

บทที่ 3 วิธีดำเนินการวิจัย

การทดสอบเว็บเซอร์วิสเป็นกระบวนการหนึ่งที่สำคัญ เพื่อทำให้เกิดความเชื่อมั่นในการใช้งานของลูกค้า และสามารถให้บริการได้อย่างมีประสิทธิภาพ ซึ่งงานวิจัยส่วนมากจะสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสจากดับเบิลยูเอสดีแอล ทำให้เกิดข้อจำกัดในการสร้างกรณีทดสอบต่างๆ เช่น การวิเคราะห์เงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส เป็นต้น ดังนั้น งานวิจัยนี้จึงนำเสนอวิธีการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสจากดับเบิลยูเอสดีแอล-เอส และเอสดับเบิลยูอาร์แอล ด้วยตารางตัดสินใจ โดยข้อมูลที่ได้จากเอกสารดับเบิลยูเอสดีแอล-เอสจะแตกต่างกับเอกสารดับเบิลยูเอสดีแอล คือ ข้อมูลของเอกสารดับเบิลยูเอสดีแอล-เอสสามารถให้คำอธิบายเว็บเซอร์วิสที่ประกอบด้วยเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิส โดยเงื่อนไขต่างๆ จะถูกนิยามในเอกสารเอสดับเบิลยูอาร์แอล ทำให้สามารถสร้างกรณีทดสอบที่ครอบคลุมค่าคาดหวังของการทดสอบเว็บเซอร์วิสได้ครบทุกกรณี โดยแผนภาพขั้นตอนการสร้างกรณีทดสอบของงานวิจัยนี้แสดงดังรูปที่ 3.1 ซึ่งข้อกำหนดของข้อมูลนำเข้ามีรายละเอียดอธิบายในหัวข้อ 3.1 และรายละเอียดของขั้นตอนต่างๆ แสดงในหัวข้อ 3.2

3.1. ข้อกำหนดของข้อมูลนำเข้า

ข้อมูลนำเข้าและข้อกำหนดของข้อมูลของงานวิจัยนี้ ประกอบด้วย

3.1.1. เอกสารดับเบิลยูเอสดีแอล-เอส

เอกสารดับเบิลยูเอสดีแอล-เอส เป็นเอกสารที่แสดงรายละเอียดของตัวแปร โอเปอเรชัน และเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส รวมทั้งเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิส โดยเงื่อนไขทั้งสองในเอกสารนี้จะแสดงรายละเอียดของเงื่อนไขโดยการอ้างอิงกฎที่กำหนดในเอกสารเอสดับเบิลยูอาร์แอล

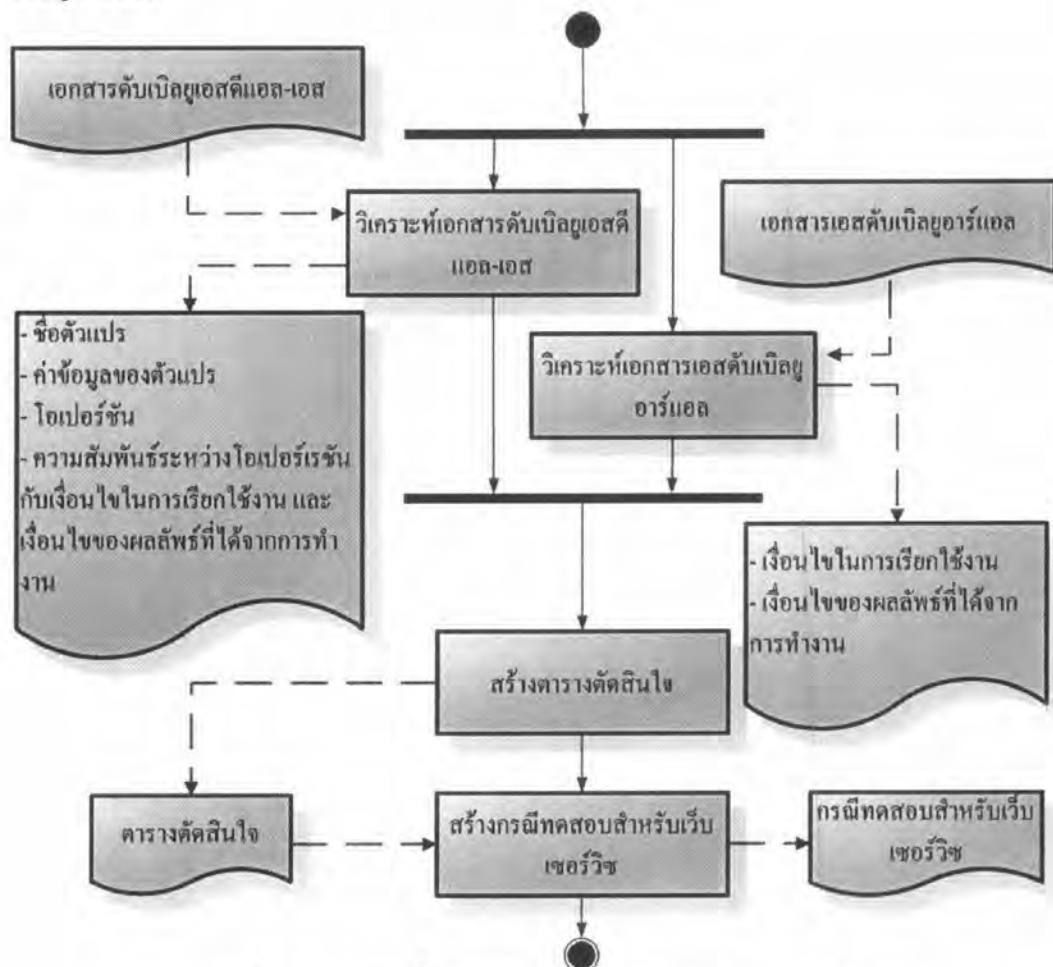
จากรูปที่ 3.2 ซึ่งเป็นเอกสารดับเบิลยูเอสดีแอล-เอส โดยแสดงรายละเอียดของเว็บเซอร์วิส Triangle ที่มีโอเปอเรชันชื่อว่า TriangleType ให้บริการตรวจสอบชนิดของรูปสามเหลี่ยม ซึ่งประกอบด้วยข้อมูลนำเข้า 3 ตัวแปร ได้แก่ ตัวแปร x ตัวแปร y และตัวแปร z โดยตัวแปรทั้ง 3 ตัวแปรแทนค่าความยาวของในแต่ละด้านของรูปสามเหลี่ยมและมีชนิดของข้อมูลเป็นตัวแปรประเภทจำนวนเต็ม และข้อมูลผลลัพธ์ ได้แก่ TriangleResult ซึ่งเป็นตัวแปรประเภทสาย

อักขระ สำหรับใช้ส่งค่ากลับคืนมาเพื่อบอกว่าค่าของข้อมูลนำเข้าที่ได้รับมานั้น จะแทนด้านของรูปสามเหลี่ยมประเภทอะไร ซึ่งประเภทของรูปสามเหลี่ยมจะมีเงื่อนไขสอดคล้องกับตารางที่ 3.1

3.1.2. เอกสารเอสดับเบิลยูอาร์แอล

เอกสารเอสดับเบิลยูอาร์แอล เป็นเอกสารที่แสดงรายละเอียดของเงื่อนไขต่างๆ สำหรับเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิส ที่ถูกใช้ในเอกสารดับเบิลยูเอสดีแอล-เอส โดยเอกสารเอสดับเบิลยูอาร์แอลจะต้องอยู่ในรูปแบบเอกซ์เอ็มแอล และผู้ใช้งานต้องเขียนประโยคเงื่อนไขสำหรับการเรียกใช้งานเว็บเซอร์วิสและผลลัพธ์ในการทำงานของเว็บเซอร์วิส มีรายละเอียดดังนี้

3.1.2.1. ตัวแปรที่เป็นพารามิเตอร์ (Parameter) ของโอเปอเรชันต้องกำหนดค่าข้อมูลในเอกสารดับเบิลยูเอสดีแอล-เอสก่อนนำมาใช้ในประโยคเงื่อนไข และตัวแปรที่ใช้สำหรับเก็บค่าที่ได้จากคำนวณเครื่องหมายทางคณิตศาสตร์ ต้องกำหนดตัวแปรในเอกสารเอสดับเบิลยูอาร์แอล



รูปที่ 3.1 แผนภาพขั้นตอนการสร้างกรณีทดสอบของงานวิจัย

```

<?xml version="1.0" encoding="utf-8"?>
<definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:tns="http://tempuri.org/"
  xmlns:s="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  targetNamespace="http://tempuri.org/"
  xmlns:wsssem="http://sdls.cs.uga.edu/projects/meteor-s/wsdl-s/examples/WSSemantics.xsd">
  <types>
    <s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <s:element name="Triangle">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="x" type="s:int" />
            <s:element minOccurs="1" maxOccurs="1" name="y" type="s:int" />
            <s:element minOccurs="1" maxOccurs="1" name="z" type="s:int" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="TriangleResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1"
              name="TriangleResult" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:schema>
  </types>
  <message name="TriangleSoapIn">
    <part name="parameters" element="tns:Triangle" />
  </message>

```

ข้อมูลนำเข้า

ผลลัพธ์

รูปที่ 3.2. รายละเอียดของไฟล์ Example.wsdl

```

<message name="TriangleSoapOut">
  <part name="parameters" element="tns:TriangleResponse" />
</message>
<portType name="Service1Soap">
  <operation name="TriangleType">
    <input message="tns:TriangleSoapIn" />
    <output message="tns:TriangleSoapOut" />
    <wssem:precondition name="TrianglePrecond"
      wssem:modelReference="http://localhost/example.swrlx#inputRule"/>
    <wssem:effect name="ResultEffect"
      wssem:modelReference="http://localhost/example.swrlx#outputRule1
        http://localhost/example.swrlx#outputRule2
        http://localhost/example.swrlx#outputRule3
        http://localhost/example.swrlx#outputRule4
        http://localhost/example.swrlx#outputRule5
        http://localhost/example.swrlx#outputRule6"/>
  </operation>
</portType>
<binding name="Service1Soap" type="tns:Service1Soap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <operation name=" TriangleType">
    <soap:operation soapAction="http://tempuri.org/Triangle" style="document" />
    <input>
      <soap:body use="literal" />
    </input>
    <output>
      <soap:body use="literal" />
    </output>
  </operation>
</binding>
<binding name="Service1Soap12" type="tns:Service1Soap">
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <operation name="TriangleType">
    <soap12:operation soapAction="http://tempuri.org/Triangle" style="document" />
    <input>
      <soap12:body use="literal" />
    </input>
    <output>
      <soap12:body use="literal" />
    </output>
  </operation>
</binding>

```

Precondition

Post-condition

นิยามของโพรเซสเรชัน

รูปที่ 3.2. รายละเอียดของไฟล์ Example.wsdl (ต่อ)

```

<service name="Service1">
  <port name="Service1Soap" binding="tns:Service1Soap">
    <soap:address location="http://localhost:3269/Service1.asmx" />
  </port>
  <port name="Service1Soap12" binding="tns:Service1Soap12">
    <soap12:address location="http://localhost:3269/Service1.asmx" />
  </port>
</service>
</definitions>

```

รูปที่ 3.2. รายละเอียดของไฟล์ Example.wsdlis (ต่อ)

ตารางที่ 3.1. เงื่อนไขของแต่ละประเภทของรูปสามเหลี่ยม

ประโยคเงื่อนไข	ประเภทของรูปสามเหลี่ยม
$x == y$ and $y == z$	An Equilateral Triangle
$x <> y$ and $x <> z$ and $y <> z$	A Scalene Triangle
$x == y$ and $x <> z$	An Isosoles Triangle
$x == z$ and $x <> y$	An Isosoles Triangle
$y == z$ and $x <> z$	An Isosoles Triangle

3.1.2.2. ผู้ใช้งานสามารถใช้แฮนด์เบิลยูอาร์แอลบิวอินได้เพียง 1 โมดูลในแต่ละพจน์ของประโยคเงื่อนไข สำหรับการคำนวณค่าทางคณิตศาสตร์เท่านั้น ตัวอย่างประโยคเงื่อนไขที่สามารถเขียนอยู่ในรูปแบบแฮนด์เบิลยูอาร์แอล เช่น $x < (y + z)$

3.1.2.3. แฮนด์เบิลยูอาร์แอลบิวอินที่งานวิจัยนี้รองรับ มีรายละเอียดดังนี้

3.1.2.3.1. swrlb:equal ใช้สำหรับการเปรียบเทียบค่าของอาร์กิวเมนต์แรกมีค่าเท่ากับอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.3.2. swrlb:notEqual ใช้สำหรับการเปรียบเทียบค่าของอาร์กิวเมนต์แรกมีค่าไม่เท่ากับอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.3.3. swrlb:lessThan ใช้สำหรับการเปรียบเทียบค่าของอาร์กิวเมนต์แรกมีค่าน้อยกว่าอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.3.4. swrlb:lessThanOrEqual ใช้สำหรับการเปรียบเทียบค่าของอาร์กิวเมนต์แรกมีค่าน้อยกว่าหรือเท่ากับอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.3.5. swrlb:greaterThan ใช้สำหรับการเปรียบเทียบค่าของอาร์กิวเมนต์แรกมีค่ามากกว่าอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.3.6. `swrlb:greaterThanOrEqual` ใช้สำหรับการเปรียบเทียบค่าของอาร์กิวเมนต์แรกมีค่ามากกว่าหรือเท่ากับอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.3.7. `swrlb:add` ใช้แทนเครื่องหมายดำเนินการบวก โดยอาร์กิวเมนต์แรก จะมีค่าเท่ากับผลรวมของอาร์กิวเมนต์ที่สองจนถึงอาร์กิวเมนต์ตัวสุดท้าย

3.1.2.3.8. `swrlb:subtract` ใช้แทนเครื่องหมายดำเนินการลบ โดยอาร์กิวเมนต์แรก จะมีค่าเท่ากับค่าความแตกต่างระหว่างของอาร์กิวเมนต์ที่สองกับอาร์กิวเมนต์ที่สาม

3.1.2.3.9. `swrlb:multiply` ใช้แทนเครื่องหมายดำเนินการคูณ โดยอาร์กิวเมนต์แรก จะมีค่าเท่ากับผลคูณของอาร์กิวเมนต์ที่สองจนถึงอาร์กิวเมนต์ตัวสุดท้าย

3.1.2.3.10. `swrlb:divide` ใช้แทนเครื่องหมายดำเนินการหาร โดยอาร์กิวเมนต์แรก จะมีค่าเท่ากับผลหารของอาร์กิวเมนต์ที่สองหารด้วยอาร์กิวเมนต์ที่สาม

3.1.2.3.11. `swrlb:stringLength` ใช้สำหรับเปรียบเทียบอาร์กิวเมนต์แรกมีค่ายาวเท่ากับอาร์กิวเมนต์ที่สองหรือไม่ ซึ่งอาร์กิวเมนต์ทั้งสองต้องเป็นสายอักขระเท่านั้น

3.1.2.3.12. `swrlb:dateTime` ใช้สำหรับตรวจสอบวันเวลาของอาร์กิวเมนต์แรก โดยอาร์กิวเมนต์ที่สองถึงแปด จะแทนปี เดือน วัน ชั่วโมง นาที วินาที และเขตเวลาตามลำดับ

3.1.2.3.13. `swrlb:date` ใช้สำหรับตรวจสอบวันของอาร์กิวเมนต์แรก โดยอาร์กิวเมนต์ที่สองถึงห้า จะแทนปี เดือน วัน และเขตเวลา (Time zone) ตามลำดับ

3.1.2.3.14. `swrlb:time` ใช้สำหรับตรวจสอบเวลาของอาร์กิวเมนต์แรก โดยอาร์กิวเมนต์ที่สองถึงห้า จะแทนชั่วโมง นาที วินาที และเขตเวลา ตามลำดับ

3.1.2.3.15. `swrlb:listConcat` ใช้สำหรับกำหนดรายการข้อมูล โดยอาร์กิวเมนต์แรกเป็นชื่อของรายการ และอาร์กิวเมนต์ที่สองถึงอาร์กิวเมนต์สุดท้ายเป็นข้อมูลในรายการ

3.1.2.3.16. `swrlb:member` ใช้สำหรับตรวจสอบข้อมูลของอาร์กิวเมนต์แรก เป็นข้อมูลที่อยู่ในรายการที่แสดงด้วยอาร์กิวเมนต์ที่สองหรือไม่

3.1.2.4. ประเภทข้อมูลของตัวแปรตามเอกซ์เอ็มแอลสคีมาที่สามารถใช้ในงานวิจัยนี้ได้แก่ ข้อมูลประเภทสายอักขระ (`xsd:string`) ข้อมูลประเภทจำนวนเต็ม (`xsd:integer`, `xsd:long`, `xsd:int`, `xsd:short`) ข้อมูลประเภทตรรกะ (`xsd:boolean`) ข้อมูลประเภทเลขจำนวนจริง (`xsd:float`, `xsd:double`, `xsd:decimal`) และข้อมูลประเภทวันที่ (`xsd:time`, `xsd:date`, `xsd:dateTime`)

จากรูปที่ 3.3 ซึ่งเป็นเอกสารเอสดับเบิลยูอาร์แอล โดยแสดงรายละเอียดของเอกสารเอสดับเบิลยูอาร์แอลสำหรับนิยามกฎต่างๆ เพื่อใช้อ้างอิงจากเอกสารดับเบิลยูเอสดีแอลเอสของเว็บเซอร์วิส Triangle จากการวิเคราะห์เอกสารเอสดับเบิลยูอาร์แอลสามารถสรุปรายละเอียดของกฎที่นิยามได้ดังตารางที่ 3.2

3.2. ขั้นตอนการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิส

3.2.1. การวิเคราะห์เอกสารดับเบิลยูเอสดีแอลเอสและเอสดับเบิลยูอาร์แอล

เครื่องมือจะเริ่มวิเคราะห์จากเอกสารดับเบิลยูเอสดีแอลเอส และเอสดับเบิลยูอาร์แอลที่ได้รับจากผู้ใช้งาน โดยแยกข้อมูลของเอกสารทั้งสองออกเป็นข้อมูล 6 ประเภท ได้แก่ ชื่อตัวแปร ช่วงของค่าข้อมูลของตัวแปร โอเปอเรชัน เงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิส รวมทั้งความสัมพันธ์ระหว่างโอเปอเรชันกับเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิสและเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิส ซึ่งข้อมูลที่วิเคราะห์ได้จากเอสดับเบิลยูอาร์แอล คือ เงื่อนไขต่างๆ ที่ใช้สำหรับเรียกใช้งานเว็บเซอร์วิสและผลลัพธ์ในการทำงาน และข้อมูลที่วิเคราะห์ได้จากดับเบิลยูเอสดีแอลเอส คือ ชื่อตัวแปร ช่วงของค่าข้อมูลของตัวแปร โอเปอเรชัน และความสัมพันธ์ระหว่างโอเปอเรชันกับเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิส หลังจากนั้นเครื่องมือจะทำการเปลี่ยนแปลงประโยคเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิสที่แสดงอยู่ในรูปแบบเอสดับเบิลยูอาร์แอลเป็นรูปแบบประโยคเงื่อนไขทางคณิตศาสตร์ โดยรหัสคำสั่งเทียม (pseudo code) สำหรับวิเคราะห์เอกสารดับเบิลยูเอสดีแอลเอส และเอกสารเอสดับเบิลยูอาร์แอล แสดงดังรูปที่ 3.4 และ รูปที่ 3.5 ตามลำดับ

```
<definitions xmlns:swrlx="http://www.w3.org/2003/11/swrlx"
  xmlns:rullem1="http://www.w3.org/2003/11/ruleml"
  xmlns:owlx="http://www.w3.org/2003/05/owl-xml"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<ruleml:imp rdf:ID="inputRule">
  <ruleml:_rlab ruleml:href="#inputRule"/>
  <ruleml:_body>
```

รูปที่ 3.3. รายละเอียดของไฟล์ Example.swrls


```

<swrlx:builtinAtom swrlx:builtin="swrlb:greaterThan">
  <ruleml:var>x</ruleml:var>
  <owlx:DataValue owlx:datatype="xsd:int">0</owlx:DataValue>
</swrlx:builtinAtom>
} x > 0

<swrlx:builtinAtom swrlx:builtin="swrlb:notEqual">
  <ruleml:var>x</ruleml:var>
  <ruleml:var>z</ruleml:var>
</swrlx:builtinAtom>
} x != z

<swrlx:builtinAtom swrlx:builtin="swrlb:notEqual">
  <ruleml:var>y</ruleml:var>
  <ruleml:var>z</ruleml:var>
</swrlx:builtinAtom>
} y <> z

</ruleml!:_body>
<ruleml!:_head>
<swrlx:datavaluedPropertyAtom swrlx:property="is">
  <ruleml:var>TriangleResult</ruleml:var>
  <owlx:DataValue owlx:datatype="xsd:string">a Scalene Triangle </owlx:DataValue>
</swrlx:datavaluedPropertyAtom>
} TriangleResult is a Scalene Triangle
</ruleml!:_head>
</ruleml:imp>

<ruleml:imp rdf:ID="outputRule3">
  <ruleml:_riab ruleml:href="#outputRule3"/>
  <ruleml:_body>
    <swrlx:builtinAtom swrlx:builtin="swrlb:add">
      <ruleml:var>sumOfY_Z</ruleml:var>
      <ruleml:var>y</ruleml:var>
      <ruleml:var>z</ruleml:var>
    </swrlx:builtinAtom>
    <swrlx:builtinAtom swrlx:builtin="swrlb:add">
      <ruleml:var>sumOfX_Z</ruleml:var>
      <ruleml:var>x</ruleml:var>
      <ruleml:var>z</ruleml:var>
    </swrlx:builtinAtom>
    <swrlx:builtinAtom swrlx:builtin="swrlb:add">
      <ruleml:var>sumOfX_Y</ruleml:var>
      <ruleml:var>x</ruleml:var>
      <ruleml:var>y</ruleml:var>
    </swrlx:builtinAtom>
    <swrlx:builtinAtom swrlx:builtin="swrlb:lessThan">
      <ruleml:var>x</ruleml:var>
      <ruleml:var>sumOfY_Z</ruleml:var>
    </swrlx:builtinAtom>
  } sumOfY_Z = y + z
  } sumOfX_Z = x + z
  } sumOfX_Y = x + y
  } x < sumOfY_Z

```

รูปที่ 3.3. รายละเอียดของไฟล์ Example.swrls (ต่อ)


```

<swrl:builtInAtom swrl:builtIn="swrlb:lessThan">
  <ruleml:var>y</ruleml:var>
  <ruleml:var>sumOfX_Z</ruleml:var>
</swrl:builtInAtom>
<swrl:builtInAtom swrl:builtIn="swrlb:lessThan">
  <ruleml:var>z</ruleml:var>
  <ruleml:var>sumOfX_Y</ruleml:var>
</swrl:builtInAtom>
</ruleml:_body>
<ruleml:_head>
<swrl:dataValuedPropertyAtom swrl:property="is">
  <ruleml:var>TriangleResult</ruleml:var>
  <owlx:DataValue owlx:datatype="xsd:string">a Triangle</owlx:DataValue>
</swrl:dataValuedPropertyAtom>
</ruleml:_head>
</ruleml:imp>
<ruleml:imp rdf:ID="outputRule4">
  <ruleml:_tab ruleml:href="#outputRule4"/>
  <ruleml:_body>
  <swrl:builtInAtom swrl:builtIn="swrlb:equal">
    <ruleml:var>x</ruleml:var>
    <ruleml:var>y</ruleml:var>
  </swrl:builtInAtom>
  <swrl:builtInAtom swrl:builtIn="swrlb:notEqual">
    <ruleml:var>x</ruleml:var>
    <ruleml:var>z</ruleml:var>
  </swrl:builtInAtom>
  </ruleml:_body>
  <ruleml:_head>
  <swrl:dataValuedPropertyAtom swrl:property="is">
    <ruleml:var>TriangleResult</ruleml:var>
    <owlx:DataValue owlx:datatype="xsd:string">an Isosceles Triangle</owlx:DataValue>
  </swrl:dataValuedPropertyAtom>
  </ruleml:_head>
</ruleml:imp>

```

$y < \text{sumOfX}_Z$
 $z < \text{sumOfX}_Y$
TriangleResult is a Triangle
 $x = y$
 $x \neq z$
TriangleResult is a Isosceles Triangle

รูปที่ 3.3. รายละเอียดของไฟล์ Example.swrls (ต่อ)

```

<ruleml:imp rdf:ID="outputRule5">
  <ruleml:_riab ruleml:href="#outputRule5"/>
  <ruleml:_body>
    <swrlx:builtinAtom swrlx:builtin="swrlb:equal">
      <ruleml:var>x</ruleml:var>
      <ruleml:var>z</ruleml:var>
    </swrlx:builtinAtom>
    <swrlx:builtinAtom swrlx:builtin="swrlb:notEqual">
      <ruleml:var>x</ruleml:var>
      <ruleml:var>y</ruleml:var>
    </swrlx:builtinAtom>
  </ruleml:_body>
  <ruleml:_head>
    <swrlx:datavaluedPropertyAtom swrlx:property="is">
      <ruleml:var>TriangleResult</ruleml:var>
      <owlx:DataValue owl:datatype="xsd:string">an Isosceles Triangle</owlx:DataValue>
    </swrlx:datavaluedPropertyAtom>
  </ruleml:_head>
</ruleml:imp>
<ruleml:imp rdf:ID="outputRule6">
  <ruleml:_riab ruleml:href="#outputRule6"/>
  <ruleml:_body>
    <swrlx:builtinAtom swrlx:builtin="swrlb:equal">
      <ruleml:var>y</ruleml:var>
      <ruleml:var>z</ruleml:var>
    </swrlx:builtinAtom>
    <swrlx:builtinAtom swrlx:builtin="swrlb:notEqual">
      <ruleml:var>x</ruleml:var>
      <ruleml:var>z</ruleml:var>
    </swrlx:builtinAtom>
  </ruleml:_body>
  <ruleml:_head>
    <swrlx:datavaluedPropertyAtom swrlx:property="is">
      <ruleml:var>TriangleResult</ruleml:var>
      <owlx:DataValue owl:datatype="xsd:string">an Isosceles Triangle</owlx:DataValue>
    </swrlx:datavaluedPropertyAtom>
  </ruleml:_head>
</ruleml:imp>
</definitions>

```

Diagram illustrating the structure of the SWRL rules and their logical conditions:

- Rule 5 (outputRule5):**
 - Condition 1: $x = z$ (represented by a curly brace on the right).
 - Condition 2: $x \neq y$ (represented by a curly brace on the right).
 - Head: TriangleResult is an Isosceles Triangle (represented by a curly brace on the right).
- Rule 6 (outputRule6):**
 - Condition 1: $y = z$ (represented by a curly brace on the right).
 - Condition 2: $x \neq z$ (represented by a curly brace on the right).
 - Head: TriangleResult is an Isosceles Triangle (represented by a curly brace on the right).

รูปที่ 3.3. รายละเอียดของไฟล์ Example.swrls (ต่อ)

ตารางที่ 3.2. รายละเอียดของกฎในเอกสารเอสดับเบิลยูอาร์แอล

ชื่อของกฎ	ส่วนพจน์นำ	ส่วนพจน์ตาม
inputRule	$x > 0$ and $y > 0$ and $z > 0$	
outputRule1	$x == y$ and $y == z$	TriangleResult is an Equilateral Triangle
outputRule2	$x <> y$ and $x <> z$ and $y <> z$	TriangleResult is a Scalene Triangle
outputRule3	$x < (y + z)$ and $y < (x + z)$ and $z < (x + y)$	TriangleResult is a Triangle
outputRule4	$x == y$ and $x <> z$	TriangleResult is an Isosceles Triangle
outputRule5	$x == z$ and $x <> y$	TriangleResult is an Isosceles Triangle
outputRule6	$y == z$ and $x <> z$	TriangleResult is an Isosceles Triangle

1. Analyze the hierarchical tree structure of operation element node.
 2. Traverse the tree, and at each tree node:
 - a. If it is an input element node, generate data type definition for input parameters.
 - b. If it is an output element node, generate data type definition for output parameters.
 - c. If it is a precondition element node, extract rule's URIs of precondition from modelReference attribute.
 - d. If it is an effect element node, extract referred rule's URIs of post-condition from modelReference attribute.
 3. Keep operation definition into database.
- In step 2.a and 2.b, the process is generated data type definition which consequence from binding message type to operation.

รูปที่ 3.4 รหัสคำสั่งเทียมสำหรับการวิเคราะห์เอกสารดับเบิลยูเอสดีแอล-เอส

1. Find ruleml:imp element node.
2. Retrieve rule name from rdf:ID attribute.
3. Analyze the hierarchical tree structure of ruleml:imp element node.
4. Traverse the tree, and at each tree node:
 - a. If it is a ruleml:_body element node, generate a logic expression of antecedent part.
 - b. If it is a ruleml:_head element node, generate a logic expression of consequent part.
5. Keep rule name and logic expression of antecedent part and consequent part into database.
6. Find next ruleml:imp element node and go to step 2. If next ruleml:imp element node is not exist, exit analysis.

รูปที่ 3.5 รหัสคำสั่งเทียมสำหรับการวิเคราะห์เอกสารเอสดับเบิลยูอาร์แอล

จากรูปที่ 3.6 เครื่องมือจะเปลี่ยนประโยคเงื่อนไขในรูปแบบเอสดับเบิลยูอาร์แอล เป็นประโยคเงื่อนไขทางคณิตศาสตร์ คือ $x < (y + z)$

```

<ruleml:_body>
  <swrx:builtinAtom swrx:builtin="swrb:add">
    <ruleml:var>sumOfY_Z</ruleml:var>
    <ruleml:var>y</ruleml:var>
    <ruleml:var>z</ruleml:var>
  </swrx:builtinAtom>

  <swrx:builtinAtom swrx:builtin="swrb:lessThan">
    <ruleml:var>x</ruleml:var>
    <ruleml:var>sumOfY_Z</ruleml:var>
  </swrx:builtinAtom>
</ruleml:_body>

```

รูปที่ 3.6. ประโยคเงื่อนไขในรูปแบบเอสดับเบิลยูอาร์แอล

จากเอกสารดับเบิลยูเอสดีแอล-เอสและเอกสารเอสดับเบิลยูอาร์แอลของเว็บเซอร์วิส Triangle ที่กล่าวในข้างต้น ซึ่งผลลัพธ์ในการวิเคราะห์เอกสารดับเบิลยูเอสดีแอล-เอสและเอกสารเอสดับเบิลยูอาร์แอล ดังตารางที่ 3.3 และตารางที่ 3.4 ตามลำดับ โดยเครื่องมือจะนำข้อมูลของทั้งสองเอกสารมาวิเคราะห์หาข้อกำหนดของโอเปอเรชันได้ดังตารางที่ 3.5

ตารางที่ 3.3. ผลลัพธ์ในการวิเคราะห์เอกสารดับเบิลยูเอสดีแอล-เอส

Operation	TriType	
	Variable	Data Type
Input Data	x	Integer
	y	Integer
	z	Integer
Output Data	TriangleResult	String

ตารางที่ 3.3. ผลลัพธ์ในการวิเคราะห์เอกสารดับเบิลยูเอสดีแอล-เอส (ต่อ)

	Rule	URI
Pre-condition	inputRule1	http://localhost/example.swrlx#inputRule1
Post-condition	outputRule1	http://localhost/example.swrlx#outputRule1
	outputRule2	http://localhost/example.swrlx#outputRule2
	outputRule3	http://localhost/example.swrlx#outputRule3
	outputRule4	http://localhost/example.swrlx#outputRule4
	outputRule5	http://localhost/example.swrlx#outputRule5
	outputRule6	http://localhost/example.swrlx#outputRule6

ตารางที่ 3.4. ผลลัพธ์ในการวิเคราะห์เอกสารเอสดับเบิลยูอาร์แอล

Rule	Antecedent	Consequent
inputRule1	$(x > 0)$ and $(y > 0)$ and $(z > 0)$	
outputRule1	$(x < (y + z))$ and $(y < (x + z))$ and $(z < (x + y))$	TriangleResult is a Triangle
Rule	Antecedent	Consequent
outputRule2	$(x == y)$ and $(y == z)$	TriangleResult is an Equilateral Triangle
outputRule3	$(x <> y)$ and $(x <> z)$ and $(y <> z)$	TriangleResult is a Scalene Triangle
outputRule4	$(x == y)$ and $(x <> z)$	TriangleResult is an Isosceles Triangle
outputRule5	$(x == z)$ and $(x <> y)$	TriangleResult is an Isosceles Triangle
outputRule6	$(y == z)$ and $(x <> z)$	TriangleResult is an Isosceles Triangle

ตารางที่ 3.5. รายละเอียดของโอเปอเรชัน TriangleType

โอเปอเรชัน	TriangleType		
ข้อมูลนำเข้า	ตัวแปร	ประเภทของข้อมูล	
	X	ตัวเลขจำนวนเต็ม	
	Y	ตัวเลขจำนวนเต็ม	
	Z	ตัวเลขจำนวนเต็ม	
ผลลัพธ์	ตัวแปร	ประเภทของข้อมูล	
	TriangleResult	สายอักขระ	
Pre-condition	ชื่อของกฎ	ส่วนพจน์นำ	ส่วนพจน์ตาม
	inputRule1	$x > 0$ and $y > 0$ and $z > 0$	
Post-condition	ชื่อของกฎ	ส่วนพจน์นำ	ส่วนพจน์ตาม
	outputRule1	$x == y$ and $y == z$	TriangleResult is an Equilateral Triangle
	outputRule2	$x <> y$ and $x <> z$ and $y <> z$	TriangleResult is a Scalene Triangle
	outputRule3	$x < (y + z)$ and $y < (x + z)$ and $z < (x + y)$	TriangleResult is a Triangle
	outputRule4	$x == y$ and $x <> z$	TriangleResult is an Isosceles Triangle
	outputRule5	$x == z$ and $x <> y$	TriangleResult is an Isosceles Triangle
	outputRule6	$y == z$ and $x <> z$	TriangleResult is an Isosceles Triangle

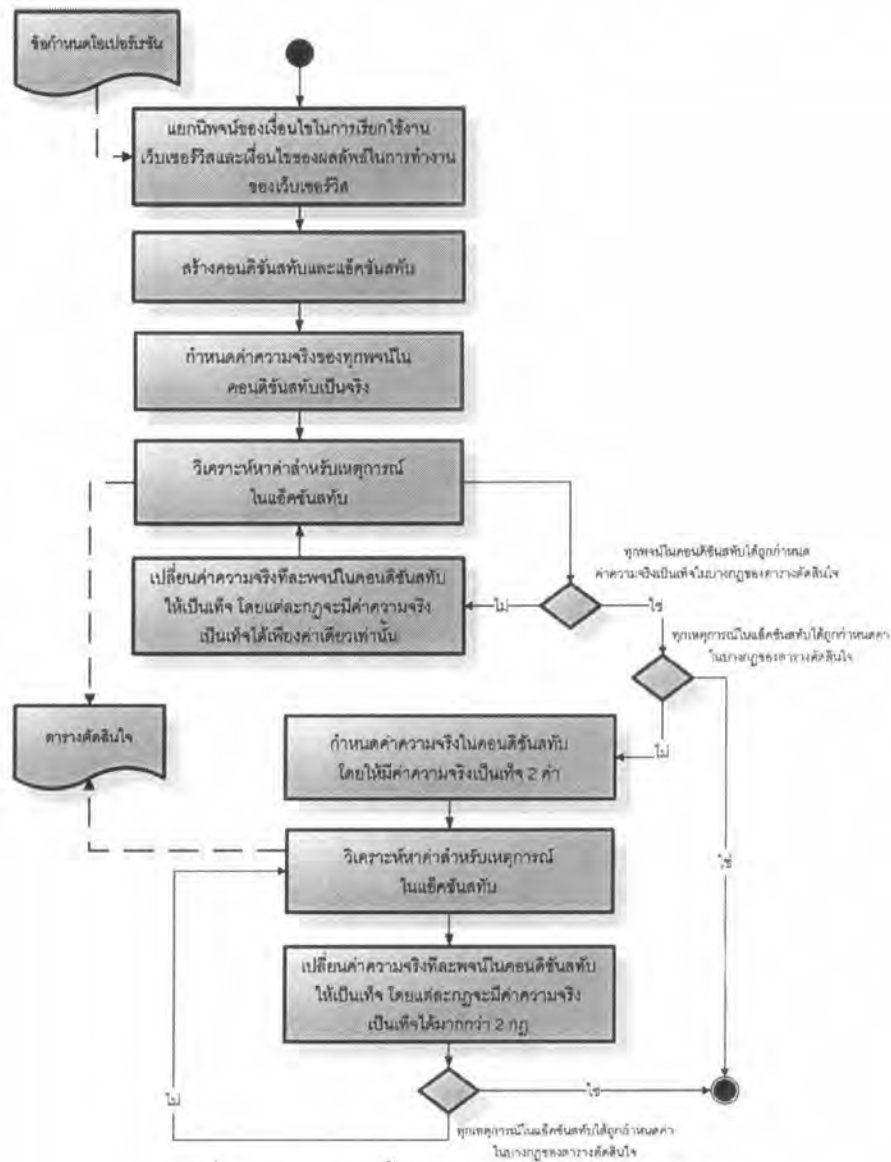
3.2.2. การสร้างตารางตัดสินใจ

จากข้อมูลที่ได้จากการวิเคราะห์เอกสารฉบับเบสิคยูเอสดีแอล-เอส และเอสฉบับเบสิคยูอาร์แอล เครื่องมือจะสร้างตารางตัดสินใจสำหรับแต่ละโอเปอเรชันด้วยตารางตัดสินใจประเภท

มีเหตุเอนทรี โดยแผนภาพขั้นตอนการสร้างตารางตัดสินใจของงานวิจัยนี้และรหัสคำสั่งเทียม สำหรับสร้างตารางตัดสินใจแสดงดังรูปที่ 3.7 และรูปที่ 3.8 ตามลำดับ โดยมีรายละเอียดของ ขั้นตอนการสร้างตารางตัดสินใจดังนี้

3.2.2.1. แยกนิพจน์ของเงื่อนไขในการเรียกใช้งานเว็บเซอริวิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอริวิสจากข้อกำหนดโอเปอเรชันสำหรับโอเปอเรชันที่ต้องการสร้างตารางตัดสินใจ รวมทั้งเหตุการณ์ที่ได้จากเงื่อนไขจากการเรียกใช้งานเว็บเซอริวิสและเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอริวิส

3.2.2.2. นำนิพจน์ที่ได้จากหัวข้อ 3.2.2.1 มาสร้างเป็นคอนดิชันสลับ โดยนิพจน์ที่เป็นนิเสธกับนิพจน์ก่อนหน้าจะไม่ถูกนำมาสร้าง และเหตุการณ์ที่ได้มาสร้างเป็นแอ็คชันสลับในตารางตัดสินใจ



รูปที่ 3.7. แผนภาพขั้นตอนการสร้างตารางตัดสินใจ

1. Retrieve operation definition.
2. Analyze the rules of precondition.
3. Traverse the rules, and at each atom of logic expression:
 - a. If atom negates other atom in condition stub, ignore it.
 - b. If atom does not negate other atom and atom has not a transitive property, insert it into condition stub.
4. Insert impossible action and invoking web service action into action stub.
5. Analyze the rule of post-condition.
6. Traverse the rule and insert each rule into action stub.
7. Define all conditions to be true.
8. Specify the first rule based on actions corresponding to the conditions in step 7.
9. For subsequent rules, change each condition to be false one by one and specify actions corresponding to the conditions.
10. If all actions are not performed, add more rules until they are completed.

รูปที่ 3.8. รหัสคำสั่งเทียมสำหรับการสร้างตารางตัดสินใจ

3.2.2.3. เครื่องมือกำหนดค่าสำหรับสร้างคอนดิชันเอนทรี โดยตัวอักษร "T" แทนค่าความจริงเป็นจริง และตัวอักษร "F" แทนค่าความจริงเป็นเท็จ จากนั้นเครื่องมือจะวิเคราะห์หาค่าของเหตุการณ์สำหรับแต่ละกฎในแอ็คชันเอนทรี โดยมีกฎเกณฑ์ ดังนี้

1) สำหรับกฎที่มีพจน์ขัดแย้งกัน เครื่องมือจะกำหนดค่าในแอ็คชันเอนทรีเป็นเหตุการณ์ที่เป็นไปไม่ได้ (Impossible)

2) สำหรับกฎที่ไม่สามารถทำให้เกิดการเรียกใช้งานเว็บเซอร์วิส เครื่องมือจะกำหนดค่าในแอ็คชันเอนทรีเป็นเหตุการณ์เว็บเซอร์วิสไม่เกิดการ ทำงาน (Not invoking Web Service)

3) สำหรับกฎที่มีเหตุการณ์ไม่สอดคล้องกับค่าในแอ็คชันเอนทรีใดๆ เครื่องมือจะกำหนดค่าในแอ็คชันเอนทรีเป็นเหตุการณ์อื่นๆ (Other action) และเหตุการณ์เว็บเซอร์วิสที่เกิดการทำงาน (Invoking Web Service)

4) สำหรับกฎที่มีเหตุการณ์สอดคล้องกับค่าในแอ็คชันเอนทรีใดๆ เครื่องมือจะกำหนดค่าในแอ็คชันเอนทรีที่สอดคล้องกับเหตุการณ์เหล่านั้น และเหตุการณ์เว็บเซอร์วิสที่เกิดการทำงาน (Invoking Web Service)

โดยรายละเอียดของขั้นตอนกำหนดค่าสำหรับสร้างคอนดิชันเอนทรี และวิเคราะห์หาค่าสำหรับเหตุการณ์สำหรับแต่ละกฎในแอ็คชันเอนทรี ดังนี้

3.2.2.3.1. กำหนดค่าความจริงของทุกพจน์ในคอนดิชันสตัป เป็นจริง

3.2.2.3.2. เปลี่ยนค่าความจริงที่ละพจน์ให้เป็นเท็จจนครบทุกพจน์ โดยแต่ละกฎจะมีค่าความจริงเป็นเท็จได้เพียงค่าเดียวเท่านั้น

3.2.2.3.3. วิเคราะห์หาค่าของเหตุการณ์สำหรับแต่ละกฎในแอ็คชันเอนทรี

3.2.2.3.4. ตรวจสอบแอ็คชันเอนทรีที่ได้ในตารางตัดสินใจครอบคลุมทุกเหตุการณ์ที่เป็นไปได้ทั้งหมดหรือไม่ ถ้าไม่ครอบคลุมจะทำขั้นตอนถัดไป

3.2.2.3.5. กำหนดค่าความจริงของทุกพจน์ที่เป็นเงื่อนไขในการเรียกใช้งานเป็นจริง และเปลี่ยนค่าความจริงของพจน์ที่เป็นเงื่อนไขของผลลัพธ์ในการทำงานให้เป็นเท็จ จนกระทั่งแอ็คชันเอนทรีที่ได้ในตารางตัดสินใจครอบคลุมทุกเหตุการณ์ที่เป็นไปได้ทั้งหมด ซึ่งค่าความจริงของพจน์ที่เป็นเท็จในขั้นตอนนี้จะมีมากกว่า 2 พจน์

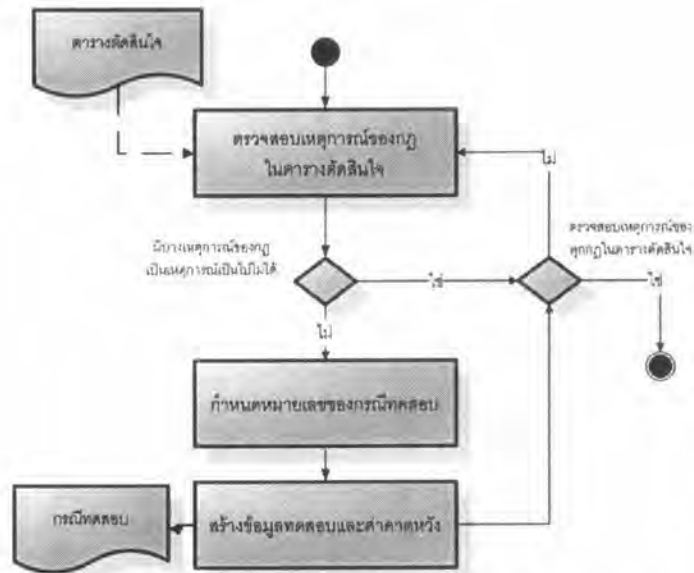
จากข้อมูลที่วิเคราะห์ได้ในตารางที่ 3.5 เครื่องมือจะนำประโยคเงื่อนไขที่ถูกระบุในส่วนพจน์นำของเงื่อนไขในการเรียกใช้งานเว็บเซอร์วิส และเงื่อนไขของผลลัพธ์ในการทำงานของเว็บเซอร์วิสมาสร้างเป็นคอนดิชันสทัท และนำข้อมูลในส่วนพจน์ตามของเงื่อนไขในการเรียกใช้งาน และเงื่อนไขของผลลัพธ์ในการใช้งานเว็บเซอร์วิสมาสร้างเป็นแอ็คชันเอนทรี ต่อจากนั้นเครื่องมือจะวิเคราะห์หาค่ากฎต่างๆ ในตารางตัดสินใจ ซึ่งจะได้ตารางตัดสินใจดังตารางที่ 3.6

3.2.3. การสร้างกรณีทดสอบสำหรับเว็บเซอร์วิส

เครื่องมือจะนำตารางตัดสินใจ และค่าข้อมูลของแต่ละตัวที่ได้วิเคราะห์เรียบร้อยแล้วมาสร้างกรณีทดสอบ โดยแผนภาพขั้นตอนการสร้างกรณีทดสอบแสดงดังรูปที่ 3.9 ซึ่งกรณีทดสอบที่สร้างขึ้นนั้น จะถูกจัดเก็บอยู่ในรูปแบบเอกซ์เอ็มแอลที่สอดคล้องกับเอกซ์เอ็มแอลสคีมาที่กำหนดขึ้นดังรูปที่ 3.10 ทำให้สามารถนำกรณีทดสอบเหล่านี้มาแสดงผลได้หลากหลายรูปแบบ นอกจากนี้ยังสามารถนำเป็นข้อมูลนำเข้าสำหรับเครื่องขับกรณีทดสอบ (Test driver) ได้อีกด้วย โดยกรณีทดสอบที่สร้างขึ้นจะอยู่บนสมมติฐานของข้อบกพร่องที่เกิดขึ้นจากตัวแปรใดตัวหนึ่งเพียงอันเดียว หรือข้อบกพร่องที่เกิดขึ้นจากหลายตัวแปร (Multiple-fault assumption) เมื่อไม่สามารถสร้างตารางตัดสินใจครอบคลุมทุกเหตุการณ์ในแอ็คชันเอนทรีด้วยสมมติฐานของข้อบกพร่องที่เกิดขึ้นจากตัวแปรใดตัวหนึ่งเพียงอันเดียว ซึ่งจำนวนของกรณีทดสอบสำหรับแต่ละโอเปอเรชันจะมีจำนวนเท่ากับกฎที่ถูกสร้างขึ้นในตารางตัดสินใจที่มีค่าในแอ็คชันเอนทรีไม่เป็นเหตุการณ์ที่เป็นไปไม่ได้และมีรายละเอียดของกรณีทดสอบ ดังนี้

3.2.3.1. หมายเลขของกรณีทดสอบ

3.2.3.2. ชื่อโอเปอเรชันที่ต้องการทดสอบ



รูปที่ 3.9. แผนภาพขั้นตอนการสร้างกรณีสอบ

```

<schema xmlns="http://www.w3.org/2001/XMLSchema">
  <element name="items">
    <complexType><sequence>
      <element name="item" type="item" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
<complexType name="item"><sequence>
  <element name="input" type="variableDesc" minOccurs="0" maxOccurs="unbounded" />
  <element name="expected" type="string" minOccurs="0" maxOccurs="unbounded" />
</sequence>
  <attribute name="operation" type="string" />
</complexType>
<complexType name="variableDesc"><sequence>
  <element name="variable" type="string" /><complexType><complexContent>
    <restriction base="string">
      <attribute name="dataType" type="string" />
    </restriction>
  </complexContent>
</complexType>
  <element name="value" type="string" />
</sequence>
</complexType>
</schema>
    
```

รูปที่ 3.10. รูปแบบเอกซ์เอ็มแอลสคีมาสำหรับกรณีสอบ

3.2.3.3. ชื่อของตัวแปรและค่าของตัวแปร ซึ่งค่าของตัวแปรนั้นเครื่องมือจะสุ่มค่าในช่วงของข้อมูลที่มีระหว่างความสัมพันธ์ของพจน์ในตารางตัดสินใจ และค่าข้อมูลที่ได้จากเอกสารดับเบิลยูเอสดีแอล-เอส

3.2.3.4. ค่าคาดหวังของการทดสอบการทำงานของเว็บเซอร์วิส ซึ่งเครื่องมือจะกำหนดค่าคาดหวังที่ได้จากกฎที่เลือกมาสร้างกรณีทดสอบ และค่าคาดหวังนั้นต้องสอดคล้องกับเงื่อนไขของผลลัพธ์ที่ได้จากการทำงานของเว็บเซอร์วิสที่กำหนดในระดับเบิลยูเอสดีแอล-เอสและดับเบิลยูเอสดีแอล

จากตารางที่ 3.6 เครื่องมือจะนำกฎในตารางตัดสินใจแต่ละกฎมาสร้างเป็นกรณีทดสอบ โดยสุ่มค่าของข้อมูลทดสอบจากค่าข้อมูลที่ได้แยกไว้ และต้องสอดคล้องกับพจน์ในกฎ ซึ่งจะยกตัวอย่างการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสตามกฎที่ 29 และมีรายละเอียดของกรณีทดสอบดังตารางที่ 3.7

ตารางที่ 3.6. ตารางตัดสินใจสำหรับโอเปอเรชัน TriangleType

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	29	60
$(x > 0)$	T	F	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T
$(y > 0)$	T	-	F	T	T	T	T	T	T	T	T	T	T	T	T	T	T
$(z > 0)$	T	-	-	F	T	T	T	T	T	T	T	T	T	T	T	T	T
$(x < (y + z))$	T	-	-	-	F	T	T	T	T	T	F	F	T	F	F	T	T
$(y < (x + z))$	T	-	-	-	T	F	T	T	T	T	F	T	F	F	T	T	T
$(z < (x + y))$	T	-	-	-	T	T	F	T	T	T	T	F	F	F	T	T	T
$(x == y)$	T	-	-	-	T	T	T	F	T	T	T	T	T	T	F	F	F
$(y == z)$	T	-	-	-	T	T	T	T	F	T	T	T	T	T	T	F	F
$(x == z)$	T	-	-	-	T	T	T	T	T	F	T	T	T	T	T	T	F
Impossible					X	X	X	X	X	X	X	X	X	X	X		
Invoking Web service	X															X	X
Not invoking Web service		X	X	X													
Other actions																	
$(\text{TriangleResult} \text{ is Equilateral})$	X																
$(\text{TriangleResult} \text{ is Scalene})$																	X
$(\text{TriangleResult} \text{ is a Triangle})$	X															X	X
$(\text{TriangleResult} \text{ is Isosceles})$																X	

ตารางที่ 3.7. ตัวอย่างกรณีทดสอบสำหรับเว็บเซอร์วิสตามกฎหมายที่ 29 ของตารางตัดสินใจ

หมายเลขกรณีทดสอบ	5	
ชื่อโอเปอเรชันที่ต้องการทดสอบ	TriangleType	
ข้อมูลทดสอบ	ตัวแปร	ค่าข้อมูล
	x	2
	y	1
	z	2
ค่าคาดหวังของการทดสอบ	Invoke web service TriangleResult is a Triangle TriangleResult is an Isosceles Triangle	