TEST DRIVEN DEVELOPMENT

EXPLORING A WHOLE NEW WORLD

MOB PROGRAMMING

PREVIOUS EXPERIENCE

- 10 ish tests

- Pair programmed for around 20hours

- Big rewriting tasks instead of smaller improvements

- Are tests reeeaaaally necessary?

PAIR/MOB PROGRAMMING

- Demands focus

- Better flow/rarely stuck

- Easy to be hung up in details

- Time consuming

- Facilitator MVP

- Danger of mob mentality

- Knowledge sharing

NEW APPROACH

Reverting the whole process

Create the test first

Fake it

Refactor early and aggressive

KNOW WHAT YOU ARE TESTING

- Naming conventions

- No abbreviations

- Fail for the right reasons

- Clear speccs of the application

- Nothing happens by mistake

KNOW WHAT YOU ARE TESTING

- Triangulation (generalization)

- Test Behaviour

- Separate production and test code

- Do as little as possible to make a red test green

- IDE is strong

TEST DRIVEN DEVELOPMENT PROS AND CONS

- Long lived applications/Legacy

- Expensive and time consuming

- Automaticly less coupling and cohesion in code

THANK YOU!

GitHub · vetle.horpestad

in · Vetle Horpestad

✉ · vetle.horpestad@bouvet.no