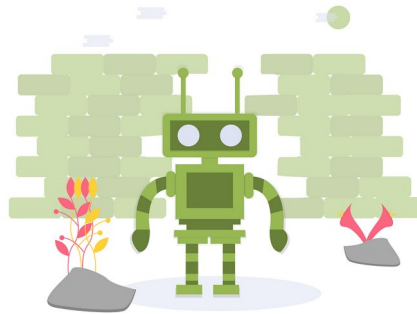


# Software Engineering und Projektmanagement Teil II

Projektgruppe 642 SoSe 2021



O'REILLY®

# Software Engineering at Google

Lessons Learned  
from Programming  
Over Time



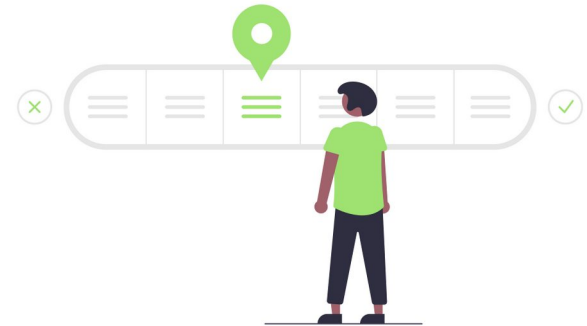
Curated by Titus Winters,  
Tom Manshreck & Hyrum Wright

## Google's software engineering landscape

- Culture
- **Processes**
- Tools

# Agenda

1. Code Review
2. Style Guides and Rules
3. Dokumentation



# 1. Code Review





## Code Review? Warum und wozu?

- Gegenseitiges Überprüfen des Codes (Review)
  - *Abnahme eures Codes durch eine geeignete dritte Person*
- Absicherung der CodeBase → Qualitätssicherung
- Kritik/Feedback (**auch positiv!**)
- Gegenseitiges Verständnis über die gemeinsame Arbeit
- Lob

## Aspekte eines Reviews

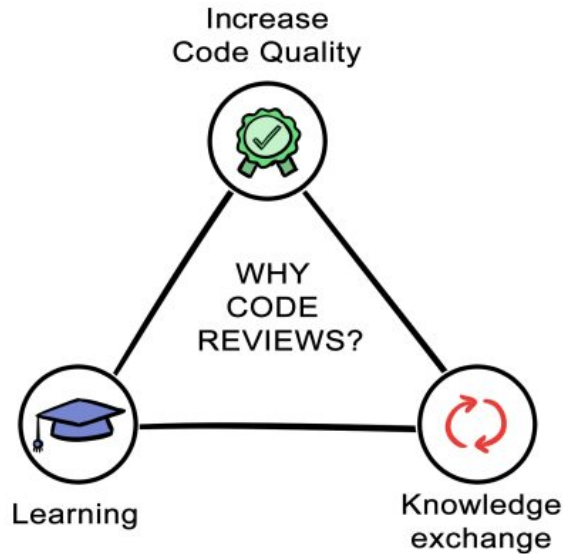
- Design
- Funktionalität
- Komplexität
- Namensgebung
- Kommentare
- Style
- Konsistenz
- Kontext
- **Good Things**



## **Code Review Ablauf bei Google**

- 1) Änderungen einprogrammieren
- 2) Reviewrequest einstellen
- 3) Review der Änderungen
- 4) Überarbeitung
- 5) Verifizierung
- 6) Submission

## Code Review Benefits



## Code Review Benefits

- Korrektheit
- Verständlichkeit des Codes im Team
- Effizienz des Codes
- Wissenstransfer
- Konsistenz und guter Stil
- Teamgeist → *geistiges Werk/Teamleistung*

## **Code Review Best Practice**

- Be Polite and Professional
- Write Small Changes
- Write Good Change Descriptions
- Keep Reviewers to a Minimum
- Automate Where Possible

## Code Review Tools

- Git
- Visual Studio
- Gerrit

## Ausblick: Scrum

