

การจัดทำ Explicit Knowledge จากงานประจำ

ชื่อเรื่อง เทคนิคและวิธีการออกแบบ Activity Diagram เพื่อใช้ในการทดสอบระบบและนำเสนอต่อเจ้าของระบบ

ชื่อผู้เสนอ นางสาวเบญจวรรณ เกื้อกิจ ตำแหน่ง นักวิชาการคอมพิวเตอร์

สังกัด/ฝ่าย พัฒนาระบบ

๑. ภาระงานที่รับผิดชอบ

พัฒนาระบบสารสนเทศให้มหาวิทยาลัยบูรพา หน่วยงานภายในและหน่วยงานภายนอกมหาวิทยาลัย โดยมีรายละเอียดเกี่ยวกับหน้าที่รับผิดชอบ ดังนี้ สืบค้นความต้องการของผู้ใช้บริการ เขียนรายละเอียดการจัดทำโครงการ และแผนการดำเนินงาน วิเคราะห์ ออกแบบระบบงาน พัฒนา ทดสอบและแก้ไขโปรแกรม จัดทำคู่มือ และให้การฝึกอบรมที่พัฒนา ติดตามและประเมินผลการใช้งานระบบสารสนเทศจากผู้ใช้งาน เพื่อนำมาปรับปรุงแก้ไขระบบได้ดีขึ้น อีกทั้งให้คำแนะนำการใช้งาน คำปรึกษาแก่ผู้ใช้งาน ควบคุม ดูแล ปรับปรุงระบบงานให้สามารถใช้งานได้อย่างต่อเนื่องและมีประสิทธิภาพ และปฏิบัติงานอื่นตามที่ได้รับมอบหมาย

๒. การเรียนรู้ ความคาดหวัง ของ "ลูกค้า/ผู้ที่เกี่ยวข้องกับกระบวนการที่นำเสนอ"

ผู้ที่มีส่วนเกี่ยวข้องกับกระบวนการพัฒนาระบบมีดังนี้

๒.๑. เจ้าของระบบและผู้ใช้งานระบบ หมายถึง ส่วนงาน/หน่วยงานผู้ให้ข้อมูล สามารถเรียนรู้ขั้นตอนการทำงาน

ภาพรวม ความสัมพันธ์ที่เกี่ยวข้องในแต่ละกระบวนการ เพื่อให้เกิดความเข้าใจที่ตรงกันและเป็นแนวทางเดียวกันในการพัฒนาระบบ

๒.๒. ผู้พัฒนาระบบ หมายถึง นักพัฒนาระบบสารสนเทศของสำนักคอมพิวเตอร์ มหาวิทยาลัยบูรพา สามารถเรียนรู้ขั้นตอนการทำงานในภาพรวม ความสัมพันธ์ที่เกี่ยวข้องในแต่ละกระบวนการ รวมทั้งเงื่อนไขต่าง ๆ เพื่อทำมาใช้สำหรับการวิเคราะห์ ออกแบบ และพัฒนาระบบเพื่อให้ตรงตามความต้องการกับผู้ใช้งานมากที่สุด

๓. วัตถุประสงค์

๓.๑. เพื่อให้มีความรู้กับบุคลากรในฝ่ายให้สามารถปฏิบัติงานแทนกัน หรือนำไปประยุกต์ใช้กับงานของตนเองได้

๓.๒. เพื่อให้ผู้พัฒนาระบบพัฒนาระบบได้อย่างถูกต้อง แก้ไขปัญหาได้อย่างรวดเร็ว

๓.๓. เพื่อให้ผู้พัฒนาระบบไม่เกิดความสับสนภายในที่ระหว่างดำเนินการเขียนโปรแกรมและทดสอบระบบ

๓.๔. เพื่อให้ผู้พัฒนาระบบทดสอบระบบได้อย่างมีประสิทธิภาพ ครบถ้วนทุกกรณี

๓.๕. เพื่อให้ผู้พัฒนาระบบและเจ้าของระบบเห็นภาพรวมของระบบร่วมกันและเข้าใจกระบวนการทำงานของระบบ

๓.๖. เพื่อนำไปต่อยอดออกแบบเป็น Test Case, Unit Testing, Integration Testing

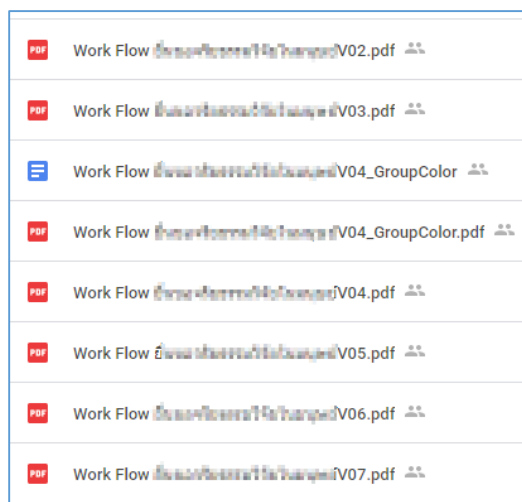
๔. บทสรุปองค์ความรู้

ในกระบวนการพัฒนาระบบสิ่งสำคัญคือ ต้องเข้าใจกระบวนการทำงานเดิมของผู้ใช้งานระบบและนำมาออกแบบเป็น Activity Diagram หรือ Workflow Diagram ให้ครบถ้วนและถูกต้องครบทุกกระบวนการ จากนั้นจะต้องมีการพูดคุยและประชุมเพื่อวิเคราะห์ว่าสิ่งที่ผู้พัฒนาระบบออกแบบมานั้นถูกต้องและสอดคล้องกับการทำงานของผู้ใช้งานหรือไม่ มีจุด

ไหนที่ต้องปรับแก้ ซึ่ง Activity Diagram เป็นสิ่งสำคัญที่จะช่วยให้ผู้เข้าร่วมประชุมเข้าใจภาพรวมของระบบได้ และเป็นสิ่งสำคัญที่ช่วยให้ทีมผู้พัฒนาระบบเขียนโปรแกรมได้อย่างถูกต้องและการทดสอบระบบได้อย่างถูกต้องอีกด้วย ซึ่งการทำ Activity Diagram มีเทคนิคและวิธีการออกแบบ ๕ ข้อ ดังนี้

๔.๑ ประชุมกับเจ้าของระบบและผู้ใช้งานอย่างต่อเนื่อง

เพื่อให้เข้าใจกระบวนการทำงานทั้งหมดได้อย่างถูกต้องชัดเจน ซึ่งอาจจะต้องประชุมกันมากกว่า ๑ ครั้ง หรือ ๑ ครั้ง/เดือน เพื่อติดตามและปรับปรุงกระบวนการอยู่ตลอดเวลา เนื่องจากแต่ละครั้งอาจจะมีผู้เข้าร่วมประชุมคนใหม่ ทำให้เกิดความต้องการใหม่ในการใช้งานหรือขอปรับเปลี่ยนกระบวนการทำงานเพิ่มเติม ดังนั้นในการประชุมแต่ละครั้งเมื่อมีการปรับแก้กระบวนการทำงาน จำเป็นจะต้องสรุปข้อมูลและแก้ไขกระบวนการงาน เมื่อแก้ไข Activity Diagram และเก็บเป็น version ตามครั้งที่ประชุมไว้เสมอ



รูปภาพที่ ๑ – แสดงตัวอย่างการเก็บ Activity Diagram ตาม version ครั้งที่ประชุม

๔.๒ ผู้พัฒนากำหนดเลขสถานะ (Status) ที่จะเกิดขึ้นในระบบให้มีความยืดหยุ่น พร้อมต่อการเปลี่ยนแปลง

โดยในการกำหนดสถานะนั้นมีเทคนิคคือ ไม่กำหนดเลขสถานะ (Status) ในกระบวนการที่ใกล้เคียงกันให้มีค่าต่างกันน้อยเกินไป ควรกำหนดเลขสถานะแบบเว้นช่วงซึ่งเป็นเทคนิคหนึ่งที่จะช่วยให้เมื่อมีการปรับเปลี่ยนความต้องการหรือมีการแทรกจากกระบวนการทำงานใหม่เข้ามานั้น จะทำให้สามารถเพิ่มเลขสถานะลงไปใหม่ได้เลยโดยไม่กระทบกับกระบวนการเดิมและสามารถแก้ไข Source Code ได้ง่ายและรวดเร็วมากขึ้น

ตัวอย่างการกำหนดเลขสถานะแบบเว้นช่วง เช่น

สถานะที่ ๓๕๐ คือ รอความคิดเห็นจากกรรมการ Exemption Determination

สถานะที่ ๓๖๐ คือ รอกรรมการแก้ไขผลการประเมิน

สถานะที่ ๓๗๐ คือ รอเปลี่ยนประเภทโครงการ

สถานะที่ ๓๘๐ คือ รอประธานลงนามรับรอง Exemption Determination

จะเห็นว่าสถานะ ๓๕๐, ๓๖๐, ๓๗๐, ๓๘๐ เป็นสถานะของกระบวนการทำงานที่ต่อเนื่องกันและมีการเว้นช่วงไว้ ๑๐ เพื่อให้รองรับกับการเปลี่ยนแปลงในกรณีที่เจ้าของระบบขอแทรกกระบวนการเพิ่ม

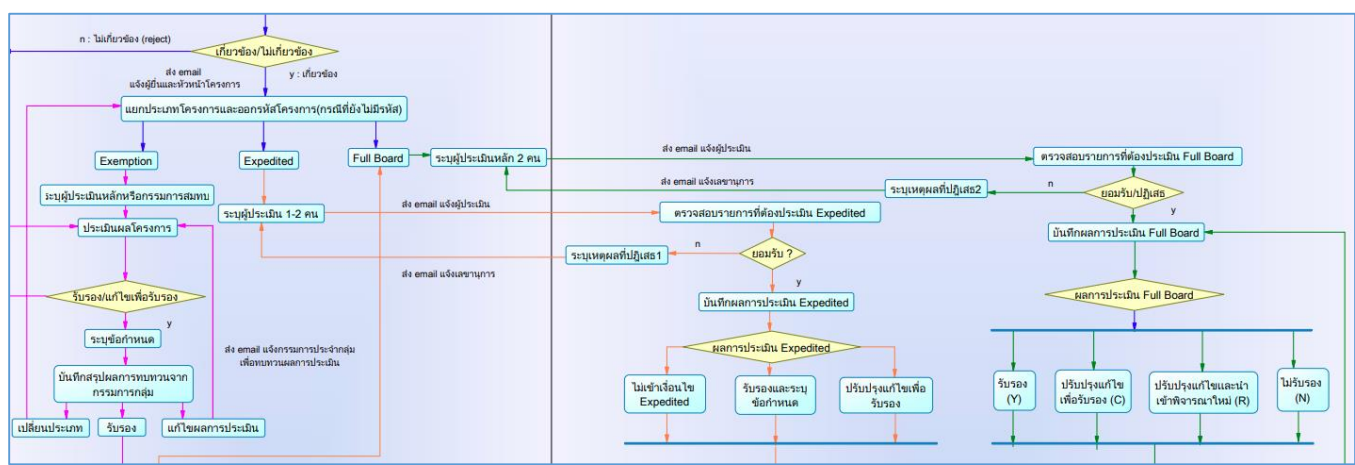
STATUS	STATUS_NAME
100	ดำเนินการกรอกข้อมูล
101	ยื่นแบบฟอร์ม และรอตรวจสอบเอกสาร
102	เอกสารไม่ครบ ดำเนินการแก้ไขเอกสาร
200	รอแยกประเภทโครงการ
201	ไม่เกี่ยวข้องกับกรอกรวิจัยในมนุษย์
202	ผ่านการลงนามจากประธานรับรองโครงการไม่เกี่ยวข้องกับกรอกรวิจัยในมนุษย์
203	รับรองโครงการไม่เกี่ยวข้องกับกรอกรวิจัยในมนุษย์
205	รอเปลี่ยนผู้ประเมินโครงการ
301	รอประเมินผลโครงการ Exemption Determination
303	แก้ไขข้อมูล Exemption Determination
350	รอความคิดเห็นจากกรรมการ Exemption Determination
360	รอกรรมการแก้ไขผลการประเมิน
370	รอเปลี่ยนประเภทโครงการ
390	รอประธานลงนามรับรอง Exemption Determination
401	รอประเมินผลโครงการ Expedited Review

รูปภาพที่ ๒ - แสดงตัวอย่างเลขสถานะของแต่ละกระบวนการทำงาน โดยมีการเว้นช่องว่างระหว่างเลขสถานะ

๔.๓ จัดทำ Activity Diagram/Workflow Diagram อธิบายการทำงานของระบบให้ชัดเจน

เพื่อให้ผู้เกี่ยวข้องทุกคนที่เข้าประชุมเห็นภาพรวมของระบบร่วมกันในขณะที่ประชุม ผู้พัฒนาระบบจะอธิบายขั้นตอนตาม Activity Diagram ไปตามลำดับ เพื่อง่ายต่อการนำเสนอ และสามารถชี้จุดที่จะต้องแก้ไขได้ถูกต้อง ในกรณีที่มีผู้ให้คำแนะนำหรือต้องการปรับเปลี่ยนขั้นตอนการทำงาน จะทำให้เราแก้ไขกระบวนการทำงานไม่เกิดความสับสนและเสียเวลาในขณะพูดคุยและหาข้อสรุปในที่ประชุม ซึ่งการจัดทำ Activity Diagram ให้ผู้เข้าร่วมประชุมอ่านได้ง่าย มีเทคนิคดังนี้

1. ใส่สีบนเส้นเพื่อแยกกระบวนการหลักออกให้เห็นอย่างชัดเจน
2. ใส่สีกล่องชื่อกิจกรรม (Activity Name)
3. ใส่ข้อความอธิบายในกระบวนการที่สำคัญ



รูปภาพที่ ๓ - แสดงตัวอย่างการใส่สีบนเส้นเพื่อแยกประเภทและการเพิ่มข้อความลงบน Activity Diagram



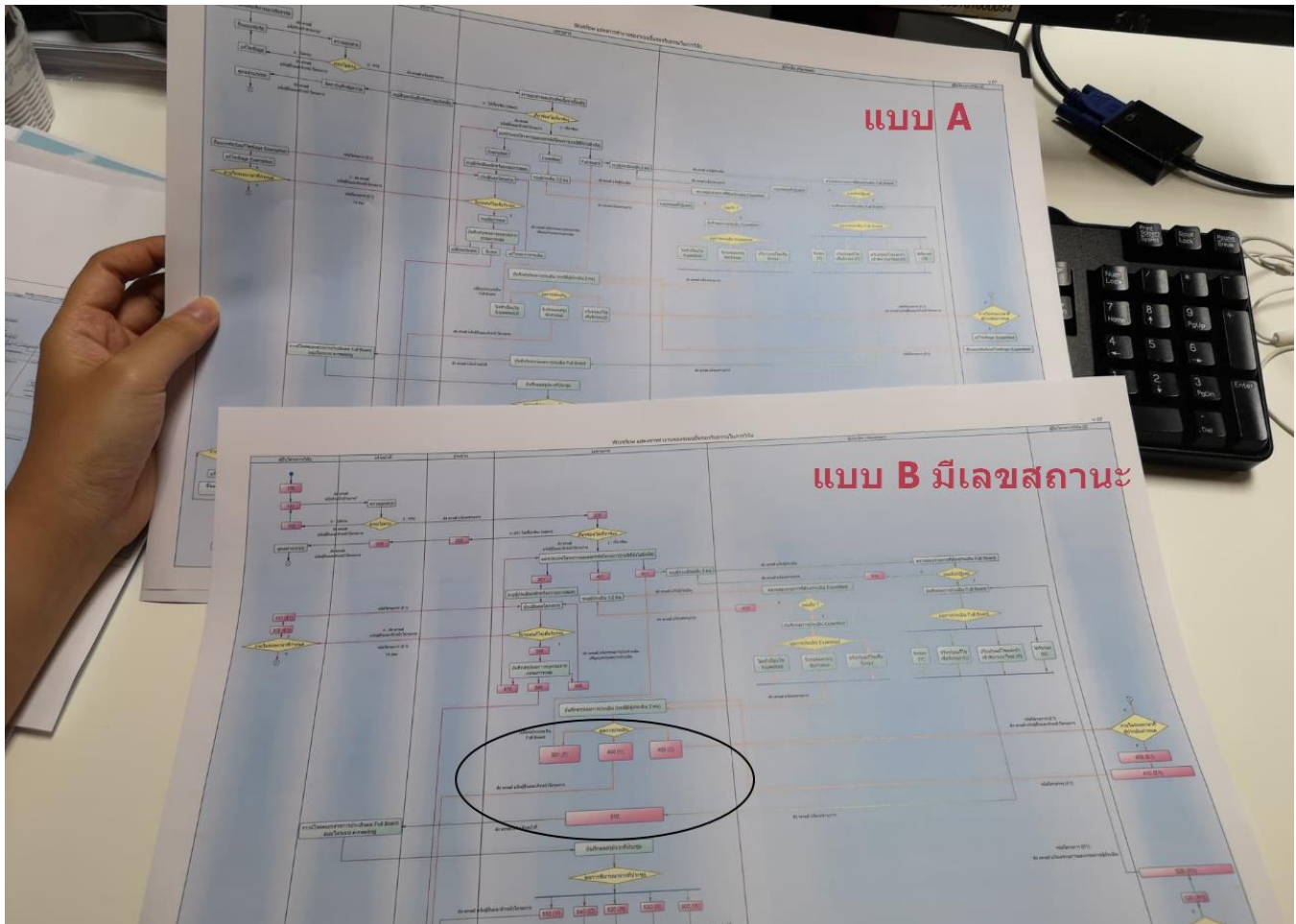
รูปภาพที่ ๔ - แสดงการประชุมโดยใช้ Activity Diagram อธิบายการทำงานให้กับผู้เข้าร่วมประชุมเห็นภาพรวมของระบบ



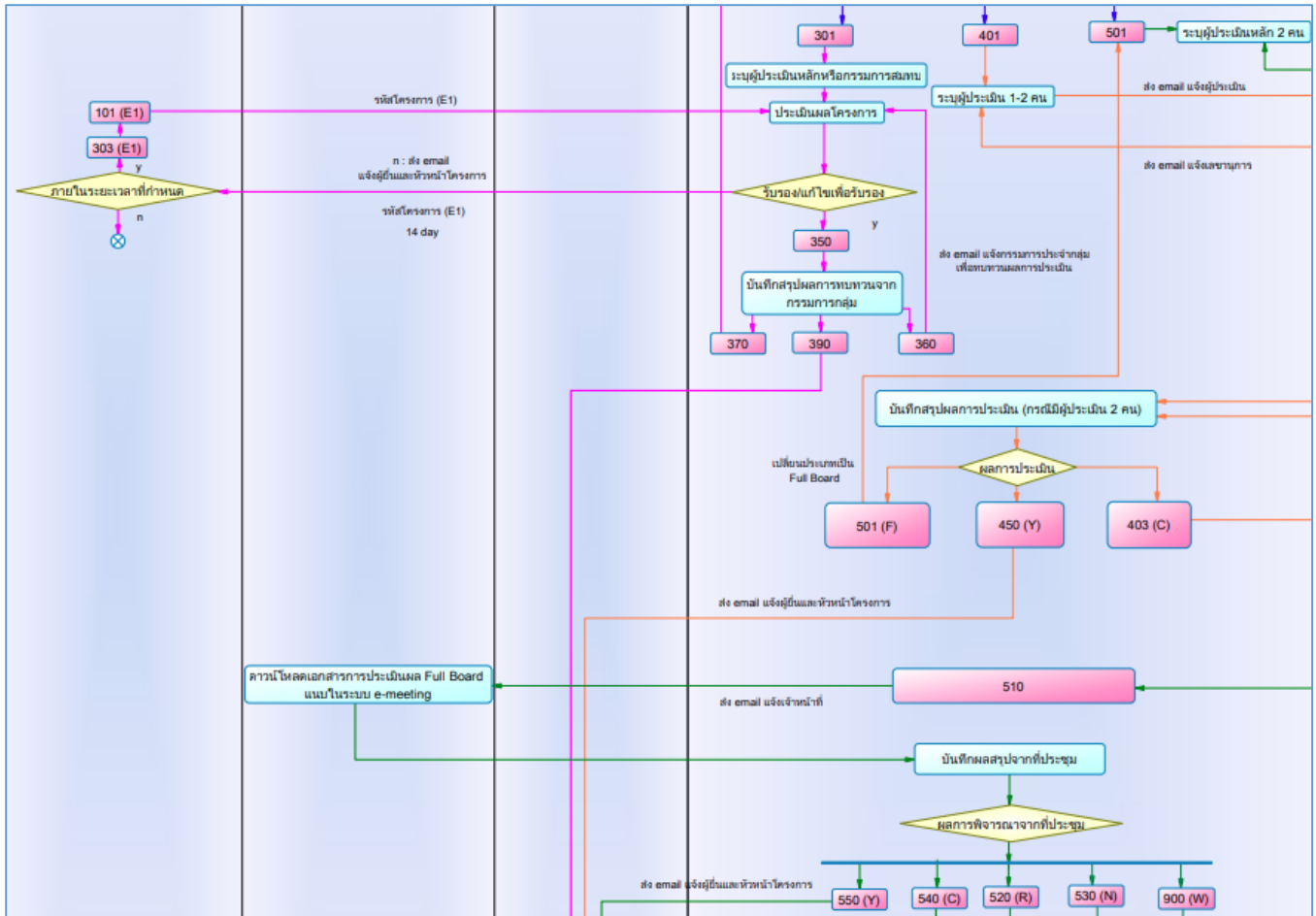
รูปภาพที่ ๕ - แสดงการประชุมโดยใช้ Activity Diagram โดยมีผู้เข้าร่วมประชุมตรวจสอบจุดที่มีข้อผิดพลาด

๔.๔ จัดทำ Activity Diagram/Workflow Diagram สำหรับทีมผู้พัฒนาเพื่อทดสอบสถานะการทำงานของระบบทั้งหมดให้ครบทุกกรณี

เป็นเทคนิคในการทดสอบระบบที่มีสถานะจำนวนมากและมีการติดตามผล (Tracking) จากผู้ใช้งานแบบ Real Time โดยผู้พัฒนาระบบจะจัดทำ Activity Diagram ที่มีเลขสถานะในแต่ละกระบวนการที่เกิดขึ้นในระบบแยกออกมาอีก ๑ แบบ เพื่อให้ง่ายต่อการทดสอบและเพื่อให้สมาชิกในทีมทดสอบได้อย่างถูกต้อง ไม่เกิดความสับสนภายในทีม สามารถตรวจสอบสถานะที่เกิดขึ้นโดยเปรียบเทียบกับฐานข้อมูลได้ง่าย หากเกิดข้อผิดพลาดสามารถแก้ไขได้ตรงจุด และไม่เสียเวลาในการแก้ไข Source Code หรือต้องเปิดไฟล์ Source Code ระหว่างการทดสอบ



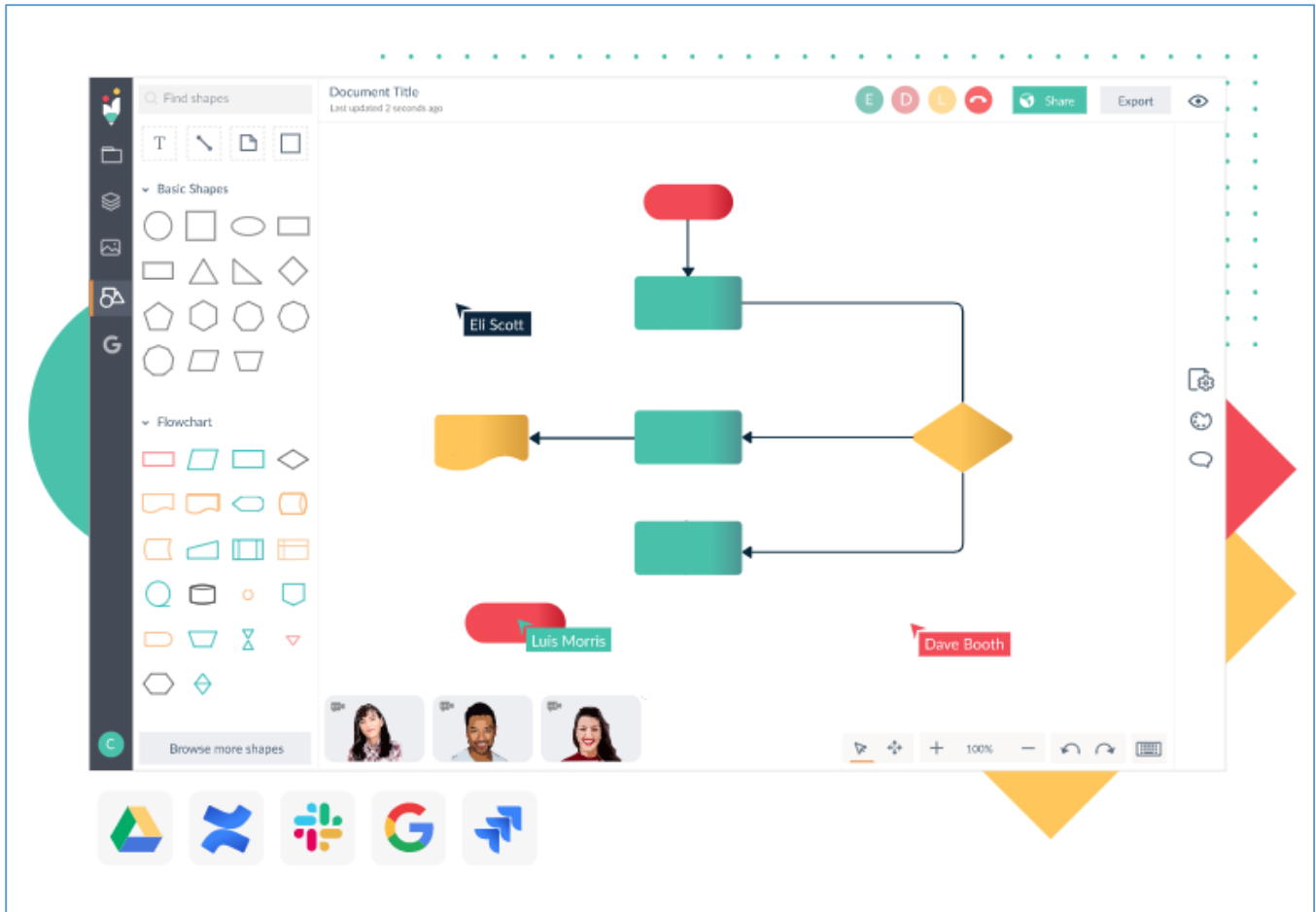
รูปภาพ ๖ - แสดง Activity Diagram แบบ A สำหรับผู้ที่มีส่วนเกี่ยวข้องและแบบ B สำหรับทดสอบระบบ



รูปภาพ ๗ - แสดง Activity Diagram เพื่อใช้ในการทดสอบระบบและเปรียบเทียบสถานะที่เกิดขึ้นในฐานข้อมูล

๔.๕ สามารถสร้าง Activity Diagram/Workflow Diagram ออนไลน์ได้ฟรี

ปัจจุบันเราสามารถสร้าง Diagram ต่าง ๆ บนเว็บไซต์ออนไลน์และดาวน์โหลดหรือแชร์ให้ทีมพัฒนาระบบ ช่วยดูแลและแก้ไขร่วมกันได้และมีวิธีการไม่ยุ่งยาก ง่ายต่อการใช้งาน เช่น www.diagrameditor.com , www.glimfy.com , creately.com , app.diagrams.net เป็นต้น



รูปภาพ ๘ - แสดงตัวอย่างเว็บไซต์ที่สามารถสร้าง Diagram ต่างๆได้ และสามารถแชร์ให้กับผู้พัฒนาภายในทีมเข้าไปได้

๕. ประโยชน์ที่ได้รับ

- ๕.๑. บุคลากรในฝ่ายสามารถปฏิบัติงานแทนกัน หรือนำไปประยุกต์ใช้กับงานของตนเองได้
- ๕.๒. ผู้พัฒนาระบบและเจ้าของระบบเห็นภาพรวมของระบบร่วมกันและเข้าใจกระบวนการทำงานของระบบ
- ๕.๓. ผู้พัฒนาระบบสามารถพัฒนาระบบได้อย่างถูกต้อง แก้ไขปัญหาได้อย่างรวดเร็ว และไม่เกิดความสับสนภายในทีมระหว่างดำเนินการเขียนโปรแกรมและทดสอบระบบ ทดสอบระบบได้อย่างมีประสิทธิภาพ ครบถ้วนทุกกรณี และนำไปต่อยอดออกแบบเป็น Test Case, Unit Testing, Integration Testing ต่อไปได้ง่าย