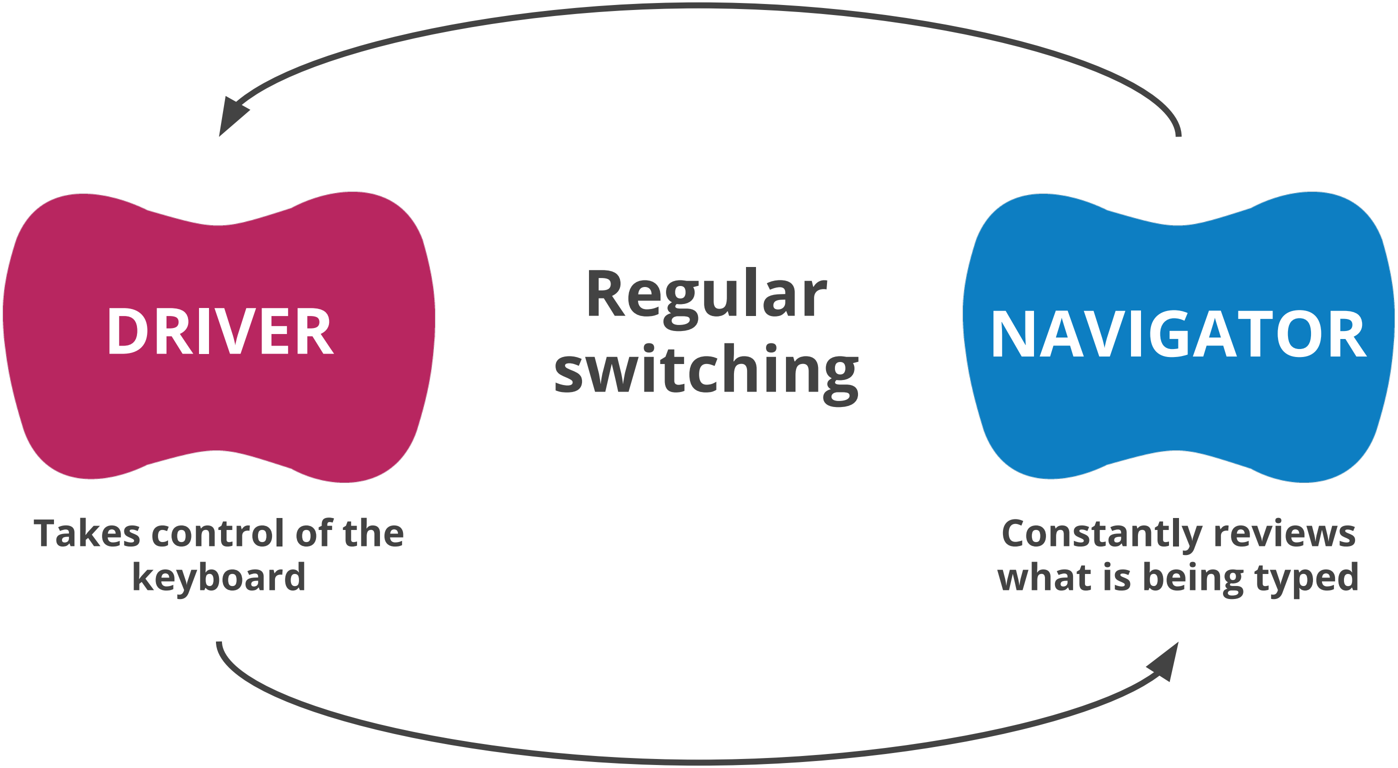
ThoughtWorks®

LevelUp Build

PAIR PROGRAMMING AND TEST-DRIVEN DEVELOPMENT

PAIR PROGRAMMING ROLES



DRIVER

Regular switching

NAVIGATOR

Takes control of the keyboard

Constantly reviews what is being typed

EXPECTED BENEFITS

- **Continuous Code Reviews.** Many mistakes get caught as they are being typed in rather than in QA test or in the field, leading to lower end defect count

- **Design.** Better code design/structure

- **Line-of-sight learning.** The people learn significantly more about the system and about software development

- **Shared Understanding.** The project ends up with multiple people understanding each piece of the system

CORE PAIRING GUIDELINES

1. **Collaborate; don't critique**
2. Actively contribute
3. Switch roles frequently
4. Find comfortable work stations
5. Rest if you must.

PAIRING SMELLS

1. **Unequal access**

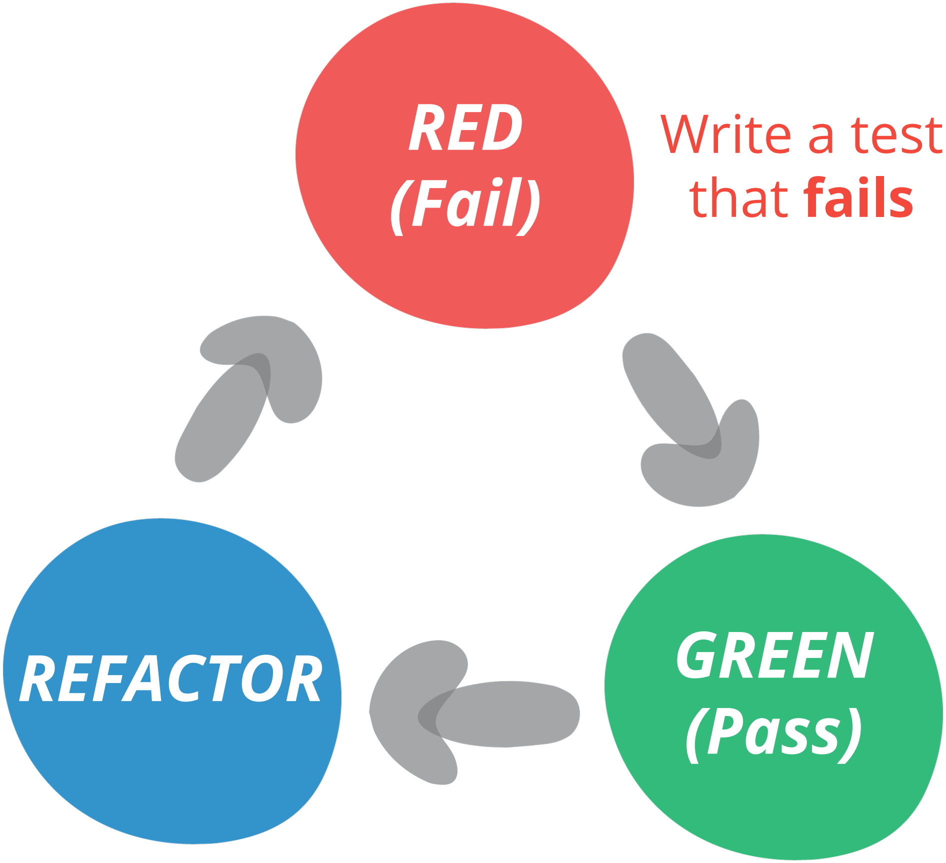
   e.g. one person dominating the keyboard

2. **Unhealthy relationship**
3. **Worker-rester relationship**
4. **Endless debate**
5. **“Go make a cup of tea”** syndrome

ThoughtWorks®

TEST-DRIVEN DEVELOPMENT (TDD)

RED-GREEN-REFACTOR: THE TDD MANTRA



RED (Fail)

Write a test that **fails**

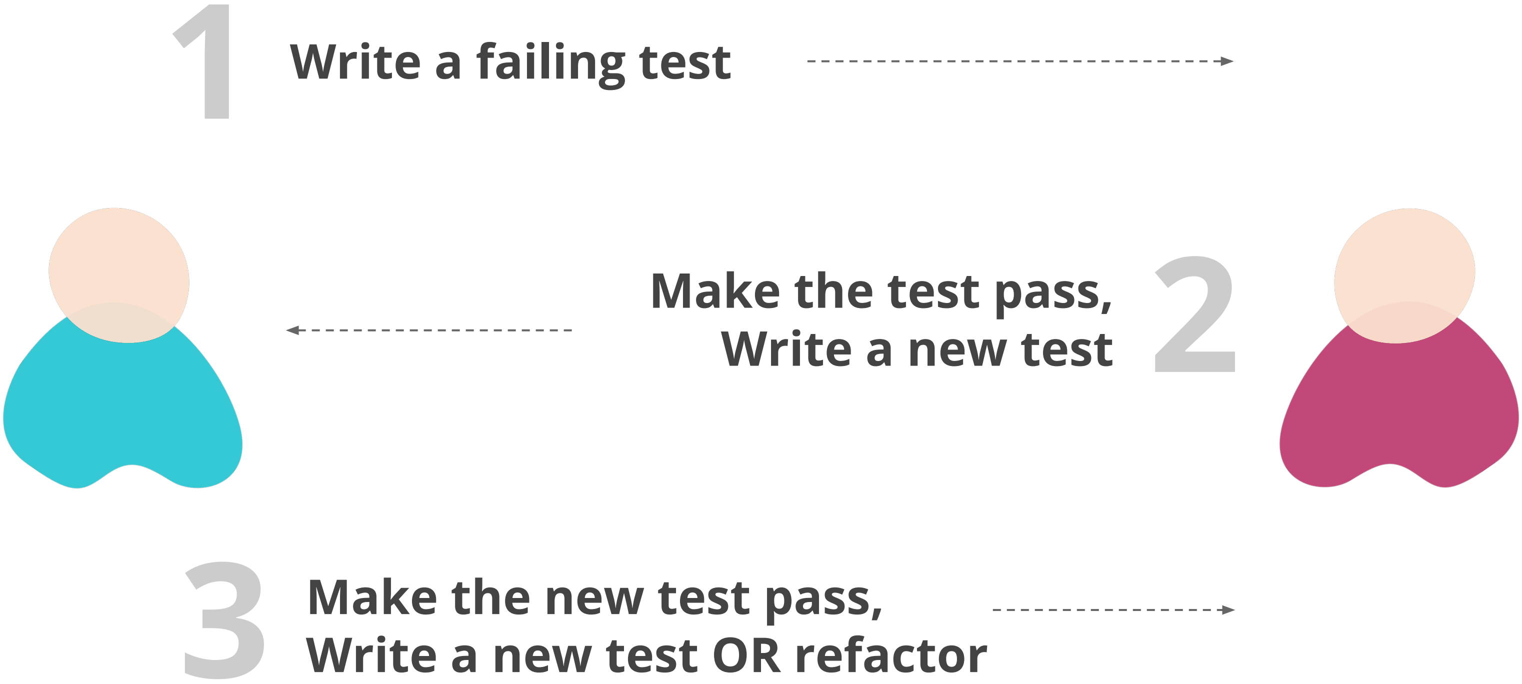
GREEN (Pass)

Write only enough code for it to **pass**

REFACTOR

Improve code quality

TDD + PAIR PROGRAMMING (PING PONG METHOD)



1 Write a failing test

Make the test pass,
Write a new test 2

3 Make the new test pass,
Write a new test OR refactor

(continue back and forth with step 3)