

บทที่ 8

การออกแบบซอฟต์แวร์: การออกแบบส่วนต่อประสานผู้ใช้ (Software Design: User Interface Design)

วิชา วิศวกรรมซอฟต์แวร์ (04-06-306)



วัตถุประสงค์การเรียนรู้

- เพื่อให้ผู้เรียนมีความความเข้าใจเกี่ยวกับกระบวนการออกแบบส่วนต่อประสานผู้ใช้งาน รวมทั้งการประเมินคุณภาพของการออกแบบส่วนต่อประสานในด้านต่างๆ

หัวข้อ

- บทนำ (Overview)
- การออกแบบส่วนประสาน (Interface Design)
- กระบวนการในการพัฒนาส่วนประสาน
- รูปแบบของส่วนประสานกับผู้ใช้ (User Interface)
- การประเมินส่วนประสาน และเกณฑ์คุณภาพ
- การวิเคราะห์และประเมินคุณภาพงานออกแบบซอฟต์แวร์
- การวัดการออกแบบซอฟต์แวร์
- สรุป (Summary)

บทนำ (Overview)

การออกแบบส่วนต่อประสาน ระหว่าง
องค์ประกอบย่อยภายในซอฟต์แวร์

การออกแบบส่วนต่อประสานระหว่าง
ซอฟต์แวร์และองค์ประกอบอื่นๆ ที่ไม่ใช่
มนุษย์ที่เป็นส่วนผลิต และใช้ข้อมูล

การออกแบบส่วนต่อประสานระหว่างมนุษย์
กับคอมพิวเตอร์

การออกแบบส่วนประสาน (Interface Design)

Easy to learn?

Easy to use?

Easy to understand?



การออกแบบส่วนประสาน (ต่อ)

- ส่วนประสานกับผู้ใช้ (User Interface)

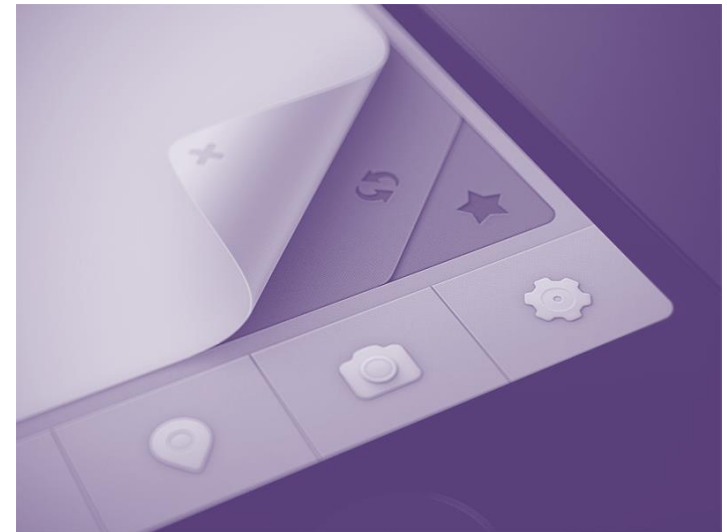
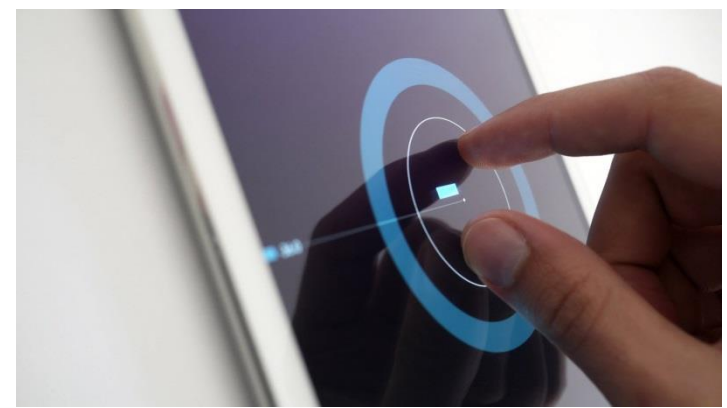
ส่วนติดต่อระหว่างผู้ใช้งานระบบ เพื่อเตรียมสารสนเทศการทำงานและนำเสนอสารสนเทศ เรียกว่า **“การออกแบบจอภาพ (Screen Design)”**

เกิดการทำงานตามวัตถุประสงค์ สามารถเป็น**เครื่องบ่งชี้การใช้งานซอฟต์แวร์**
เน้นส่วนประสานแบบกราฟฟิก (Graphic User Interface: GUI)

การใช้งานต้อง**สอดคล้องกัน** **ความแตกต่างกันของผู้ใช้งาน**ทั้งพฤติกรรมและ
ประสบการณ์ทำงาน

การออกแบบส่วนประสาน (ต่อ)

- Theo Mandel ได้บัญญัติกฎ 3 ข้อในการออกแบบส่วนต่อประสาน ดังนี้
 - ให้ผู้ใช้เป็นผู้ควบคุมการทำงาน - Place the user in control
 - ลดภาระการต้องจดจำของผู้ใช้ - Reduce the user's memory load
 - สร้างส่วนต่อประสานอย่างคงเส้นคงวา (สอดคล้องกัน) - Make the interface consistent



ให้ผู้ใช้เป็นผู้ควบคุมการทำงาน

- หลักเกณฑ์ในการออกแบบที่ใช้ผู้ใช้ควบคุม ประกอบด้วยดังนี้
 1. กำหนดโหมดการโต้ตอบในลักษณะที่ไม่บังคับผู้ใช้โดยไม่จำเป็น หรือในทางที่ผู้ใช้ไม่ต้องการที่จะทำ
 2. จัดให้มีการโต้ตอบที่ยืดหยุ่น สามารถโต้ตอบกับระบบได้มากกว่า 1 ช่องทาง
 3. อนุญาตให้ผู้ใช้ทำการหยุดหรือสามารถยกเลิกได้
 4. ออกแบบให้การโต้ตอบเป็นไปตามระดับความชำนาญในการใช้งาน เตรียมเครื่องมือสร้างการทำงานแบบอัตโนมัติให้กับผู้ใช้

ให้ผู้ใช้เป็นผู้ควบคุมการทำงาน (ต่อ)

5. ซ่อนรายละเอียดด้านเทคนิค จาก
ผู้ใช้ทั่วไป ไม่ควรให้ผู้ใช้ติดต่อกับ
ระบบปฏิบัติการด้วยการพิมพ์
คำสั่งโดยตรง
6. การออกแบบวัตถุที่วางไว้บนจอ
ให้เข้าถึงโดยตรง



ลดภาระการต้องจดจำของผู้ใช้

- ซอฟต์แวร์ที่ให้ผู้ใช้งานจดจำรายละเอียดการทำงานมากเกินไป
 - เสี่ยงต่อการเกิดความผิดพลาด
 - ช่วยเตือนความจำให้ผู้ใช้งาน
- Mandel ออกแบบหลักการที่ช่วยลดภาระการจดจำของผู้ใช้ ดังนี้
 - ลดความต้องการใช้งาน หน่วยความจำระยะสั้น ของผู้ใช้ ขณะที่ใช้โปรแกรมอยู่
 - การกำหนด ค่าโดยปริยาย ที่มีความหมาย
 - นิยาม ปุ่มลัด (Shortcut) ที่เข้าใจง่าย
 - การจัดภาพของส่วนต่อประสานควรเป็นไปตาม อุปลักษณะของโลกจริง
 - เปิดเผยข่าวสารในลักษณะค่อยๆ เพิ่มพูน

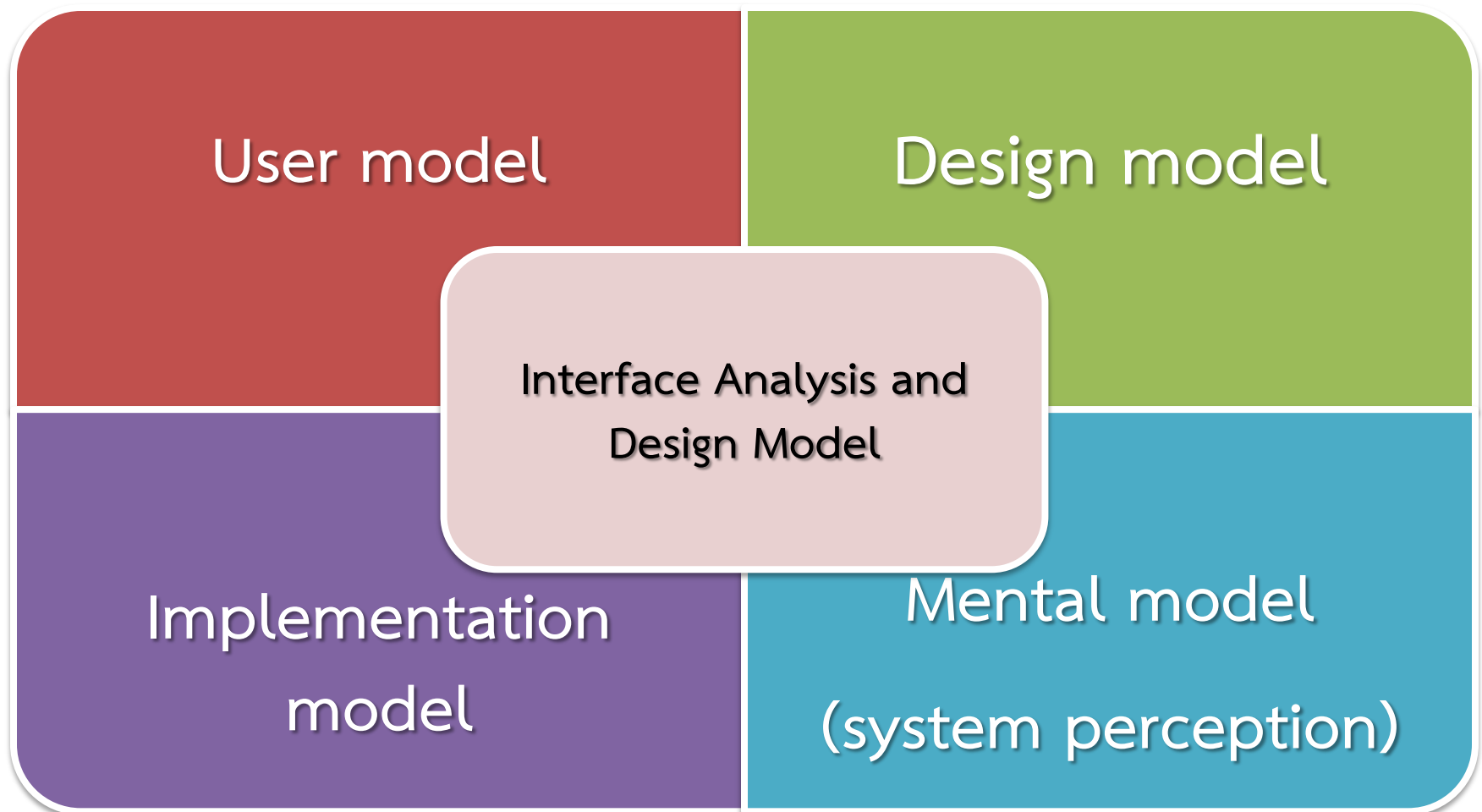
ส่วนประสานต้องสอดคล้องกัน

- ส่วนประสานควรรับและแสดงผลในลักษณะคงเส้นคงวา
 - ข่าวสารทางภาพจัดระเบียบตามมาตรฐานการออกแบบเดียวกันตลอดทุกหน้าจอของระบบ
 - กลไกการท่องระบบจากงานหนึ่งสู่งานหนึ่งเป็นไปอย่างคงเส้นคงวาสอดคล้องกัน เชื่อมโยงกันเป็นลำดับขั้นตอน

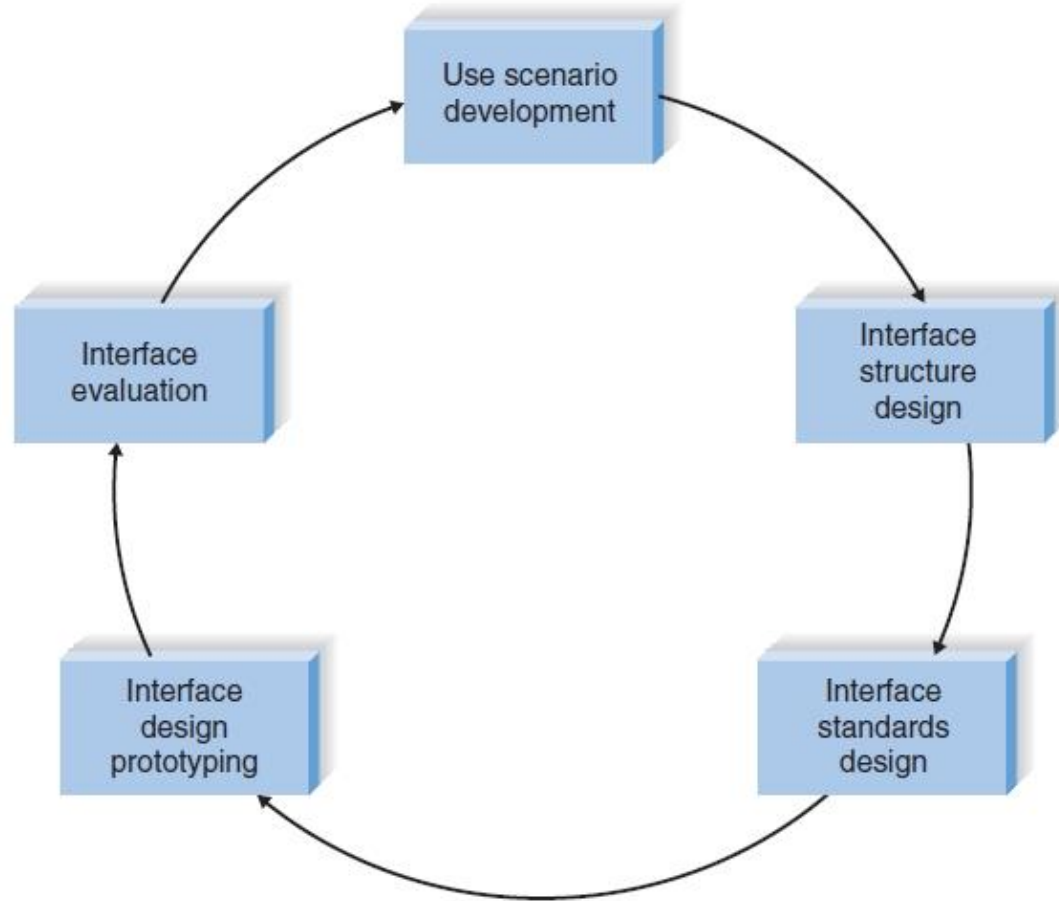
ส่วนประสานต้องสอดคล้องกัน (ต่อ)

- หลักการออกแบบที่ช่วยให้ส่วนต่อประสานคงเส้นคงวา มีดังนี้
 - ช่วยให้ผู้ใช้ทราบว่าการปัจจุบันอยู่ภายใต้บริบทใด
 - คงเส้นคงวาตลอดทั้งตระกูลของแอปพลิเคชัน
 - ถ้ารูปแบบการโต้ตอบที่ผ่านมาทำให้ผู้ใช้เกิดความสับสน อย่าเปลี่ยนกฎนั้น ยกเว้นมีเหตุผลสมควร

การวิเคราะห์และออกแบบส่วนประสานกับผู้ใช้



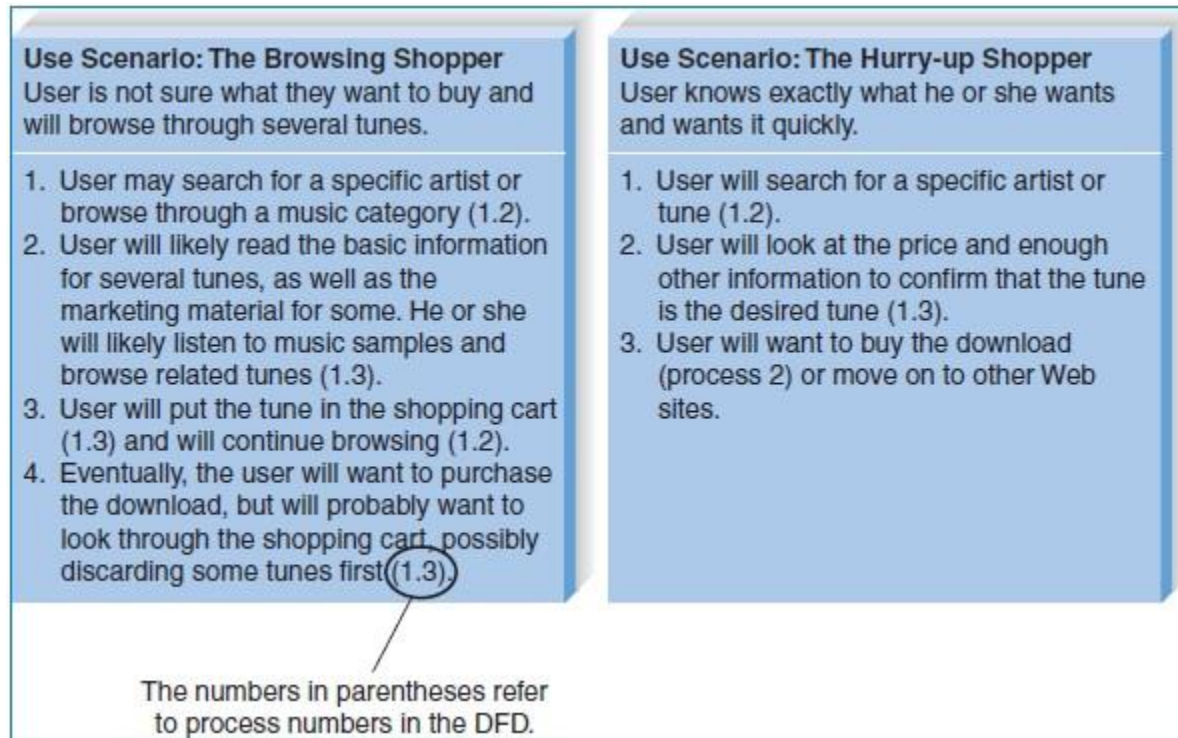
กระบวนการในการพัฒนาส่วนประสาน



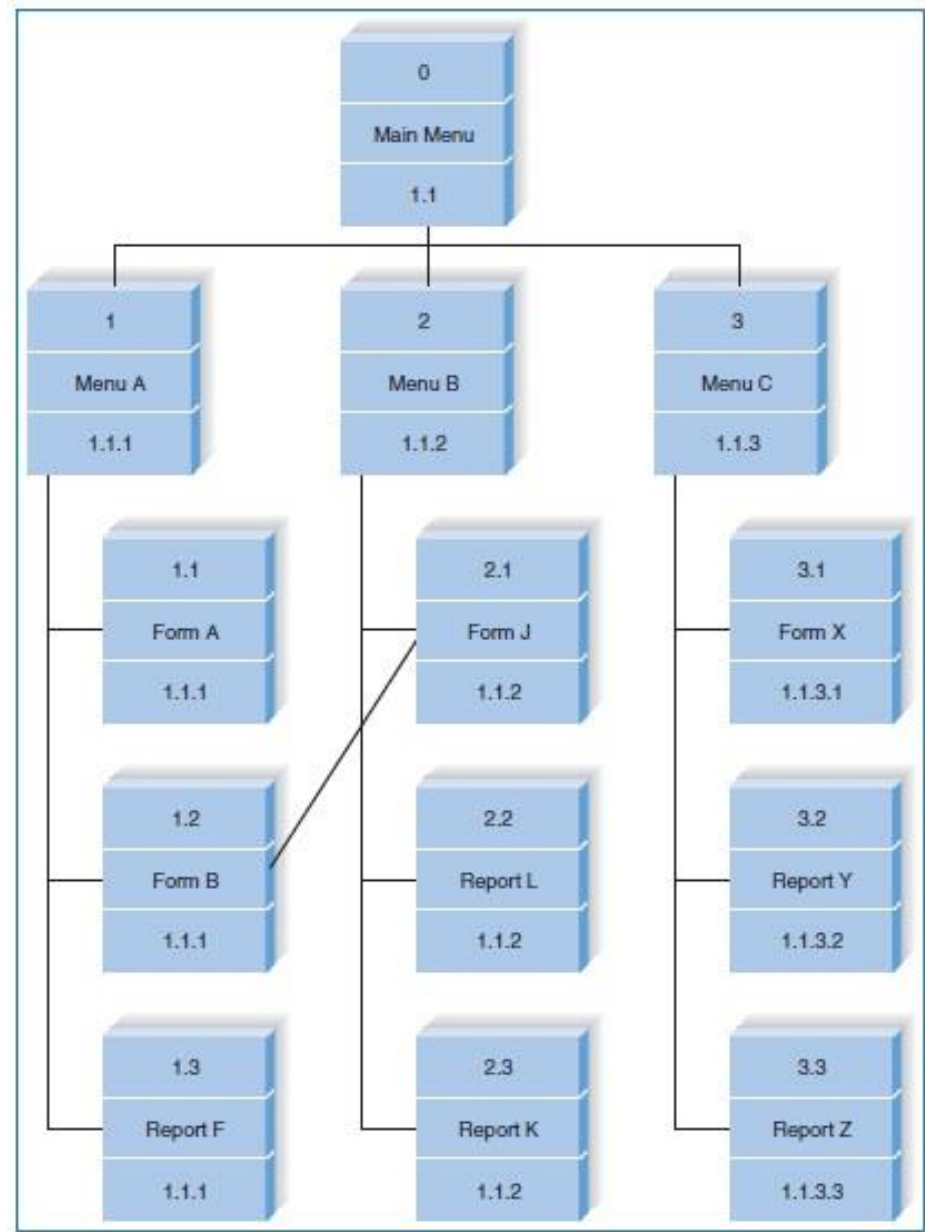
ที่มา: Alan Dennis, Barbara Haley Wixom and Roberta M, *Systems Analysis and Design Fifth Edition*, 2012.

Example:

Two Use Scenarios for the Search and Browse Tunes Use Case

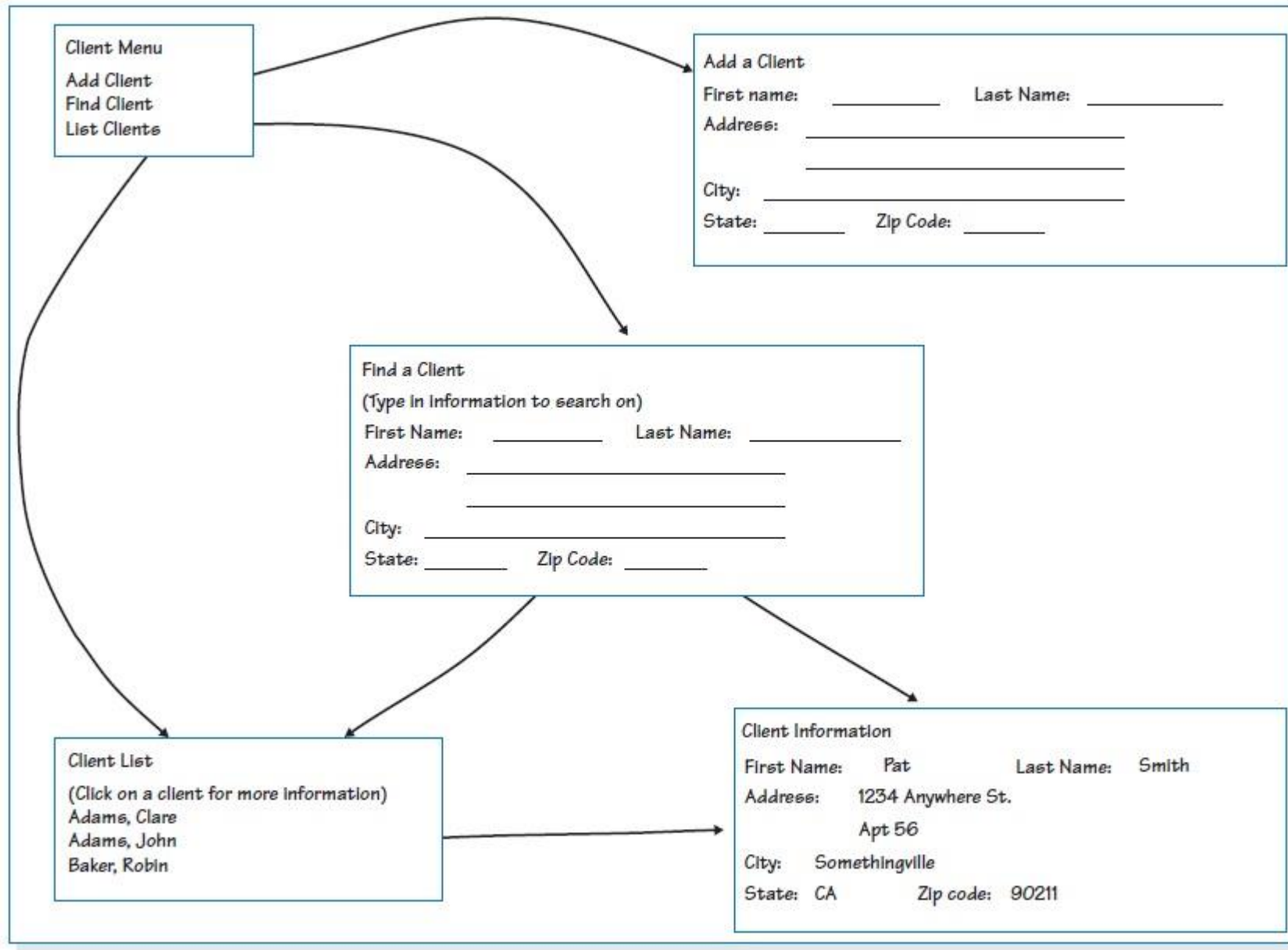


Example: Interface Structure Diagram (ISD)



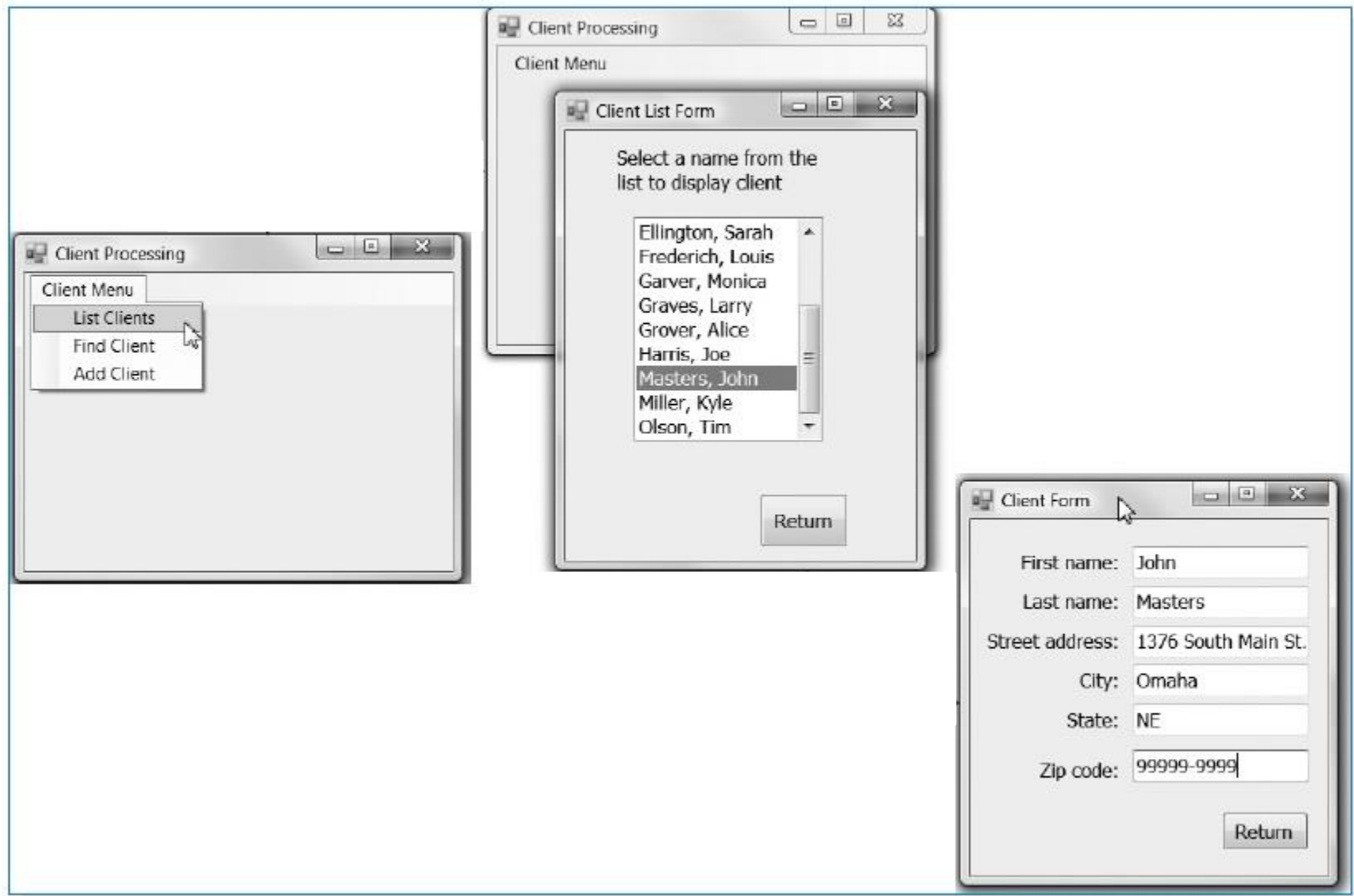
ที่มา: Alan Dennis, Barbara Haley Wixom and Roberta M, *Systems Analysis and Design Fifth Edition*, 2012.

Example: Storyboard



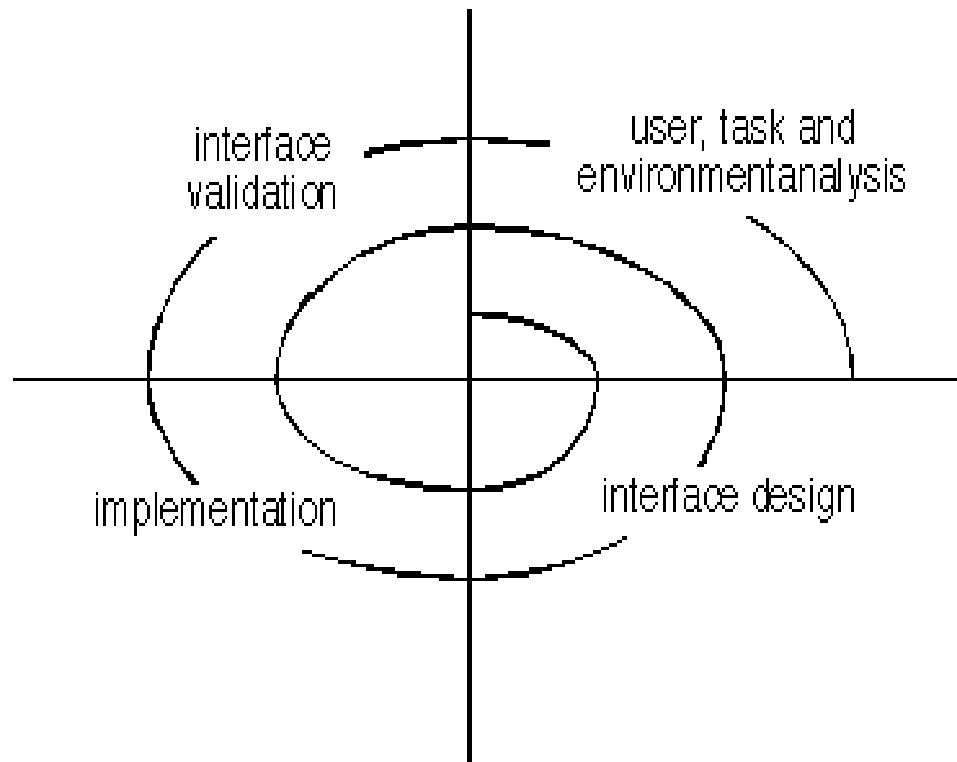
ที่มา: Alan Dennis, Barbara Haley Wixom and Roberta M, Systems Analysis and Design Fifth Edition, 2012.

Example: Sample Language Prototype



ที่มา: Alan Dennis, Barbara Haley Wixom and Roberta M, *Systems Analysis and Design Fifth Edition*, 2012.

กระบวนการในการพัฒนาส่วนประสาน (ต่อ)



การวิเคราะห์ส่วนต่อประสาน (Interface Analysis)

Interface analysis means understanding

- ในการวิเคราะห์ส่วนต่อประสานต้องเข้าใจปัญหา ดังนี้
 - **the people** - เข้าใจคน บุคลากรที่ใช้งานระบบ
 - **the tasks** - เข้าใจงานที่ผู้ใช้ต้องการใช้เพื่อให้งานให้สำเร็จ
 - **the content** - เข้าใจเนื้อหาที่จะต้องนำเสนอในส่วนต่อประสาน
 - **the environment** - เข้าใจสิ่งแวดล้อมที่งานเหล่านั้นทำงานอยู่

การวิเคราะห์ผู้ใช้งาน (User Analysis)

- ควรเข้าใจผู้ใช้งานว่าจะใช้ระบบอย่างไร ต้องการส่วนต่อประสานลักษณะใด
- การสัมภาษณ์ ซึ่งจะช่วยให้นักออกแบบเข้าใจว่า
 - ใครคือผู้ใช้งาน
 - แบ่งกลุ่มผู้ใช้อย่างไร
 - ผู้ใช้แต่ละกลุ่มมีทักษะและประสบการณ์ในระดับใด
 - แบบจำลองสภาพจิตใจของผู้ใช้ที่มีต่อระบบเป็นอย่างไร
 - ส่วนต่อประสานจะตอบสนองความต้องการของผู้ใช้ได้อย่างไร

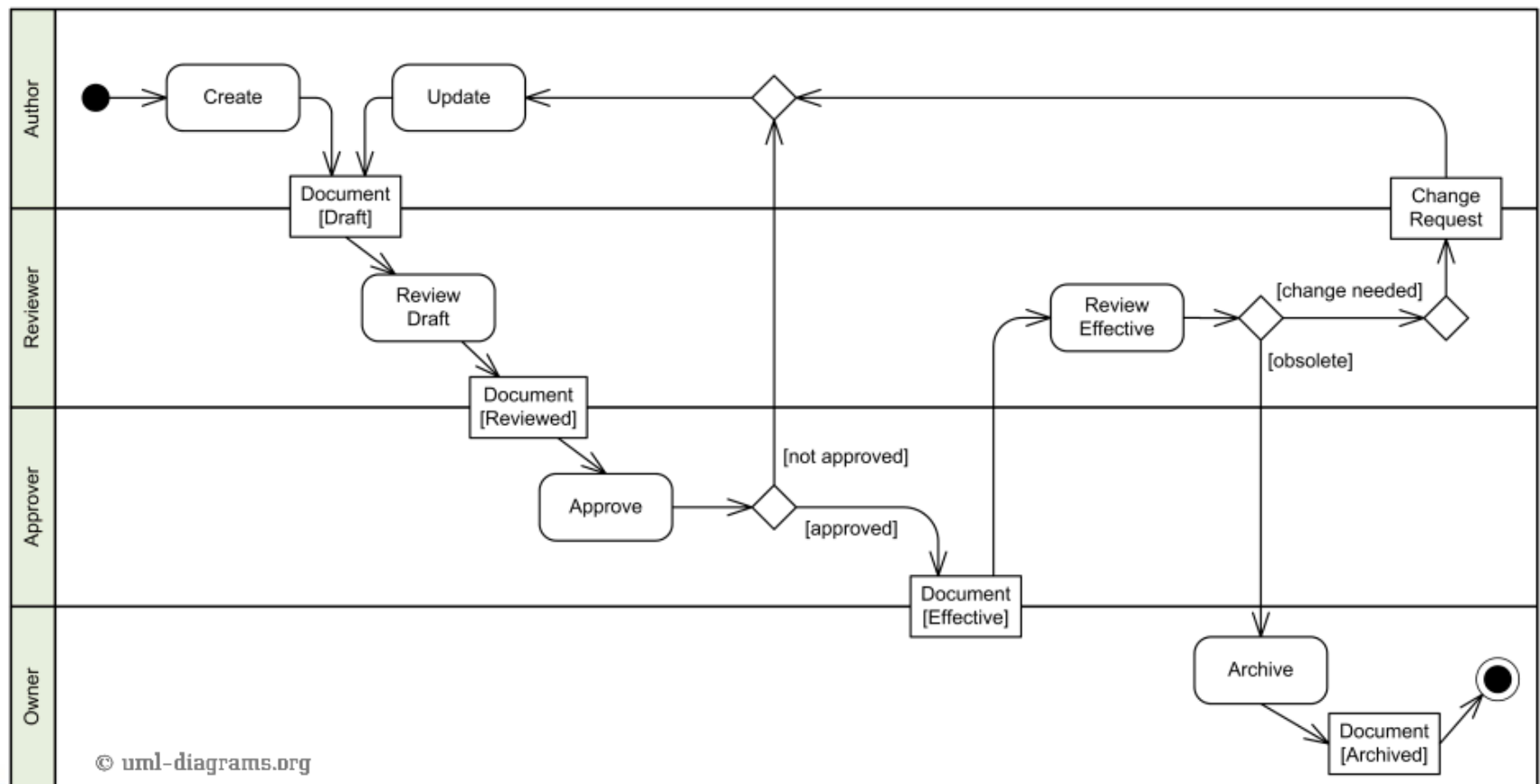
การจำลองและวิเคราะห์งานย่อย

- Answers the following questions ...
 - What work will the user perform in specific circumstances? (ลักษณะของการปฏิบัติงาน)
 - What tasks and subtasks will be performed as the user does **the work**? (งานหลักและงานย่อยที่ต้องปฏิบัติ)
 - What specific problem domain objects will the user manipulate as work is performed? (ปัญหาหลักที่ต้องใช้งานเพื่อทำงาน ระบบงานและส่วนที่เกี่ยวข้อง)
 - What is the sequence of work tasks—the workflow? (ลำดับของงานที่ทำ)
 - What is the hierarchy of tasks? (ลำดับชั้นของงานย่อย)

การจำลองและวิเคราะห์งานย่อย (ต่อ)

- เทคนิคในการวิเคราะห์ออกแบบส่วนต่อประสาน
 - Use-cases define basic interaction
 - Task elaboration refines interactive tasks
 - Object elaboration identifies interface objects (classes)
 - Workflow analysis defines how a work process is completed when several people (and roles) are involved

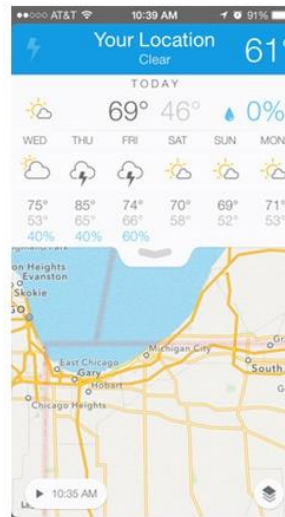
- แผนภาพกิจกรรม กระบวนการทำงานบริหารจัดการเอกสาร (Document Management Process)
- มีผู้เกี่ยวข้อง 4 ส่วน ดังนี้ (a) _____, (b) _____, (c) _____ และ (d) _____
- นั้กออกแบบส่วนต่อประสานควรคำนึงถึงลักษณะของผู้ใช้ และกิจกรรมงาน



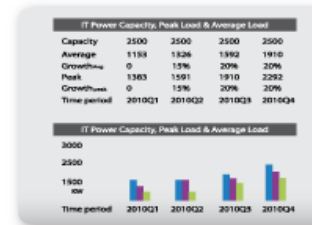
การวิเคราะห์การนำเสนอเนื้อหา

- **เนื้อหาที่นำเสนอ:** รายงานที่เป็นตัวอักษร รูปภาพ หรือข้อมูลเฉพาะ เช่น เสียง หรือภาพเคลื่อนไหว เป็นต้น
 - ถูกสร้างโดยส่วนประกอบของระบบที่ไม่เกี่ยวข้องกับส่วนต่อประสาน
 - ดึงมาจากข้อมูลที่เก็บในฐานข้อมูล
 - ส่งมาจากระบบภายนอก
- **การวิเคราะห์การนำเสนอเนื้อหา** ทำให้ทราบถึงเอกสารที่ต้องการและการแสดงผลที่ต้องการ รูปแบบและความสวยงามของเนื้อหาจะถูกพิจารณา

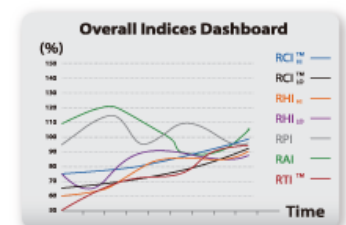
การนำเสนอเนื้อหา



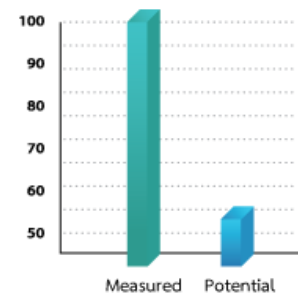
Power Analysis Report



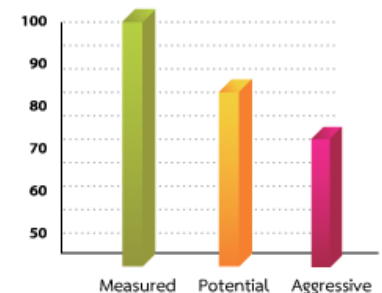
Real-time and Continuous RCI and RTI Data



Fan Energy Saving %



Chiller Energy Saving %



การวิเคราะห์สิ่งแวดล้อมการทำงาน

- นักออกแบบควรคำนึงถึงสภาพแวดล้อมที่ใช้งานของระบบ ข้อจำกัดทางกายภาพที่อาจเป็นอุปสรรคในการใช้งาน
 - โรงงานเสียงดัง การใช้ลำโพงอาจไม่เหมาะสม
 - การใช้เมาส์ คีย์บอร์ดในพื้นที่คับแคบ
- วัฒนธรรมในการทำงาน เช่น
 - ข้อมูลต้องได้รับการรับรองจากหลายฝ่ายก่อนบันทึกหรือไม่
 - ผู้ใช้งานจะได้รับความช่วยเหลือจากระบบอย่างไร
- นักออกแบบต้องตอบคำถามเหล่านี้ก่อนการออกแบบเสร็จสิ้น และควรเพิ่มส่วนต่อประสานที่จะอำนวยความสะดวกด้วย

ขั้นตอนการออกแบบส่วนต่อประสาน

- Define interface objects and actions (operations)
 - นิยามวัตถุและตัวดำเนินการ โดยใช้ข้อมูลจากการวิเคราะห์
- Define events (user actions)
 - กำหนดเหตุการณ์ที่เป็นการกระทำของผู้ใช้
- Depict each interface state
 - แสดงด้วยรูปถึงสถานะของส่วนต่อประสานที่ผู้ใช้จะได้สัมผัส
- Indicate how the user interprets the state of the system
 - อธิบายให้ทราบความหมายของข้อมูลที่แสดง ระบุว่าผู้ใช้จะเข้าใจสถานะของระบบอย่างไร

ข้อควรคำนึงในการออกแบบ

- **Response time:** เวลาในการตอบสนองของระบบ
- **Help facilities:** การช่วยเหลือแก่ผู้ใช้งาน
- **Error handling:** การจัดการความผิดพลาด
- **Menu and command labeling:** การกำหนดชื่อคำสั่งและเมนู
- **Application accessibility:** การเข้าถึงระบบงาน
- **Internationalization:** ความเป็นสากล



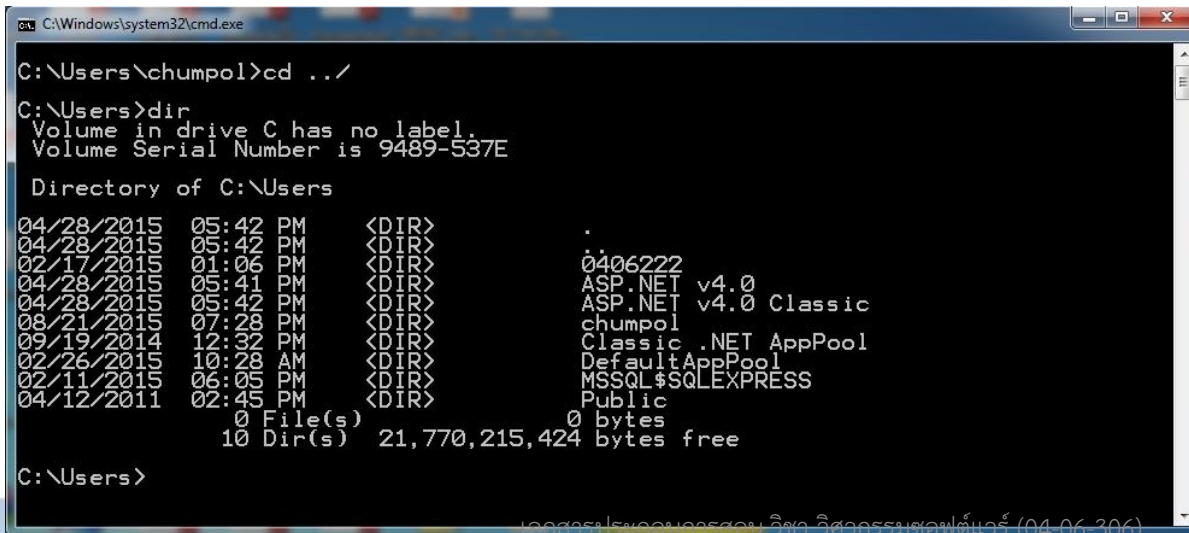
รูปแบบของ User Interfaces

- ผู้ใช้งานสามารถโต้ตอบกับระบบอย่างมีประสิทธิภาพ **นิยมใช้แบบกราฟิก (Graphic User Interface: GUI)** ซึ่งมีรูปแบบดังนี้
 - การโต้ตอบด้วยคำสั่ง (Command Language Interaction)
 - การโต้ตอบด้วยเมนูคำสั่ง (Menu Interaction)
 - การโต้ตอบด้วยแบบฟอร์ม (Form Interaction)
 - การโต้ตอบด้วยการทำงานเชิงวัตถุ (Object-Based Interaction)
 - การโต้ตอบด้วยภาษามนุษย์ (Natural Language Interaction)

การโต้ตอบด้วยคำสั่ง

(Command Language Interaction)

- เป็นการโต้ตอบกับระบบโดยที่ผู้ใช้จะต้องพิมพ์คำสั่งลงในช่องป้อนคำสั่ง เพื่อกระตุ้นให้เกิดการทำงานในระบบ
- ผู้ใช้จะต้องจำคำสั่ง ไวยากรณ์ และกฎเกณฑ์ต่างๆ
 - ผู้ใช้ที่ชำนาญการใช้ระบบปฏิบัติการ DOS
- ลดความนิยมในปัจจุบัน



```
C:\Windows\system32\cmd.exe
C:\Users\chumpol>cd ../
C:\Users>dir
Volume in drive C has no label.
Volume Serial Number is 9489-537E

Directory of C:\Users

04/28/2015  05:42 PM    <DIR>          .
04/28/2015  05:42 PM    <DIR>          ..
02/17/2015  01:06 PM    <DIR>          0406222
04/28/2015  05:41 PM    <DIR>          ASP.NET v4.0
04/28/2015  05:42 PM    <DIR>          ASP.NET v4.0 Classic
08/21/2015  07:28 PM    <DIR>          chumpol
09/19/2014  12:32 PM    <DIR>          Classic .NET AppPool
02/26/2015  10:28 AM    <DIR>          DefaultAppPool
02/11/2015  06:05 PM    <DIR>          MSSQL$SQLEXPRESS
04/12/2011  02:45 PM    <DIR>          Public
               0 File(s)              0 bytes
            10 Dir(s)  21,770,215,424 bytes free

C:\Users>
```

คำสั่ง

\$ cd ../

\$ dir

การโต้ตอบด้วยเมนูคำสั่ง (Menu Interaction)

- เป็นการโต้ตอบกับระบบด้วยการแสดง**เมนูคำสั่ง** โดยผู้ใช้ไม่จำเป็นต้องป้อนคำสั่งเอง
- รูปแบบเมนูมีดังนี้ คือ
 - Pull-down Menu
 - Pop-up Menu

การโต้ตอบด้วยเมนูคำสั่ง (ต่อ)

The screenshot displays the Microsoft SQL Server Management Studio interface. The 'View' menu is open, showing various toolbars and panes. A red arrow points from the 'View' menu to the 'Object Explorer' pane, labeled 'Pull-down Menu'. The 'Results' pane shows a table with 5 rows of data. A context menu is open over the table, and a red arrow points from the 'Copy' option in the context menu to the 'Copy' option in the 'Edit' menu, labeled 'Pop-up Menu'.

Pull-down Menu

Pop-up Menu

CustomerID	Name	Email	CountryCode	Budget	Used
1	C001	Win Weerachai	win.weerachai@thaicreate.com	TH	1000000 600000
2	C002	John		2000000	800000
3	C003	James		3000000	600000
4	C004	Chalee		4000000	100000
5	C005	John		5000000	100000

Query executed successfully. 203.158.128.55 (11.0 RTM) | sa (54) | master | 00:00:00 | 5 rows

หลักเกณฑ์ในการออกแบบเมนูคำสั่ง

- ควรเลือกใช้คำสั่งที่สื่อความหมายได้ชัดเจน
- ควรมีการใช้ตัวอักษรพิมพ์ใหญ่หรือตัวอักษรพิมพ์เล็กตามความเหมาะสม
- ควรมีการจัดกลุ่มคำสั่งที่มีความเกี่ยวข้องกันไว้ในกลุ่มเดียวกัน
- ไม่ควรมีจำนวนเมนูคำสั่งมากเกินไป
- ไม่ควรมีเมนูย่อยสำหรับเมนูคำสั่งที่มีการทำงานย่อยภายในมากเกินไป
- เมื่อมีการเลือกเมนูคำสั่ง ควรออกแบบให้มีแถบสีปรากฏที่เมนูคำสั่งที่ถูกเลือก

การโต้ตอบด้วยแบบฟอร์ม (Form Interaction)

- เป็นการโต้ตอบที่ผู้ใช้ระบบจะต้องป้อนข้อมูลลงในช่องว่างที่อยู่ในแบบฟอร์มที่แสดงหน้าจอคอมพิวเตอร์
- คล้ายการกรอกแบบฟอร์มลงในกระดาษ
- ชื่อของช่องป้อนข้อมูลต้องสื่อความหมาย
- แบ่งส่วนของข้อมูลบนฟอร์มให้เหมาะสม
- ควรแสดงข้อมูลเริ่มต้นให้กับช่องป้อนข้อมูลที่ต้องใช้ข้อมูลนั้นบ่อยครั้ง
- ช่องป้อนข้อมูลไม่ควรยาวมากจนเกินไป

การโต้ตอบด้วยแบบฟอร์ม (ต่อ)

Input Form

Name

Email

Subject

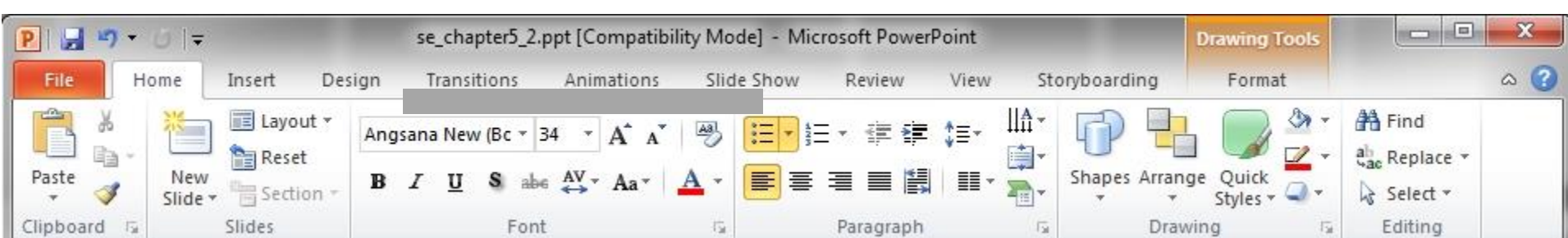
☒ Milk
☐ Sugar
☐ Lemon (Disabled)

☒ Male
☐ Female
☐ Don't know (Disabled)

Send >

การโต้ตอบเชิงวัตถุ (Object-Based Interaction)

- เป็นการโต้ตอบกับระบบที่ใช้สัญลักษณ์
- สัญลักษณ์เป็นตัวแทนคำสั่งที่ใช้ในการปฏิบัติงาน
- สัญลักษณ์รูปภาพแทนคำสั่งการทำงานเรียกว่า **ไอคอน (Icon)**
- ประหยัดพื้นที่บนหน้าจอ



การโต้ตอบด้วยภาษามนุษย์ (Natural Language Interaction)

- เป็นการโต้ตอบกับระบบด้วยการใช้เสียงพูดของผู้ใช้ระบบ
- ใช้เสียงพูดทั้งการนำข้อมูลเข้าและออกจากระบบ

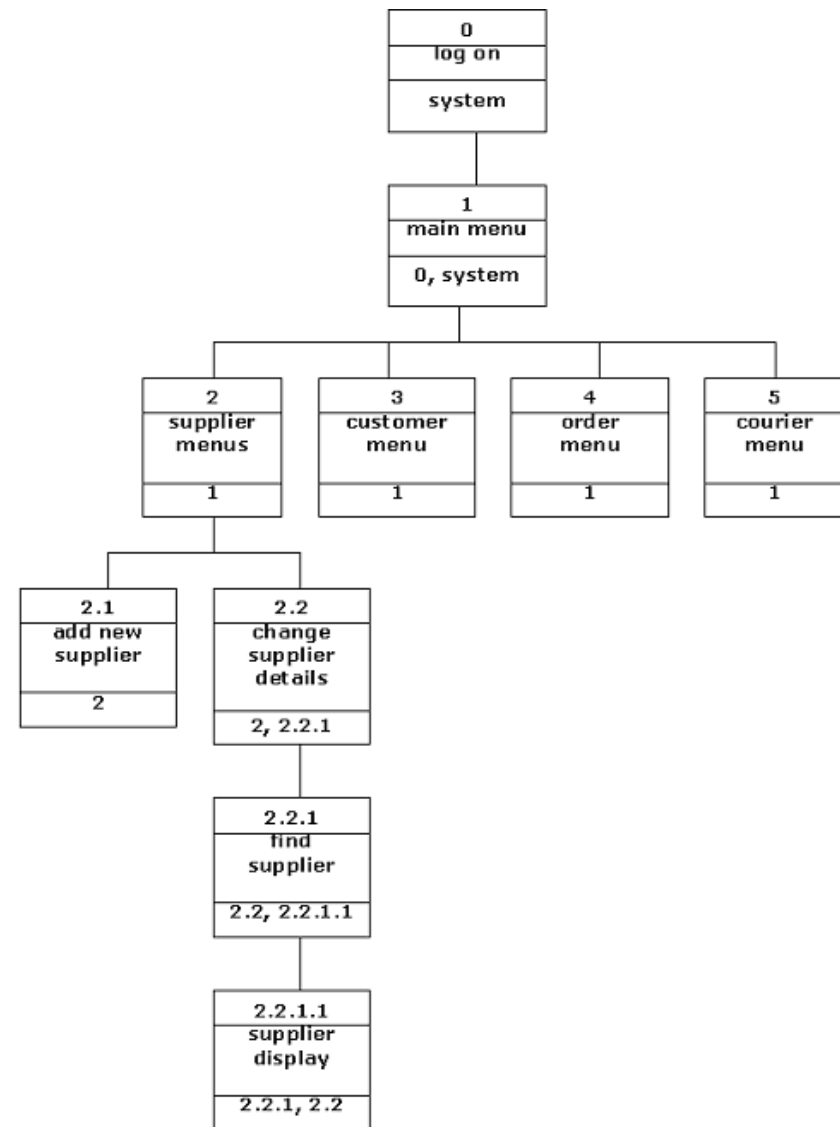
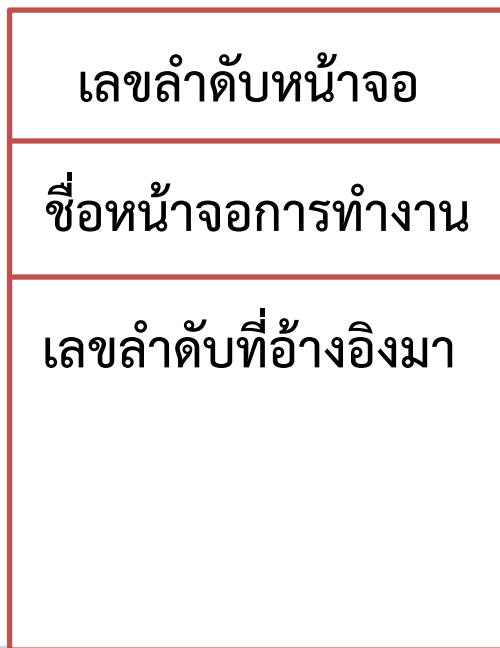


การออกแบบลำดับการเชื่อมโยงจอภาพ

- เป็นการออกแบบลำดับของการแสดงส่วนติดต่อกับผู้ใช้ของโปรแกรม หรือลำดับของการแสดงส่วน User Interface ทางหน้าจอคอมพิวเตอร์
- แผนภาพแสดงลำดับการเชื่อมโยงจอภาพ (Dialogues Diagram) ประกอบไปด้วย 3 ส่วน คือ
 - ส่วนบน: เลขลำดับหน้าจอ
 - ส่วนกลาง: ชื่อหน้าจอการทำงาน
 - ส่วนล่าง: เลขลำดับหน้าจอที่อ้างอิงมา ต่อไป หรือ ย้อนกลับ

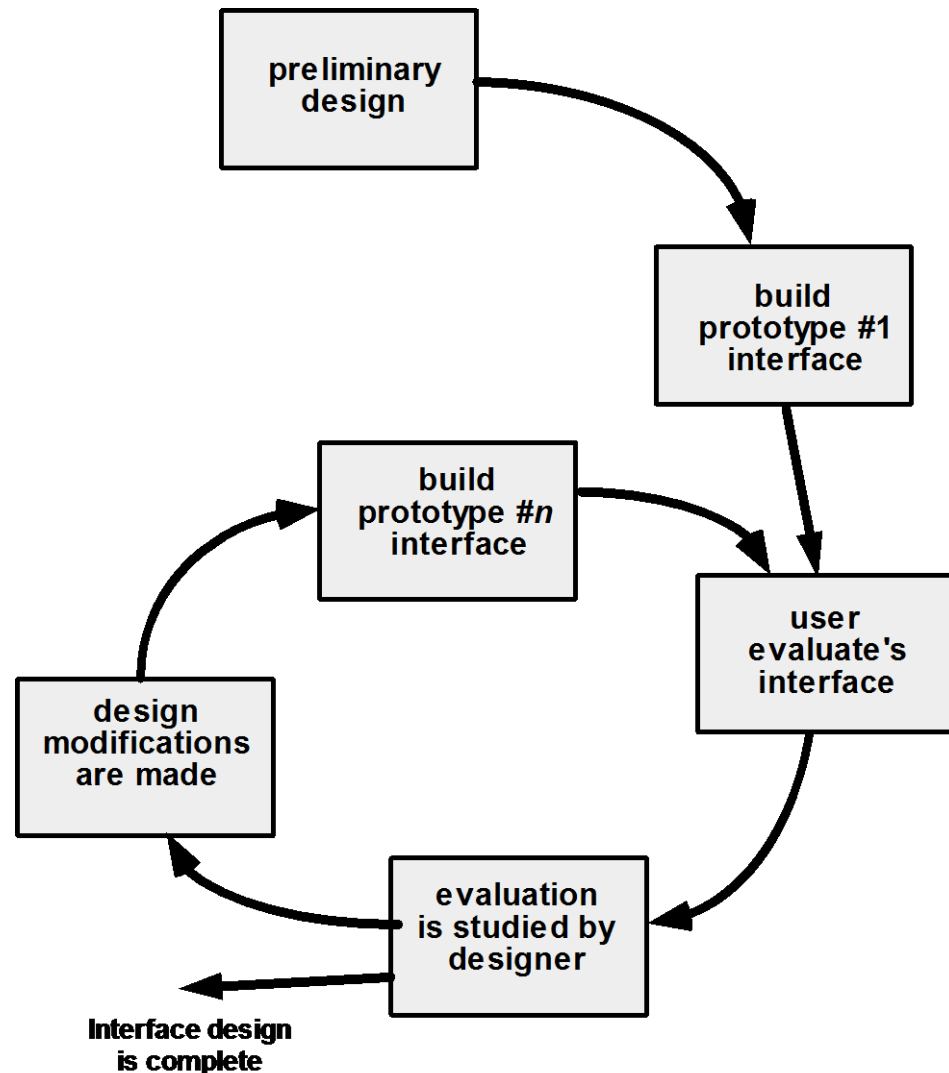
Dialogues Diagram

- Dialogues Diagram เป็น
แผนภาพแสดงลำดับการเชื่อมโยง
ของจอภาพ



การประเมินการออกแบบ

- หลังจากสร้างต้นแบบแล้ว จะต้องมีการ**ประเมินว่าเป็นไปตามความต้องการของผู้ใช้หรือไม่**
- ทีมงานต้อง**เก็บข้อมูลความคิดเห็นของผู้ใช้** เพื่อนำไปปรับปรุงต้นแบบให้สมบูรณ์ที่สุด



คุณภาพและการประเมินคุณภาพงานออกแบบซอฟต์แวร์

- สามารถใช้เทคนิคเดียวกับ **การเก็บรวบรวมข้อมูลของการวิศวกรรมความต้องการ**
 - การสัมภาษณ์
 - การสอบถาม
- แต่ทีมงานต้อง **นำข้อมูลมาวิเคราะห์**

เกณฑ์คุณภาพ (Quality Attributes)

- การทำงานของโปรแกรม
(Functionality)
 - ประเมินจากลักษณะ (Feature Set) และความสามารถ (Capability) ของโปรแกรม
 - ประเมินจากหน้าที่ทั่วไปของโปรแกรม และความปลอดภัยเมื่อต้องทำงานรวมเป็นระบบ
- ความสามารถในการใช้งาน
(Usability)
 - ผลตอบกลับจากการใช้งานของผู้ใช้
 - การใช้งานง่าย และเรียนรู้ง่าย

เกณฑ์คุณภาพ (ต่อ)

- ความน่าเชื่อถือ (Reliability)
 - วัดจากความถี่และความรุนแรงของความผิดพลาด
 - ความถูกต้องของผลลัพธ์ที่ได้ เวลาเฉลี่ยของความล้มเหลว (Mean Time To Failure: MTTF) ความสามารถในการกู้คืนระบบ และความสามารถในการคาดการณ์ได้ของโปรแกรม

$$MMTF = 0.3509 ?$$

เกณฑ์คุณภาพ (ต่อ)

- ประสิทธิภาพ (Performance)
 - วัดจากความเร็วของการประมวลผล
 - ระยะเวลาตอบสนอง ทรัพยากรที่ใช้ ปริมาณที่ทำได้ในช่วงเวลาหนึ่ง

เกณฑ์คุณภาพ (ต่อ)

- ความสามารถในการสนับสนุนการใช้งาน (Supportability) และ
ความสามารถในการบำรุงรักษา (Maintainability)
 - พิจารณาจากความสามารถในการเพิ่มเติมส่วนการทำงาน
 - ความสามารถในการแปลงการทำงาน และการบริการ
 - ความสามารถในการทดสอบ การทำงานข้ามระบบได้ และการจัด
สภาพแวดล้อมของระบบด้วย

การวิเคราะห์และประเมินคุณภาพ (Quality Analysis and Evaluation)

- ทบทวนงานออกแบบซอฟต์แวร์ (Software Design Review)
เทคนิคที่ช่วยให้การทบทวนงานออกแบบมีประสิทธิภาพ
 - การออกแบบควรนำเสนอด้านข้อมูล สถาปัตยกรรม ส่วนประสาน และคอมโพเน้นท์ที่ชัดเจน
- วิเคราะห์งานออกแบบ (Static Analysis) เทคนิคที่ช่วยให้การวิเคราะห์มีประสิทธิภาพ
- การจำลองสถานการณ์และการสร้างต้นแบบ (Simulation and Prototyping)

การวัดซอฟต์แวร์ (Software Measurement)

- การวัดสามารถใช้กับการประเมินหรือการประมาณการคุณลักษณะบางอย่าง แบ่งออกเป็น 2 ประเภท ดังนี้
 - การวัดการออกแบบเชิงฟังก์ชัน
(Function-Oriented Measure)
 - การวัดการออกแบบเชิงวัตถุ
(Object-Oriented Design Measure)

ตัวอย่าง

การวัดคุณลักษณะภายนอก ด้วยคุณลักษณะภายในของซอฟต์แวร์

คุณลักษณะภายนอก (External Attribute)	คุณลักษณะภายใน (Internal Attribute)
ความสามารถในการบำรุงรักษา (Maintainability) วัดได้จากหน่วยวัดลำดับที่ ?	1. จำนวนพารามิเตอร์ในแต่ละกระบวนการ
ความไว้วางใจได้ (Reliability) วัดได้จากหน่วยวัดลำดับที่ ?	2. ความซับซ้อนของปัญหา (Cyclomatic complexity)
ความเข้ากันได้ (Portability) วัดได้จากหน่วยวัดลำดับที่ ?	3. ขนาดของซอฟต์แวร์ (แบบ LoC/FP)
ความสามารถในการใช้งาน (Usability) วัดได้จากหน่วยวัดลำดับที่ ?	4. จำนวน Error Message
	5. ความยาวของคู่มือการใช้งาน

สรุป

- **ส่วนประสานกับผู้ใช้ (User Interface)** หมายถึง ส่วนติดต่อระหว่างผู้ใช้กับระบบ เพื่อการเตรียมสารสนเทศ การทำงานและการนำเสนอสารสนเทศนั้นไปใช้ด้วยการโต้ตอบกับคอมพิวเตอร์ ซึ่งมีหลายชนิด ได้แก่ รายงาน การป้อนข้อมูล เอกสาร และการโต้ตอบกับระบบ
- **กระบวนการออกแบบส่วนประสาน** ประกอบด้วย (1) วิเคราะห์ผู้ใช้และระบบ (2) ออกแบบโครงสร้างส่วนประสาน (3) ออกแบบมาตรฐานส่วนประสาน (4) สร้างต้นแบบที่ใช้งานจริง (5) ผู้ใช้ประเมินผล
- **เกณฑ์คุณภาพของซอฟต์แวร์** ประกอบด้วย การประเมินจากลักษณะ ดังนี้ Functionality, Usability, Reliability, Performance, Supportability และ Maintainability
 - การวัดคุณภาพของซอฟต์แวร์ต้องอาศัย **การวัดซอฟต์แวร์** เพื่อให้ได้ค่าที่เป็นตัวเลขของคุณลักษณะบางอย่างของซอฟต์แวร์หรือกระบวนการผลิต

กิจกรรมท้ายบท

- จงบอกความสำคัญของการออกแบบส่วนประสานมาพอสังเขป
- บัญญัติกฎ 3 ข้อในการออกแบบส่วนประสานของ Theo Mandel มีกี่ข้อประกอบด้วยอะไรบ้าง พร้อมกับอธิบายแต่ละข้อพอสังเขป
- อะไรคือข้อควรคำนึงในการออกแบบ จงอธิบายและยกตัวอย่างประกอบในแต่ละข้อมาพอสังเขป
- รูปแบบของ User Interfaces มีกี่รูปแบบประกอบด้วยอะไรบ้าง พร้อมกับอธิบายแต่ละรูปแบบมาพอสังเขป
- จงบอกเกณฑ์ประเมินคุณภาพของซอฟต์แวร์ ประกอบด้วยเกณฑ์อะไรบ้าง พร้อมกับอธิบายแต่ละเกณฑ์มาพอสังเขป
- กำหนดให้วิเคราะห์และออกแบบลำดับของการแสดงส่วนติดต่อกับผู้ใช้ของ ระบบกลางในการใช้บริการห้องพักรักษาอินเทอร์เน็ต โดยใช้แผนภาพแสดงลำดับการเชื่อมโยงจอภาพ (Dialogues Diagram)

เอกสารอ้างอิง

- กิตติ ภัคดีวัฒนะกุล, วิศวกรรมซอฟต์แวร์ (Software Engineering), กรุงเทพฯ: เคทีพี คอมพ์ แอนด์ คอนซัลท์, 2552
- Alan Dennis, Barbara Haley Wixom and Roberta M, Systems Analysis and Design Fifth Edition, John Wiley and Sons, Inc. 2012.
- Lan Sommerville, Software Engineering Ninth Edition, Pearson Education, Inc., publishing as Addison-Wesley, 2011.