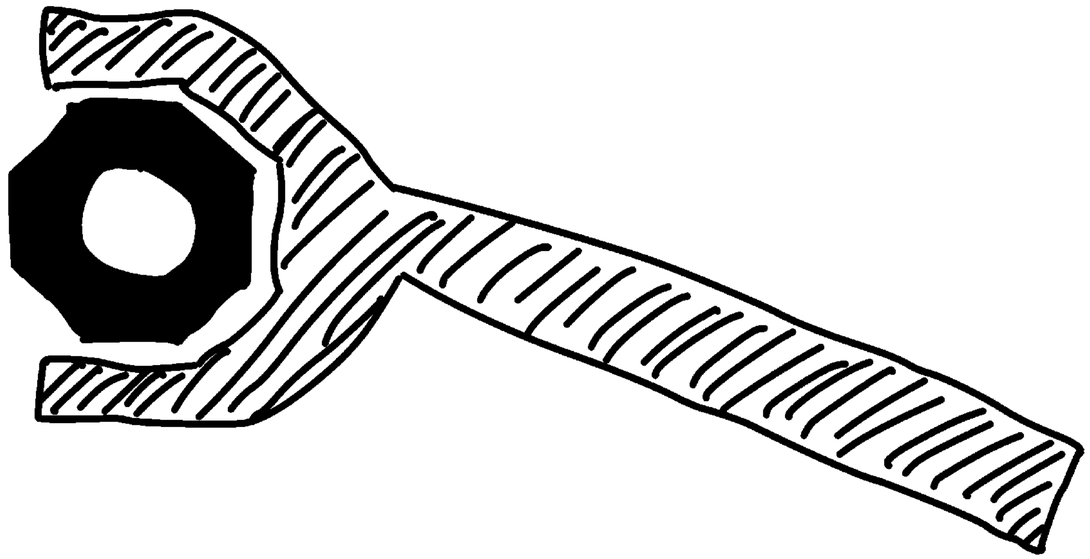
Tighten the feedback loop

Using Double Loop TDD



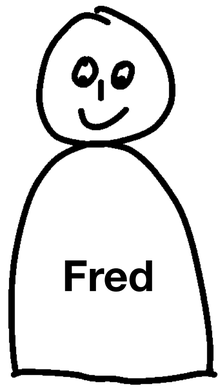
September 24, 2020
by Peti Koch

Today

- Learn about Double Loop TDD
- Using a comparison with a fictive other approach
- See the Benefits of Double Loop TDD
- Conclusions

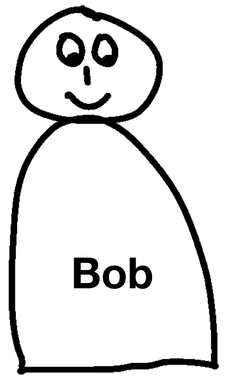
Introduction of the characters

- Likes to program
- Is an experienced Software Engineer



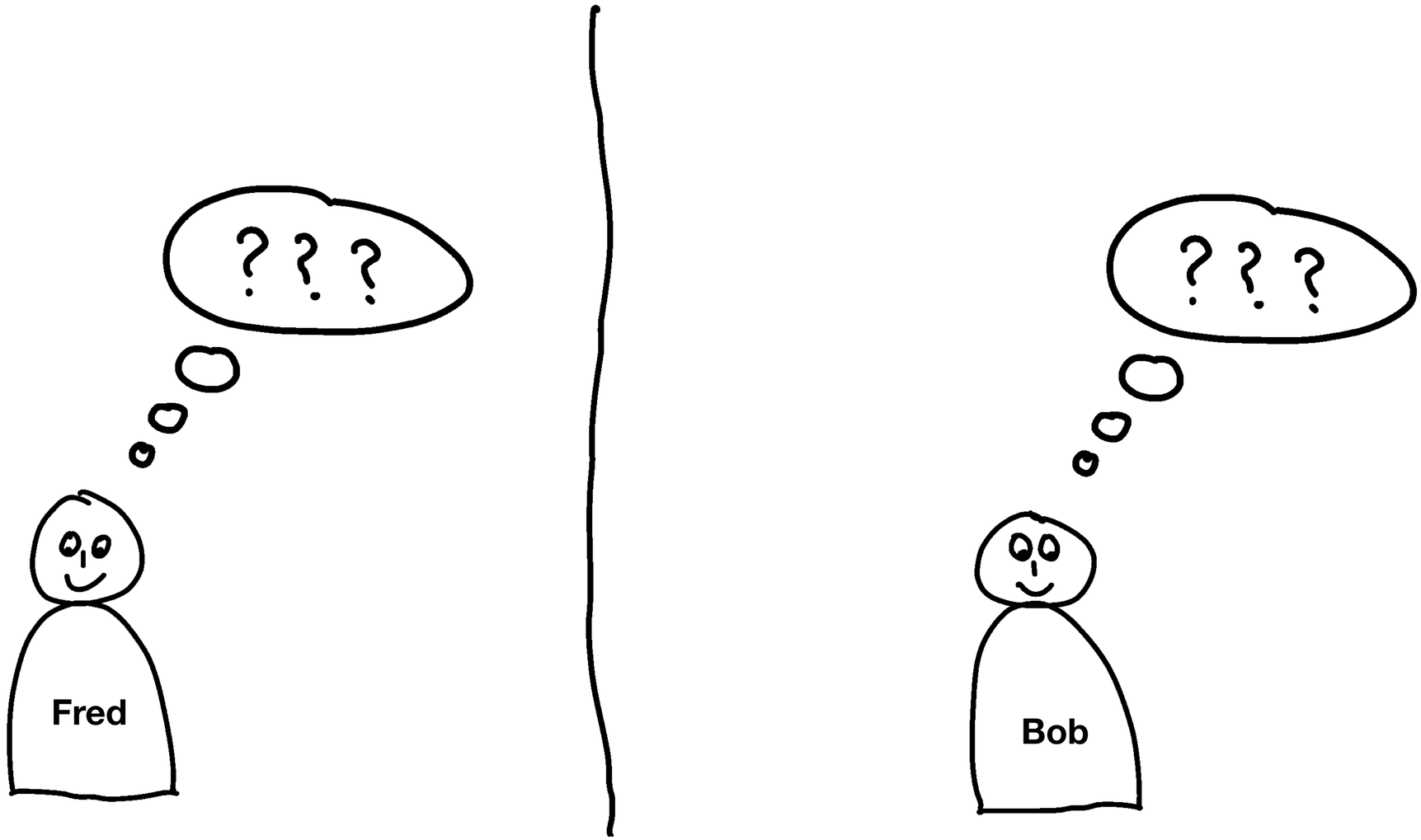
- Likes to program
- Is an experienced Software Engineer

- Uses some advanced Software Engineering practices and likes to automate everything

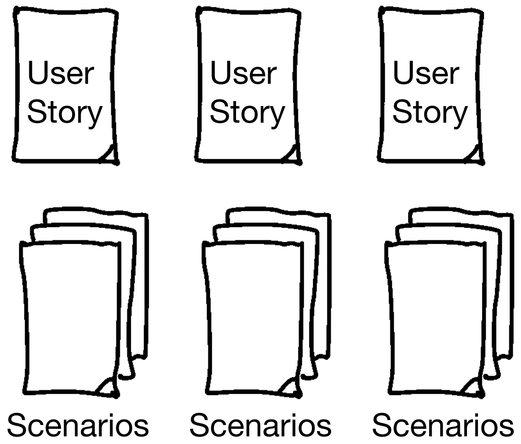


The boss comes in…

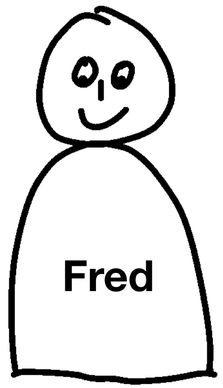
How do Fred and Bob approach this task?

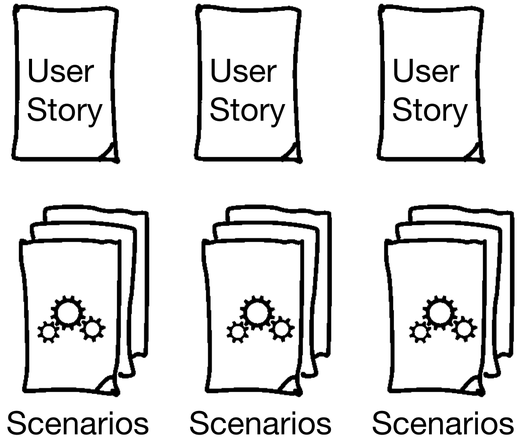


Well, Fred and George are - of course - AGILE!

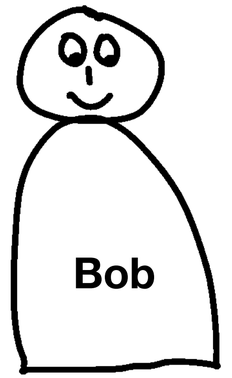


- **Agile Tools!**
- **Jira**
- **Confluence**

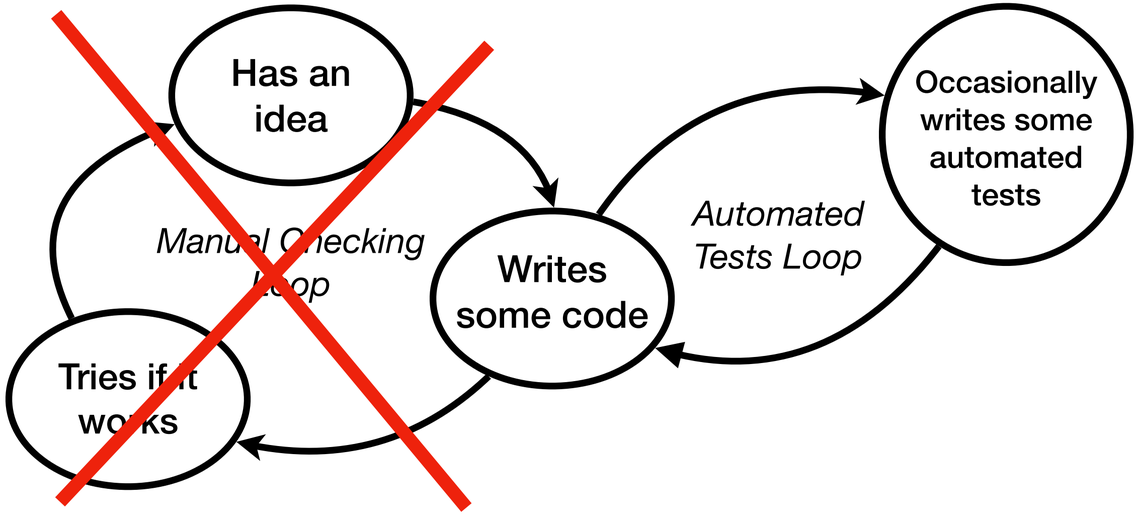


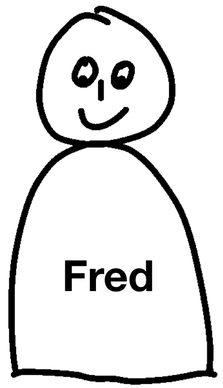


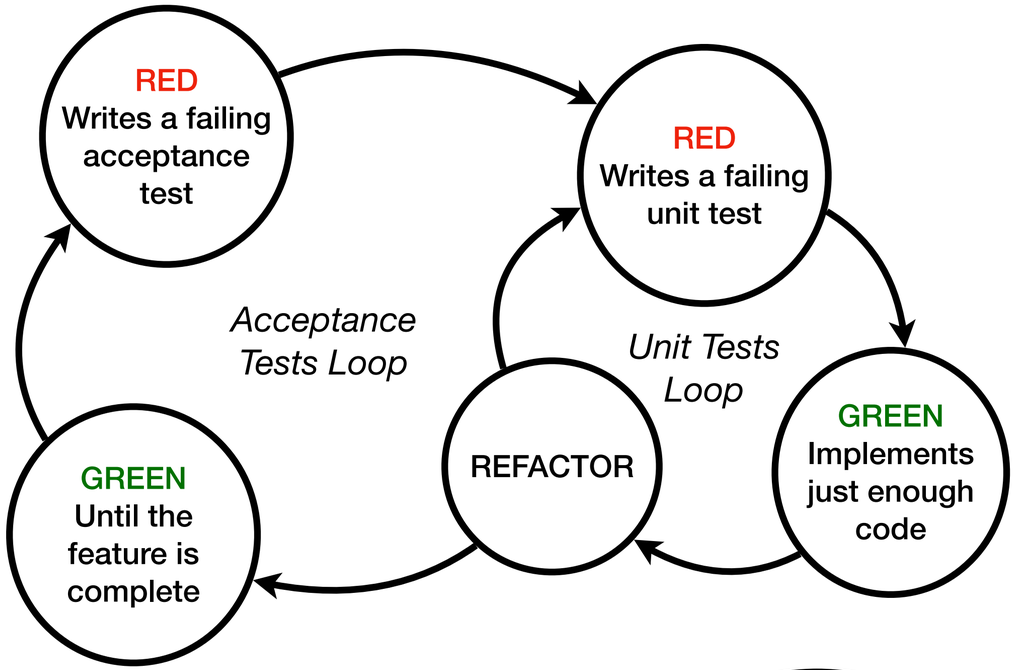
- **Software Engineering Tools!**
- **Source Code**
- **Git**

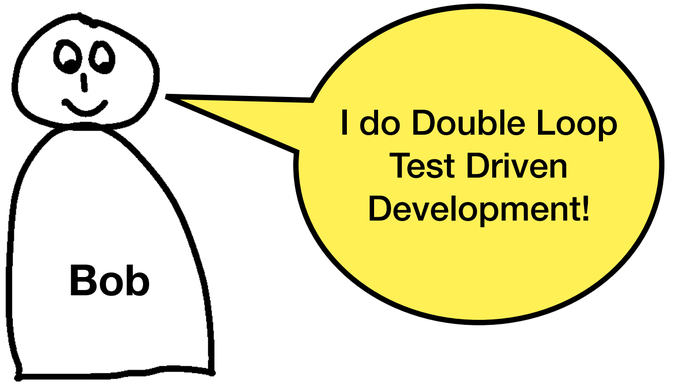


How do they implement the web application?

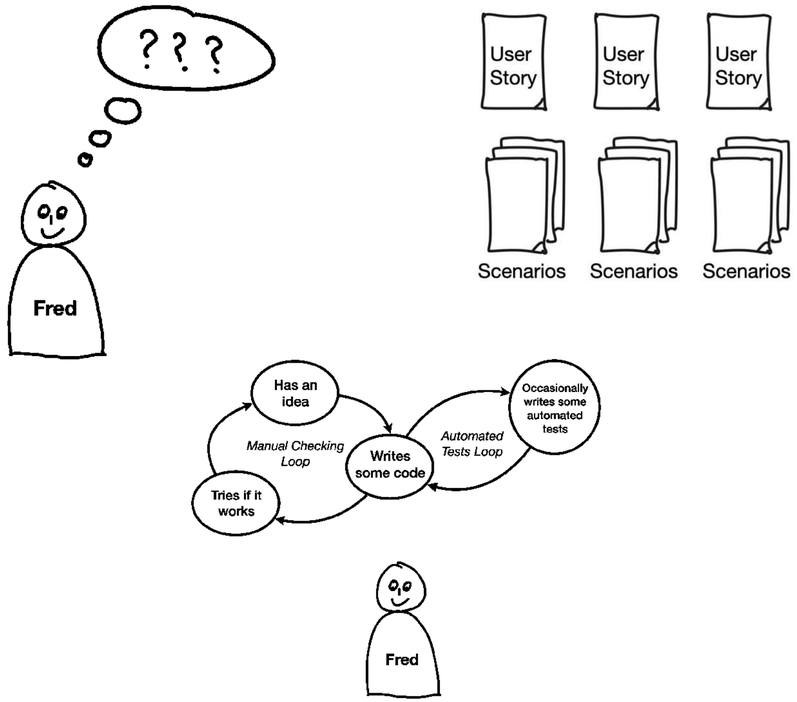




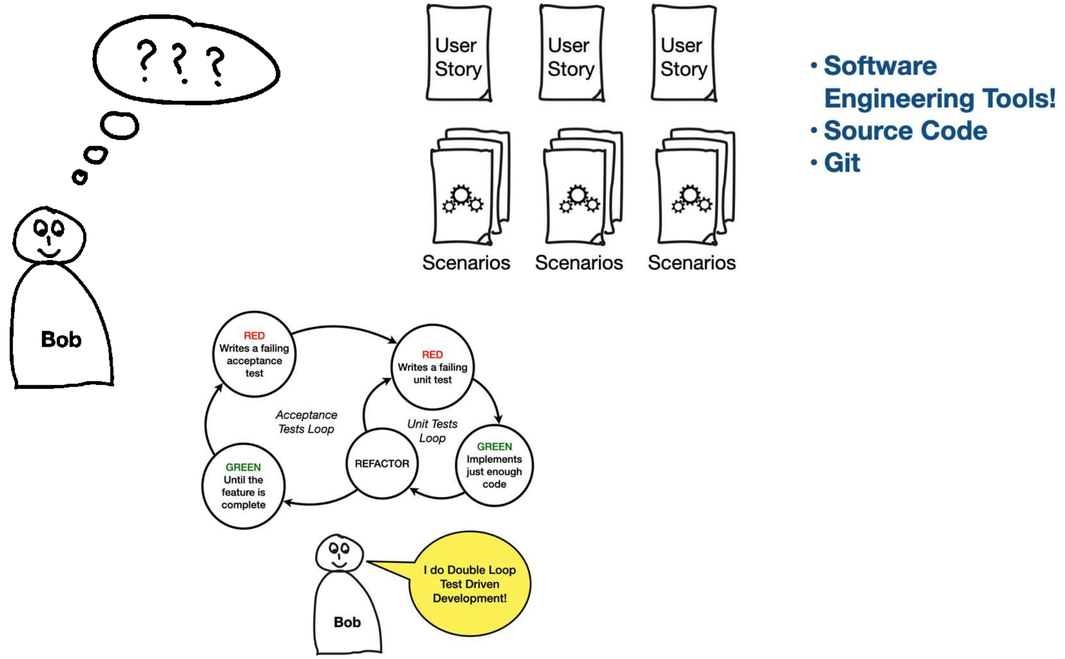




Let’s compare the two approaches



- Agile Tools!
- Jira
- Confluence



- Software Engineering Tools!
- Source Code
- Git

- User Stories with Scenarios are just documentation
- Some (or few) automated (acceptance) tests
- The tests are probably out of sync with the documentation
- (Brittle?) Feedback within hours or even days
- Fear / Resitance to change the web application

- Documentation == Source Code == Executable Specs
- Fast, automated & repeatable tests
- Reliable Feedback within minutes
- Confident to change the web application

Conclusion

- Think about using Double Loop TDD to tighten the feedback loop
- Like Bob
- Make sure you have good resources or a good teacher to learn it properly

Questions?