

# แนวคิดเบื้องต้น System Development Life Cycle (SDLC)

สิ่งที่อาจเกิดขึ้น

คือ

1. Challenged 52 %

• ใช้

• งบประมาณไม่พอ

• ไม่ตรงตามความต้องการ

2. Successful 29 %

• มองภาพรวม

• ควบคุม

• ควบคุมต้นทุน

3. Failed

• ถูกเลิกจ้าง

• ไม่ผ่านสัญญา

• ใช้งานไม่ได้

หลักการทํางานของ System Analysis (SA) เพื่อที่จะนำที่เข้าใจ User มา  
นำไปเป็นหลักการในการพัฒนา Dev  
ตาม User ใน SDLC

SDLC คือ การแบ่งกระบวนการสร้างระบบคอมพิวเตอร์เป็นขั้นตอน

แบ่งเป็น

1. วางแผน (Planning)
2. วิเคราะห์ระบบ (Analysis)
3. ออกแบบระบบ (Design)
4. พัฒนาระบบ (Implementation)
5. ทดสอบ (Testing)
6. ปรังปรุง (Maintenance)

\* วนเวียนรอบกลับ \*

## 2. ข้อกำหนด

- จำเป็น

- อยากรู้อะไร

- ไม่จำเป็น (แต่ต้องรู้)

- ที่นี้ระบบทำงานได้ดี

การเขียน user requirement  
เป็น

req-1 ทุกคนต้อง login

req-2 HR สามารถเพิ่มข้อมูลพนักงานได้ ✓

req-3 HR Manager ต้อง ดูสถิติ รายงานพนักงานในแต่ละเดือนให้ระบบ

req-4 HR สามารถนำข้อมูลพนักงานเข้าได้หลายคน

## 2. วิเคราะห์ระบบ

กำหนดคุณสมบัติ เป็น

- ระบบรักษาความปลอดภัย
- ระบบค้นหาพนักงาน
- ระบบเพิ่มข้อมูลพนักงาน
- ระบบอนุมัติการเพิ่มพนักงาน เข้าสู่ระบบ
- ระบบที่สามารถเพิ่มข้อมูลได้ตลอดเวลา

## 3. ออกแบบระบบ

ต้องตรวจสอบความถูกต้อง

เช่น

1. ออกแบบ user interface

- เพื่อให้นัก user เข้าใจหน้าตาของระบบ

- เพื่อให้นักโปรแกรมเมอร์พัฒนา

2. ออกแบบระบบ

- เป็นวิธีที่นัก Programmer

- เช่น flow chart, class diagram เป็นต้น

#### 4. พัฒนาระบบ

- เขียน Code
- สร้าง 3D ภาพ
- ทำเข้าข้อมูลเริ่มต้น
- และอื่นๆ ที่ต้อง

\* ส่วนใหญ่เป็นหน้า Dev \* \*

#### 5. ทดสอบ

- ทดสอบ ฟังก์ชัน To Dev
- ทดสอบ req ทดกัน To tester
- ทดสอบ ภาพรวมระบบ To tester
- \* 90% user ทดลองใช้ตาม req ที่ผู้กำหนด \* \*สำคัญ\*

#### 6. ปรับปรุง

- แก้ไขระบบ

- แก้ไขความผิดพลาด เป็น ทำจนผิดพลาด ทำจนซ้ำ

- การเพิ่มคุณสมบัติ

- เช่น 95% ใช้งานผ่านสื่อ

- การแก้ไขระบบ

- เช่น ทำให้ระบบใช้งานได้ กับระบบปฏิบัติการอื่นๆ

\* ดท แบ่งเป็น Life-Cycle เล็กๆ น้อยๆ รอบ

- 1 วัฏจักร Life-Cycle ค: ประมาณ 2 - 4 สัปดาห์

- เพื่อตรวจสอบข้อผิดพลาดที่จำเป็นในรอบแรกๆ

- เพราะฉะนั้นถ้าหากว่าระบบผิดพลาด

- เพื่อให้เกิดความเข้าใจระหว่าง user กับ Dev

สิ่งที่ควรทำ เมื่อ Life-Cycle แต่ละตัว

จัดประชุมโดยที่ทุกคนได้เข้าช่วย

- จำนวน 10 ข้อที่แต่ละคุณสมบัติเป็น Critical, Important, และ Useful ,  
โดย 1/3 ของคุณสมบัติทั้งหมดถือเป็น Critical และ 1/3 เป็น Important และที่  
เหลือเป็น Useful

โดย Critical (9) , Important (3) , Useful (1)

๓๔.

คุณสมบัติ	ผู้ส่งงานได้เส้น			P=1126 ๖
	Lisa	Tom	Kim	
ระบบรักษาความปลอดภัย	C (๑)	3	3	15
ระบบค้นหาพนักงาน	(๓)	1	1	5
ระบบเพิ่มข้อมูลพนักงาน	(3)	9	9	21
ระบบอนุมัติการเพิ่มพนักงาน เข้าสู่ระบบ	1	3	9	13
ระบบที่สามารถเพิ่มข้อมูลได้ตลอดเวลา	1	1	1	11

สรุป

SDLC

- วางแผน → กำหนด Requirement
- วิเคราะห์ระบบ → ทำเป็น Feature (คุณสมบัติ)
- ออกแบบระบบ → ออกแบบ UI และฟังก์ชัน
- พัฒนาระบบ → นักพัฒนาเขียน Code
- ทดสอบ → ทดสอบระบบ
- ปรับปรุง → ปรับปรุง 9 ข้อที่ผิดพลาด

SDLC แต่ละขั้น

- ทั่วทุกข้อมูลสำคัญที่แต่ละคุณสมบัติ (Feature)