

0.1 Importance of Issues and Pull Requests

Before GitHub, open source project developers could receive feedback from multiple means, e.g. forums, e-mail, chat, etc. which means there wasn't a clear way to find and contribute to existing problems.

Issues and Pull Requests come to fix this, by centralizing everything. Take for example this year's third project, where students had to learn how to contribute to open-source projects on GitHub.

The Issues and Pull Requests tabs are incredibly useful since they aggregate the projects current and past issues and the decision making that goes into fixing them, making it easier to contribute to them. This way, it's also easier to find current problems, thus denying the possibility of duplicate threads.

0.2 ChatOps

Concept introduced by a GitHub employee. In an example shown in the talk, GitHub has a Slack workspace where they have deployed a bot that can do almost everything they need. From showing cat images to displaying informations about employees, repositories, pull requests, etc.

0.3 Advice in Engineering

- Shift left - perform testing earlier in the lifecycle
- Pattern importance
- Readable code & architecture
- Effective technical communication
- Re-use more / invent less
- Automate everything
- Deploy, measure, improve
- Early on, prioritise learning
- Find ways to constantly be challenged
- Understand selling