

รายงาน

เรื่อง

Test Driven Development

จัดทำโดย

นายภควัต	วงศ์แหวน	55010904
นายวุฒิพงศ์	ศรีภูม้น	55011179
นายศุภกร	วโรดม	55011222
นายอรรถกร	เทพสิทธิ์	55011432
นางสาวอัญชัญ	ฉันทวรลักษณ์	55011454

เสนอ

ผศ. ดร.ชุตีเมษฏ์      ศรีนิลทา

รายงานนี้เป็นส่วนหนึ่งของวิชา Software Engineering

ภาคเรียนที่ 1 ปีการศึกษา 2557

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

## Test Driven Development

### TDD คืออะไร?

TDD คือ กระบวนการพัฒนาซอฟต์แวร์ ที่อาศัยหลักการทำซ้ำในวงจรพัฒนาที่สั้นๆ โดยสิ่งแรกๆ ที่ developer จะทำคือเขียน automated test case (ที่จะ fail) เพื่อที่จะกำหนดคุณสมบัติของฟังก์ชันใหม่ที่ต้องการ และจากนั้นจะใช้วิธีการเขียนโค้ดให้น้อยที่สุดที่จะผ่าน test แล้วจึงค่อยปรับปรุงโค้ดให้ได้มาตรฐานอีกที

### ประวัติของ TDD

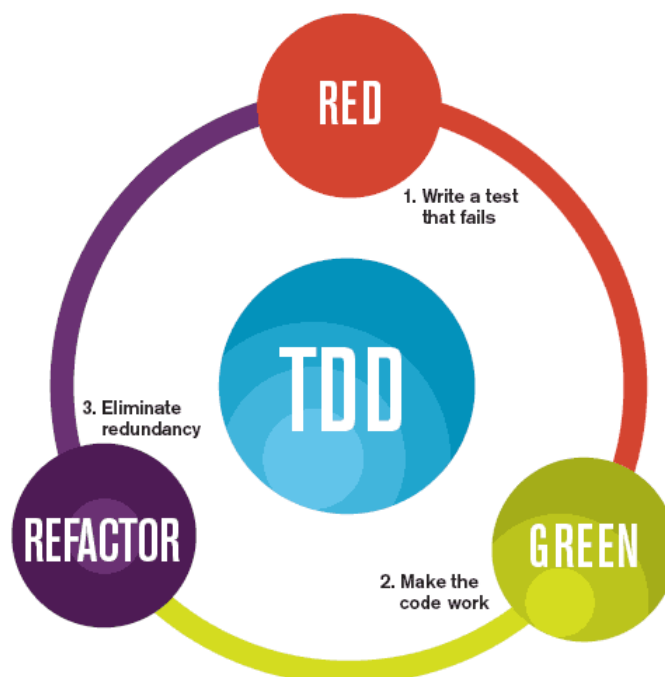
Kent Beck ผู้คิดค้น TDD ซึ่งใช้กับ Extreme Programming ในปี 90 และเริ่มใช้ตัว test ที่สร้างขึ้นมาเอง

JUnit framework พัฒนาโดย Beck และ Erich Gamma ในปี 97 ทำให้การเขียน test และ เรียกใช้งานคำสั่ง test นั้นง่ายมากขึ้น

จากนั้น JUnit ก็ได้ถูกพัฒนาไปใช้กับภาษาอื่นๆอีกหลายภาษา เช่น .NET, C++, Python, PHP เป็นต้น ซึ่งเป็นที่เรียกกันในชื่อว่า xUnit frameworks

### ขั้นตอนการทำ TDD

1. Quickly add a test - เขียน test case
2. Run the tests - รัน test เพื่อให้เห็นว่าไม่สามารถรัน test case ใหม่ได้
3. Update the code - แก้ไขโค้ดให้พอเท่าที่สามารถรัน test ได้
4. Re-run tests - รัน test และแก้ไขโค้ดจนกว่าจะผ่าน test
5. Refactor - แก้ไขโค้ดให้ดีขึ้น และได้มาตรฐานและลบสิ่งที่ไม่จำเป็นออก
6. Repeat - ทำขั้นตอน 1 ถึง 5 ซ้ำ



The mantra of Test-Driven Development (TDD) is "red, green, refactor."

ตัวอย่าง

```
@Test
public void should_be_able_to_add_two_number_together(){
    int firstNumber=1;
    int secondNumber=2;
    Calculator calculator = new Calculator();

    int result = calculator.add(firstNumber,secondNumber);

    assertEquals(3,result);
}
```

เขียน Test case

1/1 test cases complete. 0 passes and 1 fails.

1/1 test cases complete. 1 passes and 0 fails.

ในตอนแรกการรัน test ควรจะ fail

1/1 test cases complete. 0 passes and 1 fails.

1/1 test cases complete. 1 passes and 0 fails.

จากนั้นให้แก้ไขโค้ดจนสามารถผ่าน test ได้

เครื่องมือที่ใช้ในการ Test



## ข้อดีของ TDD

ผู้พัฒนาจะพัฒนาโดยมีความเข้าใจถึงผลลัพธ์สุดท้ายที่ต้องการอยู่ในใจเสมอและที่สำคัญคือผลลัพธ์นั้นจะเป็นสิ่งที่ลูกค้าเห็นและเข้าใจตรงกันด้วยนอกจากนั้นการแบ่งการทดสอบเป็นส่วนย่อยๆ ยังทำให้สามารถ roll back ได้ง่ายหากโปรแกรมเกิดความผิดพลาดหรือพัฒนาแล้วใช้ไปแล้วไม่ชอบใจ และด้วยลักษณะที่เน้นให้โปรแกรมผ่าน Test Case ก่อน แล้วค่อยมาปรับแต่งที่หลังทำให้การพัฒนารวดเร็วขึ้น เช่นถ้าหากมีส่วนต่อไปที่รอการพัฒนาอยู่ ก็สามารถทำต่อไปได้เลยในขณะที่ส่วนแรกกำลังทำการปิดกวดในขั้นตอน Refactor โดยพอสรุปเป็นข้อย่อยๆดังนี้

- + เมื่อทำงานบนโค้ดแบบ test-driven จะเพิ่มความมั่นใจให้กับ develop
- + ลดการเกิดข้อบกพร่อง และการที่ต้องกลับมาแก้ไขใหม่
- + โค้ดมีคุณภาพที่ดีขึ้น (แยกเป็นส่วนๆ, เกี่ยวข้องกันน้อย และยืดหยุ่น)
- + เพิ่มความสามารถในการ maintain โค้ด
- + สามารถ refactor ได้โดยไม่ต้องกลัวว่าจะทำให้ส่วนอื่นพัง
- + ทำให้สามารถตรวจพบความเข้าใจผิดหรือความกำกวมของ requirement ตั้งแต่ระยะแรกๆ
- + มีโค้ดแต่ละส่วนที่น้อยลงด้วยการออกแบบที่เรียบง่าย
- + การตรวจปัญหาระหว่างการติดต่อกันของ Object สามารถทำได้ง่ายขึ้น
- + ลดการ manual testing
- + ได้รับข้อเสนอแนะ, ผลตอบรับที่ไว ว่าโค้ดเขียนได้ถูก

## ข้อเสียของ TDD

หากแบ่งส่วนของโปรแกรมไม่เหมาะสม เช่น ย่อยเกินไปก็จะเสีย overhead ในการสร้าง automated test case เยอะ และถ้าหากแบ่งส่วนของโปรแกรมใหญ่เกินไป ก็จะทำให้การสร้าง automated test case มีความซับซ้อน และมีความเสี่ยงที่จะเกิดความผิดพลาดที่ตัว Test Case เอง โดยพอสรุปเป็นข้อย่อยๆดังนี้

- ใช้เวลานานในการพัฒนา เพราะต้องใช้ในการพัฒนาโปรแกรมมาก
- ต้องทำความเข้าใจ code ที่มากขึ้นเพราะมีการทดสอบที่แตกต่างกัน
- มีราคาสูงขึ้นในการทดสอบระบบที่มีขนาดใหญ่
- บางครั้งเสียเวลาไปกับการทดสอบกรณีที่ไม่สำคัญ
- ประสิทธิภาพขึ้นอยู่กับประสบการณ์และวินัยของทีมพัฒนา
- ยากที่จะนำเอา TDD ส่วนเดิมไปไปสร้างบน Project ใหม่
- ลดความสามารถในการเขียน code อย่างรวดเร็วและเกิดข้อมูลที่ไม่จำเป็น
- ความยากที่จะประเมินต้นทุนจำนวน TDD เวลาเทียบกับจำนวนบันทึก
- ยากที่จะประเมินว่าเสียเวลาให้กับ TDD เท่าไหร่ หรือว่าช่วยประหยัดเวลาในการทำ TDD
- มีผลการทดสอบที่ลดลง หากมีโปรแกรมทำงานช้า
- มีราคาในการติดตั้งสูง