-ignore-revs	chore: exclude trivial refactors from `git blame` results (#12550)	3 months ago
.gitignore	workflow: fix release check + remove dist file from git index	3 months ago

About

🖖 Vue.js is a progressive, incrementally-adoptable JavaScript framework for building UI on the web.

v2.vuejs.org

javascript framework vue frontend

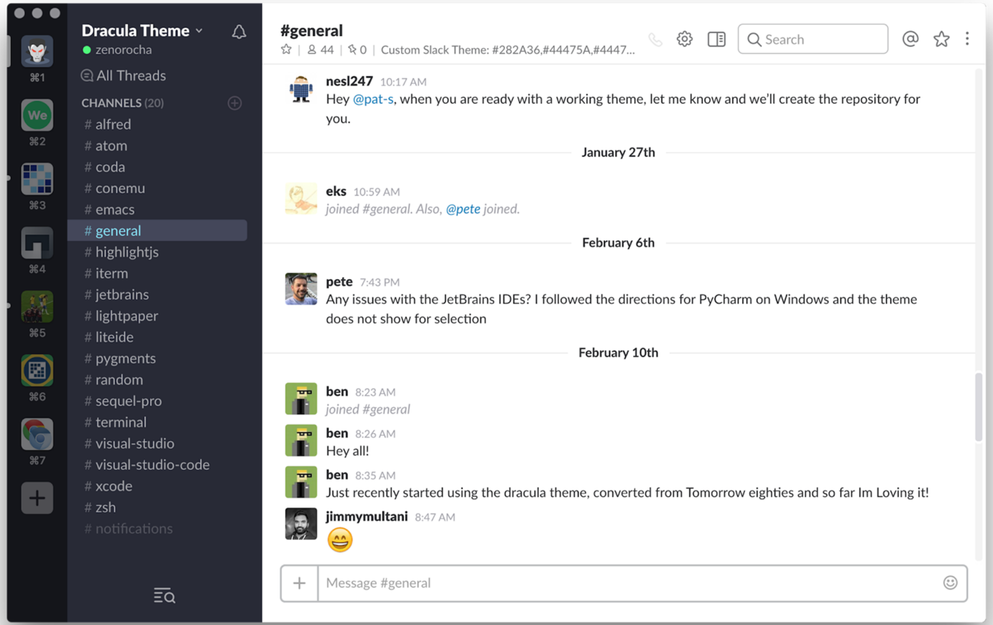
Readme
MIT license
Code of conduct
199k stars
6.1k watching
32.8k forks

Releases 241

v2.7.10 Latest
15 days ago

+ 240 releases

Slack - Knowledge Sharing Tools



nesl247 10:17 AM
Hey @pat-s, when you are ready with a working theme, let me know and we'll create the repository for you.

January 27th

eks 10:59 AM
joined #general. Also, @pete joined.

February 6th

pete 7:43 PM
Any issues with the JetBrains IDEs? I followed the directions for PyCharm on Windows and the theme does not show for selection

February 10th

ben 8:23 AM
joined #general

ben 8:26 AM
Hey all!

ben 8:35 AM
Just recently started using the dracula theme, converted from Tomorrow eighties and so far Im Loving it!

jimmymultani 8:47 AM
😄

Message #general

Stack Overflow - Knowledge Sharing Tools

- Helps to resolve issues that you are facing
- Provides links to additional learning resources

Importance of Knowledge Sharing Tools

- Knowledge of one project might be needed in another project. Useful if it is documented and archived properly

Importance of Knowledge Sharing Tools

- Knowledge of one project might be needed in another project. Useful if it is documented and archived properly
- When people leave the organization - specialised knowledge goes along with them

Importance of Knowledge Sharing Tools

- Knowledge of one project might be needed in another project. Useful if it is documented and archived properly
- When people leave the organization - specialised knowledge goes along with them
    - Mitigation - “cross-training” - rotating developers between projects

Productivity

- Traditionally - work done per unit time

Productivity

- Traditionally - work done per unit time
- Difficult to define in software engineering - Not necessarily no of lines of code

Productivity

- Traditionally - work done per unit time
- Difficult to define in software engineering - Not necessarily no of lines of code
- E.g. - delivered the features assigned to you, while ignoring other team members - harms the team’s productivity

Productivity - Right Tools

- Project management tools -
  - Enables all members of the team to get a big picture as well as detailed view of progress
  - E.g. JIRA, Pivotal Tracker

Productivity - Right Tools

- Project management tools -
  - Enables all members of the team to get a big picture as well as detailed view of progress
  - E.g. JIRA, Pivotal Tracker
- Development tools
  - IDEs, features in IDEs help developers become more productive

References

Material for this video taken from
Cooperative Software Development by
Dr. Amy J. Ko, University of Washington

- Chapter 3 - Communication
  https://faculty.washington.edu/ajko/books/cooperative-software-development/productivity

- Chapter 4 - Productivity
  https://faculty.washington.edu/ajko/books/cooperative-software-development/communication

Software Engineering

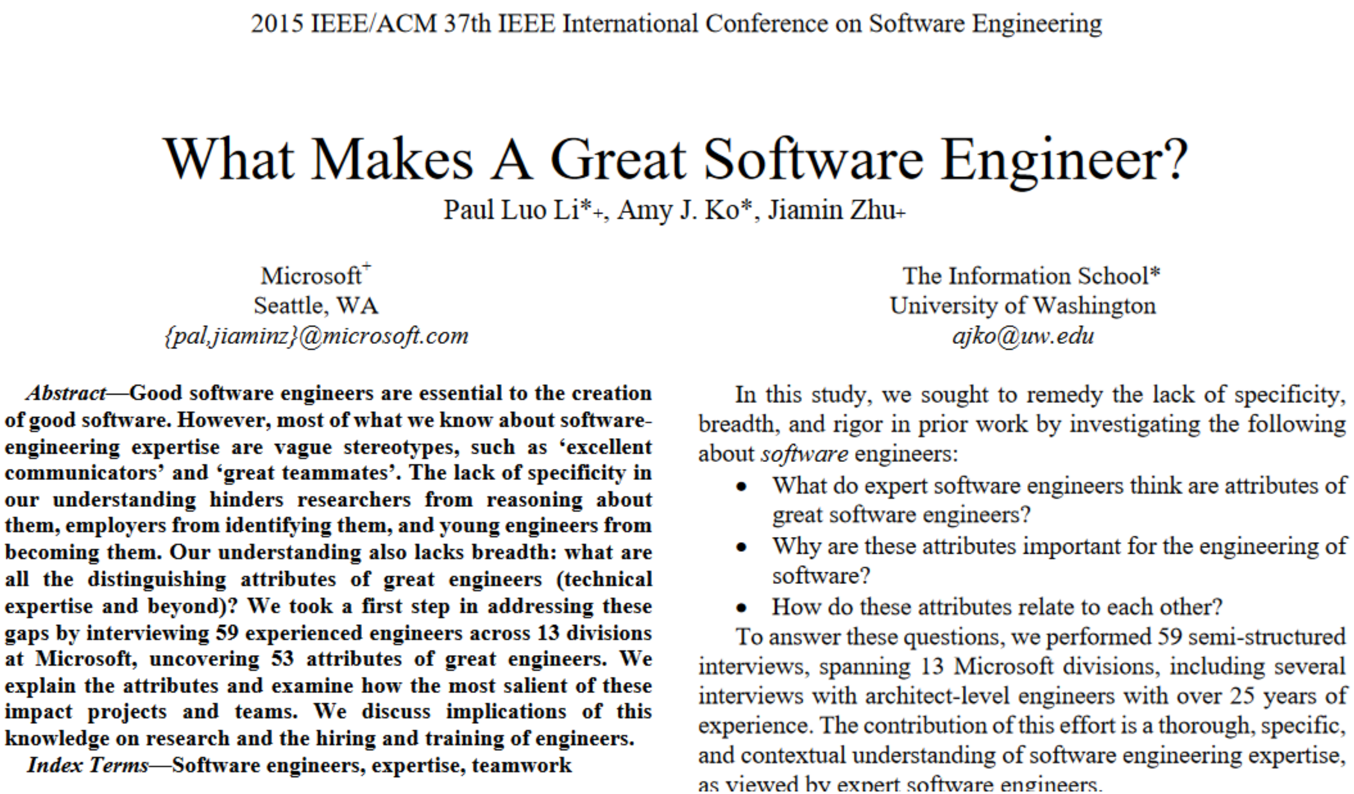
What Makes a Great Software Engineer?

Dr. Sridhar Iyer, IIT Bombay
Dr. Prajish Prasad, FLAME University

Reflection Spot

What do you think are the qualities and skills that are required to be a good, effective software engineer?

Please pause the video and written down your responses



2015 IEEE/ACM 37th IEEE International Conference on Software Engineering

What Makes A Great Software Engineer?

Paul Luo Li*+, Amy J. Ko*, Jiamin Zhu+

Microsoft+
Seattle, WA
{pal,jiaminz}@microsoft.com

The Information School*
University of Washington
ajko@uw.edu

***Abstract*—Good software engineers are essential to the creation of good software. However, most of what we know about software-engineering expertise are vague stereotypes, such as 'excellent communicators' and 'great teammates'. The lack of specificity in our understanding hinders researchers from reasoning about them, employers from identifying them, and young engineers from becoming them. Our understanding also lacks breadth: what are all the distinguishing attributes of great engineers (technical expertise and beyond)? We took a first step in addressing these gaps by interviewing 59 experienced engineers across 13 divisions at Microsoft, uncovering 53 attributes of great engineers. We explain the attributes and examine how the most salient of these impact projects and teams. We discuss implications of this knowledge on research and the hiring and training of engineers.**

***Index Terms*—Software engineers, expertise, teamwork**

In this study, we sought to remedy the lack of specificity, breadth, and rigor in prior work by investigating the following about *software* engineers:

- What do expert software engineers think are attributes of great software engineers?
- Why are these attributes important for the engineering of software?
- How do these attributes relate to each other?

To answer these questions, we performed 59 semi-structured interviews, spanning 13 Microsoft divisions, including several interviews with architect-level engineers with over 25 years of experience. The contribution of this effort is a thorough, specific, and contextual understanding of software engineering expertise, as viewed by expert software engineers.

http://dl.acm.org/citation.cfm?id=2818839

Decision Making

- Macro decisions -
  - design and architecture level
  - E.g. which libraries, frameworks to use
- Micro decisions
  - Algorithms, data structures for a particular module

Rational Decision Making Process

1. Identify the decision to be made
2. Systematically identify alternatives
3. Think through potential outcomes
4. Evaluate which of the outcome is best for the given context
5. Make a decision

Course Summary

- Software processes - Different processes used in software development
- Tools - Used to build software - capture requirements, software planning, development, testing
- Code - how to organize your code, best practices