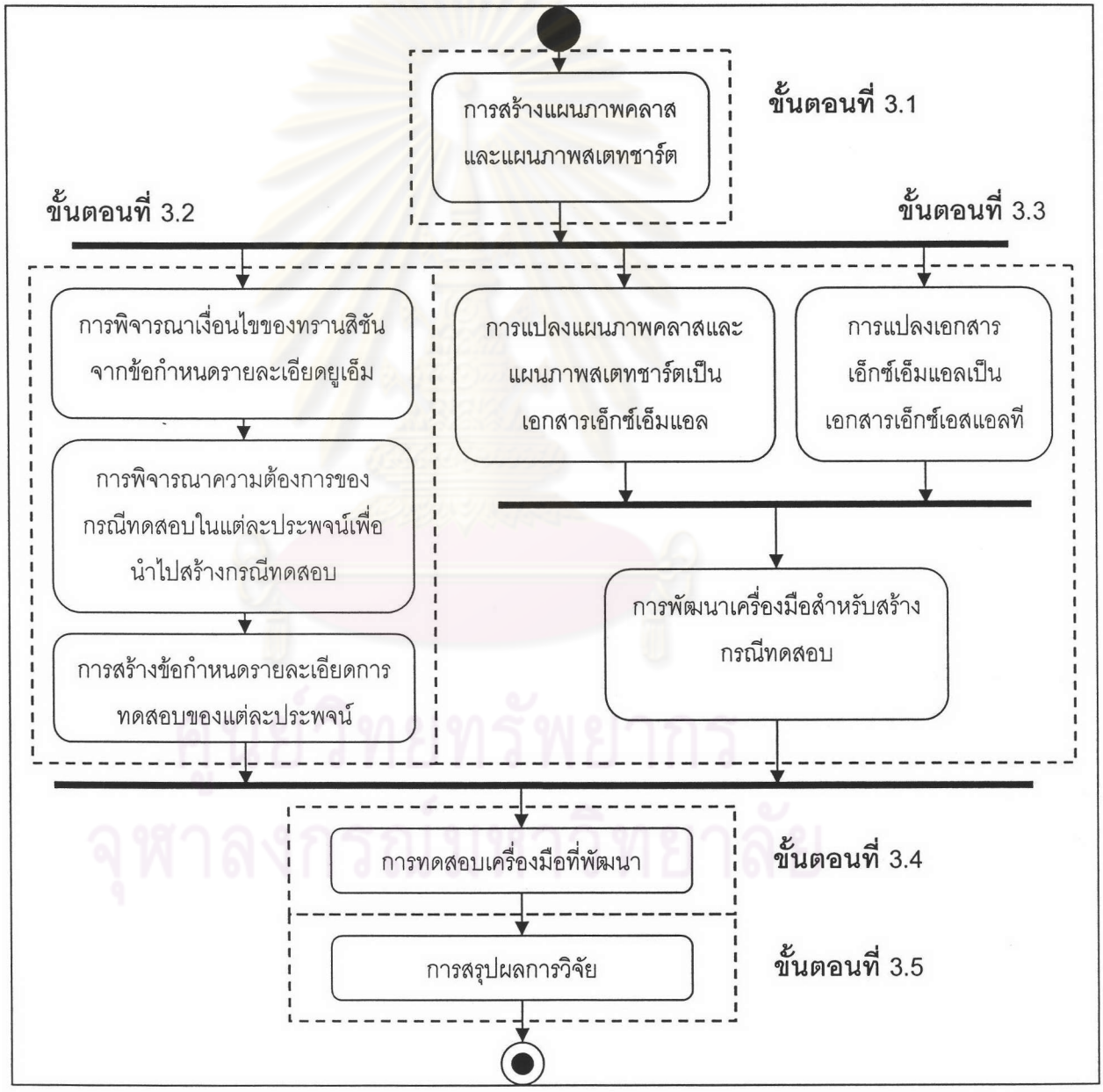
# บทที่ 3

# การวิเคราะห์และออกแบบการสร้างกรณีทดสอบ

งานวิจัยนี้แบ่งขั้นตอนในการดำเนินงานวิจัยออกเป็น 4 ส่วน สามารถแสดงได้ด้วยแผนภาพกิจกรรม (Activity diagram) ดังรูปที่ 3.1



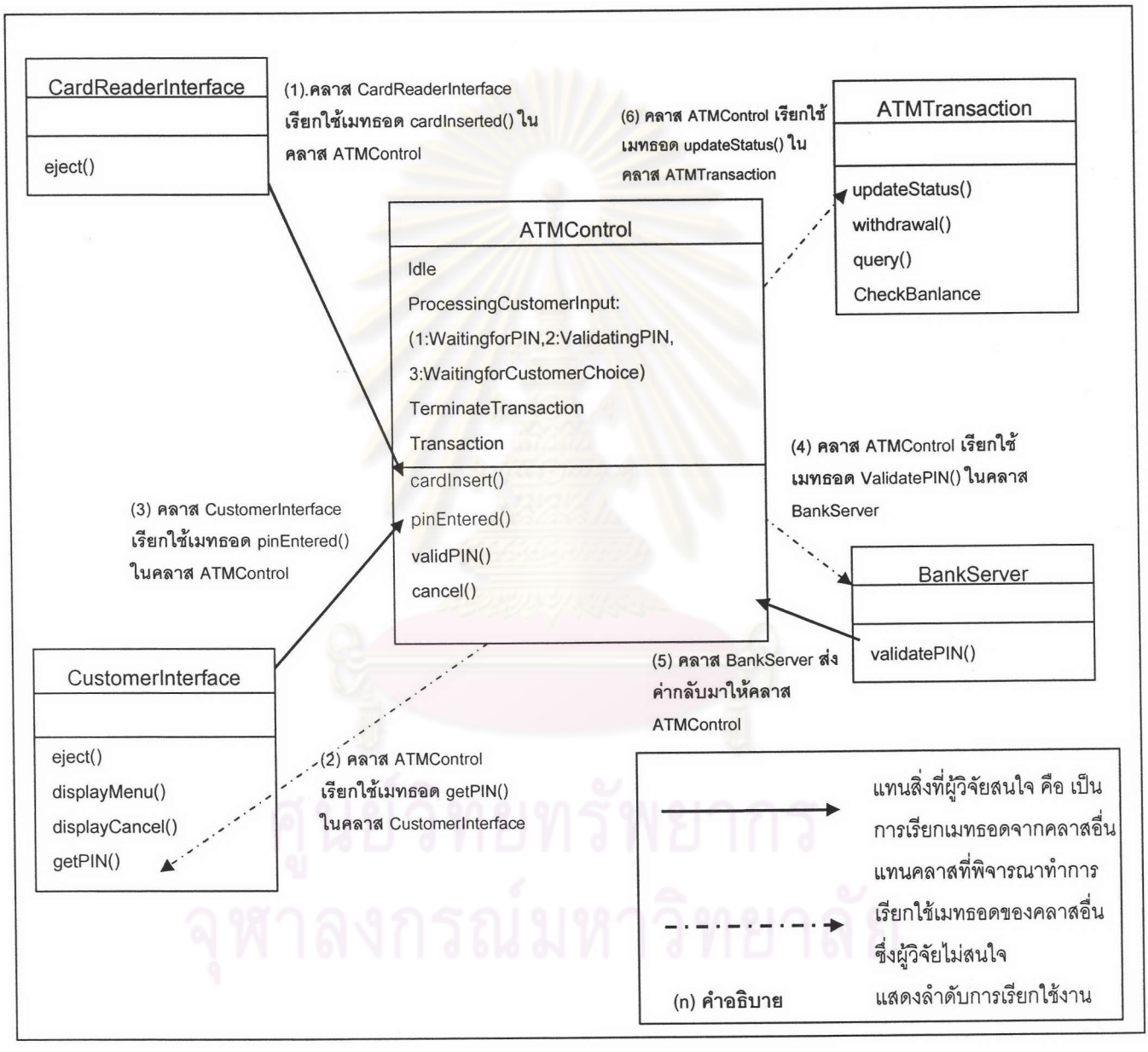

รูปที่ 3.1 แผนภาพขั้นตอนการดำเนินการวิจัย

ขั้นตอนที่ 3.1 เป็นขั้นตอนเริ่มต้นของการดำเนินงาน โดยเริ่มจากการสร้างแผนภาพคลาสและแผนภาพสเตทชาร์ตสำหรับระบบที่ต้องการสร้างกรณีทดสอบและแสดงถึงความสัมพันธ์ระหว่างแผนภาพคลาสและแผนภาพสเตทชาร์ต ขั้นตอนที่ 3.2 นำเสนอแนวคิดในการสร้างกรณีทดสอบ ขั้นตอนที่ 3.3 แสดงขั้นตอนของการวิเคราะห์และออกแบบเครื่องมือสร้างกรณีทดสอบ โดยจะกล่าวถึงการออกแบบโครงสร้างพื้นฐานของเครื่องมือและเทคโนโลยีที่ใช้ในการสร้างกรณีทดสอบ ซึ่งประกอบไปด้วยการแปลงแผนภาพคลาสและแผนภาพสเตทชาร์ตเป็นเอกสารเอ็กซ์-เอ็มแอล ขั้นตอนการแปลงเอกสารเอ็กซ์เอ็มแอลเป็นเอ็กซ์เอสแอลที และสถาปัตยกรรมของเครื่องมือ หลังจากนั้นจะอธิบายถึงการวิเคราะห์และออกแบบโดยใช้แผนภาพยูสเคส (Use case-diagram) แผนภาพคลาส แผนภาพกิจกรรม โครงสร้างข้อมูลเอ็กซ์เอ็มแอล โครงสร้างการเรียกเทมเพลต และการออกแบบสถาปัตยกรรมของเครื่องมือ เป็นต้น รายละเอียดของแต่ละขั้นตอนแสดงในหัวข้อ 3.1 3.2 3.3 ตามลำดับ ส่วนขั้นตอน 3.4 3.5 แสดงขั้นตอนของการทดสอบเครื่องมือที่ได้พัฒนาขึ้นและการสรุปผลการวิจัย จะกล่าวถึงต่อไปในบทที่ 5 และ 6 ตามลำดับ

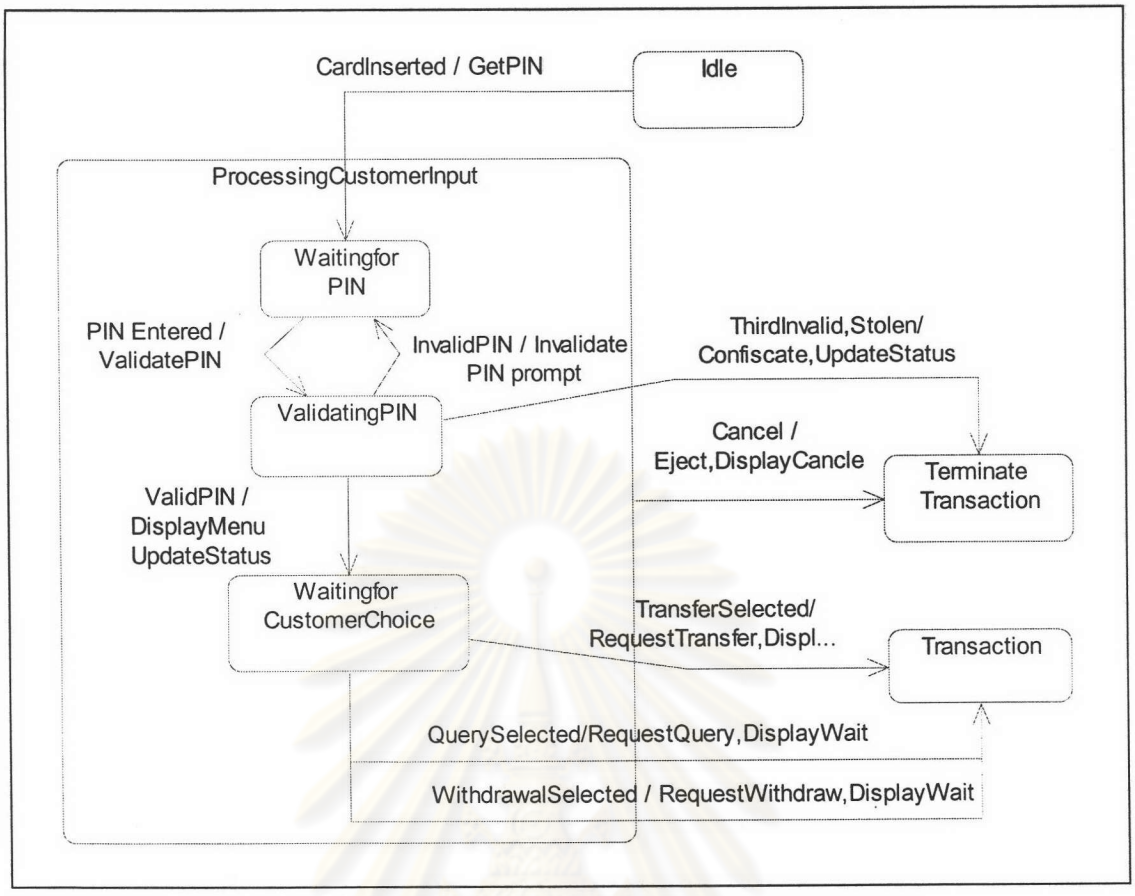
## 3.1 การวิเคราะห์ความสัมพันธ์ระหว่างแผนภาพคลาสและแผนภาพสเตทชาร์ต

ในขั้นตอนนี้จะทำการพิจารณาการสร้างแผนภาพคลาสและแผนภาพสเตทชาร์ตที่จะนำมาใช้ในการสร้างกรณีทดสอบ ซึ่งจะแสดงให้เห็นถึงความสัมพันธ์ที่มีในการอ้างถึงกันระหว่างแผนภาพ 2 แผนภาพ โดยจะแสดงให้เห็นว่าสถานะของแผนภาพสเตทชาร์ตมาจากส่วนใดของคลาสที่ทำการพิจารณา เป็นต้น โดยจากรูป 3.2 และ 3.3 แสดงให้เห็นความสัมพันธ์ระหว่างคลาส ATMControl และแผนภาพสเตทชาร์ตที่เกิดขึ้นของคลาสนี้ แสดงได้ดังรูปที่ 3.2 และ 3.3 ตามลำดับ จากรูปที่ 3.2 ตัวอย่างนี้แสดงคลาส ATMControl ซึ่งในแผนภาพคลาสที่แสดงจะมีคลาสต่างๆ ที่เกี่ยวข้องกัน เช่น คลาส CardReaderInterface ทำการเรียกใช้เมทธอด cardInserted() ภายในคลาส ATMControl ซึ่งในคลาส ATMControl จะมีสถานะการทำงานอยู่ภายใน แสดงได้ด้วยแผนภาพสเตทชาร์ต ดังรูปที่ 3.2 ซึ่งสถานะของคลาสจะเก็บเป็นแอทริบิวต์หนึ่งของคลาสที่พิจารณา โดยมีการให้ค่ากับแต่ละสถานะที่เกิดขึ้น เช่น คลาส ATMControl มีแอทริบิวต์ชื่อว่า Status ซึ่งมีการเก็บสถานะของคลาส ซึ่งค่า 1 แทนสถานะ Idle ค่า 2 แทนสถานะ ProcessingCustomerInput ค่า 3 แทนสถานะ TerminateTransaction และค่า 4 แทนสถานะ Transaction เพื่อทำให้ทราบว่า ณ ปัจจุบันคลาสที่กำลังพิจารณาอยู่ในสถานะใดบ้าง ซึ่งในแผนภาพสเตทชาร์ตจะมีการทำงานได้ก็ต่อเมื่อมีเหตุการณ์มากระตุ้น ซึ่งเรียกว่า เหตุการณ์ที่เป็นตัวกระตุ้น ซึ่งจะทำให้เกิดการเปลี่ยนสถานะหนึ่งไปเป็นอีกสถานะหนึ่ง โดยจากตัวอย่าง คลาส CardReaderInterface ทำการเรียกใช้ เมทธอดภายในคลาส ATMControl จึงทำให้สถานะภายในคลาสเกิดการทำงาน ซึ่งเหตุการณ์ที่เป็นตัวกระตุ้นในแผนภาพสเตทชาร์ตก็จะเป็น เมทธอด

หนึ่งในคลาส ATMControl ซึ่งจะถูกเรียกใช้โดยคลาส CardReaderInterface นั่นเอง ส่วนการกระทำที่เกิดขึ้นในแผนภาพสเตท-ชาร์ตจะแทนด้วย การที่คลาส ATMControl ทำการเรียกใช้เมท-ธอดภายในคลาส หรือส่งเมจเสจไปเรียกใช้เมทธอดในคลาสอื่น เช่น คลาส ATMControl เมื่อมีเหตุการณ์ภายนอกมากระตุ้นแล้วทำให้เกิดการกระทำคือ GetPIN ก็จะส่งเมจเสจไปยังคลาส CustomerInterface เพื่อให้ผู้ใช้ใส่รหัสเข้ามา




รูปที่ 3.2 แผนภาพแสดงลำดับการเรียกใช้งานระหว่างคลาส ATMControl และคลาสอื่นๆ ที่เกี่ยวข้องในยูสเคสการทดสอบรหัส [11]

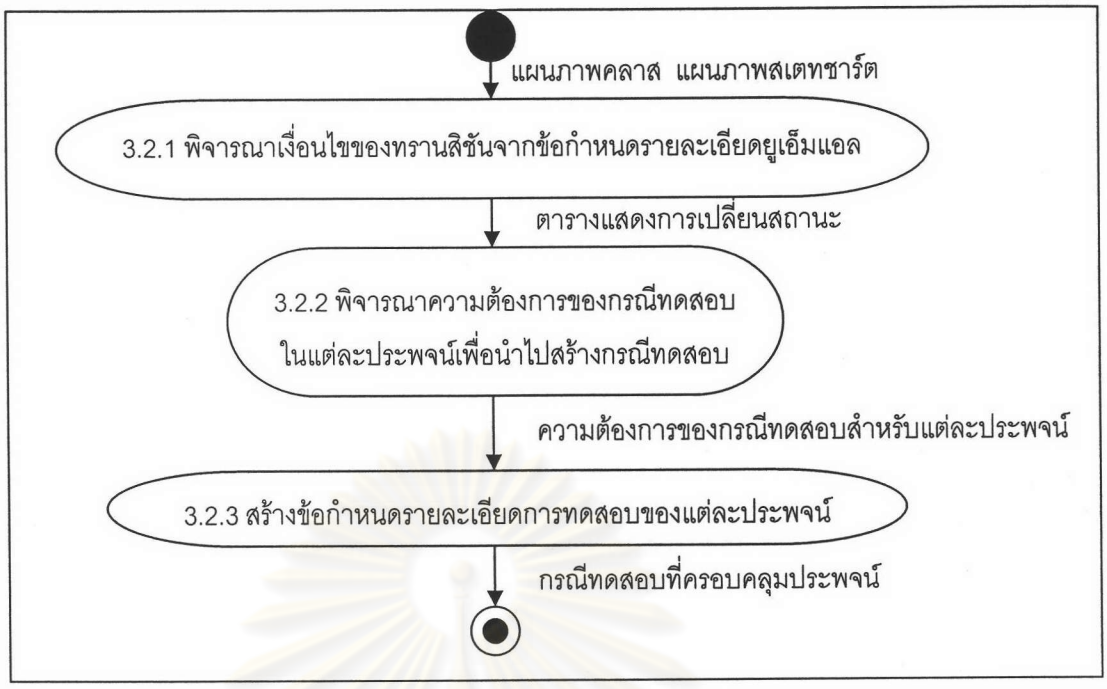



รูปที่ 3.3 แผนภาพสเตทชาร์ตของคลาส ATMControl [11]

โดยที่ผู้วิจัยให้ความสนใจกับเหตุการณ์ที่เข้ามาในคลาสโดยมีคลาสอื่นเรียกใช้เมทธอดในคลาสที่พิจารณา ซึ่งจะทำให้เกิดการเปลี่ยนสถานะภายในคลาสที่พิจารณา ในที่นี้ก็คือเหตุการณ์ที่เป็นตัวกระตุ้น โดยไม่สนใจเมทธอดที่คลาสที่พิจารณาไปเรียกใช้ในคลาสอื่น ซึ่งในที่นี้คือ การกระทำที่เกิดขึ้นภายในแผนภาพสเตทชาร์ต

## 3.2 แนวคิดการสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต

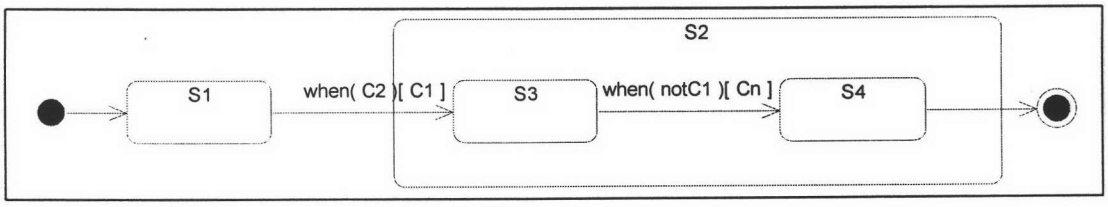
ในงานวิทยานิพนธ์นี้ ได้นำขั้นตอนในการสร้างกรณีทดสอบมาประยุกต์ใช้เพิ่มเติมจากงานวิจัย [7, 23] ซึ่งจะพิจารณาแผนภาพสเตทชาร์ตที่มีสถานะซับซ้อน โดยขั้นตอนของแนวคิดในการสร้างกรณีทดสอบเป็นดังรูปที่ 3.4




รูปที่ 3.4 ขั้นตอนของแนวคิดในการสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต

3.2.1 พิจารณาเงื่อนไขของทรานสิชันจากข้อกำหนดรายละเอียดยูเอ็มแอล

ขั้นตอนนี้เป็นการพิจารณาเงื่อนไขของทรานสิชันว่ามีประพจน์ใดภายใต้เงื่อนไขอะไรบ้างในแต่ละทรานสิชันที่เกิดขึ้น แสดงได้ดังตารางที่ 3.1 โดยมี 4 สดมภ์ (Column) ด้วยกันคือ สดมภ์แรกคือ สถานะแม่ (Parent state) หรือเรียกว่าสถานะซับซ้อน ซึ่งจะแทนสถานะที่มีสถานะย่อยภายใน โดยที่สถานะย่อยภายในเป็นสถานะปัจจุบันที่กำลังพิจารณาอยู่ กรณีที่ไม่มีสถานะย่อยภายในก็จะไม่สนใจคือใส่เครื่องหมาย "-" เข้าไปในตาราง สดมภ์ที่สอง คือ สถานะปัจจุบัน (Current state) เป็นสถานะที่กำลังพิจารณาในแต่ละทรานสิชันที่มีเหตุการณ์เข้ามากระตุ้น สดมภ์ที่สาม คือ เงื่อนไขเหตุการณ์ (Conditioned event) จะประกอบไปด้วยเหตุการณ์ที่มีการกระตุ้นและเงื่อนไขการ์ด ส่วนสดมภ์ที่สี่ คือ สถานะเป้าหมาย (Target state) คือ สถานะที่คาดว่าจะเกิดขึ้นเมื่อมีเหตุการณ์เข้ามากระตุ้นซึ่งทำให้เกิดการเปลี่ยนสถานะหรือยังคงอยู่สถานะเดิม โดยในแต่ละแถว (Row) ของตารางจะทำการกำหนดค่าเงื่อนไขเหตุการณ์ ที่ไปกระตุ้นทรานสิชันทำให้เกิดการเปลี่ยนสถานะจากสถานะปัจจุบัน ไปยังสถานะเป้าหมาย โดยแถวที่มีการใส่สัญลักษณ์ @T(C) หรือ @F(C) ที่ใส่ภายใต้สดมภ์ของเงื่อนไขเหตุการณ์ จะเป็นเหตุการณ์ที่มีการกระตุ้น และแถวที่มีการใส่ค่า t หรือ f แทน ค่าของเงื่อนไขการ์ด เช่น [C] หมายความว่า ทรานสิชันจะเกิดได้ถ้าค่า C เป็นจริง และ [¬C] หมายความว่า ทรานสิชันจะเกิดได้ถ้าค่า C เป็นเท็จ แต่ถ้าค่าของเงื่อนไขสิ่งแวดล้อม (Environment condition) C ไม่ส่งผลต่อเงื่อนไขของเหตุการณ์ก็ให้ทำเครื่องหมาย "-" คือไม่สนใจเงื่อนไข โดยพิจารณาจากแผนภาพสเตทชาร์ตดังรูปที่ 3.5




รูปที่ 3.5 แผนภาพสเตทชาร์ต

ตารางที่ 3.1 การเปลี่ยนสถานะ

| สถานะแม่ (Parent state) | สถานะปัจจุบัน | เงื่อนไขเหตุการณ์ | | | | สถานะเป้าหมาย |
|---|---|---|---|---|---|---|
| | | C1 | C2 | ... | Cn | |
| - | S1 | t | @T | - | - | S3 |
| S2 | S3 | @F | - | - | f | S4 |

ซึ่งเหตุการณ์ที่เป็นตัวกระตุ้น ในตารางที่ 3.1 นี้ทำการเขียนให้ขยายออกมา โดยพิจารณาจากกฎการขยายเหตุการณ์ที่มีการกระตุ้นตามทฤษฎีที่ได้แสดงไว้ในบทที่ 2 โดยในตารางจะใช้กฎการขยายข้อที่ 1 ซึ่งอธิบายได้ดังตัวอย่างต่อไปนี้ โดยจะพิจารณาสถานะจาก S1 ไปยัง S3 และจาก S3 ไปยัง S4 จะแทนได้ด้วยประพจน์ที่ 1 และ 2 ตามลำดับ

ประพจน์ที่ 1 มีเหตุการณ์ที่มีการกระตุ้นคือ C2 และมีเงื่อนไขการ์ด คือ C1 เมื่อใช้กฎการขยายจะได้ดังนี้ $@T(C2) \wedge C1 = \neg C2 \wedge C2' \wedge C1$

ประพจน์ที่ 2 มีเหตุการณ์ที่มีการกระตุ้นคือ C1 และมีเงื่อนไขการ์ด คือ Cn เมื่อใช้กฎการขยายจะได้ดังนี้ $@T(C1) \wedge Cn = \neg C1 \wedge C1' \wedge Cn$

เมื่อทำการขยายกฎครบทุกประพจน์แล้ว นำค่าที่ได้ไปใส่ในตารางที่ 3.2 เพื่อนำไปสร้างความต้องการของกรณีทดสอบ (Test case requirements) ในขั้นตอนต่อไป

ตารางที่ 3.2 ข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์

| หมายเลขประพจน์ | สถานะแม่ | สถานะปัจจุบัน | เงื่อนไขเหตุการณ์ | สถานะเป้าหมาย |
|---|---|---|---|---|
| P1 | - | S1 | $\neg C2 \wedge C2' \wedge C1$ | S3 |
| P2 | S2 | S3 | $C1 \wedge \neg C1' \wedge Cn$ | S4 |

3.2.2 พิจารณาความต้องการของกรณีทดสอบในแต่ละประพจน์เพื่อนำไปสร้างกรณีทดสอบ

เมื่อได้ข้อกำหนดรายละเอียดประพจน์แล้ว จากนั้นสร้างความต้องการของกรณีทดสอบในแต่ละประพจน์ให้ครอบคลุมตามหลักการสร้างการทดสอบแบบครอบคลุมประพจน์ นั่นคือ พิจารณาทุกประพจน์ที่เกิดขึ้นในทุกทรานสิชัน โดยทำการสร้างตารางค่าความจริง (Truth table) สำหรับทุกๆ ประพจน์ที่เกิดขึ้นจากข้อ 3.2.1 ซึ่งจะแยกความแตกต่างระหว่างค่าของเหตุการณ์ก่อนและหลังการตัวกระตุ้น โดยความต้องการของกรณีทดสอบเป็นดังตารางที่ 3.3

จากตารางที่ 3.3 ค่าของตัวแปรจะได้มาจากการพิจารณาแต่ละประพจน์ ซึ่งแสดงได้ด้วย T F t หรือ f โดยที่ T หรือ F หมายถึงเหตุการณ์ที่พิจารณาเป็นเหตุการณ์ที่มีการกระตุ้นประกอบด้วยค่าของเหตุการณ์ก่อนและหลังการกระตุ้น ส่วนค่า t และ f มาจากการพิจารณาค่าของเงื่อนไขการ์ด ตัวอย่างเช่น ในแถวแรกของตารางจะทำการพิจารณากรณีที่ข้อมูลการทดสอบเป็นถูกต้องทุกกรณี โดยทำการให้ค่าของเงื่อนไขต่างๆ เป็นจริง ซึ่งจะทำให้เกิดการเปลี่ยนสถานะจากสถานะ S1 ไปยังสถานะ S3 ส่วนแถวที่เหลือจะเป็นกรณีของการใส่ข้อมูลการทดสอบที่ไม่ถูกต้อง ซึ่งคาดการณ์ได้ว่าการทำงานยังอยู่ที่สถานะเดิมซึ่งเป็นสถานะเป้าหมาย โดยการใส่ข้อมูลการทดสอบของแถวที่เหลือนั้นจะทำการใส่ข้อมูลที่ไม่ถูกต้องครั้งละ 1 (Single fault) เงื่อนไขในแต่ละกรณีหรือแต่ละแถว เช่น ในแถวที่ 2 จะทำการใส่ข้อมูลที่ไม่ถูกต้องโดยให้ค่าของ C2 เป็นเท็จ ดังนั้นจึงใส่ค่า T ที่ตารางเนื่องจากว่าจะได้สอดคล้องกับกฎการขยายข้อ 1 คือ @T(C2) $\wedge$ C1 = $\neg$C2 $\wedge$ C2' $\wedge$ C1 จะให้ค่าที่เป็นเท็จ จึงทำให้ไม่สามารถเปลี่ยนสถานะได้ ส่วนแถวที่ 3 และ แถวที่ 4 จะทำการใส่ข้อมูลที่ผิดให้กับเงื่อนไขการ์ด C1 และเงื่อนไขที่มีเหตุการณ์เป็นตัวกระตุ้น ซึ่งเมื่อพิจารณาตามกฎการขยายแล้วผลลัพธ์ที่ได้จะให้ค่าเป็นเท็จ ตามลำดับเช่นเดียวกันกับแถวที่ 2

ตารางที่ 3.3 ความต้องการของกรณีทดสอบของประพจน์ที่ 1 (P1)

| หมายเลขประพจน์ | สถานะแม่ | สถานะปัจจุบัน | C1 | C2 | ... | Cn | เหตุการณ์หลังการกระตุ้น | สถานะเป้าหมาย |
|---|---|---|---|---|---|---|---|---|
| P1 | - | S1 | t | F | | | C2'=True | S3 |
| | - | S1 | t | T | | | C2'=True | S1 |
| | - | S1 | f | F | | | C2'=True | S1 |
| | - | S1 | t | F | | | C2'=False | S1 |

3.2.3 สร้างข้อกำหนดรายละเอียดการทดสอบของแต่ละประพจน์

เมื่อได้ความต้องการของกรณีทดสอบ จากขั้นตอนที่ 3.2.2 เรียบร้อยแล้ว จากนั้นนำแต่ละประพจน์ที่ได้มาสร้างข้อกำหนดรายละเอียดการทดสอบเพื่อสร้างกรณีทดสอบ โดยข้อกำหนดรายละเอียดกรณีทดสอบแสดงได้ดังตารางที่ 3.4

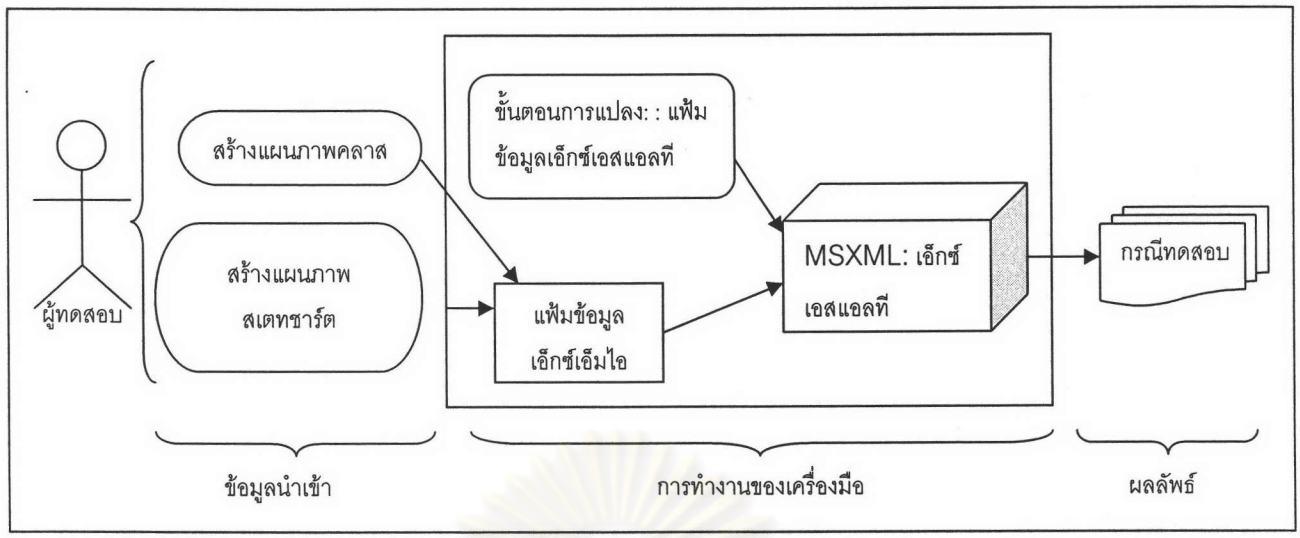
ตารางที่ 3.4 ข้อกำหนดรายละเอียดกรณีทดสอบของแต่ละประพจน์ [7]

| ข้อกำหนดกรณีทดสอบ (Test case specification) : P1-1: | |
|---|---|
| ค่าของพรีฟิก: | เป็นเงื่อนไขที่เป็นจริง ก่อนทำการทดสอบให้สามารถทำงานได้ซึ่งก็คือเงื่อนไขก่อนหน้าหรือเป็นสิ่งแวดล้อมของกรณีทดสอบ |
| ค่าของกรณีทดสอบ: | เป็นการกำหนดค่าก่อนหน้า ให้สำหรับเหตุการณ์ที่เป็นตัวกระตุ้น และเงื่อนไขใน when ซึ่งจะทำการกำหนดค่าให้เหตุการณ์ที่เป็นตัวกระตุ้นมีค่าเป็นจริง |
| ผลลัพธ์ที่คาดว่าจะเกิดขึ้น: | เป็นสถานะเป้าหมายที่คาดว่าจะเกิดขึ้น |

## 3.3 การวิเคราะห์และออกแบบเครื่องมือสร้างกรณีทดสอบโดยอัตโนมัติ

3.3.1 การออกแบบโครงสร้างพื้นฐานของเครื่องมือและเทคโนโลยีที่ใช้ในการสร้างกรณีทดสอบ

รูปที่ 3.6 เป็นภาพรวมของกระบวนการทำการสร้างกรณีทดสอบของเครื่องมือที่ออกแบบ โดยจะมีผู้ทดสอบเป็นผู้ใช้เครื่องมือ ซึ่งจะพิจารณาแผนภาพคลาสและแผนภาพสเตทชาร์ตของคลาสนั้น เป็นข้อมูลนำเข้า ซึ่งจะต้องส่งออกข้อมูลให้อยู่ในรูปของแฟ้มเอ็กซ์-เอ็มไอก่อน ในส่วนของการทำงานของเครื่องมือจะนำแฟ้มข้อมูลเอ็กซ์เอสแอลทีกับแฟ้มข้อมูลเอ็กซ์เอ็มไอ แปลงเป็นกรณีทดสอบ ซึ่งเป็นผลลัพธ์ที่ต้องการ โดยจะใช้เอ็กซ์เอส-แอลทีโปรเซสเซอร์ในการแปลงเป็นกรณีทดสอบ โดยเครื่องมือนี้ได้ออกแบบมาให้ทำงานร่วมกับเครื่องมือสร้างแผนภาพยูเอ็มแอลที่มีอยู่ในปัจจุบัน โดยได้มีการเลือกใช้เทคโนโลยีในการแลกเปลี่ยนข้อมูลกับเครื่องมือสร้างแผนภาพยูเอ็มแอลและเทคโนโลยีที่นำมาอิมพลี-เมนต์การสร้างกรณีทดสอบที่มีความเหมาะสมกับงานวิจัยนี้ ดังในรายละเอียดต่อไปนี้




รูปที่ 3.6 ภาพรวมของกระบวนการทำการสร้างกรณีทดสอบของเครื่องมือที่ออกแบบ

3.3.1.1 การแปลงแผนภาพคลาสและแผนภาพสเตทชาร์ตเป็นเอกสารเอ็กซ์เอ็ม-แอล

สาเหตุที่ต้องมีการการแปลงแผนภาพคลาสและแผนภาพสเตทชาร์ตเป็นเอกสารเอ็กซ์เอ็มแอล เนื่องจากไม่สามารถนำแผนภาพทั้ง 2 ซึ่งเป็นแผนภาพยูเอ็มแอลมาใช้ในการสร้างกรณีทดสอบได้โดยตรง ผู้วิจัยจึงได้สำรวจวิธีการในการที่จะได้มาซึ่งข้อมูลจากแผนภาพยูเอ็มแอล พบว่าสามารถที่จะจัดเก็บแบบจำลองยูเอ็มแอลในรูปของแฟ้มข้อมูลชนิดเอ็กซ์เอ็มแอลตามมาตรฐานเอ็กซ์เอ็มไอซึ่งเป็นมาตรฐานที่สามารถทำให้การแลกเปลี่ยนข้อมูลแบบจำลองยูเอ็มแอลระหว่างเครื่องมือต่าง ๆ เป็นไปได้โดยสะดวก ซึ่งในงานวิจัยนี้ได้เลือกที่จะให้เครื่องมือสามารถทำงานกับแฟ้มข้อมูลที่ส่งออกจาก เรชันแนลโรส รุ่น 2002 (Rational Rose 2002) [25] เนื่องจากเป็นเครื่องมือสร้างแผนภาพยูเอ็มแอลที่นิยมใช้กันอย่างแพร่หลายและใช้เครื่องมือเสริม ยูนิซิสโรสเอ็กซ์เอ็มแอลทูล รุ่น 1.3.6 (Unisys Rose XML Tools 1.3.6) เป็นเครื่องมือในการส่งออกข้อมูลให้อยู่ในรูปเอกสารเอ็กซ์เอ็มไอ [26] ซึ่งกระบวนการในการส่งออกข้อมูลจากเรชันแนลโรส ได้อธิบายในภาคผนวก ค

3.3.1.2 การแปลงเอกสารเอ็กซ์เอ็มแอลเป็นเอกสารเอ็กซ์เอสแอลที

ในส่วนของการอิมพลีเมนต์การแปลงจากแผนภาพสเตทชาร์ตให้เป็นกรณีทดสอบที่ต้องการ ผู้วิจัยเลือกใช้ภาษาเอ็กซ์เอสแอลที เนื่องจากเป็นภาษาที่ออกแบบมาเพื่อให้มีความเหมาะสมกับการแปลงเอกสารเอ็กซ์เอ็มแอลโดยเฉพาะ โดยเอ็กซ์เอสแอลทีโปรเซสเซอร์ที่ใช้เป็นตัวแปลภาษานี้มีให้เลือกใช้อยู่หลายตัวบนหลายแพลตฟอร์ม

(Platform) ซึ่งจะทำให้เครื่องมือที่จะพัฒนาขึ้นมามีความสามารถในการปรับเปลี่ยนให้ทำงานบนแพลตฟอร์มต่าง ๆ (Portability) สูงตามไปด้วย โดยผู้วิจัยได้ทำการทดสอบเอ็กซ์เอสแอลทีโปรเซสเซอร์ที่ใช้กันอย่างแพร่หลายในปัจจุบันที่ทำตามเอกสารแนะนำเอ็กซ์เอสแอลที รุ่น 1.0 (XSLT 1.0 recommendation) [22] และมีส่วนขยายสำหรับทำการแปลงส่วนของต้นไม้ผลลัพธ์ (Result tree fragment) ไปเป็นเซตของโหนด (Node-set) ซึ่งจากผลการทดสอบผู้วิจัยได้เลือกใช้ไมโครซอฟท์เอ็กซ์เอ็มแอลคอร์เซอร์วิส รุ่น 4.0 (Microsoft XML Core Services 4.0) [19] เป็นเอ็กซ์เอสแอลทีโปรเซสเซอร์สำหรับเครื่องมือสำหรับสร้างกรณีทดสอบ

สำหรับขั้นตอนในการแปลงเอกสารเอ็กซ์เอ็มแอลให้เป็นเอกสารเอ็กซ์เอสแอลทีมีอยู่ด้วยกัน 5 ขั้นตอนดังนี้ (โดยตัวอย่างเอกสารเอ็กซ์เอ็มแอลและตัวอย่างเอ็กซ์เอสแอลทีที่สร้างได้จากขั้นตอนการแปลงแสดงในภาคผนวก ข)

(1) กำหนดชื่อสถานะ

ทำการตรวจสอบโหนดในเอกสารเอ็กซ์เอ็มแอลที่ชื่อ UML:CompositeState UML:Psedostate UML:SimpleState โดยพิจารณาแอทริบิวต์ภายในของแต่ละโหนดที่ชื่อว่า name จากนั้นทำการกำหนดชื่อสถานะในเอ็กซ์เอสแอลทีด้วยการอ้างถึงด้วยพาธดังนี้ ซึ่งจะให้ผลลัพธ์เป็นชื่อของสถานะของโหนดที่อ้างถึง

```
<xsl:template match="(//UML:Pseudostate) or (//UML:Simplestate) or
(//UML:Compositestate)">
    <xsl:value-of select="@name"/>
</xsl:template>
```

(2) กำหนดชนิดของสถานะ

ทำการตรวจสอบโหนดในเอกสารเอ็กซ์เอ็มแอลที่ชื่อ UML:Psedostate โดยพิจารณาแอทริบิวต์ที่ชื่อว่า kine จากนั้นทำการกำหนดชนิดของสถานะในเอ็กซ์เอสแอลทีด้วยการอ้างถึงด้วยพาธดังนี้ ซึ่งจะให้ผลลัพธ์เป็นชนิดของสถานะของโหนดที่อ้างถึง

```
<xsl:template match="//UML:Pseudostate">
    <xsl:value-of select="@kind"/>
</xsl:template>
```

(3) ตรวจสอบสถานะเริ่มต้นและสถานะเป้าหมาย

ทำการตรวจสอบโหนดในเอกสารเอ็กซ์เอ็มแอลที่ชื่อ UML:Transition โดยพิจารณาแอทริบิวต์ที่ชื่อ source และ target ซึ่งเป็นสถานะปัจจุบันและสถานะเป้าหมายของแต่ละทรานสิชันที่ทำการพิจารณา โดยค่าภายในของแอทริบิวต์สามารถอ้างไปยังค่าของแอทริบิวต์ xmi.id ของโหนด UML:CompositeState UML:SimpleState และ UML:Pseudostate ซึ่งสามารถกำหนดชื่อสถานะเริ่มต้นและสถานะเป้าหมายในเอ็กซ์เอสแอลทีด้วยการอ้างถึงพาธดังนี้

```
<xsl:template match="//UML:Transition">
    <xsl:variable name="sourceState" select="@source"/>
    <xsl:choose>
    <xsl:when test="//UML:Pseudostate[@xmi.id=$sourceState]">
            <xsl:value-of select="//UML:Pseudostate/@kind"/>
    </xsl:when>
    <xsl:when test="//UML:SimpleState[@xmi.id=$sourceState]">
            <xsl:value-of
            select="//UML:SimpleState[@xmi.id=$sourceState]
            /@name"/>
    </xsl:when>
    <xsl:otherwise>
            <xsl:value-of
            select="//UML:Compositestate[@xmi.id=$sourceState]
            /@name"/>
    </xsl:otherwise>
    </xsl:choose>
</xsl:template>
```

(4) กำหนดเหตุการณ์ที่เกิดขึ้น

ทำการตรวจสอบโหนดในเอกสารเอ็กซ์เอ็มแอลที่ชื่อ UML:Transition โดยพิจารณาแอทริบิวต์ที่ชื่อ trigger โดยค่าภายในของแอทริบิวต์สามารถอ้างไปยังค่าของแอทริบิวต์ xmi.id ของโหนด

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

UML:SignalEvent ซึ่งสามารถกำหนดเหตุการณ์ที่เกิดขึ้นในเอ็กซ์เอสแอลที ด้วยการอ้างถึงพาธดังนี้

```
<xsl:template match="//UML:Transition">
    <xsl:variable name="trigger" select="@trigger"/>
    <xsl:value-of select="UML:SignalEvent[@xmi.id=$trigger]
            //Parameter/@name"/>
</xsl:template>
```

(5) กำหนดเงื่อนไขการ์ด

ทำการตรวจสอบโหนดในเอกสารเอ็กซ์เอ็มแอลที่ชื่อ UML:BooleanExpression โดยพิจารณาแอทริบิวต์ที่ชื่อ body ซึ่งสามารถกำหนดเงื่อนไขการ์ดในเอ็กซ์เอสแอลทีด้วยการอ้างถึงพาธดังนี้

```
<xsl:template match="//UML:Transition">
    <xsl:value-of select=".//UML:BooleanExpression/@body"/>
</xsl:template>
```

3.3.1.3 สถาปัตยกรรมของการทำการแปลงจากแผนภาพสเตทชาร์ตให้เป็นกรณีทดสอบที่ต้องการ

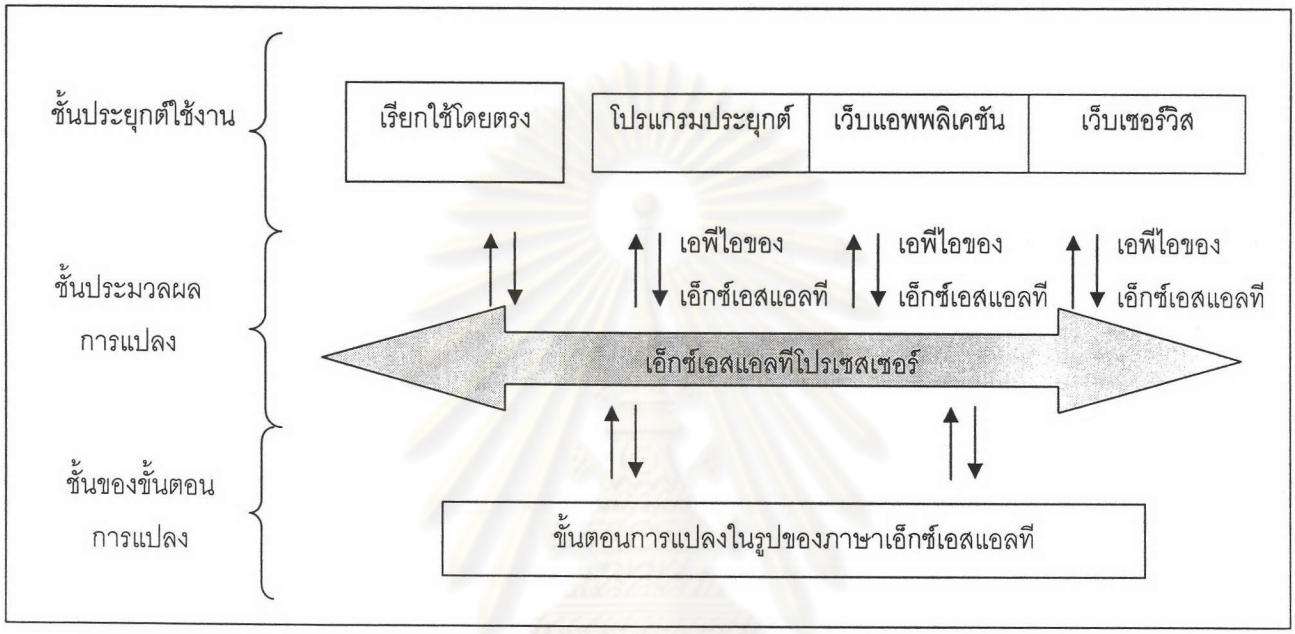
รูปที่ 3.7 แสดงให้เห็นถึงสถาปัตยกรรมของการทำการแปลงจากแผนภาพสเตท-ชาร์ตให้เป็นกรณีทดสอบที่ต้องการ จะประกอบไปด้วย 3 ชั้นคือ ชั้นล่างสุดเป็นชั้นของขั้นตอนการแปลง ซึ่งจะได้นำเอาขั้นตอนที่ได้ออกแบบขึ้นมาในงานวิจัยนี้มาระบุด้วยภาษาเอ็กซ์เอสแอลที ชั้นกลางจะเป็นชั้นประมวลผลการแปลง ซึ่งจะได้ใช้เอ็กซ์เอสแอล-ทีโปรเซสเซอร์เป็นตัวประมวลผลกฎการแปลงที่ได้ระบุไว้ ส่วนชั้นบนสุดจะเป็นชั้นการประยุกต์ใช้งาน ซึ่งผู้ใช้อาจสั่งให้เอ็กซ์เอสแอลทีโปรเซสเซอร์ทำการแปลงโดยตรง หรืออาจให้โปรแกรมประยุกต์ในรูปแบบต่าง ๆ สั่งให้เอ็กซ์เอสแอลทีโปรเซสเซอร์ทำการแปลงผ่านทางเอพีไอของเอ็กซ์เอสแอลทีโปรเซสเซอร์ก็ได้

3.3.2 การออกแบบเครื่องมือ

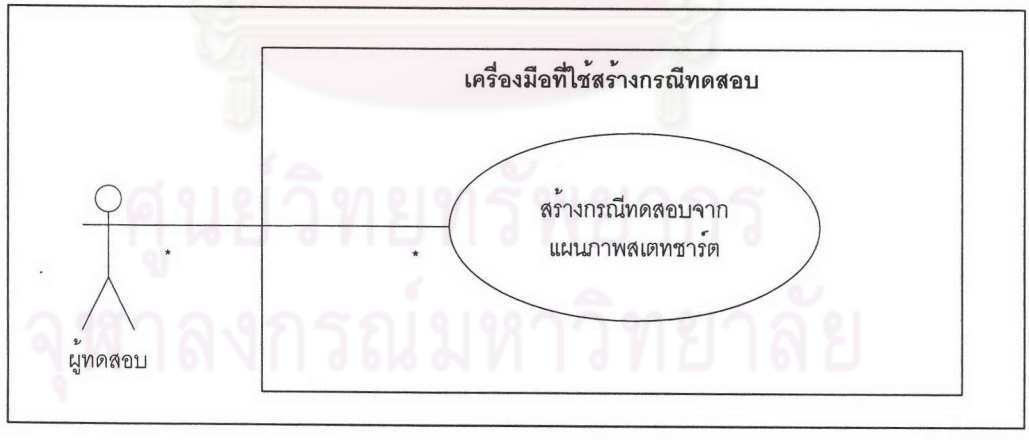
ในส่วนของการวิเคราะห์และออกแบบจะใช้แผนภาพยูสเคส แผนภาพคลาส แผนภาพกิจกรรมเป็นเครื่องมือในการใช้วิเคราะห์และออกแบบ รวมถึงการออกแบบโครงสร้างข้อมูลเอ็กซ์เอ็มแอล โครงสร้างการเรียกเทมเพลต และสถาปัตยกรรมของเครื่องมือดังมีรายละเอียดดังต่อไปนี้

3.3.2.1 แผนภาพยูสเคสของเครื่องมือที่สนับสนุนการสร้างกรณีทดสอบ

แผนภาพยูสเคสเป็นแผนภาพที่ใช้แสดงหน้าที่ต่างๆ ของระบบในมุมมองของผู้ใช้ ซึ่งจะแสดงให้เห็นว่าผู้ใช้สามารถทำอะไรกับระบบได้บ้าง โดยแผนภาพยูสเคสของเครื่องมือสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ตเป็นดังรูปที่ 3.8 โดยผู้ใช้สามารถให้เครื่องมือแปลงแผนภาพสเตทชาร์ตที่อยู่ในรูปของแฟ้มข้อมูลเอ็กซ์เอ็มไอไปเป็นกรณีทดสอบที่ต้องการ




รูปที่ 3.7 สถาปัตยกรรมของการทำการแปลงจากแผนภาพสเตทชาร์ตให้เป็นกรณีทดสอบที่ต้องการ




รูปที่ 3.8 แผนภาพยูสเคสของเครื่องมือสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต

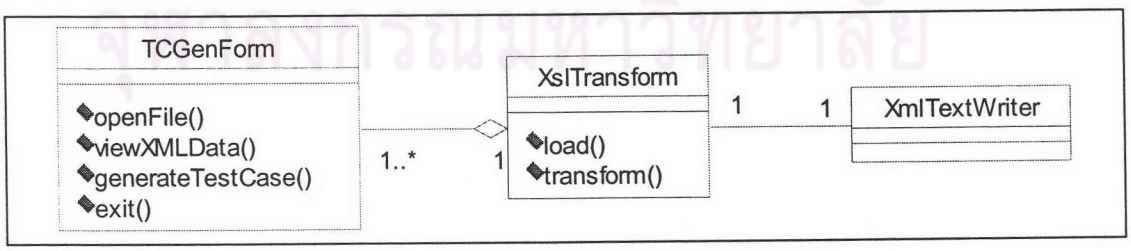
สำหรับรายละเอียดของยูสเคสเป็นดังตารางที่ 3.5 โดยรายละเอียดของกิจกรรมต่างๆ ที่เครื่องมือดังกล่าวต้องทำเพื่อให้ได้ผลการสร้างกรณีทดสอบที่ต้องการจะได้กล่าวถึงในหัวข้อถัดไป

ตารางที่ 3.5 รายละเอียดยูสเคสสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต

| | |
|---|---|
| ชื่อยูสเคส: | สร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต |
| แอกเตอร์ (Actors): | ผู้ทดสอบ |
| เป้าหมาย (Goal): | เพื่อสร้างกรณีทดสอบตามวิธีที่ใช้สร้างกรณีทดสอบ |
| ยูสเคสที่สัมพันธ์กัน (Related use case): | - |
| เงื่อนไขก่อนหน้า (Preconditions): | แผนภาพสเตทชาร์ตที่ต้องการสร้างกรณีทดสอบจะต้องทำการส่งออกข้อมูลให้อยู่ในรูปแฟ้มข้อมูลเอ็กซ์เอ็มไอ |
| ขั้นตอนการทำงาน (Steps): | 1. ผู้ทดสอบเลือกแฟ้มข้อมูลเอ็กซ์เอ็มไอมาเพื่อสร้างกรณีทดสอบ<br>2. ผู้ทดสอบทำการเลือกแสดงข้อมูลเอ็กซ์เอ็ม-แอล<br>3. ผู้ทดสอบทำการเลือกสร้างกรณีทดสอบ<br>4. เครื่องมือแสดงกรณีทดสอบที่สร้างได้ |
| เงื่อนไขหลังทำงาน: | กรณีทดสอบที่ได้ต้องครอบคลุมประพจน์ในแผนภาพสเตทชาร์ตที่นำมาพิจารณา |

3.3.2.2 แผนภาพคลาสในการสร้างกรณีทดสอบของเครื่องมือ

แผนภาพคลาสเป็นแผนภาพที่ใช้แสดงคลาส และโครงสร้างความสัมพันธ์ของแต่ละคลาสในระบบ โดยแผนภาพคลาสของเครื่องมือสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต เป็นดังรูปที่ 3.9




รูปที่ 3.9 แผนภาพคลาสของเครื่องมือสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต

จากแผนภาพคลาสในรูปที่ 3.9 แต่ละคลาสมีจุดประสงค์และรายละเอียดของคลาสดังตารางที่ 3.6

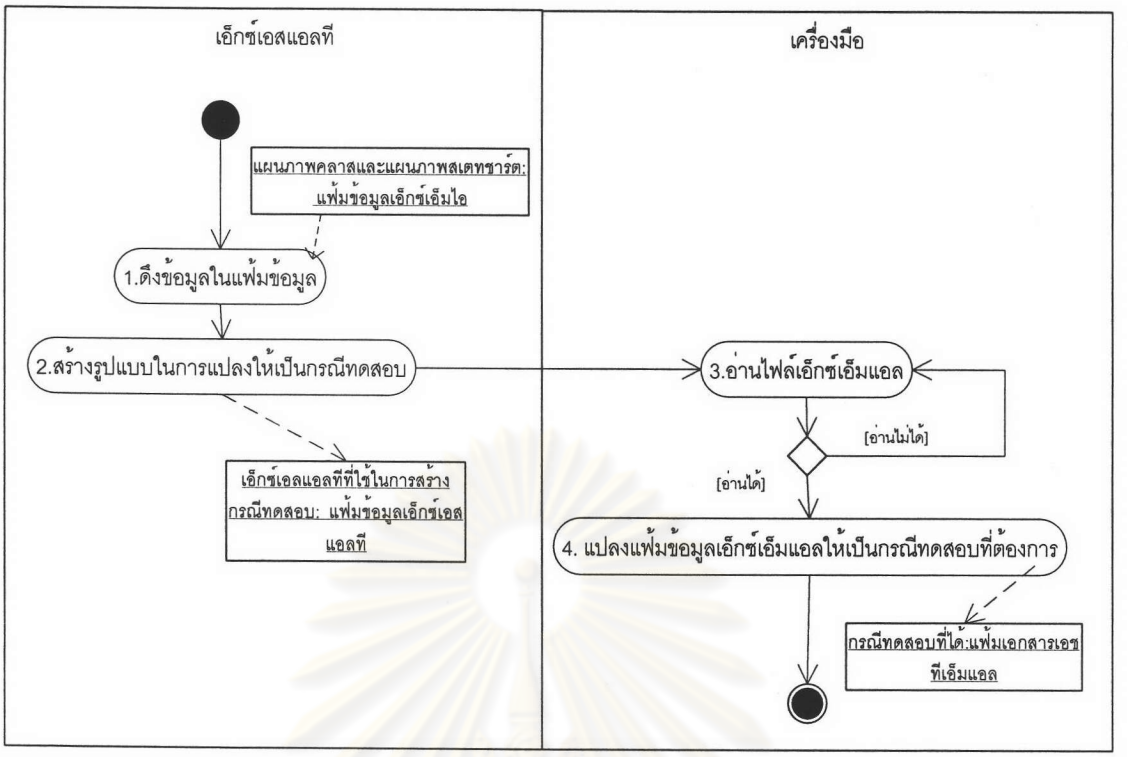
ตารางที่ 3.6 รายละเอียดของคลาสของเครื่องมือสร้างกรณีทดสอบ

| คลาส | คำอธิบาย |
|---|---|
| TCGenForm | ทำหน้าที่เป็นคลาสหลักของเครื่องมือ สามารถทำการเปิดแฟ้มข้อมูลเอ็กซ์เอ็มแอล เรียกดูแฟ้มข้อมูลเอ็กซ์เอ็มไอที่ต้องการสร้างกรณีทดสอบ สร้างกรณีทดสอบ เลือกออกจากโปรแกรม |
| XslTransform | ทำหน้าที่ในการแปลงข้อมูลจากแฟ้มข้อมูลเอ็กซ์เอ็มแอลของแผนภาพสเตทชาร์ตที่ต้องการไปเป็นกรณีทดสอบ |
| XmlTextWriter | ทำหน้าที่สร้างแฟ้มข้อมูลผลลัพธ์ของกรณีทดสอบที่แปลงได้ |

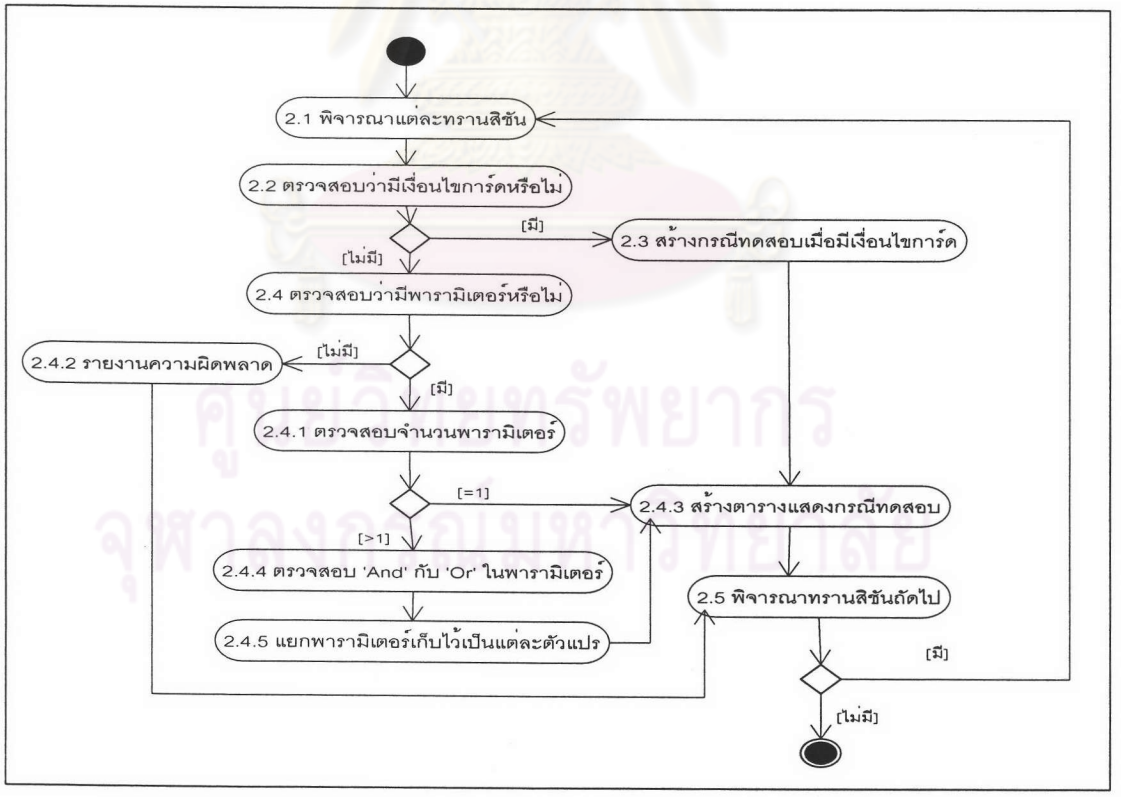
3.3.2.3 แผนภาพกิจกรรมในการสร้างกรณีทดสอบของเครื่องมือ

แผนภาพกิจกรรมเป็นแผนภาพที่ใช้แสดงลำดับขั้นตอนการทำงานจากกิจกรรมหนึ่งไปยังอีกกิจกรรมหนึ่ง เพื่อให้บรรลุเป้าหมายในการทำงานอย่างใดอย่างหนึ่งภายในระบบ สำหรับแผนภาพกิจกรรมของเครื่องมือสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ตที่จะแสดงต่อไปนี้จะอธิบายตามหน้าที่ที่แสดงไว้ในแผนภาพยูส-เคส

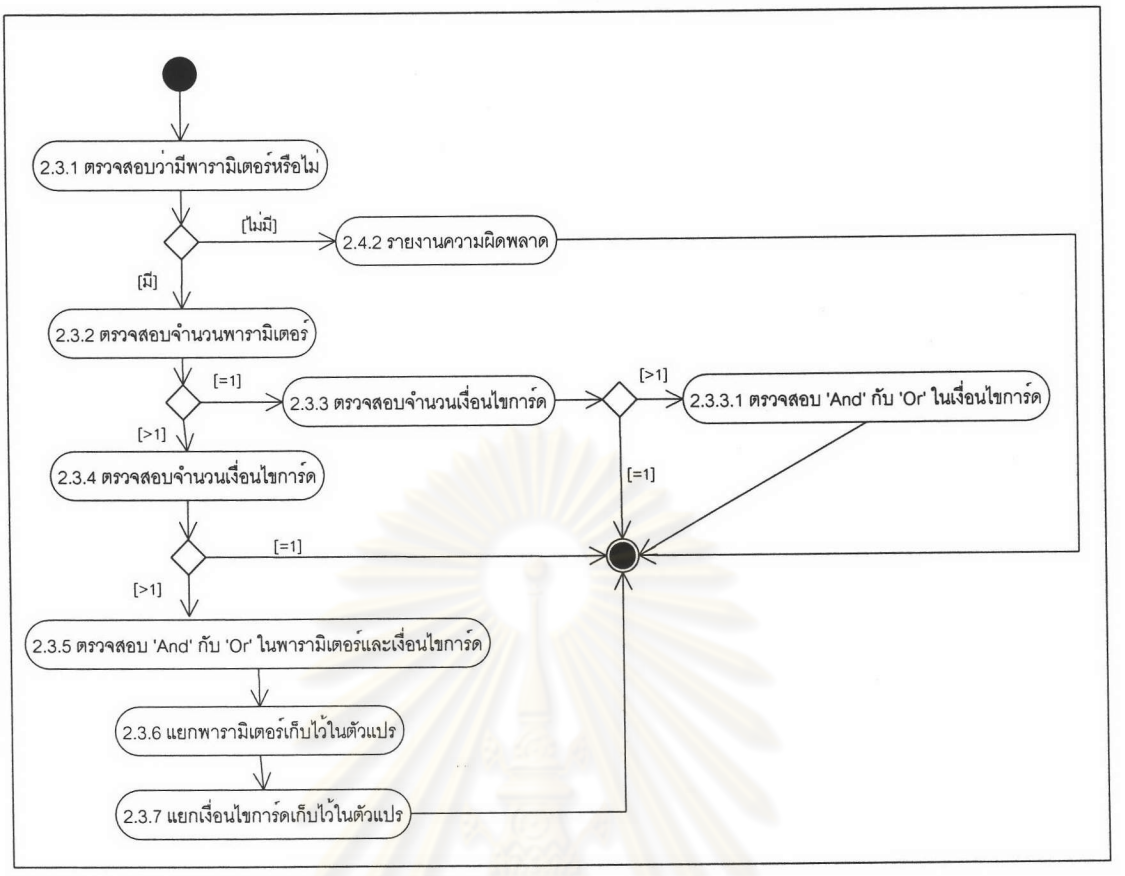
จากแผนภาพกิจกรรมรูปที่ 3.10 3.11 และ 3.12 เป็นการออกแบบกิจกรรมในการแปลงแฟ้มข้อมูลเอ็กซ์เอ็มไอของแผนภาพสเตทชาร์ตที่ต้องการไปเป็นกรณีทดสอบ ส่วนการสร้างรูปแบบในการแปลงให้เป็นกรณีทดสอบ ส่วนการสร้างกรณีทดสอบเมื่อมีเงื่อนไขการ์ด ตามลำดับ




รูปที่ 3.10 แผนภาพกิจกรรมของเครื่องมือสร้างกรณีทดสอบจากแผนภาพสเตทชาร์ต



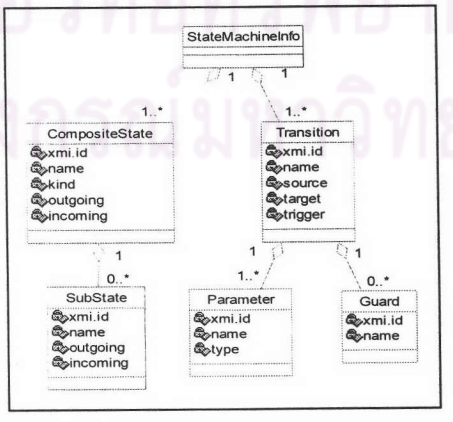

รูปที่ 3.11 ส่วนการสร้างรูปแบบในการแปลงให้เป็นกรณีทดสอบ




รูปที่ 3.12 ส่วนการสร้างกรณีทดสอบเมื่อมีเงื่อนไขการ์ด

3.3.2.4 การออกแบบโครงสร้างข้อมูลเอ็กซ์เอ็มแอลที่ใช้ในเครื่องมือ

ในหัวข้อนี้จะกล่าวถึงการออกแบบโครงสร้างข้อมูลเอ็กซ์เอ็มแอลที่ใช้ในเครื่องมือ ซึ่งจะเป็นข้อมูลที่เกี่ยวกับการเกิดขึ้นของเหตุการณ์ที่มีการกระตุ้น ณ ทรานสิชันหนึ่งๆ ซึ่งข้อมูลเหล่านี้ได้มาจากแผนภาพสเตทชาร์ต ดังรูปที่ 3.13 เป็นการแสดงโครงสร้างเอ็กซ์เอ็มแอลที่ใช้ในเครื่องมือนี้




รูปที่ 3.13 โครงสร้างข้อมูลเอ็กซ์เอ็มแอลที่ใช้ในเครื่องมือ

ตารางที่ 3.7 ถึง 3.12 เป็นการแสดงรายละเอียดของแต่ละส่วนย่อยในโครงสร้างเอ็กซ์เอ็มแอลที่ใช้ในเครื่องมือ

ตารางที่ 3.7 รายละเอียดของส่วนย่อย “StateMachineInfo”

| ชื่อส่วนย่อย | StateMachineInfo |
|---|---|
| คำอธิบาย | เป็นรากของข้อมูลสถานะ |

ตารางที่ 3.8 รายละเอียดของส่วนย่อย “Transition”

| ชื่อส่วนย่อย | Transition |
|---|---|
| คำอธิบาย | เก็บข้อมูลทรานสิชัน |
| แอทริบิวต์ | |
| xmi.id | รหัสเพื่อใช้ในการระบุตัวตนของทรานสิชัน |
| name | ชื่อของทรานสิชัน |
| source | รหัสของสถานะปัจจุบัน |
| target | รหัสของสถานะเป้าหมาย |
| trigger | รหัสของเหตุการณ์ที่มากระตุ้นเพื่อใช้อ้างถึงพารามิเตอร์ที่เกิดขึ้น |

ตารางที่ 3.9 รายละเอียดของส่วนย่อย “CompositeState”

| ชื่อส่วนย่อย | CompositeState |
|---|---|
| คำอธิบาย | เก็บข้อมูลของสถานะที่เกิดขึ้นในแผนภาพสเตทชาร์ต |
| แอทริบิวต์ | |
| xmi.id | รหัสเพื่อใช้ในการระบุตัวตนของสถานะ |
| name | ชื่อของสถานะนั้นๆ |
| kind | เก็บชนิดของสถานะ ว่าเป็นสถานะเริ่มต้น หรือสถานะสิ้นสุด |
| incoming | ชื่อของสถานะก่อนหน้า |
| outgoing | ชื่อของสถานะถัดไป |

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 3.10 รายละเอียดของส่วนย่อย “SubState”

| ชื่อส่วนย่อย | SubState |
|---|---|
| คำอธิบาย | เก็บข้อมูลของสถานะที่เกิดขึ้นในแผนภาพสเตทชาร์ตที่เป็นสถานะย่อย |
| แอทริบิวต์ | |
| xmi.id | รหัสเพื่อใช้ในการระบุตัวตนของสถานะ |
| name | ชื่อของสถานะย่อยนั้นๆ |
| incoming | ชื่อของสถานะก่อนหน้า |
| outgoing | ชื่อของสถานะถัดไป |

ตารางที่ 3.11 รายละเอียดของส่วนย่อย “Guard”

| ชื่อส่วนย่อย | Guard |
|---|---|
| คำอธิบาย | เก็บข้อมูลเงื่อนไขการ์ด |
| แอทริบิวต์ | |
| xmi.id | รหัสเพื่อใช้ในการระบุตัวตนของเงื่อนไขการ์ด |
| name | ชื่อของเงื่อนไขการ์ด |

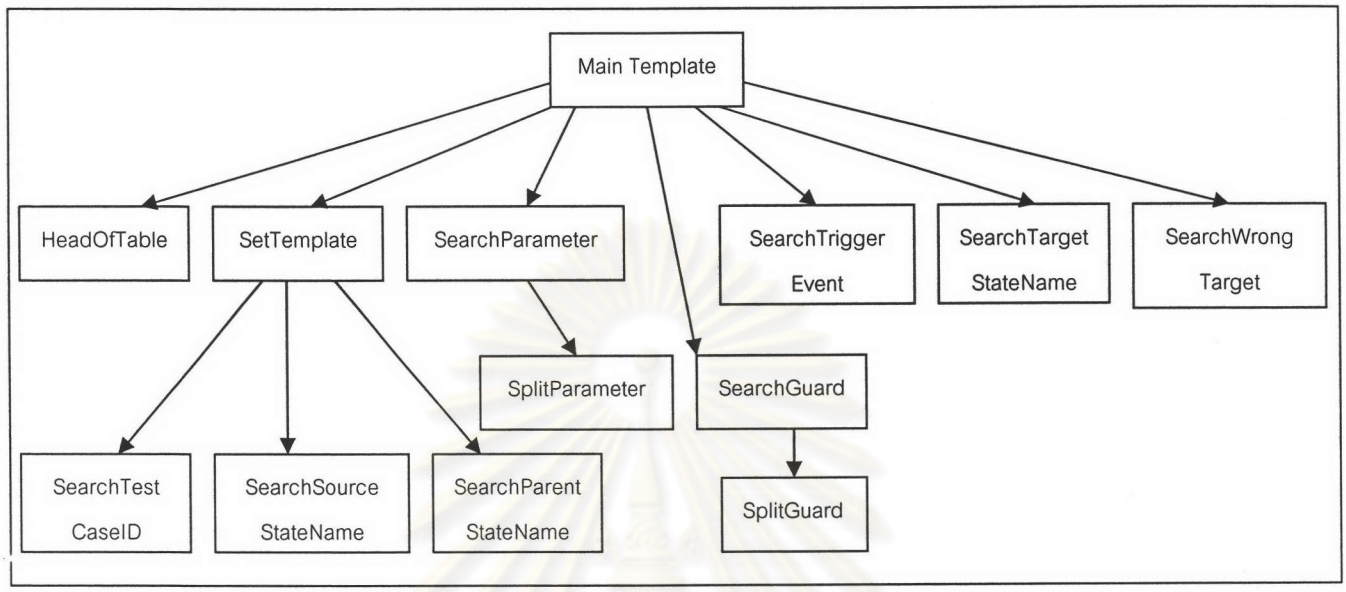
ตารางที่ 3.12 รายละเอียดของส่วนย่อย “Parameter”

| ชื่อส่วนย่อย | Parameter |
|---|---|
| คำอธิบาย | เก็บข้อมูลพารามิเตอร์ |
| แอทริบิวต์ | |
| xmi.id | รหัสเพื่อใช้ในการระบุตัวตนของพารามิเตอร์ |
| name | ชื่อของพารามิเตอร์ |
| type | ชนิดของพารามิเตอร์ |

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.3.2.5 การออกแบบโครงสร้างการเรียกเทมเพลตของเครื่องมือ

รูปที่ 3.14 เป็นการแสดงโครงสร้างการเรียกเทมเพลตของเครื่องมือ โดยหน้าที่ของแต่ละเทมเพลตแสดงอยู่ในตารางที่ 3.13



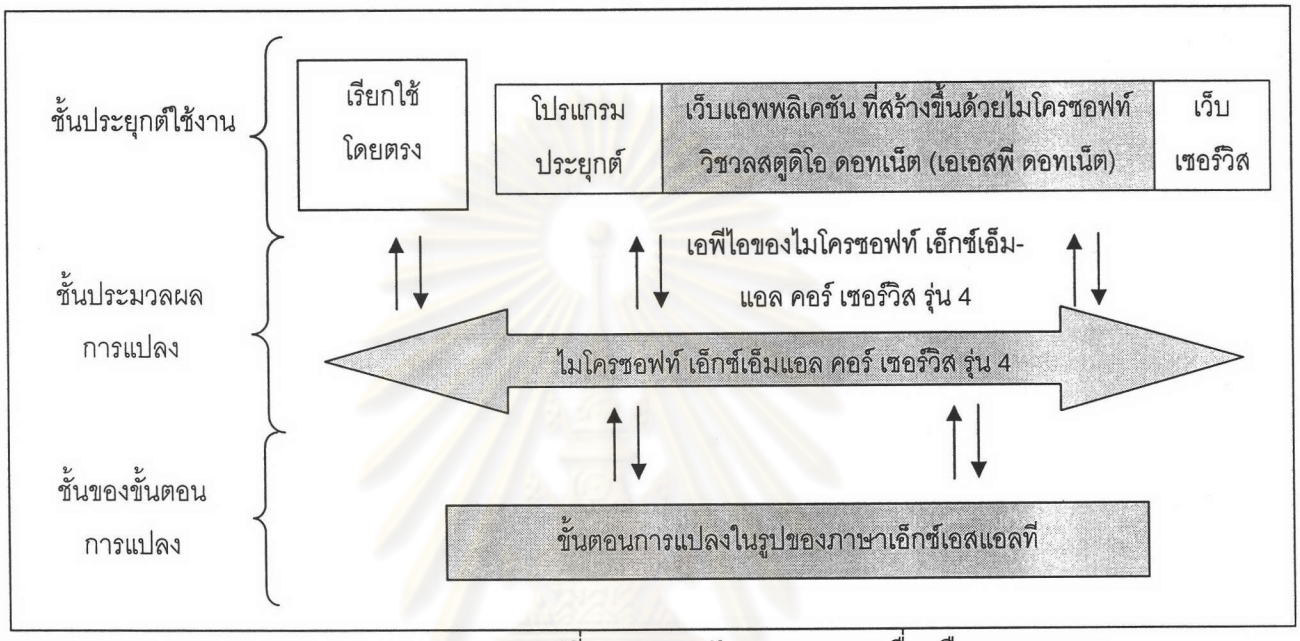

รูปที่ 3.14 โครงสร้างการเรียกเทมเพลตของเครื่องมือ

ตารางที่ 3.13 หน้าที่ของแต่ละเทมเพลต

| เทมเพลต | หน้าที่ |
|---|---|
| Main Template | สั่งการให้เกิดกระบวนการแปลงจากแผนภาพสเตทชาร์ตให้เป็นกรณีทดสอบทั้งหมด |
| HeadOfTable | สร้างหัวของตารางกรณีทดสอบ |
| SetTemplate | ตั้งค่าหมายเลขกรณีทดสอบ ชื่อสถานะแม่ ชื่อสถานะปัจจุบัน |
| SearchTestCaseID | สร้างหมายเลขกรณีทดสอบ |
| SearchSourcStateName | สร้างชื่อสถานะปัจจุบันที่ทำการทดสอบ |
| SearchParentStateName | สร้างชื่อสถานะแม่ กรณีที่มีสถานะย่อย |
| SearchParameter | ตั้งค่ากรณีทดสอบสำหรับพารามิเตอร์ |
| SearchGuard | ตั้งค่ากรณีทดสอบสำหรับเงื่อนไขการ์ด |
| SearchTriggerEvent | ตั้งค่ากรณีทดสอบสำหรับเหตุการณ์ที่มีตัวกระตุ้น |
| SearchTargetStateName | สร้างชื่อสถานะเป้าหมาย |
| SearchWrongTarget | ตั้งค่าสถานะเป้าหมายกรณีที่เป็นกรณีทดสอบที่ผิด |
| SplitGuard | แยกแต่ละเงื่อนไขการ์ดไปเก็บไว้ในแต่ละตัวแปร |
| SpllitParameter | แยกแต่ละพารามิเตอร์ไปเก็บไว้ในแต่ละตัวแปร |

3.3.2.6 การออกแบบสถาปัตยกรรมของเครื่องมือ

รูปที่ 3.15 เป็นการแสดงสถาปัตยกรรมของเครื่องมือ ซึ่งมีการใช้ไมโครซอฟท์เอ็กซ์เอ็มแอลคอร์เซอร์วิส รุ่น 4.0 ให้ทำหน้าที่เป็น เอ็กซ์เอสแอลที-โปรเซสเซอร์ โดยใช้ภาษาเอเอสพี ดอทเน็ต (ASP .NET) ในการสร้างส่วนต่อประสานกับผู้ใช้




รูปที่ 3.15 สถาปัตยกรรมของเครื่องมือ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย