



SWEEPING IT UNDER THE RUG

By Team 6



TEAM 6

01.

What is sweeping it
under the rug?

02.

Problem of
Development System

03.

Solution for
Development System



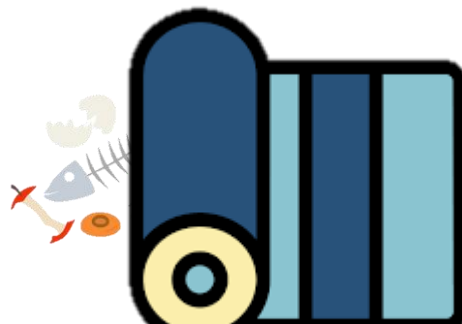
01

What is sweeping it
under the rug?

การซ่อนปัญหาไว้ใต้พรมคืออะไร

หมายถึง

การซ่อนปัญหา หรือเก็บปัญหาไว้เป็นความลับ
แทนที่จะแก้ปัญหา



“

ตัวอย่าง การซ่อนปัญหาไว้ได้พรม

”

ภัยพิบัติเชียร์โนบีล

ถูกขนานนามว่าเป็นอุบัติเหตุทางนิวเคลียร์ครั้งเลวร้ายที่สุดในประวัติศาสตร์



ที่มา : <https://www.hebergementwebs.com>

ภัยพิบัติเชอร์โนบีล (2)



ที่มา : <https://www.sarakadeelite.com/better-living/chernobyl-nuclear-power-plant>

วิธีการซ่อมแทนการแก้ไข



ภาพจิตรกรรมฝาผนัง

ที่มา : <https://pantip.com/topic/35291591>

วิธีการซ่อนแทนการแก้ไข (2)

เด็ก ๆ มักจะซ่อนของเล่นที่ซำรุดไว้ในตู้ ดึกกว่าวางไว้กระจัดกระจายบนพื้นห้อง เพื่อไม่ให้แม่พบเห็น



02

Problem of Development System

ปัญหาในการพัฒนาระบบ

“ ปัญหาที่มักจะพบในการพัฒนาระบบ ”

Spaghetti Code คืออะไร

❖ หมายถึงซอร์สโค้ด (Source code)

- ที่ไม่มีโครงสร้างและดูแลรักษายาก
- ที่พันกัน ทำให้การควบคุมภายในโปรแกรมทำได้ยากมาก
- ที่ไม่เป็นไปตามกฎเกณฑ์ใด ๆ ถูกเขียนขึ้นตามที่คุณเขียนต้องการเขียน



ตัวอย่าง Spaghetti Code

```
1  i=0
2  i=i+1
3  PRINT i; "squared=";i*i
4  IF i>=100 THEN GOTO 6
5  GOTO 2
6  PRINT "Program Completed."
7  END
```

Spaghetti Code

VS

```
1  ✓ FOR i=1 TO 100
2      PRINT i;"squared=";i*i
3  NEXT i
4  PRINT "Program Completed."
5  END
```

Structured Code

ข้อดีของ Spaghetti code

❖ ได้แก่

- นักพัฒนาสามารถทำสิ่งที่ตนเองต้องการได้เร็วขึ้น
- ไม่ต้องเรียนรู้แนวคิดใหม่ ๆ
- ไม่มีกฎเกณฑ์

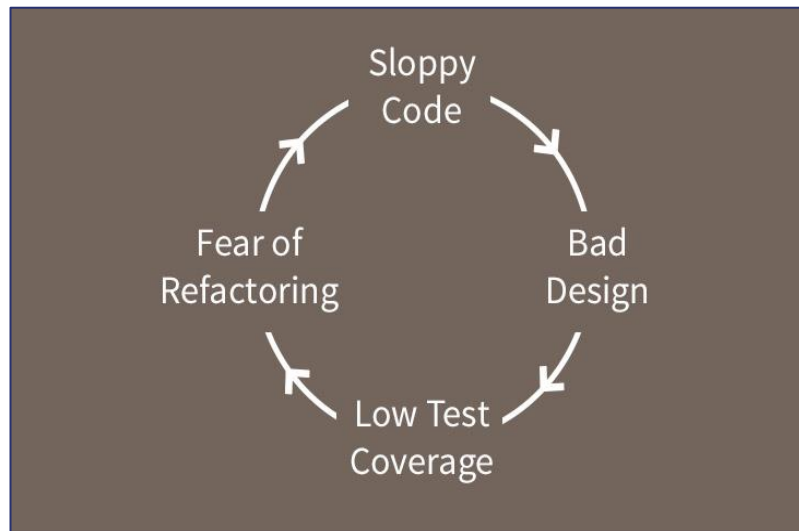
ข้อเสียของ Spaghetti code

❖ ได้แก่

- ยากต่อการใช้รูปแบบสถาปัตยกรรม
- ยากต่อการใช้รูปแบบการออกแบบ
- โปรแกรมทำงานได้ยากต่อการอ่าน และดูแลรักษา
- ยากที่จะเพิ่มคุณสมบัติเพิ่มเติมให้กับโปรแกรม หรือรวมเข้ากับระบบอื่น ๆ
- ยากที่จะเพิ่มประสิทธิภาพซอร์สโค้ด (Source code)

The Big Ball of Mud

ถ้าการพัฒนาโดยการออกแบบที่ไม่ดี หรือการเขียน Code ที่ซับซ้อน เข้าใจยาก สมาชิกในทีมไม่นำ Source code ส่วนนั้น ๆ มาใช้ต่อ (Reuse) ซึ่งเรียกโค้ดแบบนั้นว่า **“BIG BALL OF MUD”**



03

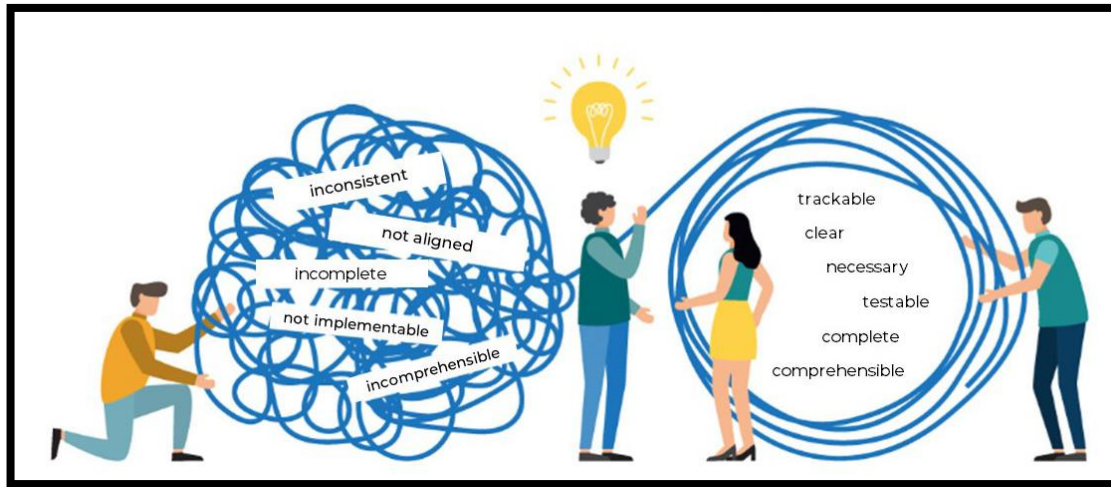
Solution for Development System

วิธีการแก้ปัญหาสำหรับ
การพัฒนาระบบ

สิ่งที่จำเป็นในการแก้ไขระบบที่มีความซับซ้อน

ความเข้าใจ : Requirements Source code

กำลังใจ : เพราะการแก้ไขแค่เล็กน้อยอาจจะทำให้ต้องแก้ไขทั้งระบบ



การรับมือกับระบบที่มีความซับซ้อน

- การค้นหาส่วนต่าง ๆ ที่ดูซับซ้อน
- การแยกส่วนที่ซับซ้อนออกจากระบบ
- การออกแบบ Interface ของส่วนที่ซับซ้อนใหม่
- การ Comment ส่วนที่ซับซ้อนไว้

สรุป

- บางครั้งคุณต้องรู้ก่อนว่าปัญหาเกิดขึ้นที่ไหน คุณจึงจะสามารถแก้ไขได้ง่ายขึ้น
- หากแก้ไขปัญหานั้นไม่ได้อย่างครบถ้วนก็ให้ซ่อนไว้ แล้วจึงแก้ไขในภายหลัง
- การระบุพื้นที่ที่เป็นปัญหา และปิดล้อมได้แล้ว จะทำให้สามารถแบ่งมอดูลแก้ไขได้

“

Ironically, inscrutable code can persist
because it is difficult to comprehend.

”

Foote & Yoder