

การวิเคราะห์ผลกระทบต่อการทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ

Impact Analysis of Test Cases Based on Changes of a Web Application

สุรศักดิ์ เพ็ชรmani¹ และ ธราทิพย์ สุวรรณศาสตร์²
 ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
 E-mail: surasak.ph@student.chula.ac.th, taratip.s@chula.ac.th

บทคัดย่อ

การเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ ส่งผลกระทบต่อกรณีทดสอบที่มีอยู่เดิม ทำให้กรณีทดสอบบางส่วนไม่สามารถนำมาใช้ทดสอบได้ งานวิจัยนี้จึงมีวัตถุประสงค์เพื่อนำเสนอแนวคิดในการวิเคราะห์ผลกระทบต่อการทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ โดยการเปรียบเทียบโปรแกรมประยุกต์บนเว็บระหว่าง 2 เวอร์ชัน การวิเคราะห์หาผลกระทบและปรับปรุงกรณีทดสอบ และการสร้างกรณีทดสอบใหม่ขึ้นมากทดแทนกรณีทดสอบเดิม ผลลัพธ์ที่ได้จากงานวิจัยนี้เป็นแนวคิดในการวิเคราะห์ผลกระทบทำให้ทราบว่าโปรแกรมประยุกต์บนเว็บส่วนใดบ้างที่เปลี่ยนแปลงไป มีการเปลี่ยนแปลงและส่งผลกระทบต่อกรณีทดสอบอย่างไร รวมถึงผู้ใช้สามารถนำกรณีทดสอบที่ได้รับการปรับปรุงเรียบร้อยแล้วไปทดสอบโปรแกรมประยุกต์บนเว็บที่มีการเปลี่ยนแปลงได้

คำสำคัญ

การวิเคราะห์ผลกระทบ กรณีทดสอบ โปรแกรมประยุกต์บนเว็บ

Abstract

Changes of a web application impact on existing test cases, for this reason some test cases cannot be longer used to test the web application again. This paper aims at introducing an impact analysis of test cases when a web application is changed. We present a comparison method between two versions of a web application, and then develop a process for finding and handling changes on existing test cases. Finally, we introduce an approach for generating new test cases to replace the existing test cases. The results show the concept of impact analysis that indicates which parts of a web application have been changed and which test cases are affected by the changes. Moreover, new test cases from this method can be used to test the new version of the web application correctly.

Keywords

impact analysis, test case, web application

1. บทนำ

โปรแกรมประยุกต์บนเว็บเข้ามามีบทบาทสำคัญในชีวิตประจำวันของมนุษย์เพิ่มมากขึ้น และพัฒนาเปลี่ยนแปลงไปอย่างรวดเร็ว การเปลี่ยนแปลงที่เกิดขึ้นนี้ส่งผลกระทบต่อกระบวนการทดสอบอย่างหลีกเลี่ยงไม่ได้ คือ กรณีทดสอบเดิมบางส่วนไม่สามารถนำมาใช้ทดสอบโปรแกรมประยุกต์บนเว็บที่เปลี่ยนแปลงไปได้ ซึ่งเดิมการแก้ปัญหาเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บในอดีตจะเป็นการสร้างกรณีทดสอบขึ้นมาใหม่ทั้งหมด ทำให้ผู้ทดสอบจะต้องใช้เวลารวมทั้งทรัพยากรอย่างมากในการสร้างกรณีทดสอบและนำมาทดสอบระบบใหม่ทั้งหมด หรือไม่ก็ต้องตรวจสอบทุกครั้งก่อนนำไปทดสอบโปรแกรมประยุกต์บนเว็บที่มีการเปลี่ยนแปลง

จากการศึกษางานวิจัยที่เกี่ยวข้องกับการสร้างกรณีทดสอบ [1, 2] การทดสอบโปรแกรมประยุกต์บนเว็บ [3-5] และเครื่องมือทดสอบที่ใช้ในปัจจุบัน [6] พบว่างานวิจัยและเครื่องมือเหล่านั้นมุ่งเน้นเพื่อสร้างกรณีทดสอบ และดำเนินการทดสอบโปรแกรมประยุกต์บนเว็บเท่านั้น ยังไม่ได้คำนึงถึงการเปลี่ยนแปลงที่จะเกิดขึ้นในอนาคต แม้ว่าที่ผ่านมาผู้วิจัยได้ศึกษาเกี่ยวกับการวิเคราะห์ผลกระทบของการเปลี่ยนแปลงอย่างแพร่หลาย [7-10] แต่ก็ยังไม่ได้ครอบคลุมไปถึงในเรื่องของการวิเคราะห์ผลกระทบต่อกรณีทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ

ดังนั้นผู้วิจัยจึงเสนอแนวคิดในการวิเคราะห์ผลกระทบต่อกรณีทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ โดยเริ่มต้นด้วยการนำเสนอ

วิธีเปรียบเทียบโปรแกรมประยุกต์บนเว็บระหว่าง 2 เวอร์ชัน การวิเคราะห์หาผลกระทบและปรับปรุงกรณีทดสอบ และการสร้างกรณีทดสอบใหม่ขึ้นมาทดแทนกรณีทดสอบเดิม

2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 การวิเคราะห์ผลกระทบ

การวิเคราะห์ผลกระทบ เป็นกิจกรรมของการระบุถึงการเปลี่ยนแปลงที่เกิดขึ้น และผลกระทบที่เกิดจากการเปลี่ยนแปลงนั้น ด้วยวิธีการต่าง ๆ เช่น การใช้คู่มืออ้างอิง การใช้ระบบตามรอย เป็นต้น [9] จากความหมายข้างต้นทำให้เกิดประเด็นสำคัญในการวิจัยนี้ คือ

1. โปรแกรมประยุกต์บนเว็บส่วนใดบ้างที่มีการเปลี่ยนแปลง
2. การเปลี่ยนแปลงที่เกิดขึ้นส่งผลกระทบต่อกรณีทดสอบอย่างไร
3. สามารถปรับปรุง แก้ไข หรือสร้างกรณีทดสอบมาทดแทนได้หรือไม่

งานวิจัยที่ผ่านมา [8, 9] พบว่ามีการวิจัยเกี่ยวข้องกับการวิเคราะห์ผลกระทบโดยใช่วิธีการที่หลากหลาย อย่างไรก็ตามงานวิจัยเหล่านั้นยังไม่ได้คำนึงถึงเรื่องของการวิเคราะห์ผลกระทบต่อกรณีทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บเกิดขึ้น

2.2 กรณีทดสอบ

กรณีทดสอบที่ใช้ในงานวิจัยนี้ คือ กรณีทดสอบที่สร้างขึ้นสำหรับการทดสอบแบบแบล็กบ็อกซ์ โดยมีที่มาจากเครื่องมือสำหรับสร้างกรณีทดสอบจากเอกซ์เอ็มแอลและเอกซ์เอ็มแอลสคิม่า ซึ่งเป็นงานวิจัยของ [1, 2]

ประกอบด้วยวิธีการดังนี้

1. การทดสอบโดยใช้ชั้นสมมูล (Equivalence Class Testing) [12]

การทดสอบโดยใช้ชั้นสมมูลเป็นวิธีการทดสอบเพื่อตรวจหาข้อผิดพลาด โดยนำข้อมูลนำเข้ามาจัดกลุ่มและแบ่งข้อมูลออกเป็นช่วงที่เรียกว่าชั้นสมมูล

(Equivalence Class) ซึ่งมีหลักเกณฑ์ในการกำหนดชั้นสมมูล คือ ข้อมูลนำเข้าเป็นค่าคงที่ ค่าเซต หรือค่าตรรกะ จะได้หนึ่งชั้นสมมูลที่ถูกต้อง (valid) และหนึ่งชั้นสมมูลที่ไม่ถูกต้อง (invalid) ส่วนข้อมูลนำเข้าเป็นค่าช่วง จะได้หนึ่งชั้นสมมูลที่ถูกต้อง และสองชั้นสมมูลที่ไม่ถูกต้อง ดังตัวอย่างในตารางที่ 1

ตารางที่ 1 ตัวอย่างการกำหนดชั้นสมมูล

ตัวแปร	ชั้น	ข้อมูลนำเข้า	ประเภท	ค่าตัวแปร
pwd	1	$6 \leq \text{length of pwd} \leq 18$	valid	123456789
	2	$\text{length of pwd} < 6$	invalid	12345
	3	$\text{length of pwd} > 18$	invalid	1234567890123456789
sex	1	sex in {male, female}	valid	male
	2	sex not in {male, female}	invalid	other

2. การทดสอบโดยใช้ค่าขอบเขต (Boundary Value Testing) [12]

การทดสอบโดยใช้ค่าขอบเขตเป็นวิธีการทดสอบโดยยึดหลักว่าข้อผิดพลาดที่เกิดขึ้นมักจะเกิดใกล้ ๆ กับ

ค่าขอบเขตของตัวแปรนำเข้า ซึ่งวิธีการนี้ทำงานได้ดีกับตัวแปรที่มีความเป็นอิสระต่อกัน และมีค่าเป็นช่วงที่ชัดเจน วิธีการนี้จะมีการวิเคราะห์ค่าขอบเขตของตัวแปรนำเข้ดังตัวอย่างในตารางที่ 2

ตารางที่ 2 ตัวอย่างการวิเคราะห์ค่าขอบเขต

ตัวแปร	ลำดับ	ช่วงค่าที่ถูกต้อง	ค่าขอบเขต	ค่าตัวแปร
pwd	1	$6 \leq \text{length of pwd} \leq 18$	max+	1234567890123456789
			max	123456789012345678
			max-	12345678901234567
			nom	123456789
			min+	1234567
			min	123456
			min-	12345

2.3 โปรแกรมประยุกต์บนเว็บ

1. เอกซ์เอ็มแอล (HTML: Hyper Text Markup Language)

งานวิจัยนี้ให้ความสนใจเรื่องฟอร์มของเอกซ์เอ็มแอล (HTML form) สำหรับรับค่าตัวแปรนำเข้า (input variable) จากผู้ใช้ เนื่องจากเป็นส่วนต่อประสานกับผู้ใช้ในการส่งผ่านข้อมูลของโปรแกรมประยุกต์บนเว็บ และเป็นส่วนหลักที่ใช้สำหรับการดำเนินการทดสอบ โดยฟอร์มของเอกซ์เอ็มแอล ประกอบไปด้วยแท็ก ดังนี้ เขตข้อมูลข้อความ เขตข้อมูลรหัสผ่าน ปุ่มเรดิโอ เช็คบ็อกซ์ ปุ่มเลือกรายการ ข้อความหลายบรรทัด และปุ่มยืนยัน ตัวอย่างเอกซ์เอ็มแอลที่ใช้ในงานวิจัยนี้ ดังภาพที่ 1

```
<!DOCTYPE html>
<html>
<head><title>Demo</title></head>
<body>
<h1>Demo</h1>
<form name="demo" action="landing.html">
  First name: <input type="text" name="firstname">
  Last name: <input type="text" name="lastname">
  Password: <input type="password" name="pwd"> * require 6-18 characters
  <input type="submit" value="Submit">
</form>
</body>
```

ภาพที่ 1 ตัวอย่างเอกซ์เอ็มแอล

2. เอกซ์เอ็มแอลสคีม่า (XML Schema)

เอกซ์เอ็มแอลสคีม่า เป็นวิธีการนิยามหรือกำหนดกฎเกณฑ์ไวยากรณ์โครงสร้างข้อมูลของเอกสารเอกซ์เอ็มแอล สามารถรองรับการกำหนดขอบเขตของข้อมูล และนำมาใช้ในการตรวจสอบความถูกต้องของข้อมูลได้ งานวิจัยนี้ได้นำแนวคิดมาจากงานวิจัยของ [1, 2, 13] ในการนำเอกซ์เอ็มแอลสคีมามาใช้ในการเปรียบเทียบการเปลี่ยนแปลง ตรวจสอบความถูกต้องของข้อมูล คือ ชื่อตัวแปร ชนิดของข้อมูล ค่าของตัวแปร ลำดับของตัวแปร การเชื่อมต่อ โดยการกำหนดค่าต่าง ๆ ต้องอยู่ในรูปแบบที่มาตรฐาน และถูกต้องตามข้อกำหนดภาษา

[11] ตัวอย่างเอกซ์เอ็มแอลสคีม่าที่ใช้ในงานวิจัยนี้ ดังภาพที่ 2

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="firstname" type="xs:string"/>
  <xs:element name="lastname" type="xs:string"/>
  <xs:element name="pwd">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="6"/>
        <xs:maxLength value="18"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="submit" type="xs:string" default="Submit" fixed="landing.html"/>
</xs:schema>
```

ภาพที่ 2 ตัวอย่างเอกซ์เอ็มแอลสคีม่า

3. วิธีดำเนินงานวิจัย

งานวิจัยนี้เสนอการวิเคราะห์ผลกระทบต่อการทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ โดยมีภาพรวมของงานวิจัย ดังภาพที่ 3



ภาพที่ 3 ภาพรวมของงานวิจัย

จากภาพที่ 3 ผู้วิจัยได้แบ่งขั้นตอนการวิเคราะห์ผลกระทบต่อการทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บออกเป็น 3 ขั้นตอน คือ การเปรียบเทียบไฟล์เอกสารระหว่าง 2 เวอร์ชัน การวิเคราะห์ผลกระทบและปรับปรุงกรณีทดสอบและการสร้างกรณีทดสอบใหม่ ผลลัพธ์ที่ได้จากงานวิจัยนี้ทำให้ทราบว่า มีโปรแกรมประยุกต์บนเว็บส่วนใดบ้างที่เปลี่ยนแปลงไป และมีการเปลี่ยนอย่างไร รวมถึงผู้ใช้สามารถนำกรณีทดสอบที่ได้รับการปรับปรุงเรียบร้อยแล้วไปทดสอบโปรแกรมประยุกต์บนเว็บที่มีการเปลี่ยนแปลงได้ สำหรับรายละเอียดของแนวคิด และวิธีการดำเนินงานในแต่ละขั้นตอนอธิบายได้ดังนี้

3.1 การเปรียบเทียบไฟล์เอกสารระหว่าง 2 เวอร์ชัน

3.1.1 ข้อมูลนำเข้า

ผู้วิจัยจะนำเข้าไฟล์เอกสารเวอร์ชันใหม่ เพื่อนำมาเปรียบเทียบกับไฟล์เอกสารเวอร์ชันเดิมจากฐานข้อมูลโครงสร้างเว็บ ดังนี้

1. ไฟล์เอกสารเอกซ์เอ็มแอล ที่อยู่ในรูปแบบที่มาตรฐาน และถูกต้องตามข้อกำหนดภาษา เมื่อนำเข้าไฟล์เอกสารแล้วนำมาวิเคราะห์เพื่อหา ชื่อแท็กของตัวแปร และชื่อตัวแปร โดยมีแท็กที่สนใจ คือ เขตข้อมูล ข้อความ เขตข้อมูลรหัสผ่าน ปุ่มเรดิโอ เช็คบ็อกซ์ ปุ่มเลือกรายการ ข้อความหลายบรรทัด และปุ่มยืนยัน

2. ไฟล์เอกสารเอกซ์เอ็มแอลสคิม่า ที่อยู่ในรูปแบบที่มาตรฐาน และถูกต้องตามข้อกำหนดภาษา เมื่อนำเข้าไฟล์เอกสารแล้วนำมาวิเคราะห์เพื่อหา ชื่อตัวแปร ชนิดของข้อมูล ค่าของตัวแปร ลำดับของตัวแปร การเชื่อมต่อ และจำนวนของตัวแปร โดยที่ชื่อตัวแปรจะต้องตรงกันกับไฟล์เอกสารเอกซ์เอ็มแอล สำหรับงานวิจัยนี้ชนิดของข้อมูลที่รองรับ คือ สตริง จำนวนเต็ม จำนวนเต็มบวก ตรรกะ ทศนิยม และวันที่ สำหรับค่าที่ถูกต้องของตัวแปรแบ่งเป็น ค่าคงที่ ค่าช่วง ค่าเซต และ

ตรรกะ จากนั้นเมื่อผู้ใช้นำเข้าไฟล์เอกสารเวอร์ชันใหม่เรียบร้อยแล้ว จะนำมาเปรียบเทียบกับไฟล์เอกสารเวอร์ชันเดิมจากฐานข้อมูลโครงสร้างเว็บ ในขั้นตอนการเปรียบเทียบไฟล์เอกสารต่อไป

3.1.2 การเปรียบเทียบไฟล์เอกสาร

วิธีการเปรียบเทียบไฟล์เอกสาร ทำได้โดยนำค่าที่ได้จากการอ่านไฟล์เอกสารเอกซ์เอ็มแอล และไฟล์เอกสารเอกซ์เอ็มแอลสคิม่า ทั้ง 2 เวอร์ชันมาเปรียบเทียบกันเพื่อหาจุดที่แตกต่างกัน ซึ่งผลลัพธ์ที่ได้ประกอบด้วยค่าต่าง ๆ ที่สามารถเปลี่ยนแปลงได้จากเดิม คือ ชื่อตัวแปร ชนิดของข้อมูล ค่าของตัวแปร ชื่อแท็กของตัวแปร ลำดับของตัวแปร การเชื่อมต่อ และจำนวนของตัวแปร

3.1.3 การวิเคราะห์การเปลี่ยนแปลง

ขั้นตอนนี้นำผลการเปรียบเทียบมาวิเคราะห์การเปลี่ยนแปลงที่เกิดขึ้นตามรูปแบบต่างๆ โดยงานวิจัยนี้ได้เสนอมีรูปแบบของการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ 7 รูปแบบ ดังนี้

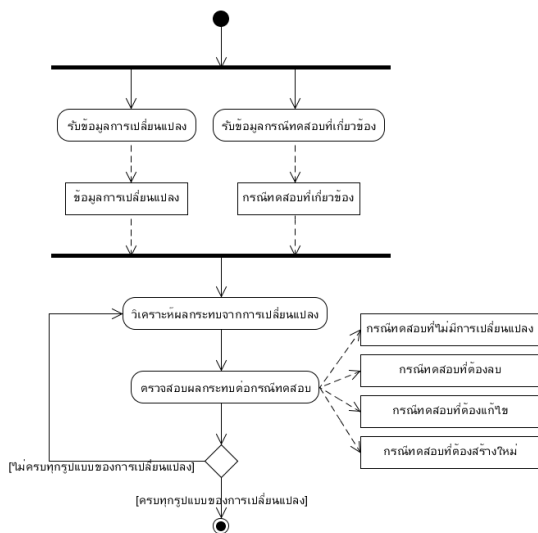
1. ชื่อตัวแปรมีการเปลี่ยนแปลง
2. ชนิดของข้อมูลมีการเปลี่ยนแปลง
3. ค่าของตัวแปรมีการเปลี่ยนแปลง
4. ชื่อแท็กของตัวแปรมีการเปลี่ยนแปลง
5. ลำดับของตัวแปรมีการเปลี่ยนแปลง
6. การเชื่อมต่อมีการเปลี่ยนแปลง
7. จำนวนของตัวแปรมีการเปลี่ยนแปลง

การเปลี่ยนแปลงที่เกิดขึ้นอาจเป็นการเพิ่ม การลบ การแก้ไขเปลี่ยนแปลงค่าต่างๆ ของตัวแปร จากขั้นตอนนี้จะทราบว่าอะไรที่เปลี่ยนแปลงไปบ้างและมีการเปลี่ยนแปลงอย่างไร ต่อมาเมื่อยืนยันการปรับปรุงเวอร์ชัน ก็จะมีการปรับปรุงฐานข้อมูลโครงสร้างเว็บให้เป็นเวอร์ชันใหม่ พร้อมทั้งเก็บไฟล์เอกสารเวอร์ชันเดิมไว้ หากผู้ใช้ต้องการย้อนกลับมาใช้งานอีกครั้ง จากนั้นเข้าสู่ขั้นตอนการตรวจสอบกรณีทดสอบต่อไป

3.2 การวิเคราะห์ผลกระทบและปรับปรุงกรณีทดสอบ

3.2.1 การค้นหากรณีทดสอบเดิมที่เกี่ยวข้อง กรณีทดสอบเดิมที่เกี่ยวข้อง คือ กรณีทดสอบที่ใช้ทดสอบในหน้าโปรแกรมประยุกต์บนเว็บนั้นๆ ในขั้นตอนนี้จะดึงกรณีทดสอบที่เกี่ยวข้องนั้นมาจากฐานข้อมูลกรณีทดสอบ เพื่อนำมาวิเคราะห์ผลกระทบต่อการทดสอบเมื่อมีการเปลี่ยนแปลง โดยรูปแบบของกรณีทดสอบที่ใช้ในงานวิจัยนี้ ประกอบด้วย หมายเลขกรณีทดสอบ ชื่อตัวแปร ค่าของตัวแปร ลำดับของตัวแปร และค่าผลลัพธ์ที่คาดหวัง ซึ่งค่าผลลัพธ์ที่คาดหวังมี 2 กรณี คือ ถูกต้อง (valid) และไม่ถูกต้อง (invalid)

3.2.2 การวิเคราะห์ผลกระทบต่อการกรณีทดสอบ การวิเคราะห์ผลกระทบจากการเปลี่ยนแปลงที่มีต่อการกรณีทดสอบที่เกี่ยวข้อง ดำเนินการวิเคราะห์และตรวจสอบผลกระทบโดยมีขั้นตอนดังภาพที่ 4



ภาพที่ 4 การวิเคราะห์ผลกระทบต่อการกรณีทดสอบเมื่อมีการเปลี่ยนแปลง

จากภาพที่ 4 ขั้นตอนการวิเคราะห์ผลกระทบจากการเปลี่ยนแปลงที่มีต่อการกรณีทดสอบ เริ่มต้นด้วยการรับข้อมูลการเปลี่ยนแปลง และรับข้อมูลกรณีทดสอบที่เกี่ยวข้องมาจากขั้นตอนที่ผ่านมา จากนั้นนำมาวิเคราะห์และตรวจสอบผลกระทบต่อการกรณีทดสอบ โดยกรณีทดสอบจะถูกแบ่งกลุ่มออกเป็น 4 กลุ่ม คือ กลุ่มกรณีทดสอบที่ไม่มีการเปลี่ยนแปลง กลุ่มกรณีทดสอบที่ต้องลบ กลุ่มกรณีทดสอบที่ต้องแก้ไข และกลุ่มที่ต้องสร้างกรณีทดสอบใหม่ วิธีการนี้สามารถรองรับรูปแบบของการเปลี่ยนแปลงพร้อมๆ กันได้หลายรูปแบบ โดยจะดำเนินการวิเคราะห์และตรวจสอบผลกระทบต่อการกรณีทดสอบไปจนกว่าจะครบทุกรูปแบบของการเปลี่ยนแปลง ขั้นตอนนี้จะทำให้ทราบว่าเมื่อมีการเปลี่ยนแปลงเกิดขึ้นแล้วจะส่งผลกระทบต่อการกรณีทดสอบอย่างไร ซึ่งงานวิจัยนี้ได้เสนอรูปแบบของการเปลี่ยนแปลงและผลกระทบต่อการกรณีทดสอบ โดยมีรายละเอียดและหลักการวิเคราะห์ผลกระทบ ดังตารางที่ 3

ตารางที่ 3 รูปแบบของการเปลี่ยนแปลงและผลกระทบต่อการกรณีทดสอบ

รูปแบบของการเปลี่ยนแปลง		ผลกระทบต่อการกรณีทดสอบ			
		แก้ไข	ลบ	สร้างใหม่	เป็นไปไม่ได้
1. ชื่อตัวแปรมีการเปลี่ยนแปลง	เพิ่ม	*		*	
	แก้ไข	*			
	ลบ	*	*		
2. ชนิดของข้อมูลมีการเปลี่ยนแปลง	เพิ่ม				*
	แก้ไข	ตรวจสอบรูปแบบการเปลี่ยนแปลงค่าของตัวแปร			
	ลบ				*
3. ค่าของตัวแปรมีการเปลี่ยนแปลง					
3.1 ค่า default, fixed	เพิ่ม ลบ แก้ไข	*			
3.2 ค่า restrictions					
3.2.1 ค่าคงที่	เพิ่ม	*		*	
	แก้ไข	*			
	ลบ	*	*		
ค่าคงที่ -> ค่าช่วง	เปลี่ยน	*		*	

ค่าคงที่ -> ค่าเขต	เปลี่ยน	*			
ค่าคงที่ -> ค่าตรรกะ	เปลี่ยน	*			
3.2.2 ค่าช่วง	เพิ่ม	*		*	
	แก้ไข	*			
	ลบ	*	*		
ค่าช่วง -> ค่าคงที่	เปลี่ยน	*	*		
ค่าช่วง -> ค่าเขต	เปลี่ยน	*	*		
ค่าช่วง -> ค่าตรรกะ	เปลี่ยน	*	*		
3.2.3 ค่าเขต	เพิ่ม	*		*	
	แก้ไข	*			
	ลบ	*	*		
ค่าเขต -> ค่าคงที่	เปลี่ยน	*			
ค่าเขต -> ค่าช่วง	เปลี่ยน	*		*	
ค่าเขต -> ค่าตรรกะ	เปลี่ยน	*			
3.2.4 ค่าตรรกะ	เพิ่ม	*		*	
	แก้ไข	*			
	ลบ	*	*		
ค่าตรรกะ -> ค่าคงที่	เปลี่ยน	*			
ค่าตรรกะ -> ค่าช่วง	เปลี่ยน	*		*	
ค่าตรรกะ -> ค่าเขต	เปลี่ยน	*			
4. ชื่อแท็กของตัวแปรมีการเปลี่ยนแปลง	เป็น อินพุต	ตรวจสอบรูปแบบของการเปลี่ยนแปลงในกรณีอื่น			
	ไม่เป็น	*	*		
5. ลำดับของตัวแปรมีการเปลี่ยนแปลง	เปลี่ยน	*			
6. การเชื่อมต่อมีการเปลี่ยนแปลง	เพิ่ม	*		*	
	แก้ไข	*			
	ลบ	*	*		
7. จำนวนของตัวแปรมีการเปลี่ยนแปลง	เพิ่ม	*		*	
	ลด	*	*		

3.2.3 การปรับปรุงกรณีทดสอบ

กรณีทดสอบที่ปรับปรุงในขั้นตอนนี้ คือ กรณีทดสอบที่อยู่ในกลุ่มที่ต้องแก้ไข และกลุ่มที่ต้องลบ ขั้นตอนนี้จะเข้าไปปรับปรุงกรณีทดสอบให้มีค่าถูกต้องตามการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บเวอร์ชันใหม่ ซึ่งหากเป็นวิธีการทดสอบโดยใช้ชั้นสมมูล จะปรับปรุงกรณีทดสอบเดิมให้มีความถูกต้องตรงกันกับข้อมูลชั้นสมมูล ส่วนวิธีการทดสอบโดยใช้ค่าขอบเขต จะปรับปรุงกรณีทดสอบเดิมให้มีความถูกต้องตรงกันกับข้อมูลการวิเคราะห์ค่าขอบเขตของตัวแปร ตัวอย่างการปรับปรุงกรณีทดสอบในกรณีที่เป็นวิธีการทดสอบโดยใช้ชั้นสมมูล ดังภาพที่ 5

Variable	Class No.	Input Equivalence Class	Class Type	Value
firstname	1	Not Specific	valid	xyz
lastname	1	Not Specific	valid	abcdefg
pwd	1	6 <= length of pwd <= 18	valid	123456789
	2	length of pwd < 6	invalid	12345
	3	length of pwd > 18	invalid	1234567890123456789
sex	1	sex in (male, female)	valid	male
	2	sex not in (male, female)	invalid	other
vehicle	1	vehicle in (bike, car)	valid	bike
	2	vehicle not in (bike, car)	invalid	other
country	1	country in (thailand, united states, france, uk)	valid	thailand
	2	country not in (thailand, united states, france, uk)	invalid	other
comments	1	default is Compose new comments...	valid	Compose new comments...
			invalid	other
submit	1	link is landing.html	valid	landing.html
			invalid	e2d265581f4.html

ภาพที่ 5 ตัวอย่างการปรับปรุงข้อมูลในชั้นสมมูล

จากตัวอย่างในภาพที่ 5 เป็นการปรับปรุงกรณีทดสอบตามวิธีการทดสอบโดยใช้ชั้นสมมูล โดยการปรับปรุงค่าตัวแปร pwd ให้มีค่าถูกต้องตรงกันกับชั้นสมมูล ผลลัพธ์ที่ได้จากขั้นตอนนี้หลังจากปรับปรุงข้อมูลในชั้นสมมูล หรือข้อมูลการวิเคราะห์ค่าขอบเขตของตัวแปรแล้ว จึงปรับปรุงค่าของตัวแปร pwd ในกรณีทดสอบ ให้มีค่าถูกต้องตามการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บเวอร์ชันใหม่ ได้เป็นกรณีทดสอบเวอร์ชันใหม่ที่ได้รับการปรับปรุงเรียบร้อยแล้ว ดังภาพที่ 6

Test Cases Project ID: 001 Project Name: Demo URL: http://localhost/demo/index.html Version: 2013032602							
ID	Variable Name	Value	Class Type	Sequence	Output		
TC1	firstname	xyz	valid	1	valid		
	lastname	abcdefg	valid	2			
	pwd	123456789	valid	3			
	sex	male	valid	4			
	vehicle	bike	valid	5			
	country	thailand	valid	6			
	comments	Compose new comments...	valid	7			
	submit	landing.html	valid	8			
TC2	firstname	xyz	valid	1	invalid		
	lastname	abcdefg	valid	2			
	pwd	12345	invalid	3			
	sex	male	valid	4			
	vehicle	bike	valid	5			
	country	thailand	valid	6			
	comments	Compose new comments...	valid	7			
	submit	landing.html	valid	8			
TC3	firstname	xyz	valid	1	invalid		
	lastname	abcdefg	valid	2			
	pwd	1234567890123456789	invalid	3			
	sex	male	valid	4			
	vehicle	bike	valid	5			
	country	thailand	valid	6			
	comments	Compose new comments...	valid	7			
	submit	landing.html	valid	8			

ภาพที่ 6 กรณีทดสอบเวอร์ชันใหม่

การระบุเวอร์ชันของกรณีทดสอบ มีการระบุตัวเลขที่แสดงถึงเวอร์ชันที่มีการเปลี่ยนแปลงไป โดยมีรูปแบบของเวอร์ชัน ดังนี้ ปปปปดตวทท โดยที่ ปปปป คือ ปี ค.ศ. ที่มีการปรับปรุงกรณีทดสอบ ดต คือ เดือน ที่มีการปรับปรุงกรณีทดสอบ วว คือ วันที่มีการปรับปรุงกรณีทดสอบ และ ทท คือ ครั้งที่มีการปรับปรุงกรณีทดสอบในรอบวันนั้น ยกตัวอย่างเช่น 2013032601 หมายถึง กรณีทดสอบนี้มีการปรับปรุงเมื่อปี ค.ศ. 2013 เดือนมีนาคม วันที่ 26 ครั้งที่ 1 เป็นต้น อย่างไรก็ตาม จากตารางที่ 3 จะเห็นได้ว่าในบางรูปแบบของการเปลี่ยนแปลงมีความจำเป็นต้องสร้างกรณีทดสอบใหม่เพิ่มเติม เพื่อให้ครอบคลุมการทดสอบโปรแกรมประยุกต์บนเว็บที่กล่าวมาข้างต้น คือ การทดสอบโดยใช้ชั้นสมมูล และการทดสอบโดยใช้ค่าขอบเขต โดยจะอธิบายรายละเอียดเกี่ยวกับเรื่องนี้ในหัวข้อการสร้างกรณีทดสอบใหม่

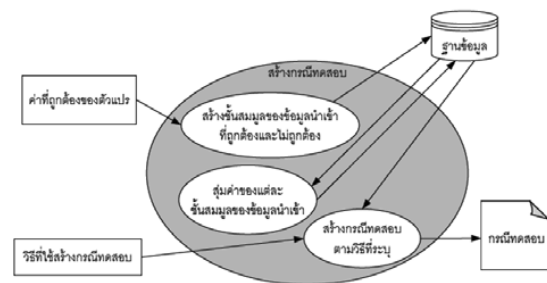
3.3 การสร้างกรณีทดสอบใหม่

การสร้างกรณีทดสอบใหม่จะใช้วิธีการสร้างกรณีทดสอบโดยใช้ชั้นสมมูล และวิธีการสร้างกรณีทดสอบโดยใช้ค่าขอบเขต ซึ่งสามารถเลือกได้ว่าจะสร้างกรณีทดสอบด้วยวิธีการใด งานวิจัยนี้ได้้นำวิธีการสร้างกรณีทดสอบจาก [1, 2] มาประยุกต์ใช้โดยมีรายละเอียดของวิธีการสร้างกรณีทดสอบทั้ง 2 วิธี ดังต่อไปนี้

3.3.1 วิธีการสร้างกรณีทดสอบโดยใช้ชั้นสมมูล

วิธีการสร้างกรณีทดสอบโดยใช้ชั้นสมมูล เริ่มต้นโดยการนำค่าที่ถูกต้องของตัวแปรมาสร้างชั้นสมมูลของข้อมูลนำเข้า จากนั้นสุ่มค่าของแต่ละชั้นสมมูลของข้อมูลนำเข้า แล้วนำค่าที่สุ่มมาสร้างเป็นกรณีทดสอบตามวิธีที่เลือกใช้ในการสร้างกรณีทดสอบ ซึ่งจะสามารถเลือกได้ 4 วิธี คือ การทดสอบโดยใช้ชั้นสมมูลแบบวิคนอร์มอล (Weak Normal Equivalence Class Testing) การทดสอบโดยใช้ชั้นสมมูลแบบสตรองนอร์มอล (Strong Normal Equivalence Class Testing)

การทดสอบโดยใช้ชั้นสมมูลแบบวิคโรบัส (Weak Robust Equivalence Class Testing) และการทดสอบโดยใช้ชั้นสมมูลแบบสตรองโรบัส (Strong Robust Equivalence Class Testing) โดยขั้นตอนต่างๆ ในการสร้างกรณีทดสอบโดยใช้ชั้นสมมูล สามารถอธิบายได้ ดังภาพที่ 7

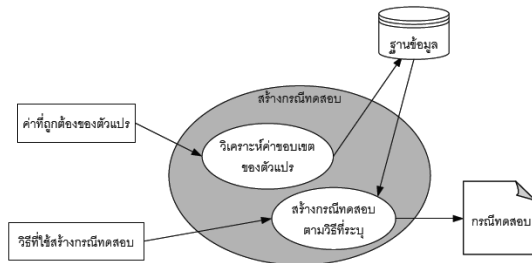


ภาพที่ 7 การสร้างกรณีทดสอบโดยใช้ชั้นสมมูล

3.3.2 วิธีการสร้างกรณีทดสอบโดยใช้ค่าขอบเขต

วิธีการสร้างกรณีทดสอบโดยใช้ค่าขอบเขต เริ่มต้นโดยนำค่าที่ถูกต้องของตัวแปรมาวิเคราะห์ค่าขอบเขตของตัวแปร ประกอบด้วยค่าที่น้อยกว่าค่าขอบเขตที่ต่ำที่สุด (min-) ค่าขอบเขตที่ต่ำที่สุด (min) ค่าที่มากกว่าค่าขอบเขตที่ต่ำที่สุด (min+) ค่าปกติที่อยู่ในขอบเขต (nom) ค่าที่น้อยกว่าค่าขอบเขตที่สูงที่สุด (max-) ค่าขอบเขตที่สูงที่สุด (max) และค่าที่มากกว่าค่าขอบเขตที่สูงที่สุด (max+) แล้วนำค่าที่ได้มาสร้างเป็นกรณีทดสอบตามวิธีที่เลือกใช้ในการสร้างกรณีทดสอบ ซึ่งจะสามารถเลือกได้ 4 วิธี คือ การวิเคราะห์ค่าขอบเขต (Boundary Value Analysis) การทดสอบแบบเวสต์เคส (Worst Case Testing) การทดสอบแบบโรบัสเนส (Robustness Testing) และการทดสอบแบบโรบัสเวสต์เคส (Robust Worst-Case Testing) ทั้งนี้การสร้างกรณีทดสอบโดยใช้ค่าขอบเขต จะสนใจเฉพาะค่าที่เป็น

ค่าช่วงแบบปิด (closed range) เท่านั้น โดยขั้นตอนต่างๆ
ในการสร้างกรณีทดสอบโดยใช้ค่าขอบเขตสามารถ
อธิบายได้ ดังภาพที่ 8



ภาพที่ 8 การสร้างกรณีทดสอบโดยใช้ค่าขอบเขต

4. สรุปและแนวทางการพัฒนาต่อไป

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอแนวคิดในการวิเคราะห์ผลกระทบต่อการทดสอบเมื่อมีการเปลี่ยนแปลงของโปรแกรมประยุกต์บนเว็บ ผลลัพธ์ที่ได้จากงานวิจัยนี้ทำให้ทราบว่าโปรแกรมประยุกต์บนเว็บส่วนใดบ้างที่เปลี่ยนแปลงไป มีรูปแบบของการเปลี่ยนแปลง และผลกระทบต่อการทดสอบอย่างไร รวมถึงนำเสนอวิธีการปรับปรุง หรือสร้างกรณีทดสอบขึ้นมาใหม่ เพื่อนำกรณีทดสอบที่ได้รับการปรับปรุงเรียบร้อยแล้วนั้นไปทดสอบโปรแกรมประยุกต์บนเว็บที่มีการเปลี่ยนแปลง สำหรับแนวทางการพัฒนาต่อไปในอนาคต ทางผู้วิจัยจะนำแนวคิดที่ได้แนะนำมานี้มาพัฒนาเป็นเครื่องมือเพื่อนำมาใช้งานต่อไป

เอกสารอ้างอิง

- [1] สุภาพร หมั่นเพียรสุข. “เครื่องมือสำหรับสร้างกรณีทดสอบจากแฟ้มเอกสารเอชทีเอ็มแอลและเอกซ์เอ็มแอลสคีม่า.” วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2546
- [2] สุชาติ ศุภผล. “การสร้างกรณีทดสอบสำหรับโปรแกรมประยุกต์บนเว็บด้วยเทคนิคการทดสอบแบบล็กบ็อกซ์.” วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2547
- [3] Benedikt, M. Freire, J. and Godefroid, P. “VeriWeb: Automatically Testing Dynamic Web Sites,” World Wide Web Conference Series, 2002.
- [4] Sneed, H. M. “Testing a Web Application,” Website Evolution, pp. 3–10, 2004.
- [5] Bruns, A. Kornstadt, A. and Wichmann, D. “Web Application Tests with Selenium,” IEEE Software, vol. 26, no. 5, pp. 88–91, 2009.
- [6] Huggins, J. (2013, June 8). Selenium [Online]. Available: <http://www.seleniumhq.org>
- [7] Bohner, S. A “Software Change Impacts - An Evolving Perspective,” International Conference on Software Maintenance, pp. 263–272, 2002.
- [8] Lehnert, S. “A Taxonomy for Software Change Impact Analysis,” Proceedings of the 12th International Workshop on Principles of Software Evolution and the 7th annual ERCIM Workshop on Software Evolution, New York, NY, USA, pp. 41–50, 2011.

- [9] Arnold R. S. and Bohner, S. A. "Impact Analysis - Towards a Framework for Comparison," International Conference on Software Maintenance, pp. 292-301, 1993.
- [10] Parashar, P. Bhatia, R. and Kalia, A. "Change Impact Analysis: A Tool for Effective Regression Testing," 5th International Conference, ICISTM, pp. 160-169, 2011.
- [11] W3C. (2013, June 8). W3C XML Schema [Online]. Available: <http://www.w3.org/XML/Schema.html>
- [12] Jorgensen, P.C. Software Testing: A Craftsman's Approach third edition. United States of America: CRC Press LLC, 2007.
- [13] Bassil, Y. and Alwani M. "Autonomic HTML Interface Generator for Web Applications," International Journal of Web & Semantic Technology (IJWesT), Vol.3, No.1, 2012.