

ISAD - User Interface Design

Date: 8/3/2560

Introduction

Interface design เป็นสิ่งที่จะแสดงให้เห็นว่าผู้ใช้จะตอบสนองกับระบบอย่างไร แบ่งเป็นสองส่วนคือ

- System Interfaces เป็นส่วนต่อประสานระหว่าง machine - machine
- User Interfaces เป็นส่วนต่อประสานระหว่างมนุษย์กับเครื่องจักร

หลักสำคัญของ User interfaces design

Layout

วัตถุต่าง ๆ ในระบบควรถูกจับกลุ่มแยกจากกัน ไม่ปนกัน และแต่ละพื้นที่เป็น self-contained และข้อมูลและการออกแบบควรคำนึงถึงการออกแบบตามเชื้อชาติ เช่น คนฝรั่งอ่านหนังสือจากซ้ายไปขวา แต่ญี่ปุ่นอ่านจากขวาไปซ้าย

Content Awareness

ควรมี title ในแต่ละหน้าของระบบ, เมนูต่าง ๆ ควรอยู่ในที่ที่ผู้ใช้รู้ว่ามันควรจะอยู่ตรงนั้น, ส่วนต่าง ๆ ของระบบควรจับเป็นกลุ่มเข้าด้วยกันและอยู่ในพื้นที่ที่ผู้ใช้เห็นได้ชัดเจน และแต่ละส่วนควรแบ่งเป็นสัดส่วนของตนเอง ไม่ปะปนกันไปกับหมวดอื่น ๆ

Aesthetics

ระบบควรดูน่าใช้ มองแล้วสบายตา ดูเรียบง่ายและไม่ซับซ้อน การใช้งานสีและฟอนต์ที่ดี มีพื้นที่ว่างพอเหมาะไม่อัดเนื้อหาหรือองค์ประกอบดีไซน์มากเกินไป โดยต้องมีเนื้อหาที่ว่างมากพอสำหรับการยอมรับของผู้ใช้แต่ละกลุ่ม

Acceptable information density is proportional to the user's expertise

- Novice users prefer lower density (< 50%)
- Expert users prefer higher density (> 50%)

User Experience

ความง่ายดายในการใช้งานสำหรับผู้ใช้งานใหม่ ๆ ง่ายต่อการเรียนรู้ในการใช้งานระบบในครั้งแรก ๆ โดยสามารถทำได้โดยไม่ต้องเปิดคู่มือหรือเปิดให้น้อยที่สุด

Consistency

ความสอดคล้องกันของเนื้อหา หรือดีไซน์ของระบบ ทั้งระบบควรอยู่ในโทนหรือธีมเดียวกันทั้งระบบ ทั้งรูปแบบการใช้งานและสีสันเอฟเฟกต์ เพื่อให้ผู้ใช้สามารถคาดการณ์วิธีการใช้งานได้โดยไม่ต้องเรียนรู้ใหม่เมื่อต้องมาเจอฟีเจอร์อื่น ๆ

สิ่งสำคัญของคอนเซ็ปต์การออกแบบระบบ

- ผู้ใช้ต้องสามารถเดาได้ว่าต้องเจอรูปแบบการใช้งานแบบไหนในหน้าต่อ ๆ ไปของระบบ
- ทุกส่วนของระบบต้องสามารถใช้งานหรือทำงานได้ในรูปแบบเดียว ๆ กัน
- ผู้ใช้ต้องสามารถเรียนรู้ได้ว่าแต่ละส่วนต้องทำงานไ้ และทำงานกับมันได้ทันทีเมื่อต้องพบเจอ

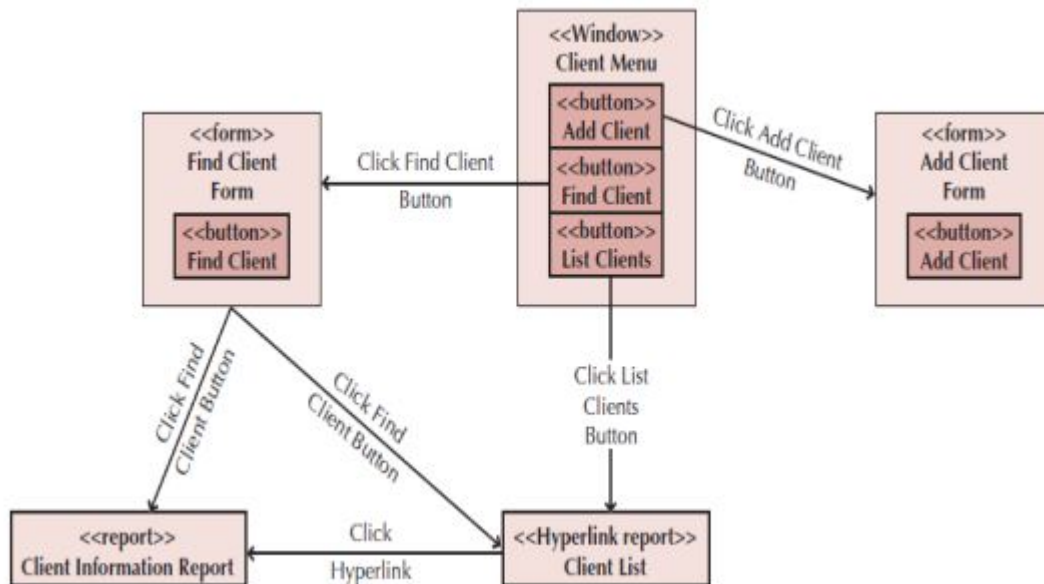
Minimal user effort

ระบบต้องออกแบบให้ไม่ต้องใช้ความพยายามมากในการใช้งาน เช่น หลักการพื้นฐาน **three-clicks rule** คือ ผู้ใช้ไม่ควรจ้องกดเมาส์เกินสามครั้งในการทำงานอย่างใดอย่างหนึ่ง

User interface design process

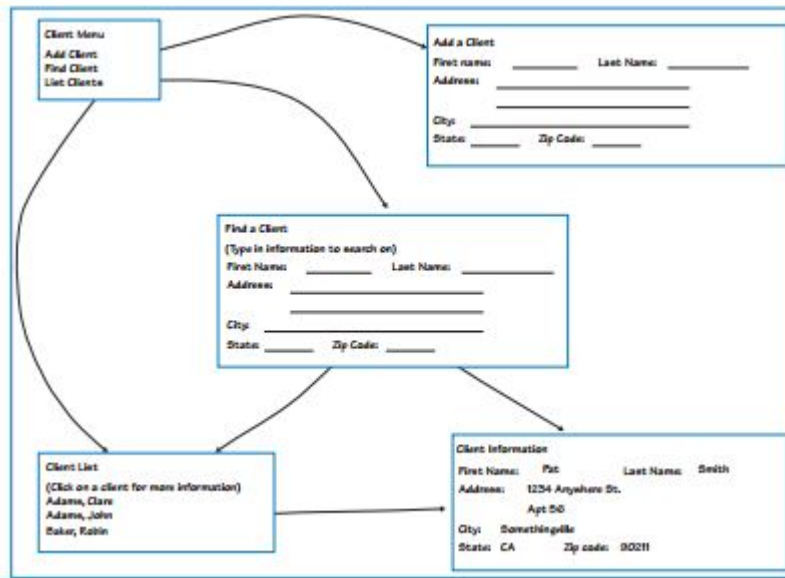
ขั้นตอนการออกแบบ User interface design มี 5 ขั้นตอนดังนี้

1. **Use scenario development** คือ การใช้สถานการณ์จริงของผู้ใช้เพื่อนำไปช่วยในการออกแบบ UI
2. **Interface structure design** สร้างโครงสร้างสำคัญของ UI โดยกำหนดองค์ประกอบพื้นฐานที่จำเป็นต้องมี ว่าต้องมีปุ่มหรือฟอร์มอะไรบ้าง มีหน้าต่างอะไรบ้าง โดยอาจออกแบบคร่าว ๆ ได้จาก Windows navigation diagrams (WND)



3. **Interface standard design** กำหนดพื้นฐานส่วนประกอบของ UI โดยต้องสิ่งเหล่านี้ในการออกแบบ
 1. **Interface metaphor** ระบุว่า interface จะทำงานอย่างไร เช่น shopping cart จะเก็บสินค้าได้จากการกดซื้อเท่านั้น
 2. **Interface template** ระบุภาพลักษณ์ (General appearance) ของโปรแกรมโดยมาตรฐานการออกแบบนี้มีเพื่อช่วยให้ระบบมีพื้นฐานการออกแบบเหมือน ๆ กันไม่หลุดธีม, สามารถระบุชื่อให้แต่ละการทำงานในหน้า interface
4. **Interface design prototyping** เป็นสิ่งที่เหมือนระบบจริงทุกอย่าง แต่มีแค่ UI ระบบด้านหลังยังไม่ต้องมี โดยอาจทำแบบ Mock-ups หรือ simulation ตัวอย่างหน้าจอ โดยมีสามวิธีในการทำงานส่วนนี้ (เรียงตามความละเอียดของข้อมูลรายละเอียดจากน้อยไปมาก)

1. Storyboard อธิบายว่าผู้ใช้เมื่อเข้ามาต้องมาเจออะไรบ้างในระบบ



2. HTML Prototype เป็นเว็บเพจธรรมดาที่มีลิงค์สามารถกดไปมาได้

3. Language prototype เขียนด้วยภาษาเดียวกับที่ต้องการจะใช้งานในระบบจริง

5. Interface evaluation การประเมินงานที่ออกแบบก่อนระบบจะเสร็จสมบูรณ์ โดยต้องใช้คนให้มากที่สุดเท่าที่ทำได้ โดยตามทฤษฎี การประเมินนี้จะต้องเสร็จก่อนการสร้างจริง ๆ เพื่อช่วยขจัดปัญหาต่าง ๆ และลดการเปลี่ยนแปลงระหว่างพัฒนาโดยประเมินได้สี่วิธีดังนี้

1. Heuristic evaluation เปรียบเทียบดีไซน์กับหลักหรือกฎที่ตั้งไว้ตั้งแต่แรก

2. Walkthrough evaluation นำเสนอสิ่งที่ออกแบบมาให้กับ User เพื่อแจ้งรายละเอียดการออกแบบและการทำงานต่าง ๆ

3. Interactive evaluation ผู้ใช้จะลองใช้งาน Prototype ร่วมกับสมาชิกในทีมพัฒนา

4. Formal usability testing ประเมินผลในห้องทดลองด้วยผู้ใช้งานบน language prototype

Common Sense Approach to UI Design

- ผู้ใช้งานไม่ควรต้องมาคิดมากว่าจะต้องใช้งานระบบอย่างไร
- จำนวนการคลิกต้องสอดคล้องกับความซับซ้อนของงาน
- ไม่ใช่คำพูดเร้นเรื่อเยอะเกินบนหน้าจอของระบบ

Navigation Design

ระบบควรบอกให้ผู้รู้เสมอว่ากำลังทำอะไรอยู่ตรงไหน เช่น แจ้งตลอดว่า action ล่าสุดล้มเหลวหรือสำเร็จ ส่วนประกอบต่าง ๆ ในระบบควรสามารถเปิดให้ผู้เข้าถึงผ่านระบบได้ โดยสิ่งต่าง ๆ ในการนำทางผู้ใช้ไปที่ไหนสักแห่งควรเข้าใจง่าย

Types of Navigation Controls

การพาผู้ใช้ไปในส่วนต่าง ๆ ของระบบ หรือทำงานต่าง ๆ ในระบบสร้างได้สามแบบคือ

1. **Language** ทำงานแบบ command line
2. **Menu**
3. **Direct manipulation** เช่นการลากวาง

Messages

มีข้อความแจ้งสถานะต่าง ๆ ตลอด เช่นข้อผิดพลาด, ความสำเร็จ, สถานะรอโหลด, ข้อความช่วยเหลือ

Input Design

ข้อมูลทั้งสองแบบ คือ แบบมีโครงสร้าง (ชื่อ วันเดือนปี เบอร์โทร) และ แบบไม่มีโครงสร้าง (คอมเมนต์ต่าง ๆ)

Types of inputs

1. **Free form controls** กล่องข้อความต่าง ๆ
2. **User input options**
3. **Selection boxes** เช่น checkbox, radio button, list box

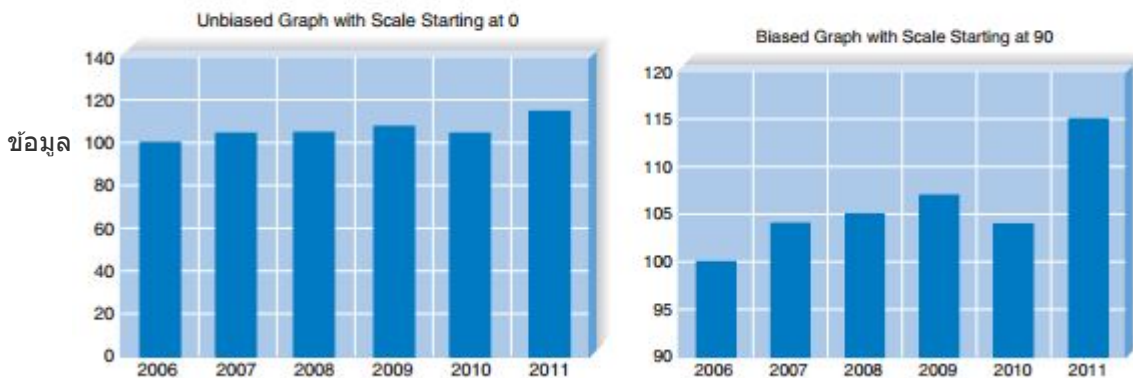
Input validation

ข้อมูลต่าง ๆ ควรตรวจสอบก่อนบันทึก เช่น ช่องกรอกเงิน ห้ามกรอกอย่างอื่นเข้าไป ห้ามข้อมูลที่ผิดประเภทหลุดเข้าไปในระบบเด็ดขาด

Output Design

ข้อมูลที่น่าเสนอต้องเป็นข้อมูลที่เข้าใจได้ และนำเสนอถูกรูปแบบ เช่น ข้อมูลสถิติบางอย่างอาจแสดงด้วยกราฟดีกว่าแสดงด้วยตาราง และต้องทราบถึงจุดประสงค์การใช้งานของข้อมูลเพื่อการออกแบบที่เหมาะสมมากขึ้น และสิ่งสุดท้ายคือการลดอคติใน

Example of bias: Bias in graphs.



ส่วนหลังจากนี้ ให้อ่านเองจากสไลด์ เพราะเนื้อหาไม่สำคัญ