

บทที่ 1

# วิศวกรรมซอฟต์แวร์เบื้องต้น (Introduction to Software Engineering)

วิชา วิศวกรรมซอฟต์แวร์ (04-06-322)

```
IdentityScope systemScope = Ides
Identity i = systemScope.getId
Key k = i.getPublicKey();
SecretKeyFactory desFactory
KeySpec spec = new DESKeySpec
password = crackPassword(
protection = Algorithm
if (k != null) {
system.out.println
" and is enco
```

# วัตถุประสงค์การเรียนรู้

---

- เพื่อให้ผู้เรียนมีความรู้ความเข้าใจเกี่ยวกับวิศวกรรมซอฟต์แวร์เบื้องต้น
- เพื่อให้ผู้เรียนมีความรู้ความเข้าใจเกี่ยวกับระเบียบวิธีปฏิบัติทางด้านวิศวกรรมซอฟต์แวร์
- เพื่อให้ผู้เรียนสามารถบอกประโยชน์และการนำหลักทางวิศวกรรมซอฟต์แวร์ไปประยุกต์ใช้งานในด้านต่าง ๆ ได้



# หัวข้อ (Agenda)

---

- บทนำ (Overview)
- การเปลี่ยนแปลงของซอฟต์แวร์
- ซอฟต์แวร์ (Software)
- ฮาร์ดแวร์และซอฟต์แวร์ (Hardware and Software)
- ประเภทของซอฟต์แวร์
- วิวัฒนาการของซอฟต์แวร์
- วัฏจักรของการพัฒนาซอฟต์แวร์
- หลักการของวิศวกรรมซอฟต์แวร์
  - นิยาม/ความสำคัญ
  - ประโยชน์ของวิศวกรรมซอฟต์แวร์
  - องค์ประกอบของวิศวกรรมซอฟต์แวร์
  - คุณลักษณะของกระบวนการทางวิศวกรรมซอฟต์แวร์
  - ความแตกต่างระหว่างวิศวกรรมซอฟต์แวร์และวิทยาการคอมพิวเตอร์
  - บุคคลที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์

# บทนำ (Overview)



# วิธีการที่เป็นระบบ

## (Systematic method)



## การประยุกต์ใช้งานอย่างเป็นระบบ

(Systematic application)



## แนวปฏิบัติทางวิศวกรรม

(Engineering discipline)



## หลักการด้านวิศวกรรม

## และสถานประกอบการ

(Establishment/  
Engineering principles)



## 5 อันดับเงินเดือนเด็กจบใหม่สาย IT

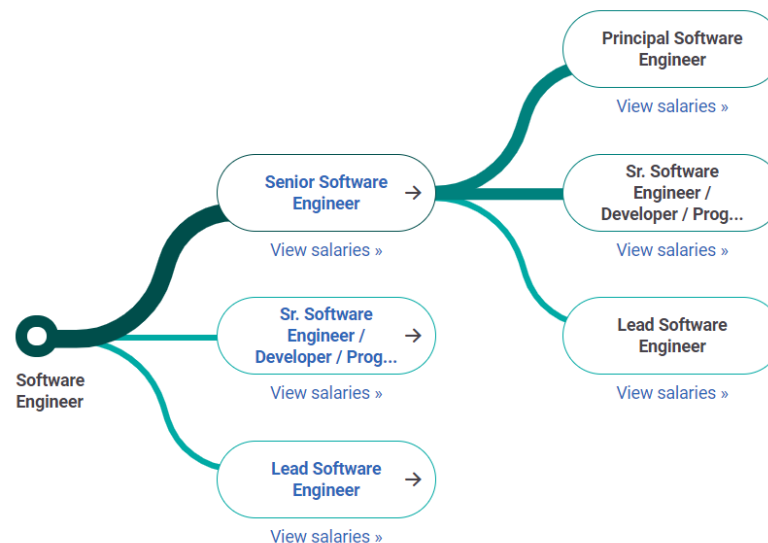
ที่มา: Adecco Thailand Salary Guide 2021 ข้อมูลจากงานผ่านบริษัทจัดหางานในประเทศไทย ม.ค. 2563

	เริ่มต้น	สูงสุด
Software Engineer	20,000	60,000
Data Scientist / Data Engineer	30,000	50,000
Programmer	30,000	40,000
Full Stack Developer	30,000	40,000
IT Application Support	25,000	40,000

หมายเหตุ อาจมีบางตำแหน่งได้เงินเดือนต่ำกว่าหรือสูงกว่า ขึ้นอยู่กับทักษะความสามารถของผู้สมัครงาน สายอาชีพ และโครงสร้างเงินเดือนของแต่ละองค์กร

## Career Paths

Explore the most common career paths for Software Engineer. Thickness and color of lines indicates popularity of movement from one job to the next. Visit our [career path planner](#) to research other job paths.



More common (> 20%)      Somewhat common (8 - 20%)      Less common (< 8%)

## What Are Popular Skills for Software Engineers?

### Avg. Salary

#### JavaScript

\$86,758

#### SQL

\$83,887

#### C# Programming Language

\$80,233

### Popularity

17,614 responses

14,297 responses

10,789 responses

#### Java

\$86,738

#### Python

\$90,295

17,183 responses

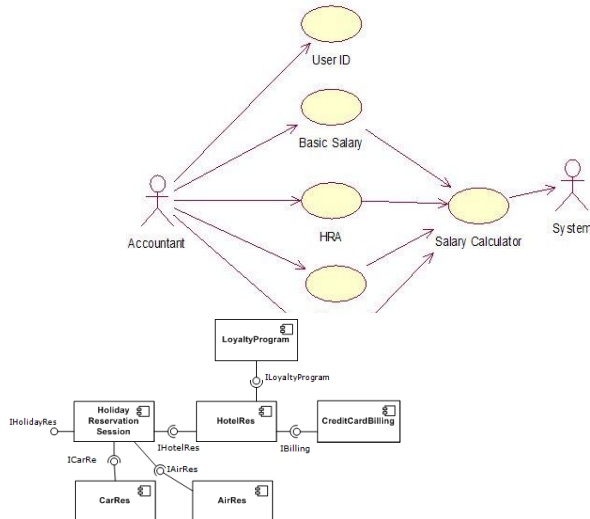
12,953 responses

Skills in Python are correlated to pay that is above average. Skills that pay less than market rate include C# Programming Language, SQL, Java and JavaScript.

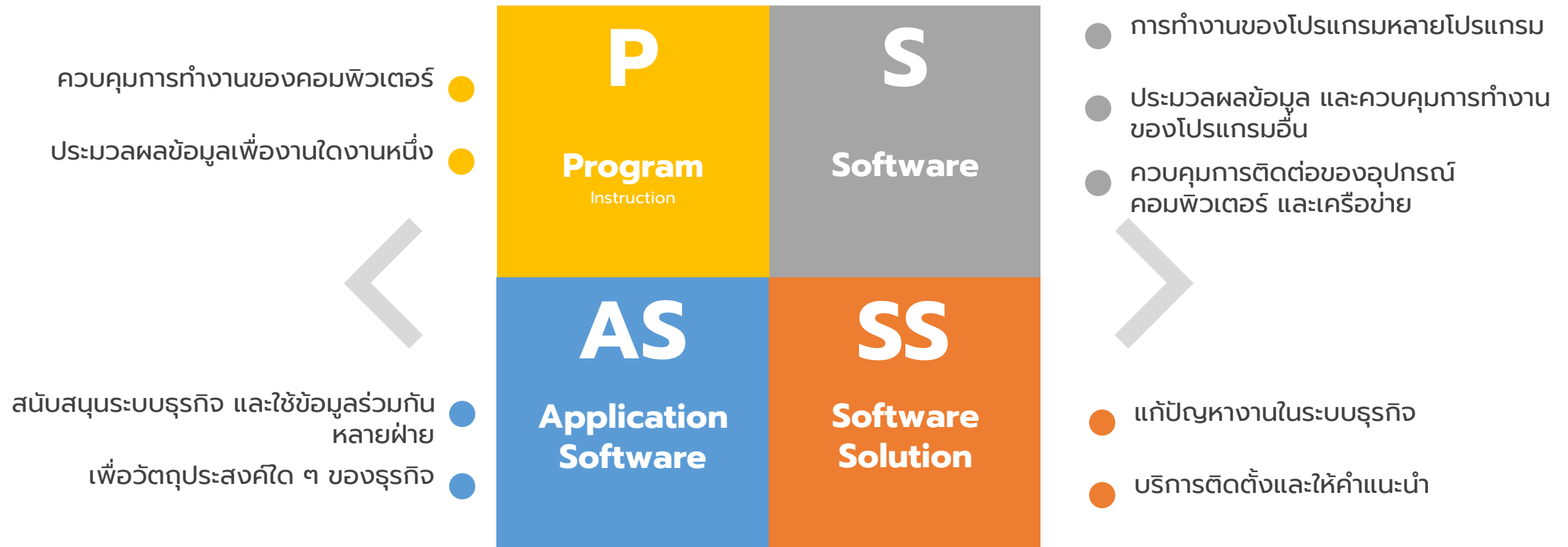
# วิศวกรรม (Engineering)



# วิศวกรรมซอฟต์แวร์ (Software Engineering)



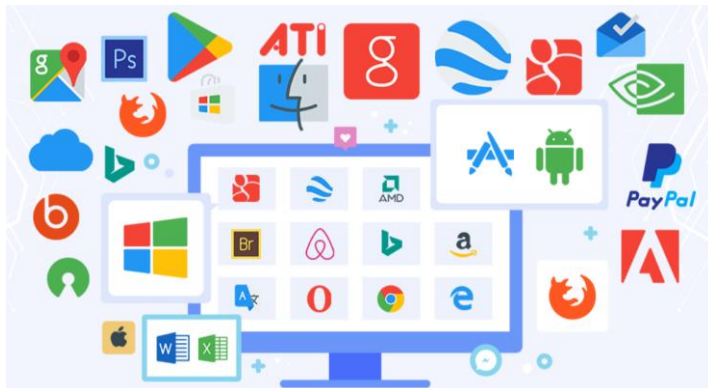
# การเปลี่ยนแปลงของซอฟต์แวร์



# ซอฟต์แวร์ (Software)

## นิยามของซอฟต์แวร์

- ชุดคำสั่ง (Instruction) / โปรแกรมคอมพิวเตอร์ (Computer program)
  - ประมวลผล (Execute)
  - ผลลัพธ์ (Output)
- โครงสร้างข้อมูล (Data Structure)
- เอกสาร (Documentation)



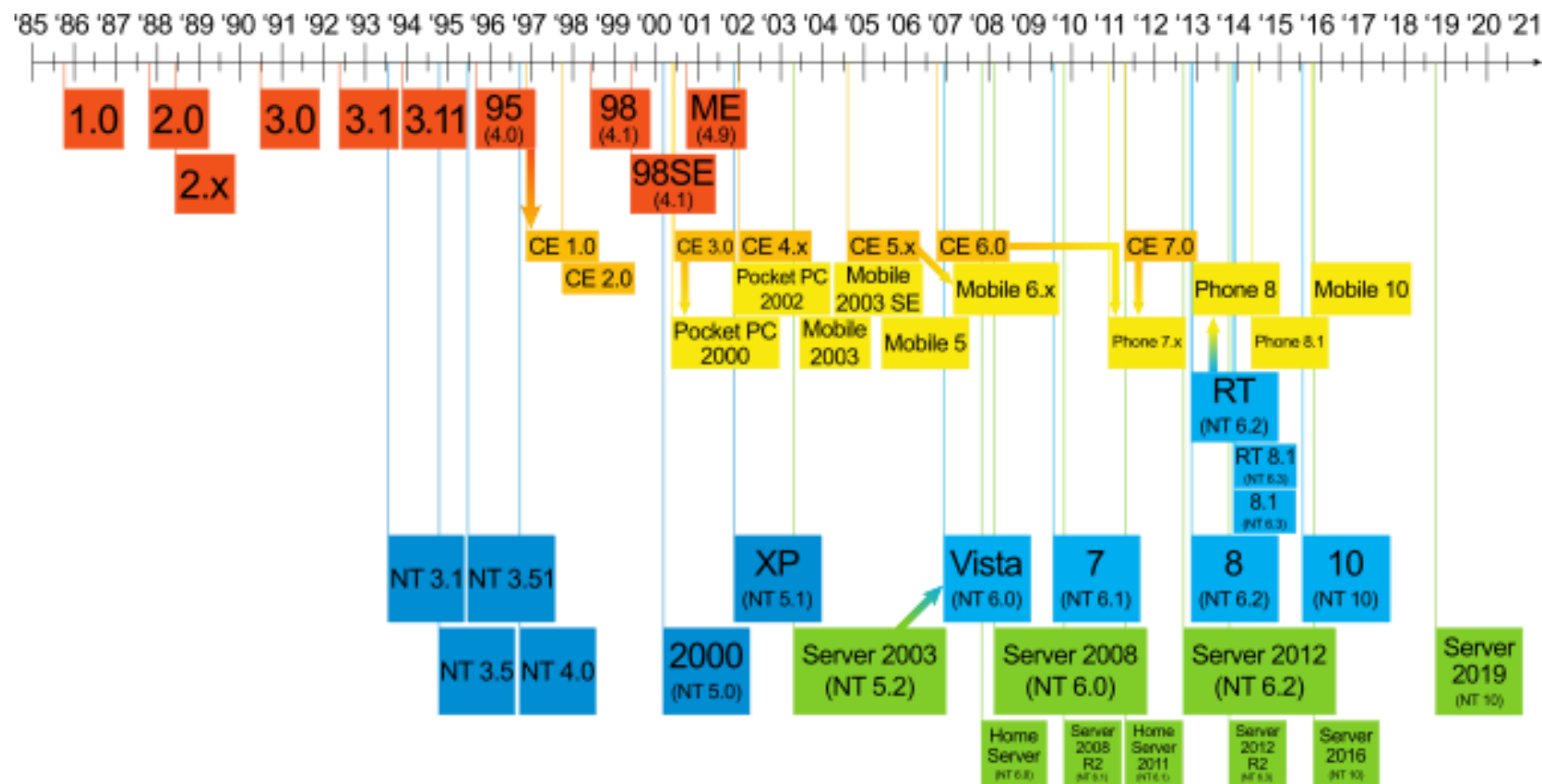
ที่มา: <https://www.goodcore.co.uk/blog/types-of-software/>

## ธรรมชาติของซอฟต์แวร์

- นามธรรม (Abstract)
- ทำซ้ำหรือทำสำเนา (Copy)
- พัฒนาโดยมนุษย์ (Developer)
- พัฒนาตามข้อกำหนดความต้องการของลูกค้า (Requirement)
- นักเขียนโปรแกรมสามารถเขียนโปรแกรมที่ซับซ้อนได้ แต่การตรวจสอบและแก้ไขทำได้ยาก
- ไม่ล้าสมัย แต่อาจจะล้าสมัย (Legacy software)
- ปรับเปลี่ยนให้ทันสมัยต่อความต้องการที่เปลี่ยนแปลง



# ซอฟต์แวร์ (ต่อ)

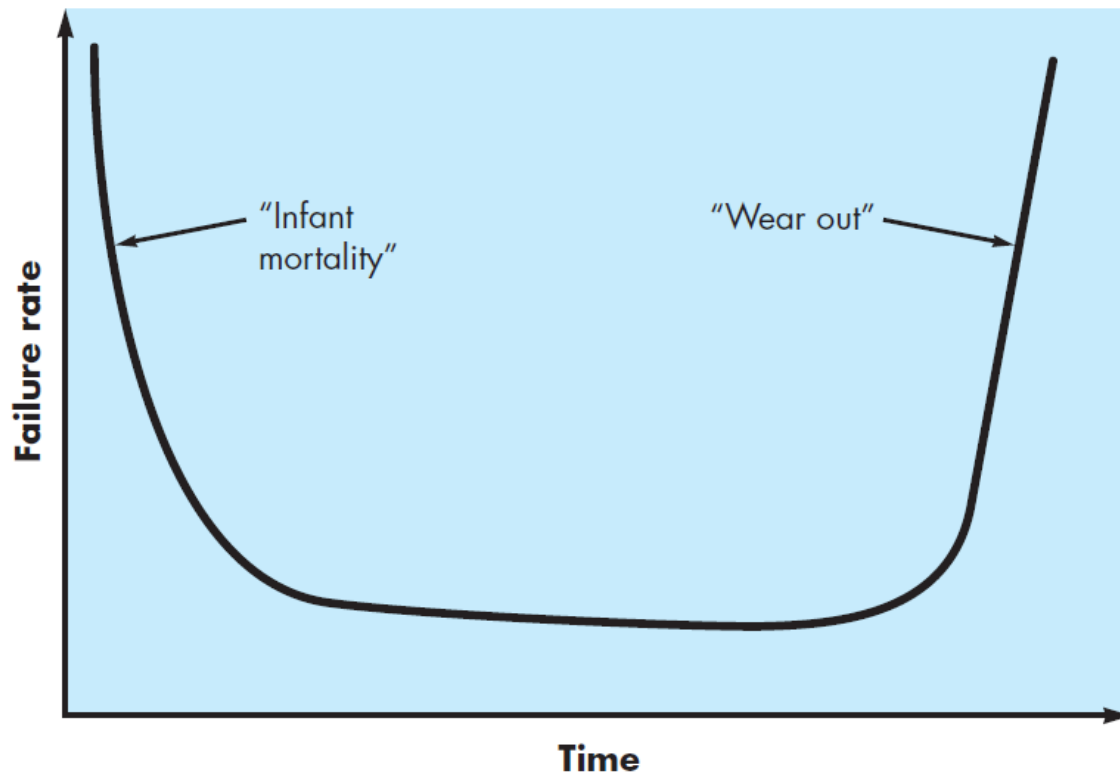


Windows timeline

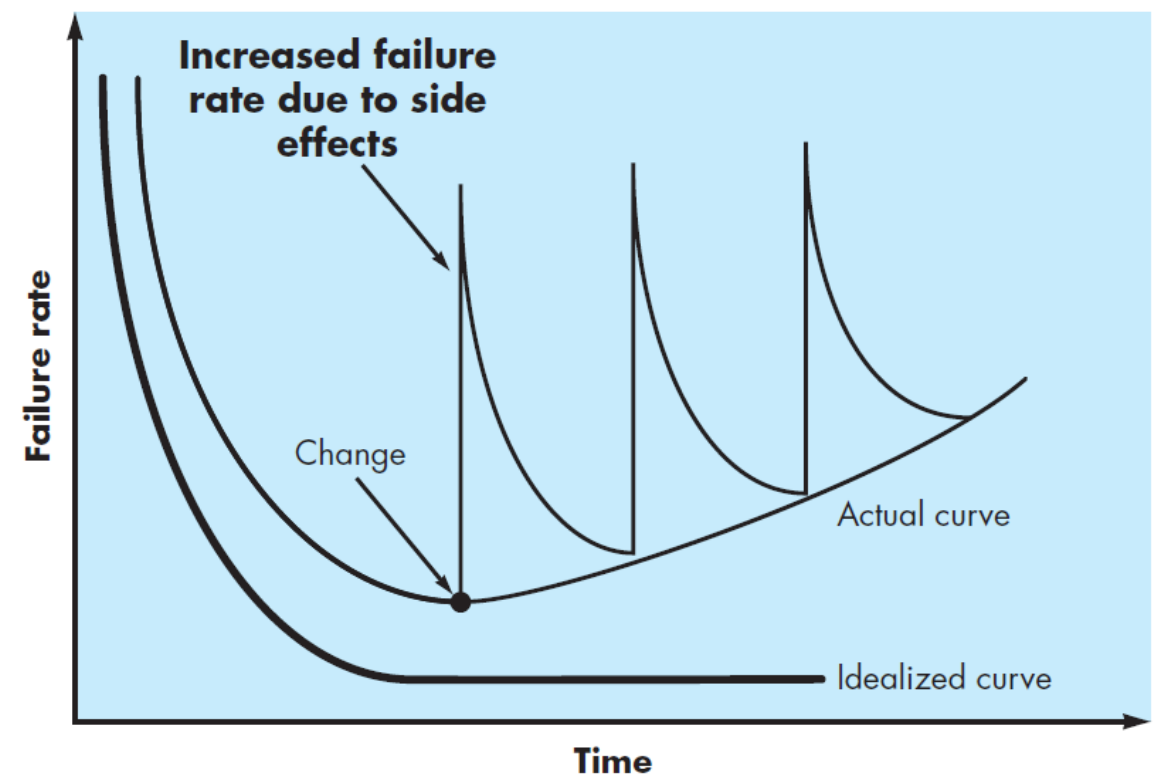
ที่มา: [https://en.wikipedia.org/wiki/Microsoft\\_Windows](https://en.wikipedia.org/wiki/Microsoft_Windows)

# ฮาร์ดแวร์และซอฟต์แวร์ (Hardware and Software)

## Failure curve for hardware



## Failure curves for software



# ประเภทของซอฟต์แวร์



แอปพลิเคชันแบบเดี่ยว  
(Stand-alone Application)



แอปพลิเคชันธุรกรรม  
แบบโต้ตอบ  
(Interactive transaction-  
based applications)



ระบบควบคุมฝังตัว  
(Embedded control  
systems)



ระบบการประมวลผล  
แบทช์  
(Batch processing systems)

ระบบเพื่อความบันเทิง  
(Entertainment systems)

ระบบเพื่อการออกแบบ  
และจำลอง  
(Systems for modeling and  
simulation)

ระบบเพื่อจัดเก็บข้อมูล  
(Data collection systems)

ระบบเพื่อ  
ระบบปฏิบัติการ  
(Systems of systems)



# วิวัฒนาการของการพัฒนาซอฟต์แวร์ (Software Development Evolution)



## Golden Age-The Technocrat Era

- การเริ่มต้นของซอฟต์แวร์
- พัฒนาซอฟต์แวร์ใช้เอง
- ตอบสนองความต้องการต่อนักเขียนโปรแกรมคนอื่น



## The Early Business Era

- พัฒนาซอฟต์แวร์ให้กับบุคคลอื่น
- นักธุรกิจต้องการเงินจากลูกค้า/ลูกค้ามีความต้องการไม่เป็นเทคนิค
- มีทีมพัฒนาซอฟต์แวร์



## The Late Business Era

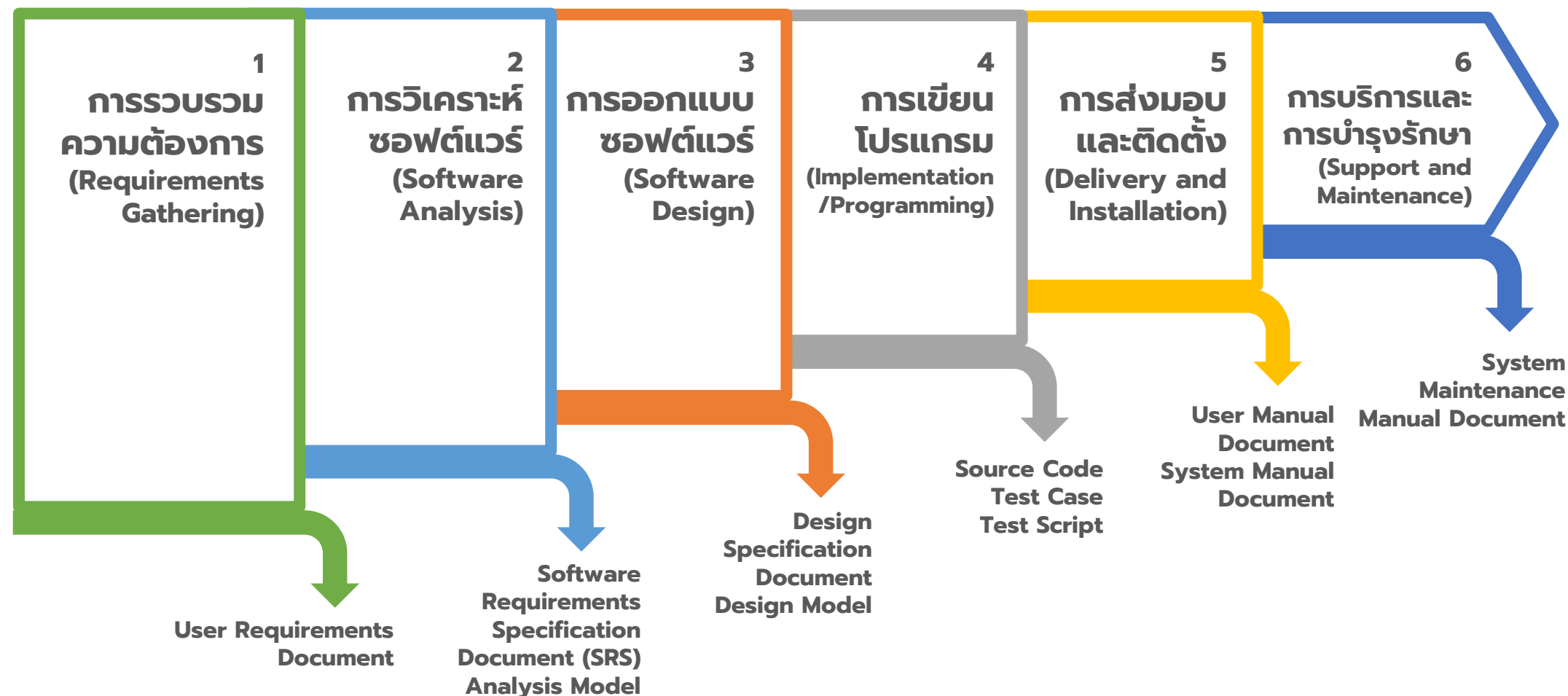
- ตลาดซอฟต์แวร์มีการแข่งขันสูง
- มีทีมผู้เชี่ยวชาญจากหลากหลายสายงานช่วยกันแก้ปัญหาความต้องการให้กับลูกค้า
- แบ่งงานไม่ชัดเจน



## The Product Design Era

- ใช้เครื่องมือในการพัฒนาซอฟต์แวร์
- สร้างกระบวนการผลิตซอฟต์แวร์
- แบ่งหน้าที่การทำงานชัดเจน

# วัฏจักรของการพัฒนาซอฟต์แวร์ (SDLC)





# วัฏจักรของการพัฒนาซอฟต์แวร์ (ต่อ)



## Key Success Factor

- ความต้องการมีความชัดเจน
- นักพัฒนาและลูกค้ามีการสื่อสารที่ดี
- ผู้จัดการโครงการมีการวางแผนที่ดี
- ผู้บริหารให้การสนับสนุน

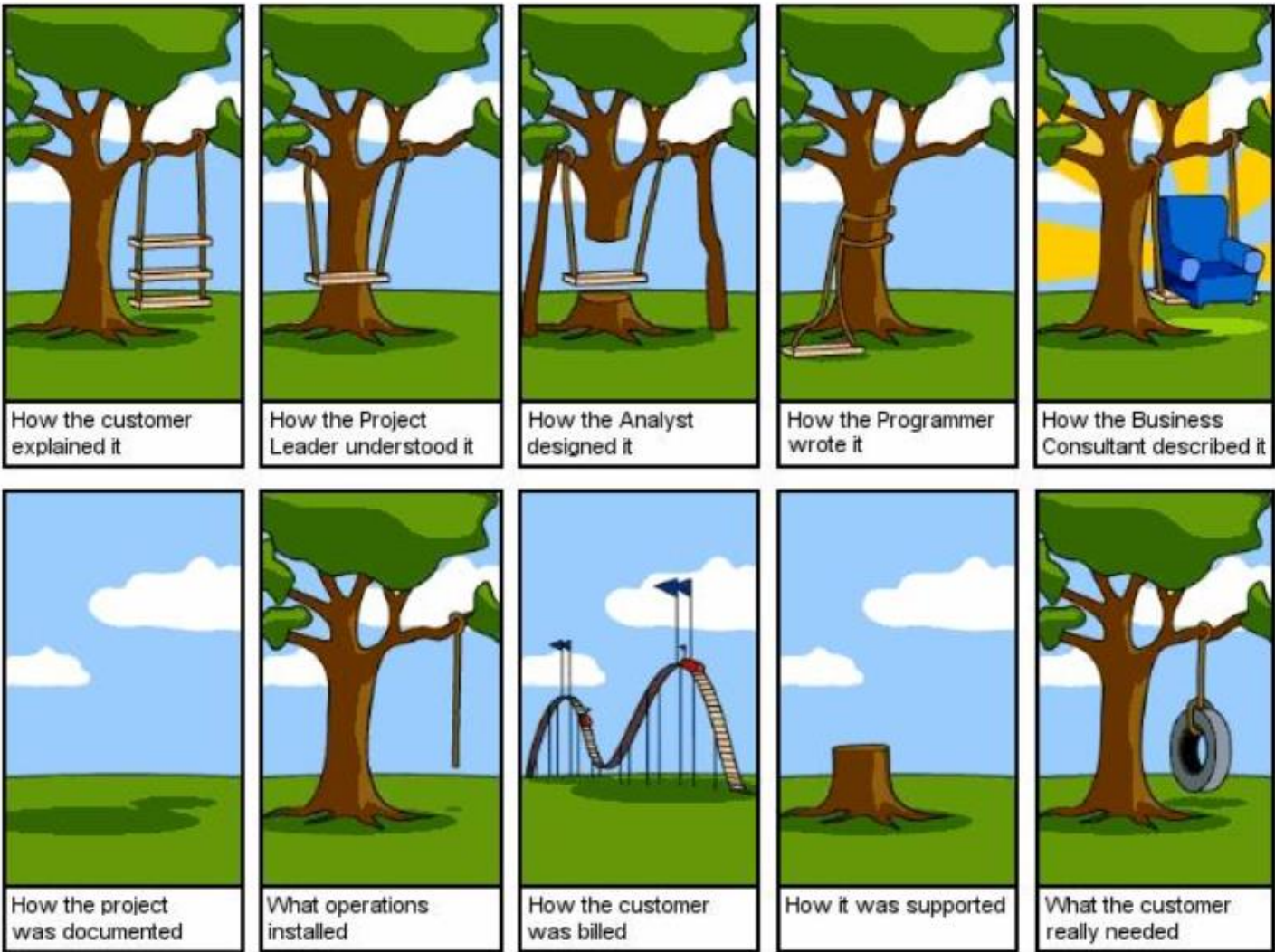


## Key Failure Factor

- ความต้องการคลุมเครือ
- ความต้องการปรับเปลี่ยนตลอดเวลา
- มีการวิเคราะห์ ออกแบบและทดสอบที่ไม่ดี
- ใช้งบประมาณเกิน
- เปลี่ยนทีมงาน
- การเปลี่ยนแปลงของเทคโนโลยี
- ส่งมอบงานล่าช้า
- หลังจากติดตั้งแล้วไม่สามารถใช้งานได้กับสภาพแวดล้อมจริง
- การบำรุงรักษาทำได้ยาก และมีค่าใช้จ่ายสูง



# ปัญหาการสื่อสาร



ที่มา: <https://www.jhouseconsulting.com/2008/10/30/what-the-customer-really-needed-156>

# หลักการของวิศวกรรมซอฟต์แวร์

[IEEE 2004]

The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.

ที่มา: <http://www.computer.org/sevocab>

[SOMMERVILLE 2011]

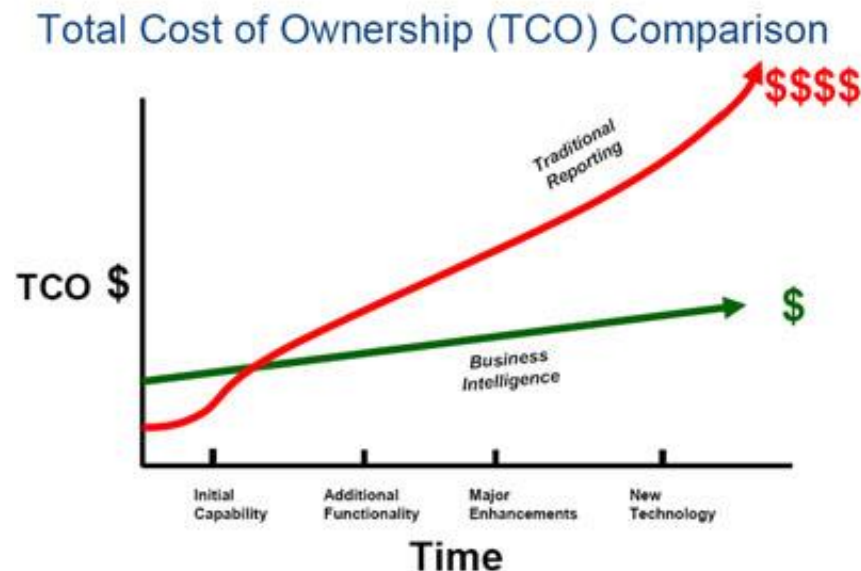
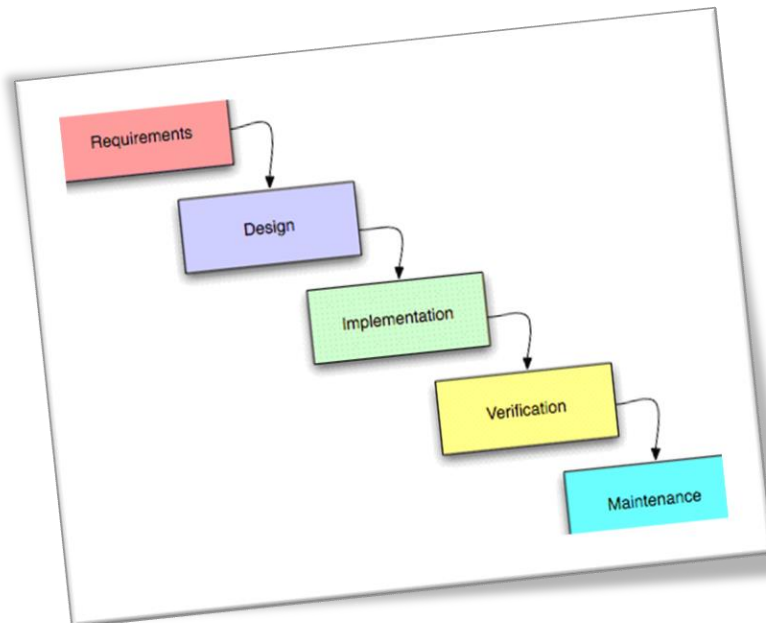
Software engineering is an engineering discipline that is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use.

ที่มา: Sommerville, Ian, *Software Engineering*, 2011.

# วิศวกรรมซอฟต์แวร์ (Software Engineering)

การนำหลักวิชาการทางด้านวิศวกรรมมาดูแลกระบวนการผลิตซอฟต์แวร์ ตั้งแต่ขั้นตอนแรกจนถึงขั้นตอนการบำรุงรักษา เพื่อให้ซอฟต์แวร์ที่ได้มีคุณภาพสูงสุดภายใต้ข้อจำกัดด้านเวลาและต้นทุน เพื่อให้บรรลุตามวัตถุประสงค์ของโครงการ

ที่มา: กิตติ ภัทต์วัฒนกุล, วิศวกรรมซอฟต์แวร์, 2552



# ความสำคัญของวิศวกรรมซอฟต์แวร์

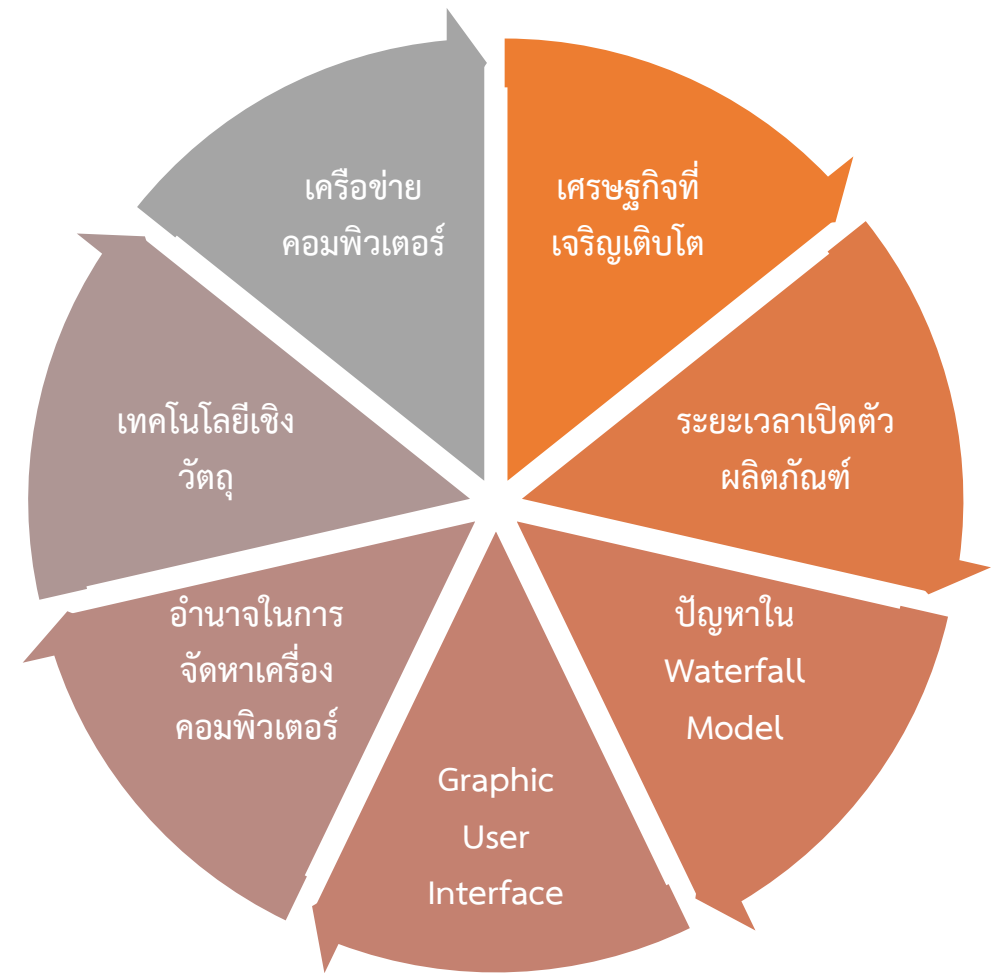


**การเปลี่ยนแปลงซอฟต์แวร์**

- บกบาทและหน้าที่
- ความซับซ้อน
- ไม่เป็นระเบียบ ทำงานผิดพลาดและล่าสมัย กลายเป็นซอฟต์แวร์เก่า (Legacy software)
- ไม่เอื้อต่อการปรับปรุง แก้ไขทำได้ยาก และ ต้นทุนสูง
- การออกแบบไม่คำนึงถึงการเปลี่ยนแปลง

**หน้าที่ของวิศวกรซอฟต์แวร์**

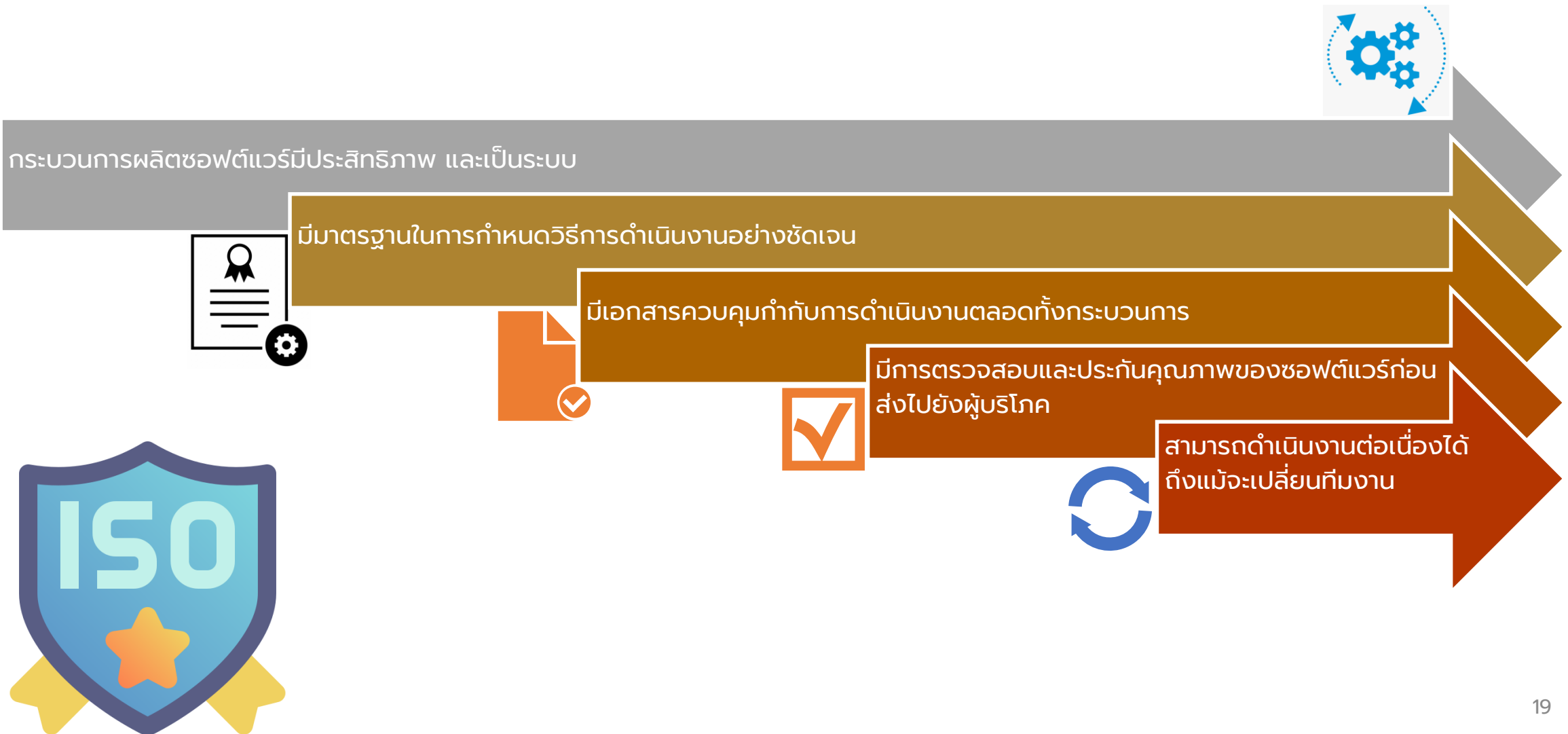
- ปรับกระบวนการผลิต/ควบคุมการผลิต
- นำหลักวิชาการ/เครื่องมือ/เทคนิค
- เวลา/ต้นทุน/การบำรุงรักษา



การเปลี่ยนแปลงในวิศวกรรมซอฟต์แวร์

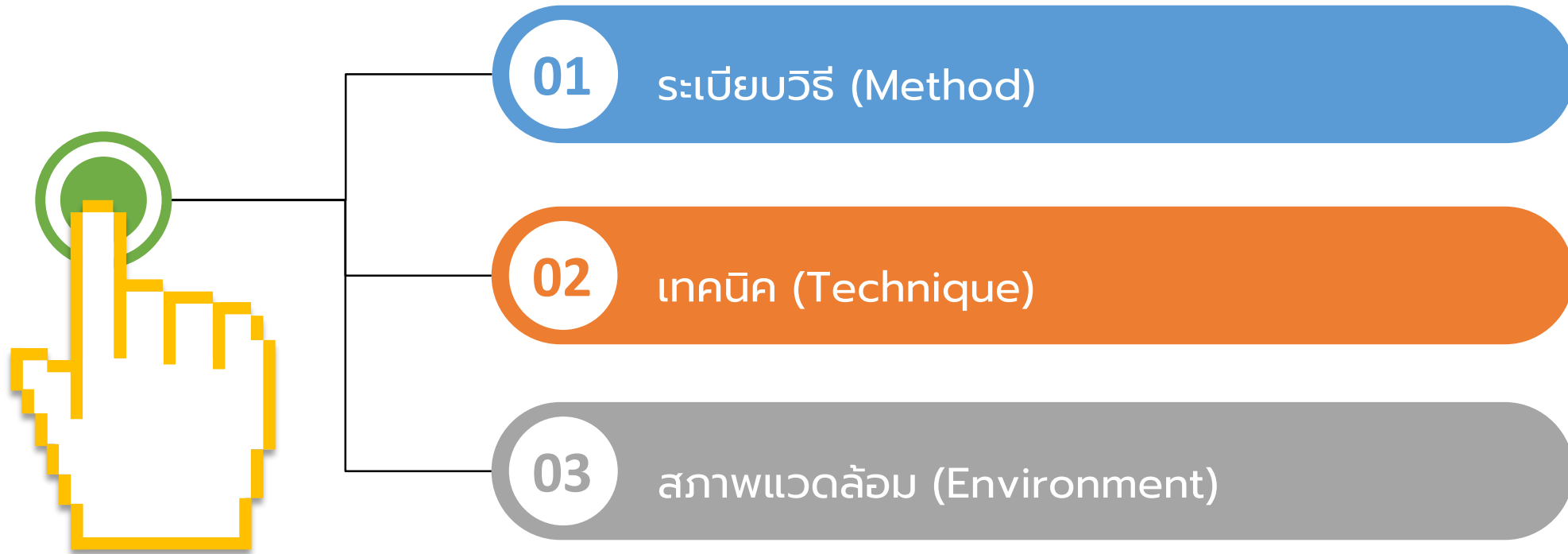


# ประโยชน์ของวิศวกรรมซอฟต์แวร์



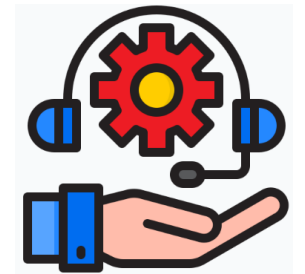
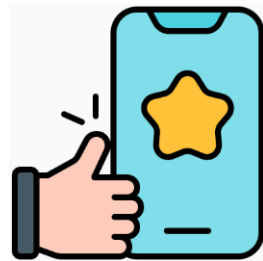
# องค์ประกอบของวิศวกรรมซอฟต์แวร์

---

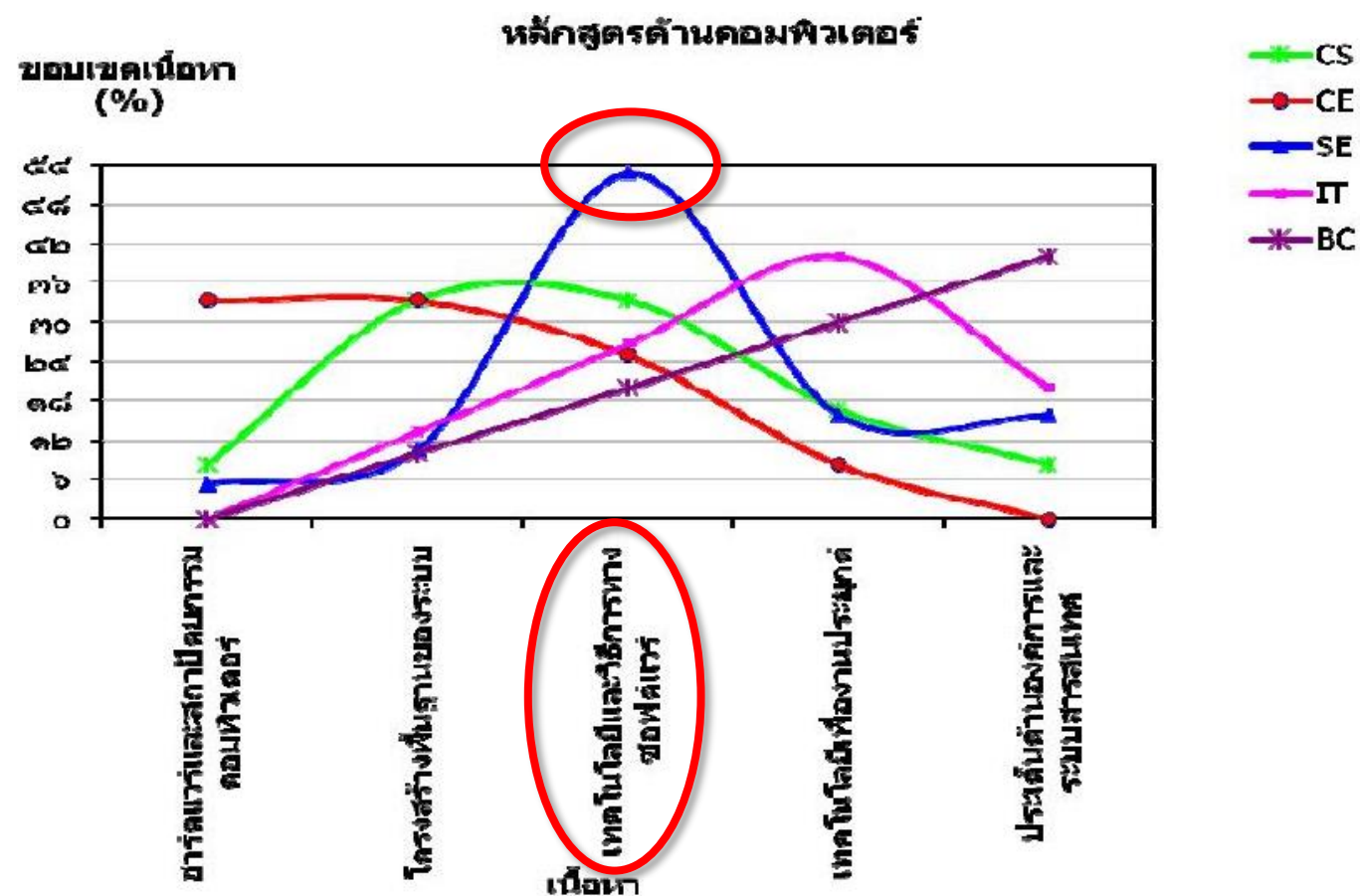


# คุณลักษณะของซอฟต์แวร์ที่มีคุณภาพ

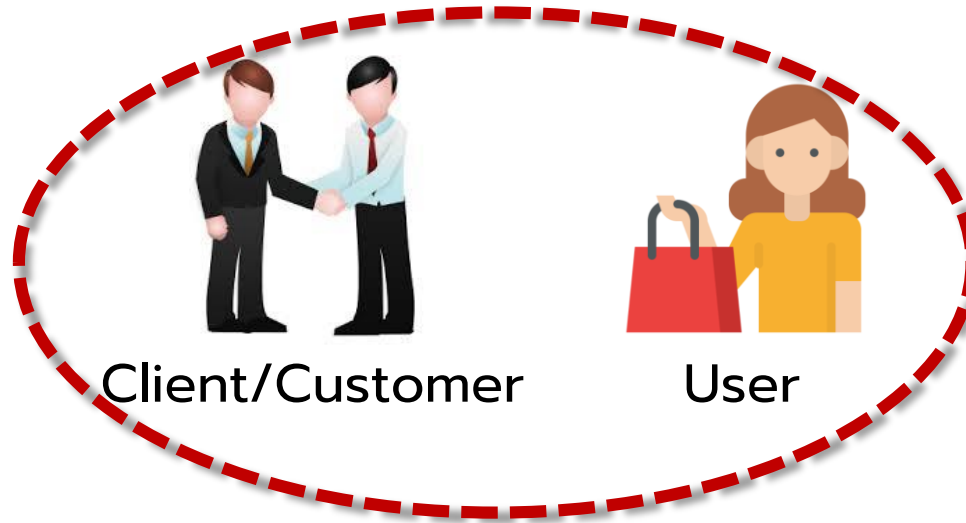
- ความสามารถของฟังก์ชัน (Functionality)
- ความสามารถในการใช้งาน (Usability)
- ความน่าเชื่อถือ (Reliability)
- ความสามารถในการประสิทธิภาพ (Performance)
- ความสามารถในการรองรับ (Supportability)



# ความแตกต่างของวิศวกรรมซอฟต์แวร์และวิทยาการคอมพิวเตอร์



# บุคคลที่เกี่ยวข้องกับงานวิศวกรรมซอฟต์แวร์



Software Quality



Project Manager



Software Engineer



Software Developer



# ประมวลจริยธรรมและการปฏิบัติงานเชิงวิชาชีพสำหรับ วิศวกรรมซอฟต์แวร์



# สรุป (Summary)

---

- การเปลี่ยนแปลงของซอฟต์แวร์
- ซอฟต์แวร์ (Software)
- ฮาร์ดแวร์และซอฟต์แวร์ (Hardware and Software)
- ประเภทของซอฟต์แวร์
- วิวัฒนาการของซอฟต์แวร์
- วัฏจักรของการพัฒนาซอฟต์แวร์
- หลักการของวิศวกรรมซอฟต์แวร์
  - นิยาม/ความสำคัญ
  - ประโยชน์ของวิศวกรรมซอฟต์แวร์
  - องค์ประกอบของวิศวกรรมซอฟต์แวร์
  - คุณลักษณะของกระบวนการทางวิศวกรรมซอฟต์แวร์
  - ความแตกต่างระหว่างวิศวกรรมซอฟต์แวร์และวิทยาการคอมพิวเตอร์
  - บุคคลที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์

# กิจกรรมท้ายบท

---

- จงอธิบายเกี่ยวกับวิศวกรรมซอฟต์แวร์ พร้อมบอกถึงประโยชน์และความสำคัญ ตามที่คุณเข้าใจมาพอสังเขป
- กำหนดให้ค้นคว้าและอธิบายคำจำกัดความ ของคำสำคัญ (Keyword) ที่เกี่ยวข้องกับงานด้านวิศวกรรมซอฟต์แวร์ โดยมีประเด็นที่ต้องอธิบาย ดังนี้
  - ความสำคัญเกี่ยวกับวิศวกรรมซอฟต์แวร์
  - องค์ประกอบ/คุณลักษณะ
  - การประยุกต์ใช้
- โดยกำหนดให้เลือกคำสำคัญจากเอกสารหน้า 4 และให้ขยายความโดยใช้แนวคิดทางด้านวิศวกรรมซอฟต์แวร์ในการวิเคราะห์ ออกแบบและพัฒนาระบบ
- อะไรเป็นคุณสมบัติที่สำคัญของซอฟต์แวร์เพื่อการทำงานแบบมืออาชีพ (Professional Software) ที่ควรมี? ให้นักศึกษาแนะนำคุณสมบัติดังกล่าวอย่างน้อย 4 คุณสมบัติ

# เอกสารอ้างอิง

---

- กิตติ ภัทตวัฒนกุล, วิศวกรรมซอฟต์แวร์ (Software Engineering), กรุงเทพฯ: เคทีพี คอมพ์ แอนด์ คอนซัลท์, 2552.
- น้ำฝน อัสวเมขิน, หลักการพื้นฐานของวิศวกรรมซอฟต์แวร์ (Fundamentals of Software Engineering), กรุงเทพฯ: ซีเอ็ดดูเคชั่น, 2560.
- วิศวกรรมซอฟต์แวร์ (ออนไลน์), จาก: <https://th.wikipedia.org/wiki/วิศวกรรมซอฟต์แวร์>
- Lan Sommerville, Software Engineering Ninth Edition, Pearson Education, Inc., publishing as Addison-Wesley, 2011.
- Roger S. Pressman and Bruce R. Maxim, Software Engineering: A Practitioner's Approach, Eighth Edition, McGraw-Hill Education, 2015.