

หัวข้อวิทยานิพนธ์	การศึกษาปัจจัยเสี่ยงในโครงการซอฟต์แวร์ขนาดเล็กในประเทศไทย	<div data-bbox="1235 392 1570 506" style="border: 1px solid red; padding: 5px;"> ข้อคิดเห็น[U1]: ให้เลือกพิมพ์(วิทยานิพนธ์ / การค้นคว้าอิสระ / โครงการ ทางวัฒนธรรม) </div> <div data-bbox="1235 506 1570 554" style="border: 1px solid blue; padding: 5px;"> ข้อคิดเห็น[N2]: พิมพ์คำนำหน้านาม </div> <div data-bbox="1235 554 1570 602" style="border: 1px solid blue; padding: 5px;"> ข้อคิดเห็น[N3]: พิมพ์คำนำหน้านาม </div> <div data-bbox="1235 602 1570 716" style="border: 1px solid red; padding: 5px;"> ข้อคิดเห็น[U4]: ให้เลือกพิมพ์ (สาขาวิชาการบริหารเทคโนโลยี/ สาขาวิชาการบริหารงานวัฒนธรรม) </div> <div data-bbox="1235 716 1570 800" style="border: 1px solid blue; padding: 5px;"> ข้อคิดเห็น[N5]: ให้พิมพ์ปีการศึกษาที่ทำงานวิจัย (ไม่ใช่ปี พ.ศ.) </div>
ชื่อผู้เขียน	นายพิภพ ฉัตรภุติ	
แผนกวิชา/คณะ	การบริหารเทคโนโลยี วิทยาลัยนวัตกรรม มหาวิทยาลัยธรรมศาสตร์	
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ดร.ภูมิพร ธรรมสถิตย์เดช	
ปีการศึกษา	พ.ศ.2552	

บทสรุปงานวิจัย

บทนำ

เนื่องจากสภาวะการแข่งขันของธุรกิจในปัจจุบัน ทวีความรุนแรงมากขึ้น ดังนั้นเพื่อเพิ่มขีดความสามารถในการแข่งขันให้กับธุรกิจตนเอง ผู้ประกอบการจึงได้นำเอาเครื่องมือบริหารต่างๆมาใช้ จากผลสำรวจในปี ค.ศ. 2007 เครื่องมือบริหารที่ได้รับความนิยมมากที่สุดใน 3 ลำดับแรก ได้แก่ 1) การวางแผนกลยุทธ์ (Strategic Planning) 2) การบริหารลูกค้าสัมพันธ์ (Customer Relation Management) และ 3) การแบ่งส่วนลูกค้า (Customer Segmentation) [1] อย่างไรก็ตาม ผู้บริหาร 9 ใน 10 มีความเห็นสอดคล้องกันว่าเครื่องมือบริหารต่างๆเหล่านี้ จะไม่สามารถทำงานได้หากปราศจากระบบสารสนเทศ ที่มีประสิทธิภาพ (Information Technology) [1] นอกจากนี้ ระบบสารสนเทศที่มีประสิทธิภาพ ยังช่วยเพิ่มขีดความสามารถในการแข่งขันของธุรกิจอีกด้วย [2-3] ซอฟต์แวร์เป็นเครื่องมือที่นักบริหารรุ่นใหม่มักนำมาใช้ ในองค์กร แต่เนื่องจากความแตกต่างของการทำธุรกิจ จึงมีการประยุกต์ใช้ซอฟต์แวร์ ให้เหมาะกับธุรกิจของตนเอง อย่างไรก็ตาม การทำโครงการซอฟต์แวร์ที่ผ่านมา มีอัตราล้มเหลวสูง และยังคงต่อเนื่อง มาถึงปัจจุบัน [4-5-6-7] ในประเทศไทยแนวโน้มการเติบโตของธุรกิจซอฟต์แวร์ลดลง แต่ยังคงมีมูลค่าสูงอยู่ อุตสาหกรรมซอฟต์แวร์ไทยส่วนใหญ่เป็นธุรกิจขนาดกลางและขนาดเล็ก [8] ในการทำโครงการซอฟต์แวร์ หากผู้ประกอบการซอฟต์แวร์ไทย นำเอาหลักการบริหารความเสี่ยงมาใช้ จะช่วยเพิ่มโอกาสให้การทำโครงการประสบความสำเร็จมากขึ้น [9-10-11] ในการบริหารความเสี่ยง การระบุปัจจัยเสี่ยงเป็นขั้นตอนที่สำคัญที่สุด แต่ที่ผ่านมา งานวิจัยด้านปัจจัยเสี่ยง ส่วนใหญ่ ทำการศึกษาการทำโครงการซอฟต์แวร์ขนาดใหญ่ [12] แต่ยังไม่มีความสนใจในการศึกษาปัจจัยเสี่ยงในโครงการขนาดเล็กมาก่อน งานวิจัยนี้มีวัตถุประสงค์เพื่อทำการศึกษาปัจจัยเสี่ยงในโครงการซอฟต์แวร์ขนาดเล็ก ซึ่งมีผู้ประกอบการจำนวนมาก

อุตสาหกรรมซอฟต์แวร์

รายงานการลงทุนในอุตสาหกรรมซอฟต์แวร์ ประเทศสหรัฐอเมริกา แสดงให้เห็นแนวโน้มการเติบโตในอุตสาหกรรมนี้ ซึ่งยังคงเพิ่มขึ้นเรื่อยๆ ในปีค.ศ. 2002 มีมูลค่าการลงทุนด้านนี้ถึง 88,156 ล้านดอลลาร์และเพิ่มขึ้นเป็น 104,167 ล้านดอลลาร์ ในปี ค.ศ. 2006 [13] ส่วนมูลค่าการลงทุนของอุตสาหกรรมซอฟต์แวร์ในประเทศไทย จากการสำรวจโดย

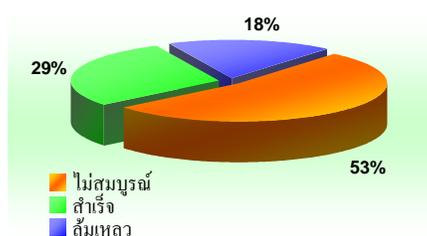
สำนักงานส่งเสริมอุตสาหกรรมซอฟต์แวร์แห่งชาติ (SIPA: Software Industry Promotion Agency) และสมาคมอุตสาหกรรมเทคโนโลยีซอฟต์แวร์ไทย (ATCI : The Association of Thai ICT Industry) ระหว่างเดือนตุลาคม-เดือนพฤศจิกายน พ.ศ. 2551 พบว่า มีมูลค่าตลาดรวม 62,937 ล้านบาท เติบโตจากปีที่ผ่านมา 11.2% แต่มีแนวโน้มจะลดลงเหลือ 5.1% ณ สิ้นปี พ.ศ. 2552 ซอฟต์แวร์สำเร็จรูป (Software Package) ถือครองตลาดที่ 71.8% ในขณะที่ซอฟต์แวร์ที่พัฒนาใหม่ (Software Development) มีสัดส่วนที่ 28.2% [8] มูลค่าอุตสาหกรรมซอฟต์แวร์ไทยมีแนวโน้มเติบโตสูงขึ้นเรื่อยๆ ซึ่งสอดคล้องกับนโยบายภาครัฐที่ต้องการส่งเสริมอุตสาหกรรมนี้ให้เติบโตมากขึ้น นอกจากนี้รัฐบาลยังมีนโยบาย ส่งเสริมผู้ประกอบการอุตสาหกรรมขนาดย่อมและขนาดเล็ก ให้มีศักยภาพในการแข่งขันได้มากขึ้น คาดว่าในอนาคตแนวโน้มความต้องการใช้ซอฟต์แวร์ขนาดเล็ก จะมีแนวโน้มเพิ่มขึ้นด้วยเช่นกัน

โครงการซอฟต์แวร์

โดยทั่วไปแล้ว โครงการซอฟต์แวร์สามารถจำแนกออกเป็นสองประเภท คือ โครงการพัฒนาขึ้นใหม่ (Software Project Development) และโครงการประยุกต์หรือโครงการที่พัฒนาเฉพาะบางหน่วยงาน (Software Project Implementation) บางครั้งระบบซอฟต์แวร์ถูกพัฒนา เพื่อประยุกต์ใช้เพียงเฉพาะด้าน เช่น ระบบวิเคราะห์การเงิน ระบบวิเคราะห์ประสิทธิภาพของการผลิต ลักษณะดังกล่าวเข้าข่ายโครงการขนาดเล็ก ประเภทหนึ่ง ในการทำโครงการซอฟต์แวร์ ขนาดของโครงการที่แตกต่างกัน วิธีการปฏิบัติและการบริหารก็แตกต่างด้วยเช่นกัน โครงการขนาดใหญ่ใช้เงินลงทุนสูง ระยะเวลายาวนาน และบุคลากรจำนวนมาก นอกจากนั้น ยังใช้เทคโนโลยีที่มีความซับซ้อนสูง ทำให้มีปัจจัยเสี่ยงต่างๆ ในการทำโครงการมีมาก ในขณะที่โครงการขนาดเล็กใช้เงินลงทุนต่ำ ระยะเวลาสั้น มีความซับซ้อนและใช้บุคลากรในโครงการน้อยกว่า [9] แต่พึ่งพาความสามารถของบุคลากรเป็นหลัก แนวปฏิบัติที่ใช้ในการบริหาร โครงการมีความแตกต่างไปจากโครงการขนาดใหญ่ และมีปัจจัยเสี่ยงและผลกระทบที่แตกต่างด้วยเช่นกัน

ความล้มเหลวโครงการซอฟต์แวร์

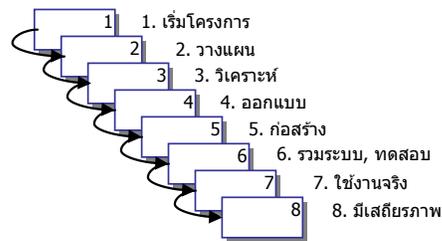
ในปี ค.ศ. 1997 ผลการสำรวจความล้มเหลวของโครงการซอฟต์แวร์ในแคนาดา พบว่า 30% เกิดจากการใช้งบประมาณเกิน อีกจำนวน 30% ไม่สามารถ ส่งมอบโครงการได้ตามกำหนดเวลา นอกนั้นไม่ตรงตามที่ลูกค้าต้องการ [7] ในปี ค.ศ. 2004 ผลการสำรวจโดยกลุ่มบริษัทสแตนดิช (รูปที่ 1) พบว่า โครงการซอฟต์แวร์ที่ประสบผลสำเร็จมีจำนวน 29% ล้มเหลว 18% และไม่สมบูรณ์ 53% [14] จากสถิติที่ผ่านมา แนวโน้มการพัฒนาโครงการซอฟต์แวร์ มีอัตราความล้มเหลวสูงมากขึ้น [4-5-6-7]



รูปที่ 1 ความล้มเหลวโครงการซอฟต์แวร์
จากการสำรวจของกลุ่มบริษัทสแตนดิช

การบริหารโครงการซอฟต์แวร์

รูปแบบและขั้นตอนวงจรชีวิตของโครงการซอฟต์แวร์มีผู้ ออกแบบไว้หลายแบบซึ่งส่วนใหญ่จะคล้ายคลึงกัน รูปที่ 2 เป็นแบบที่จัดทำโดยกระทรวงเกษตร ประเทศสหรัฐอเมริกา มีทั้งหมด 8 ช่วง ตั้งแต่เริ่มโครงการ วางแผนโครงสร้าง ด้านบุคลากร เวลา ต้นทุน จากนั้นทำการวิเคราะห์ความต้องการ แล้วจึงนำมาออกแบบ นำไปปฏิบัติงาน จนกระทั่งพร้อมแล้วก็ทำการทดสอบ และประสานกับระบบเดิม จากนั้นจึงจัดอบรมให้ผู้ใช้ เมื่อทุกอย่างพร้อม ก็เริ่มใช้งานจริง จนกระทั่งแน่ใจว่าระบบมีเสถียรภาพมั่นคง จึงส่งมอบโครงการ [15]



รูปที่ 2 วงจรชีวิตของโครงการซอฟต์แวร์

การบริหารความเสี่ยง

“ความเสี่ยง คือความไม่เที่ยงแท้แน่นอนของเหตุการณ์ในอนาคต เกิดขึ้นได้จากปัจจัยภายในและปัจจัยภายนอก ซึ่งหากเกิดขึ้นแล้วอาจส่งผลกระทบต่อทั้งด้านดี และไม่ดี” [9-16-17] ความเสี่ยงสามารถเกิดขึ้นได้ในทุกช่วงของการทำงานโครงการ การใช้ทฤษฎีและหลักการของการบริหารความเสี่ยงเข้ามาใช้ในการบริหารโครงการเป็นการเพิ่มโอกาสการพัฒนาโครงการให้สำเร็จมากขึ้น[9-18] โดยการลดหรือจัดความเสี่ยงก่อนที่จะเกิดความผิดพลาดขึ้น รวมถึงการวางมาตรการบรรเทา หากเกิดความผิดพลาดขึ้นในโครงการ ประโยชน์ที่ได้จากการบริหารความเสี่ยงมีหลายประการ ดังนั้นหลายสถาบัน ต่างให้ความสำคัญและพัฒนารูปแบบ เพื่อใช้เป็นกรอบและแนวทางสำหรับการบริหารความเสี่ยงในโครงการซอฟต์แวร์ [19-20-21-22-23] การบริหารความเสี่ยงแบบเต็มรูปแบบบางครั้งใช้ทุนสูงมาก เนื่องจากมีปัจจัยเสี่ยงมากที่ต้องบริหารจัดการ [9] ดังนั้น ผู้บริหารโครงการ ต้องกำหนดขอบเขตในการดำเนินการบริหารความเสี่ยง แหล่งปัจจัยเสี่ยงเกิดขึ้นได้จากปัจจัยภายนอกและปัจจัยภายใน ปัจจัยภายนอกบางอย่างไม่สามารถควบคุมได้เช่น นโยบายภาครัฐ แต่สามารถที่จะสร้างแผนรองรับได้ พบว่า ในการลงทุนเพื่อพัฒนาโครงการขนาดใหญ่ ผู้บริหารโครงการจะวางแผนป้องกันความผิดพลาดจากความเสี่ยงต่างๆ อย่างรัดกุมและเป็นทางการ แตกต่างจากโครงการขนาดเล็ก ซึ่งมักจะละเลยและไม่ค่อยให้ความสำคัญกับการบริหารความเสี่ยง [24-25] นอกจากนี้ พบว่าปัจจัยเสี่ยงเกิดขึ้นในโครงการ ทุกๆขนาด [26] กล่าวว่าการบริหารความเสี่ยงที่ดีที่สุด คือการจัดการกับเรื่องที่ไม่แน่นอนซึ่งเป็นเรื่องสำคัญ ในแนวทางที่เกิดผลและได้ผลตามที่ต้องการ ซึ่งต้องทำความเข้าใจกับเหตุการณ์ที่ไม่แน่นอนในเบื้องต้นเสียก่อน อะไรเป็นความเสี่ยง ทำไมจึงเสี่ยง ควรทำอะไร และอะไรควรทำ ใครมีหน้าที่ในการจัดการและต้องใช้งเงินเท่าไร [10] แนวคิดพื้นฐานของการบริหารความเสี่ยงอย่างง่ายคือ การตรวจสอบความเป็นไปได้ที่จะเกิดสิ่งไม่ดีขึ้น และลดทอนความรุนแรงที่เกิดขึ้น หรือเพื่อเพิ่มโอกาส แต่กระนั้น วิธีการและกระบวนการในการพัฒนาถือเป็นเรื่องที่สำคัญมากกว่า การบริหารความเสี่ยงควรประกอบด้วยพื้นฐาน ของการป้องกันความเสี่ยง การบรรเทาและการแก้ไข

ความเสี่ยง ซึ่งทำเพื่อสร้างความมั่นใจว่าระบบจะปลอดภัย[27] กล่าวได้ว่า ผู้บริหารโครงการที่ประสบความสำเร็จนั้น ควรเป็นผู้ที่ซึ่งบริหารความเสี่ยงได้เป็นอย่างดี[9] การบริหารความเสี่ยงจะประสบความสำเร็จและเป็นผลดีกับโครงการ หากว่าได้รับการสนับสนุนจากผู้บริหารในด้านต่างๆ เช่น สนับสนุนเครื่องมือ การวางแผน การติดตามความเสี่ยง [28] การประเมินความเสี่ยงในโครงการซอฟต์แวร์นั้นสามารถแบ่งเป็นด้านการบริหาร และด้านเทคนิค ในด้านการบริหารนั้นปรากฏในรูปของ การวางแผน การดูแลติดตาม การควบคุมคุณภาพ และการจัดการต่างๆ ส่วนด้านเทคนิคนั้นจะปรากฏในรูปของ การวิเคราะห์ความต้องการของลูกค้า การออกแบบ การเขียนโปรแกรม และการทดสอบ[11-25] หลักการบริหารความเสี่ยงโดยทั่วไป ประกอบด้วยพื้นฐาน 6 ขั้นตอน [9]

1. ระบุความเสี่ยง
2. วิเคราะห์ความเสี่ยง
3. จัดระดับความเสี่ยง
4. วางแผนบริหารความเสี่ยง
5. ลงมือทำตามความเสี่ยง
6. ควบคุมติดตามความเสี่ยง

ขั้นตอนที่สำคัญที่สุดคือการระบุความเสี่ยง เนื่องจากเป็นช่วงแรกของการบริหารความเสี่ยง หากการระบุความเสี่ยง ถูกต้องตั้งแต่ต้น ขบวนการถัดไปจึงจะทำได้ถูกต้อง อย่างไรก็ตาม ขั้นตอนการบริหารความเสี่ยง ไม่ได้กำหนดไว้ตายตัว ขึ้นอยู่กับความถนัดและความเหมาะสมในแต่ละโครงการว่าจะดำเนินตามแนวทางใดหรือใช้รูปแบบใด [19-20-21-22-27]

การประเมินขนาดของโครงการ

การทราบขนาดของโครงการ มีความสำคัญมาก เนื่องจากการกำหนดระยะเวลา งบประมาณ และทรัพยากรที่จะต้องจัดสรรเข้าไปในโครงการ มีความสัมพันธ์โดยตรงกับขนาดของโครงการ ผลการสำรวจจากผู้เชี่ยวชาญ 18% ระบุว่า การประเมินถือเป็นจุดแข็งของผู้ประกอบการ และ 70-80% ระบุว่า การประเมินโครงการที่ไม่มีดี มีผลกระทบต่อการเงิน การสูญเสียความสามารถในการแข่งขัน และทำให้การส่งมอบโครงการล่าช้า [29-30] การประเมินที่ผิดพลาดจะส่งผลให้เกิดความเสี่ยงขึ้น และอาจส่งผลให้โครงการไม่ประสบความสำเร็จ การกำหนดขนาดของโครงการที่ผ่านมายังไม่มีวิธีที่ดีที่สุด ซึ่งก็ขึ้นอยู่กับผู้บริหาร โครงการว่าจะใช้รูปแบบใด หากสามารถแยกโครงการที่มีความซับซ้อนมาก ให้เป็นส่วนประกอบย่อยๆ ได้ชัดเจนเพียงใด จะทำให้การประเมินโครงการถูกต้องมากขึ้น [9] ในการประเมินโครงการซอฟต์แวร์ให้ได้อย่างสมบูรณ์แบบ เวลาและต้นทุน เป็นสิ่งที่สามารถพยากรณ์ได้ยาก [31] การประเมินโครงการอย่างง่ายมีวิธีการที่ได้รับความนิยม 4 วิธี ดังนี้

1. ประเมินจากจำนวนแถวของรหัส (Line Of Code) หรือ จำนวนของฟังก์ชัน (Function Point) รวมถึงรายละเอียดปลีกย่อยอื่นๆ
2. ประเมินจากจำนวนแรงคนต่อเดือน
3. ประเมินจากจำนวนวันต้องทำงานจริง
4. ประเมินค่าใช้จ่ายที่เกิดขึ้นในโครงการ

การประเมินโครงการอย่างละเอียด อาจจะใช้การประเมินโดย การเปรียบเทียบจากขนาดของโครงการที่เคยได้ทำมา หรือการประเมินจากกรณีวิเคราะห์จำนวนรหัส พังก์ชันที่ต้องใช้งานเป็นต้น การประเมินควรจะดำเนินการทันทีที่ได้รับ ข้อมูลต่างๆครบถ้วน เช่น รายงานระบุความต้องการของผู้ใช้ ระบบที่ต้องใช้ และซอฟต์แวร์ที่ใช้พัฒนา การประเมิน โครงการควรได้รับการปรับปรุงวิธีการอยู่เสมอ การปรับปรุงการประเมินที่ได้ผลดีที่สุดคือการปรับปรุงความเข้าใจ ขบวนการต่างๆที่เกี่ยวข้องในการทำซอฟต์แวร์นั่นเอง [30]

ขนาดของโครงการเล็ก

จากงานวิจัยที่พบ มีผู้กำหนด ขนาดของโครงการขนาดเล็ก โดยใช้จำนวนแรงคน 6 ถึง 24 แรงต่อเดือน (Man-month) [24] ในขณะที่ผู้วิจัยบางท่านกำหนดโครงการขนาดเล็ก โดยที่ใช้เวลาในการดำเนินการประมาณ 12 เดือนหรือน้อย กว่า [7] การประเมินขนาดของโครงการ ให้ได้ใกล้เคียงกับความเป็นจริงนั้นยังคงเป็นสิ่งที่ยังไม่มีวิธีที่ดีพอ จนเป็นที่ ยอมรับของทุกฝ่าย พบว่ามีวิธีและเทคนิคการประเมินหลายวิธีในปัจจุบัน ซึ่งแต่ละองค์กรก็ใช้วิธีการที่แตกต่างกันไป [32-33-34-35] อย่างไรก็ตาม การศึกษานี้โครง การขนาดเล็กหมายถึงโครงการที่ใช้ระยะเวลาในการทำโครงการน้อย กว่า 12 เดือน

ขั้นตอนวิธีวิจัย

เนื่องจากการประเมินปัจจัยความเสี่ยงเป็นเรื่องที่ต้องสร้างความเข้าใจในนิยาม การบริหารความเสี่ยงโครงการ ซอฟต์แวร์ ในประเทศไทย เป็นเรื่องที่ยังไม่คุ้นเคยสำหรับผู้บริหารโครงการ นอกจากนั้นการประเมินความเสี่ยงแต่ละ บุคคลก็แตกต่างกัน จึงต้องทำความเข้าใจเพื่อให้การประเมินอยู่ในระดับเดียวกัน ซึ่งจะทำให้ข้อมูลที่ได้รับถูกต้องมาก ขึ้น การสัมภาษณ์เชิงลึก (In-depth Interview) จึงเป็นสิ่งที่เหมาะสม เนื่องจากการบริหารความเสี่ยงเป็นเรื่องใหม่ สำหรับผู้ประกอบการซอฟต์แวร์ไทย ดังนั้นจึงต้องอธิบายให้เข้าใจในนิยาม และถามคำถามที่ง่ายต่อความเข้าใจกับทั้ง สองฝ่าย คำถามได้ถูกทดสอบล่วงหน้าโดยการสัมภาษณ์ผู้จัดการ โครงการด้านไอทีท่านหนึ่งจากนั้นจึงทำการปรับปรุง ให้คำถามเข้าใจง่ายและออกแบบให้ใช้เวลาสัมภาษณ์ภายใน 1 ชั่วโมง

กลุ่มตัวอย่าง

การเลือกกลุ่มตัวอย่างเพื่อสัมภาษณ์ พิจารณาจากประการต่างๆดังต่อไปนี้

1. เป็นผู้ดำรงตำแหน่งผู้บริหาร โครงการหรือเป็นผู้ซึ่งมีอำนาจในการตัดสินใจ
2. มีประสบการณ์ในการบริหาร โครงการมากกว่า 10 ปี
3. มีประสบการณ์ในการบริหาร โครงการขนาดเล็กมากกว่า 10 โครงการ

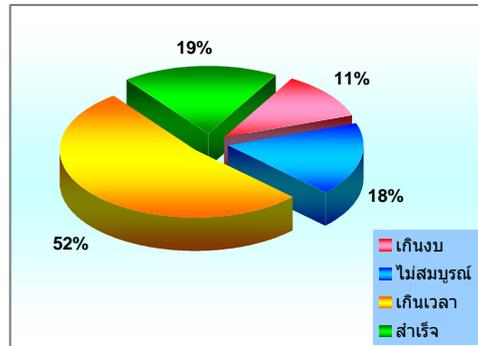
ในการศึกษานี้ กลุ่มตัวอย่าง แบ่งได้เป็น (1) กลุ่มที่ได้จากการเจาะจง จำนวน 9 ท่าน (2) กลุ่มที่ได้จากการแนะนำโดย กลุ่มแรก จำนวน 3 ท่าน และ (3) กลุ่มที่เลือกจากรายชื่อผู้ประกอบการที่ได้รับรองคุณภาพจากสมาคมอุตสาหกรรม ซอฟต์แวร์ไทย จำนวน 3 ท่าน เลือกโดยผู้วิจัยเอง เหตุผลที่ต้องเลือกกลุ่มตัวอย่างเอง เนื่องจากเพื่อให้ได้ข้อมูลที่เป็น ประโยชน์และตรงกับการศึกษา ผู้ให้สัมภาษณ์ส่วนใหญ่ทำงานในบริษัทซอฟต์แวร์ขนาดใหญ่ เช่น บริษัท ปตท ไอซีที, บริษัท ซิเมนต์ประเทศไทย และบริษัทฟูจิตสียูนิเจนประเทศไทย เป็นต้น

กรอบคำถาม

คำถามจะแบ่งเป็นสองกลุ่มคือ คำถามทั่วไปเกี่ยวกับการทำโครงการ เช่น จำนวนโครงการขนาดเล็กที่ทำต่อปี ระยะเวลา งบประมาณ กระบวนการในการทำโครงการทั้งหมด ตั้งแต่เริ่มโครงการจนกระทั่งส่งมอบโครงการ ซึ่งทำให้ทราบการทำงานและรูปแบบการทำงานของผู้ให้สัมภาษณ์ จากนั้นจึงเป็นคำถามที่เจาะลึกปัญหาที่พบ ในระหว่างที่ทำโครงการ ที่ซึ่งส่งผลกระทบต่อความสำเร็จของโครงการทั้งด้านเวลา งบประมาณ และคุณภาพ นอกจากนี้ยังถามถึงปัญหาที่พบ ในแต่ละช่วงของโครงการที่ผู้ถูกสัมภาษณ์ได้อธิบายไปก่อนหน้านี้ แต่ปัญหาที่พบจะให้ผู้ถูกสัมภาษณ์ ระบุความถี่และรุนแรงหากเกิดปัญหาดังกล่าว คำถามที่ถามได้รับอิทธิพลจาก งานวิจัยของ Ropponen [36] คำถามมี 20 คำถามจัดเป็น 6 หมวดได้แก่ ด้านบุคลากร ด้านการเปลี่ยนความต้องการ ด้านเวลา ด้านบริหาร ด้านเทคโนโลยี และด้านการวางแผน การสัมภาษณ์ใช้เวลาประมาณ 1 ชั่วโมงต่อคน จากนั้นคำตอบทั้งหมดจะถูกนำมาวิเคราะห์ (Content Analysis) และจัดลำดับความถี่ที่เกิดและความรุนแรงแล้วจึงนำมาสรุปเป็นปัจจัยเสี่ยง

ผลการศึกษา

พบว่า โครงการซอฟต์แวร์ขนาดเล็ก ประสบความสำเร็จ 19% โครงการส่งมอบไม่ตรงเวลา 52.3 % และสมรรถนะของผลิตภัณฑ์ไม่เป็นที่พอใจของลูกค้า 17.7% ที่เหลือเกิดจากการใช้งบประมาณเกินกว่ากำหนด 11.3% ดังรูปที่ 3 ปัจจัยเสี่ยงต่างๆที่พบจากการศึกษา ประกอบด้วยปัจจัยเสี่ยง 14 ประการ แบ่งเป็น 5 กลุ่มดังในภาคผนวก ตารางที่ 1



รูปที่ 3 ความล้มเหลว

ของโครงการซอฟต์แวร์ขนาดเล็ก

จะสังเกตได้ว่ามีปัจจัยเสี่ยงบางอย่างที่พบในโครงการขนาดใหญ่ แต่ไม่พบในโครงการขนาดเล็กเลย คือ ปัจจัยเสี่ยงจากการใช้ระบบคอมพิวเตอร์ที่ใช้งาน ปัจจัยการเมือง และปัจจัยเสี่ยงจากผู้รับเหมาช่วง จากการวิเคราะห์โอกาสเกิดขึ้นและระดับความรุนแรง ในแต่ละกลุ่มเสี่ยงทำให้ได้ข้อสรุปว่า มีปัจจัยสำคัญ 5 ประการที่อย่างน้อยควรได้รับการติดตามและควบคุมในโครงการซอฟต์แวร์ขนาดเล็กดังนี้

1. บุคลากรขาดความรู้และทักษะ
2. การบริหารและการทำงานที่บกพร่อง
3. ลูกค้าเปลี่ยนความต้องการ
4. การบริหารและการทำงานที่บกพร่อง
5. เทคโนโลยีไม่ตอบสนองต่อความต้องการ

สรุปผลการศึกษา

การศึกษาปัจจัยเสี่ยงในโครงการซอฟต์แวร์ขนาดเล็กในประเทศไทย สรุปได้ว่า มีปัจจัยเสี่ยงที่พบ 14 ประการ แบ่งเป็น 5 กลุ่มคือ (1) ความเสี่ยงจากบุคลากร (2) ความเสี่ยงจากการเปลี่ยนความต้องการ (3) ความเสี่ยงจากการจัดการบริหาร (4) ความเสี่ยงจากการประเมิน และ (5) ความเสี่ยงจากเทคโนโลยี พบว่ามีปัจจัยเสี่ยงบางประการสอดคล้องกับงานวิจัยที่ผ่านมา [7-9-36] โดยที่ปัจจัยเสี่ยงที่พบในโครงการขนาดเล็กทั้งหมด เป็นปัจจัยเสี่ยงที่เกิดขึ้นในโครงการขนาดใหญ่ด้วยเช่นกัน ดูรายละเอียดในภาคผนวก ในตารางที่ 2

ข้อจำกัดในการศึกษา

การนำกลุ่มตัวอย่างเพื่อสัมภาษณ์ ทำได้ยาก เนื่องจากกลุ่มตัวอย่างไม่มีเวลาและไม่เห็นความสำคัญของการศึกษาดังนั้น การเลือกกลุ่มตัวอย่างเพื่อให้ครอบคลุมธุรกิจซอฟต์แวร์หลายๆด้าน จึงไม่สามารถทำได้ นอกจากนี้พื้นเพความรู้ความเข้าใจในเรื่องการบริหารความเสี่ยงแตกต่างกัน คำถามแบบเดียวกัน คำตอบอาจจะได้รับไม่เหมือนกัน ในการบันทึกข้อมูล บางครั้งไม่สามารถให้บันทึกตอบได้ จึงต้องใช้การจด ดังนั้นจึงแนะนำให้ทำการถอดคำสัมภาษณ์ให้เสร็จภายในวันเดียวกัน

ข้อเสนอแนะในการศึกษาต่อ

เนื่องจากการบริหารความเสี่ยงในโครงการซอฟต์แวร์ในประเทศไทย เป็นเรื่องใหม่ ดังนั้นเมื่อเริ่มมีการศึกษาด้านนี้ จึงมีอุปสรรคอยู่บ้างในตอนแรก เนื่องจากทัศนคติ และการรับรู้ในเรื่องความเสี่ยงแตกต่างกัน การวิจัยเชิงปริมาณ โดยใช้การสำรวจจากการส่งแบบสอบถามอาจจะไม่เหมาะสม อาจจะทำให้ได้รับคำตอบที่อาจผิดพลาด อย่างไรก็ตามในอนาคต อาจจะกระทำได้สะดวกมากขึ้น เนื่องจากการบริหารความเสี่ยงเป็นสิ่งที่เข้าใจกันมากขึ้น ดังนั้น หากมีการศึกษาเพิ่มเติมโดย ขยายขนาดของกลุ่มตัวอย่างให้มากขึ้น จะทำให้การศึกษาด้านนี้หนักแน่นมากขึ้น และนำไปสู่ การศึกษาเพิ่มเติมอื่นๆ จะกระทำวันหนึ่ง ประเทศไทย อาจจะมีหลักการมาตรฐานในการบริหารโครงการ ที่เหมาะสมกับสภาพของธุรกิจ เช่น หลักการสำรวจความเสี่ยง การระบุความเสี่ยง ตลอดจนวิธีวิเคราะห์ และแนวทางในการบรรเทาความเสี่ยงต่างๆ เพื่อใช้เป็นแนวทางในการบริหารสำหรับผู้ประกอบการในประเทศไทย

ภาคผนวก

ตารางที่ 1 ปัจจัยเสี่ยงในโครงการซอฟต์แวร์ขนาดเล็ก

ลำดับที่	กลุ่มปัจจัยเสี่ยง	ปัจจัยเสี่ยง	จำนวนคนที่ระบุ
1	บุคลากรในโครงการ		
1.1		ลูกค้ามีภาระกิจจากงานประจำและไม่ให้ความสำคัญในการร่วมทำโครงการ	15
1.2		บุคลากรภายในและภายนอก(ลูกค้า)ขาดความรู้และประสบการณ์ในการทำงาน	10
1.3		วัฒนธรรมภายในและภายนอก(ลูกค้า)	10
1.4		ความรับผิดชอบแต่ละคนไม่ชัดเจน	3
2	ลูกค้าเปลี่ยนความต้องการ		
2.1		ปัญหาความต้องการของผู้ใช้ไม่ชัดเจน	15
2.2		การเปลี่ยนความต้องการ เนื่องจากมีปัจจัยรอบข้างอื่นๆ	7
2.3		ไม่ค่อยให้ความสำคัญในขั้นตอนระบุความต้องการในช่วงสำรวจ	6
2.4		ขาดความชัดเจนในการเปลี่ยนแปลงและไม่มีการทำเอกสารเพื่อบันทึก	4
3	การบริหาร		
3.1		บุคลากรแต่ละคนมีหลายโครงการที่ต้องทำในเวลาเดียวกัน	14
3.2		ผู้นำโครงการในฝั่งลูกค้า ไม่กล้าตัดสินใจ	10
3.3		ผู้บริหารระดับสูงไม่ค่อยให้ความสำคัญ	4
3.4		เอกสารที่ต้องส่งมอบและการอบรมไม่สมบูรณ์	4
4	การประเมิน		
4.1		ประเมินงบประมาณที่ต้องใช้ผิดพลาด	7
4.2		ประเมินเวลาในการทำโครงการผิดพลาด	7
5	เทคโนโลยี		
5.1		เลือกใช้เทคโนโลยีหรือเครื่องมือไม่เหมาะสม	2

ตารางที่ 2 เปรียบเทียบปัจจัยเสี่ยงในโครงการขนาดเล็ก
กับโครงการขนาดใหญ่และของ Boehm

ลำดับที่	โครงการขนาดใหญ่	Boehm	โครงการขนาดเล็ก
1	สาเหตุเนื่องจากบุคลากรไม่เพียงพอ	X	X
2	การประมาณต้นทุนค่าใช้จ่ายและระยะเวลาในการดำเนินงานไม่ถูกต้อง	X	X
3	การคาดการณ์ที่เกินจริง	X	
4	การระบุความต้องการของผู้ใช้งานไม่ครบถ้วน	X	X
5	การเปลี่ยนแปลงความต้องการของผู้ใช้หรือลูกค้าอย่างต่อเนื่อง	X	X
6	ระบบการทำงานที่ไม่ดีพอทั้งด้านการบริหารและกระบวนการทำงาน		X
7	ผู้บริหารโครงการมีประสบการณ์ความสามารถไม่เพียงพอ		X
8	การออกแบบเอกสารที่ขาดความสมบูรณ์		X
9	ผู้ใช้งานหรือลูกค้าขาดการมีส่วนร่วมในการทดสอบและระหว่างการตกลงยอมรับ	X	X
10	ผู้รับงานย่อยหรือผู้รับเหมาช่วงไม่มีความสามารถเพียงพอ		
11	การเมืองในองค์กรซึ่งไม่มีส่วนเกี่ยวข้องกับความต้องการ เช่น การขอเพิ่มคุณประโยชน์บางอย่างซึ่งเกินความจำเป็นที่ตัวเองต้องใช้		
12	ไม่ค่อยได้รับการสนับสนุนหรือสนใจจากผู้บริหาร		X
13	ไม่มีผู้รับผิดชอบหลัก ในโครงการที่มีผู้มีส่วนร่วมจากหลายๆ ฝ่าย		X
14	วัฒนธรรมองค์กรไม่เอื้ออำนวย และบางครั้งมีการเปลี่ยนแปลงเกิดขึ้น		X
15	ข้อจำกัดของเทคโนโลยีที่จะทำให้บรรลุผล		X
16	การออกแบบส่วนที่ผู้ใช้ต้องใช้งาน ออกแบบไม่ดี	X	
17	ฟังก์ชันของซอฟต์แวร์ไม่ตรงกับความต้องการใช้งาน	X	
18	การเพิ่มเติมคุณสมบัติที่เกินความจำเป็น	X	
19	การขัดแย้งระหว่างลูกค้ากับคู่สัญญา		
20	การพัฒนาที่คิดไปจากงานของผู้ใช้งาน		X
21	คุณภาพการทำงานของพนักงานที่ไม่ดีหรือขาดความสามารถ	X	X

เอกสารอ้างอิง

- [1] Bain & Company and D. K.Rigby, "Management Tools 2007, An Executive's Guide," 2007.
- [2] M. E.Porter and V. E.Millar, "How information gives you competitive advantage," *Havard Business Review*, pp. 149-174, 1985.
- [3] N. Melville,K. L.Kraemer, and V. Gurbaxani, "Information Technology and Organizational Performance:An Integrative Model of IT Business Value," *Center for Research on Information Technology and Organization,University of California*, pp. 1-73, January 2004.
- [4] T. Addison and S. Vallabh, "Controlling Software Project Risks-an Empirical Study of Methods used by Experienced Project Managers.," *Management of Computing and Information Systems*, vol. 1, pp. 128-140, 2002.
- [5] D. Baccarini,G. Salm, and P. E. D. Love, "Management of risks in information technology projects," *Industrial Management & Data Systems*, vol. 104, pp. 286-295, 2004.
- [6] D. Tesch,T. J. Kloppenborg, and M. N. Frolick, "IT Project Risk Factors: The Project Management professionals Perspective," *The Journal of Computer Information Systems*, vol. 47, pp. 61-69, 2007.
- [7] B. Whittaker, "What went wrong? Unsuccessful information technology projects," *Information Management & Computer Security*, vol. 7, pp. 23-29, 1999.
- [8] SIPA,NECTEC, and ASTI, "Thailand Outlook ICT Market 2008," 2008.
- [9] B. W. Boehm, "Software Risk Management:Principles and Practices," *IEEE Software*, pp. 32-40, January 1991 1991.
- [10] C. Chapman and S. Ward, "Why risk efficiency is a key aspect of best practice projects," *International Journal of Project Management*, vol. 22, pp. 619-632, 2004.
- [11] E. M. Hall, "Managing Risk:Methods for Software Systems Development," pp. 1-400, 1998.
- [12] T. Suraweera,V. Pulakanam, and O. Guler, "Managing the Implementation of IT Project in SMEs:An Exploratory Investigation," *IEEE Software*, pp. 381-388, 2006.
- [13] U.S. Census Bureau, "2008 Capital Spending Report," 2008.
- [14] Standish Group, "Chaos(Application Project Failure and Success)," 2004.
- [15] U. S. Department Of Agriculture, "Information Technology System Development Life Cycle Guide," 2007.
- [16] Kayis,M.Zhou,S.Savci,Y.B.Khoo,A.Ahmad,R.Kusumo, and A.Rispler, "IRMAS - Development of a risk management tool for collaborative multi-site,multi-partner new product development projects," *Manufacturing Technology management*, vol. 18, pp. 387-414, May 2006 2006.
- [17] R. Olsson, "In search of opportunity management: Is the risk management process enough?," *International Journal of Project Management*, vol. 25, pp. 745-752, 2007.
- [18] D. Hillson, "Using a Risk Breakdown Structure in project management," *Journal of Facilities Management*, vol. 2, pp. 85-97, 13 March,2003 2003.

- [19] Australia Standard, *Risk Management AS/NZS 4360:1999*, 1999.
- [20] Department of Defense, *Risk Management Guide for DOD Acquisition*: Department of Defense, 2006.
- [21] IEEE Standard, *IEEE Standard for Software Life Cycle Process--Risk Management*: IEEE, 2001.
- [22] National Institute of Standards and Technology, *Risk Management Guide for Information Technology Systems*, 2002.
- [23] Project Management Institute, "PMBOK Guide (A Guide to the Project Management Body of Knowledge)," *Project Management*, pp. 237-264, 2004.
- [24] J. Ropponen and K. Lyytinen, "Components of software development risk: how to address them? A project manager survey," *IEEE Transactions on Software Engineering* vol. 26, pp. 98-112, 2000.
- [25] O. Zwikael and A. Sadeh, "Planning effort as an effective risk management tool," *Journal of Operations Management*, vol. 25, pp. 755-767, 2006.
- [26] H. Wen-Ming and H. Sun-Jen, "An empirical analysis of risk components and performance on software projects," *Journal of Systems and Software*, vol. 80, pp. 42-50, 2007.
- [27] Software Engineering Institute, Ronald P. Higuera, and Yacov Y. Haimes, "Software Risk Management (Technical Report)" *CMU/SEI-96-TR-012*, pp. 1-48, 1996.
- [28] R. James, J. Voetsch, and D. F. Cioffi, "Risk Management's Positive Effect on Projects," *Improving business performance*, vol. 5, pp. 1-7, 2004.
- [29] D. D. Hewson and K. Peters, "Fundamentals of Software Project Estimation," 2007.
- [30] B. Srinivasan and G. Martin, "MONSET-A Prototype Software Development Estimating Tool," *IEEE Explore*, vol. 3, pp. 70-82, 1994.
- [31] W. S. Humphrey, D. H. Kitson, and T. C. Kasse, "The State of Software Engineering Practice: A Preliminary Report," *Communication of the ACM* vol. 89, pp. 277-288, 1989.
- [32] A. Bryant and J. A. Kirkham, "B. W. Boehm software engineering economics: a review essay," *SIGSOFT Softw. Eng. Notes*, vol. 8, pp. 44-60, 1983.
- [33] B. James, "Improving Size Estimates Using Historical Data," *IEEE Software*, pp. 27-35, November 2000 2000.
- [34] Bala Srinivasan and G. Martin, "Monset - A Prototype Software Development Estimating Tool," *IEEE Software*, vol. 3, pp. 70-82, 1994.
- [35] B. W. Boehm, C. Abts, A. W. Brown, B. K. C. S. Chulani, E. Horowitz, R. Madachy, D. Reifer, and B. Steece, *Software Cost Estimation with COCOMO II*: Prentice Hall, 2000.
- [36] J. Ropponen and K. Lyytinen, "Components of software development risk: how to address them? A project manager survey," *IEEE Transactions on Software Engineering*, vol. 26, pp. 98-112, 2000.