INFOPRESS

Fundamental of SOFTWARE ENGINEERING & Digital Transformation

Software Engineering : An Important Role in Digital Transformation

อธิบายกระบวนการ Digital Transformation
ด้วย Software Engineering ถ่ายทอดจากประสบการณ์
ในการพัฒนาระบบเทคโนโลยีดิจิทัลให้กับหน่วยงาน
ทั้งภาครัฐและเอกชน

ผู้แต่ง รศ. ดร.ปานใจ ธารทัศนวงศ์
บรรณาธิการ กีรพล คชาจริญ

CONTENT

CHAPTER 1

การปรับเปลี่ยนสู่ดิจิทัล DIGITAL TRANSFORMATION 1

1.1 เศรษฐกิจดิจิทัล (Digital Economy) 2

1.2 ความหมายของเทคโนโลยีดิจิทัล (Digital Technology Definition) 3

1.3 องค์กรดิจิทัล (Digital Enterprises) 5

1.4 การปรับเปลี่ยนสู่ดิจิทัล (Digital Transformation) 6

1.4.1 ขั้นตอนการปรับเปลี่ยนสู่ดิจิทัล (Digital Transformation Roadmap) 6

1.4.2 กรอบการปรับเปลี่ยนสู่ดิจิทัล (Digital Transformation Framework) 8

1.5 บทบาทของวิศวกรรมซอฟต์แวร์ในการปรับเปลี่ยนสู่ดิจิทัล (The Role of Software Engineer in Digital Transformation) 12

1.5.1 ความสำคัญของวิศวกรรมซอฟต์แวร์ 12

1.5.2 ความท้าทายของการปรับเปลี่ยนสู่ดิจิทัลในวิศวกรรมซอฟต์แวร์ 13

1.5.3 แนวทางการเอาชนะความท้าทายของการปรับเปลี่ยนสู่ดิจิทัลในด้านวิศวกรรมซอฟต์แวร์ 15

1.6 คำนิยามและคำจำกัดความที่ใช้ในวิศวกรรมซอฟต์แวร์ 17

CHAPTER 2

การวิเคราะห์เชิงโครงสร้าง STRUCTURED ANALYSIS 23

2.1 การวิเคราะห์เชิงโครงสร้าง (Structure Analysis And Design) 24

2.2 แผนภาพการไหลของข้อมูล (Data Flow Diagram) 25

2.2.1 แผนภาพบริบท (Context Diagram) 26

2.2.2 แผนภาพการไหลของข้อมูลระดับที่ 0 (DFD Level 0) 27

2.2.3 การแยกองค์ประกอบ (Decomposition) 28

2.3 ตัวอย่างการใช้แผนภาพการไหลของข้อมูลกับการปรับรื้อกระบวนการทางธุรกิจ (Using DFD in BPR) 30

CHAPTER 3

การวิเคราะห์และออกแบบเชิงวัตถุ OBJECT-ORIENTED ANALYSIS AND DESIGN 33

3.1 กระบวนทัศน์เชิงวัตถุ (Object-Oriented Paradigm) 34

3.2 หลักการเชิงวัตถุ (Object-Oriented Concept) 36

3.3 นามธรรม (Abstraction) 37

3.3.1 คลาส (Class) และอ็อบเจ็กต์ (Object) 38

CONTENT

3.3.2 การสื่อสารของออบเจ็กต์ (Object Communication) 40
3.3.3 ปฏิสัมพันธ์ระหว่างออบเจ็กต์ (Interacting Objects) 41
3.3.4 การสื่อสารของออบเจ็กต์ (Object Communication) 42
3.4 การห่อหุ้ม (Encapsulation) 42
3.5 ความเป็นโมดูลาร์ (Modularity) 43
3.6 ลำดับชั้น (Hierarchy) 43
3.6.1 ลำดับชั้นแบบ IS-A 43
3.6.2 ลำดับชั้นแบบ PART-OF 46
3.7 โพลิมอร์ฟิซึม (Polymorphism) 47
3.8 การวิเคราะห์และออกแบบเชิงวัตถุ (Object-Oriented Analysis and Design) 47
3.8.1 บริบทของระบบและรูปแบบการใช้งาน 48
3.8.2 การออกแบบคลาสและเมธอด 49
3.8.3 การระบุออบเจ็กต์และคลาส 49
3.8.4 แบบจำลองการออกแบบ 49
3.8.5 กำหนดส่วนต่อประสานของออบเจ็กต์ 50

CHAPTER 4

UML ภาษาการสร้างแบบจำลองแบบครบวงจร 53

4.1 ภาษาการสร้างแบบจำลอง (Modeling Language) 54
4.2 ความเป็นมาของภาษาการสร้างแบบจำลองแบบครบวงจร (Unified Modeling Language, UML) 55
4.3 แผนภาพยูเอ็มแอล (UML Diagram) 62
4.4 แผนภาพยูสเคส (Use Case Diagrams) 63
4.4.1 องค์ประกอบของยูสเคส (Use Case) 63
4.4.2 ความสัมพันธ์ระหว่างยูสเคส 64
4.4.3 หลักการวิเคราะห์ระบบด้วยยูสเคส 65
4.5 แผนภาพคลาส (Class Diagrams) 68
4.5.1 การวิเคราะห์ความทนทาน (Robustness Analysis) 74
4.5.2 ขั้นตอนในการทำแผนภาพคลาส 76
4.5.3 เทคนิคที่ดีของการทำแผนภาพคลาส 77
4.6 แผนภาพลำดับ (Sequence Diagrams) 77
4.6.1 ขั้นตอนในการเขียนแผนภาพลำดับ 81
4.6.2 แนวทางปฏิบัติที่ดีของแผนภาพลำดับ 84
4.6.3 กระบวนการสร้างแผนภาพลำดับด้วยการวิเคราะห์ความทนทาน 86
4.7 แผนภาพสถานะ (State Diagrams) 87
4.7.1 ขั้นตอนการเขียนแผนภาพสถานะ 89
4.8 แผนภาพกิจกรรม (Activity Diagrams) 91
4.8.1 ขั้นตอนการทำแผนภาพกิจกรรม 92
4.8.2 ตัวอย่างแผนภาพกิจกรรม 94
4.9 แผนภาพการนำไปใช้ (Deployment Diagrams) 95
4.9.1 ขั้นตอนการทำแผนภาพการนำไปใช้ 96

4.10 การสร้างรายงานโดยยูเทิมเลอ 98
4.11 การพัฒนาโปรแกรมโดยใช้แบบจำลอง 100
4.12 การพัฒนาด้วยวิศวกรรมย้อนกลับ (Reverse Engineering) 102

CHAPTER 5

กระบวนการซอฟต์แวร์ SOFTWARE PROCESS 107

5.1 แบบจำลองกระบวนการซอฟต์แวร์ (Software Process Models) 110
5.1.1 แบบจำลองการพัฒนาระบบแบบน้ำตก (Water Fall Model) 110
5.1.2 แบบจำลองการพัฒนาต้นแบบ (Prototyping Model) 114
5.1.3 แบบจำลองการพัฒนาระบบแบบมีส่วนร่วม (Joint Application Development Model) 114
5.1.4 แบบจำลองการพัฒนาซอฟต์แวร์อย่างเร็ว (Rapid Application Development Model) 115
5.1.5 การทำงานแบบอไจล์ (Agile Methodology) 116
5.1.6 แบบจำลองการพัฒนาโปรแกรมแบบเอ็กซ์ตรีม (Extreme Programming) 124
5.1.7 แบบจำลองการพัฒนาโดยวิธีเชิงวัตถุ (Object Oriented Development) 126
5.1.8 สถาปัตยกรรมเชิงเซอร์วิส (Service Oriented Architecture) 128
5.2 กิจกรรมในกระบวนการซอฟต์แวร์ (Process Activities) 129
5.2.1 การวิเคราะห์ข้อกำหนดของซอฟต์แวร์ (Software Specification) 130
5.2.2 การออกแบบซอฟต์แวร์และการนำไปใช้ (Software Design and Implementation) 132
5.2.3 การตรวจสอบความถูกต้องของซอฟต์แวร์ (Software Validation) 134
5.2.4 วิวัฒนาการของซอฟต์แวร์ (Software Evolution) 135

CHAPTER 6

การจัดการเทคโนโลยีดิจิทัล DIGITAL TECHNOLOGY MANAGEMENT 139

6.1 ปัญหาในการบริหารจัดการองค์กร 141
6.2 สถาปัตยกรรมองค์กร (Enterprise Architecture) 143
6.2.1 สถาปัตยกรรมองค์กรในปัจจุบัน (As is Enterprise Architecture) 148
6.2.2 สถาปัตยกรรมองค์กรในอนาคต (To be Enterprise Architecture) 149
6.3 สถาปัตยกรรมองค์กรกับวิศวกรรมซอฟต์แวร์ (Enterprise Architecture and Software Engineering) 154
6.4 การกำกับดูแลการใช้เทคโนโลยีดิจิทัล (Digital Technology Governance) 155

CONTENT

6.5 การจัดการซอฟต์แวร์ (Software Management) 159

6.5.1 การจัดการโครงการด้านซอฟต์แวร์ (Software Project Management) 159

6.5.2 การบริหารโครงการซอฟต์แวร์แบบอไจล์ (Agile Project Management) 165

6.5.3 การจัดการความเสี่ยง (Risk Management) 167

6.5.4 การประมาณการต้นทุนซอฟต์แวร์ (Software Cost Estimation) 172

6.6 การจัดการคุณภาพซอฟต์แวร์ (Software Quality Management) 174

6.6.1 การปรับปรุงคุณภาพซอฟต์แวร์ให้มีประสิทธิภาพ 175

6.6.2 CMMI กับการกระบวนการพัฒนาแบบอไจล์ 177

6.7 ปัจจัยสู่ความสำเร็จ (Keys Success Factors) 178

CHAPTER 7

วิศวกรรมความต้องการ REQUIREMENT ENGINEERING 181

7.1 ความหมายของความต้องการด้านซอฟต์แวร์ (Software Requirement Definition) 183

7.2 กระบวนการวิศวกรรมความต้องการ (Requirement Engineering Process) 185

7.3 กระบวนการจัดทำข้อกำหนดความต้องการด้านซอฟต์แวร์ (Software Requirement Specification Process) 189

7.4 การสกัดและวิเคราะห์ความต้องการ (Requirements Elicitation and Analysis) 190

7.5 การตรวจสอบความต้องการ (Requirements Validation) 222

7.6 การจัดการความต้องการ (Requirements Management) 223

7.7 วิศวกรรมความต้องการแบบอไจล์ (Agile Requirement Engineering) 224

CHAPTER 8

การออกแบบระบบ SYSTEM DESIGN 231

8.1 กระบวนการออกแบบระบบ (System Design) 234

8.2 การออกแบบสถาปัตยกรรม (Architectural Design) 235

8.2.1 ลักษณะการออกแบบสถาปัตยกรรมที่ดี 235

8.2.2 มุมมองการออกแบบสถาปัตยกรรม (Architecture View) 236

8.2.3 รูปแบบสถาปัตยกรรม (Architectural Pattern) 238

8.2.4 สถาปัตยกรรมแอปพลิเคชัน (Application Architecture) 244

8.2.5 การออกแบบสถาปัตยกรรมสำหรับกรณีศึกษา การพัฒนาระบบการออกใบอนุญาตก่อสร้างที่เป็นอันตรายต่อสุขภาพ 246

8.3 การออกแบบส่วนต่อประสาน (Interface Design) 250

8.3.1 การออกแบบฟอร์มและรายงาน (Designing Forms and Reports) 250

8.4 การออกแบบส่วนประกอบ (Component Design) 255

8.5 การออกแบบฐานข้อมูล (Database Design) ... 257

8.5.1 ธรรมาภิบาลข้อมูล (Data Governance) ... 258

8.5.2 การบริหารและการจัดการข้อมูลหรือวงจรชีวิตของข้อมูล (Data Life Cycle) 261

8.5.3 การบูรณาการข้อมูล (Data Integration) ... 262

8.6 รูปแบบการออกแบบ (Design Pattern) 263

CHAPTER 9

การติดตั้งระบบและการบำรุงรักษา SYSTEM IMPLEMENTATION AND MAINTENANCE 269

9.1 การพัฒนาและติดตั้งระบบ (System Implementation) 270

9.1.1 การพัฒนาโปรแกรม (Program Coding) ... 270

9.1.2 การพัฒนาด้วยวิศวกรรมย้อนกลับ (Reverse Engineering) 275

9.1.3 วิธีการพัฒนาโปรแกรมแบบ Agile และ DevOps 277

9.2 การทดสอบซอฟต์แวร์สำหรับแบบจำลองน้ำตก (Water Fall Model Software Testing) 281

9.2.1 การทดสอบการพัฒนา (Development Testing) 282

9.2.2 การทดสอบการเผยแพร่ (Release Testing) 286

9.2.3 การทดสอบโดยผู้ใช้ (User Testing) 287

9.3 การทดสอบซอฟต์แวร์สำหรับอไจล์ (Agile Software Testing) 290

9.4 การติดตั้งระบบ (System Installation) 292

9.4.1 การติดตั้งโดยตรง (Direct installation) ... 292

9.4.2 การติดตั้งแบบขนาน (Parallel installation) 293

9.4.3 การติดตั้งแบบนำร่อง (Single-Location installation) 293

9.4.4 การติดตั้งเป็นระยะ (Phase installation) ... 293

9.5 การจัดทำเอกสาร (Documentation) 294

9.6 การฝึกอบรม (Training) 295

9.7 การบำรุงรักษาซอฟต์แวร์ (Software Maintenance) 295

CHAPTER 10

วิศวกรรมซอฟต์แวร์เชิงเซอร์วิส SERVICE ORIENTED SOFTWARE ENGINEERING 299

10.1 วิศวกรรมซอฟต์แวร์เชิงเซอร์วิส (Service Oriented Software Engineering) 300

CONTENT

10.2 รูปแบบการพัฒนาซอฟต์แวร์เชิงเซอร์วิส.....304
10.3 เทคโนโลยีเว็บเซอร์วิส (Web Services).....307
10.3.1 เว็บเซอร์วิสแบบ SOAP (SOAP Web Services).....308
10.3.2 เว็บเซอร์วิสแบบ REST (REST Web Services).....315
10.3.3 จัดเปรียบเทียบระหว่างเว็บเซอร์วิสแบบ SOAP และ REST.....317
10.4 รูปแบบข้อมูลที่ใช้ในเว็บเซอร์วิส.....317
10.4.1 ภาษา XML.....318
10.4.2 JavaScript Object Notation (JSON).....318
10.4.3 มาตรฐานการแลกเปลี่ยนข้อมูล.....319
10.5 สถาปัตยกรรมเชิงเซอร์วิส (Service Oriented Architecture, SOA).....330
10.5.1 การประยุกต์ใช้สถาปัตยกรรมเชิงเซอร์วิส (SOA).....330
10.5.2 ประโยชน์ในการพัฒนาด้วยเทคโนโลยี SOA.....330
10.5.3 ขั้นตอนในการพัฒนาสถาปัตยกรรมเชิงเซอร์วิส (SOA).....331
10.6 ไมโครเซอร์วิส (Microservices).....337
10.7 ซอฟต์แวร์คอนเทนเนอร์ (Software Container).....342

CHAPTER 11

การพัฒนาบุคลากรด้านดิจิทัล DIGITAL HUMAN RESOURCE DEVELOPMENTS 347

11.1 ทักษะด้านดิจิทัล.....348
11.2 สมรรถนะของบุคลากรด้านการพัฒนาเทคโนโลยีดิจิทัล.....354
11.3 การพัฒนาบุคลากรด้านการพัฒนาเทคโนโลยีดิจิทัล.....355
11.3.1 ระดับประธานฝ่ายสารสนเทศ (Chief Information Officer, CIO).....355
11.3.2 ระดับผู้บริหารโครงการ (Project Management Officer, PMO).....356
11.3.3 ระดับผู้พัฒนาระบบ (Developer).....356
11.3.4 ระดับผู้ดูแลระบบ (System Administrator).....358
11.3.5 ระดับผู้ดูแลโครงสร้างพื้นฐาน (Network/Infrastructure).....359