

# Grundlagen der Testgetriebebenen Entwicklung

11.00 - 12.30

# Agenda

- Grundlagen
- Erste Tests im Swift Package
- Einstellungen in Xcode
- Unsere erste Beispiel-App

# 3 Regeln

Robert C. Martin

1. You are not allowed to write any production code unless it is to make a failing unit test pass.
2. You are not allowed to write any more of a unit test than is sufficient to fail; and compilation failures are failures.
3. You are not allowed to write any more production code than is sufficient to pass the one failing unit test.

<http://www.butunclebob.com/ArticleS.UncleBob.TheThreeRulesOfTdd>

# 3 Regeln

1. Du darfst keinen Produktionscode schreiben, ausser er dient dazu einen fehlschlagenden Test erfolgreich zu machen.
2. Du darfst nicht mehr in einem Unit-Tests schreiben als nötig ist, diesen fehlschlagen zu lassen; ein Test der nicht kompiliert ist ein fehlschlagender Test
3. Du darfst nicht mehr Produktionscode schreiben als nötig ist, diesen einen fehlschlagenden Tests erfolgreich zu machen.

# Demo

# 3 Regeln

1. Du darfst keinen Produktionscode schreiben, ausser er dient dazu einen fehlschlagenden Test erfolgreich zu machen.
2. Du darfst nicht mehr in einem Unit-Tests schreiben als nötig ist, diesen fehlschlagen zu lassen; ein Test der nicht kompiliert ist ein fehlschlagender Test
3. Du darfst nicht mehr Produktionscode schreiben als nötig ist, diesen einen fehlschlagenden Tests erfolgreich zu machen.

# Rot - Grün - Verbesserung

(Red - Green - Refactor)

Red: Schreibe einen Test der fehlschlägt

Green: Schreibe Code, sodass der Test durchläuft

Refactor: Verbessere den Code

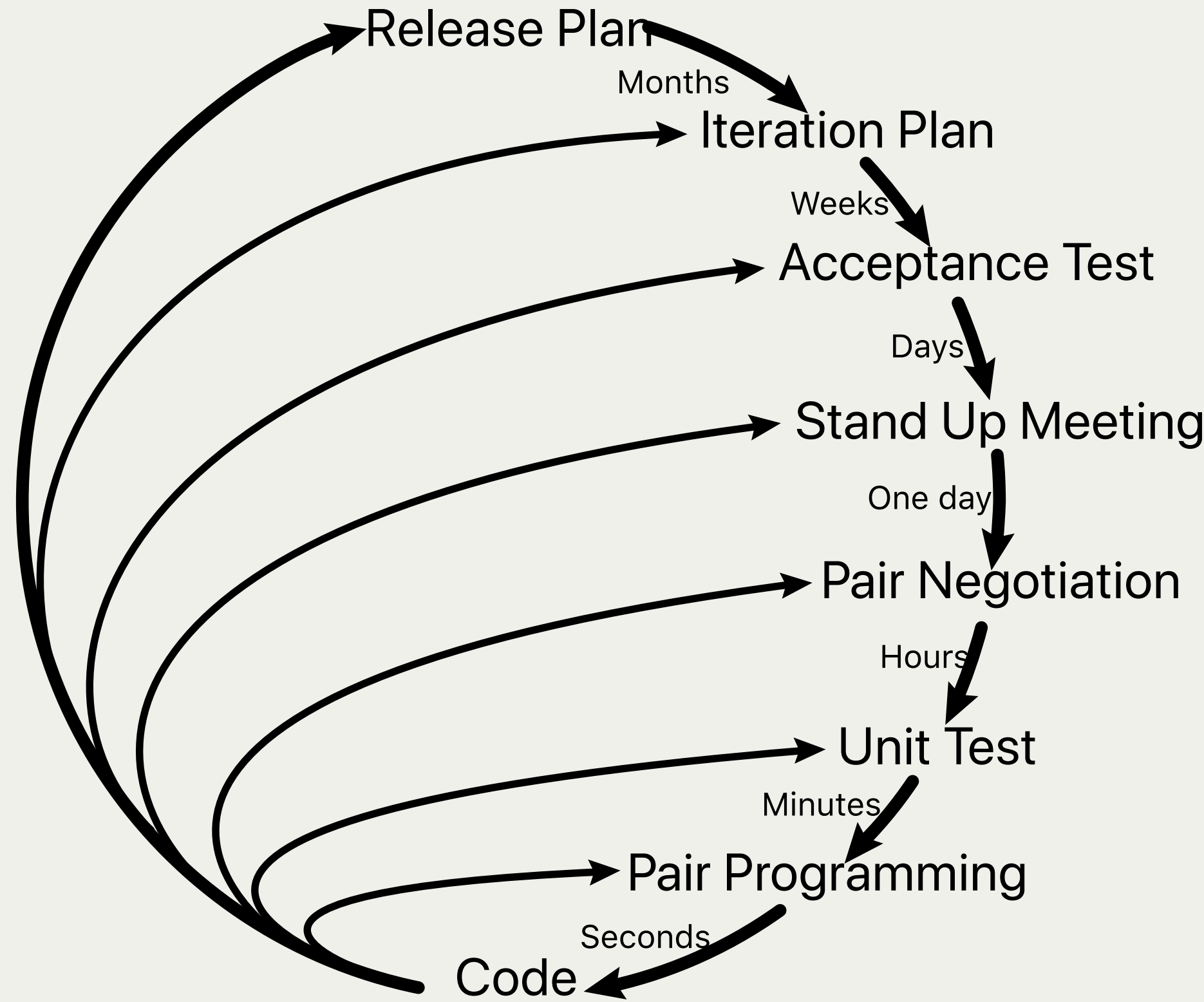
# Once upon a time

- Kent Beck et. al.
- 1996
- Chrysler Comprehensive Compensation System
- payroll project



DonWells  
(CC BY-SA 3.0)

# Planning/Feedback Loops



Numbers 9

2533

2532

2531

2530

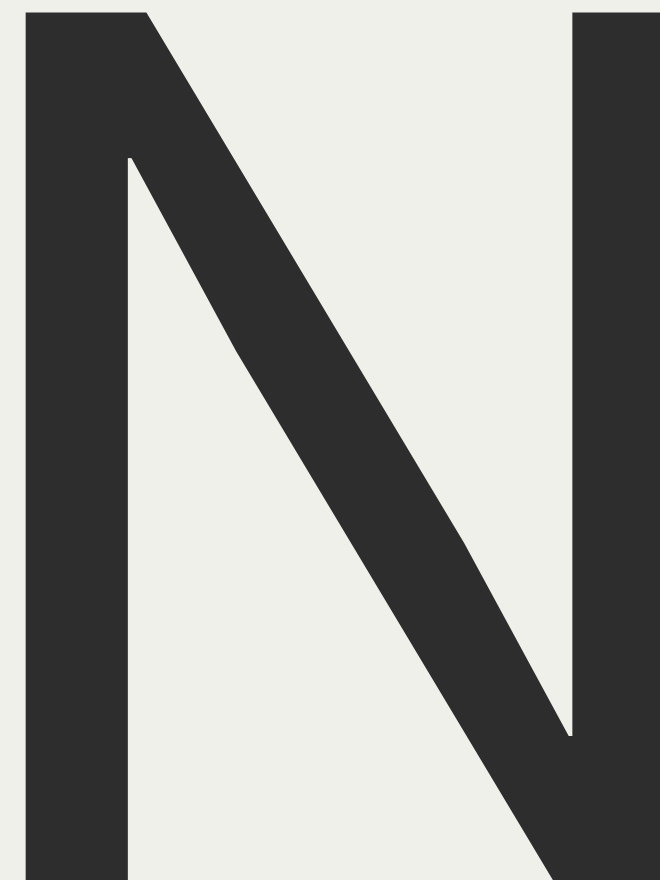
2529

2533



Date:

Age:





# Vor der Pause einen fehlschlagenden Test!

<https://leanpub.com/tddiosgimmethocode/c/macdev>

