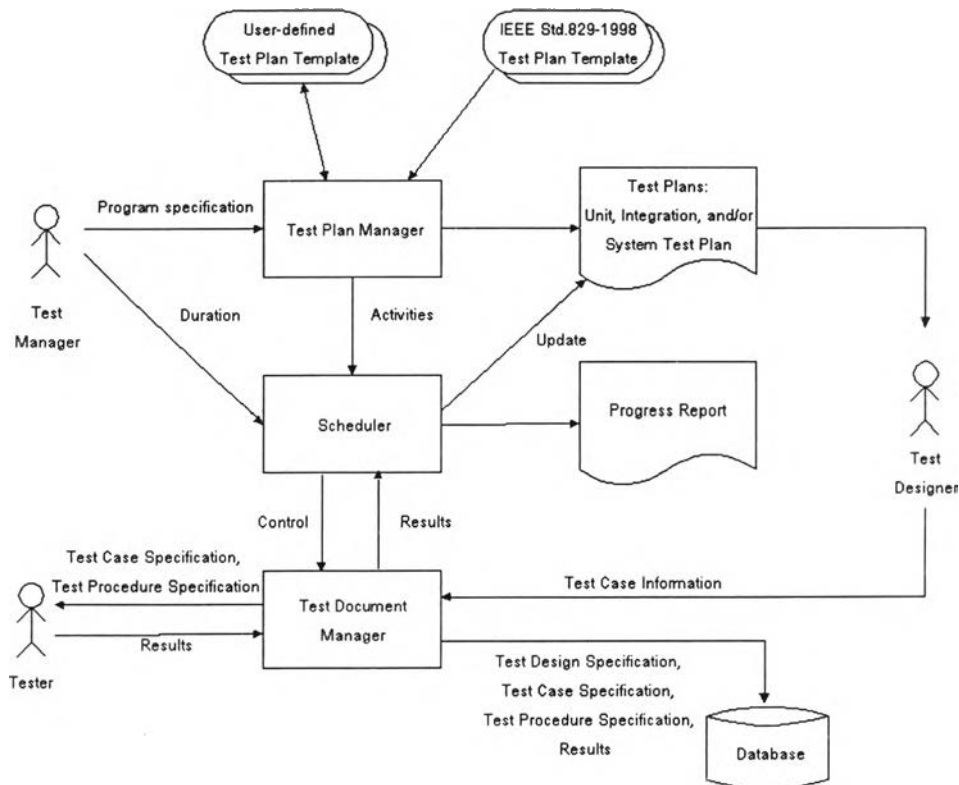


บทที่ 4

การพัฒนาเครื่องมือซอฟต์แวร์สำหรับจัดการกระบวนการวางแผนการทดสอบซอฟต์แวร์

4.1 องค์ประกอบหลักของเครื่องมือ

จะเห็นว่าในการวางแผนการทดสอบซอฟต์แวร์ในแต่ละระดับมีขั้นตอนที่ต้องทำหลายขั้นตอน ซึ่งควรมีการจัดกำหนดการของแต่ละขั้นตอนให้เป็นระเบียบ นอกจากนี้ยังเกิดเอกสารที่เกี่ยวข้องกับการทดสอบเป็นจำนวนมาก โดยเฉพาะเอกสารที่เกี่ยวกับกรณีทดสอบ ซึ่งจำเป็นต้องมีการจัดการที่ดี เพื่อให้สามารถควบคุมการทำงานในแต่ละขั้นตอนได้ ดังนั้นผู้วิจัยจึงได้ออกแบบเครื่องมือเพื่อจัดการกับความ ต้องการเหล่านี้ สามารถแสดงองค์ประกอบหลักได้ดังรูปที่ 4.1



รูปที่ 4.1 องค์ประกอบหลักของเครื่องมือ

ตัวจัดการสร้างแผนการทดสอบ จะทำตามขั้นตอนในการสร้างแผนการทดสอบทั้ง 3 ระดับ โดยรับข้อมูลแผนภูมิโครงสร้างในรูปแบบของแฟ้มข้อความ (Text file) เพื่อนำมาสร้างเป็นกิจกรรมต่าง ๆ และสร้างแผนการทดสอบที่อิงกับ IEEE Std.829-1998 หรือตามแม่แบบที่ผู้ใช้ต้องการได้ ตัวจัดการกำหนดการจะใช้จัดกำหนดการทดสอบโดยใช้เทคนิคของแผนภูมิแกนต์ และตัวจัดการเอกสารเกี่ยวกับการทดสอบ จะจัดการเอกสารทั้งหมดที่เกิดขึ้นระหว่างการทดสอบ รายละเอียดสำหรับแต่ละองค์ประกอบอธิบายได้ดังนี้

4.1.1 ตัวจัดการสร้างแผนการทดสอบ (Test Plan Manager)

เป็นส่วนที่สร้างแผนการทดสอบในระดับต่าง ๆ ได้แก่ การทดสอบระดับหน่วย การทดสอบการรวม และการทดสอบระบบ โดยมีรายละเอียดดังนี้

- ผู้ใช้สามารถเลือกแม่แบบของแผนการทดสอบมาใช้โดยอิงกับมาตรฐาน IEEE Std.829-1998 หรือสามารถสร้างแม่แบบขึ้นมาเองได้
- จากแม่แบบดังกล่าว ผู้ใช้สามารถบันทึกรายละเอียดต่าง ๆ ที่มีในแผนการทดสอบตามมาตรฐาน IEEE Std.829-1998 ดังที่ระบุไว้ในภาคผนวก ก. ได้แก่ ลักษณะที่จะทดสอบ แนวทางที่ใช้ เกณฑ์ในการพิจารณาว่าผ่านหรือไม่ผ่าน เป็นต้น ซึ่งมีรายละเอียดดังนี้
 - หากเป็นการทดสอบระดับหน่วย ผู้ใช้ต้องระบุโมดูลที่มีทั้งหมดในโปรแกรม เพื่อนำมาสร้างเป็นกิจกรรมสำหรับทดสอบทุก ๆ โมดูล นอกจากนั้นยังต้องระบุความต้องการอื่น ๆ ที่มีในการทดสอบระดับหน่วย
 - หากเป็นการทดสอบการรวม ผู้ใช้จะต้องระบุโมดูลที่มีทั้งหมดในโปรแกรม เพื่อนำมาสร้างเป็นลำดับการรวม โมดูล และระบุความต้องการอื่น ๆ ที่มีในการทดสอบการรวม
 - หากเป็นการทดสอบระบบ ผู้ใช้จะต้องระบุความต้องการตามหน้าที่ (Functional Requirements) ที่มีทั้งหมดเพื่อนำมาเชื่อมโยงกับโมดูลในโปรแกรม
- เครื่องมือได้กำหนดรูปแบบของแผนภูมิโครงสร้างที่จะนำมาใช้ระบุจำนวนโมดูลและความสัมพันธ์ระหว่างโมดูล โดยให้บรรจุอยู่ในแฟ้มข้อมูล สามารถดูรายละเอียดในภาคผนวก ข.
- หลังจากนั้น เครื่องมือสามารถกำหนดกิจกรรมเพื่อดำเนินการทดสอบได้ โดยหากเป็นการทดสอบระดับหน่วย เครื่องมือจะสร้างกิจกรรมพร้อมทั้งระบุตัวขับและตัวดำเนินการให้ หากเป็นการทดสอบการรวม เครื่องมือจะสร้างลำดับของการรวมพร้อมระบุตัวขับและตัวดำเนินการให้จากวิธีการรวมที่ผู้ใช้ได้กำหนด และหากเป็นการทดสอบระบบ ผู้ใช้จะต้องเชื่อมโยงความสัมพันธ์ระหว่างความต้องการแต่ละข้อกับโมดูลทั้งหมดที่มี เพื่อใช้เป็นแนวทางในการสร้างกรณีทดสอบพร้อมทั้งตัวขับและตัวดำเนินการ
- เครื่องมือจะให้ผู้ใช้แบ่งกิจกรรมที่มีความซับซ้อนออกเป็นกิจกรรมย่อย เพื่อให้การออกแบบการทดสอบสำหรับแต่ละกิจกรรมทำได้ง่าย
- ผู้ใช้จะต้องระบุทรัพยากรที่ต้องใช้ในการทดสอบสำหรับแต่ละกิจกรรม อันได้แก่ ฮาร์ดแวร์ ซอฟต์แวร์ และผู้ทดสอบ รวมทั้งกำหนดวันเริ่มต้น ระยะเวลาในการทดสอบสำหรับแต่ละกิจกรรมและความเกี่ยวข้องกับกิจกรรมอื่น โดยเครื่องมือจะแสดงอยู่ในรูปของแผนภูมิแกนต์ เพื่อแสดงความสัมพันธ์ได้ชัดเจน
- ในขั้นตอนนี้ เครื่องมือสามารถสร้างแผนการทดสอบที่ระดับต่าง ๆ ตามที่ผู้ใช้ได้ระบุแม่แบบไว้ และกำหนดการของกิจกรรมในการทดสอบที่ระดับนั้น

4.1.2 ตัวจัดกำหนดการ (Scheduler)

เป็นส่วนที่ช่วยในการจัดกำหนดการสำหรับแต่ละกิจกรรมที่ได้ระบุไว้ที่ตัวจัดการสร้างแผนการทดสอบ มีรายละเอียดดังนี้

- เครื่องมือสามารถให้ผู้ใช้สามารถปรับแต่งกำหนดการได้หากมีกิจกรรมที่สามารถดำเนินการพร้อมกันได้ เพื่อให้เกิดความยืดหยุ่นในการทดสอบ
- เครื่องมือสามารถตรวจสอบความก้าวหน้าในการทำกิจกรรมให้ผู้ใช้ได้โดยอัตโนมัติ หากเป็นกิจกรรมที่เครื่องมือกำหนดให้ จะสามารถตรวจสอบได้จากเอกสารที่เกิดขึ้นภายหลังจากที่ได้ทำกิจกรรมนั้นเสร็จ แต่หากเป็นกิจกรรมที่ผู้ใช้สร้างขึ้น ผู้ใช้จะเป็นผู้กำหนดความก้าวหน้าเอง
- เครื่องมือสามารถออกรายงานเกี่ยวกับการทำกิจกรรมได้ ได้แก่ รายงานแสดงกำหนดการที่ได้วางแผนไว้ รายงานแสดงความก้าวหน้าของแต่ละกิจกรรม รายงานแสดงกิจกรรมที่เลขกำหนด รายงานสรุปผลการดำเนินการกิจกรรม และรายงานแสดงกิจกรรมที่ต้องทำแบ่งตามผู้ทดสอบ

4.1.3 ตัวจัดการเอกสารเกี่ยวกับการทดสอบ (Test Document Manager)

เป็นส่วนที่ใช้ในการจัดการเอกสารที่จะเกิดขึ้นในการทดสอบหลังจากที่ได้วางแผนไว้ โดยในแต่ละกิจกรรม ผู้ใช้จะเป็นผู้ออกแบบแนวทางที่จะทดสอบ รวมทั้งสร้างกรณีทดสอบที่เกี่ยวข้องกับกิจกรรมนั้น ซึ่งเครื่องมือสามารถเชื่อมโยงกับระบบจัดการกรณีทดสอบ (TCMS : Test Case Management System) ที่สามารถช่วยผู้ใช้ในการสร้างกรณีทดสอบโดยอัตโนมัติได้ เพื่อลดขั้นตอนในการจัดการข้อมูลที่มีปริมาณมาก และช่วยในการบันทึกผลการทดสอบและข้อผิดพลาดที่เกิดขึ้น

เมื่อได้องค์ประกอบหลักของเครื่องมือแล้ว ขั้นตอนต่อไปจะเป็นการออกแบบโครงสร้างของเครื่องมือ โดยเครื่องมือจะทำงานในเชิงเหตุการณ์ (Event-Driven) สามารถแสดงความสัมพันธ์ระหว่างหน้าจอได้ดังรูปที่ 4.2 รายละเอียดของแต่ละหน้าจอสามารถดูได้จากคู่มือการใช้งานเครื่องมือในภาคผนวก ฉ

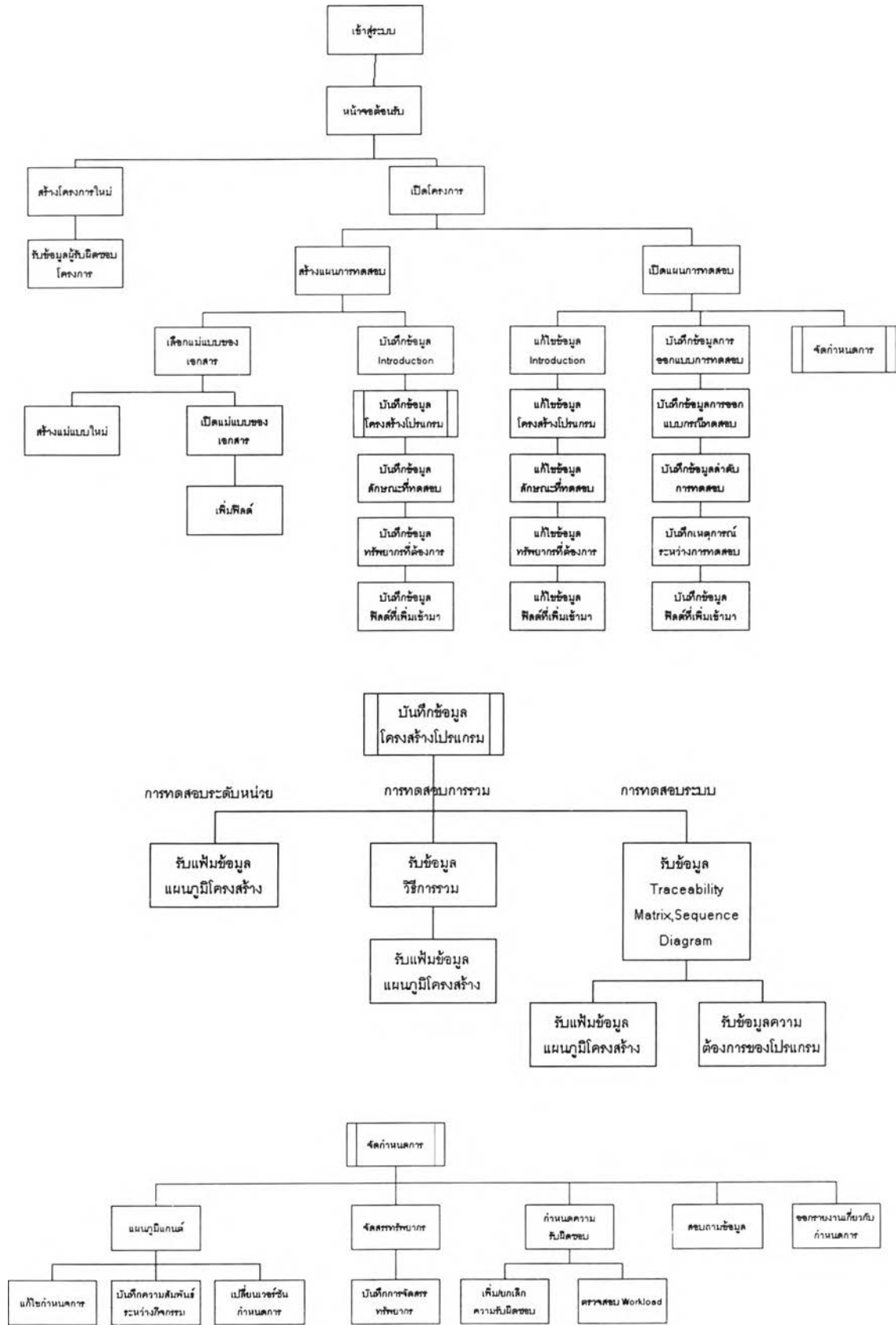
4.2 รายละเอียดของส่วนต่าง ๆ ในองค์ประกอบหลัก

เมื่อกำหนดองค์ประกอบหลักของเครื่องมือแล้ว ในขั้นตอนต่อไปจะเป็นการกำหนดส่วนต่าง ๆ เพื่อนำมาประกอบกันให้ได้เป็นองค์ประกอบหลักตามต้องการ ซึ่งสามารถแสดงรายละเอียดดังนี้

4.2.1 ตัวจัดสร้างแผนการทดสอบ

4.2.1.1 ส่วนจัดการแม่แบบของเอกสาร

เป็นส่วนที่ใช้เพิ่ม แก้ไข ลบ นำเข้า (import) และส่งออก (export) แม่แบบของเอกสารการทดสอบซอฟต์แวร์ในระบบ มีรายละเอียดดังนี้



รูปที่ 4.2 ความสัมพันธ์ระหว่างหน้าจอของเครื่องมือ

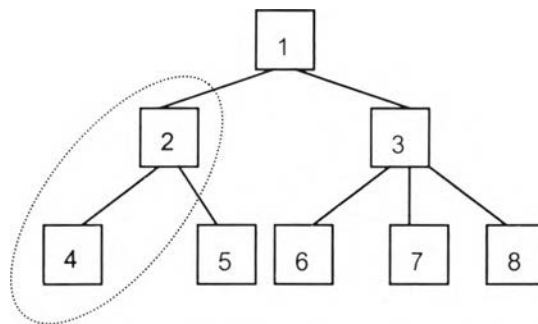
- การเพิ่ม ผู้ใช้สามารถสร้างแม่แบบของเอกสารขึ้นมาใหม่ เพื่อนำมาใช้กับแผนการทดสอบนั้น โดยเมื่อเริ่มสร้าง จะมีฟิลด์ของข้อมูลสอดคล้องกับมาตรฐาน IEEE Std.829-1998
- การแก้ไข ผู้ใช้สามารถแก้ไขฟิลด์ของข้อมูลในเอกสารได้โดยเลือกใช้เฉพาะฟิลด์ที่ต้องการ หรือสร้างฟิลด์ขึ้นมาใหม่ได้
- การลบ ผู้ใช้สามารถลบแม่แบบของเอกสารที่ไม่ได้ถูกใช้แล้วออกจากระบบได้
- การส่งออก ผู้ใช้สามารถบันทึกแม่แบบของเอกสารให้อยู่ในรูปของเท็กซ์ไฟล์ (Text file) เพื่อนำไปใช้กับโครงการอื่น ๆ ได้
- การนำเข้า ผู้ใช้สามารถนำเข้าแม่แบบของเอกสารในรูปของเท็กซ์ไฟล์ได้

4.2.1.2 ส่วนรับข้อมูลแผนภาพ โครงสร้างและกำหนดกิจกรรม

เป็นส่วนที่ใช้รับข้อมูลแผนภาพโครงสร้างในรูปของเท็กซ์ไฟล์และกำหนดเป็นกิจกรรมต่าง ๆ เพื่อให้การทดสอบนั้นครอบคลุมทุก ๆ โมดูลของโปรแกรมและสอดคล้องกับวิธีที่ผู้ใช้เลือก โดยปกติแล้วแผนภาพโครงสร้างมักจะถูกสร้างจากแผนภาพกระแสข้อมูล ดังนั้น เครื่องมือจึงถูกออกแบบให้รับข้อมูลแผนภาพโครงสร้างซึ่งถูกสร้างขึ้นจากโปรแกรมแปลงแผนภาพกระแสข้อมูลให้เป็นแผนภาพโครงสร้าง [10] มีรูปแบบดังในภาคผนวก ข เพื่อให้เกิดความต่อเนื่อง หากผู้ใช้ต้องการระบุแผนภาพโครงสร้างเองก็สามารถทำได้โดยเครื่องมือได้กำหนดรูปแบบไว้ในภาคผนวก ค ซึ่งจะง่ายและไม่ซับซ้อน

4.2.1.3 ส่วนกำหนดตัวขับและตัวดำเนินการ

เมื่อเครื่องมือได้ตรวจสอบแผนภาพโครงสร้างแล้ว จะนำมาวิเคราะห์เป็นกิจกรรมสำหรับการทดสอบที่ระดับนั้น พร้อมทั้งระบุตัวขับและตัวดำเนินการ ดังตัวอย่างแผนภูมิโครงสร้างในรูปที่ 4.2



รูปที่ 4.3 ตัวอย่างแผนภูมิโครงสร้าง

หากต้องการรวม โมดูล 2 และ 4 จะต้องใช้ตัวขับและตัวดำเนินการดังแสดงในตารางที่ 4.1

ตารางที่ 4.1 ผลลัพธ์จากการหาตัวขับและตัวดำเนินการ

โมดูล	ตัวขับ (Driver)	ตัวดำเนินการ (Stub)
2	1	4,5
4	2	-
2,4	1	5

4.2.1.4 ส่วนวิเคราะห์ลำดับการรวมสำหรับการทดสอบการรวม

จากที่ได้กล่าวมาแล้วข้างต้น หากเป็นการทดสอบระดับหน่วย สามารถนำแต่ละโมดูลมากำหนด เป็นกิจกรรมที่สอดคล้องกับโมดูลนั้นได้ทันที โดยที่แต่ละกิจกรรมนั้นไม่ขึ้นต่อกัน สามารถทำการ ทดสอบแยกกันได้

หากเป็นการทดสอบการรวม กิจกรรมจะถูกกำหนดขึ้นจากลำดับการรวมที่เกิดขึ้น โดยใน ปัจจุบันมีแนวความคิดเกี่ยวกับการรวมโมดูลเกิดขึ้นหลายวิธี ซึ่งเครื่องมือได้เลือกวิธีที่นิยมใช้ด้วยกัน 6 วิธี ได้แก่

1) การรวมจากบนลงล่าง แบ่งเป็น

1.1) ไปทางแนวลึกก่อน

จากตัวอย่างในรูปที่ 4.2 สามารถสร้างเป็นกิจกรรมได้ดังตารางที่ 4.2 สังเกตว่าการรวมโมดูลจาก บนลงล่างจะไม่จำเป็นต้องใช้ตัวจับเลย

ตารางที่ 4.2 กิจกรรมที่ได้จากวิธีการรวมจากบนลงล่างแบบไปทางแนวลึกก่อน

กิจกรรม	รายละเอียด	ตัวจับ	ตัวดำเนินการ
B1	1 + 2	-	4,5
B2	B1 + 4	-	5
B3	B2 + 5	-	-
B4	B3 + 3	-	6,7,8
B5	B4 + 6	-	7,8
B6	B5 + 7	-	8
B7	B6 + 8	-	-

1.2) ไปทางแนวกว้างก่อน

จากตัวอย่างในรูปที่ 4.2 สามารถสร้างเป็นกิจกรรมได้ดังตารางที่ 4.3

ตารางที่ 4.3 กิจกรรมที่ได้จากวิธีการรวมจากบนลงล่างแบบไปทางแนวกว้างก่อน

กิจกรรม	รายละเอียด	ตัวจับ	ตัวดำเนินการ
B1	1 + 2	-	4,5
B2	B1 + 3	-	4,5,6,7,8
B3	B2 + 4	-	5,6,7,8
B4	B3 + 5	-	6,7,8
B5	B4 + 6	-	7,8
B6	B5 + 7	-	8
B7	B6 + 8	-	-

2) การรวมจากล่างขึ้นบน แบ่งเป็น

2.1) แบบจำกัด (Strictly)

จากตัวอย่างในรูปที่ 4.2 สามารถสร้างเป็นกิจกรรมได้ดังตารางที่ 4.4
ตารางที่ 4.4 กิจกรรมที่ได้จากวิธีการรวมจากล่างขึ้นบนแบบจำกัด

กิจกรรม	รายละเอียด	ตัวขับ	ตัวดำเนินการ
B1	8 + 7	3	-
B2	B1 + 6	3	-
B3	B2 + 5	2,3	-
B4	B3 + 4	2,3	-
B5	B4 + 3	1,2	-
B6	B5 + 2	1	-
B7	B6 + 1	-	-

2.2) แบบบิลด์ (Builds)

จากตัวอย่างในรูปที่ 4.2 สามารถสร้างเป็นกิจกรรมได้ดังตารางที่ 4.5 สังเกตว่าจากตัวอย่างจะประกอบด้วย 2 บิลด์ คือ บิลด์ที่มีโมดูลที่ 2 เป็นโมดูลแม่ และบิลด์ที่มีโมดูล 4 เป็นโมดูลแม่ ซึ่งสามารถรวมบิลด์ทั้งสองนี้ได้พร้อม ๆ กันหากมีทรัพยากรเพียงพอ

ตารางที่ 4.5 กิจกรรมที่ได้จากวิธีการรวมจากล่างขึ้นบนแบบบิลด์

กิจกรรม	รายละเอียด	ตัวขับ	ตัวดำเนินการ
B1	2 + 4	1	-
B2	B1 + 5	1	-
B3	B2 + 1	-	-
B4	3 + 6	1	-
B5	B4 + 7	1	-
B6	B5 + 8	1	-
B7	B3 + B6	-	-

3) การรวมแบบแซนด์วิช

จากตัวอย่างในรูปที่ 4.2 สามารถสร้างเป็นกิจกรรมได้ดังตารางที่ 4.6 โดยแบ่งโมดูลออกเป็น 2 ระดับ ระดับบนประกอบด้วยโมดูล 1 เพียงโมดูลเดียว และระดับล่างประกอบด้วย 7 โมดูลที่เหลือ

ตารางที่ 4.6 กิจกรรมที่ได้จากวิธีการรวมแบบแซนควิช

กิจกรรม	รายละเอียด	ตัวจับ	ตัวดำเนินการ
B1	1	-	2,3
B2	2 + 4	1	-
B3	B2 + 5	1	-
B4	B1 + B3	-	-
B5	3 + 6	1	-
B6	B5 + 7	1	-
B7	B6 + 8	1	-
B8	B4 + B7	-	-

4) การรวมโมดูลวิกฤต

จากตัวอย่างในรูปที่ 4.2 สามารถสร้างเป็นกิจกรรมได้ดังตารางที่ 4.7 หากโมดูลวิกฤตคือ โมดูล 3 และ 4

ตารางที่ 4.7 กิจกรรมที่ได้จากการรวมแบบโมดูลวิกฤต

กิจกรรม	รายละเอียด	ตัวจับ	ตัวดำเนินการ
B1	3 + 6	1	-
B2	B1 + 7	1	-
B3	B2 + 8	1	-
B4	2 + 4	1	-
B5	B4 + 5	1	-
B6	B5 + 1	-	-
B7	B6 + B3	-	-

นอกจากนี้ เครื่องมือยังสามารถให้ผู้ใช้ปรับแต่งวิธีการรวมได้ตามต้องการ โดยเครื่องมือจะระบุตัวจับและตัวดำเนินการให้ ซึ่งรายละเอียดของการปรับแต่งอยู่ที่คู่มือการใช้งานซึ่งอยู่ในภาคผนวก จ

4.2.1.5 ส่วนวิเคราะห์ความสัมพันธ์ระหว่างความต้องการกับส่วนของโปรแกรมสำหรับการทดสอบระบบ

ในการทดสอบระบบ มีจุดมุ่งหมายหลักเพื่อให้แน่ใจว่า โปรแกรมที่พัฒนาขึ้นมีความสามารถสอดคล้องกับความต้องการของผู้ใช้ได้ ดังนั้น กิจกรรมของการทดสอบในระดับนี้จะถูกสร้างขึ้นมาจากความต้องการของผู้ใช้ โดยเครื่องมือจะรับข้อมูลความต้องการของผู้ใช้ทั้งหมดในการทดสอบระบบ จาก

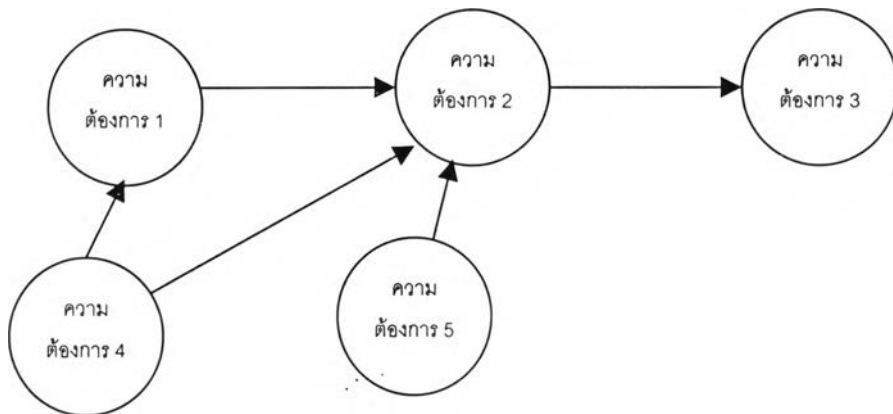
นั่นจึงนำมาเชื่อมโยงกับโมดูลทั้งหมด สามารถสร้างเป็นตารางที่แสดงการเชื่อมโยงนี้ ซึ่งเรียกว่าเมตริกซ์ที่ติดตามได้ มีรูปแบบดังตารางที่ 4.8

ตารางที่ 4.8 ตัวอย่างของเมตริกซ์ที่ติดตามได้

	โมดูล #1	โมดูล #2	โมดูล #3	โมดูล #4	...	โมดูล #n
ความต้องการ #1	✓	✓				
ความต้องการ #2		✓	✓			
ความต้องการ #3			✓			
...						
ความต้องการ #n	✓					✓

สำหรับความต้องการหนึ่ง ๆ นั้น จะมีโมดูลที่เกี่ยวข้องอยู่ 1 หรือมากกว่าได้ และในทำนองเดียวกัน สำหรับโมดูลหนึ่ง ๆ นั้นจะมีความต้องการที่เกี่ยวข้องอยู่ 1 หรือมากกว่าก็ได้ ดังนั้น เมตริกซ์นี้จึงมีประโยชน์ในการตรวจสอบว่า หากต้องการทดสอบความต้องการนั้น ๆ แล้ว จะต้องดำเนินการทดสอบที่โมดูลใด นอกจากนี้ยังช่วยตรวจสอบความครบถ้วนของการทดสอบด้วย จากตัวอย่างในตารางที่ 4.8 สังเกตได้ว่า โมดูลที่ 4 ยังไม่มีความต้องการใดเข้ามาเกี่ยวข้องเลย

หลังจากนั้น เครื่องมือจะรับข้อมูลลำดับการทดสอบความต้องการเหล่านี้ ซึ่งสามารถนำมาสร้างเป็นแผนภาพลำดับดังตัวอย่างในรูปที่ 4.3 เพื่อให้เข้าใจได้ง่าย สังเกตได้ว่า หากต้องการทดสอบความต้องการที่ 2 จะต้องทดสอบความต้องการที่ 1,4 และ 5 ก่อน นอกจากนี้ ความต้องการที่ 4 และ 5 สามารถทดสอบแยกกันได้



รูปที่ 4.4 ตัวอย่างแผนภาพลำดับ

4.2.1.6 ส่วนรับข้อมูลอื่น ๆ เกี่ยวกับแผนการทดสอบ

เป็นส่วนที่รับข้อมูลอื่น ๆ นอกเหนือจากที่ได้กล่าวมาแล้วข้างต้น โดยส่วนใหญ่จะเป็นข้อมูลเชิงบรรยาย อันได้แก่ การแนะนำ (Introduction) ลักษณะที่ไม่ทดสอบ (Features not to be tested) แนวความ

คิด (approaches) ผลลัพธ์ที่ได้จากการทดสอบ (deliverables) ทรัพยากรที่ใช้ (Environmental needs) เป็นต้น

4.2.1.7 ส่วนออกรายงานแผนการทดสอบ

เป็นส่วนที่ทำการออกรายงานแผนการทดสอบโดยมีรูปแบบตามแม่แบบที่ผู้ใช้เลือก หากเป็นผลิตภัณฑ์ผู้ใช้เพิ่มเข้ามา ผู้ใช้จะต้องกรอกข้อมูลให้เรียบร้อยก่อน

4.2.2 ตัวจัดกำหนดการ

4.2.2.1 ส่วนจัดกำหนดการของโครงการ

เป็นส่วนที่รับข้อมูลกิจกรรมต่าง ๆ ของโครงการซึ่งแบ่งออกเป็น 2 ประเภท คือ กิจกรรมที่ผู้ใช้งานดำเนินการเอง เช่น การประชุม เป็นต้น และกิจกรรมที่เครื่องมือจัดการให้ ได้แก่ การดำเนินการทดสอบระดับหน่วย การทดสอบการรวม และการทดสอบระบบ หากเป็นกิจกรรมที่เครื่องมือจัดการให้ เครื่องมือสามารถปรับปรุงข้อมูล (update) วันที่เริ่มต้น ระยะเวลา ความก้าวหน้าได้โดยอัตโนมัติ หากเป็นกิจกรรมอื่น ผู้ใช้จะต้องเป็นผู้กรอกข้อมูลเอง

4.2.2.2 ส่วนจัดกำหนดการของแผนการทดสอบแต่ละระดับ

เป็นส่วนที่รับข้อมูลวันเริ่มต้น ระยะเวลา และความสัมพันธ์กับกิจกรรมอื่นของแต่ละกิจกรรมแล้วทำการคำนวณและแสดงผลให้อยู่ในรูปของแผนภูมิแกนต์ เพื่อแสดงถึงวันที่กิจกรรมต่าง ๆ เหล่านั้นสามารถดำเนินการได้โดยสอดคล้องกับเงื่อนไขที่กำหนด แสดงความสัมพันธ์ระหว่างกิจกรรม แสดงวันที่คาดว่าจะแล้วเสร็จ และแสดงกิจกรรมที่ดำเนินงานแบบคู่ขนานกัน สามารถทำให้ผู้ใช้เข้าใจได้ง่าย โดยผู้ใช้สามารถปรับแต่งกำหนดการได้ตามต้องการ

4.2.2.3 ส่วนวิเคราะห์ความสัมพันธ์ระหว่างกิจกรรม

4.2.2.3.1 ส่วนวิเคราะห์เวลายืดหยุ่น วันที่เริ่มต้นเร็วที่สุด และวันที่เริ่มต้นช้าที่สุด สำหรับแต่ละกิจกรรม

เป็นส่วนที่ให้ข้อมูลกับผู้ใช้เกี่ยวกับความยืดหยุ่นของแต่ละกิจกรรม อันได้แก่ เวลายืดหยุ่น วันที่เริ่มต้นเร็วที่สุด และวันที่เริ่มต้นช้าที่สุด เพื่อช่วยในการตัดสินใจปรับแต่งกำหนดการ ให้มีการใช้ทรัพยากรอย่างคุ้มค่าที่สุด และงานเสร็จทันตามกำหนดเวลา

4.2.2.3.2 ส่วนวิเคราะห์วิถีวิกฤต (Critical Path)

เป็นส่วนที่วิเคราะห์ว่า วิถีใดใช้เวลานานที่สุด ซึ่งหมายความว่า ผู้จัดการโครงการจะต้องให้ความสำคัญกับลำดับกิจกรรมนี้เป็นอันดับแรก หากมีงานใดงานหนึ่งในลำดับนี้ล่าช้าไป จะมีผลกระทบต่อกำหนดการของทั้งโครงการ เครื่องมือจะแสดงผลวิถีวิกฤตในแผนภูมิแกนต์ด้วยสัญลักษณ์ที่เด่นกว่ากิจกรรมอื่น ๆ

4.2.2.3 ส่วนจัดการเวอร์ชัน (version) ของกำหนดการ

เป็นส่วนที่ให้ผู้ใช้งานสามารถจัดกำหนดการได้หลายเวอร์ชัน โดยเลือกเอาเวอร์ชันหนึ่งเป็นกำหนดการที่ใช้งานจริง มีประโยชน์ในการปรับเปลี่ยนกำหนดการได้ทันทีหากเวอร์ชันปัจจุบันนั้นมีข้อผิดพลาด หรือมีความล่าช้าเกิดขึ้น ซึ่งจะส่งผลกระทบต่อกำหนดการของทั้งโครงการ

4.2.2.4 ส่วนจัดสรรทรัพยากร

เป็นส่วนที่รับข้อมูลการจัดสรรทรัพยากรสำหรับแต่ละกิจกรรม แล้วนำมาแสดงในรูปของกราฟแท่งแบบระดับ เพื่อแสดงความต้องการใช้ทรัพยากรในแต่ละวัน และเพื่อเป็นข้อมูลที่สำคัญในการปรับแต่งกำหนดการเพื่อทำให้มีการใช้ทรัพยากรอย่างสม่ำเสมอและคุ้มค่า

4.2.2.5 ส่วนกำหนดความรับผิดชอบ

4.2.2.5.1 ส่วนกำหนดและยกเลิกความรับผิดชอบ

เป็นส่วนที่ใช้จัดสรรผู้ออกแบบกรณีทดสอบ และผู้ทดสอบให้เพียงพอกับความต้องการในแต่ละกิจกรรม รวมทั้งยกเลิกความรับผิดชอบสำหรับกิจกรรมนั้น ๆ ด้วย

4.2.2.5.2 ส่วนตรวจสอบปริมาณงาน (workload) ของผู้ทดสอบ

เป็นส่วนที่แสดงวันที่ผู้ออกแบบกรณีทดสอบและผู้ทดสอบมีกิจกรรมที่ต้องทำในรูปแบบของตาราง และแสดงสัญลักษณ์พิเศษในวันที่ต้องทำกิจกรรมมากกว่า 1 กิจกรรม มีประโยชน์ในการตรวจสอบวันว่างของผู้ทดสอบแต่ละคน เพื่อที่สามารถกำหนดความรับผิดชอบลงไปได้ โดยไม่ซ้ำซ้อนกับกิจกรรมอื่น ๆ

4.2.2.6 ส่วนตรวจสอบความก้าวหน้าของกิจกรรม

เป็นส่วนที่เครื่องมือสามารถตรวจสอบความก้าวหน้าของแต่ละกิจกรรมได้โดยอัตโนมัติ โดยพิจารณาจากเอกสารที่เกิดขึ้นระหว่างการทดสอบ โดยในการทดสอบแต่ละครั้งจะมีขั้นตอนต่าง ๆ ดังนี้

- การออกแบบการทดสอบ (Test Design) เป็นการออกแบบชั้นสมมูล (Equivalence Class) ของส่วนของโปรแกรมที่นำมาทดสอบ แล้วจึงนำไปสร้างเป็นกรณีทดสอบ เพื่อให้ครอบคลุมทุก ๆ กรณีที่เป็นไปได้
- การออกแบบกรณีทดสอบ เป็นการออกแบบกรณีทดสอบจากชั้นสมมูลที่ได้กำหนดไว้
- การดำเนินการทดสอบ เป็นการที่ผู้ทดสอบนำกรณีทดสอบที่ได้ไปดำเนินการทดสอบ แล้วรายงานผลกลับมายังเครื่องมือ โดยแบ่งออกเป็น
 - ผลลัพธ์ถูกต้อง ถือว่าสิ้นสุดการทดสอบกรณีทดสอบนั้นแล้ว
 - เกิดข้อผิดพลาด หากมีการแก้ไขแล้วก็ถือว่าสิ้นสุดการทดสอบกรณีทดสอบนั้นแล้วเช่นกัน
 จะสังเกตได้ว่า ทุกขั้นตอนในการทดสอบจะมีเอกสารเกิดขึ้น ซึ่งโปรแกรมจะทำการตรวจสอบการทำงานแต่ละขั้นตอนจากเอกสารเหล่านี้เพื่อแสดงความก้าวหน้าของแต่ละกิจกรรม

4.2.2.7 ส่วนออกรายงานและสอบถามข้อมูลเกี่ยวกับกำหนดการ

4.2.2.7.1 ส่วนสอบถามข้อมูลเกี่ยวกับกำหนดการของผู้ทดสอบแต่ละคน

เป็นส่วนที่ช่วยในการค้นหาข้อมูลความรับผิดชอบของผู้ทดสอบแต่ละคนสำหรับโครงการนั้น ๆ

4.2.2.7.2 ส่วนออกรายงานแสดงกำหนดการที่ได้วางแผนไว้

เป็นส่วนที่ใช้ออกรายงานแสดงกำหนดการของแต่ละกิจกรรม ประกอบด้วย ชื่อกิจกรรม วันที่เริ่มต้น ระยะเวลา และกิจกรรมก่อนหน้า (predecessor)

4.2.2.7.3 ส่วนออกรายงานแสดงความก้าวหน้าของแต่ละกิจกรรม

เป็นส่วนที่ใช้ออกรายงานแสดงความก้าวหน้าของกิจกรรม ประกอบด้วย ชื่อกิจกรรม วันที่เริ่มต้น ระยะเวลา และความก้าวหน้า

4.2.2.7.4 ส่วนออกรายงานแสดงกิจกรรมที่เลยกำหนด

เป็นส่วนที่ใช้ออกรายงานแสดงกิจกรรมที่มีความก้าวหน้าล่าช้ากว่าที่กำหนดไว้เมื่อเสร็จสิ้นวันนั้นแล้ว ประกอบด้วย ชื่อกิจกรรม วันที่เริ่มต้น ระยะเวลา ความก้าวหน้าที่น่าคาดหวังไว้ (Expected Progress) ความก้าวหน้าที่เกิดขึ้นจริง (Actual Progress) และเปอร์เซ็นต์ความล่าช้า (% delayed)

4.2.2.7.5 ส่วนออกรายงานแสดงการจัดสรรทรัพยากร

เป็นส่วนที่ใช้ออกรายงานแสดงการจัดสรรทรัพยากรหนึ่ง ๆ สำหรับทุก ๆ กิจกรรมในโครงการในรูปของตัวหนังสือ ประกอบด้วย ชื่อทรัพยากร ชื่อกิจกรรม การจัดสรร (Allocated)

4.2.2.7.6 ส่วนออกรายงานแสดงกิจกรรมที่ต้องทำแบ่งตามผู้ทดสอบ

เป็นส่วนที่ใช้ออกรายงานแสดงกิจกรรมที่ผู้ทดสอบต้องรับผิดชอบในโครงการนั้น ๆ ประกอบด้วย ชื่อผู้ทดสอบ ชื่อกิจกรรม วันที่เริ่มต้น และระยะเวลา

4.2.2.7.7 ส่วนออกรายงานแสดงความรับผิดชอบแบ่งตามกิจกรรม

เป็นส่วนที่ใช้ออกรายงานแสดงความรับผิดชอบของผู้ทดสอบสำหรับกิจกรรมนั้น ๆ ประกอบด้วย ชื่อกิจกรรม ชื่อผู้ทดสอบ และตำแหน่ง

4.2.3 ตัวจัดการเอกสารเกี่ยวกับการทดสอบ

4.2.3.1 ส่วนจัดการเอกสารเกี่ยวกับการออกแบบการทดสอบ

เป็นส่วนที่ใช้เพิ่ม แก้ไข และลบเอกสารเกี่ยวกับการออกแบบการทดสอบที่ได้กำหนดไว้ โดยจะเชื่อมโยงไปยังเอกสารเกี่ยวกับกรณีทดสอบ และรายงานสรุปผลการทดสอบด้วย

4.2.3.2 ส่วนจัดการเอกสารเกี่ยวกับกรณีทดสอบ

เป็นส่วนที่ใช้เพิ่ม แก้ไข และลบเอกสารเกี่ยวกับกรณีทดสอบที่ได้กำหนดไว้ ซึ่งจะสัมพันธ์กับการแสดงกรณีทดสอบ (Test Identification) ที่ได้กำหนดขึ้นสมบูรณ์แล้วในการออกแบบการทดสอบ

- 4.2.3.3 ส่วนจัดการรายงานส่วนของโปรแกรมที่ถูกส่งไปทดสอบ
เป็นส่วนที่ช่วยบันทึกข้อมูลเกี่ยวกับส่วนของโปรแกรมที่ถูกส่งไปทดสอบ
- 4.2.3.4 ส่วนจัดการเอกสารเกี่ยวกับลำดับการทดสอบ
เป็นส่วนที่ช่วยบันทึกข้อมูลเกี่ยวกับกระบวนการต่าง ๆ ที่ต้องทำก่อนและหลังการดำเนินการทดสอบด้วยกรณีทดสอบที่ได้ออกแบบไว้ รวมทั้งการเรียงลำดับกรณีทดสอบด้วย
- 4.2.3.5 ส่วนจัดการบันทึกการทดสอบ
เป็นส่วนที่ช่วยบันทึกข้อมูลเกี่ยวกับเหตุการณ์ต่าง ๆ ที่เกิดขึ้นในระหว่างการทดสอบ ซึ่งจะเชื่อมโยงไปยังรายงานการทดสอบที่ต้องดำเนินการต่อหากเหตุการณ์นั้นมีข้อผิดพลาด
- 4.2.3.6 ส่วนจัดการรายงานการทดสอบที่ต้องดำเนินการต่อ
เป็นส่วนที่ช่วยบันทึกข้อมูลความผิดพลาดและผลกระทบจากเหตุการณ์ที่เกิดขึ้นระหว่างการทดสอบ
- 4.2.3.7 ส่วนจัดการรายงานสรุปผลการทดสอบ
เป็นส่วนที่ช่วยบันทึกข้อมูลสรุปผลการทดสอบว่าตรงตามที่ได้วางแผนไว้หรือไม่

ในตัวจัดการเอกสารเกี่ยวกับการทดสอบทั้ง 7 ส่วนนี้ สามารถออกรายงานที่มีรูปแบบตามแม่แบบที่ได้กำหนดไว้ในแผนการทดสอบที่ระดับนั้นได้ทันที

4.3 การกำหนดระดับการเข้าถึงเครื่องมือของผู้ใช้

เครื่องมือได้กำหนดกลุ่มผู้ใช้และระดับการเข้าถึงไว้ 3 กลุ่มคือ

- ก) ผู้จัดการ (Manager) เป็นผู้ริเริ่มโครงการ วางแผนการทดสอบแต่ละระดับ วางแผนการใช้งานทรัพยากรให้เหมาะสมกับแต่ละกิจกรรม และมีสิทธิ์ในการเข้าถึงการดำเนินกิจกรรมทุก ๆ กิจกรรม
- ข) ผู้ออกแบบกรณีทดสอบ (Designer) เป็นผู้ที่มีหน้าที่ในการออกแบบวิธีการทดสอบและกรณีทดสอบสำหรับแต่ละกิจกรรมในโครงการที่ได้รับมอบหมาย และบันทึกข้อมูลต่าง ๆ เกี่ยวกับการดำเนินการทดสอบได้
- ค) ผู้ทดสอบ (Tester) เป็นผู้ที่มีหน้าที่ในการดำเนินการทดสอบด้วยกรณีทดสอบที่ผู้ออกแบบกรณีทดสอบได้กำหนดไว้ และรายงานผลการทดสอบกลับมายังเครื่องมือ

สามารถสรุปความสามารถของผู้ใช้แต่ละกลุ่มได้ดังตารางที่ 4.9

ตารางที่ 4.9 การกำหนดระดับการเข้าถึงเครื่องมือของผู้ใช้แต่ละกลุ่ม

ความสามารถ	ผู้จัดการ	ผู้ออกแบบ กรณีทดสอบ	ผู้ทดสอบ
1) สร้างโครงการใหม่	✓	-	-
2) เปิดโครงการที่รับผิดชอบ	✓	✓	✓
3) สร้างแผนการทดสอบแต่ละระดับ	✓	-	-
4) เปิดแผนการที่สอบที่รับผิดชอบ	✓	✓	✓
5) บันทึกรายละเอียดหรือแก้ไขแผนการทดสอบ	✓	-	-
6) ลบแผนการทดสอบ	✓	-	-
7) พิมพ์รายละเอียดของแผนการทดสอบ	✓	✓	✓
8) ออกแบบวิธีการทดสอบและกรณีทดสอบ	✓	✓	-
9) รายงานผลการทดสอบกลับมายังเครื่องมือ	✓	✓	✓
10) บันทึกข้อมูลลำดับการทดสอบ	✓	✓	-
11) บันทึกข้อมูลส่วนของโปรแกรมที่ถูกส่งไปทดสอบ	✓	✓	-
12) บันทึกข้อมูลเหตุการณ์ที่เกิดขึ้นระหว่างการทดสอบ	✓	✓	-
13) บันทึกข้อมูลเหตุการณ์ที่ต้องดำเนินการต่อ	✓	✓	-
14) บันทึกข้อมูลสรุปผลการทดสอบ	✓	✓	-
15) พิมพ์รายละเอียดของเอกสารเกี่ยวกับการทดสอบ	✓	✓	✓
16) จัดและแก้ไขกำหนดการ	✓	-	-
17) ระบุปริมาณการใช้ทรัพยากร	✓	-	-
18) กำหนดความรับผิดชอบสำหรับแต่ละกิจกรรม	✓	-	-
19) สืบค้นข้อมูลกำหนดการและการดำเนินกิจกรรม	✓	✓	✓
20) จัดการข้อมูลผู้ใช้และข้อมูลองค์กร	✓	-	-

4.4 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือซอฟต์แวร์

เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนามีรายละเอียดดังนี้

◆ เครื่องให้บริการ (Server)

▪ ฮาร์ดแวร์

- คอมพิวเตอร์แบบพีซี Pentium III 500 เมกกะเฮิร์ต
- หน่วยความจำ 128 เมกกะไบต์
- ฮาร์ดดิสก์ 12 กิกะไบต์

- ซอฟต์แวร์
 - ระบบปฏิบัติการ ไมโครซอฟต์วินโดวส์เอ็นที 4.0
 - ระบบฐานข้อมูล ไมโครซอฟต์เอสคิวแอล เซิร์ฟเวอร์ (SQL Server) 6.5

- ◆ เครื่องรับบริการ (Client)
 - ฮาร์ดแวร์
 - คอมพิวเตอร์แบบพีซี Pentium III 450 เมกกะเฮิร์ต
 - หน่วยความจำ 64 เมกกะไบต์
 - ฮาร์ดดิสก์ 6.4 กิกะไบต์
 - ซอฟต์แวร์
 - ระบบปฏิบัติการ ไมโครซอฟต์วินโดวส์ 98
 - เครื่องมือพัฒนา บอร์แลนด์เดลไฟ 4.0