

Ch 3

Information System Development

Kulachatr C. Na Ayudhya

Chapter Agenda

- Business Strategy
- Information System Planning
- Strategic Planning
- Information Technology
- การได้มาซึ่งระบบสารสนเทศ
 - Off the shelf Software Package
 - Outsourcing
 - In-house System Development
- System Development

Business Strategy ; How do we get there?

Developing Strategy



Business Level Strategy

Porter's Generic Model

Focus Strategy

“เชี่ยวชาญเฉพาะกลุ่ม”

Narrow

Cost Leadership

เป็นผู้นำด้านต้นทุน

Differentiation

“เน้นความ
แตกต่าง”

Board

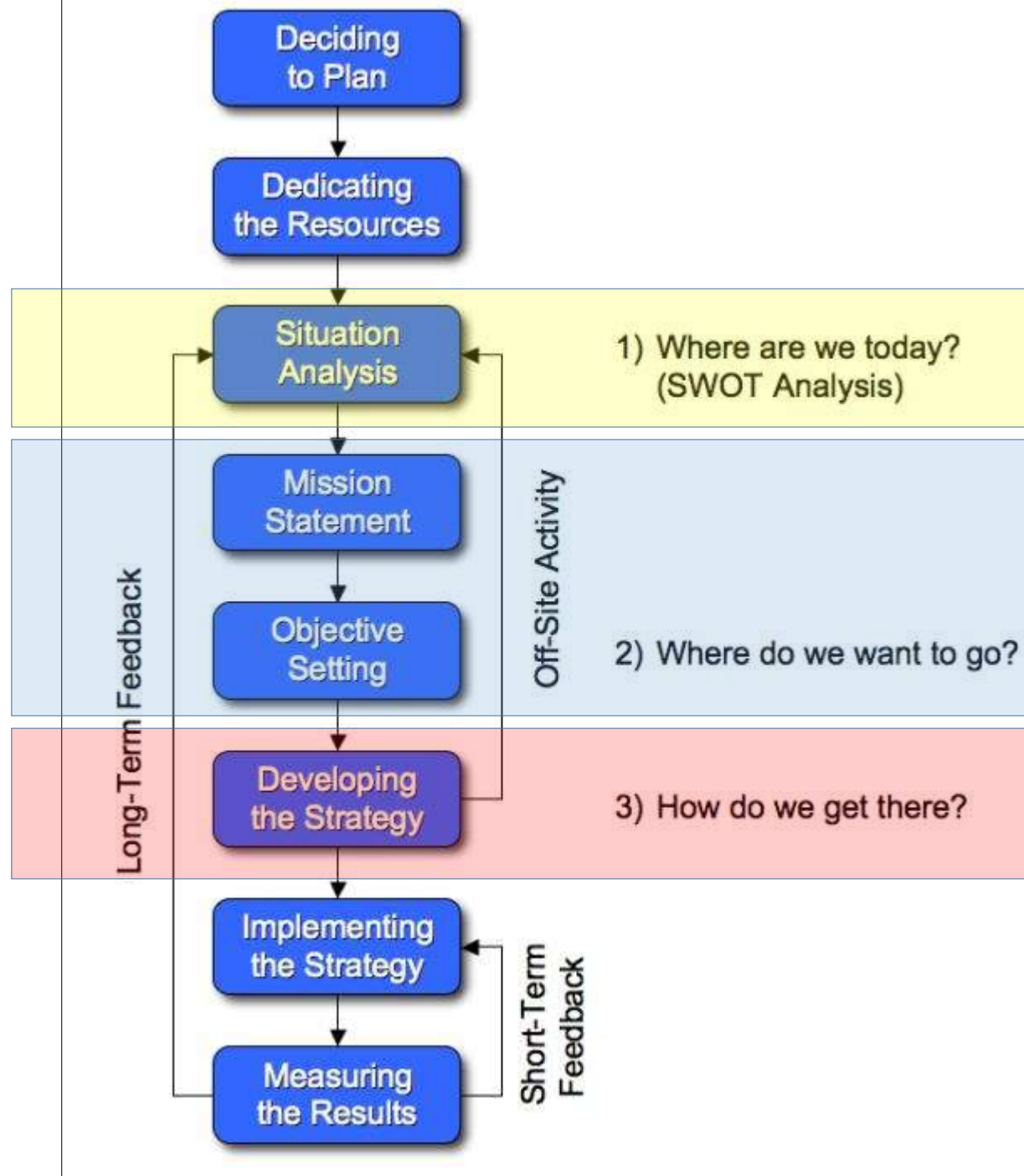
Strategic Planning

“ตอนนี้เราอยู่จุดไหน”

“เราต้องการจะไปถึงไหน”

“เราจะทำอย่างไร”

“ปฏิบัติจริงและตรวจสอบได้”





การเลือกใช้เทคโนโลยีสารสนเทศ

การประยุกต์ใช้ IT ในปัจจุบันและแนวโน้มอนาคต

Now

- Computer
- Database
- E-Commerce
- E-Payment
- Data Security
- Digital Optimization
- Wireless Technology
- Virtual Office
- Messaging Application
- Mobile Technology

Future

Chip
Storage / Cloud
Object Oriented
Self Healing Computer
Quantum Computer
Nano Technology

วิธีการให้ได้มาซึ่งระบบสารสนเทศ

- ซื้อระบบสำเร็จรูปมาใช้ (Application Software Packages)
- การพัฒนาโดยผู้ใช้ (End User Computing)
- ว่าจ้างบริษัทที่ปรึกษาจัดทำระบบให้ (IT-Outsourcing)
- การสร้างทีมงานคอมพิวเตอร์ขึ้นในหน่วยงาน (Information System Unit , In House Information System Development)

IT - Outsourcing

- การว่าจ้างบริษัทอื่น เข้ามาดำเนินการพัฒนาระบบสารสนเทศให้องค์กร รวมถึงการออกแบบ พัฒนาโปรแกรมและบริหารจัดการ ดูแลตลอดจนบำรุงรักษาระบบ ตามความต้องการขององค์กร ภายใต้อำนาจร่วมกัน
 - IT Outsourcing
- ลักษณะของการทำ Outsourcing
 - Strategic Focus
 - Economics Reasons
 - Market Forces
 - Technical Consideration

IT-Outsourcing

• ข้อดีของ Outsourcing

- ต้นทุนการพัฒนาระบบลดลง (Reduce of Cost)
- ได้รับคุณภาพการบริการตามต้องการ (On Demand)
- โปรแกรมสามารถยืดหยุ่นได้ (Flexibility)
- สามารถประมาณการการลงทุนได้ (Cost Budgeting)
- ไม่สูญเสียกำลังคนภายในในการพัฒนา

• ข้อเสียของ Outsourcing

- การควบคุมของระบบสารสนเทศอาจตกอยู่ในอำนาจของบริษัทภายนอก
- ขาดความอิสระในการบริหารระบบ ต้องพึ่งพาบริษัทภายนอกตลอดเวลา
- เสี่ยงต่อการรั่วไหลของข้อมูลสำคัญ

Off-the-shelf Software Package

- การเลือกซื้อโปรแกรมหรือซอฟต์แวร์ที่มีขายในท้องตลาด ซึ่งเป็นโปรแกรมเฉพาะ เช่น โปรแกรมสำเร็จรูปทางบัญชี โปรแกรมการจัดการสินค้าคงคลัง โปรแกรมเกี่ยวกับการขายสินค้าหน้าร้าน ซึ่งอาจต้องมาปรับปรุงบ้างตามความเหมาะสม
- **ข้อดี** ของการซื้อซอฟต์แวร์สำเร็จรูป คือ ลดต้นทุนเงิน เวลาและกำลังคน โปรแกรมส่วนใหญ่ไม่ต้อง Test ใช้งานได้เลย การปรับปรุงเป็นไปโดยง่าย ไม่ยุ่งยากซับซ้อน สามารถทดลองโปรแกรมได้ก่อนซื้อจริง (Trial) และมักได้รับการสนับสนุนจากผู้ขาย
- **ข้อเสีย** อาจไม่ตรงกับความต้องการขององค์กรที่แท้จริง บาง Application ก็ไม่ได้ใช้ เปลืองหน่วยความจำ

• การพิจารณาเลือกซอฟต์แวร์สำเร็จรูป

- Functions : ความสามารถในการทำงาน ตอบสนองความต้องการหรือไม่
- Flexibility : มีความยืดหยุ่นในการใช้หรือไม่ ผู้ขายพร้อมปรับปรุงให้หรือไม่
- User Friendliness : โปรแกรมใช้งานง่าย เข้าใจได้รวดเร็ว
- Database File Characteristic : พิจารณาลักษณะแฟ้มข้อมูล
- Installation Effort : ความยากง่ายของการติดตั้งระบบ
- Maintenance : ความยากง่ายในการดูแลรักษาระบบ หรือ Upgrade
- Documentation : ความยากง่ายในการใช้คู่มือใช้งาน
- Vendor Quality : คุณภาพผู้ขาย ความน่าเชื่อถือและแนวโน้มกิจการ
- Cost : ต้นทุนซื้อหรือราคาคู่กับการลงทุนหรือไม่ มีส่วนควบอะไรบ้าง

End-user Computing :EUC

- วิธี EUC คือการออกแบบพัฒนาและบำรุงรักษาระบบสารสนเทศโดยผู้ใช้ระบบเอง ซึ่งมีแนวโน้มเพิ่มขึ้นมากเนื่องจาก
 - คอมพิวเตอร์ฮาร์ดแวร์-ซอฟต์แวร์ มีความสามารถสูงขึ้น
 - คอมพิวเตอร์แข่งขันสูง ราคาจึงต่ำลง
 - ซอฟต์แวร์มีมากขึ้นและมีเวอร์ชันใหม่ๆ ที่มีประสิทธิภาพมากขึ้น
 - ผู้ใช้ระบบเริ่มมีทักษะความรู้ทางคอมพิวเตอร์มากขึ้น
 - หน่วยงานสารสนเทศในองค์กรไม่สามารถพัฒนาระบบตอบสนองความต้องการได้
 - การพัฒนาระบบในปัจจุบันทำได้ง่ายและรวดเร็วมากขึ้น เพราะมี Tools มากขึ้น
 - ความรู้เชิงธุรกิจของผู้ใช้ระบบมีมากกว่านักโปรแกรมเมอร์
 - ผู้ใช้ระบบสารสนเทศมีความต้องการควบคุมระบบเองทั้งหมด
 - ประหยัดกว่าทุกๆ วิธีที่ผ่านมา

End-user Computing :EUC

- ปัญหาและวิธีการจัดการกับปัญหาของ EUC
 - ปัญหาส่วนใหญ่เกิดจากความเอกเทศของผู้ใช้ระบบ จึงอาจเกิดความไม่มาตรฐานในการพัฒนาระบบ การจัดการ EUC ที่ดีนั้นจะต้อง
 - Coordination การประสานงานระหว่างผู้ใช้ ผู้พัฒนาและเจ้าหน้าที่ IT ในการตั้งมาตรฐานระบบที่จะพัฒนาและเลือกอุปกรณ์ที่ต้องสั่งซื้อใหม่ทั้งหมด
 - Support หน่วยงานสารสนเทศที่ให้การสนับสนุน ควรแนะนำผู้ใช้ระบบ ได้แก่ การให้คำแนะนำเรื่องเลือกตั้งระบบ ระบบที่จะใช้ การสำรองข้อมูล คำแนะนำหากเกิดข้อผิดพลาด การทำเอกสารต่างๆ
 - Evaluation การประเมินผลของระบบที่พัฒนาขึ้น และวิธีการจัดการต่างๆ ของ EUC เพื่อหาแนวทางในการป้องกันต่อไป

In-house System Development

การพัฒนาระบบสารสนเทศขึ้นใช้เองในหน่วยงาน

วงจรการพัฒนาระบบสารสนเทศ

System Development Life Cycle : SDLC

ขั้นการศึกษาและให้คำจำกัดความระบบ (System Definition)



ขั้นการวิเคราะห์ระบบ (System Analysis)



ขั้นการออกแบบและเขียนโปรแกรม
(System Design and Programming)

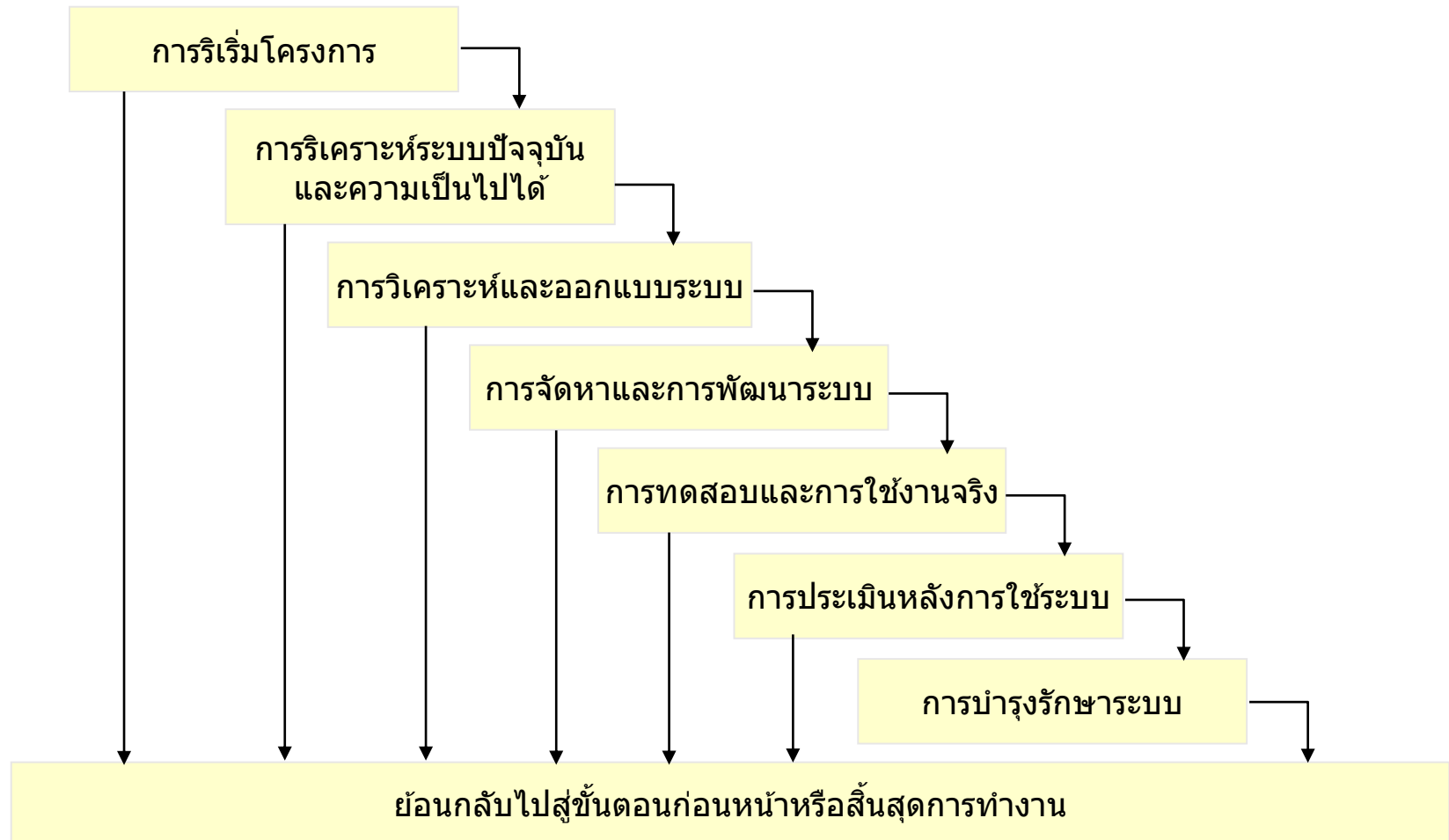


ขั้นการทดสอบระบบและการนำระบบไปใช้
(System Testing and Implementation)



ขั้นการบำรุงรักษาระบบ (System Maintenance)

System Development Life Cycle : SDLC : Waterfall Model



System Definition

- เป็นกระบวนการบ่งบอกว่าปัญหาที่แท้จริงคืออะไร
 - ระบบบัญชีผิดพลาดบ่อย ใช้บุคลากรมากเกินไป ทำอย่างไรจึงดี เร็ว ถูกกว่าเดิม
 - ปัญหาด้านบริหารจัดการระบบบัญชีภายใน (Accounting System)
 - ลูกค้าต้องการความมีสะดวกในการใช้บริการ มีระบบใดบ้างช่วยได้
 - ปัญหาด้านการบริการลูกค้า (Customer Service)
- มีจุดประสงค์หลัก คือ ต้องตอบคำถามได้สองข้อ
 - ทำไมเราจึงต้องการระบบใหม่
 - ระบบใหม่จะช่วยแก้ปัญหาได้อย่างไร

เช่น ปัญหาเกี่ยวกับการเช็คจำนวนสินค้าในคลังไม่ตรงกับจำนวนในบัญชีสต็อก

ทำไมจึงต้องการระบบใหม่ → ระบบเดิมตรวจเช็คสต็อกช้า สินค้าหาย ไม่ตรงบัญชีต้องทำการปรับปรุงยอดบัญชีบ่อยครั้ง เอาผิดผู้ใดไม่ได้ องค์กรเสียหาย

ระบบใหม่จะช่วยให้ → การตรวจเช็คเร็วขึ้น ตรวจสอบได้ตลอดเวลา และสร้างระบบการรับผิดชอบได้ชัดเจน ควบคุมสินค้าคงคลังได้ดีขึ้น

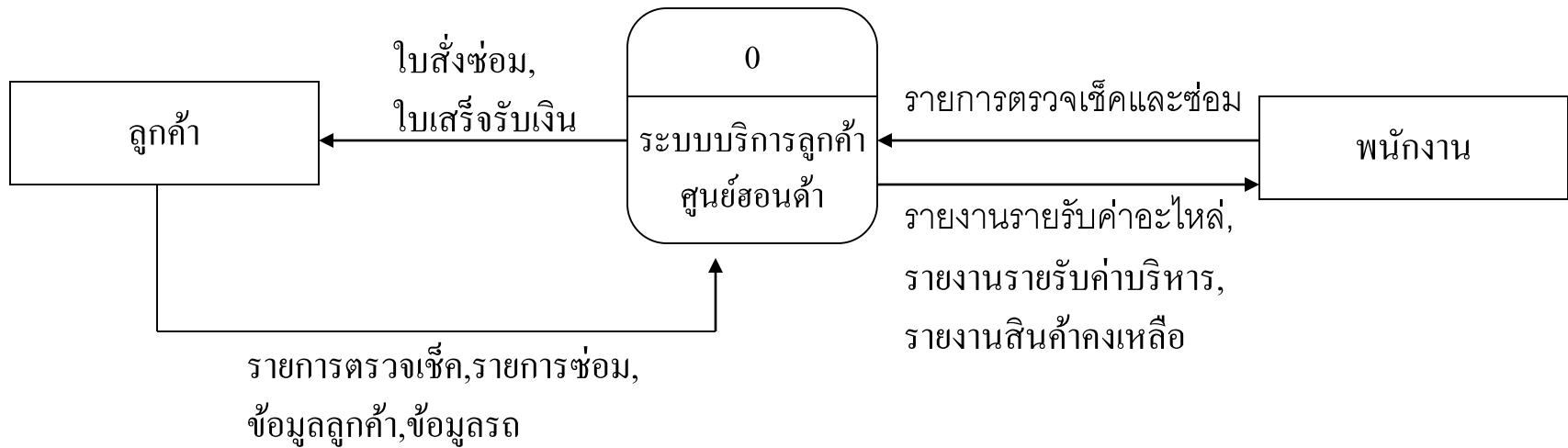
System Analysis

- กระบวนการวิเคราะห์ปัญหาอย่างละเอียด เพื่อให้ผู้พัฒนาระบบเข้าใจมากขึ้นในขอบเขต ความเป็นไปได้ และสิ่งที่ต้องการจากระบบใหม่นั้น
- ประกอบด้วยกิจกรรมหลัก 3 กิจกรรม คือ
 - การพยายามเข้าใจปัญหาอย่างแท้จริง ได้แก่ การเข้าใจถึงจุดแข็ง จุดอ่อนของปัญหาที่เกิดขึ้น จากการใช้ระบบเก่าและจากการนำระบบใหม่เข้ามา
 - การศึกษาความเป็นไปได้ของการนำระบบมาใช้ ในด้านต่างๆ ดังนี้
ด้านเทคนิค , ด้านเศรษฐศาสตร์ , ด้านการปฏิบัติการ ด้านตารางเวลา ด้านกฎหมาย และด้านกลยุทธ์ธุรกิจ
 - การกำหนดสิ่งที่ต้องการจากระบบใหม่
 - ใครคือคนที่ต้องการระบบ ต้องการไปเพื่ออะไร ใครคือผู้ได้รับผลของระบบ และต้องการระบบเมื่อไร ผลลัพธ์จะถูกส่งให้ผู้ใช้อย่างไร จะฝึกอบรมอย่างไรให้ใช้ได้เต็มที่ประสิทธิภาพ

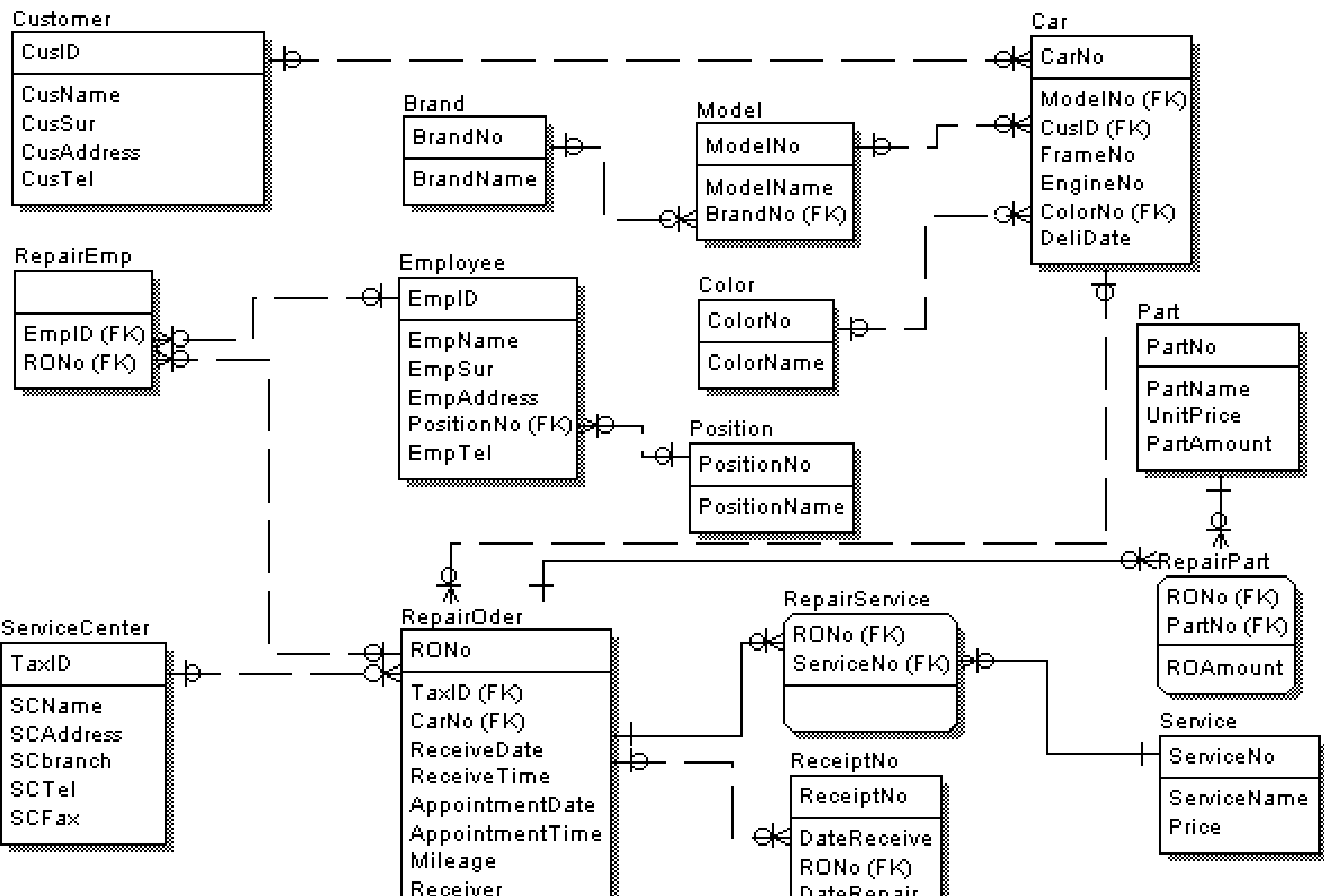
System Design & Programming

- **การออกแบบเชิงตรรกะ (Logical Design)**
 - คือการออกแบบความสัมพันธ์ของระบบต่างๆ ตลอดจนการออกแบบฐานข้อมูล
 - เครื่องมือที่นิยมใช้ได้แก่ Data Flow Diagram :DFD และ Entity Relationship Diagram : ER-Diagram และการสร้างตารางคำอธิบายข้อมูล (Data Dictionary)
- **การออกแบบเชิงกายภาพ (Physical Design)**
 - คือการออกแบบเรื่องอุปกรณ์ต่างๆ ที่ใช้ในระบบสารสนเทศ โครงสร้างเครือข่าย ระบบหน่วยความจำข้อมูลและระบบรักษาความปลอดภัย (Security System)
- **การเขียนโปรแกรม (Programming)**
 - การนำเอาผลลัพธ์จากการออกแบบระบบไปเขียนด้วยภาษาคอมพิวเตอร์ซึ่งใช้เวลานานมาก

Data Flow Diagram : DFD



Context Level Diagram ระบบบริการลูกค้าศูนย์ HONDA



Entity – Relationship Model ระบบบริการลูกค้าศูนย์ HONDA

Data Dictionary

ชื่อตาราง	ชื่อเขตข้อมูล	คำอธิบาย	ประเภท	ขนาด	PK	FK	Reference
Customer	CusID	รหัสลูกค้า	Text	4	/		
	CusName	ชื่อลูกค้า	Text	30			
	CusSur	นามสกุลลูกค้า	Text	30			
	CusAddress	ที่อยู่ลูกค้า	Text	100			
	CusTel	เบอร์โทรศัพท์ลูกค้า	Text	9			
Employee	EmplID	รหัสพนักงาน	Text	3	/		
	EmpName	ชื่อพนักงาน	Text	30			
	EmpSur	นามสกุลพนักงาน	Text	30			
	EmpAddress	ที่อยู่พนักงาน	Text	100			
	EmpTel	เบอร์โทรศัพท์พนักงาน	Text	9			
	PostionNo	รหัสตำแหน่ง	Text	2		/	Position
Position	PostionNo	รหัสตำแหน่ง	Text	2	/		
	PostionName	ชื่อตำแหน่ง	Text	50			
Color	ColorNo	รหัสสี	Text	2	/		
	ColorName	ชื่อสี	Text	20			
Brand	BrandNo	รหัสยี่ห้อ	Text	2	/		
	BrandName	ชื่อยี่ห้อ	Text	20			

Data Dictionary ระบบบริการลูกค้าศูนย์ HONDA

System Testing & Implementation

- การทดสอบระบบ (Testing)

- เพื่อให้ผู้ที่เกี่ยวข้องกับระบบมั่นใจว่าระบบที่พัฒนาขึ้นนี้จะสามารถทำงานได้ตามคาดหวัง และเป็นการทดสอบด้วยการพยายามทำให้ระบบนั้นไม่สามารถทำงานได้ (System Hardening) เพื่อหาทางแก้ไขไม่ให้เกิดข้อผิดพลาด

- การทดสอบมี 3 รูปแบบ คือ

- **Unit Testing** : การทดสอบระบบย่อยๆ แต่ละระบบ ไม่พร้อมกัน
- **System Testing** : การทดสอบระบบทั้งระบบ เพื่อตรวจสอบการทำงานร่วมกันของฮาร์ดแวร์และซอฟต์แวร์
- **Acceptance Testing** : ทดสอบการยอมรับของผู้ใช้

System Testing & Implementation

- **การนำระบบไปใช้ (Implementing)**
 - **Parallel Conversion** การนำระบบใหม่เข้าไปใช้คู่ขนานกับระบบเก่า จนกว่าระบบใหม่จะทำงานโดยไร้ข้อผิดพลาด
 - **Direct Cut-over** การนำระบบใหม่เข้าแทนระบบเดิม โดยตัดระบบเก่าทิ้งอย่างสิ้นเชิง
 - **Pilot Study** การนำระบบใหม่เข้าทดลองใช้เพียงบางหน่วยงาน จนกว่าจะเห็นว่าทำงานได้ดี จึงจะนำไปใช้ทั้งองค์กร
 - **Phased Conversion** การนำระบบใหม่ไปใช้แทนที่ระบบเก่าเพียงบางส่วน (**Phased**) เช่น การพัฒนาระบบการจัดการสินค้าคงคลังแผนกเดียว

System Maintenance and Limit of SDLC

- **การบำรุงรักษาระบบ** ได้แก่การสร้างความมั่นใจว่าระบบตรงกับความต้องการผู้ใช้ หากมีความต้องการปรับปรุง เปลี่ยนแปลง ผู้พัฒนาระบบต้องสามารถแก้ไขได้ จึงต้องมีระบบการบำรุงรักษา หากการใช้งานจริงเกิดการขัดข้องทางเทคนิคต่างๆ
- **ข้อจำกัดของ SDLC**
 - เหมาะกับการพัฒนาระบบที่มีรูปแบบชัดเจน หรืองานประจำ
 - ใช้เวลานานในการพัฒนาและไม่ค่อยยืดหยุ่น
 - ใช้งบประมาณค่อนข้างสูงในการพัฒนา
 - เหมาะกับการพัฒนาระบบสารสนเทศสำหรับองค์กรขนาดใหญ่

การวัดความสำเร็จของระบบสารสนเทศ

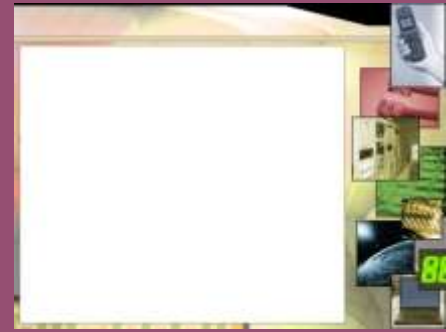
- High Level of System Use
 - มีผู้ใช้ระบบเป็นจำนวนมาก
- User Satisfaction with the System
 - ผู้ใช้มีความพึงพอใจในระบบมาก
- Favorable Attitudes
 - มีความชื่นชอบ
- Achieved Objectives
 - สามารถบรรลุเป้าหมายที่กำหนดไว้
- Financial Payoff
 - ลดค่าใช้จ่ายองค์กรโดยรวม

ปัจจัยที่ทำให้ระบบสารสนเทศสำเร็จ

- ผู้ใช้ต้องให้ความร่วมมือในการพัฒนาระบบ
- ผู้บริหารต้องเห็นความสำคัญของโครงการ สนับสนุนและให้ความช่วยเหลือ
- ทีมงานผู้พัฒนาระบบจะต้องมีความรู้ ความสามารถและประสบการณ์ในการพัฒนาระบบเป็นอย่างดี
- ผู้ใช้ต้องมีการกำหนดให้มีผู้ประสานงานในการพัฒนาระบบ
- อุปกรณ์ Hardware , Software ที่มีสมรรถนะที่ดี
- การพัฒนาระบบต้องมีการวางแผนโครงการที่ดีและเป็นไปตามแผน
- จะต้องมีการทดสอบระบบอย่างรอบคอบ
- จะต้องมีการอบรมการใช้ในองค์กรอย่างทั่วถึง

ปัจจัยที่ทำให้ระบบสารสนเทศล้มเหลว

- **User Involvement and Influence**
(การมีส่วนร่วมและอิทธิพลของผู้ใช้มีน้อย)
- **Management Support and Commitment**
(ผู้บริหารไม่สนับสนุนและไม่ทำตามคำมั่นสัญญา)
- **Level of Complexity and Risk**
(มีระดับความซับซ้อนและความเสี่ยงจากการใช้งานระบบสารสนเทศสูง)
- **Management of the Implementation Process**
(ขาดการบริหารกระบวนการในการนำระบบไปใช้งาน)



End of Chapter