

บทที่ 2

กรอบแนวคิดทางทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 System Development Life Cycle (SDLC)

2.1.1 คำจำกัดความของ SDLC

The Systems Development Life Cycle หรือ SDLC คือ กระบวนการของการสร้างหรือพัฒนาขั้นตอนต่างๆ รวมถึงเป็นต้นแบบ และวิธีการซึ่งผู้ใช้นำไปพัฒนาระบบที่เกี่ยวข้องกับด้านคอมพิวเตอร์ หรือ ระบบสารสนเทศ (วิกิพีเดีย, 2009)

System Development Life Cycle หรือ SDLC คือ กระบวนการ ที่เกี่ยวข้องกับหลากหลายขั้นตอน (เริ่มจากการศึกษาความเป็นไปได้ไปจนถึงหลังการติดตั้ง) ซึ่งถูกใช้ในการแปลงความต้องการด้านการจัดการไปยังระบบแอปพลิเคชัน ซึ่งอาจจะเป็นการพัฒนาจากรูปแบบเดิม (Custom-developed) หรือ เป็นการซื้อมา หรืออาจจะเป็นได้ทั้งสองอย่างรวมกัน (ISACA, 2009)

System Development Life Cycle หรือ SDLC คือ วงจรการพัฒนาระบบ เป็นวงจรที่แสดงถึงกิจกรรมต่างๆที่เป็นลำดับขั้นตอนในการพัฒนาระบบ (ฝ่ายผลิตหนังสือตำราวิชาการคอมพิวเตอร์, 2551)

สรุปความหมายของ System Development Life Cycle หรือ SDLC คือ กระบวนการหรือวงจรในการพัฒนาระบบด้านคอมพิวเตอร์หรือระบบสารสนเทศ โดยแปลงจากความต้องการของผู้ใช้งานมาเป็นในรูปแบบของระบบแอปพลิเคชัน โดยมีการกำหนดกิจกรรมต่างๆที่เกิดขึ้นในแต่ละระยะของการพัฒนาระบบขึ้นอย่างชัดเจนตั้งแต่ระยะเริ่มแรกไปจนถึงหลังระยะสิ้นสุดการพัฒนาระบบ

2.1.2 ที่มาของ SDLC

SDLC เป็นประเภทของวิธีการที่ใช้ในการอธิบายถึงกระบวนการสำหรับการสร้างหรือพัฒนาข้อมูลสารสนเทศ เพื่อเป็นการพัฒนาระบบสารสนเทศอย่างรอบคอบ มีโครงสร้าง และมีระเบียบแบบแผน เนื่องจาก SDLC นั้นมีการทวนซ้ำเป็นรอบในแต่ละขั้นตอนของวงจรพัฒนาระบบสารสนเทศ วงจรการพัฒนาระบบแบบดั้งเดิมกำเนิดขึ้นในปี ค.ศ. 1960 เพื่อนำมาพัฒนาในการใช้งานในระบบธุรกิจขนาดใหญ่ ซึ่งเป็นช่วงที่มีการรวมกลุ่มของธุรกิจขนาดใหญ่ ทำให้กิจกรรมของข้อมูลสารสนเทศนั้นเป็นกระบวนการที่ประกอบไปด้วยข้อมูลขนาดใหญ่ และมีจำนวนในการเก็บข้อมูลประเภทงานประจำอยู่เป็นจำนวนมาก (Boggs, 2004)

ในปี ค.ศ.1980 โครงสร้างการวิเคราะห์ระบบและวิธีการออกแบบ (Structured Systems Analysis and Design Method) หรือ SSADM นั้นพัฒนาอยู่บนพื้นฐานของ SDLC SSADM เป็นระบบที่ทำการวิเคราะห์และออกแบบระบบสารสนเทศ พัฒนาขึ้นสำหรับ กระทรวงพาณิชย์ของสหราชอาณาจักร กระทั่งปี ค.ศ. 1980 วงจรระบบแบบดั้งเดิมได้มีการพัฒนาเพื่อนำมาใช้แทนที่ระบบเดิม โดยมีการเพิ่มในเรื่องของการพิจารณาทางเลือกและกรอบการดำเนินงานที่พยายามปรับปรุงในบางส่วนที่บกพร่องของระบบ SDLC แบบดั้งเดิม (วิกิพีเดีย, 2009)

SDLC ยังคงได้รับความนิยมมากกว่าครึ่งศตวรรษ และได้กลายมาเป็นกระบวนการที่แพร่หลายอย่างรวดเร็ว ซึ่งนักพัฒนาระบบส่วนใหญ่นำมาปฏิบัติตามในการวิเคราะห์และมีการทวนซ้ำจนกระทั่งได้รับวิธีการที่ถูกต้อง ประวัติของ SDLC เมื่อกลางปี ค.ศ.1960 เกิดขึ้นเพื่อการปฏิบัติงานสำหรับแผนการป้องกันของ USA A. Enthoven and Henry Rowan ได้พัฒนาระบวนการที่ช่วยในการแก้ปัญหาสำหรับการบริหารโครงการพัฒนาระบบสารสนเทศขนาดใหญ่ SDLC ได้ถูกพัฒนาขึ้นในแบบเชิงเส้น (Linear) เป็นการพัฒนาโครงการที่สามารถนำความต้องการกลับมาใช้อีกได้ ซึ่งการวิเคราะห์ระบบนั้นมีการใช้ SDLC กว่าครึ่งทศวรรษและ SDLC ยังใช้งานได้ดีกว่าวิธีการแบบอื่นๆ อีกทั้งยังมีกระบวนการที่มีประสิทธิภาพมากพอที่จะเป็นยึดเป็นแนวทางในการปฏิบัติได้ (Boggs, 2004)

SDLC เป็นการกำหนดขั้นตอนที่คงที่สำหรับการพัฒนาระบบ รวมถึงการประเมินความสามารถของระบบที่เกิดขึ้นในขั้นตอนสุดท้าย ซึ่งเหมาะกับการพัฒนาระบบขนาดใหญ่ มีความซับซ้อน มีการปฏิบัติที่ไม่แน่นอน และเหมาะสมกับสภาพแวดล้อมการดำเนินงานแบบประจำ ซึ่ง Model ของ SDLC ในแต่ละระยะนั้นจะสอดคล้องกับกิจกรรมตามลำดับขั้นที่กำหนดไว้ โดยแต่ละระยะในการพัฒนาจะต้องสมบูรณ์ก่อนที่จะเริ่มพัฒนาระยะถัดไป (Kushniruk, (2002), Watt and Willey, (2003))

ขั้นตอนการวิเคราะห์ระบบของ SDLC นั้นมีความสำคัญในการช่วยลดต้นทุนของโครงการลงได้ (He and Griggs, 1994) ซึ่ง SDLC ได้ถูกพัฒนาในลักษณะแบบโครงสร้างและปรับปรุงการจัดการของการพัฒนาระบบตลอดจนมีการระบุระยะและกิจกรรมของ SDLC ที่หลากหลายในแต่ละระยะ ซึ่งกิจกรรมเหล่านี้ได้กำหนดถึง สิ่งที่จะต้องปฏิบัติเพื่อให้บรรลุผล และมีการกำหนดเครื่องมือในกิจกรรมเหล่านี้อย่างเหมาะสม (Nasution and Weistroffer, 2009)

2.1.3 เป้าหมายของ SDLC

Maryland, (2008) ได้กล่าวถึงเป้าหมายของ SDLC ประกอบไปด้วย ดังต่อไปนี้

- ส่งมอบระบบที่มีคุณภาพและตรงตามความคาดหวังหรือเกินกว่าความคาดหวังของลูกค้าหรือผู้ใช้งาน
- จัดทำกรอบการทำงานในการพัฒนาระบบอย่างมีคุณภาพโดยต้องสามารถ พิสูจน์ได้ (Identifiable) วัดผลได้ (Measurable) และสามารถทำซ้ำกระบวนการเดิมได้ (Repeatable process)
- เพื่อให้เกิดการพัฒนาระบบสารสนเทศของโครงการนั้นมีการจัดการอย่างมีประสิทธิภาพตลอดทั้งโครงการ
- กำหนดและมอบหมายหน้าที่ความรับผิดชอบของผู้ที่เกี่ยวข้องทั้งหมด รวมถึงหน้าที่ของผู้จัดการด้านปฏิบัติงานและด้านเทคนิคตลอดทั้งการพัฒนาระบบ

2.1.4 SDLC Model

Boggs, (2004) ได้ระบุถึง Model ของ SDLC มีทั้งหมด 6 รูปแบบดังต่อไปนี้

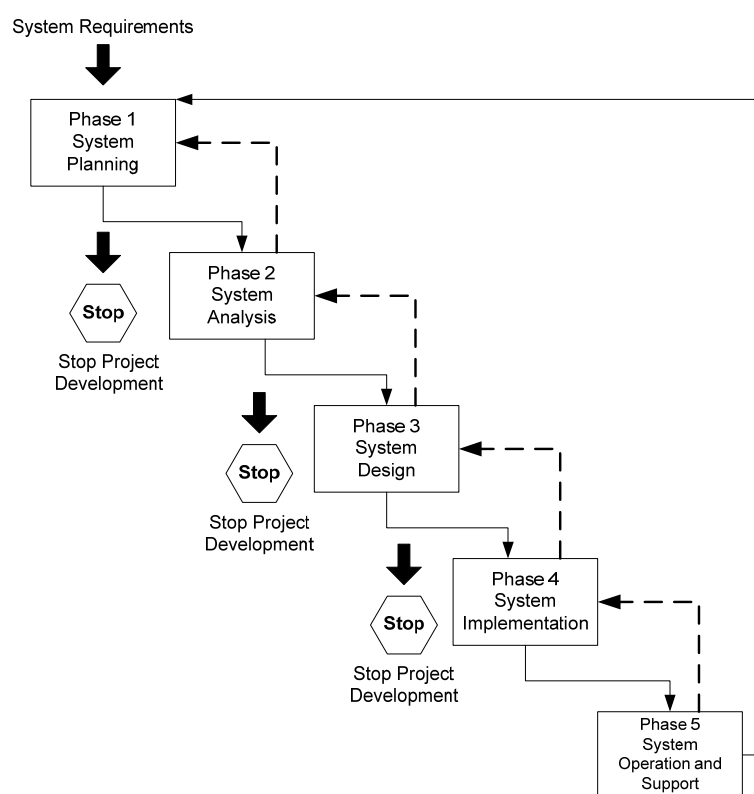
1. Waterfall Model

Winston Royce ได้แนะนำ Model ของ SDLC แบบแรกในปี ค.ศ. 1970 ซึ่ง Model นี้ได้กลายมาเป็นที่รู้จักกันในชื่อ “Waterfall-Model” หรือเป็นที่รู้จักในชื่อ “Linear Sequential Model” หรือ “Classic Life Cycle” ลำดับขั้นทั้งหมดของ Model นี้สามารถที่จะสรุปออกมาได้เป็น 4 ขั้น คือ การค้นพบ (Discover) การออกแบบ (Design) การพัฒนา (Develop) และการส่งมอบ (Deliver)

เริ่มแรก Model นี้ไม่ได้ตระหนักถึง กลุ่มผู้ใช้งาน (End-User) เข้าไปมีส่วนร่วมในขอบเขตการพัฒนา ระบบ จึงทำให้เกิดต้นทุนในการดำเนินงานเกินกว่าที่คาดการณ์ไว้ ซึ่งความผิดพลาดนี้ทำให้เกิดประสบการณ์ในการวางแผนด้านการเงินของโครงการ รวมทั้งทำให้ตระหนักถึงในเรื่องของฮาร์ดแวร์ที่ต้องใช้ในโครงการอีกด้วย

Model นี้โดยมากแล้วจะอยู่ในขอบเขตความรับผิดชอบของวิศวกรระบบ (Engineer)

ภาพที่ 2-1 Waterfall Model



ที่มา: The SDLC and SIX SIGMA an essay on which is which and why (Boggs, 2004)

ภักดีวัฒนะกุล, (2551) ได้กล่าวข้อดีของ Waterfall Model คือ

- มีการจัดทำเอกสารในทุกๆขั้นตอนหรือทุกระยะ
- ดำเนินงานทีละขั้นตอน ไม่มีการทำลัดขั้นตอน ทำให้ตรวจสอบการทำงานได้ง่าย
- ขอบเขตงานมีความชัดเจนแน่นอน ทำให้จัดการได้ง่าย

- เหมาะกับระบบทั้งขนาดเล็กและขนาดใหญ่

นักวิวัฒนาการ, (2551) ได้กล่าวข้อเสียของ Waterfall Model คือ

- ใช้เวลาในขั้นตอนการวางแผน วิเคราะห์ และออกแบบนานเกินไป
- ผู้ใช้ได้เห็นระบบก็ต่อเมื่อผ่านขั้นตอนการพัฒนาไปแล้ว ทำให้ไม่สามารถแก้ไขระบบได้ทันตามความต้องการของผู้ใช้ที่เปลี่ยนแปลงไป
- ต้องมีการวางแผนการทำงานที่ดี
- ทีมงานและนักวิเคราะห์จะต้องเป็นผู้มีประสบการณ์และความชำนาญ
- หากวางแผนไม่รอบคอบ จะทำให้โครงการไม่ประสบความสำเร็จหรือไม่ก็ต้องใช้ต้นทุนที่สูงเกินไป

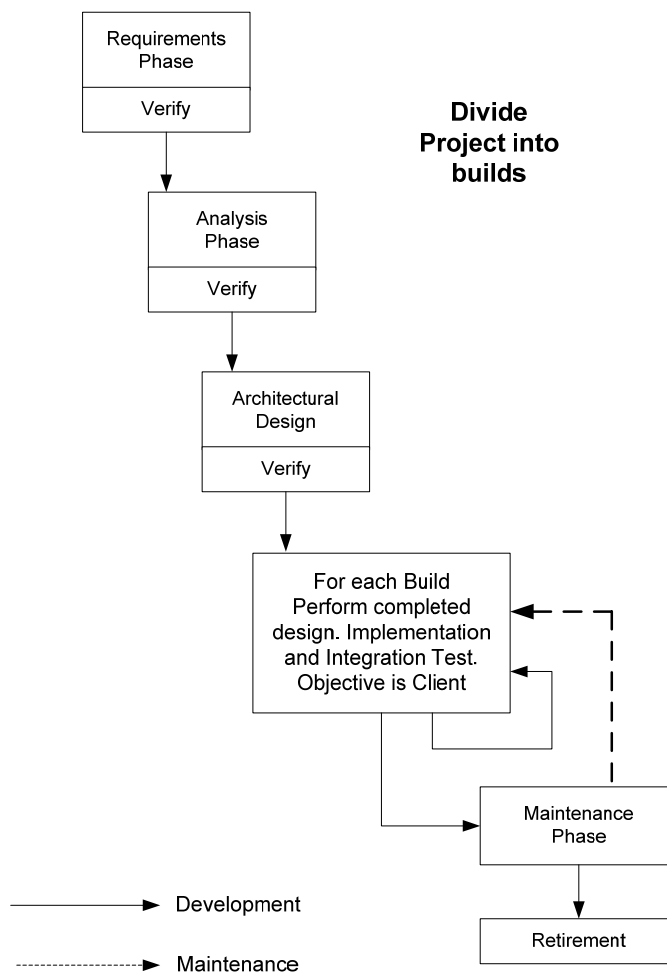
2. Incremental Model

Model นี้ได้เกิดขึ้นในช่วงเริ่มแรกของปี ค.ศ. 1980 เพื่อเพิ่มความยืดหยุ่นในการพัฒนาระบบที่มากขึ้น เนื่องจากกระบวนการแบบเชิงเส้น (Linear) นั้นยังไม่ประสบความสำเร็จมากนักในตลอดช่วงเวลาที่ผ่านมา ซึ่งในแต่ละลำดับขั้นของ Model นี้ก่อให้เกิดการส่งมอบผลลัพธ์ (Output) เพิ่มมากขึ้นตลอดทั้งโครงการ และคุณสมบัติที่เพิ่มขึ้นของ Model นี้คือ จะไม่พัฒนาในขั้นถัดไป จนกว่าผู้ใช้งานจะทำการทวนซ้ำและอนุมัติการสร้างของขั้นตอนในปัจจุบันอย่าสมบูรณ์ กระบวนการจะเกิดขึ้นอย่างนี้เรื่อยไปจนได้ระบบที่สมบูรณ์ที่สุด

โดย Model นี้ได้มีการตระหนักถึงกลุ่มผู้ใช้งาน (End-User) เข้าไปมีส่วนร่วมในขอบเขตการพัฒนาระบบ โดยให้กลุ่มผู้ใช้งานมีการทวนซ้ำ พิจารณาและยอมรับการพัฒนาระบบในแต่ละลำดับขั้นตอน เพื่อป้องกันไม่ให้เกิดความล่าช้า และต้นทุนในการดำเนินงานเกินกว่าที่คาดการณ์ไว้

Model นี้โดยมากแล้วจะอยู่ในขอบเขตความรับผิดชอบของนักวิเคราะห์ระบบ (Analyst)

ภาพที่ 2-2 Incremental Model



ที่มา: The SDLC and SIX SIGMA an essay on which is which and why (Boggs, 2004)

นักพัฒนาระบบ, (2551) ได้กล่าวข้อดีของ Incremental Model คือ

- ผู้ใช้งานได้ใช้ระบบเร็ว
- ผู้ใช้งานปรับตัวกับระบบใหม่แบบค่อยเป็นค่อยไปได้ ทำให้ไม่รู้สึกต่อต้านระบบใหม่
- ลดความเสี่ยง เนื่องจากแต่ละรอบของการพัฒนาได้นำระบบในรอบก่อนหน้ามาทดสอบร่วมด้วย

นักพัฒนาระบบ, (2551) ได้กล่าวข้อเสียของ Incremental Model คือ

- หากวางแผนการประสานงานไม่ดี อาจทำให้เกิดข้อผิดพลาดได้

3. Spiral Model

ในปี ค.ศ. 1990 Model รูปแบบใหม่นี้ได้เริ่มปรากฏขึ้นจาก Barry W. Boehm ซึ่ง Model นี้ได้ถูกออกแบบเป็นกรณีตัวอย่างแรกที่ใช้เพื่อปรับปรุงในเรื่องของการจัดการความเสี่ยง และการรักษาภาพระบบ และเป็นการปรับกระบวนการทำงานแบบตายตัวจาก Waterfall Model มาสู่การทำงานที่ยืดหยุ่นขึ้น

Model นี้ได้เริ่มต้นด้วยการจำกัดต้นแบบที่ใช้ในการประสานงานในแต่ละขั้นตอน และมีการจัดการความเสี่ยงที่เป็นปัจจัยสำคัญในการพัฒนาระบบ โดยได้ให้ผู้ใช้งานที่เกี่ยวข้องทั้งหมดเข้ามามีส่วนร่วมในการระบุความต้องการในแต่ละวงรอบ แต่ก็จะเป็นการยากในการประเมินต้นทุน เวลา และกำหนดการในแผนการลงทุนของโครงการด้วย

ภักดีวัฒนกุล, (2551) ได้กล่าวข้อดีของ Spiral Model คือ

- มีความยืดหยุ่นมาก เนื่องจาก การวิเคราะห์ ออกแบบ ติดตั้ง และ ทดสอบ ในแต่ละรอบนั้นจะสั้นหรือยาวเท่าใดก็ได้
- ข้ามบางขั้นตอนไปได้ หากขั้นตอนนั้นไม่จำเป็นต้องดำเนินการในบางรอบ
- เหมาะกับระบบที่มีการเปลี่ยนแปลงความต้องการบ่อยครั้ง

ภักดีวัฒนกุล, (2551) ได้กล่าวข้อเสียของ Spiral Model คือ

- มีความเสี่ยงสูง
- ต้องวิเคราะห์ความเสี่ยงทุกรอบ

4. Win-Win Spiral Model

ในปี ค.ศ. 1994 Boehm ได้แนะนำ SDLC ในรูปแบบลำดับขั้น Win-Win Spiral Model ขึ้น แต่เดิมรูปแบบของ Spiral Model นั้นได้ให้ผู้ใช้งานหรือผู้มีส่วนได้ส่วนเสีย (Stakeholder) เข้ามามีส่วนร่วมในการระบุความต้องการตลอดทั้งวงรอบการพัฒนาแบบใด ๆ ก็ตามอาจจะทำให้เกิดความขัดแย้งขึ้นในกลุ่มของผู้ใช้งานหรือผู้มีส่วนได้ส่วนเสียที่มีความต้องการไม่ตรงกันได้

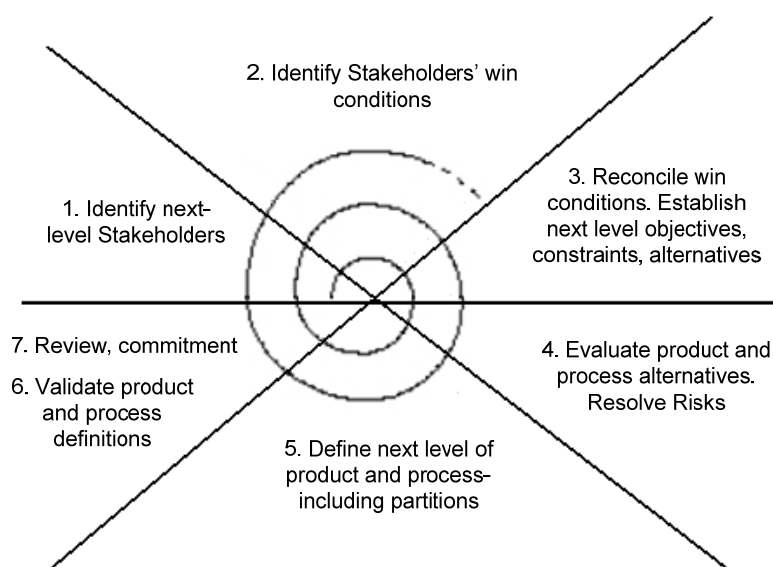
ดังนั้น Win-Win Spiral Model จึงได้ปรับปรุงกระบวนการแบบเดิม โดยเน้นในเรื่องของการกำหนดผู้ที่เกี่ยวข้องในแต่ละขั้นตอนอย่างชัดเจน และแนะนำวิธีการในการแก้ปัญหาเรื่องความขัดแย้งที่เกิดขึ้นในผู้ใช้งานหรือผู้มีส่วนได้ส่วนเสียทั้งหมด ในขณะเดียวกันก็ได้เพิ่มเทคนิคใน

การจัดการความเสี่ยงที่สมบูรณ์ขึ้นอีกด้วย อย่างไรก็ตามในบางกรณีผู้มีส่วนได้ส่วนเสียอาจจะไม่ได้รับการพิจารณาความต้องการเหมือนกับผู้ใช้งานในทุกๆกรณีก็เป็นได้

ผลลัพธ์ที่ได้คือ SDLC ได้เกิดการยอมรับจากผู้ใช้งานและผู้มีส่วนได้ส่วนเสียที่เกี่ยวข้อง รวมถึงมีการจัดการควบคุมความเสี่ยงที่สมบูรณ์ โดยทำให้โครงการสามารถควบคุมด้านการเงินได้อย่างมีประสิทธิภาพ

Model นี้โดยมากแล้วจะอยู่ในขอบเขตความรับผิดชอบของผู้จัดการระบบ (Manager)

ภาพที่ 2-3 Win-Win Spiral Model



ที่มา: The SDLC and SIX SIGMA an essay on which is which and why (Boggs, 2004)

5. V-Model

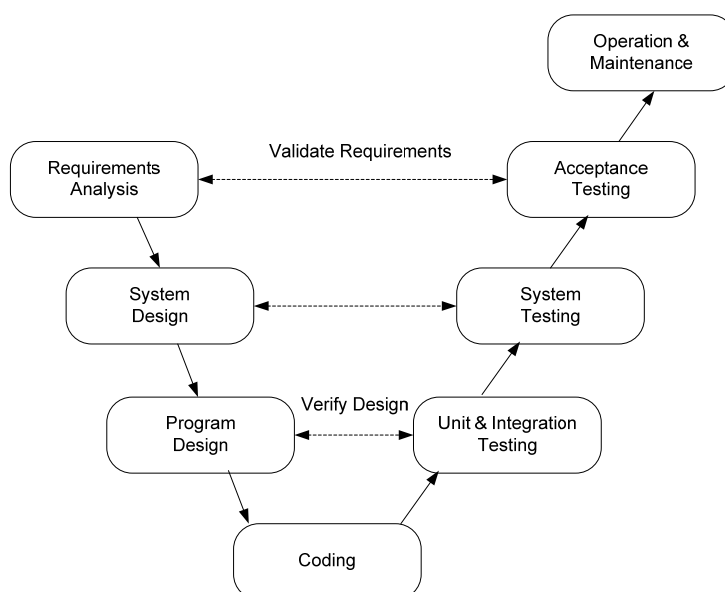
V-Model เป็นทิศทางใหม่สำหรับ SDLC โดยเกิดขึ้นในประเทศเยอรมัน ปัจจุบัน Model นี้ เป็น Model ที่กำลังได้รับความนิยมสำหรับโครงการด้านเทคโนโลยีสารสนเทศ ที่ใช้ในหน่วยทหาร พลเรือน และรัฐบาลกลางของเยอรมัน

Model นี้มุ่งเน้นที่การบริหารโครงการ การพัฒนาซอฟต์แวร์ การรับรองคุณภาพ และการจัดการการกำหนดคุณสมบัติ ด้วยการให้ความสำคัญในเรื่องของการสื่อสารที่มีประสิทธิภาพระหว่างผู้พัฒนาระบบและลูกค้า โดยคาดหวังว่า Model นี้จะทำให้ต้นทุนในการบริหารโครงการ

ลดลงตลอดจนได้รับการรับรองคุณภาพ ซึ่ง V-Model นั้นได้กลายมาเป็นที่ยอมรับโดยสากลและส่งผลให้เกิดมาตรฐานต่างๆตามมา อย่างเช่น ISO/IEC 12207 หรือ ISO 9001

สำหรับ Model แบบใหม่นี้ โครงการ WEIT Model ได้มีการกำหนดขึ้นในช่วงเมษายน ค.ศ. 2005 โดยมุ่งเน้นประเด็นในเรื่องของความยืดหยุ่น การเพิ่มหรือลดความสามารถของการพัฒนาระบบ การปรับปรุง เปลี่ยนแปลง และกฎเกณฑ์ต่างๆของระบบแอปพลิเคชัน

ภาพที่ 2-4 V-Model



ที่มา: The SDLC and SIX SIGMA an essay on which is which and why (Boggs, 2004)

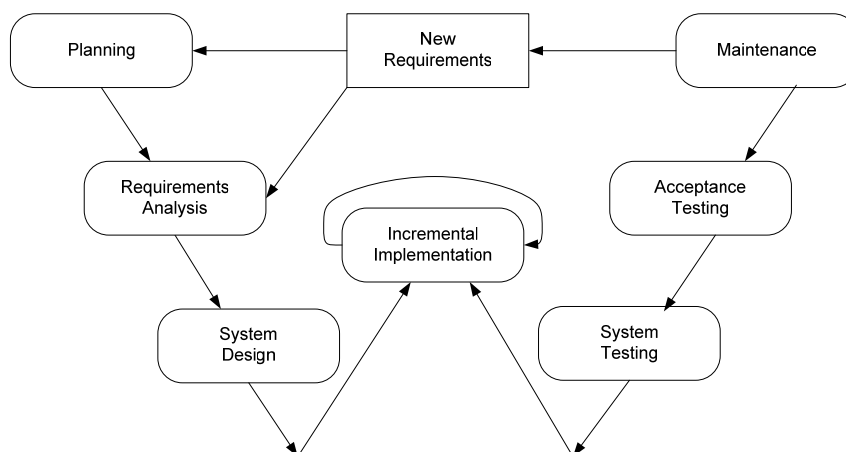
6. W-Model

หนึ่งใน Model ที่เพิ่มขึ้นด้วยรูปแบบที่คล้ายกับรูปแบบเดิมคือ W-Model ของ Fraunhofer Institute of Production Technology โดย Model นี้ได้มุ่งเน้นในช่วงเริ่มต้นของการเขียนโปรแกรม (Coding) และการทดสอบ (Testing) เพื่อให้เกิดการประเมินต้นทุนและการควบคุมความเสี่ยงอย่างมีประสิทธิภาพมากขึ้น ซึ่ง Model นี้ได้ถูกนำมาใช้โดยนักพัฒนาระบบจากบริษัทที่มีชื่อเสียงต่างๆ เช่น Microsoft, Oracle, HP, IBM, และ Texas Instruments เป็นต้น

การให้ผลย้อนกลับ (Feedback) จากผู้ใช้งาน รวมถึงการให้ความสำคัญในเรื่องของการเขียนโปรแกรมและแนวทางปฏิบัติในการเขียนโปรแกรมนั้นเป็นสิ่งสำคัญใน Model นี้

Model นี้โดยมากแล้วจะอยู่ในขอบเขตความรับผิดชอบของผู้พัฒนาซอฟต์แวร์ (Software Developer)

ภาพที่ 2-5 W-Model



ที่มา: The SDLC and SIX SIGMA an essay on which is which and why (Boggs, 2004)

2.1.5 กระบวนการในการพัฒนาระบบสารสนเทศ (SDLC)

กระบวนการในการพัฒนาระบบสารสนเทศ หรือ System development life cycle (SDLC) นั้น โดยปกติแล้วจะมีอยู่หลายระยะขึ้นอยู่กับความเหมาะสมในการนำไปใช้งานในการพัฒนาระบบสารสนเทศของแต่ละองค์กร ซึ่งจากการศึกษาจะกล่าวถึง SDLC ที่นำไปใช้ในโครงการพัฒนาระบบสารสนเทศโดยประกอบไปด้วยขั้นตอนทั้งหมด 8 ระยะด้วยกัน ดังนี้ ((Cervone, 2007), (ภักดีวัฒนะกุล, 2551), (house.gov, 1999))

1. การสำรวจเบื้องต้น (Preliminary Investigation)
2. การวิเคราะห์ปัญหา (Problem Analysis)
3. การวิเคราะห์ความต้องการ (Requirements Analysis)
4. การวิเคราะห์การตัดสินใจ (Decision Analysis)
5. การออกแบบระบบ (Design)
6. การสร้างหรือพัฒนาระบบ (Construction)
7. การติดตั้งระบบ (Implementation)
8. การปฏิบัติงานและการให้ความช่วยเหลือ (Operation and Support)

ระยะที่ 1 การสำรวจเบื้องต้น (Preliminary Investigation Phase)

การสำรวจเบื้องต้นเป็นระยะเริ่มต้นโครงการ ซึ่งจะต้องสามารถอธิบายได้ว่า “โครงการนี้ควรที่จะมีแนวปฏิบัติอย่างไร” ในระยะนี้ทีมงานในโครงการจะต้องทำการตรวจสอบ ค้นหาความเป็นไปได้ โดยมักจะเกิดขึ้นจากการประชุมอย่างเป็นทางการของระดับผู้บริหาร ซึ่งจะต้องสามารถกำหนดผลลัพธ์ที่ต้องการได้รับจากโครงการ รวมถึงสามารถระบุข้อจำกัดได้ในระยะนี้

หลังจากกำหนดการของโครงการได้ถูกกำหนดขึ้นมาแล้ว ระยะนี้จะต้องตัดสินใจว่าจะดำเนินการต่อหรือไม่ (Go/No-go) โดยผู้สนับสนุนโครงการ

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- จัดทำเอกสารเพื่อกำหนดภาพรวมโครงการ
- กำหนดรายละเอียดที่ต้องจัดทำภายในโครงการ หรือ WBS (Work Breakdown Structure)
- วิเคราะห์ต้นทุนของโครงการ
- วางแผนเพื่อกำหนดรายละเอียดในโครงการ (ระยะเวลา เงินทุน งานที่ต้องทำ เป็นต้น)
- ทบทวนแผนกำหนดการโครงการ

โดยในระยะนี้ ทางเลือกที่เป็นไปได้ มีดังต่อไปนี้

- ยอมรับในกำหนดการของโครงการ ที่มีการกำหนดเงื่อนไขในขอบเขตของโครงการ งบประมาณ และระยะเวลาที่ใช้
- พิจารณาทั้งในเรื่องของการลดต้นทุนและระยะเวลาที่ใช้
- ปฏิเสธโครงการ

เมื่อการสำรวจเบื้องต้นในระยะเริ่มต้นประสบความสำเร็จแล้ว โครงการก็จะเข้าสู่ระยะที่สอง

ระยะที่ 2 การวิเคราะห์ปัญหา (Problem Analysis)

ระยะนี้จะเป็นการค้นหาสาเหตุของปัญหา รวมถึงตรวจสอบรายละเอียดต่างๆภายในสภาพแวดล้อมปัจจุบันของกระบวนการทางธุรกิจที่เกิดขึ้นในโครงการ องค์ประกอบหลักของการค้นหา คือ การค้นหาสาเหตุของปัญหาและรวบรวมปัญหาที่เกิดขึ้นของระบบงานเดิม ที่เป็นกระบวนการทางธุรกิจในปัจจุบัน ซึ่งเป็นไปได้ที่อาจจะมีความต้องการเพิ่มมากขึ้นในระยะนี้ เนื่องจากระบบเดิมนั้นการใช้งานยังไม่ครอบคลุม เป็นขั้นตอนการรวบรวมข้อมูลและข้อเท็จจริงของระบบโดยใช้วิธีการเก็บรวบรวมข้อมูล เพื่อนำข้อมูลที่ได้มาทำการวิเคราะห์หาปัญหาที่เกิดขึ้นและนำไปสู่แนวทางในการแก้ไขปัญหาดังกล่าว

ข้อควรตระหนักถึงในเบื้องต้นของระยะนี้คือ การระบุผลประโยชน์ที่ได้รับ วิเคราะห์ความคุ้มค่าจากการลงทุนพัฒนาระบบใหม่ ในช่วงสุดท้ายของระยะนี้คือ การรายงานถึงรายละเอียดของการพัฒนาระบบตามวัตถุประสงค์ที่กำหนดขึ้น ซึ่งจะต้องสามารถอธิบายได้ว่า “อะไรคือปัญหาที่ระบบใหม่สามารถจะเข้ามาแก้ไขให้ดีขึ้นได้”

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- สัมภาษณ์ลูกค้าหรือผู้ใช้งานและวิเคราะห์ความต้องการ
- วิเคราะห์ความเสี่ยงของโครงการ
- วิเคราะห์ต้นทุนและผลประโยชน์ที่ได้รับ
- การประเมินเทคโนโลยีที่เกี่ยวข้องในการดำเนินโครงการ (Hardware and Software)

เหมือนกับในระยะแรก ระยะนี้จะต้องตัดสินใจว่าจะดำเนินการต่อหรือไม่ (Go/No-go) โดยผู้สนับสนุนโครงการ

โดยในระยะนี้ ทางเลือกที่เป็นไปได้ มีดังต่อไปนี้

- อนุมัติโครงการให้ดำเนินต่อไป
- ลดหรือขยายขอบเขตของโครงการ ในกรณีที่มีการปรับปรุงเปลี่ยนแปลงตามความเหมาะสม ซึ่งจะต้องอยู่ภายใต้งบประมาณ กำหนดการ ที่กำหนดไว้
- ยกเลิกโครงการ

เมื่อการวิเคราะห์ปัญหาประสบความสำเร็จแล้ว โครงการก็จะเข้าสู่ระยะที่สาม

ระยะที่ 3 การวิเคราะห์ความต้องการ (Requirement Analysis)

ในระยะนี้ควรที่จะให้ความสำคัญเป็นอย่างมาก โดยทีมงานในโครงการจะต้องตระหนักถึงว่า “ระบบใหม่ต้องการที่จะทำอะไร” เป็นการวิเคราะห์ความต้องการของระบบใหม่จากปัญหาที่พบในการทำงานของระบบเดิม ระยะการวิเคราะห์ความต้องการเป็นระยะที่สำคัญที่สุดระยะหนึ่ง เนื่องจากในระยะนี้จะประกอบไปด้วย ข้อมูล กระบวนการ และการติดต่อประสานงานจากความต้องการที่กำหนดขึ้น ซึ่งการติดต่อประสานงานนั้นเป็นเรื่องสำคัญมาก เพราะหากการดำเนินงานขาดความราบรื่น ขาดประสิทธิภาพ อาจจะทำให้เกิดต้นทุนที่สูงได้ในภายหลัง

การกำหนดความต้องการและการจัดลำดับในการพัฒนาระบบใหม่นั้น จะต้องมีการเก็บรวบรวมข้อมูล อาจจะใช้วิธีการโดยการ สัมภาษณ์ สอบถามถึงรายละเอียด และจัดประชุมกับผู้ที่เกี่ยวข้องในการพัฒนาระบบใหม่ทั้งหมด เพื่อทบทวนข้อกำหนดความต้องการของระบบให้ถูกต้องและสอดคล้องกันทั้งหมด

โดยในระยะนี้จะมีการใช้แบบจำลองระบบ (System Model) เพื่อเป็นเครื่องมือในการนำเสนอข้อกำหนดความต้องการของระบบ การจำลองขั้นตอนการทำงานของระบบ (Process Model) เพื่ออธิบายขั้นตอนการทำงานของระบบ ทำให้ทราบการทำงานของระบบได้ชัดเจนขึ้น รวมถึงจัดเตรียมเอกสารความต้องการ (Requirement Documentation) และจัดลำดับความสำคัญของงานเหล่านั้นเพื่อเตรียมพัฒนาระบบต่อไป

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- การกำหนดขอบเขตหน้าที่ความรับผิดชอบในการปฏิบัติงาน
- สัมภาษณ์เก็บข้อมูลผู้ใช้งานหรือลูกค้า
- กำหนดกระบวนการดำเนินงานของผู้ใช้งาน
- จัดทำเอกสารความต้องการของผู้ใช้งาน

เมื่อการวิเคราะห์ความต้องการประสบความสำเร็จแล้ว โครงการก็จะเข้าสู่ระยะที่สี่

ระยะที่ 4 การวิเคราะห์การตัดสินใจ (Decision Analysis)

ในระยะนี้มีหลากหลายทางเลือกจากการวิเคราะห์ความเป็นไปได้ และแนะนำทางเลือกที่เหมาะสมที่สุดในการนำไปแก้ไขปัญหา ซึ่งในกรณีของระยะก่อนหน้านี้ ทีมงานจะต้องดำเนินงานกับหลากหลายผู้ที่เกี่ยวข้องในโครงการที่สามารถจะวิเคราะห์รูปแบบต่างๆของความเป็นไปได้ที่เกิดขึ้น โดยความเป็นไปได้ที่ควรคำนึงถึงมีดังต่อไปนี้

- ด้านเทคนิค (Technical) เทคนิคที่จะนำมาใช้ของระบบใหม่นั้นสามารถนำมาใช้เพื่อช่วยแก้ปัญหาภายในสภาพแวดล้อมนี้ได้หรือไม่ เทคโนโลยีที่มีอยู่เดิมนั้นสามารถนำมาปรับใช้กับระบบใหม่ได้หรือไม่ บุคลากรในองค์กรมีความเชี่ยวชาญกับเทคโนโลยีนี้มากเพียงพอหรือไม่
- ด้านการปฏิบัติงาน (Operational) ระบบใหม่จะสามารถเติมเต็มความต้องการของผู้ใช้งานได้หรือไม่ จะต้องใช้ทรัพยากรอย่างมีประสิทธิภาพ การใช้งานของระบบนั้นเรียนรู้และใช้งานง่ายหรือไม่ รวมถึงผู้ใช้งานจะสามารถปรับตัวสำหรับการเปลี่ยนแปลงได้อย่างไร
- ด้านเศรษฐศาสตร์ (Economic) เป็นแนวทางที่ช่วยทำให้เกิดการบริหารและวิเคราะห์ต้นทุนได้อย่างมีประสิทธิภาพหรือไม่ พิจารณาถึงผลตอบแทนที่คาดว่าจะได้รับจากโครงการ
- ด้านระยะเวลาดำเนินงาน (Time frame) ระบบใหม่สามารถที่จะออกแบบและติดตั้งระบบ รวมถึงแนวทางในการแก้ปัญหาภายในระยะเวลาที่กำหนดไว้ได้หรือไม่
- ด้านความเสี่ยง (Risk) กิจกรรมอะไรที่อาจจะทำให้เกิดความเสี่ยงที่มีผลกระทบในระดับรุนแรง และมีประเด็นอะไรที่จะทำให้เกิดปัญหาได้ เพื่อเตรียมการวางแผนและป้องกันความเสี่ยงที่จะเกิดขึ้น

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- การวิเคราะห์ทางเลือกในการจัดซื้อ Hardware และ Software

การดำเนินงานในระยะนี้ หนึ่งในแนวทางที่ดีที่สุด ควรจะมีการนำโครงร่างระบบงาน (System Proposal) มาใช้เป็นแนวทางในการแก้ปัญหา โดยโครงร่างระบบงานนั้น จะทำให้เกิดการตัดสินใจว่าจะดำเนินการต่อหรือไม่ (Go/No-go)

โดยในระยะนี้ ทางเลือกที่เป็นไปได้ มีดังต่อไปนี้

- อนุมัติโครงการและให้เงินสนับสนุนจากโครงร่างระบบงาน
- อนุมัติโครงการและให้เงินสนับสนุนจากการนำเสนอทางเลือกในการแก้ปัญหา
- การลดขอบเขตจากโครงร่างระบบงาน ได้รับการอนุมัติและให้เงินสนับสนุน
- ยกเลิกโครงการ

เมื่อการวิเคราะห์การตัดสินใจประสบความสำเร็จแล้ว โครงการก็จะเข้าสู่ระยะที่ห้า

ระยะที่ 5 การออกแบบระบบ (Design)

การออกแบบระบบใหม่นั้นอยู่บนพื้นฐานของโครงร่างระบบงาน โดยระยะนี้จะเป็นขั้นตอนการแปลงความต้องการทางธุรกิจออกมาในรูปแบบของการออกแบบรายละเอียดของการพัฒนาระบบ ซึ่งในระยะนี้จะเป็นการนำแนวทางด้านเทคโนโลยีเข้ามาใช้ในการแก้ปัญหาทางธุรกิจที่เกิดขึ้น โดย ผู้ขาย (Vender) และ ผู้ชำนาญด้านเทคโนโลยี (IT Specialist)

กิจกรรมที่เกิดขึ้นในระยะการออกแบบ คือ การวางแผนการรวมกันกับระบบการใช้งานอื่นๆ และออกแบบระบบใหม่ที่กระบวนการทำงานเดิมยังไม่ครอบคลุม เพื่อเพิ่มประสิทธิภาพในการดำเนินงานให้ดียิ่งขึ้น

โดยการออกแบบระบบนั้นควรจะพิจารณาดังต่อไปนี้

- การออกแบบแบบฟอร์ม รายงานและส่วนประสานกับผู้ใช้ (Forms/Reports/User Interface Design) ซึ่งเป็นแหล่งของข้อมูลนำเข้าและข้อมูลที่ออกจากระบบหรือได้จากการทำงานของระบบ
- การออกแบบฐานข้อมูล (Database Design)
- การออกแบบสถาปัตยกรรมแอปพลิเคชัน (Application Architecture Design) เป็นการออกแบบแอปพลิเคชันที่ได้จากการวิเคราะห์และออกแบบให้สามารถเข้ากันได้กับสภาพแวดล้อมที่กำหนดไว้

และในระยะนี้ยังคงต้องทำการตัดสินใจว่าจะดำเนินการต่อหรือไม่ (Go/No-go) ซึ่งการที่จะล้มเลิกโครงการในระยะนี้ค่อนข้างจะเป็นไปได้ยาก ยกเว้นแต่การพัฒนาระบบนั้นมีการใช้ต้นทุนเกินงบประมาณหรือเกินกำหนดการไปอย่างมาก

กิจกรรมในระยนี้ มีดังต่อไปนี้

- จัดทำเอกสารออกแบบระบบ (System Design)
- กำหนดความต้องการด้านเทคนิคของระบบ
- ระบุความปลอดภัยของระบบและการควบคุมการดำเนินงานภายใน
- วิเคราะห์รายละเอียดของต้นทุน ผลประโยชน์ที่ได้รับจากการดำเนินโครงการ
- จัดทำรายงานการประเมินเทคโนโลยีที่ใช้
- เปรียบเทียบระบบเดิมและระบบใหม่

โดยในระยนี้ ทางเลือกที่เป็นไปได้ มีดังต่อไปนี้

- ระบบเริ่มเข้าสู่ระยะการสร้างหรือการพัฒนา
- การลดขอบเขตโครงการลงในเรื่องของ ต้นทุน เวลา หรือทั้งสองอย่าง
- การเพิ่มกำหนดการ เพื่อจัดเตรียมแนวทางที่สมบูรณ์ในหลายๆแนวทาง
- โครงการถูกยกเลิกระหว่างการดำเนินโครงการ เนื่องจาก งบประมาณ และกำหนดการที่สูงเกินกว่ากำหนดไว้มากจนเกินยอมรับได้

ระยะที่ 6 การสร้างหรือพัฒนาระบบ (Construction)

การพัฒนาระบบ นั้นเป็นขั้นตอนในการพัฒนาหรือการเขียนโปรแกรม ออกแบบรายละเอียดและทดสอบระบบให้ตรงตามความต้องการที่กำหนดไว้ นอกจากนั้นการประสานงานระหว่างระบบใหม่และระบบเดิมก็เป็นเรื่องที่ต้องตระหนักถึงในการพัฒนาระบบด้วย ในระยะนี้โดยส่วนมากแล้วจะมีการใช้เวลาในการพัฒนานานกว่าระยะอื่นๆ

ระหว่างการพัฒนาในระยนี้ จะมีการทวนซ้ำเป็นรอบของกระบวนการพัฒนา รวมถึงทำการทดสอบโปรแกรมเพื่อค้นหาข้อผิดพลาด และดำเนินการแก้ไขให้ถูกต้องและสมบูรณ์ที่สุด ก่อนที่จะนำโปรแกรมนั้นมาติดตั้งเพื่อใช้งานระบบ มีการสาธิตและกำหนดรายละเอียด เพื่อให้แน่ใจได้ว่าระบบที่พัฒนานั้นตรงกับความต้องการและความคาดหวังของผู้สนับสนุนโครงการ

ซึ่งกระบวนการกำหนดรายละเอียดนั้นจะต้องควบคุมไม่ให้เกิดการเปลี่ยนแปลงหรือเพิ่มเติมรายละเอียดในช่วงระหว่างการพัฒนาเกินไป เพราะอาจจะทำให้เกิดการยืดระยะในการพัฒนาระบบออกไปจนเกินกำหนดการที่กำหนดไว้ได้

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- จัดทำรายงานต้นแบบ/ทดลองระบบ (Prototype)
- รับรองและยืนยันความถูกต้อง แผนการทดสอบระบบ (System Testing)
- จัดทำรายงานการรับรองความสมบูรณ์ของระบบ
- จัดทำรายงานการประเมินความปลอดภัย

เมื่อระบบพัฒนาตรงตามความต้องการเสร็จสมบูรณ์ตลอดจนได้ทำการทดสอบเรียบร้อยแล้ว การพัฒนาระบบก็จะเข้าสู่ระยะถัดไป

ระยะที่ 7 การติดตั้งระบบ (Implementation)

ระยะนี้เป็นระยะที่ทำให้ระบบเกิดการเปลี่ยนแปลงครั้งสำคัญโดยอาจเกิดความเสี่ยงจากการต่อต้านการเปลี่ยนแปลงของผู้ใช้งานจากการแทนที่ระบบเก่าสู่ระบบใหม่ ดังนั้นองค์ประกอบที่มีความสำคัญในการติดตั้งคือกระบวนการถ่ายโอนอย่างรอบคอบและราบรื่นด้วยการเลือกประเภทของการติดตั้งระบบที่เหมาะสมที่สุด ซึ่งจะช่วยให้ผู้ใช้งานสามารถเตรียมตัวกับปัญหาที่อาจจะเกิดขึ้นในช่วงเริ่มต้นได้

ก่อนที่จะมีการใช้งานจริงบนระบบใหม่นั้น จะต้องมีการเตรียมการฝึกอบรม การใช้งานและขั้นตอนการทำงานของระบบใหม่แก่ผู้ใช้งานที่เกี่ยวข้องทุกคน มีการเตรียมความพร้อมในการช่วยเหลือเมื่อเกิดปัญหา รวมถึงจะต้องมีรายละเอียดการจัดทำเอกสาร ทั้งเอกสารของระบบและเอกสารของผู้ใช้งาน เพื่อใช้อ้างอิงในระหว่างการใช้งานระบบใหม่เมื่อเกิดปัญหานั้น มีการจัดทำฐานข้อมูลส่วนกลางในการเก็บข้อมูลทั้งหมดเพื่อง่ายต่อการจัดการ และสุดท้ายจะต้องมีการทดสอบระบบเพื่อความถูกต้องในการใช้งาน โดยในระยะนี้ทีมงานควรจะได้รับ ผลตอบรับ (Feedback) จากผู้ใช้งาน เพื่อรับทราบถึงปัญหาหรือประเด็นใหม่ๆที่อาจจะเกิดขึ้นจากระบบใหม่ได้

มี 2 วิธีการในการเปลี่ยนแปลงการใช้งานเข้าสู่ระบบใหม่ วิธีแรก คือ สลับทันที (ก่อนเวลากำหนดการ) จากระบบเก่าไปสู่ระบบใหม่ โดยระบบเก่าจะถูกยกเลิกและนำระบบใหม่มาใช้แทนที่ ข้อดีของวิธีนี้คือ หลีกเลี่ยงการบำรุงรักษาพร้อมๆกันทั้งระบบเก่าและระบบใหม่ แต่ข้อเสียก็คือ หากระบบใหม่ไม่สมบูรณ์ การย้อนกลับหรือกู้คืนกลับมาใช้ระบบเดิมนั้น อาจะเกิดข้อผิดพลาดบางประการขึ้นได้ อีกทั้งค่อนข้างจะเป็นไปได้ยากในการที่กลับมาใช้ระบบเดิมได้อย่างราบรื่น วิธีที่สอง คือ การใช้งานดำเนินงานแบบคู่ขนานอยู่บนทั้งระบบเก่าและระบบใหม่

จนกว่าจะมีการยอมรับและวางใจในระบบใหม่ จึงจะนำมาแทนที่ระบบเดิม ซึ่งการบำรุงรักษานั้นก็จะต้องเกิดขึ้นตลอดจนกระทั่งเหลือระบบเดียวที่สมบูรณ์ รวมถึงวิธีนี้อาจจะทำให้เกิดการซ้ำซ้อนกันของข้อมูลในระบบทั้ง 2 ระบบได้อีกด้วย

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- กำหนดแผนการติดตั้ง
- รับรองและยืนยันความถูกต้อง จัดทำแผนการทดสอบระบบ
- จัดทำเอกสารระบบ (System Document)
- ปรับปรุงเอกสารการควบคุมการเชื่อมต่อระบบ (Interface system)
- จัดทำรายงานวิเคราะห์การทดสอบระบบ
- การยอมรับจากผู้ใช้งานรวมถึงจัดทำเอกสารรับรองระบบงาน
- กำหนดแผนการเปลี่ยนแปลงระบบ
- กำหนดแผนการฝึกอบรม
- จัดทำคู่มือผู้ใช้งาน

เมื่อการติดตั้งระบบเสร็จสมบูรณ์แล้ว ก็ถือได้ว่าการพัฒนาตามกระบวนการ SDLC ก็ได้รับความสมบูรณ์ไปด้วย และการพัฒนาระบบก็จะเข้าสู่ระยะถัดไป

ระยะที่ 8 การปฏิบัติงานและการให้ความช่วยเหลือ (Operation and Support)

แม้ว่าการพัฒนาระบบจะดำเนินการเสร็จสิ้นแล้ว แต่ระบบใหม่ก็ยังคงต้องการความเตรียมพร้อมในการให้ความช่วยเหลือหลังการติดตั้งระบบ โดยระยะนี้ควรที่จะมีทีมงานที่ทำหน้าที่คอยให้ความช่วยเหลือเมื่อมีความผิดพลาดหรือปัญหาเกิดขึ้นระหว่างการใช้งานหรือมีข้อสงสัยในระบบใหม่

กิจกรรมในระยะนี้ มีดังต่อไปนี้

- จัดทำเอกสารการวัดผลการปฏิบัติงาน
- จัดทำเอกสารการตั้งค่าระบบ (Configuration)
- จัดทำเอกสารการจัดการการเปลี่ยนแปลง

- ทบทวนการปรับปรุงเทคโนโลยี

ท้ายที่สุดแล้ว ระบบใหม่ก็จะกลายมาเป็นระบบเก่า ดังนั้นการเปลี่ยนแปลงความต้องการก็อาจจะเกิดขึ้นได้ตลอดเวลา การบำรุงรักษาระบบ (System Maintenance) จะเป็นการรักษาการทำงานของระบบใหม่ที่ได้ติดตั้งและใช้งานไปแล้ว และจะต้องสามารถปรับปรุงเปลี่ยนแปลงขั้นตอนหรือการทำงานของระบบได้ตามความต้องการของผู้ใช้งาน เพื่อให้ระบบยังคงมีประสิทธิภาพและสามารถตอบสนองต่อความต้องการในระหว่างการใช้งานได้ รวมถึงจะต้องมีการคิดและวางแผนในการนำเทคโนโลยีเข้ามาช่วยบำรุงรักษาระบบให้คงไว้ซึ่งประสิทธิภาพต่อไปนั้นก็เป็นสิ่งที่ต้องควรตระหนักถึง จากนั้นก็จะเข้าสู่การเริ่มต้นวางแผนกำหนดโครงการขึ้นมาใหม่ การเริ่มต้นของ SDLC ก็จะกลับมาอีกครั้ง เพื่อให้สอดคล้องต่อการพัฒนาระบบในอนาคตต่อไป

2.2 โครงการพัฒนาระบบสารสนเทศ (Information System Development Project)

2.2.1 คำจำกัดความและลักษณะของโครงการพัฒนาระบบสารสนเทศ

โครงการพัฒนาระบบสารสนเทศ คือ กระบวนการพัฒนาระบบสารสนเทศ ที่ได้รับรู้ถึงกิจกรรมต่างๆ ในแต่ละระยะของการพัฒนาระบบ โดยรวมถึงกลุ่มผู้ใช้งานที่เกี่ยวข้องและวิธีการในการพัฒนาระบบ เพื่อแนะนำและเป็นแนวทางให้นักพัฒนาระบบปฏิบัติตามเพื่อลดปัญหาต่างๆ ที่อาจจะเกิดขึ้นจากกระบวนการพัฒนาระบบ และเพื่อให้โครงการบรรลุผลตามวัตถุประสงค์ที่กำหนดไว้อย่างมีประสิทธิภาพ (Katainen and Nahar, 2008)

ระบบสารสนเทศเป็นสิ่งสำคัญที่ต้องได้รับการจัดการอย่างมีประสิทธิภาพ เพื่อประโยชน์ในการดำเนินงานและตัดสินใจทางธุรกิจ ดังนั้นระบบสารสนเทศจึงเป็นกลไกหนึ่งในการนำเทคโนโลยีสารสนเทศมาประยุกต์ใช้กับเทคโนโลยีด้านคอมพิวเตอร์ โดยมีส่วนประกอบที่สำคัญ 5 ส่วน คือ ฮาร์ดแวร์ ซอฟต์แวร์ ข้อมูล บุคลากร และขั้นตอนการทำงาน (ฝ่ายผลิตหนังสือตำราวิชาการคอมพิวเตอร์, 2551)

การพัฒนาระบบสารสนเทศนั้นเป็นส่วนสำคัญในองค์กรที่มีอิทธิพลต่อการพัฒนาเทคโนโลยีสารสนเทศ การใช้งาน และแอปพลิเคชันต่างๆ ในองค์กร ที่เกี่ยวข้องในด้านธุรกิจหรือ

ด้านต่างๆในองค์กร วิธีการหรือกระบวนการที่สามารถที่จะถูกใช้ในการพัฒนาในระบบสารสนเทศ ซึ่งนักพัฒนาระบบส่วนมากมักจะมีการนำกระบวนการ อย่างเช่น System Development Life cycle หรือ SDLC ที่ซึ่งเป็นกระบวนการของการพัฒนาอย่างเป็นขั้นตอนและพัฒนาอย่างเป็นลำดับขั้น มาใช้ในการพัฒนาระบบสารสนเทศภายในองค์กร (วิกิพีเดีย, 2009)

หน้าที่หลักของโครงการระบบสารสนเทศคือ การอำนวยความสะดวกในการได้มาซึ่งผลลัพธ์ที่สัมฤทธิ์ผลของโครงการ เนื่องจากลักษณะของโครงการระบบสารสนเทศนั้นมีความแตกต่างกับโครงการทั่วไป ทั้งในเรื่องของทรัพยากร วิธีการและเงื่อนไขต่างๆ (Tulnon and Jean2, 2005) โดยโครงการระบบสารสนเทศนั้นมักจะยากต่อการจัดการ เนื่องจากผลลัพธ์ที่ได้จากการพัฒนาระบบสารสนเทศนั้นจะไม่ตรงกับความต้องการของผู้ใช้งาน รวมถึงโครงการจะต้องประสบผลสำเร็จ ภายใต้ข้อจำกัด 4 ข้อ (Quadruple Constraints) ตามที่ Pinto และ Slevin ได้กล่าวไว้คือ โครงการที่ดีนั้นจะต้องประสบความสำเร็จภายใต้ข้อจำกัดในเรื่องของงบประมาณ กำหนดการ คุณภาพ และความพึงพอใจของผู้ใช้งานหรือลูกค้า (Tulnon and Jean2, 2005)

2.2.2 ระยะของโครงการพัฒนาระบบสารสนเทศ

การพัฒนาระบบสารสนเทศนั้นมีการกำหนดกระบวนการอย่างชัดเจนเพื่อช่วยในการลดปัญหาที่เกิดขึ้นได้จากสาเหตุการพัฒนาระบบ การอธิบายในแต่ละระยะมีดังต่อไปนี้ (Katainen and Nahar, 2008)

1. ระยะเริ่มต้น (Start-up Phase)

ระยะเริ่มต้นของโครงการเป็นสิ่งสำคัญ ถ้าในระยะเริ่มต้นนั้นดี บริษัทจะสามารถทำกำไรจากการพัฒนาโครงการและสามารถตอบสนองต่อความคาดหวังของลูกค้าได้ ถ้าในระยะเริ่มต้นมีการดำเนินการอย่างไม่เป็นลำดับขั้นตอน อาจส่งผลต่อการพัฒนาระบบที่จะทำให้เกิดความล่าช้า ใช้เวลานาน และมีการใช้ทรัพยากรมากเกินไปจนกว่าที่จัดสรรไว้ ทำให้ทรัพยากรไม่เพียงพอ ก่อนที่โครงการจะเสร็จสมบูรณ์

โดยในระยะเริ่มต้นจะมีดังต่อไปนี้

การศึกษาความเป็นไปได้ (Feasibility Study)

การศึกษาความเป็นไปได้อาจจะมีการศึกษาก่อนที่จะเริ่มต้นพัฒนาโครงการ โดยการศึกษาความเป็นไปได้นั้นได้รวมถึง เป้าหมายของการใช้งานโปรแกรม ปัญหาที่เป็นไปได้งบประมาณ การเริ่มต้นของกำหนดการ และการสิ้นสุดการพัฒนาระบบ โดยจะต้องสนับสนุนถึงมุมมองทั้งในด้านเทคนิคและด้านการเงิน ซึ่งการศึกษาความเป็นไปได้นั้นต้องการให้เกิดการสื่อสารและการประสานงานระหว่างหน่วยงานต่างๆอย่างมีประสิทธิภาพ เพื่อหลีกเลี่ยงการเกิดความเข้าใจผิดที่อาจจะเกิดขึ้นได้ โดยข้อมูลและการใช้เครื่องมือร่วมกันเป็นสิ่งสำคัญอย่างหลีกเลี่ยงไม่ได้ในระยะนี้ วิธีการสื่อสารที่ดีและชัดเจนที่สุดคือ การสื่อสารแบบเผชิญหน้า (Face to Face) หรือเครื่องมืออื่นที่มีความเหมาะสม อย่างเช่น อุปกรณ์ทางโสต (Audio) หรือ การประชุมผ่านจอภาพ (Video Conference) หรือ จดหมายอิเล็กทรอนิกส์ (E-mail) เป็นต้น

การกำหนดลักษณะโครงการ (Setting the Project)

การกำหนดลักษณะโครงการ ควรที่จะอธิบายได้ว่า โครงการนี้จัดตั้งขึ้นมาเพื่อวัตถุประสงค์อะไร มีกำหนดการเป็นอย่างไร และใครบ้างเป็นผู้มีส่วนรับผิดชอบในโครงการ ซึ่งในระยะนี้ จะเป็นการเจรจาในระดับการจัดการ เช่น ระหว่างผู้จัดการโครงการและคณะกรรมการของโครงการ สิ่งที่จะทำให้โครงการประสบความสำเร็จได้ ผู้จัดการโครงการและลูกค้าจะต้องมีความเข้าใจตรงกัน และการสื่อสารระหว่างผู้จัดการโครงการและคณะกรรมการของโครงการควรที่จะเป็นการสื่อสารแบบเผชิญหน้า (Face to Face) มากที่สุดเท่าที่จะเป็นไปได้ หรืออาจจะใช้เครื่องมืออื่นที่มีความเหมาะสม อย่างเช่น อุปกรณ์ทางโสต (Audio) หรือ การประชุมผ่านจอภาพ (Video Conference) หรือ จดหมายอิเล็กทรอนิกส์ (E-mail) เป็นต้น ในระยะนี้ยังไม่ได้กล่าวถึงการวางแผนการจัดการความเสี่ยงที่จะเกิดขึ้นในโครงการ

การออกแบบโครงการ (Design the Project)

เพื่อให้โครงการประสบความสำเร็จ การวางแผนการออกแบบจึงเป็นสิ่งสำคัญของการพัฒนาระบบ ถ้าการพัฒนาและกระบวนการจัดการไม่เหมาะสม ก็จะเป็นการเพิ่มความเสี่ยงให้โครงการไม่ประสบผลสำเร็จได้ การวางแผนโครงการนั้นรวมถึงการจัดการความเสี่ยง การสื่อสาร การทดสอบ และการวางแผนการติดตั้ง

การวางแผนโครงการควรที่จะอธิบายได้ว่า โครงการนี้จัดทำขึ้นเพื่ออะไร และทำอย่างไรจึงจะบรรลุผลที่ต้องการได้ ความรู้อะไรที่ต้องการนำมาใช้ในการพัฒนาโครงการ การยอม

ให้เกิดการกระจายงานและแบ่งแยกงาน โดยกำหนดผู้รับผิดชอบสำหรับงานนั้นๆ และนำวิธีในการแบ่งแยกรายละเอียดงานมาใช้ เช่น Work Break down Structure (WBS) และ Critical Path Analysis (CPA) เป็นต้น การแบ่งแยกรายละเอียดของงานสามารถทำได้โดยการใช้เครื่องมือทางเทคนิค อย่างเช่น Microsoft Project และ Lotus notes ถ้าองค์กรมีลักษณะโครงสร้างเป็นแบบ matrix อาจจะทำให้ยากต่อการกำหนดความรับผิดชอบใน WBS

การตัดสินใจในการวางแผนนั้นถูกจำกัดโดยทรัพยากรที่มีอยู่ ความแตกต่างของระดับความหลากหลายของเทคโนโลยี และโครงสร้างพื้นฐาน (Infrastructure) โดยจำนวนสมาชิกในทีม อาจจะเป็นส่วนสำคัญที่ทำให้เกิดความแตกต่าง และอาจจะทำให้เกิดปัญหาในการพัฒนาระบบสารสนเทศได้ สมาชิกในโครงการควรที่จะสามารถเข้าถึงข้อมูลต่างๆในโครงการได้ แต่อย่างไรก็ตามควรที่จะมีการกำหนดสิทธิในการเข้าถึงข้อมูลได้ตามความเหมาะสมด้วย

การวางแผนสำหรับการจัดการความเสี่ยงเป็นสิ่งสำคัญมากในการพัฒนาระบบสารสนเทศ เมื่อโครงการได้ถูกกำหนดขึ้น หน้าที่ความรับผิดชอบจะต้องมีการระบุอย่างชัดเจน รวมทั้งควรที่จะต้องมีการอบรมเพื่อให้เกิดความรู้และประสบการณ์ในหน้าที่ความรับผิดชอบนั้นๆ การวางแผนการทดสอบระบบควรที่จะรวมอยู่ในกระบวนการพัฒนาระบบสารสนเทศด้วย รวมถึงการร่วมแบ่งปันความรู้กันภายในโครงการ ก็จะทำให้โครงการบรรลุผลสำเร็จได้อย่างมีประสิทธิภาพ ในหลายๆกรณี การพัฒนาระบบสารสนเทศอาจจะพัฒนาโดยจ้างหน่วยงานภายนอก (Outsourcing) ก็เป็นไปได้

การกำหนดความชัดเจนของโครงการ (Defining)

ในขณะนี้ ได้กำหนดคุณสมบัติหลักๆในการพัฒนาระบบสารสนเทศ ระเบียบนี้ได้อธิบายถึงการวางแผนทางด้านเทคนิคที่อยู่บนพื้นฐานของการพัฒนาระบบ ซึ่งระยะนี้จะประสบความสำเร็จได้โดยการ แจ้งความต้องการให้ผู้ที่เกี่ยวข้องรับทราบอย่างทั่วถึง เพื่อช่วยต่อการประสานงานระหว่างสมาชิกในทีม เพราะหากมีการเปลี่ยนแปลงเกิดขึ้นในโครงสร้างการพัฒนา ระบบจะทำให้เกิดต้นทุนที่สูงและใช้เวลานาน โดยความรับผิดชอบหลักๆนั้นจะเป็นหน้าที่ของนักออกแบบระบบสารสนเทศและกลุ่มผู้ใช้งาน (End-User)

ขอบเขตของการพัฒนาระบบในระยะนี้คือ กิจกรรมไหนที่ต้องทำและไม่ต้องทำ ซึ่งระยะนี้การสื่อสารควรที่จะมีความชัดเจนมากๆ เพื่อไม่ให้เกิดความผิดพลาดในการติดต่อประสานงานระหว่างหน่วยงานต่างๆ ถ้าเป็นไปได้ควรที่จะใช้เครื่องมือที่ช่วยทั้งในการเก็บข้อมูลและสนับสนุนการสื่อสารในเวลาเดียวกัน เช่น Lotus notes เป็นต้น ในระยะนี้จะมีการสร้าง

เอกสารสำคัญต่างๆมากมายและเอกสารเหล่านี้ควรที่จะสะดวกต่อการเข้าถึงและง่ายต่อการจัดการ

2. ระยะการจัดสร้างหรือพัฒนา (Construction Phase)

ในระยะการจัดสร้างหรือพัฒนาระบบสารสนเทศ คือ การเขียนโปรแกรม (Coding) โดยระยะนี้จะต้องสามารถอธิบายได้ว่า ทีมงานพัฒนากำลังทำอะไร ต้องทำอะไร และทำอย่างไร การพัฒนาโครงการระบบสารสนเทศมี 2 องค์ประกอบในการสื่อสารคือ การสื่อสารแบบเป็นทางการและไม่เป็นทางการ ซึ่งการสื่อสารถึงความต้องการในการพัฒนาระบบนั้นควรจะเป็นการสื่อสารที่ติดต่อถึงกันได้มากที่สุดเท่าที่จะเป็นไปได้

โดยในระยะการจัดสร้างและพัฒนาจะมีดังต่อไปนี้

การวางแผน การเขียนโปรแกรม การทดสอบ และการติดตั้ง (Planning, Coding, Testing and Implementation)

การวางแผนควรรวมถึงการอธิบายทางด้านเทคนิคในการนำมาใช้ในการพัฒนาระบบ โดยในระยะนี้ กำหนดการ (Schedule) ได้ถูกกำหนดขึ้นบนพื้นฐานจากการวางแผนการออกแบบและการจัดสรรทรัพยากรในโครงการ รวมถึงการเลือกวิธีการสำหรับการเขียนโปรแกรม

การวางแผนจะต้องมีจุดประสงค์ที่อธิบายได้ว่า อะไรคือสิ่งสำคัญในการพัฒนาโครงการแทนที่จะวางแผนไปอย่างไร้วัตถุประสงค์ นอกจากนี้มักจะเป็นเรื่องปกติสำหรับการออกแบบการเปลี่ยนแปลงในระหว่างการพัฒนา ดังนั้นการเปลี่ยนแปลงต่างๆที่อาจเกิดขึ้นในการออกแบบการวางแผนนั้น จะต้องมีการควบคุมกำหนดเอกสารความต้องการไว้ด้วย

เครื่องมือที่ใช้ในการสื่อสารมีความสำคัญมากในระยะนี้ เนื่องจากทีมงานอาจจะมีข้อสงสัยหรืออาจจะมีการร่วมแบ่งปันข้อมูลระหว่างกัน การวางแผนของโครงสร้างไม่ใช่เรื่องที่ย่ายและมักจะเกี่ยวข้องกับหลากหลายผู้ใช้งานที่เกี่ยวข้อง เครื่องมือ CASE (Computer Aided Software Engineering) อาจจะถูกนำมาใช้อย่างไม่เป็นทางการได้เพื่อช่วยสนับสนุนในเกิดการสร้างโครงสร้างการพัฒนาระบบและการทำงานร่วมกันขึ้นอย่างรวดเร็ว โดยทั่วไปแล้ว นักออกแบบไม่จำเป็นต้องสื่อสารกับลูกค้าโดยตรง แต่มีความจำเป็นในการสื่อสารร่วมกับผู้ที่มีส่วนเกี่ยวข้องในระยะการวางแผนและการเขียนโปรแกรมมากกว่า

ในการเขียนโปรแกรมนั้นได้ถูกพัฒนาขึ้นตั้งแต่การออกแบบในระยะเริ่มแรก โดยการเขียนโปรแกรมนั้นควรที่จะมีเครื่องมือที่ช่วยในการควบคุมฉบับ (Version) ของการเขียนโปรแกรม โดยปกติแล้วจะเป็นเครื่องมือแบบ Single-user การเขียนโปรแกรมนั้นควรที่จะเป็นไปตาม

รายละเอียดความต้องการที่กำหนดไว้ ซึ่งปัญหาหลักๆของการเขียนโปรแกรมในการพัฒนาระบบคือ นักพัฒนาใช้รูปแบบในการเขียนโปรแกรมที่แตกต่างกันตามความถนัดของแต่ละบุคคล ทำให้ยากต่อการจัดการหรือการดำเนินงานต่อของนักพัฒนาในภายหลัง ซึ่งการเขียนโปรแกรมที่ดีนั้นควรที่จะมีการประชุมถึงความคืบหน้าในแต่ละจุดที่สำคัญของการพัฒนา การจัดให้มีการประชุมประจำสัปดาห์ (Weekly meeting) และการทบทวน ตรวจสอบ code ที่เขียนอย่างสม่ำเสมอ

เครื่องมือ CASE ได้ถูกนำมาใช้ในการทดสอบระบบ เพื่อให้แน่ใจได้ว่าการเขียนโปรแกรมนั้นได้พัฒนาตรงตามแผนงานที่วางไว้ และเมื่อหากมีข้อผิดพลาดเกิดขึ้น จะได้ทำการแก้ไขได้ในทันที การทดสอบระบบนั้นจะต้องทำตลอดทั้งการพัฒนาโครงการ เพื่อยืนยันความถูกต้องของการใช้งาน ซึ่งเป็นสิ่งสำคัญที่จะทำให้เกิดความแน่ใจได้ว่าโปรแกรมนั้นพร้อมสำหรับการใช้งาน ยกตัวอย่างเช่น ถ้าโปรแกรมพร้อมใช้งานแล้วแต่ยังไม่มีผู้ใช้งานรับทราบ ก็อาจจะทำให้เกิดการปรับเปลี่ยนความต้องการในภายหลังได้ แผนการทดสอบระบบที่ดีควรที่จะระบุถึงวิธีการสำหรับการทดสอบตลอดจนมีการรายงานผลการทดสอบให้ทราบ

เมื่อมีการออกแบบและทดสอบระบบเป็นอย่างดีแล้ว การนำเสนอระบบสู่ผู้ใช้งานควรที่จะมีการเตรียมความพร้อมในการดำเนินงานและส่งมอบได้ทันเวลา ซึ่งในระบายนี้นี้ต้องการการติดต่อสื่อสารที่รวดเร็วในการทำงานร่วมกัน เนื่องจากหากมีปัญหาเกิดขึ้น อาจจะมีเวลาไม่มากนักในการแก้ไขปัญหาเหล่านั้น

อย่างไรก็ตาม การจัดเตรียมเอกสารที่ดีนั้นควรที่จะมีความสอดคล้องในการอธิบายตามแผนการของโครงการ

การยอมรับ (Acceptance)

การยอมรับเกิดขึ้นเมื่อโครงการสิ้นสุดอย่างสมบูรณ์ การพัฒนาระบบสารสนเทศนั้นได้ส่งมอบให้กับผู้ใช้งานหรือลูกค้า ซึ่งในระบายนี้นี้มีความสำคัญในการส่งมอบระบบตรงตามเป้าหมายที่กำหนดไว้ โดยระยะนี้จะเกี่ยวข้องกับหลายๆทีมงานที่เกี่ยวข้องกับโครงการ เช่น ลูกค้ากลุ่มผู้ใช้งาน ทีมงานในการพัฒนาระบบ เป็นต้น เพื่อเห็นพ้องในการพัฒนาระบบตรงตามแผนงานที่วางไว้ หากลูกค้าหรือผู้ใช้งานได้รับความพึงพอใจ จะทำให้โครงการประสบความสำเร็จอย่างสมบูรณ์

มีหลายกรณีที่โครงการมักจะเสร็จสิ้นล่าช้า ลูกค้าหรือผู้ใช้งานไม่พึงพอใจกับระบบที่ได้ ดังนั้นในช่วงเริ่มแรกของโครงการ ควรจะมีการเขียนทุกๆความต้องการลงในรายละเอียดอย่างเป็นรูปแบบ

3. ระยะสิ้นสุดโครงการ (Ending Phase)

ในระยะสุดท้ายนี้ การพัฒนาโครงการระบบสารสนเทศได้สิ้นสุดลง โดยอาจจะประสบความสำเร็จหรือไม่ประสบความสำเร็จก็เป็นได้ ในระยะนี้อาจจะมีปัญหาในการยอมรับของลูกค้าหรือผู้ใช้งานได้ ถ้ารายละเอียดของระบบไม่ตรงกับความต้องการหรืออาจจะยังคงมีข้อสงสัยที่ยังไม่ชัดเจนจากการใช้งาน

โดยในระยะการสิ้นสุดโครงการจะมีดังต่อไปนี้

การบำรุงรักษา (Maintenance)

หลังจากมีการส่งมอบระบบสู่ลูกค้าหรือผู้ใช้งานแล้ว จะเป็นการยืนยันได้ว่าระบบสารสนเทศนั้นจะสามารถดำเนินงานต่อไปได้ตามแผนที่กำหนดไว้ อย่างไรก็ตามระบบสารสนเทศจะต้องได้รับการทดสอบ เพื่อตรวจสอบข้อบกพร่องที่อาจเกิดขึ้นได้ จากการใช้งานที่ไม่ครอบคลุม ซึ่งประเด็นต่างๆที่เกิดขึ้นจะต้องรายงานผลกลับเพื่อนำข้อผิดพลาดที่ถูกรับรู้เหล่านี้ไปแก้ไขและพัฒนาโครงการต่อไป ในระยะนี้การบริหารโครงการจะมุ่งเน้นในเรื่องของประสิทธิภาพที่ได้จากการพัฒนาระบบและการแก้ปัญหาแก่ผู้ใช้งานได้อย่างรวดเร็ว

การสิ้นสุดโครงการ (Ending the Project)

คณะกรรมการได้พิสูจน์และอนุมัติว่าโครงการนั้นตรงกับความต้องการทั้งหมด จึงจะทำการตัดสินใจปิดโครงการ ภายในระยะนี้ควรที่จะมีการสื่อสารแบบเผชิญหน้า (Face to Face) เพื่อตรวจสอบความต้องการจากระบบที่พัฒนาขึ้น เมื่อโครงการได้รับการอนุมัติในการสิ้นสุดโครงการ ทีมงานต่างๆที่เกี่ยวข้องก็จะสิ้นสุดหน้าที่ในการพัฒนาโครงการ

2.3 การพัฒนาระบบสารสนเทศ (Information System Development)

2.3.1 คำจำกัดความและลักษณะของการพัฒนาระบบสารสนเทศ

ระบบสารสนเทศเป็นสิ่งสำคัญที่ต้องได้รับการจัดการอย่างมีประสิทธิภาพ เพื่อประโยชน์ในการดำเนินงานและตัดสินใจทางธุรกิจ ดังนั้นระบบสารสนเทศจึงเป็นกลไกหนึ่งในการนำเทคโนโลยีสารสนเทศมาประยุกต์ใช้กับเทคโนโลยีด้านคอมพิวเตอร์ โดยมีส่วนประกอบที่สำคัญ

5 ส่วน คือ ฮาร์ดแวร์ ซอฟต์แวร์ ข้อมูล บุคลากร และขั้นตอนการทำงาน (ฝ่ายผลิตหนังสือตำราวิชาการคอมพิวเตอร์, 2551)

ระบบสารสนเทศ หมายถึง ชุดขององค์ประกอบที่ทำหน้าที่รวบรวม ประมวลผล จัดเก็บ และแจกจ่ายสารสนเทศ เพื่อช่วยการตัดสินใจ และการควบคุมในองค์กร ในการทำงานของระบบสารสนเทศประกอบไปด้วยกิจกรรม 3 อย่าง คือ การนำข้อมูลเข้าสู่ระบบ (Input) การประมวลผล (Processing) และ การนำเสนอผลลัพธ์ (Output) ระบบสารสนเทศอาจจะมีการสะท้อนกลับ (Feedback) เพื่อการประเมินและปรับปรุงข้อมูลนำเข้า ระบบสารสนเทศอาจจะเป็นระบบที่ประมวลด้วยมือ (Manual) หรือระบบที่ใช้คอมพิวเตอร์ก็ได้ (Computer-based information system –CBIS) (Laudon and Laudon, 2001)

การพัฒนาระบบสารสนเทศนั้นเป็นส่วนสำคัญขององค์กรที่มีอิทธิพลต่อการพัฒนาเทคโนโลยีสารสนเทศ การใช้งาน และแอปพลิเคชันต่างๆในองค์กร ที่เกี่ยวข้องในด้านธุรกิจหรือด้านต่างๆในองค์กร วิธีการหรือกระบวนการที่สามารถที่จะถูกใช้ในการพัฒนาในระบบสารสนเทศ ซึ่งนักพัฒนาระบบส่วนมากมักจะมีการนำกระบวนการ อย่างเช่น System Development Life cycle หรือ SDLC ที่ซึ่งเป็นกระบวนการของการพัฒนาอย่างเป็นขั้นตอนและพัฒนาอย่างเป็นลำดับขั้น มาใช้ในการพัฒนาระบบสารสนเทศภายในองค์กร (วิกิพีเดีย, 2009)

2.3.2 องค์ประกอบหลักในการพัฒนาระบบสารสนเทศ

องค์ประกอบสำคัญที่ควรจะต้องนำมาพิจารณาในการพัฒนาระบบสารสนเทศ ประกอบไปด้วย 5 องค์ประกอบสำคัญ ดังนี้ (Yuval, 1996)

1. เป้าหมาย (Goal)

ระบบจะต้องมีเป้าหมายที่ชัดเจน และเป็นเป้าหมายที่สามารถทำให้สำเร็จได้ ทั้งยังต้องสามารถช่วยทำให้องค์กรบรรลุวัตถุประสงค์ทางธุรกิจได้ เป้าหมายจะต้องรวมถึง การวิเคราะห์ปัญหา การ วิเคราะห์ต้นทุนและผลประโยชน์ที่ได้รับ ซึ่งจะต้องคำนึงถึงโครงสร้างองค์กรและการวางแผนกลยุทธ์ด้านระบบสารสนเทศ

2. แอปพลิเคชัน (Application)

แอปพลิเคชันเป็นสิ่งสำคัญของระบบสารสนเทศ ซึ่งเป็นหน้าที่หลักของระบบสารสนเทศ โดยจะรวมถึงหน้าที่การทำงาน (Functions) ข้อมูล (Data) ระบบย่อย (Subsystems) การดำเนินการ (Transactions) รายงาน (Reports) ไฟล์ (file) และอื่นๆ

3. เทคโนโลยี (Technology)

ประกอบไปด้วย องค์ประกอบทั้งหมดที่ต้องการนำมาใช้ในการพัฒนาระบบสารสนเทศ การดำเนินงานและการบำรุงรักษาระบบสารสนเทศ ทั้งที่มีอยู่แล้วในองค์กรและที่จัดหาจากแหล่งภายนอก จากอุปกรณ์ฮาร์ดแวร์และซอฟต์แวร์

4. การตระหนักถึง (Realization)

การร่วมกัน (Join together) ขององค์ประกอบต่างๆ ทั้งแอปพลิเคชันหลักและเทคโนโลยี ทั้งในระดับกลางหรือการทำงานขั้นสุดท้าย ซึ่งมักจะอยู่ในโครงการสำคัญ หรือแผนงานที่ใหญ่และเป็นแผนงานที่มีความเฉพาะเจาะจง การตระหนักถึงยังรวมถึงในเรื่องของเอกสารผู้ใช้งาน (User documentation) การวางแผนสำหรับการปฏิบัติงานและการบำรุงรักษา การบริการและการทดสอบระบบ

5. ต้นทุน (Cost)

พิจารณาจากการพัฒนาระบบสารสนเทศ ต้นทุนยังรวมถึงค่าใช้จ่ายทั้งหมดที่มีความเกี่ยวข้องกับระบบ การติดตั้ง (Installation) การปฏิบัติงานอย่างต่อเนื่อง (Running) การกำหนดค่า (Configurations) และอื่นๆ

2.3.3 ประเภทของระบบสารสนเทศ

หากพิจารณาจำแนกระบบสารสนเทศตามการสนับสนุนระดับการทำงานในองค์กร ประเภทของระบบสารสนเทศที่สนับสนุนการทำงานของปฏิบัติงาน/ผู้บริหารระดับต่างๆมี ดังนี้ (blog.eduzones, 2009)

1. ระบบประมวลผลรายการ (Transaction Processing Systems: TPS)

เป็นระบบที่ทำหน้าที่ในการปฏิบัติงานประจำ ทำการบันทึกจัดเก็บ ประมวลผลรายการที่เกิดขึ้นในแต่ละวัน โดยใช้ระบบคอมพิวเตอร์ทำงานแทนการทำงานแบบ Manual ทั้งนี้เพื่อที่จะทำการสรุปข้อมูลในการสร้างเป็นสารสนเทศ ระบบประมวลผลรายการนี้ ส่วนใหญ่จะเป็นระบบที่เชื่อมโยงกิจการกับลูกค้า ในระบบต้องสร้างฐานข้อมูลที่จำเป็น ระบบนี้มักจัดทำเพื่อสนองความต้องการของผู้บริหารระดับต้นเป็นส่วนใหญ่เพื่อให้สามารถปฏิบัติงานประจำได้ ผลลัพธ์ของระบบนี้ มักจะอยู่ในรูปของ รายงานที่มีรายละเอียด รายงานผลเบื้องต้น

2. ระบบสำนักงานอัตโนมัติ (Office Automation Systems: OAS)

เป็นระบบที่สนับสนุนงานในสำนักงาน หรืองานธุรการของหน่วยงาน ระบบจะประสานการทำงานของบุคลากรรวมทั้งกับบุคคลภายนอก หรือหน่วยงานอื่น ระบบนี้จะเกี่ยวข้องกับการจัดการเอกสาร โดยการใช้ซอฟต์แวร์ด้านการพิมพ์ การติดต่อผ่านระบบไปรษณีย์อิเล็กทรอนิกส์ เป็นต้นผลลัพธ์ของระบบนี้ มักอยู่ในรูปของเอกสาร กำหนดการ สิ่งพิมพ์

3. ระบบงานสร้างความรู้ (Knowledge Work Systems: KWS)

เป็นระบบที่ช่วยสนับสนุนบุคลากรที่ทำงานด้านการสร้างความรู้เพื่อพัฒนาการคิดค้น สร้างผลิตภัณฑ์ใหม่ๆ บริการใหม่ ความรู้ใหม่เพื่อนำไปใช้ประโยชน์ในหน่วยงาน หน่วยงานต้องนำเทคโนโลยีสารสนเทศเข้ามาสนับสนุนให้การพัฒนาเกิดขึ้นได้โดยสะดวก สามารถแข่งขันได้ในด้านเวลา คุณภาพ และราคา ระบบต้องอาศัยแบบจำลองที่สร้างขึ้นตลอดจนการทดลองการผลิตหรือดำเนินการ ก่อนที่จะนำเข้ามาดำเนินการจริงในธุรกิจ ผลลัพธ์ของระบบนี้ มักอยู่ในรูปของ สิ่งประดิษฐ์ ตัวแบบ (Model) รูปแบบ เป็นต้น

4. ระบบสารสนเทศเพื่อการจัดการ (Management Information Systems: MIS)

เป็นระบบสารสนเทศสำหรับผู้ปฏิบัติงานระดับกลาง ใช้ในการวางแผน การบริหารจัดการ และการควบคุม ระบบจะเชื่อมโยงข้อมูลที่มีอยู่ในระบบประมวลผลรายการเข้าด้วยกันเพื่อประมวลและสร้างสารสนเทศที่เหมาะสมและจำเป็นต่อการบริหารงาน ตัวอย่าง เช่น ระบบบริหารงานบุคลากร ผลลัพธ์ของระบบนี้ มักอยู่ในรูปของรายงานสรุป รายงานของสิ่งผิดปกติ

5. สนับสนุนการตัดสินใจ (Decision Support Systems: DSS)

เป็นระบบที่ช่วยผู้บริหารในการตัดสินใจสำหรับปัญหา หรือที่มีโครงสร้างหรือขั้นตอนในการหาคำตอบที่แน่นอนเพียงบางส่วน ข้อมูลที่ใช้ต้องอาศัยทั้งข้อมูลภายในกิจการและภายนอกกิจการประกอบกัน ระบบยังต้องสามารถเสนอทางเลือกให้ผู้บริหารพิจารณา เพื่อเลือกทางเลือกที่เหมาะสมที่สุดสำหรับสถานการณ์นั้น หลักการของระบบ สร้างขึ้นจากแนวคิดของการใช้คอมพิวเตอร์ช่วยการตัดสินใจ โดยให้ผู้ใช้ได้ตอบโดยตรงกับระบบ ทำให้สามารถวิเคราะห์ปรับเปลี่ยนเงื่อนไขและกระบวนการพิจารณาได้ โดยอาศัยประสบการณ์ และ ความสามารถของผู้บริหารเอง ผู้บริหารอาจกำหนดเงื่อนไขและทำการเปลี่ยนแปลงเงื่อนไขต่างๆ ไปจนกระทั่งพบสถานการณ์ที่เหมาะสมที่สุด แล้วใช้เป็นสารสนเทศที่ช่วยตัดสินใจ รูปแบบของผลลัพธ์ อาจจะอยู่

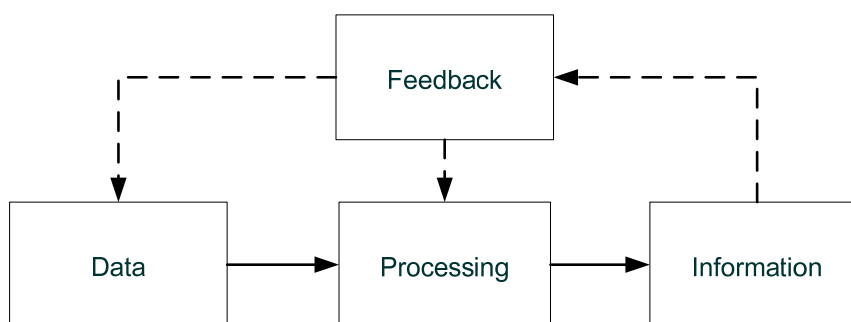
ในรูปของ รายงานเฉพาะกิจ รายงานการวิเคราะห์เพื่อตัดสินใจ การทำนาย หรือ พยากรณ์ เหตุการณ์

6. ระบบสารสนเทศสำหรับผู้บริหารระดับสูง (Executive Information System: EIS)

เป็นระบบที่สร้างสารสนเทศเชิงกลยุทธ์สำหรับผู้บริหารระดับสูง ซึ่งทำหน้าที่กำหนด แผนระยะยาวและเป้าหมายของกิจการ สารสนเทศสำหรับผู้บริหารระดับสูงนี้จำเป็นต้องอาศัย ข้อมูลภายนอกกิจการเป็นอย่างมาก ยิ่งในยุคปัจจุบันที่เป็นยุค Globalization ข้อมูลระดับโลก แนวโน้มระดับสากลเป็นข้อมูลที่สำคัญสำหรับการแข่งขันของธุรกิจ ผลลัพธ์ของระบบนี้ มักอยู่ใน รูปของการพยากรณ์/การคาดการณ์

ถึงแม้ว่าระบบสารสนเทศจะมีหลายประเภท แต่องค์ประกอบที่จำเป็นของระบบ สารสนเทศทุกประเภท ก็จะต้องมีการนำองค์ประกอบต่างๆของเทคโนโลยีสารสนเทศเข้ามาใช้ในการ รวบรวม ประมวลผล และจัดเก็บข้อมูล เพื่อสร้างเป็นผลลัพธ์คือสารสนเทศที่องค์กรต้องการ ระบบสารสนเทศจึงประกอบไปด้วยองค์ประกอบต่างๆที่ต้องทำงานร่วมกันเพื่อวัตถุประสงค์ เดียวกัน คือ การนำข้อมูลเข้าสู่ระบบ การประมวลผลข้อมูล และการแสดงผลของข้อมูลเป็น สารสนเทศที่องค์กรธุรกิจต้องการนั่นเอง (ภักดีวัฒนะกุล, 2551) จำลองภาพการทำงานของระบบ สารสนเทศ ดังรูป

ภาพที่ 2-6 แสดงกระบวนการเปลี่ยนข้อมูลให้เป็นสารสนเทศ



ที่มา: การวิเคราะห์และออกแบบระบบ (Systems Analysis and Design) (ภักดีวัฒนะกุล, 2551)

จากภาพกระบวนการเปลี่ยนข้อมูลให้เป็นสารสนเทศ สามารถอธิบายได้ดังนี้

- Input คือ ข้อมูลหรือข้อเท็จจริงต่างๆที่นำสู่ระบบเพื่อประมวลผล
- Processing คือ การเปลี่ยนแปลงหรือแปรสภาพข้อมูล ให้เป็นสารสนเทศที่ต้องการ ด้วยขั้นตอนหรือวิธีการต่างๆ
- Information (Output) คือ สารสนเทศที่ได้จากการประมวลผลข้อมูล
- Feedback คือ ข้อมูลสะท้อนกลับจากผู้ใช้สารสนเทศ โดยอยู่ในรูปของข้อเสนอแนะ หรือข้อผิดพลาดที่เกิดขึ้นเป็นประโยชน์ต่อการปรับปรุงระบบให้ดีขึ้น

2.4 งานวิจัยที่เกี่ยวข้อง

Boggs, (2004) ได้กล่าวถึง SDLC และ Six Sigma โดยได้กล่าวถึงจุดประสงค์และข้อแตกต่างของการนำ SDLC และ Six Sigma ไปใช้ในการพัฒนาระบบ โดยบทความได้กล่าวถึงประวัติความเป็นมาของ SDLC และลักษณะ Model ต่างๆของ SDLC ที่เกิดขึ้นตามความเหมาะสมในการพัฒนาโครงการระบบสารสนเทศ โดยแต่ละ Model ก็มีรูปแบบกระบวนการพัฒนา และข้อดี ข้อเสียแตกต่างกันไป

บทความได้กล่าวถึง Model ของ SDLC ว่ามีทั้งหมด 6 รูปแบบ คือ

- Waterfall Model
- Incremental Model
- Spiral Model
- Win-Win Spiral Model
- V-Model
- W-Model

อย่างไรก็ตาม โดยทั่วไปนิยมผสมผสานทุกรูปแบบเข้าด้วยกันตามความเหมาะสม และความคล่องตัวในการนำไปใช้เพื่อพัฒนาระบบให้มีประสิทธิภาพ นอกจากนั้นบทความได้กล่าวถึงลักษณะของ Six Sigma กระบวนการในการนำ Six Sigma ไปใช้ในการพัฒนาระบบ

บทความยังกล่าวถึง SDLC ว่าเป็นกระบวนการที่ใช้ในการพัฒนาระบบสารสนเทศที่ได้รับความนิยมมากกว่าครึ่งศตวรรษ ในขณะที่ Six Sigma เพิ่งเริ่มปรากฏขึ้นเมื่อไม่นานมานี้ ซึ่ง

จากการวิเคราะห์ของนักพัฒนาระบบที่ผ่านมาพบว่า SDLC นั้นนำไปใช้งานได้ดีกว่าและมีกระบวนการที่มีประสิทธิภาพมากกว่า โดย SDLC นั้นสามารถที่จะควบคุมกำหนดการ และการดำเนินกิจกรรมต่างๆที่มีผลกระทบต่อการจัดการโครงการพัฒนาระบบสารสนเทศได้อย่างมีประสิทธิภาพ และอีกทั้ง Six Sigma นั้นก็เป็นกระบวนการย่อย ที่อยู่ภายใต้ขอบเขตกระบวนการของ SDLC อีกด้วย

จากผลงานวิจัยของ Dr. Roy พบว่า SDLC เป็นเครื่องมือที่มีการนำมาใช้ได้ดีกว่าเครื่องทศวรรษและเป็นกระบวนการที่มีประสิทธิภาพในการนำมาใช้ในการพัฒนาระบบสารสนเทศ

Cervone, (2007) ได้ศึกษาถึงการนำกระบวนการ SDLC มาพัฒนาในโครงการระบบสารสนเทศของห้องสมุดดิจิทัล เพื่อช่วยแก้ปัญหาที่เกิดขึ้นจากการเปลี่ยนแปลงความต้องการที่เกิดขึ้นตลอดทั้งโครงการ และความคลาดเคลื่อนของความต้องการจากการพัฒนาระบบ เพื่อให้การบริหารโครงการพัฒนาระบบสารสนเทศมีความสมบูรณ์และตรงตามความต้องการของผู้ใช้งาน

โดยในบทความได้กำหนดระยะในการพัฒนา SDLC ออกเป็น 8 ระยะด้วยกัน ดังนี้

1. การสำรวจเบื้องต้น (Preliminary Investigation)
2. การวิเคราะห์ปัญหา (Problem Analysis)
3. การวิเคราะห์ความต้องการ (Requirements Analysis)
4. การวิเคราะห์การตัดสินใจ (Decision Analysis)
5. การออกแบบระบบ (Design)
6. การสร้างหรือพัฒนาระบบ (Construction)
7. การติดตั้งระบบ (Implementation)
8. การปฏิบัติงานและการให้ความช่วยเหลือ (Operation and Support)

จากผลงานวิจัยของ H. Frank Cervone กล่าวว่า SDLC นั้นคือมาตรฐานที่ดีมากพอที่จะนำมาใช้ในการพัฒนาระบบและเพื่อใช้ในการติดตาม ตรวจสอบความก้าวหน้าของโครงการให้สำเร็จลุล่วงได้อย่างมีประสิทธิภาพ และท้ายที่สุด ระบบใหม่ก็จะกลายมาเป็นระบบเก่า ดังนั้นการเปลี่ยนแปลงความต้องการก็อาจจะเกิดขึ้นได้ตลอดเวลา และก็จะทำให้เกิดการเริ่มต้นวางแผน กำหนดโครงการขึ้นมาใหม่ การเริ่มต้นของ SDLC ก็จะกลับมาอีกครั้ง เพื่อให้สอดคล้องต่อการพัฒนาระบบในอนาคตต่อไป และพบว่า SDLC คือมาตรฐานที่ดีมากพอที่จะนำมาใช้ใน

การพัฒนาระบบสารสนเทศและทำให้สามารถติดตาม รับทราบถึงความคืบหน้าของโครงการได้เป็นอย่างดี

Nasution and Weistroffer, (2009) ได้ศึกษาถึงโครงการพัฒนาระบบสารสนเทศ ซึ่งควรจะต้องให้ความสำคัญในการบริหารทั้งต้นทุน เวลา และทรัพยากรอย่างมีประสิทธิภาพ โดยระบบที่ได้จากการพัฒนาระบบสารสนเทศควรจะเป็นระบบที่ตรงตามเป้าหมายที่โครงการกำหนดไว้ และต้องอยู่ภายในงบประมาณ และเวลาที่กำหนดไว้ด้วย ซึ่งปัจจุบันนี้ผลลัพธ์ที่ได้จากโครงการพัฒนาระบบสารสนเทศมักจะใช้เวลาเกินกว่ากำหนด อีกทั้งยังไม่ตรงตามความต้องการผู้ใช้งานอีกด้วย

โดยบทความได้กล่าวถึงโครงการพัฒนาระบบสารสนเทศที่ดีควรที่จะมีองค์ประกอบดังนี้

1. การส่งมอบระบบควรตรงตามความต้องการของผู้ใช้งาน/ลูกค้า
2. โครงการควรดำเนินงานภายใต้งบประมาณที่กำหนดไว้
3. โครงการควรดำเนินงานภายใต้เวลาที่กำหนดไว้
4. โครงการควรดำเนินงานเสร็จอย่างครบถ้วนสมบูรณ์
5. การส่งมอบระบบควรที่จะมีคุณภาพและผ่านการทดสอบอย่างสมบูรณ์
6. ระบบควรที่จะง่ายต่อการใช้งาน

จากบทความได้มีการนำ SDLC มาใช้ในบริษัท Ricoh ซึ่งเป็นบริษัทเกี่ยวกับ กล้องและอุปกรณ์เครื่องใช้อิเล็กทรอนิกส์ (Cameras and electronic office equipment) โดย SDLC นั้นถูกนำมาใช้งานร่วมกับวิธีการต่างๆ เช่น ต้นแบบ (Prototyping) และ การวิเคราะห์แบบ (object-oriented analysis) และการออกแบบ (Design) โดยได้เสนอการนำ Waterfall Model มาใช้ควบคู่กับการนำต้นแบบ (Prototyping) เพื่อแก้ปัญหาที่เกิดขึ้นจาก Waterfall Model และเพื่อให้เกิดความเหมาะสมในการพัฒนาระบบสารสนเทศที่มีประสิทธิภาพ ซึ่ง SDLC ที่นำมาใช้ประกอบไปด้วย 8 ระยะ ดังนี้

- Feasibility study
- Systems investigation
- Analysis
- Design

- Development
- Implementation
- Testing
- Maintenance

จากผลงานวิจัยของ M. Faisal Fariduddin Attar Nasution และ H. Roland Weistroffer พบว่า การพัฒนาระบบสารสนเทศตามกรอบ SDLC จะส่งผลให้ผู้ใช้งานเกิดความพึงพอใจจากการส่งมอบระบบที่ตรงตามความต้องการ และช่วยลดความล้มเหลวที่เกิดขึ้นจากการบริหารโครงการได้ (ระยะเวลา, ต้นทุน, คุณภาพ)

Kushniruk, (2002) ได้ศึกษาถึงการตรวจสอบหน้าที่สำหรับการประเมินในการออกแบบการพัฒนาระบบสารสนเทศประเภท Health care กรอบการวิจัยได้แสดงถึงการพิจารณาการประเมินของกระบวนการพัฒนาซอฟต์แวร์ โดยมีการนำ SDLC แบบดั้งเดิม (Waterfall Model) มาใช้ในการพัฒนาระบบสารสนเทศเต็มรูปแบบเพื่อใช้สำหรับการประเมินระบบ Health care ซึ่งเป็นระบบที่ซับซ้อนและยากต่อการวิเคราะห์

บทความกล่าวถึงลักษณะในแต่ละระยะของ SDLC โดยระบุถึงกิจกรรมในการชีวิตในแต่ละระยะ โดย SDLC ที่นำมาใช้ประกอบไปด้วย 5 ระยะดังนี้

- Planning
- Analysis
- Design
- Implementation
- Maintenance and support

จากผลงานวิจัยของ Andre Kushniruk พบว่า การนำ SDLC มาใช้นั้นการนำ SDLC แบบดั้งเดิมมาใช้ควบคู่กับต้นแบบ (Prototyping) นั้นจะทำให้เกิดประสิทธิภาพตลอดทั้งวงจรในการพัฒนาระบบสารสนเทศ และ SDLC เหมาะสมกับกระบวนการพัฒนาระบบสารสนเทศที่มีความซับซ้อน ยากต่อการวิเคราะห์ รองรับการเปลี่ยนแปลงที่เกิดขึ้นในแต่ละระยะของโครงการ รวมถึงกระบวนการ SDLC ทำให้ผลลัพธ์ที่ได้จากการพัฒนาระบบสารสนเทศตรงตามความต้องการของผู้ใช้งานอีกด้วย

Tsai et al., (2009) ได้ศึกษาถึงการนำ SDLC มาใช้ในโครงการพัฒนาระบบสารสนเทศของรัฐบาลสหรัฐอเมริกา (E-Government) ซึ่งเป็นระบบการใช้งานบนอินเทอร์เน็ตและ Web-base เทคโนโลยีเพื่อให้บริการจากภาครัฐออนไลน์

โดยขั้นตอนการวางแผนของ E-Government จะต้องมีการกำหนดวัตถุประสงค์ข้อจำกัด และขอบเขต ของระบบที่นำมาใช้ โดยวัตถุประสงค์ของระบบนั้นเป็นแบบไม่ต้องการกำไรแต่จะต้องให้บริการข้อมูลที่มีคุณภาพสูงและให้บริการสื่อสารตรงตามเป้าหมาย

จากบทความกล่าวว่าการนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศนั้นเหมาะสมกับการทำงานของรัฐบาล ซึ่งเป็นแบบลำดับขั้น เป็นตรรกวิทยา และอยู่บนพื้นฐานของกฎระเบียบ แต่ก็ยังอนุญาตให้เกิดความยืดหยุ่นได้ โดยขึ้นอยู่กับความซับซ้อนของโครงการ การส่งมอบ และการจัดสรรงบประมาณ

โดย SDLC ที่รัฐบาลนำมาใช้ในโครงการพัฒนาระบบสารสนเทศนั้นใช้รูปแบบ Waterfall Model ซึ่งประกอบไปด้วย 4 ระยะ ดังนี้

- Strategic planning
- Systems analysis
- Design
- Implementation

จากผลการวิจัยของ Nancy Tsai, Beomjin Choi and Mark Perry พบว่า ขณะที่มีความเลือกอื่นในการวิเคราะห์ระบบและการพัฒนาระบบสารสนเทศ อย่างเช่น ตัวต้นแบบ (Prototyping), Object-oriented และอื่นๆ การนำกระบวนการ SDLC แบบดั้งเดิม (Waterfall Model) นั้นมีความเหมาะสมมากที่สุดสำหรับการนำไปใช้กับโครงการที่ใหญ่ ซับซ้อน และเหมาะสมกับสภาพแวดล้อมการดำเนินงานแบบประจำ

He and Griggs, (1994) ได้ศึกษาถึงขั้นตอนการวิเคราะห์ระบบโดยการนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศ มีการจำลอง Model ของการพัฒนาระบบสารสนเทศ โดยการนำ SDLC มาใช้เพื่อประเมินประสิทธิภาพของระบบและการตรวจจับข้อผิดพลาดที่เกิดขึ้น ผลลัพธ์ที่ได้จากการแสดงการจำลองของ Model ในการพัฒนาระบบสารสนเทศ (IS Model) ทำให้สามารถจัดเตรียมการประเมินประสิทธิภาพของระบบและการตรวจจับข้อผิดพลาดที่ขั้นตอนแรกๆ ของการพัฒนาระบบจากกระบวนการ SDLC ได้

การจำลองของ Model ในการพัฒนาระบบสารสนเทศ (IS Model) มีดังต่อไปนี้

- Improved system performance estimation
- Improved hardware and platform selection
- Improved software environment selection
- Improvement in the re-engineering process

จากบทความได้มีการนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศโดยใช้รูปแบบ Waterfall Model ซึ่งประกอบไปด้วย 5 ระยะ ดังนี้

- System analysis
- System design
- Implementation and testing
- System testing
- Operation and maintenance

จากผลงานวิจัยของ JingXiang He, Kenneth A. Griggs พบว่า SDLC นั้นมีความสำคัญและก่อให้เกิดประสิทธิภาพในการพัฒนาระบบสารสนเทศตลอดจนช่วยในการลดต้นทุนในโครงการพัฒนาระบบสารสนเทศ

Verma and Jha, (1992) นี้ได้มุ่งเน้นถึงวิธีการ ที่ซึ่งรวมถึงกระบวนการและระเบียบการปฏิบัติสำหรับ SDLC ของอุตสาหกรรมประเภท Telecommunication โดยกล่าวถึงผลประโยชน์ที่ได้รับจากการนำ SDLC มาใช้ในกลไกระบบ Telecommunication ซึ่งสามารถจะช่วยในเรื่องดังต่อไปนี้

- SDLC นั้นช่วยให้การพัฒนาระบบสารสนเทศทราบความต้องการที่ชัดเจน
- สามารถตรวจพบและแก้ไขการออกแบบที่ผิดพลาดก่อนที่จะทำการติดตั้งระบบ
- ลดต้นทุนในการแก้ไขข้อผิดพลาดที่เกิดขึ้นจากการพัฒนาระบบ
- ลดระยะเวลาในการพัฒนาระบบ
- เพิ่มประสิทธิภาพในการทดสอบ การบำรุงรักษา และปรับปรุงระบบสารสนเทศให้ดี
- ยืดหยุ่นได้ตามการเปลี่ยนแปลงที่เกิดขึ้นจากการใช้งาน

ขึ้น

- ทำให้เกิดการพัฒนาแบบอย่างมีประสิทธิภาพตรงตามความต้องการลูกค้า

จากบทความได้มีการนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศโดยใช้รูปแบบ Incremental Model ซึ่งประกอบไปด้วย 8 ระยะ ดังนี้

- Requirements analysis
- Requirements specification
- High level (Preliminary) software design
- Low level (Detailed) software design
- Software implementation
- Verification & Validation
- Deployment & Maintenance
- Enhancement & Optimization

จากผลงานวิจัยของ S. Devendra K. Verma and Ram D. Jha พบว่า SDLC นั้นช่วยให้การพัฒนาระบบสารสนเทศเกิดความชัดเจน สามารถตรวจพบและแก้ไขการออกแบบที่ผิดพลาดก่อนที่จะทำการติดตั้งระบบ ลดระยะเวลาในการพัฒนาระบบ และยืดหยุ่นได้ตามการเปลี่ยนแปลงที่เกิดขึ้นจากการใช้งานและความต้องการของลูกค้าอีกด้วย

Dennis et al., (1987) ได้ทำการทบทวน (Review) ระบบสารสนเทศปัจจุบันโดยกล่าวถึงวิธีการในการพัฒนาระบบสารสนเทศและตรวจสอบจุดแข็ง จุดอ่อน โดยอธิบายถึงความสำคัญและกระบวนการของระยะในการออกแบบ (Phased design) และกล่าวถึงข้อดีข้อเสีย จากการนำเครื่องมือมาใช้ในการพัฒนาระบบสารสนเทศ โดยงานวิจัยนี้ได้กล่าวถึงการพัฒนาระบบ Application ทั่วไป ซึ่งได้อธิบายถึงวิธีการในการพัฒนาระบบสารสนเทศเพื่อนำมาแก้ปัญหาที่เกี่ยวกับ ความใหญ่ ซับซ้อน ในระบบสารสนเทศที่ต้องการความยืดหยุ่นในการตัดสินใจของข้อมูล ในขณะเดียวกันก็ยังคงควบคุมกระบวนการพัฒนาทั้งโครงการ

งานวิจัยนี้ได้เสนอวิธีการนำ SDLC มาใช้ควบคู่กับตัวต้นแบบ (Prototyping) ในการพัฒนาระบบสารสนเทศเพื่อให้เกิดความเหมาะสมในการพัฒนาระบบ Application อย่างหลากหลาย

จากบทความได้มีการนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศโดยใช้รูปแบบ Incremental Model ซึ่งประกอบไปด้วย 4 ระยะ ดังนี้

- Analysis
- Design
- Coding
- Implementation

จากผลงานวิจัยของ A. R. Dennis, R. N. Burns, and R. B. Gallupe พบว่า SDLC มีข้อจำกัดในการนำมาใช้ เมื่อความต้องการข้อมูลนั้นไม่แน่นอนหรือระบบสารสนเทศได้เปลี่ยนแปลงไปตามความต้องการของธุรกิจ จึงควรมีตัวต้นแบบ (Prototyping) มาใช้ควบคู่ในการพัฒนาระบบสารสนเทศในระยะ Phased design เพื่อนำมาแก้ปัญหาที่เกี่ยวกับ ความใหญ่ ชับซ้อน ในระบบสารสนเทศที่ต้องการความยืดหยุ่นในการตัดสินใจ และ SDLC สามารถทำให้ควบคุมกระบวนการพัฒนาระบบสารสนเทศได้ตลอดทั้งโครงการ

ตารางที่ 2-1 สรุปงานวิจัยที่เกี่ยวข้อง

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
Documentation in systems development: A significant criterion for project success	SDLC	Ricoh Company, Cameras and electronic office equipment	M. Faisal Fariduddin Attar Nasution, H. Roland Weistroffer	2009	Waterfall Model	ประกอบไปด้วย 8 ระยะ ดังนี้ - Feasibility study - Systems investigation - Analysis - Design - Development - Implementation - Testing - Maintenance	- เกิดกระบวนการที่มีความเหมาะสมในการพัฒนาระบบสารสนเทศอย่างมีประสิทธิภาพ - ทำให้ผู้ใช้งานเกิดความพึงพอใจจากการส่งมอบระบบที่ตรงตามความต้องการ - การพัฒนาระบบสารสนเทศตามกรอบ SDLC จะส่งผลให้ช่วยลดความล้มเหลวที่เกิดขึ้นจากการบริหารโครงการได้ (ระยะเวลา, ต้นทุน, คุณภาพ)

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
Improving the process of E-Government initiative: An in-depth case study of web-based GIS implementation	SDLC	E-Government in USA	Nancy Tsai, Beomjin Choi and Mark Perry	2009	Waterfall Model	ประกอบไปด้วย 4 ระยะ ดังนี้ - Strategic planning - Systems analysis - Design - Implementation	- การนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศนั้นทำให้เกิดการทำงานอย่างเป็นลำดับขั้นเหมาะสมกับการทำงานของรัฐบาล - SDLC สามารถยืดหยุ่นและปรับตามความเหมาะสมได้ - SDLC เหมาะกับโครงการที่ใหญ่ ซับซ้อนและเหมาะสมในสภาพแวดล้อมการดำเนินงานแบบประจำ

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
The system development life cycle and digital library development	SDLC	Libraly online	H. Frank Cervone	2007	Waterfall Model	ประกอบไปด้วย 8 ระยะ ดังนี้ - Preliminary investigation Phase - Problem analysis - Requirement analysis - Decision analysis - Design - Construction - Implementation - Operation and Support	- SDLC คือมาตรฐานที่ดีมากพอที่จะนำมาใช้ในการพัฒนาระบบสารสนเทศ - ทำให้สามารถติดตาม รับทราบถึง ความคืบหน้าของโครงการได้เป็นอย่างดี ดี และตรงตามความต้องการของ ผู้ใช้งาน

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
The SDLC and SIX SIGMA an essay on which is which and why?	SDLC Six Sigma	Medical and hospitality	Dr. Roy A. Boggs	2004	Waterfall Model Incremental Model Spiral Model Win-Win Spiral Model V-Model W-Model	-	<ul style="list-style-type: none"> - ทำให้สามารถติดตามผลของการดำเนินโครงการและทราบถึงผลกระทบที่เกิดขึ้นในแต่ละระยะได้เป็นอย่างดี - การพัฒนาระบบสารสนเทศที่ดีจะทำให้เกิดการส่งมอบระบบที่มีคุณภาพตรงตามความต้องการผู้ใช้งาน - เป็นกระบวนการที่มีประสิทธิภาพในการนำมาใช้ในการพัฒนาระบบสารสนเทศ

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
Evaluation in the design of health information systems: Application of approaches emerging from usability engineering	SDLC	Health care	Andre Kushniruk	2002	Waterfall Model	ประกอบไปด้วย 5 ระยะ ดังนี้ - Planning - Analysis - Design - Implementation - Maintenance and support	<ul style="list-style-type: none"> - การนำ SDLC มาใช้นั้นจะทำให้เกิดประสิทธิภาพตลอดทั้งวงจรในการพัฒนาระบบสารสนเทศ - SDLC เหมาะสมกับกระบวนการพัฒนาระบบสารสนเทศที่มีความซับซ้อน ยากต่อการวิเคราะห์รองรับต่อการเปลี่ยนแปลงที่เกิดขึ้นในแต่ละระยะของโครงการ - กระบวนการ SDLC ทำให้ผลลัพธ์ที่ได้จากการพัฒนาระบบสารสนเทศตรงตามความต้องการของผู้ใช้งาน

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
A Tool for hypertext-based systems analysis and dynamic evaluation	SDLC	Software Engineering	JingXiang He, Kenneth A. Griggs	1994	Waterfall Model	ประกอบไปด้วย 5 ระยะ ดังนี้ - System analysis - System design - Implementation and testing - System testing - Operation and maintenance	- SDLC นั้นมีความสำคัญและก่อให้เกิดประสิทธิภาพในการพัฒนาระบบสารสนเทศ - ช่วยในการลดต้นทุนโครงการพัฒนาระบบสารสนเทศ

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
Methodology and tools for mechanization of telecommunication systems specification and development	SDLC	Telecommunication	S. Devendra K. Verma and Ram D. Jha	1992	Incremental Model	ประกอบไปด้วย 8 ระยะ ดังนี้ - Requirements analysis - Requirements specification - High level (Preliminary) software design - Low level (Detailed) software design - Software implementation - Verification & Validation - Deployment & Maintenance - Enhancement & Optimization	- SDLC นั้นช่วยให้การพัฒนาระบบสารสนเทศทราบความต้องการที่ชัดเจน - สามารถตรวจพบและแก้ไขการออกแบบที่ผิดพลาดก่อนที่จะทำการติดตั้งระบบ - ลดต้นทุนในการแก้ไขข้อผิดพลาดที่เกิดขึ้นจากการพัฒนาระบบ - ลดระยะเวลาในการพัฒนาระบบ - เพิ่มประสิทธิภาพในการทดสอบ การบำรุงรักษา และปรับปรุงระบบสารสนเทศให้ดีขึ้น - ยืดหยุ่นได้ตามการเปลี่ยนแปลงที่เกิดขึ้นจากการใช้งาน - ทำให้เกิดการพัฒนาระบบอย่างมีประสิทธิภาพตรงตามความต้องการลูกค้า

ตารางสรุปงานวิจัยที่เกี่ยวข้อง (ต่อ)

ชื่องานวิจัย	ทฤษฎี	อุตสาหกรรม	ผู้วิจัย	ปี	Model	ระยะของ SDLC	ผลลัพธ์ที่ได้จาก SDLC
Phased design: A mixed methodology for application system development	SDLC	Several application systems	A. R. Dennis, R. N. Burns, and R. B. Gallupe	1987	Waterfall Model	ประกอบไปด้วย 4 ระยะ ดังนี้ - Analysis - Design - Coding - Implementation	- การพัฒนาระบบสารสนเทศเพื่อนำมาแก้ปัญหาที่เกี่ยวกับ ความสำเร็จ ซับซ้อน ในระบบสารสนเทศที่ต้องการความยืดหยุ่นในการตัดสินใจ - SDLC สามารถทำให้ควบคุมกระบวนการพัฒนาระบบสารสนเทศได้ตลอดทั้งโครงการ

ตารางที่ 2-2 สรุปผลลัพธ์ SDLC ที่ได้จากงานวิจัย

ผลงานวิจัย	ประโยชน์ที่ได้จาก SDLC					
	เกิดกระบวนการในการพัฒนาระบบอย่างมีประสิทธิภาพ	ส่งมอบระบบที่มีคุณภาพตรงความต้องการผู้ใช้งาน	ช่วยเพิ่มประสิทธิภาพในการบริหารโครงการ (ระยะเวลา, ต้นทุน, คุณภาพ)	ยืดหยุ่น ปรับเปลี่ยนได้เมื่อความ ต้องการเปลี่ยนแปลงไป	เหมาะสมกับการพัฒนาระบบที่ใหญ่ ซับซ้อน	ติดตาม รับทราบความเคลื่อนไหวได้ ตลอดทั้งโครงการ
M. Faisal Fariduddin Attar Nasution and H. Roland Weistroffer	X	X	X			
Nancy Tsai, Beomjin Choi and Mark Perry	X			X	X	
H. Frank Cervone	X	X				X
Dr. Roy A. Boggs	X	X				X
Andre Kushniruk	X	X			X	
JingXiang He and Kenneth A. Griggs	X		X			
S. Devendra K. Verma and Ram D. Jha	X	X		X		
A. R. Dennis, R. N. Burns, and R. B. Gallupe				X	X	X

2.4.1 ประโยชน์ของ SDLC

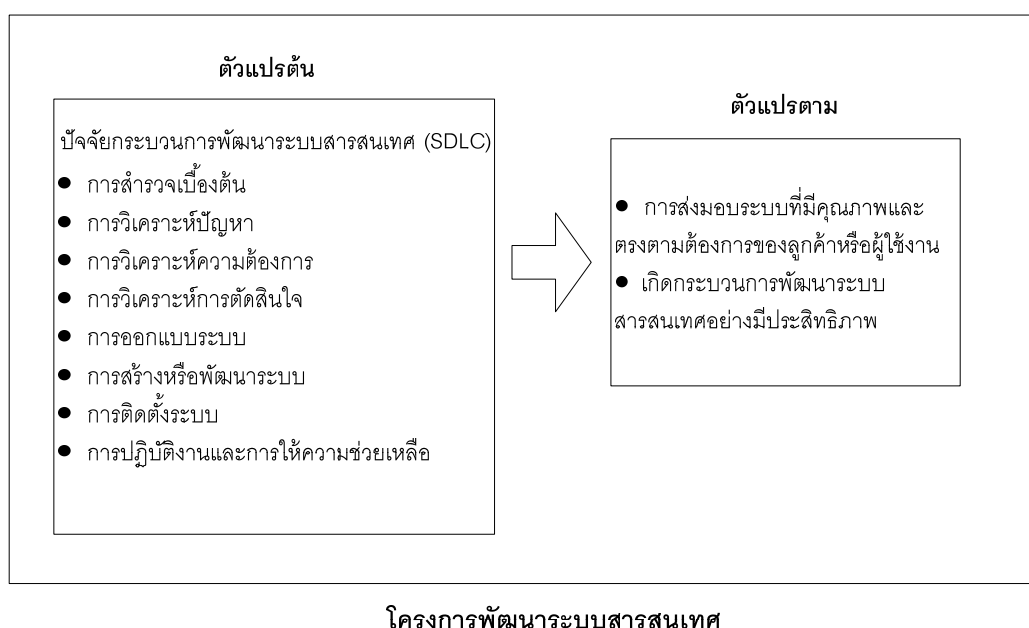
จากตารางที่ 2-2 สรุปผลลัพธ์ SDLC ที่ได้จากงานวิจัย ได้กล่าวถึงประโยชน์ในการนำ SDLC มาใช้ในการพัฒนาระบบสารสนเทศ ดังต่อไปนี้

- เกิดกระบวนการในการพัฒนาระบบอย่างมีประสิทธิภาพ
- ส่งมอบระบบที่มีคุณภาพตรงความต้องการผู้ใช้งาน
- ช่วยเพิ่มประสิทธิภาพในการบริหารโครงการพัฒนาระบบสารสนเทศ (ระยะเวลา, ต้นทุน, คุณภาพ)

- ยืดหยุ่น ปรับเปลี่ยนได้เมื่อความต้องการเปลี่ยนแปลงไป
- เหมาะกับการพัฒนาระบบสารสนเทศที่ใหญ่ ซับซ้อน
- สามารถติดตาม รับทราบความเคลื่อนไหวได้ตลอดทั้งโครงการ

2.4.2 กรอบแนวคิดวิจัย

ภาพที่ 2-7 กรอบแนวคิดวิจัย



จากกรอบแนวคิดการวิจัยสามารถสรุปได้ว่า ปัจจัยของ SDLC ทั้ง 8 ขั้นตอนซึ่งในแต่ละขั้นตอนก็จะมีกิจกรรมที่จะต้องดำเนินการต่างๆ เพื่อให้บรรลุวัตถุประสงค์ของขั้นตอนนั้นๆ จะมีผลต่อการส่งมอบระบบที่มีคุณภาพและตรงตามความต้องการของลูกค้าหรือผู้ใช้งานและการเกิดกระบวนการพัฒนาระบบสารสนเทศอย่างมีประสิทธิภาพ ภายในโครงการพัฒนาระบบสารสนเทศหรือไม่ ซึ่งข้อมูลดังกล่าวนี้จะมาจากการรวบรวมข้อมูลที่ได้จากแบบสอบถามและใช้เครื่องมือในการวิเคราะห์ทางสถิติ