

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต

FRAMELET BASED APPROACH TO FRAMEWORK
REQUIREMENT



กรกิต จิรสุนัย

KORAKIT JIRASUVINAI

อพ.
ก15271
2550

เลขหมู่.....
เลขทะเบียน..... 74839
วัน,เดือน,ปี..... 11 ต.ค. 2550

b. 118 3013x
i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**FRAMELET BASED APPROACH TO FRAMEWORK
REQUIREMENT**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN COMPUTER SCIENCE
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2007



SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต
Framelet Based Approach to Framework Requirement
ชื่อนักศึกษา นายกรกิต จิรสุนัย
รหัสประจำตัว 46063607
ปริญญา วิทยาศาสตรมหาบัณฑิต
สาขาวิชา วิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร.ศรัณย์ อินทโกสุม

คณะกรรมการสอบวิทยานิพนธ์	ลายมือชื่อ
รศ.ดร.วีระ บุญจริง	
ผศ.ดร.จิรพร ศรีสวัสดิ์	
ดร.เฉลิมศักดิ์ เลิศวงศ์เสถียร	
ผศ.ดร.ศรัณย์ อินทโกสุม	

วัน/เดือน/ปี ที่สอบ 16 มีนาคม 2550 เวลา 17.00-18.00 น.

สถานที่สอบ ณ อาคารจุฬารามวลัยลักษณ์ 1 ห้อง 219

บัณฑิตวิทยาลัยรับรองแล้ว

(ผศ.ดร.จารุวัตร เจริญสุข)

คณบดีบัณฑิตวิทยาลัย

วันที่.....11.....เดือน.....พฤษภาคม.....พ.ศ.....๒๕๕๐.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต
นักศึกษา	นายกรกิต จิรสุนัย
รหัสประจำตัว	46063607
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	วิทยาการคอมพิวเตอร์
พ.ศ.	2550
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.ศรัณย์ อินทโกสุม

บทคัดย่อ

งานวิจัยนี้นำเสนอระเบียบวิธีใหม่ในการสร้างข้อกำหนดเฟรมเวิร์กจากข้อกำหนดเฟรมเลต ระเบียบวิธีที่นำเสนอนี้จะทำให้ผู้พัฒนาเฟรมเวิร์กสามารถพัฒนาข้อกำหนดเฟรมเวิร์กได้อย่างเป็นระบบ และง่ายต่อการจัดการ สำหรับการทดสอบระเบียบวิธีดังกล่าวทำได้โดยการสร้างข้อกำหนดของโปรแกรมอีเลิร์นนิ่งสาม โปรแกรมประยุกต์ ได้แก่ มูเดิล เอติวเตอร์ และเท็น และใช้ข้อกำหนดดังกล่าวเป็นข้อมูลเข้าสำหรับสร้างข้อกำหนดของเฟรมเวิร์กสำหรับโดเมนอีเลิร์นนิ่ง

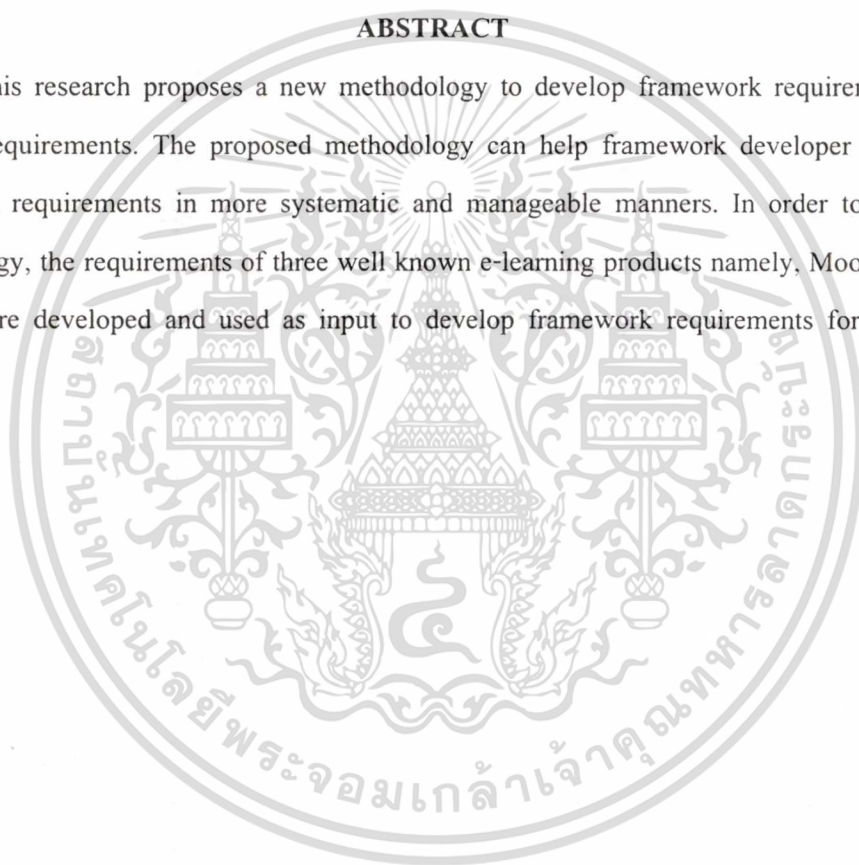


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Framelet Based Approach to Framework Requirement
Student	Mr. Korakit Jirasuvinaï
Student ID.	46063607
Degree	Master of Science
Program	Computer Science
Year	2007
Thesis Advisor	Asst.Prof.Dr.Sarum Intakosum

ABSTRACT

This research proposes a new methodology to develop framework requirements from framelct requirements. The proposed methodology can help framework developer to develop framework requirements in more systematic and manageable manners. In order to prove the methodology, the requirements of three well known e-learning products namely, Moodle, aTutor and Ten are developed and used as input to develop framework requirements for e-learning domain



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยดี ข้าพเจ้าขอกราบขอบพระคุณ ผศ.ดร. ศรัณย์ อินทโกสุม อาจารย์ผู้ควบคุมวิทยานิพนธ์ ที่กรุณาให้คำปรึกษา คำติชม คำแนะนำ แนวคิด ความเอาใจใส่ ตลอดจนการแก้ปัญหาต่างๆ อันเป็นประโยชน์ต่อวิทยานิพนธ์เป็นอย่างดี

ขอกราบขอบพระคุณ รศ.ดร. วีระ บุญจริง และ ผศ.ดร. จีรพร ศรีสวัสดิ์ สำหรับคำแนะนำ และคำปรึกษาต่างๆ และยังให้ความกรุณามาเป็นตัวแทนกรรมการจากภาควิชา

ขอกราบขอบพระคุณ ดร. เฉลิมศักดิ์ เลิศวงศ์เสถียร ซึ่งให้ความกรุณาเป็นตัวแทนกรรมการจากบุคคลภายนอก ทำให้ได้รับคำปรึกษา และคำแนะนำต่างๆ ในการสอบวิทยานิพนธ์

ขอกราบขอบพระคุณคณาจารย์ภาควิชาคณิตศาสตร์ และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาความรู้ต่างๆ ให้กับข้าพเจ้าในระดับปริญญาโท

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีราชมงคล ทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาความรู้ต่างๆ ในระดับปริญญาตรี ซึ่งช่วยสร้างฐานความรู้ในสาขาวิชาคอมพิวเตอร์ที่มั่นคงให้กับข้าพเจ้า

ขอกราบขอบพระคุณคณาจารย์โรงเรียนสุรศักดิ์มนตรี ซึ่งช่วยสร้างฐานความรู้ที่มั่นคง และประสบการณ์ชีวิตที่แข็งแกร่งให้ข้าพเจ้า

ขอขอบคุณ ปู๊ป พี่แก่น พี่นุช พี่ออย พี่สาว พี่ก๊ว พี่หล้า พี่จ๊ก พี่หญิง พี่ด่าย พี่โรจน์ พี่ชาย พี่น้อย พี่ต๋ม อัน วิ ต๋ม และเพื่อนๆ พี่ๆ น้องๆ ในระดับปริญญาโททุกคนที่ให้คำแนะนำ คำปรึกษา คำติชม ความช่วยเหลือต่างๆ ตลอดจนเป็นกำลังใจในการทำวิทยานิพนธ์เป็นอย่างดีเสมอมา

ขอขอบคุณ ยะ ลาว อ้วน แสน บัง เสก พฤษย์ แจ็ก บี โกว ไหมยม เด็ก จอนห์ และเพื่อนๆ พี่ๆ น้องๆ ในระดับปริญญาตรีทุกคนที่ให้คำปรึกษา ตลอดจนคำเสียสละต่างๆ จนเป็นพลังให้ข้าพเจ้าทำวิทยานิพนธ์ได้สำเร็จ

ขอขอบคุณ คุ้ง เต๋ เอส กอล์ฟ สัน ทัด พิ และเพื่อนๆ พี่ๆ น้องๆ ในระดับมัธยมศึกษาที่ให้คำปรึกษาที่ดีทั้งเรื่องวิทยานิพนธ์ และเรื่องต่างๆ ที่เกิดขึ้นในชีวิตข้าพเจ้า

สุดท้ายนี้ขอขอบพระคุณ ปาป้า माम้า พี่ต๋ม พี่ด้อม พี่ต๋ม ตลอดจนญาติๆ ของข้าพเจ้า ที่ให้การสนับสนุนด้านการเรียนเป็นอย่างดีมาโดยตลอด อีกทั้งยังเป็นกำลังใจอันยิ่งใหญ่ในการทำวิทยานิพนธ์ฉบับนี้ให้สำเร็จลุล่วงด้วยดี

สำหรับคุณงานความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้แก่ผู้มีพระคุณทุกท่านทั้งที่ได้กล่าวในข้างต้น และที่ไม่ได้กล่าว

กรกิต จิรสุนัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูป.....	VI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานการศึกษา.....	2
1.4 ขอบเขตการวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	4
2.1 ความรู้พื้นฐานเรื่องเฟรมเวอร์ก.....	4
2.1.1 ความหมายของเฟรมเวอร์ก.....	4
2.1.2 ประเภทของเฟรมเวอร์ก.....	6
2.1.3 คุณสมบัติที่เฟรมเวอร์กควรมี.....	6
2.1.4 ประโยชน์ของเฟรมเวอร์ก.....	7
2.2 การพัฒนาเฟรมเวอร์ก.....	8
2.3 ปัญหาของเฟรมเวอร์ก.....	11
2.4 ความรู้ทั่วไปเรื่องเฟรมเลต.....	12
2.5 ความรู้ทั่วไปเกี่ยวกับยูสเคส.....	13
2.6 ความรู้ทั่วไปเกี่ยวกับอีเลิร์นนิ่ง.....	16
2.7 ผลกระทบที่เกี่ยวข้องกับอีเลิร์นนิ่ง.....	17
2.7.1 โปรแกรมมูเดิล.....	17
2.7.2 โปรแกรมเอติวเตอร์.....	18
2.7.3 โปรแกรมแท่น.....	20

บทที่ 3 การพัฒนาความต้องการสำหรับเฟรมเวอร์กโดยใช้เฟรมเลต..... 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่อผู้ใช้ได้เห็นไปใช้ประโยชน์ที่ตนการรัก
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.1 รูปแบบข้อกำหนดสำหรับการพัฒนาเฟรมเวิร์กโดยใช้เฟรมเลต.....	22
3.1.1 ส่วนข้อมูลข้อกำหนดพื้นฐาน.....	24
3.1.2 ส่วนจัดการความแปรผัน.....	
3.2 กระบวนการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต.....	29
3.2.1 การแบ่งข้อกำหนด.....	31
3.2.2 การปรับแต่งข้อกำหนดย่อย.....	31
3.2.3 การสร้างข้อกำหนดเฟรมเลต.....	33
3.2.4 การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต.....	34
บทที่ 4 การประเมินผล.....	39
4.1 วิธีการประเมินผล.....	39
4.2 การประเมินผล.....	40
4.2.1 การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต.....	40
4.2.2 การสร้างข้อกำหนดโปรแกรมประยุกต์จากข้อกำหนดเฟรมเวิร์ก.....	54
4.2.3 การเปรียบเทียบกระบวนการในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต กับการสร้างข้อกำหนดเฟรมเวิร์กเดิม.....	67
4.3 สรุปผลการประเมิน.....	70
บทที่ 5 สรุปและข้อเสนอแนะ.....	72
5.1 สรุป.....	72
5.2 ข้อเสนอแนะ.....	73
เอกสารอ้างอิง.....	74
ประวัติผู้เขียน.....	76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แสดงส่วนต่างๆ ในการพัฒนาโปรแกรมประยุกต์โดยใช้เฟรมเวิร์ก.....	5
2.2 แสดงหน้าหลักวิชาของ โปรแกรมมูเดิล	17
2.3 แสดงหน้าหลักวิชาของ โปรแกรมเอ็ดวเเตอร์.....	19
2.4 แสดงหน้าหลักวิชาของ โปรแกรมเท็น	20
3.1 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดสำหรับเฟรมเวิร์กด้วยเฟรมเลต	23
3.2 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดสำหรับเฟรมเวิร์กด้วยเฟรมเลต ในส่วนข้อมูลข้อกำหนดพื้นฐาน	24
3.3 แสดงตัวอย่างการใช้งานสายลำดับการทำงานหลักที่มีจุดแปรผัน	25
3.4 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดสำหรับเฟรมเวิร์กด้วยเฟรมเลต ในส่วนจัดการความแปรผัน	26
3.5 แสดงตัวอย่างการใช้งานจุดแปรผัน	26
3.6 แสดงตัวอย่างการใช้งานส่วนการแสดงผลเฟรมเวิร์ก และเฟรมเลต	27
3.7 แสดงตัวอย่างที่ใช้ในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต	28
3.8 แสดงการเปรียบเทียบลักษณะของข้อกำหนดโปรแกรมประยุกต์.....	30
3.9 แสดงการแบ่งข้อกำหนดโปรแกรมประยุกต์ B.....	31
3.10 แสดงตัวอย่างของการแบ่งข้อกำหนดย่อยที่ไม่สอดคล้องกัน	32
3.11 แสดงตัวอย่างของการแบ่งข้อกำหนดที่ได้ปรับปรุงให้สอดคล้องกัน	33
3.12 แสดงการสร้างข้อกำหนดเฟรมเลต.....	34
3.13 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อย 1	35
3.14 แสดงการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต.....	36
3.15 แสดงกระบวนการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต	37
4.1 แสดงการแบ่งข้อกำหนดโปรแกรมประยุกต์เท็นในระดับแรก	41
4.2 แสดงการแบ่งข้อกำหนดของเท็นในส่วนจัดการเนื้อหา.....	41
4.3 แสดงการสร้างข้อกำหนดเฟรมเลตสร้างเนื้อหา.....	42
4.4 แสดงข้อกำหนดสร้างเนื้อหาของ โปรแกรมประยุกต์เท็น	43
4.5 แสดงข้อกำหนดสร้างเนื้อหาของ โปรแกรมประยุกต์มูเดิล	44
4.6 แสดงข้อกำหนดสร้างเนื้อหาของ โปรแกรมประยุกต์เอ็ดวเเตอร์	45
4.7 แสดงข้อกำหนดเฟรมเลตสร้างเนื้อหา	46
4.8 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหา.....	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.9 แสดงข้อกำหนดโปรแกรมประยุกต์เห็นในส่วนจัดการเนื้อหา.....	48
4.10 แสดงข้อกำหนดโปรแกรมประยุกต์มัลติมีเดียในส่วนจัดการเนื้อหา.....	48
4.11 แสดงข้อกำหนดโปรแกรมประยุกต์เดวิดเตอร์ในส่วนจัดการเนื้อหา.....	49
4.12 แสดงข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหา.....	50
4.13 แสดงการสร้างข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์ก.....	51
4.14 แสดงข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์ก.....	51
4.15 แสดงข้อกำหนดแสดงรายชื่อนักเรียน.....	56
4.16 แสดงคำอธิบายสรุปในส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อนักเรียน.....	57
4.17 แสดงข้อกำหนดแสดงรายชื่อนักเรียนของโปรแกรมประยุกต์เห็นเดิม.....	57
4.18 แสดงสายลำดับการทำงาน และจุดแปรผันในส่วนเฟรมเวิร์กย่อยจัดการนักเรียน.....	59
4.19 แสดงข้อกำหนดโปรแกรมประยุกต์เห็นในส่วนจัดการนักเรียน.....	59
4.20 แสดงข้อกำหนดเฟรมเวิร์กแสดงรายชื่อนักเรียน.....	61
4.21 แสดงคำอธิบายสรุปในส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อนักเรียน.....	61
4.22 แสดงข้อกำหนดโปรแกรมประยุกต์เห็นในส่วนแสดงรายชื่อนักเรียน.....	62
4.23 แสดงข้อกำหนดโปรแกรมประยุกต์อีเลิร์นนิ่งแอสควอร์ในส่วนแสดงรายชื่อนักเรียน.....	64
4.24 แสดงคำอธิบายสรุปในส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อนักเรียน.....	64
4.25 แสดงสายลำดับการทำงานบางส่วนของโปรแกรมประยุกต์อีเลิร์นนิ่งแอสควอร์.....	65
4.26 แสดงสายลำดับการทำงานบางส่วนของอีเลิร์นนิ่งเฟรมเวิร์ก.....	66
4.27 แสดงสายลำดับการทำงานบางส่วนของอีเลิร์นนิ่งเฟรมเวิร์กที่ได้รับการปรับปรุง.....	66
4.28 แสดงสายลำดับการทำงานข้อกำหนดเฟรมเวิร์กจัดการวิชาจากกระบวนการเดิม.....	68
4.29 แสดงสายลำดับการทำงานข้อกำหนดเฟรมเวิร์กจัดการวิชาที่ได้จากงานวิจัยนี้.....	69
4.30 แสดงการสร้างเฟรมเวิร์กสร้างเนื้อหาใหม่เพื่อทดแทนเฟรมเวิร์กสร้างเนื้อหาเดิม.....	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เฟรมเวิร์ก (Framework) เป็นแนวทางหนึ่งในการพัฒนาโปรแกรมประยุกต์ที่มีประสิทธิภาพ ซึ่งช่วยให้การพัฒนาเป็นไปอย่างรวดเร็ว และสามารถลดต้นทุนที่เกิดขึ้นในการพัฒนาโปรแกรมประยุกต์แต่ละครั้งได้ ทั้งนี้เนื่องจากเฟรมเวิร์กเป็นการนำเอาหลักการนำกลับมาใช้ใหม่ (Reuse) [9][4] มาประยุกต์ใช้ในกระบวนการพัฒนาโปรแกรมประยุกต์ เป็นผลให้ผู้พัฒนาโปรแกรมประยุกต์สามารถนำเอาทั้งส่วนของการออกแบบ และโค้ดโปรแกรม กลับมาใช้ในการพัฒนาโปรแกรมประยุกต์ใหม่ได้ อย่างไรก็ตามหลักการวิเคราะห์และพัฒนาเฟรมเวิร์กสำหรับโปรแกรมประยุกต์นั้นค่อนข้างยุ่งยากและซับซ้อนมากกว่าหลักการพัฒนาโปรแกรมประยุกต์ทั่วไป เนื่องจากการพัฒนาเฟรมเวิร์กจำเป็นที่จะต้องพิจารณาจากกลุ่มโปรแกรมประยุกต์ซึ่งสัมพันธ์กัน เพื่อกำหนดเป็นโครงสร้างหลัก และส่วนต่อขยายของเฟรมเวิร์ก เป็นผลให้เกิดความผิดพลาดในการพัฒนาเฟรมเวิร์กได้ง่าย โดยเฉพาะอย่างยิ่งการพัฒนาเฟรมเวิร์กที่มีขนาดใหญ่ อาทิ เฟรมเวิร์กสำหรับอิเล็กทรอนิกส์ ด้วยเหตุนี้เองแนวคิดเฟรมเลต (Framelet) จึงได้รับการพัฒนาขึ้นเพื่อช่วยลดปัญหาความยุ่งยากซับซ้อน ตลอดจนความผิดพลาดที่อาจเกิดขึ้นได้ในการพัฒนาเฟรมเวิร์ก อาจกล่าวได้ว่าเฟรมเลตเป็นเฟรมเวิร์กขนาดเล็กที่มีหน้าที่การทำงานเฉพาะ มีความซับซ้อนน้อยกว่าและง่ายต่อการทำความเข้าใจ แต่ยังคงเป็นไปตามหลักการทั่วไปของเฟรมเวิร์ก ผู้พัฒนาเฟรมเวิร์กจึงสามารถนำเอาเฟรมเลตที่มีหน้าที่การทำงานแต่ละตัวมาประกอบกันเข้าเป็นเฟรมเวิร์กได้ ตลอดจนช่วยลดระยะเวลาในการวิเคราะห์ และพัฒนาเฟรมเวิร์กลงได้อย่างมีประสิทธิภาพ ซึ่งกระบวนการพัฒนาเฟรมเวิร์กนั้นประกอบด้วย ขั้นตอนการวิเคราะห์ (Analysis Phase) ขั้นตอนออกแบบ (Design Phase) ขั้นตอนการพัฒนา (Implementation Phase) และขั้นตอนการทดสอบ (Test Phase) การพัฒนาเฟรมเวิร์กโดยใช้เฟรมเลตนั้น มีกระบวนการไม่ต่างไปจากการพัฒนาเฟรมเวิร์กเดิมมากนัก อย่างไรก็ตามในบางขั้นตอนของการพัฒนาเฟรมเวิร์กโดยใช้เฟรมเลตนั้นมีรายละเอียดที่แตกต่างไปจากการพัฒนาเฟรมเวิร์กเดิม โดยเฉพาะอย่างยิ่งในขั้นตอนการวิเคราะห์ที่มีการแบ่งข้อกำหนดออกเป็นส่วนย่อย เพื่อลดความซับซ้อนของระบบ ซึ่งในปัจจุบันยังขาดระเบียบวิธีในการสร้างข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเลต ให้ผู้พัฒนาเฟรมเวิร์กสามารถนำระเบียบวิธีดังกล่าวไปใช้ในการพัฒนาเฟรมเวิร์ก

ดังนั้นงานวิจัยนี้จึงมีวัตถุประสงค์เพื่อนำเสนอระเบียบวิธีในการสร้างข้อกำหนดเฟรม

เวิร์กจากข้อกำหนดของเฟรมเลต โดยทำการศึกษาและออกแบบกระบวนการ ตลอดจนเครื่องมือที่เอกรสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำเป็นต่อการใช้งาน และทำการทดสอบระเบียบวิธีดังกล่าว โดยการสร้างข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตในโดเมน (Domain) อีเลิร์นนิ่ง ซึ่งเป็นโดเมนที่มีขนาดใหญ่เพียงพอสำหรับนำมาใช้ในการทดสอบ เพื่อทำการทดสอบความถูกต้อง และประสิทธิภาพของระเบียบวิธีที่นำเสนอในงานวิจัยนี้

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

เพื่อนำเสนอระเบียบวิธีใหม่สำหรับการสร้างข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลต โดยนำเสนอกระบวนการและเครื่องมือที่มีอยู่มาปรับปรุง และประยุกต์ใช้ให้เหมาะสม กับระเบียบวิธีการที่ได้นำเสนอ

1.3 สมมติฐานการศึกษา

ระเบียบวิธีในการพัฒนาข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตที่ได้นำเสนอในงานวิจัยนี้ จะเป็นแนวทางหนึ่งที่คุณพัฒนาเฟรมเวิร์กสามารถนำไปใช้เพื่อสร้างข้อกำหนดสำหรับเฟรมเวิร์กที่เหมาะสม และข้อกำหนดเฟรมเวิร์กที่ได้สามารถทำความเข้าใจ และรองรับการเปลี่ยนแปลงได้มากกว่าข้อกำหนดเฟรมเวิร์กจากกระบวนการเดิม

1.4 ขอบเขตการวิจัย

1.4.1 งานวิจัยนี้มุ่งศึกษาและพัฒนาระเบียบวิธีในการสร้างข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลต โดยใช้กระบวนการและเครื่องมือที่มีอยู่มาปรับปรุง และประยุกต์ใช้ให้เหมาะสม

1.4.2 กลุ่มผู้ใช้งานระเบียบวิธีที่ได้จากงานวิจัย คือ กลุ่มผู้เกี่ยวข้องกับการพัฒนาข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลต

1.4.3 กรณีศึกษาที่นำมาใช้ในการประเมินผลระเบียบวิธีการสร้างข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตนี้ จะประกอบด้วยโปรแกรมประยุกต์จำนวน 3 โปรแกรม คือ โปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และ โปรแกรมประยุกต์เอ็ดิตเตอร์ ซึ่งโปรแกรมประยุกต์ทั้งสามนี้ เป็นโปรแกรมประยุกต์ที่อยู่ภายใต้โดเมนอีเลิร์นนิ่ง และจะเน้นเฉพาะฟังก์ชันที่เกี่ยวข้องกับการทำงานสำหรับกลุ่มผู้สร้างบทเรียนเท่านั้น

1.4.4 ในกระบวนการพัฒนาข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตจากงานวิจัยนี้ จะเน้นในส่วนโดเมนปัญหา (Problem Domain) โดยในส่วนฐานข้อมูล (DataBase) และส่วนต่อประสานผู้ใช้ (User Interface) ในงานวิจัยนี้จะไม่แสดงรายละเอียดภายในส่วนดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้และจะเปิดเผยหรือจะตีพิมพ์ในสิ่งพิมพ์ใด ๆ ก็ตามให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 คำอธิบายยูสเคสที่ได้จากงานวิจัยนี้จะสามารถนำไปใช้ในการสร้างข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตได้อย่างมีประสิทธิภาพ

1.5.2 ผู้พัฒนาข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตสามารถนำระเบียบวิธีที่ได้จากงานวิจัยนี้ ไปใช้เพื่อเป็นแนวทางในการพัฒนาข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎี และงานวิจัยที่เกี่ยวข้องกับการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต ซึ่งแบ่งการนำเสนอออกเป็นเจ็ดส่วน โดยส่วนแรกจะนำเสนอความรู้พื้นฐานเกี่ยวกับเฟรมเวิร์กซึ่งประกอบด้วย ความหมายของเฟรมเวิร์ก ประเภทของเฟรมเวิร์ก คุณสมบัติที่เฟรมเวิร์กควรมี และประโยชน์ของเฟรมเวิร์ก ส่วนที่สองเสนอความรู้ในเรื่องการพัฒนาเฟรมเวิร์ก ส่วนที่สามเป็นการอธิบายถึงปัญหาของเฟรมเวิร์ก ส่วนที่สี่เสนอความรู้ทั่วไปเรื่องเฟรมเลต ส่วนที่ห้าเสนอความรู้ทั่วไปเกี่ยวกับข้อกำหนด ส่วนที่หกเสนอความรู้ทั่วไปเกี่ยวกับอิเล็กทรอนิกส์ และส่วนสุดท้ายเสนอผลิตภัณฑ์ที่เกี่ยวข้องกับอิเล็กทรอนิกส์ ตามลำดับ

2.1 ความรู้พื้นฐานเกี่ยวกับเฟรมเวิร์ก

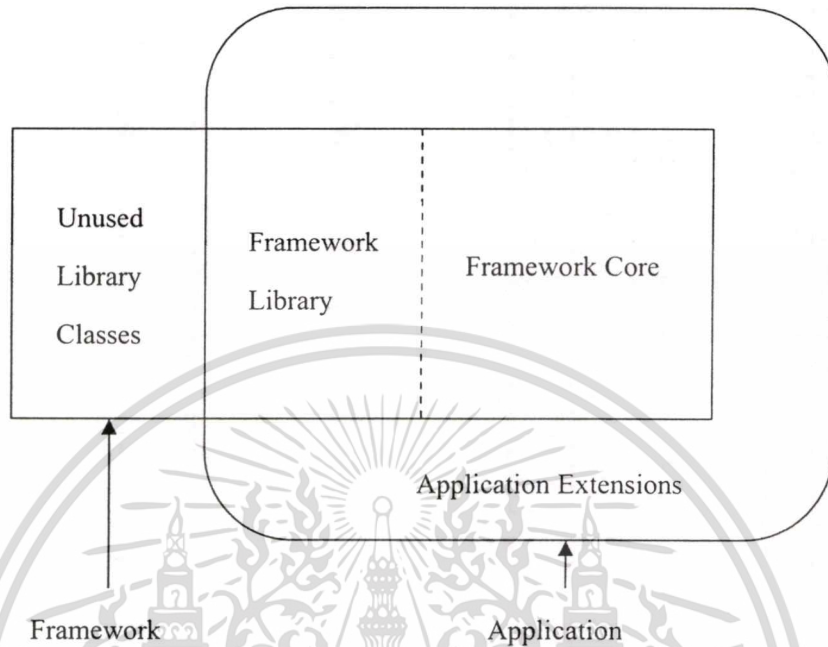
2.1.1 ความหมายของเฟรมเวิร์ก

เฟรมเวิร์กเป็นเทคนิคหนึ่งในการนำกลับมาใช้ใหม่สำหรับการพัฒนาโปรแกรมประยุกต์ ซึ่งช่วยให้การพัฒนาโปรแกรมประยุกต์ทำได้สะดวกรวดเร็ว และสามารถลดต้นทุนที่เกิดขึ้นจากการพัฒนาโปรแกรมประยุกต์ในแต่ละครั้ง โดยข้อดีที่สำคัญของเฟรมเวิร์ก คือ สามารถนำกลับมาใช้ใหม่ได้ทั้งการออกแบบ และ โค้ด ซึ่งเฟรมเวิร์กจะพัฒนาโครงสร้าง และขั้นตอนการทำงานที่เป็นพื้นฐานจากกลุ่มโปรแกรมประยุกต์ที่มีความสัมพันธ์กัน ผู้ใช้งานเฟรมเวิร์กเพียงแค่พัฒนาในส่วนที่ต้องการเพิ่ม หรือแก้ไขบางส่วนเข้าไปในเฟรมเวิร์กเพื่อให้ตรงกับข้อกำหนด หรือตรงกับงานที่ต้องการนำไปใช้ เฟรมเวิร์กประกอบด้วย [4] ส่วนที่สามารถเปลี่ยนแปลงได้ หรือที่เรียกกันว่าฮอตสปอต (Hot Spot) และส่วนที่ไม่สามารถเปลี่ยนแปลงได้ หรือที่เรียกกันว่าโฟรเซนสปอต (Frozen Spot) โดยฮอตสปอตเป็นส่วนที่ผู้พัฒนาเฟรมเวิร์กเห็นว่าเป็นส่วนที่สามารถทำงานได้หลากหลาย จึงพัฒนาให้ผู้ใช้งานเฟรมเวิร์กสามารถเปลี่ยนแปลงการทำงานในส่วนดังกล่าวได้ ซึ่งในการเปลี่ยนแปลงการทำงานดังกล่าวผู้ใช้งานเฟรมเวิร์กจะต้องสร้างฮุก (Hooks) เข้ามาประกอบกับฮอตสปอตต่างๆ ภายในเฟรมเวิร์ก ขณะที่โฟรเซนสปอตจะเป็นส่วนที่ผู้พัฒนาเฟรมเวิร์กเห็นว่าเป็นการทำงานพื้นฐานของเฟรมเวิร์ก และไม่ควรมีการเปลี่ยนแปลง ผู้พัฒนาเฟรมเวิร์กจึงไม่ได้พัฒนาให้ผู้ใช้งานเฟรมเวิร์กสามารถเปลี่ยนแปลงการทำงานส่วนนั้นๆ ได้

ในการพัฒนาโปรแกรมประยุกต์โดยใช้เฟรมเวิร์กสามารถแบ่งส่วนต่างๆ ในการพัฒนาได้ [6] ดังนี้ แกนเฟรมเวิร์ก (Framework Core) เฟรมเวิร์กไลบรารี (Framework Library) ส่วนต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขยายโปรแกรมประยุกต์ (Application Extensions) โปรแกรมประยุกต์ (Application) และไลบรารีที่ไม่นำมาใช้ (Unused Library Classes) ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 แสดงส่วนต่างๆ ในการพัฒนาโปรแกรมประยุกต์โดยใช้เฟรมเวิร์ก [6]

ในรูปที่ 2.1 แสดงองค์ประกอบในมุมมองของการพัฒนาโปรแกรมประยุกต์โดยใช้เฟรมเวิร์ก เพื่อแสดงให้เห็นภาพรวม โครงสร้างของเฟรมเวิร์ก และโปรแกรมประยุกต์ โดยในส่วนต่างๆ สามารถอธิบายได้ ดังนี้

1. แกนเฟรมเวิร์ก คือ ส่วนการทำงานที่เป็นโครงสร้างทั่วไป คุณสมบัติพื้นฐานของเฟรมเวิร์กสำหรับการพัฒนาโปรแกรมประยุกต์
2. เฟรมเวิร์กไลบรารี คือ ส่วนขยายหน้าที่การทำงานของแกนเฟรมเวิร์ก ซึ่งโปรแกรมประยุกต์สามารถนำไปใช้ได้โดยไม่ต้องปรับปรุง หรือปรับปรุงเพียงเล็กน้อย
3. ส่วนต่อขยายโปรแกรมประยุกต์ คือ ส่วนการทำงานเฉพาะของโปรแกรมประยุกต์ที่ไม่มีการจัดเตรียมไว้ในเฟรมเวิร์ก โดยผู้ใช้งานเฟรมเวิร์กจะต้องพัฒนาในส่วนนี้เอง
4. โปรแกรมประยุกต์ คือ โปรแกรมที่สามารถนำไปใช้ได้จริง ซึ่งในมุมมองของโปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์ก จะประกอบไปด้วย 3 ส่วน คือ แกนเฟรมเวิร์ก เฟรมเวิร์กไลบรารี และส่วนต่อขยายโปรแกรมประยุกต์
5. ไลบรารีที่ไม่นำมาใช้ คือ ส่วนของไลบรารีที่เฟรมเวิร์กได้เตรียมไว้ให้ แต่โปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์กไม่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ประเภทของเฟรมเวิร์ก

ประเภทของเฟรมเวิร์กนั้นสามารถแบ่งได้หลายลักษณะขึ้นอยู่กับลักษณะการแบ่ง ซึ่งสามารถแบ่งได้ ดังนี้

2.1.2.1 แบ่งตามขอบเขตของเฟรมเวิร์ก [6][7] ได้เป็น 3 ประเภท คือ

1. เฟรมเวิร์กสำหรับโปรแกรมประยุกต์ (Application Framework) ประกอบด้วยฟังก์ชันการทำงานที่สามารถใช้ได้กับหลายโดเมน อาทิ ส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphical User Interface : GUI) ฐานข้อมูล (Data Base) เป็นต้น

2. เฟรมเวิร์กโดเมน (Domain Framework) ประกอบด้วยฟังก์ชันการทำงานที่ใช้ได้เฉพาะสำหรับโดเมนใดโดเมนหนึ่ง อาทิ ระบบการผลิต (Manufacturing System) เฟรมเวิร์กสื่อประสม (Multimedia Framework) เป็นต้น

3. เฟรมเวิร์กสนับสนุน (Support Framework) เป็นเฟรมเวิร์กที่เตรียมการทำงานขั้นพื้นฐานของระบบ อาทิ การเข้าถึงไฟล์ (File Access) โปรแกรมขับอุปกรณ์ (Device Driver) เป็นต้น

2.1.2.2 แบ่งตามลักษณะการใช้งานของเฟรมเวิร์ก [4][6] แบ่งได้เป็น 2 ประเภท คือ

1. ไลท์บ็อกซ์เฟรมเวิร์ก (White Box Frameworks) เป็นเฟรมเวิร์กที่ประกอบด้วยคลาสนามธรรม (Abstract Class) เป็นส่วนมาก โดยในการพัฒนาโปรแกรมประยุกต์จากเฟรมเวิร์กนั้นจะอาศัยหลักการสืบทอดคุณสมบัติ (Inheritance) ซึ่งในการสืบทอดคุณสมบัตินั้น ถ้าผู้ใช้งานเฟรมเวิร์กต้องการพัฒนาบางส่วนเพิ่มเติม ผู้ใช้งานเฟรมเวิร์กนั้นจำเป็นต้องเข้าใจรายละเอียดการทำงานของส่วนการทำงานต่างๆ และโค้ดโปรแกรมภายในเฟรมเวิร์กเป็นอย่างดีจึงทำงานได้อย่างมีประสิทธิภาพ ในทางกลับกันการพัฒนาไลท์บ็อกซ์เฟรมเวิร์กสามารถพัฒนาได้ง่าย เนื่องจากไลท์บ็อกซ์เฟรมเวิร์กไม่จำเป็นต้องสร้างส่วนการติดต่อกับผู้ใช้งานเฟรมเวิร์ก

2. แบล็กบ็อกซ์เฟรมเวิร์ก (Black Box Frameworks) เป็นเฟรมเวิร์กที่ใช้หลักการประกอบ (Composite) ในการพัฒนา ซึ่งภายในเฟรมเวิร์กจะประกอบด้วยคอมโพเนนต์ (Component) ต่างๆ ที่เกี่ยวข้องกับเฟรมเวิร์ก ซึ่งการติดต่อระหว่างคอมโพเนนต์จะติดต่อกันผ่านส่วนการติดต่อ (Interface) ซึ่งทำให้ในการพัฒนาเฟรมเวิร์กนั้นสามารถทำได้ยาก เนื่องจากผู้พัฒนาต้องพัฒนาส่วนการติดต่อเพื่อซ่อนรายละเอียดในทุกคอมโพเนนต์นั้น แต่เนื่องจากการซ่อนส่วนต่างๆ ภายใต้อินเตอร์เฟซนั้นเอง ทำให้ผู้พัฒนาโปรแกรมประยุกต์จากเฟรมเวิร์กสามารถพัฒนาได้ง่าย เนื่องจากผู้พัฒนาไม่จำเป็นต้องรู้โครงสร้าง และโค้ดโปรแกรมภายในเพียงรู้ส่วนการติดต่อก็สามารถนำไปพัฒนาโปรแกรมประยุกต์ได้

2.1.3 คุณสมบัติที่เฟรมเวิร์กควรมี

คุณสมบัติที่เฟรมเวิร์กควรมี [6] มีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ความสะดวกในการใช้ (Ease of Use) เฟรมเวิร์กที่ดีควรง่ายต่อการทำความเข้าใจ และง่ายในการพัฒนาโปรแกรมประยุกต์ ซึ่งความสะดวกในการใช้งานดังกล่าวถือเป็นคุณสมบัติสำคัญที่เฟรมเวิร์กควรมี

2. ความสามารถในการเพิ่มขยาย (Extensibility) คือ เฟรมเวิร์กควรที่จะพัฒนาให้สามารถเพิ่มขยายฟังก์ชันการทำงานต่างๆ ของเฟรมเวิร์กได้ง่าย เพื่อรองรับข้อกำหนดที่เพิ่มขึ้น เช่น อีเลิร์นนิ่งเฟรมเวิร์กที่เริ่มแรกมีฟังก์ชันในการทำงานอยู่ 2 ฟังก์ชัน คือ การทำสื่อการสอน การลงทะเบียนเรียน ต่อมาต้องการให้มีฟังก์ชันการทำแบบทดสอบ ก็สามารถเพิ่มฟังก์ชันการทำงานดังกล่าวให้กับเฟรมเวิร์กได้ง่าย เป็นต้น

3. ความสามารถในการยืดหยุ่น (Flexibility) ในการพัฒนาเฟรมเวิร์กที่ดีนั้นควรคำนึงถึงเรื่องความยืดหยุ่น เพื่อให้ผู้ใช้งานเฟรมเวิร์กสามารถที่จะนำเฟรมเวิร์กไปพัฒนาโปรแกรมประยุกต์ได้หลากหลาย เช่น ในการพัฒนาเฟรมเวิร์กให้มีจุดเชื่อมต่อเพื่อเปลี่ยนแปลงการทำงานในส่วนนั้นๆ ก็จะสามารถนำเฟรมเวิร์กนั้นไปใช้ได้หลากหลายมากยิ่งขึ้น

4. ความสมบูรณ์ (Completeness) เฟรมเวิร์กไม่สามารถที่จะครอบคลุมการทำงานทุกอย่างในโดเมน แต่เฟรมเวิร์กควรเตรียมการทำงานที่เป็นพื้นฐานทั้งหมดที่จำเป็น เพื่อให้ผู้ใช้งานเฟรมเวิร์กสามารถใช้งานได้ง่าย

5. ความต้องกัน (Consistency) ภายในเฟรมเวิร์กควรมีความต้องกันระหว่างส่วนการติดต่อ หรือ โครงสร้างของคลาส โดยในการทำงานสำหรับคลาสต่างๆ อาจมีเมทอดในการรับค่า ก็ควรที่จะใช้ชื่อที่สอดคล้องกันในการรับค่า เพื่อให้ผู้พัฒนาโปรแกรมประยุกต์จากเฟรมเวิร์กสามารถทำความเข้าใจได้ง่าย และลดระยะเวลาในการพัฒนาด้วย เช่น ในการพัฒนาคลาสต่างๆ ที่มีเมทอด get เหมือนกัน ก็ควรใช้ชื่อ get ตามด้วยชื่อสิ่งที่ได้รับ เช่น getWeight getSize เป็นต้น

อย่างไรก็ตามคุณสมบัติที่กล่าวมานั้นอาจจะขัดแย้งกัน เช่น ความสะดวกในการใช้งาน กับความสามารถในการยืดหยุ่นนั้นค่อนข้างที่จะขัดแย้งกัน ในการพัฒนาให้เฟรมเวิร์กมีความยืดหยุ่นสูงนั้น จะต้องพัฒนาเฟรมเวิร์กให้เฟรมเวิร์กนั้นมีจุดเชื่อมต่อให้เปลี่ยนแปลงการทำงานค่อนข้างมาก ก็จะทำให้ความสะดวกในการใช้งานเฟรมเวิร์กนั้นลดน้อยลง ผู้พัฒนาเฟรมเวิร์กจึงควรเลือกว่าต้องการให้เฟรมเวิร์กที่พัฒนาขึ้นมาี้คุณสมบัติใดมากกว่ากัน

2.1.4 ประโยชน์ของเฟรมเวิร์ก

ประโยชน์ของเฟรมเวิร์ก [6] มีดังต่อไปนี้

1. นำกลับมาใช้ใหม่ (Reuse) จัดว่าเป็นประโยชน์ที่สำคัญของเฟรมเวิร์กเนื่องจากเฟรมเวิร์กสามารถนำกลับมาใช้ใหม่ได้ทั้งการออกแบบ และโค้ด โดยสามารถพัฒนาโปรแกรมประยุกต์ใหม่หลายโปรแกรมได้จากเฟรมเวิร์กตัวเดียว

2. การบำรุงรักษา (Maintenance) การบำรุงรักษาโปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์ก สามารถบำรุงรักษาได้ง่าย เนื่องจาก โปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์กนั้นมี โครงสร้าง และ โค้ดพื้นฐาน เหมือนกัน

3. คุณภาพ (Quality) คุณภาพของโปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์กจะมีคุณภาพ ที่ดี เนื่องจากในการพัฒนาเฟรมเวิร์กนั้น ได้ผ่านขั้นตอนต่างๆ ที่สามารถรับประกันได้ว่า โปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์กจะมีคุณภาพที่ดี เช่น ในการพัฒนาเฟรมเวิร์กจะต้อง อาศัยผู้เชี่ยวชาญใน โดเมนนั้นในการออกแบบโครงสร้างการทำงาน ทำให้โปรแกรมประยุกต์ที่ พัฒนาจากเฟรมเวิร์กนั้นเหมือนมีการออกแบบจากผู้เชี่ยวชาญใน โดเมนนั้นด้วย และในการพัฒนา เฟรมเวิร์กนั้นจะผ่านขั้นตอนในการทดสอบว่าสามารถนำไปประยุกต์ใช้กับโปรแกรมประยุกต์ได้ เป็นอย่างดี

2.2 การพัฒนาเฟรมเวิร์ก

การพัฒนาเฟรมเวิร์กสำหรับ โดเมนใด โดเมนหนึ่งนั้น ผู้พัฒนาจะต้องแน่ใจว่าเฟรมเวิร์ก ที่พัฒนาจะสามารถใช้ได้เป็นเวลานาน เนื่องจากการพัฒนาเฟรมเวิร์กนั้นสามารถพัฒนาได้ยาก และใช้เวลาในการพัฒนามากกว่าการพัฒนาโปรแกรมประยุกต์ทั่วไป จากการศึกษางานวิจัย เกี่ยวกับการแนวทางในการพัฒนาเฟรมเวิร์ก สามารถแบ่งแนวทางในการพัฒนาเฟรมเวิร์กได้ 2 วิธี [19] คือ วิธีการพัฒนาเฟรมเวิร์กจากโปรแกรมประยุกต์ใน โดเมนเดียวกันที่มีอยู่แล้ว และ วิธีการพัฒนาเฟรมเวิร์กจากการวิเคราะห์ โดเมน ซึ่งในการพัฒนาเฟรมเวิร์กจากโปรแกรม ประยุกต์ใน โดเมนเดียวกันที่มีอยู่แล้วนั้นเริ่มจากการพัฒนาโปรแกรมประยุกต์ใน โดเมนเดียวกัน ขึ้นมา 3 โปรแกรม [15] แล้วเห็นว่าสามารถนำมาพัฒนาเป็นเฟรมเวิร์กได้ จึงนำฟังก์ชันการทำงาน ที่เป็นพื้นฐานสำหรับ โดเมนนั้นมาพัฒนาเป็นเฟรมเวิร์ก ส่วนวิธีการพัฒนาเฟรมเวิร์กจากการ วิเคราะห์ โดเมน เป็นการพัฒนาเฟรมเวิร์กจากข้อกำหนดของ โดเมนที่ต้องการพัฒนาเฟรมเวิร์ก โดยดูจากฟังก์ชันการทำงานของ กลุ่ม โปรแกรมประยุกต์ที่สัมพันธ์กันในการพัฒนา [4] ซึ่งการ พัฒนาเฟรมเวิร์กทั้ง 2 วิธีนั้นมีข้อดีข้อเสียภายในตัวเอง อาทิ การพัฒนาเฟรมเวิร์กจากโปรแกรม ประยุกต์ใน โดเมนเดียวกันที่มีอยู่แล้ว นั้นสามารถพัฒนาเฟรมเวิร์กได้ง่าย และรวดเร็ว แต่ก่อนที่ จะพัฒนาเฟรมเวิร์กได้นั้นจำเป็นที่จะต้องพัฒนาโปรแกรมประยุกต์ 3 โปรแกรมทำให้เสียเวลาใน การพัฒนาโปรแกรมประยุกต์เป็นเวลานาน ส่วนการพัฒนาเฟรมเวิร์กจากการวิเคราะห์ โดเมนมี ประโยชน์มากกว่า ในกรณีที่สามารถพัฒนาเฟรมเวิร์กได้โดยไม่ต้องมีการพัฒนาโปรแกรม ประยุกต์ก่อนหน้า มีเพียงข้อกำหนดก็สามารถพัฒนาเฟรมเวิร์กได้ แต่การพัฒนาเฟรมเวิร์กวิธีนี้ ทำได้ยากกว่าการพัฒนาเฟรมเวิร์กวิธีแรก ซึ่งมีงานวิจัยที่เกี่ยวกับการพัฒนาเฟรมเวิร์ก อาทิ งานวิจัยเรื่องการพัฒนาเฟรมเวิร์กเชิงวัตถุ (Development of Object-Oriented Frameworks) ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ในเชิงการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Landin N. และคณะ [10] งานวิจัยเรื่องระเบียบวิธีในการพัฒนาเฟรมเวิร์กเชิงวัตถุ โดยใช้ยูเอ็มแอล (A UML-based Object-Oriented Framework Development Methodology) ของ Young Y.J. และคณะ [19] เป็นต้น โดยงานวิจัยทั้ง 2 ที่นำเสนอนี้ เป็นการพัฒนาเฟรมเวิร์กจากการวิเคราะห์โดเมน ซึ่งงานวิจัยดังกล่าวมีขั้นตอนการทำงานดังนี้

งานวิจัยเรื่องการพัฒนาเฟรมเวิร์กเชิงวัตถุของ Landin N. และคณะ [10] จะเริ่มพัฒนาจากการวิเคราะห์ข้อกำหนดสำหรับโดเมนนั้นๆ ได้ออกแบบขั้นตอนการพัฒนาเฟรมเวิร์กออกเป็น 5 ขั้นตอน คือ การวิเคราะห์โดเมน การระบุข้อกำหนดและวิเคราะห์ข้อกำหนด การออกแบบเฟรมเวิร์ก การพัฒนาเฟรมเวิร์ก และการทดสอบ โดยมีรายละเอียดดังนี้

1. การวิเคราะห์โดเมน เป็นขั้นตอนในการเตรียมตัวก่อนการพัฒนาเฟรมเวิร์ก วางขอบเขตของเฟรมเวิร์กที่ต้องการพัฒนา
2. ระบุข้อกำหนด และวิเคราะห์ข้อกำหนด เป็นขั้นตอนในการค้นหาข้อกำหนดของเฟรมเวิร์กทั้งหมด โดยในงานวิจัยนี้อาศัยยูสเคสเป็นเครื่องมือในการรวบรวมข้อกำหนด
3. การออกแบบเฟรมเวิร์ก เป็นการนำข้อมูลที่ได้จากขั้นตอนที่ 2 มาออกแบบคลาสต่างๆ เพื่อนำไปพัฒนาเฟรมเวิร์กต่อไป
4. การพัฒนาเฟรมเวิร์ก เป็นการพัฒนาเฟรมเวิร์กเพื่อใช้งานจริงโดยอาศัยการออกแบบโครงสร้างดังที่ได้จัดเตรียมไว้แล้วในขั้นตอนก่อนหน้า
5. การทดสอบ เป็นขั้นตอนในการทดสอบความถูกต้องของเฟรมเวิร์ก ซึ่งมีกระบวนการในการทดสอบคล้ายกับกระบวนการในการทดสอบของโปรแกรมประยุกต์ทั่วไป อาทิ การทดสอบแบบหน่วย (Unit Testing) การทดสอบแบบรวม (Integration Testing) เป็นต้น

งานวิจัยเรื่องระเบียบวิธีในการพัฒนาเฟรมเวิร์กเชิงวัตถุ โดยใช้ยูเอ็มแอลของ Young Y.J. และคณะ [19] อาศัยยูเอ็มแอลเป็นเครื่องมือในการพัฒนา ซึ่งเป็นการพัฒนาเฟรมเวิร์กจากการวิเคราะห์โดเมน โดยในระเบียบวิธีประกอบไปด้วย 4 ส่วนหลัก คือ การวิเคราะห์เฟรมเวิร์ก การออกแบบเฟรมเวิร์ก การพัฒนาเฟรมเวิร์ก และการทดสอบเฟรมเวิร์ก ซึ่งงานวิจัยนี้ได้ทดสอบระเบียบวิธีที่ได้โดยการพัฒนาเฟรมเวิร์กสำหรับการค้าขายทางอิเล็กทรอนิกส์ (Electronic Commerce) โดยมีรายละเอียดขั้นตอนการพัฒนา ดังนี้

1. การวิเคราะห์เฟรมเวิร์ก เป็นขั้นตอนในการรวบรวมข้อกำหนดของเฟรมเวิร์กที่ต้องการ ซึ่งในการรวบรวมข้อกำหนดของเฟรมเวิร์กประกอบไปด้วย 6 ส่วน ดังนี้
 - 1.1 นำโปรแกรมประยุกต์ที่มีอยู่มาวิเคราะห์หาข้อกำหนดที่เหมือนกันของแต่ละโปรแกรมประยุกต์ หรือข้อกำหนดพื้นฐานของแต่ละโดเมน
 - 1.2 พัฒนาข้อกำหนดของเฟรมเวิร์ก จากข้อกำหนดที่เหมือนกันของโปรแกรมประยุกต์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 1.3 พัฒนายูสเคสสำหรับเก็บข้อกำหนดทั้งหมดของเฟรมเวิร์ก ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 แยกส่วนยูสเคสโดยการกำหนดระดับของข้อกำหนดเป็น 3 ส่วน ดังนี้

1.4.1 ระดับชั้นพื้นฐาน (Foundation layer) คือ ระดับชั้นของข้อกำหนดการบริการของระบบ

1.4.2 ระดับชั้นธุรกิจทั่วไป (Common Business layer) คือ ระดับชั้นของข้อกำหนดที่สามารถใช้ได้หลายโดเมน

1.4.3 ระดับชั้นธุรกิจแกน (Core Business layer) คือ ระดับชั้นของข้อกำหนดที่เป็นฟังก์ชันพื้นฐานของโดเมนนั้น

1.5 แบ่งกลุ่มภายในแต่ละระดับชั้น โดยแบ่งตามความสัมพันธ์ระหว่างข้อกำหนด

1.5.1 ข้อกำหนดที่ไม่ขึ้นกับฟังก์ชันการทำงาน ไม่จัดอยู่ในกลุ่มใดเลย

1.5.2 ข้อกำหนดที่ใช้สืบทอดคุณสมบัติให้ยู่กลุ่มเดียวกัน

1.5.3 ข้อกำหนดที่ใช้หลักการเรียกใช้จัดให้ยู่กลุ่มเดียวกัน

หลังจากที่ได้จัดกลุ่มเรียบร้อยแล้วแต่ละกลุ่มจะเป็นหนึ่งเฟรมเวิร์ก จากนั้นทำการสร้างคำอธิบายข้อกำหนดสำหรับเฟรมเวิร์ก

1.6 สร้างขั้นตอนการทำงานระหว่างแต่ละเฟรมเวิร์ก โดยใช้เอกทิวทัศน์ไดอะแกรม (Activity Diagram)

2. การออกแบบเฟรมเวิร์ก เป็นการออกแบบและกำหนดโครงสร้างของเฟรมเวิร์ก และจัดทำเอกสารของเฟรมเวิร์ก ซึ่งในส่วนของกรอบประกอบไปด้วย 10 ส่วน ดังนี้

2.1 พัฒนาคลาสไดอะแกรม (Class Diagram) จากข้อกำหนดของแต่ละโปรแกรมประยุกต์

2.2 พัฒนาซีควเอนไดอะแกรม (Sequence Diagram) สำหรับโปรแกรมประยุกต์

2.3 กำหนดจุดทดสอบสำหรับเฟรมเวิร์กจากการทำงานที่ต่างกันของแต่ละโปรแกรมประยุกต์

2.4 พัฒนาซีควเอนไดอะแกรมสำหรับเฟรมเวิร์ก

2.5 กำหนดแบบรูปการออกแบบ (Design Pattern) ที่จะใช้

2.6 กำหนดประเภทของเฟรมเวิร์กว่าจะทำเป็นไวท์บ็อกซ์เฟรมเวิร์ก หรือแบล็กบ็อกซ์เฟรมเวิร์ก

2.7 พัฒนาเฟรมเวิร์กไดอะแกรม (Framework Diagram) เป็นการกำหนดการติดต่อกันสำหรับแต่ละเฟรมเวิร์ก และสายลำดับการทำงานระหว่างคลัสในเฟรมเวิร์ก

2.8 การออกแบบและพัฒนาคลาสต่างๆ ภายในเฟรมเวิร์ก

2.9 ตรวจสอบการออกแบบเฟรมเวิร์ก ทั้งหมดที่ได้ออกแบบมาทั้ง ซีควเอนไดอะแกรม คลาสไดอะแกรม และเฟรมเวิร์กไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 ทำเอกสารสำหรับเฟรมเวิร์ก ซึ่งเป็นเอกสารในการอธิบายเฟรมเวิร์ก และคู่มือการพัฒนาโปรแกรมประยุกต์จากเฟรมเวิร์ก

3. การพัฒนาเฟรมเวิร์ก เป็นการพัฒนาคلاس และส่วนการติดต่อ ซึ่งในส่วนการพัฒนา ประกอบไปด้วย 2 ส่วน ดังนี้

3.1 พัฒนาคلاس และส่วนการติดต่อของเฟรมเวิร์ก เป็นการพัฒนาคلاسต่างๆ และสายลำดับการทำงานระหว่างคลาสต่างๆ ภายในเฟรมเวิร์ก

3.2 พัฒนาพัฒนาโปรแกรมประยุกต์จากเฟรมเวิร์ก โดยพัฒนาส่วนเฉพาะ สำหรับโปรแกรมประยุกต์ และพัฒนาฮุกต่างๆ ที่ต้องการ

4. ขั้นตอนการทดสอบ เป็นขั้นตอนในการทดสอบความถูกต้องของเฟรมเวิร์ก ซึ่งในส่วนของการทดสอบประกอบไปด้วย 2 ส่วน ดังนี้

4.1 การทดสอบแบบหน่วย เป็นการทดสอบภายในแต่ละส่วนก่อนการรวมคลาส ต่างๆ

4.2 การทดสอบแบบรวม เป็นการทดสอบการทำงานหลังจากการรวมคลาส ทั้งหมดของเฟรมเวิร์ก

จากงานวิจัยที่ได้กล่าวไปข้างต้นจะเห็นได้ว่าขั้นตอนในการพัฒนาเฟรมเวิร์กนั้น จะมีขั้นตอนหลักในการพัฒนา 4 ขั้นตอน เช่นเดียวกับขั้นตอนการพัฒนาโปรแกรมประยุกต์ทั่วไป แต่จะมีรายละเอียดภายในแตกต่างจากการพัฒนาโปรแกรมประยุกต์ โดยส่วนมากงานวิจัยที่ได้กล่าว ข้างต้นจะเน้นที่ขั้นตอนการวิเคราะห์ เนื่องจากการพัฒนาเฟรมเวิร์กนั้นจำเป็นต้องวิเคราะห์ ข้อกำหนดของกลุ่มโปรแกรมประยุกต์ที่สัมพันธ์กัน มาวิเคราะห์หาส่วนการทำงานที่เหมือนกันมา สร้างเป็นโครงสร้างหลักของเฟรมเวิร์ก ทำให้ขั้นตอนการวิเคราะห์มีความยุ่งยากซับซ้อนกว่า ขั้นตอนอื่นๆ ในกระบวนการพัฒนาเฟรมเวิร์ก ซึ่งจากงานวิจัยที่ได้กล่าวมาข้างต้นทำให้เราสามารถพัฒนาเฟรมเวิร์ก ได้ง่ายขึ้น โดยอาศัยงานวิจัยดังกล่าวเป็นต้นแบบในการพัฒนา แต่อย่างไรก็ตามเฟรมเวิร์กก็ยังมีปัญหา ดังที่จะได้กล่าวในหัวข้อถัดไป

2.3 ปัญหาของเฟรมเวิร์ก

เฟรมเวิร์กเป็นหลักการในการพัฒนาโปรแกรมประยุกต์ที่มีประสิทธิภาพ อย่างไรก็ตามเฟรมเวิร์กยังมีปัญหาในการพัฒนาเช่นกัน ซึ่งงานวิจัย [2] นำเสนอปัญหาของเฟรมเวิร์ก โดยมีรายละเอียด ดังต่อไปนี้

1. การเทียบเคียง (Mapping) คือ ผู้ที่ต้องการพัฒนาโปรแกรมประยุกต์จากเฟรมเวิร์ก เลือกเฟรมเวิร์กไม่เหมาะสมกับข้อกำหนดของโปรแกรมประยุกต์ หรืออาจเกิดจากผู้ใช้งานที่นำเฟรมเวิร์กไปใช้ยังไม่เข้าใจ และทราบขอบเขตของเฟรมเวิร์กที่จะพัฒนาดีพอ

เอกสารนี้เป็นเอกสารทบทวนเนื้อหาสำหรับการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ความเข้าใจฟังก์ชัน (Understanding Functionality) คือ เฟรมเวิร์กส่วนมากมีขนาดใหญ่ และประกอบไปด้วยฟังก์ชันการทำงานภายในจำนวนมาก ผู้ใช้งานเฟรมเวิร์กอาจจำเป็นต้องศึกษาเกี่ยวกับฟังก์ชันการทำงานทั้งหมดภายในเฟรมเวิร์กถึงจะใช้งานได้อย่างมีประสิทธิภาพ ซึ่งบางฟังก์ชันอาจเป็นส่วนที่ไม่ได้ใช้เลยก็เป็นได้

3. ความเข้าใจการโต้ตอบ (Understanding Interactions) เนื่องจากเฟรมเวิร์กประกอบด้วยคลาสภายในจำนวนมาก ซึ่งคลาสเหล่านั้นจะต้องติดต่อสื่อสารกันไปมา ยิ่งคลาสมากเท่าไรก็ยิ่งทำให้การติดต่อสื่อสารเหล่านั้นมากขึ้นด้วย ทำให้ผู้ใช้งานเฟรมเวิร์กทำความเข้าใจการติดต่อสื่อสารเหล่านั้นได้ยาก เช่น การใช้งานไวท์บ็อกซ์เฟรมเวิร์กจำเป็นจะต้องเข้าใจการติดต่อสื่อสารของคลาสเหล่านั้น เพื่อที่จะนำเฟรมเวิร์กนั้นไปใช้งานได้อย่างมีประสิทธิภาพ

4. ความเข้าใจโครงสร้างของเฟรมเวิร์ก (Understanding the Framework Architecture) เนื่องจากส่วนมากนั้นเฟรมเวิร์กจะมีขนาดใหญ่ และมีโครงสร้างที่ซับซ้อน ยากในการทำความเข้าใจ ทำให้การพัฒนา และการบำรุงรักษานั้นทำได้ยาก

จะเห็นได้ว่าปัญหาที่ได้กล่าวมาข้างต้นส่วนมากเกิดจากปัญหาความไม่เข้าใจโครงสร้างการทำงาน และองค์ประกอบของเฟรมเวิร์ก เนื่องจากส่วนมากเฟรมเวิร์กมีขนาดใหญ่ และมีความซับซ้อนสูง ทำให้เฟรมเวิร์กสามารถพัฒนาได้ยาก และอาจมีข้อผิดพลาดได้ง่าย นอกจากนี้ปัญหาดังกล่าวเฟรมเวิร์กยังมีปัญหาอื่นๆ เช่น เฟรมเวิร์กรุ่นแรกๆ ภายในเฟรมเวิร์กจะมีตัวควบคุมหลักของทั้งเฟรมเวิร์กภายในตัวเฟรมเวิร์กเอง [4] การรวมเฟรมเวิร์กจะทำให้มีตัวควบคุมหลักมากกว่าหนึ่งทำให้การรวมเฟรมเวิร์กทำได้ยาก เป็นต้น ซึ่งวิธีการหนึ่งที่จะช่วยให้แก้ปัญหาดังกล่าวในของเฟรมเวิร์ก คือ การนำเฟรมเลตเข้ามาช่วยในการพัฒนาเฟรมเวิร์ก

2.4 ความรู้ทั่วไปเรื่องเฟรมเลต

เฟรมเลต คือ เฟรมเวิร์กที่มีขนาดเล็กที่ได้จากการแบ่งเฟรมเวิร์กออกเป็นเฟรมเวิร์กย่อย (Sub-Framework) ซึ่งถ้าแบ่งออกไปจนไม่สามารถแบ่งออกได้อีกจะเรียกว่าเฟรมเลต โดยเฟรมเลตนี้จะมีความซับซ้อนน้อยกว่าเฟรมเวิร์ก เนื่องจากเฟรมเลตเป็นเพียงส่วนย่อย ไม่ซับซ้อนมากนัก จึงสามารถทำความเข้าใจได้ง่ายกว่าเฟรมเวิร์กที่มีขนาดใหญ่ ถึงแม้ว่าเฟรมเลตจะมีขนาดเล็ก แต่ยังคงลักษณะของเฟรมเวิร์ก จึงมีข้อดีไม่ต่างกับเฟรมเวิร์กที่มีขนาดใหญ่ โดยเฟรมเลตมีลักษณะ ดังนี้ [8][13][14]

1. มีขนาดเล็ก เนื่องจากเฟรมเลตเป็นส่วนย่อยการทำงานของเฟรมเวิร์ก
2. เฟรมเลตไม่ใช่ตัวควบคุมหลักของทั้งเฟรมเวิร์ก จึงสามารถรวมเฟรมเลตที่มีหน้าที่การทำงานแต่ละอย่างได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เฟรมเลตมีส่วนการติดต่อที่ชัดเจน เนื่องจากเฟรมเลตมีขนาดเล็ก ส่วนการติดต่อจึงมีความชัดเจน และสามารถทำความเข้าใจได้ง่ายกว่าส่วนการติดต่อของเฟรมเวอร์กขนาดใหญ่

จากงานวิจัยของ Intakosum S. และคณะ [8] นำเสนอการใช้เฟรมเลตสำหรับการพัฒนาเฟรมเวอร์ก ซึ่งเป็นงานวิจัยที่ให้แนวความคิดในการพัฒนาเฟรมเลตจากข้อกำหนดของโปรแกรมประยุกต์ โดยใช้ข้อกำหนดเป็นเครื่องมือในการพัฒนา โดยมีขั้นตอนการพัฒนาดังนี้

1. พัฒนาข้อกำหนดสำหรับโปรแกรมประยุกต์ที่ต้องการพัฒนา
2. พัฒนาข้อกำหนดสำหรับเฟรมเวอร์กจากข้อกำหนดของโปรแกรมประยุกต์ที่สัมพันธ์กัน
3. จัดกลุ่มข้อกำหนดที่ทำงานเหมือนกันให้เป็นข้อกำหนดสำหรับเฟรมเวอร์ก
4. แบ่งข้อกำหนดในแต่ละกลุ่มให้เล็กลงเป็นข้อกำหนดย่อย และสร้างเฟรมเวอร์กย่อยจากข้อกำหนดย่อย
5. แบ่งข้อกำหนดไปเรื่อยๆ จนกระทั่งไม่สามารถแบ่งได้แล้ว หรือแบ่งแล้วไม่มีความหมายกับโดเมนนั้นๆ
6. พัฒนาข้อกำหนดสำหรับเฟรมเลตจากข้อกำหนดย่อยที่ไม่สามารถแบ่งได้แล้ว
7. ออกแบบ พัฒนาโปรแกรม และทดสอบ เฟรมเลต
8. พัฒนาเฟรมเวอร์กโดยการรวมเฟรมเลตที่เกี่ยวข้อง

จะพบว่าการพัฒนาเฟรมเวอร์กโดยใช้เฟรมเลตดังกล่าว เป็นแนวทางในการพัฒนาที่มีประสิทธิภาพ เนื่องจากสามารถอธิบายถึงขั้นตอนในการพัฒนาเฟรมเลตได้เป็นอย่างดี ซึ่งในงานวิจัยนี้มุ่งเน้นไปที่กระบวนการวิเคราะห์ ซึ่งถือว่าเป็นขั้นตอนที่มีความสำคัญในการพัฒนาเฟรมเวอร์กโดยใช้เฟรมเลต เนื่องจากในกระบวนการวิเคราะห์นั้นมีความซับซ้อนในการวิเคราะห์ฟังก์ชันการทำงานของโปรแกรมประยุกต์ที่สัมพันธ์กัน เพื่อนำมาพัฒนาเป็นข้อกำหนดสำหรับเฟรมเวอร์ก และการแบ่งส่วนข้อกำหนดต่างๆ ให้มีขนาดเล็ก อย่างไรก็ตามงานวิจัยนี้เป็นเพียงแนวคิดในการพัฒนา ยังไม่มีการทดสอบ โดยนำมาพัฒนาเฟรมเลต และนำเฟรมเลตเหล่านั้นรวมเป็นเฟรมเวอร์ก

การพัฒนาเฟรมเวอร์ก หรือการพัฒนาเฟรมเลตนั้น ในขั้นตอนการวิเคราะห์จะอาศัยยูสเคสเป็นเครื่องมือในการรวบรวมข้อกำหนดของเฟรมเวอร์ก หรือเฟรมเลต โดยได้นำเสนอความรู้ทั่วไปเกี่ยวกับยูสเคสในหัวข้อถัดไป

2.5 ความรู้ทั่วไปเกี่ยวกับยูสเคส

ยูสเคสจัดว่าเป็นเครื่องมือที่มีประสิทธิภาพ [6] ในการรวบรวมข้อกำหนด และหน้าที่การทำงานต่างๆ ของระบบ จุดเด่นของยูสเคส คือ เป็นเครื่องมือที่ใช้งานง่าย และสะดวกต่อการเรียนรู้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเท่านั้น เมื่อผู้ใช้เห็นใบเขียวบริเวณที่นำคำ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้อย่างรวดเร็ว ยิ่งไปกว่านั้นยูสเคสยังเป็นเสมือนจุดเชื่อมต่อของกิจกรรมต่างๆ ของการพัฒนาโปรแกรมประยุกต์ได้อีกด้วย ไม่ว่าจะเป็นการเชื่อมโยงไปยังขั้นตอนการออกแบบ การเขียนโปรแกรม หรือแม้กระทั่งในการทดสอบ โดยยูสเคสเป็นการรวบรวมข้อกำหนดในมุมมองของผู้ใช้ ซึ่งประกอบด้วย แอคเตอร์ (Actor) คือผู้ที่มีส่วนเกี่ยวข้องกับระบบ และยูสเคส คือ ระบบการทำงานภายในที่เกิดขึ้น ซึ่งภายในแต่ละยูสเคสนั้นจะมีคำอธิบายยูสเคส (Use Case Description) ที่ช่วยให้สามารถทำความเข้าใจการทำงานของยูสเคสได้ดีขึ้น โดยปัจจุบันยังไม่มียานวิจัยที่เสนอคำอธิบายยูสเคสสำหรับเฟรมเวทโดยตรง มีเพียงงานวิจัยที่เกี่ยวข้องกับคำอธิบายยูสเคสทั่วไป [17] และคำอธิบายยูสเคสสำหรับเฟรมเวิร์ก [20] โดยงานวิจัยดังกล่าวยังขาดองค์ประกอบที่จำเป็นสำหรับข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเวท ซึ่งคำอธิบายยูสเคสดังกล่าวนั้นควรประกอบด้วย ส่วนที่เป็นคำอธิบายข้อมูลสำหรับยูสเคสทั่วไป และส่วนที่เป็นคำอธิบายข้อมูลเฉพาะสำหรับข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเวท โดยจากการศึกษางานวิจัยที่เกี่ยวข้องกับคำอธิบายยูสเคสที่มีอยู่นั้น มีคำอธิบายข้อมูลสำหรับยูสเคสที่หลากหลายแตกต่างกันไป แต่ในส่วนมากนั้นสามารถแบ่งองค์ประกอบสำหรับยูสเคสทั่วไปได้เป็นส่วนต่างๆ ดังนี้ คือ [17]

1. ส่วนข้อมูลพื้นฐานซึ่งเป็นใช้เป็นคำอธิบายยูสเคส ประกอบด้วย

1.1 ชื่อยูสเคส (Use Case Name) คือ ชื่อที่มีความหมายบอกลักษณะการทำงานภายในยูสเคสนั้นๆ โดยชื่อยูสเคสไม่ควรซ้ำกัน

1.2 แอคเตอร์หลัก (Primary Actor) คือ ผู้ที่กระทำต่อยูสเคสนั้น โดยตรง

1.3 ผู้เกี่ยวข้องและความสนใจ (Stakeholders and Interests) คือ ผู้ที่เกี่ยวข้องกับยูสเคสทั้งหมด และกิจกรรมที่เกิดขึ้นกับยูสเคสนั้น

1.4 เงื่อนไขก่อน (Precondition) คือ เงื่อนไขสำหรับตรวจสอบความถูกต้องก่อนที่จะเข้าใช้งานยูสเคสนั้น

1.5 เงื่อนไขหลัง (Postcondition) คือ เงื่อนไขสำหรับตรวจสอบความถูกต้องหลังจากยูสเคสนั้น

2. ส่วนสายลำดับการทำงาน ประกอบด้วย

2.1 สายลำดับการทำงาน (Main Success Scenario) คือ สายลำดับการทำงานภายในยูสเคสนั้น

2.2 สายลำดับการทำงานอื่นๆ (Extensions) คือ สายลำดับการทำงานอื่นๆ ที่เกิดขึ้นเมื่อไม่เป็นไปตามการทำงานของสายลำดับการทำงานหลัก

3. ส่วนข้อมูลเพิ่มเติม ประกอบด้วย

3.1 ข้อกำหนดพิเศษ (Special Requirements) คือ ข้อกำหนดพิเศษที่เกี่ยวข้องกับยูสเคสนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 เทคโนโลยีที่เกี่ยวข้องกับ (Technology and Data Variations List) คือ เทคโนโลยีที่นำมาใช้ในยูสเคสนั้น

จากงานวิจัยดังกล่าวเป็นเพียงองค์ประกอบสำหรับการอธิบายยูสเคสทั่วไป แต่เนื่องจากข้อกำหนดสำหรับเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตมีส่วนที่มีความแปรผัน งานวิจัยของ Bertolino A. [1] ได้นำเสนอคำอธิบายยูสเคสที่มีความแปรผัน ซึ่งเหมาะที่จะนำมาใช้ในการอธิบายยูสเคสสำหรับข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตที่มีความแปรผันค่อนข้างมาก โดยองค์ประกอบจากงานวิจัยนี้ที่สามารถนำมาประยุกต์ใช้กับคำอธิบายข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเลต คือ ความแปรผัน (Variation) ซึ่งเป็นส่วนที่ใช้อธิบายความแปรผันต่างๆ ที่เกิดขึ้นภายในยูสเคสนั้นๆ

งานวิจัยของแก่นจันทร์ [20] ได้นำเสนอคำอธิบายยูสเคสสำหรับเฟรมเวิร์ก ซึ่งมีองค์ประกอบเพิ่มเติมเพื่อให้มีความเหมาะสมกับการใช้งานเฟรมเวิร์กได้อย่างครบถ้วน และองค์ประกอบดังกล่าวสามารถนำมาใช้กับคำอธิบายยูสเคสสำหรับข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเลตได้ ซึ่งในที่นี้จะนำเสนอรูปแบบคำอธิบายยูสเคสที่สามารถนำมาประยุกต์ใช้กับรูปแบบคำอธิบายยูสเคสดังกล่าว ซึ่งประกอบด้วย

1. ส่วนฮอตสปอตเป็นส่วนที่สามารถทำการเปลี่ยนแปลงได้ภายในเฟรมเวิร์ก ประกอบด้วย

1.1 ฮอตสปอตสนับสนุน (Support Hotspot) คือ ส่วนการอธิบายความแปรผันที่ปรากฏอยู่ในส่วนต่างๆ ของยูสเคสซึ่งควรมีการจัดเตรียมไว้ตั้งแต่การพัฒนาเฟรมเวิร์ก

1.2 ฮอตสปอตเสริม (Open Hotspot) คือ ส่วนการอธิบายความแปรผันที่ปรากฏอยู่ในส่วนต่างๆ ของยูสเคสซึ่งสามารถเพิ่มเติมได้ในอนาคต

2. ส่วนฮุกเป็นส่วนที่อธิบายส่วนที่สามารถนำมาเชื่อมต่อกับส่วนฮอตสปอต ประกอบด้วย

2.1 ฮุกสนับสนุน (Support Hook) คือ ส่วนการอธิบายฮุกที่สามารถนำมาเชื่อมต่อเพื่อการใช้งานภายในเฟรมเวิร์ก

งานวิจัยข้างต้นเป็นเพียงการเสนอคำอธิบายยูสเคสสำหรับการพัฒนาเฟรมเวิร์กเท่านั้น แต่การใช้งานเฟรมเวิร์กยังจำเป็นต้องอาศัยฮุกเพื่อประกอบเข้ากับฮอตสปอตต่างๆ ภายในเฟรมเวิร์ก เพื่อให้ผู้ใช้งานเฟรมเวิร์กสามารถใช้งานเฟรมเวิร์กได้ โดยได้มีงานวิจัยของ Fayad E.M. และคณะ [4] ได้เสนอรูปแบบคำอธิบายฮุก ซึ่งประกอบด้วย

1. ชื่อ (Name) เป็นส่วนที่ใช้แสดงชื่อฮุกนั้นๆ โดยชื่อฮุกนั้นไม่ควรที่จะซ้ำกัน

2. ข้อกำหนด (Requirement) เป็นส่วนที่ใช้แสดงข้อความต่างๆ ที่ใช้อธิบายฮุกว่าเป็นฮุกที่ใช้ทำอะไร มีลักษณะการทำงานอย่างไร เป็นต้น

3. ประเภท (Type) เป็นส่วนที่บอกประเภทของฮุก

4. ขอบเขต (Area) เป็นส่วนที่ใช้ในการบอกส่วนของเฟรมเวิร์กที่เกี่ยวข้องกับฮุก

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การใช้ (Uses) เป็นส่วนที่ใช้แสดงชุดต่างๆ ที่เกี่ยวข้องกับชุด
 6. ผู้ที่มีส่วนร่วม (Participant) เป็นส่วนที่ใช้แสดงคอมโพเนนต์ที่มีส่วนร่วมกับชุด
 7. การเปลี่ยนแปลง (Changes) เป็นส่วนที่ใช้อธิบายขั้นตอนการทำงานต่างๆ ภายในชุด ที่จะไปเปลี่ยนแปลงจุดยอดสโตนภายในเฟรมเวิร์ก โดยขั้นตอนการทำงานนี้สามารถเป็นได้หลายลักษณะ เช่น สายลำดับการทำงาน โค้ด โปรแกรม เป็นต้น
 8. ข้อจำกัด (Constraint) เป็นส่วนที่ใช้แสดงข้อจำกัดต่างๆ ภายในชุด
 9. หมายเหตุ (Comments) เป็นส่วนที่ใช้อธิบายชุดเพิ่มเติม
- จากงานวิจัยข้างต้นเกี่ยวกับคำอธิบายยูสเคสทั้งหมดจะเห็นได้ว่าคำอธิบายยูสเคสดังกล่าว ยังขาดส่วนที่ใช้ในอธิบายข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเสต เนื่องจากภายในข้อกำหนดดังกล่าวจะประกอบไปด้วยเฟรมเสตต่างๆ ภายในจำนวนมาก จึงต้องอาศัยคำอธิบายยูสเคสที่สามารถแสดงให้เห็นถึงส่วนประกอบภายในเฟรมเวิร์กต่างๆ ได้ เพื่อที่จะสามารถใช้งานข้อกำหนดเฟรมเวิร์กจากข้อกำหนดของเฟรมเสตได้อย่างมีประสิทธิภาพ

2.6 ความรู้ทั่วไปเกี่ยวกับอีเลิร์นนิ่ง

อีเลิร์นนิ่ง (E-Learning) คือ การเรียนรู้ผ่านสื่ออิเล็กทรอนิกส์ทุกประเภท แต่ในปัจจุบันคอมพิวเตอร์เป็นสื่ออิเล็กทรอนิกส์ที่สำคัญที่สุด [22] ดังนั้นโดยทั่วไปจึงถือว่าอีเลิร์นนิ่งเป็นการเรียนรู้ผ่านคอมพิวเตอร์ ซึ่งมีทั้งที่เป็นแบบออนไลน์ (on-line) ผ่านระบบเครือข่าย และแบบออฟไลน์ (off-line) ในปัจจุบันสถาบันการศึกษาต่างๆ ได้นิยมนำอีเลิร์นนิ่งมาใช้ในการเรียนการสอนอย่างแพร่หลายไม่ว่าจะเป็น โรงเรียน วิทยาลัย หรือมหาวิทยาลัย

อีเลิร์นนิ่งประกอบไปด้วย 4 องค์ประกอบหลัก [21] ได้แก่

1. เนื้อหา (Content) เนื้อหาเป็นองค์ประกอบที่สำคัญที่สุด เนื่องจากเนื้อหาเป็นส่วนที่ผู้สอนสร้างขึ้นเพื่อให้ผู้เรียนทำความเข้าใจ และการที่ผู้เรียนจะสามารถทำความเข้าใจเนื้อหาดังกล่าวได้หรือไม่นั้นก็ขึ้นอยู่กับผู้สอนว่าสามารถทำสื่อการสอนให้ผู้เรียนสามารถทำความเข้าใจได้ง่ายหรือไม่
2. ระบบบริหารจัดการรายวิชา (Course Management System) เป็นเหมือนเครื่องมือที่ใช้อำนวยความสะดวกในการเรียนการสอน ระบบบริหารจัดการรายวิชาที่ดีนั้นจะต้องสามารถทำได้อย่างสะดวก ง่ายต่อความเข้าใจ เพื่อไม่ให้ผู้สอน และผู้เรียนเบื่อกว่าในการทำความเข้าใจ
3. การติดต่อสื่อสาร (Communication) การที่ผู้เรียนและผู้สอนสามารถติดต่อสื่อสารกันได้ เป็นประโยชน์อย่างมากในการเรียนการสอน เนื่องจากผู้เรียนนั้นอาจจะมีข้อสงสัยต่างๆ ในเนื้อหาที่ผู้สอนได้สร้างไว้

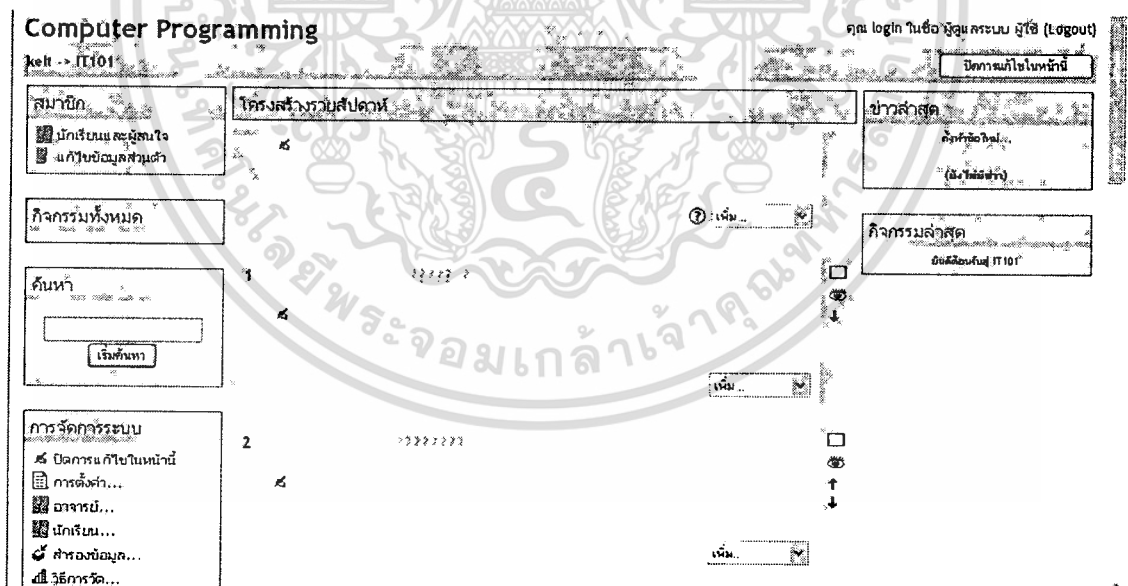
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. แบบฝึกหัด/แบบทดสอบ (Test) เป็นเหมือนกับการทดสอบผู้เรียนว่าเข้าใจบทเรียนหรือไม่

2.7 ผลิตภัณฑ์ที่เกี่ยวข้องกับอีเลิร์นนิ่ง

ปัจจุบันโปรแกรมอีเลิร์นนิ่งได้มีการใช้งานอย่างแพร่หลาย ทั้งที่เป็นเชิงพาณิชย์และที่ให้บริการโดยไม่เสียค่าใช้จ่าย โปรแกรมอีเลิร์นนิ่งที่ใช้กันอยู่ในปัจจุบันนั้นมีทั้งส่วนที่เหมือนกันและส่วนที่แตกต่างกันไปแล้วแต่ผู้ผลิต แต่ส่วนมากแล้วก็มีองค์ประกอบทั้ง 4 ส่วนหลัก คือ เนื้อหา ระบบบริหารจัดการรายวิชา การติดต่อสื่อสาร และแบบฝึกหัด/แบบทดสอบ ให้ใช้อย่างครบถ้วน การที่เราจะเลือกโปรแกรมอีเลิร์นนิ่ง โปรแกรมใด โปรแกรมหนึ่งมาใช้นั้น ก็ควรจะศึกษาความสามารถของโปรแกรมอีเลิร์นนิ่งให้ดีกว่าตรงๆกับข้อกำหนดของเรามากน้อยเพียงใด ตัวอย่างของโปรแกรมเกี่ยวกับอีเลิร์นนิ่ง ได้แก่

2.7.1 โปรแกรมมูเดิล (Moodle) เป็นโปรแกรมอีเลิร์นนิ่งที่มีความสามารถมาก โปรแกรมหนึ่ง และยังเป็นโปรแกรมที่สามารถดาวน์โหลดมาใช้ได้ โดยไม่เสียค่าใช้จ่าย ดังรูปที่ 2.2 แสดงหน้าหลักวิชาของโปรแกรมมูเดิล



รูปที่ 2.2 แสดงหน้าหลักวิชาของ โปรแกรมมูเดิล

รูปที่ 2.2 แสดงหน้าหลักวิชาของ โปรแกรมมูเดิล ซึ่งสามารถวิเคราะห์ฟังก์ชันการทำงานของโปรแกรมมูเดิลได้เป็น 6 ส่วนหลัก คือ จัดการวิชา จัดการเนื้อหา จัดการแบบทดสอบ จัดการนักเรียน จัดการไฟล์ และการสื่อสาร โดยมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลง 74839 ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1.1 จัดการวิชา ใช้สำหรับให้ผู้สร้างบทเรียนจัดการเกี่ยวกับวิชาภายในแต่ละวิชาได้ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ แสดงรายชื่อวิชา แสดงรายละเอียดวิชา และแก้ไขรายละเอียดวิชา

2.7.1.2 จัดการเนื้อหา ใช้สำหรับจัดการเกี่ยวกับเนื้อหาของวิชานั้นๆ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 5 ส่วน คือ แสดงรายชื่อเนื้อหา แสดงเนื้อหา สร้างเนื้อหา แก้ไขเนื้อหา และลบเนื้อหา

2.7.1.3 จัดการแบบทดสอบ เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับแบบทดสอบ ทั้งการสร้าง แก้ไข ตรวจ และลบแบบทดสอบ โดยสามารถสร้างแบบทดสอบได้หลายลักษณะ เช่น ตัวเลือก จับคู่ เต็มคำ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 8 ส่วน คือ แสดงแบบทดสอบ สร้างแบบทดสอบ แก้ไขแบบทดสอบ ลบแบบทดสอบ แสดงคำถาม สร้างคำถาม แก้ไขคำถาม และลบคำถาม

2.7.1.4 จัดการผู้เรียน เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับผู้เรียนได้ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ แสดงรายชื่อนักเรียน แสดงรายละเอียดนักเรียน และลบนักเรียน

2.7.1.5 จัดการไฟล์ เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับไฟล์ ที่จะใช้ในการเรียนการสอน โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ แสดงไฟล์ อัปโหลดไฟล์ และลบไฟล์

2.7.1.6 ติดต่อสื่อสาร เพื่อให้ผู้สร้างบทเรียนสามารถติดต่อสื่อสารกับผู้เรียนได้ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 2 ส่วน คือ กระดานข่าว และประกาศ ดังรูปที่ 2.10

2.7.2 โปรแกรมเอติวเตอร์ (aTutor) เป็นโปรแกรมอีเลิร์นนิ่งอีกโปรแกรมหนึ่งที่สามารถดาวน์โหลดมาใช้ได้ โดยไม่เสียค่าใช้จ่ายใดๆ จุดเด่นของเอติวเตอร์ คือ มีระบบติดต่อสื่อสารที่ดี มีรายละเอียดมาก มีเครื่องมือให้ใช้อย่างครบถ้วน ดังรูปที่ 2.3 แสดงหน้าหลักวิชาของโปรแกรมเอติวเตอร์

The screenshot shows a Moodle course web page for 'computer programming'. At the top, there is a navigation bar with 'Home', 'Forums', 'Glossary', and 'Manage'. The main content area is titled 'Home' and contains several icons for 'Forums', 'Glossary', 'Chat', 'TILE Repository Search', 'Links', 'Tests & Surveys', 'Site-map', 'Export Content', 'My Tracker', 'Polls', and 'Directory'. On the right side, there is a 'Content Navigation' sidebar with 'Home' (1) and '2 acd'. Below it are sections for 'Related Topics' (None Found), 'Users Online' (1 user, 0 guests), and a 'Glossary' section (NA). A search box is located at the bottom right. The date 'Thursday May 25, 2006' is displayed in the top right corner.

รูปที่ 2.3 แสดงหน้าหลักวิชาของโปรแกรมเอ็ดวอเตอร์

รูปที่ 2.3 แสดงหน้าหลักวิชาของโปรแกรมเอ็ดวอเตอร์ ซึ่งสามารถวิเคราะห์ฟังก์ชันการทำงานของโปรแกรมเอ็ดวอเตอร์ได้เป็น 6 ส่วนหลัก คือ จัดการวิชา จัดการเนื้อหา จัดการแบบทดสอบ จัดการนักเรียน จัดการไฟล์ และการสื่อสาร โดยมีรายละเอียดดังต่อไปนี้

2.7.2.1 จัดการวิชา เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับวิชา ภายในแต่ละวิชาได้ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 5 ส่วน คือ แสดงรายชื่อวิชา สร้างวิชา ลบวิชา แสดงรายละเอียดวิชา และแก้ไขรายละเอียดวิชา

2.7.2.2 จัดการเนื้อหา ใช้สำหรับจัดการเกี่ยวกับเนื้อหาของวิชานั้นๆ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 5 ส่วน คือ แสดงรายชื่อเนื้อหา แสดงเนื้อหา สร้างเนื้อหา แก้ไขเนื้อหา และลบเนื้อหา

2.7.2.3 จัดการแบบทดสอบ เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับแบบทดสอบ ทั้งการสร้าง แก้ไข ตรวจสอบ และลบแบบทดสอบ โดยสามารถสร้างแบบทดสอบได้หลายลักษณะ เช่น ตัวเลือก จับคู่ เติมคำ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 8 ส่วน คือ แสดงแบบทดสอบ สร้างแบบทดสอบ แก้ไขแบบทดสอบ ลบแบบทดสอบ แสดงเนื้อหา สร้างเนื้อหา แก้ไขเนื้อหา และลบเนื้อหา

2.7.2.4 จัดการผู้เรียน เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับผู้เรียน โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ แสดงรายชื่อนักเรียน เพิ่มนักเรียน และระงับนักเรียน

2.7.2.5 จัดการไฟล์ เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับไฟล์ ที่จะใช้ในการเรียนการสอน โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ แสดงไฟล์ อัปโหลดไฟล์ และลบไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.6 ติดต่อสื่อสาร เพื่อให้ผู้สร้างบทเรียนสามารถติดต่อสื่อสารกับผู้เรียนได้ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 4 ส่วน คือ กระดานข่าว ประกาศ สนทนาออนไลน์ และ เมลล์

2.7.3 โปรแกรมเท็น (Thai Education Network: TEN) เป็นโปรแกรมอีเลิร์นนิ่งในรูปแบบเชิงพาณิชย์ของบริษัทโปรแกรมเมอร์เมชั่น จำกัด ซึ่งเป็นระบบอีเลิร์นนิ่งที่สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังใช้อยู่ในปัจจุบัน จุดเด่นของเท็น คือ ระบบการสร้างเนื้อหาที่ตีมีการนำไฟล์ในรูปแบบต่างๆ มาใช้สร้างเนื้อหาได้ง่าย สามารถสร้างแบบทดสอบได้หลากหลายชนิด มีการทำสถิติของผู้เรียนที่เข้ามาใช้โปรแกรม ดังรูปที่ 2.4 แสดงหน้าหลักวิชาของโปรแกรมเท็น

The screenshot displays the user interface of the Thai Education Network (TEN). At the top, there is a header with the university logo and the text 'ระบบอีเลิร์นนิ่ง (e-Learning System)'. Below this, the user's profile is shown, including the name 'อินทิดอนรินทร์ kiserun (Serun Intakosum)', a login time of '25 พฤศจิกายน 2549 เวลา 19:24:20', and an IP address of '161.246.13.224'. The user's home page is listed as '0502330 - ระบบปฏิบัติการ' and 'ระบบอิเล็กทรอนิกส์'. A table below lists various courses, with '0502330: ระบบตัวเองตนเอง (Self Study)' selected. The user's email is '1. Serun Intakosum (Email: kiserun@kmitl.ac.th)'. The interface also shows the user's status as 'เปิดลงทะเบียน' (Registered), the course as 'วิทยาการคอมพิวเตอร์ (วท.บ.)' (Computer Science), and the current time as 'วันที่ 2 มีนาคม 2548 เวลา 11:34:18 น.'.

รูปที่ 2.4 แสดงหน้าหลักวิชาของโปรแกรมเท็น

รูปที่ 2.4 แสดงหน้าหลักวิชาของ โปรแกรมเท็น ซึ่งสามารถวิเคราะห์ฟังก์ชันการทำงาน ของโปรแกรมเท็นได้เป็น 6 ส่วนหลัก คือ จัดการวิชา จัดการเนื้อหา จัดการแบบทดสอบ จัดการ นักเรียน จัดการไฟล์ และการสื่อสาร โดยมีรายละเอียดดังต่อไปนี้

2.7.3.1 จัดการวิชา เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับวิชา และการสร้างเนื้อหา ภายในแต่ละวิชา โดยสามารถแบ่งฟังก์ชันการทำงาน ได้เป็น 3 ส่วน คือ แสดงรายชื่อวิชา แสดง รายละเอียดวิชา และแก้ไขรายละเอียดวิชา

2.7.3.2 จัดการเนื้อหา ใช้สำหรับจัดการเกี่ยวกับเนื้อหาของวิชานั้นๆ โดยสามารถ แบ่งฟังก์ชันการทำงานได้เป็น 5 ส่วน คือ แสดงรายชื่อเนื้อหา แสดงเนื้อหา สร้างเนื้อหา แก้ไข เนื้อหา และลบเนื้อหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3.3 จัดการแบบทดสอบ เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับแบบทดสอบ ทั้งการสร้าง แก้ไข ตรวจสอบ และลบแบบทดสอบ โดยสามารถสร้างแบบทดสอบได้หลายลักษณะ เช่น ตัวเลือก จับคู่ เต็มคำ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 8 ส่วน คือ แสดงแบบทดสอบ สร้างแบบทดสอบ แก้ไขแบบทดสอบ ลบแบบทดสอบ แสดงเนื้อหา สร้างเนื้อหา แก้ไขเนื้อหา และลบเนื้อหา

2.7.3.4 จัดการผู้เรียน เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับผู้เรียน โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 2 ส่วน คือ แสดงรายชื่อนักเรียน และแสดงรายละเอียดนักเรียน

2.7.3.5 จัดการไฟล์ เพื่อให้ผู้สร้างบทเรียนจัดการเกี่ยวกับไฟล์ ที่จะใช้ในการเรียนการสอน โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ แสดงไฟล์ อัปโหลดไฟล์ และลบไฟล์

2.7.3.6 ติดต่อสื่อสาร เพื่อให้ผู้สร้างบทเรียนสามารถติดต่อสื่อสารกับผู้เรียนได้ โดยสามารถแบ่งฟังก์ชันการทำงานได้เป็น 3 ส่วน คือ กระดานข่าว ประกาศ และสนทนาออนไลน์ จากตัวอย่างของโปรแกรมที่ได้กล่าวมาในข้างต้น จะเห็นได้ว่าแต่ละโปรแกรมมีส่วนที่เหมือนและแตกต่างกันไป โดยในส่วนการทำงานที่แตกต่างกันเราสามารถทำเป็นจุดยอดสปรอต เพื่อให้ผู้ใช้งานเฟรมเวิร์กสามารถเพิ่มเติมหรือปรับปรุงได้ตามข้อกำหนดของผู้ใช้งานเฟรมเวิร์ก

บทที่ 3

การสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต

ในบทนี้จะกล่าวถึงระเบียบวิธีในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต ซึ่งแบ่งการนำเสนอออกเป็น 2 ส่วน ส่วนแรกเสนอรูปแบบคำอธิบายข้อกำหนดสำหรับการพัฒนาเฟรมเวอร์กด้วยเฟรมเลต ซึ่งงานวิจัยนี้ได้ปรับปรุงรูปแบบคำอธิบายข้อกำหนดเฟรมเลต เพื่อให้เหมาะสำหรับการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต และส่วนสุดท้ายเสนอกระบวนการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต

3.1 รูปแบบข้อกำหนดสำหรับการพัฒนาเฟรมเวอร์กด้วยเฟรมเลต

จากการศึกษางานวิจัยที่ได้นำเสนอไว้แล้วในบทที่ 2 พบว่า รูปแบบการเขียนคำอธิบายข้อกำหนดที่มีอยู่ก่อนข้างมีความหลากหลายแตกต่างกันไปตามวัตถุประสงค์ของแต่ละงานวิจัย ซึ่งในบางองค์ประกอบไม่มีความจำเป็นในการอธิบายข้อกำหนดสำหรับเฟรมเลต ดังนั้นงานวิจัยนี้จึงได้ปรับปรุงรูปแบบคำอธิบายข้อกำหนดสำหรับเฟรมเลตขึ้น เพื่อให้มีความเหมาะสมกับกระบวนการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต ซึ่งในการปรับปรุงรูปแบบการเขียนคำอธิบายข้อกำหนดนี้ จะพิจารณาตามโครงสร้างของเฟรมเลต โดยลักษณะโครงสร้างของเฟรมเลตจะมีลักษณะที่คล้ายคลึงกับโครงสร้างของเฟรมเวอร์ก ซึ่งโครงสร้างของเฟรมเวอร์กประกอบด้วยแกนเฟรมเวอร์ก ส่วนต่อขยายเฟรมเวอร์ก ฮอตสปอต และฮุก [20] ซึ่งงานวิจัยนี้ได้พัฒนารูปแบบคำอธิบายข้อกำหนดสำหรับการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต โดยวิเคราะห์จากโครงสร้างของเฟรมเวอร์กเป็นหลัก ดังรูปที่ 3.1 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดสำหรับการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต

Use Case: <ชื่อที่มีความหมายสื่อถึงคุณลักษณะของข้อกำหนดนั้นๆ>
Primary Actor: <แอกเตอร์ที่ก่อให้เกิดหรือเกี่ยวข้องกับข้อกำหนดนั้นโดยตรง>
Stakeholders and Interests: <แอกเตอร์ทั้งหมดและกิจกรรมที่แอกเตอร์กระทำกับข้อกำหนดนั้น>
Preconditions: <เงื่อนไขสำหรับการตรวจสอบความถูกต้องก่อนการเข้าใช้งานข้อกำหนด>
Postconditions: <เงื่อนไขสำหรับการตรวจสอบความถูกต้องในการทำงานภายหลังการเสร็จสิ้นการทำงาน>
Main Success Scenario (or Basic Flow): <สายลำดับการทำงานภายในข้อกำหนด โดยในส่วนนี้จะมีจุดที่ใช้ชื่อว่าจุดนั้นเป็นจุดสอดสปีดภายในข้อกำหนดนั้น>
Extensions (or Alternative Flows): <สายลำดับที่เกิดขึ้นถ้าไม่เป็นไปตามการทำงานของสายลำดับการทำงานหลัก>
Variation: <ส่วนการอธิบายจุดแปรผัน>
Frameworks: <แสดงเฟรมเวิร์กที่เกี่ยวข้องกับข้อกำหนดนั้นๆ>
Framelets: <แสดงเฟรมเล็ตที่เกี่ยวข้องกับข้อกำหนดนั้นๆ>
Hotspot: <ส่วนการอธิบายสอดสปีด>

รูปที่ 3.1 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดสำหรับเฟรมเวิร์กด้วยเฟรมเล็ต

จากรูปที่ 3.1 แสดงให้เห็นองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดที่ได้รับการปรับปรุงขึ้นเพื่อรองรับกระบวนการในการพัฒนาข้อกำหนดเฟรมเวิร์กด้วยเฟรมเล็ต ซึ่งสามารถแบ่งการอธิบายขององค์ประกอบต่างๆ ของข้อกำหนดที่ได้รับการปรับปรุงออกเป็นสองส่วน คือ ส่วนข้อมูลข้อกำหนดพื้นฐานประกอบด้วย ชื่อข้อกำหนด แอกเตอร์หลัก ผู้เกี่ยวข้องและความสนใจ เงื่อนไขก่อน เงื่อนไขหลัง สายลำดับการทำงานหลัก สายลำดับการทำงานอื่นๆ ซึ่งทั้งหมดที่กล่าวมาเป็นส่วนที่นำมาจากงานวิจัยของ Larman C. [11] แต่ได้ทำการปรับปรุงในส่วนสายลำดับการทำงานหลัก และสายลำดับการทำงานอื่นๆ โดยได้ปรับปรุงให้มีการบ่งชี้ถึงความแปรผันภายในสายลำดับการทำงานหลัก และสายลำดับการทำงานอื่นๆ ส่วนจัดการความแปรผันประกอบด้วย ความแปรผัน เฟรมเวิร์ก เฟรมเล็ต และสอดสปีด ซึ่งในส่วนความแปรผันได้นำมาจากงานวิจัยของ Bertolino A. [1] และคณะ โดยได้ทำการปรับปรุงส่วนความแปรผันนี้ให้มีการบ่งชี้ถึงเฟรมเวิร์กและเฟรมเล็ตที่เกี่ยวข้อง ส่วนเฟรมเวิร์ก และเฟรมเล็ต เป็นส่วนที่งานวิจัยนี้ได้เพิ่มเติมเพื่อแสดงเฟรมเวิร์ก และเฟรมเล็ตที่เกี่ยวข้องกับข้อกำหนดนั้น สอดสปีดได้นำมาจากงานวิจัยของ แก่นจันทร์ [20] โดยมีรายละเอียดส่วนที่มีการปรับปรุงดังนี้

3.1.1 ส่วนข้อมูลข้อกำหนดพื้นฐาน

เป็นส่วนประกอบสำหรับการอธิบายข้อมูลพื้นฐานของข้อกำหนดซึ่งประกอบด้วย ชื่อข้อกำหนด แอคเตอร์หลัก ผู้ที่เกี่ยวข้องและความสนใจ เงื่อนไขก่อน เงื่อนไขหลัง สายลำดับการทำงานหลัก และสายลำดับการทำงานอื่นๆ ดังแสดงให้เห็นในรูปที่ 3.2

<p>Use Case: <ชื่อที่มีความหมายสื่อถึงคุณลักษณะของข้อกำหนดนั้นๆ></p> <p>Primary Actor: <แอคเตอร์ที่ก่อให้เกิดหรือเกี่ยวข้องกับข้อกำหนดนั้น โดยตรง></p> <p>Stakeholders and Interests: <แอคเตอร์ทั้งหมดและกิจกรรมที่แอคเตอร์กระทำกับข้อกำหนดนั้น></p> <p>Preconditions: <เงื่อนไขสำหรับการตรวจสอบความถูกต้องก่อนการใช้งานข้อกำหนด></p> <p>Postconditions: <เงื่อนไขสำหรับการตรวจสอบความถูกต้องในการทำงานภายหลังการเสร็จสิ้นการทำงาน></p> <p>Main Success Scenario (or Basic Flow): <สายลำดับการทำงานภายในข้อกำหนด โดยในส่วนนี้จะมีจุดที่ใช้ชี้ว่าจุดนั้นเป็นจุดสอตสปอตภายในข้อกำหนดนั้น></p> <p>Extensions (or Alternative Flows): <สายลำดับที่เกิดขึ้นถ้าไม่เป็นไปตามการทำงานของสายลำดับการทำงานหลัก></p>
--

รูปที่ 3.2 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดสำหรับเฟรมเวิร์กโดยใช้เฟรมเลตในส่วนข้อมูลข้อกำหนดพื้นฐาน

งานวิจัยนี้ได้ทำการปรับปรุงในบางองค์ประกอบของรูปแบบคำอธิบายข้อกำหนด เพื่อให้สามารถรองรับการทำงานในการพัฒนาข้อกำหนดในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตได้ ซึ่งในส่วนข้อมูลข้อกำหนดพื้นฐานได้ทำการปรับปรุงในส่วนสายลำดับการทำงานหลัก และสายลำดับการทำงานอื่นๆ ดังนี้

3.1.1.1 สายลำดับการทำงานหลัก

สายลำดับการทำงานหลักภายในข้อกำหนดนั้น เป็นการแสดงขั้นตอนการทำงานหลักภายในข้อกำหนดนั้นๆ ซึ่งในงานวิจัยนี้ได้ปรับปรุงภายในส่วนสายลำดับการทำงานหลัก โดยเพิ่มจุดที่ใช้บ่งชี้ถึงจุดที่มีความแปรผันภายในข้อกำหนดนั้น เพื่อช่วยให้เราสามารถทราบได้ว่ามีความแปรผันในจุดใดของสายลำดับการทำงานหลัก ซึ่งจุดแปรผันจะมีได้ 2 ลักษณะ คือ

- จุดแปรผันทางเลือก (Alternative Variation) สำหรับจุดแปรผันที่สามารถเลือกความแปรผันในจุดนั้นๆ ได้เพียงหนึ่ง หรือ ไม่เลือกเลยก็ได้ โดยใช้สัญลักษณ์ “<Alt หรือ Alternative ตามด้วยหมายเลขของจุดแปรผันนั้นๆ>”

- จุดแปรผันตัวเลือก (Option Variation) สำหรับจุดแปรผันที่สามารถเลือกความแปรผันในจุดนั้นๆ ได้มากกว่าหนึ่ง หรือไม่เลือกเลยก็ได้ โดยใช้สัญลักษณ์ “<Opt หรือ Option ตามด้วยหมายเลขของจุดแปรผันนั้นๆ>” ดังรูปที่ 3.3

Main Success Scenario (or Basic Flow):

1. ล็อกอินเข้ามาภายในระบบ
2. แสดงหน้าหลักผู้ล็อกอิน <Alt1> <Alt2>
3. เลือกวิชาที่ต้องการ
4. แสดงหน้าหลักวิชา <Alt3>
5. เลือกฟังก์ชันการทำงานต่างๆ <Opt1>

รูปที่ 3.3 แสดงตัวอย่างการใช้งานสายลำดับการทำงานหลักที่มีจุดแปรผัน

จากรูปแสดงสายลำดับการทำงานหลักส่วนหนึ่งของส่วนอีเลิร์นนิ่งเฟรมเวอร์ก ซึ่งมีจุดแปรผันภายในสายลำดับการทำงานหลัก 4 จุด โดยเป็นจุดแปรผันตัวเลือก 3 จุด และจุดแปรผันทางเลือก 1 จุด ซึ่งแต่ละจุดแปรผันนั้นจะสามารถดูรายละเอียดได้ในส่วนแสดงจุดแปรผัน ซึ่งจะได้อธิบายในลำดับถัดไป

3.1.1.2 สายลำดับการทำงานอื่นๆ

สายลำดับการทำงานอื่นๆ คือ สายลำดับการทำงานที่เกิดขึ้นเมื่อไม่เป็นไปตามการทำงานของสายลำดับการทำงานหลัก ซึ่งในงานวิจัยนี้ได้ปรับปรุงภายในส่วนสายลำดับการทำงานอื่นๆ นี้ โดยเพิ่มจุดที่ใช้บ่งชี้ว่าจุดนั้นเป็นจุดที่มีความแปรผันภายในข้อกำหนดนั้น โดยมีหลักการในการบ่งชี้เช่นเดียวกับสายลำดับการทำงานหลัก

3.1.2 ส่วนจัดการความแปรผัน

เป็นส่วนประกอบสำหรับการนำเสนอข้อมูลที่เกี่ยวข้องกับการจัดการความแปรผันของข้อกำหนด ตลอดจนการประกอบข้อกำหนดเฟรมเลตที่ได้เข้าด้วยกันเพื่อเป็นข้อกำหนดของเฟรมเวอร์กที่สมบูรณ์ ประกอบด้วย ส่วนการอธิบายความแปรผัน ส่วนการแสดงผลเฟรมเวอร์กที่เกี่ยวข้อง ส่วนการแสดงผลเฟรมเลตที่เกี่ยวข้อง ส่วนการอธิบายสอดคล้อง ดังแสดงให้เห็นในรูปที่ 3.4

Variation: <ส่วนการอธิบายจุดแปรผัน>

Frameworks: <แสดงเฟรมเวอร์กที่เกี่ยวข้องกับข้อกำหนดนั้นๆ>

Framelets: <แสดงเฟรมเลตที่เกี่ยวข้องกับข้อกำหนดนั้นๆ>

Hotspot: <ส่วนการอธิบายฮอตสปอต>

รูปที่ 3.4 แสดงองค์ประกอบของรูปแบบคำอธิบายข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตในส่วนจัดการความแปรผัน

งานวิจัยนี้ได้ทำการปรับปรุงในบางองค์ประกอบของรูปแบบคำอธิบายข้อกำหนด เพื่อให้สามารถรองรับการทำงานในการพัฒนาข้อกำหนดในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตได้ ซึ่งในส่วนจัดการความแปรผันได้ทำการปรับปรุงในส่วนจุดแปรผัน และเพิ่มเติมในส่วนเฟรมเวอร์ก และเฟรมเลต ดังนี้

3.1.2.1 จุดแปรผัน

จุดแปรผัน คือ ส่วนที่ใช้แสดงรายการทางเลือกที่สามารถเปลี่ยนแปลงได้ในจุดแปรผันนั้นๆ โดยได้นำองค์ประกอบจุดแปรผันของคำอธิบายข้อกำหนดจากงานวิจัยของ Bertolino A. และคณะ [1] เพื่อใช้อธิบายจุดแปรผันที่ปรากฏในส่วนต่างๆ โดยงานวิจัยนี้ได้ปรับปรุงให้มีการกำหนดว่าภายในจุดแปรผันนั้นประกอบด้วย ฮอตสปอต เฟรมเลต หรือเฟรมเวอร์กย่อยใดบ้าง ดังรูปที่ 3.5

Variation:

Alt1:

1. แสดงรายชื่อวิชา <FL1>

Alt2:

1. สร้างวิชา <HS1>

Alt3:

1. แสดงรายละเอียดวิชา <FL2>
2. แสดงเนื้อหา <FL3>

Opt1:

1. แสดงรายละเอียดวิชา <FL2>
2. แก้ไขรายละเอียดวิชา <FL4>
3. จัดการเนื้อหา <FW1>
4. จัดการแบบทดสอบ <FW2>

รูปที่ 3.5 แสดงตัวอย่างการใช้งานจุดแปรผัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.5 แสดงจุดแปรผันส่วนหนึ่งของอีเลิร์นนิ่งเฟรมเวิร์ก ซึ่งมีจุดแปรผันภายใน 4 จุด คือ จุดแปรผัน AI1 เป็นจุดแปรผันทางเลือก ซึ่งประกอบด้วยเฟรมเลต 1 เฟรมเลต คือ เฟรมเลต แสดงรายชื้อวิชา ส่วนจุดแปรผัน AI2 เป็นจุดแปรผันทางเลือก ซึ่งประกอบด้วยฮอตสปอต 1 ฮอตสปอต คือ ฮอตสปอตสร้างวิชา ส่วนจุดแปรผัน AI3 เป็นจุดแปรผันทางเลือก ซึ่งประกอบด้วย เฟรมเลตแสดงรายละเอียดวิชา และเฟรมเลตแสดงเนื้อหา ส่วนจุดแปรผันสุดท้ายคือส่วนแปรผัน Opt1 เป็นจุดแปรผันตัวเลือก ซึ่งประกอบด้วย เฟรมเลตแสดงรายละเอียดวิชา เฟรมเลตแก้ไขรายละเอียดวิชา เฟรมเวอร์กย่อยจัดการเนื้อหา และเฟรมเวอร์กย่อยจัดการแบบทดสอบ

3.1.2.2 ส่วนการแสดงผลเฟรมเวิร์ก และเฟรมเลต

เป็นส่วนประกอบใหม่ของคำอธิบายข้อกำหนดที่ได้ปรับปรุงขึ้นเพื่อให้สามารถอธิบายการนำข้อกำหนดของเฟรมเลตมาประกอบเป็นข้อกำหนดของเฟรมเวิร์กได้อย่างชัดเจนมากขึ้น ซึ่งในส่วนนี้ใช้สำหรับคำอธิบายข้อกำหนดที่มีการรวมข้อกำหนดเฟรมเลต หรือข้อกำหนดของเฟรมเวอร์กย่อยเข้าด้วยกัน โดยส่วนการอธิบายเฟรมเวิร์ก คือ ส่วนสำหรับการแสดงผลเฟรมเวิร์กที่เกี่ยวข้องกับข้อกำหนดนั้น และส่วนการอธิบายเฟรมเลต คือ ส่วนสำหรับการแสดงผลเลตที่เกี่ยวข้องกับข้อกำหนดนั้น ดังรูปที่ 3.6

Frameworks:
FW1: จัดการเนื้อหา
FW2: จัดการแบบทดสอบ
Framelets:
FL1: แสดงรายชื้อวิชา
FL2: แสดงรายละเอียดวิชา
FL3: แสดงเนื้อหา
FL4: แก้ไขรายละเอียดวิชา

รูปที่ 3.6 แสดงตัวอย่างการใช้งานส่วนการแสดงผลเฟรมเวิร์ก และเฟรมเลต

จากรูปที่ 3.6 ส่วนการอธิบายเฟรมเวิร์ก และเฟรมเลตส่วนหนึ่งของอีเลิร์นนิ่งเฟรมเวิร์ก ซึ่งมีเฟรมเวิร์กที่เกี่ยวข้องกับข้อกำหนดนั้น 2 เฟรมเวิร์ก คือ เฟรมเวิร์กจัดการเนื้อหา และเฟรมเวิร์กจัดการแบบทดสอบ และมีเฟรมเลตที่เกี่ยวข้องกับข้อกำหนดนั้น 4 เฟรมเลต คือ เฟรมเลต แสดงรายชื้อวิชา เฟรมเลตแสดงรายละเอียดวิชา เฟรมเลตแสดงเนื้อหา และเฟรมเลตแก้ไขรายละเอียดวิชา

ตัวอย่างคำอธิบายข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตสามารถแสดงได้ ดังรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Use Case: อีเลิร์นนิ่ง

Primary Actor: Abstract Actor

Stakeholders and Interests:

- **Abstract Actor:** ผู้สอนสามารถใช้ระบบต่างๆ ภายในระบบอีเลิร์นนิ่งได้

Preconditions: ทำการลงทะเบียนภายในระบบ

Postconditions:

Main Success Scenario (or Basic Flow):

1. ล็อกอินเข้ามาภายในระบบ
2. แสดงหน้าหลักผู้ล็อกอิน <Alt1> <Alt2>
3. เลือกวิชาที่ต้องการ
4. แสดงหน้าหลักวิชา <Alt3>
5. เลือกฟังก์ชันการทำงานต่างๆ <Opt1>

Variation:

Alt1:

1. แสดงรายชื่อวิชา <FL1>

Alt2:

1. สร้างวิชา <HS1>

Alt3:

1. แสดงรายละเอียดวิชา <FL2>
2. แสดงเนื้อหา <FL3>

Opt1:

1. แสดงรายละเอียดวิชา <FL2>
2. แก้ไขรายละเอียดวิชา <FL4>
3. จัดการเนื้อหา <FW1>
4. จัดการแบบทดสอบ <FW2>

Frameworks:

FW1: จัดการเนื้อหา

FW2: จัดการแบบทดสอบ

รูปที่ 3.7 แสดงตัวอย่างที่ใช้ในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Framelets:

FL1: แสดงรายชื่อวิชา

FL2: แสดงรายละเอียดวิชา

FL3: แสดงเนื้อหา

FL4: แก้วไขรายละเอียดยวิชา

Hotspot:

HS1: สร้างวิชา

รูปที่ 3.7 (ต่อ)

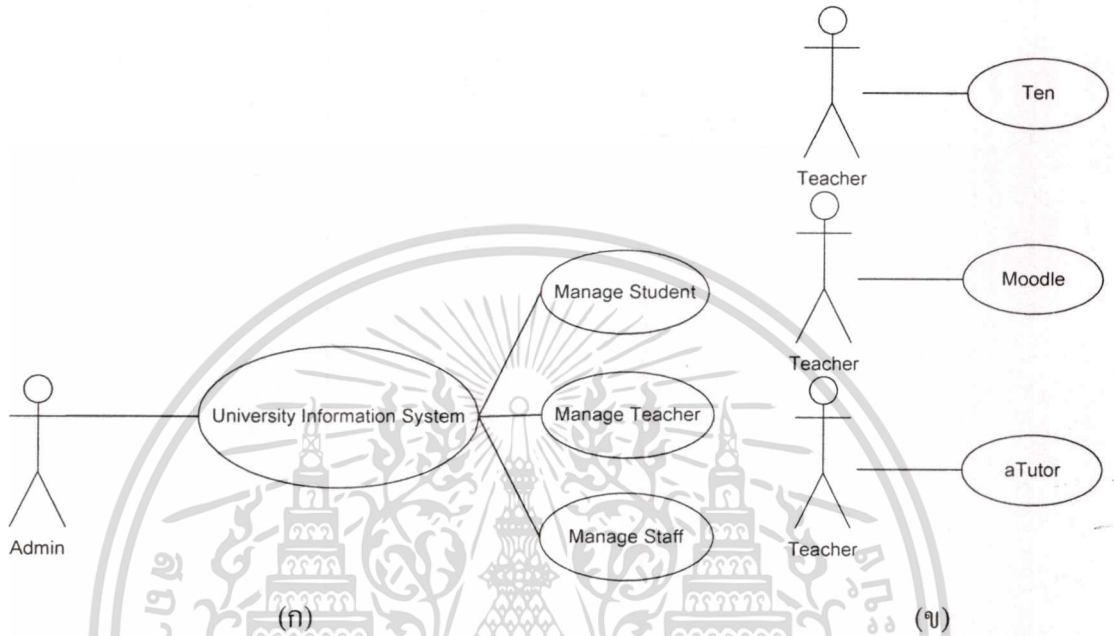
จากรูปที่ 3.7 จะเห็นได้ว่าภายในสายลำดับการทำงานหลักประกอบด้วยจุดแปรผันต่างๆ ซึ่งจุดแปรผันเหล่านั้นเกิดจากการทำงานที่แตกต่างกันระหว่างโปรแกรมประยุกต์ที่นำมาพิจารณา อาทิ จุดแปรผันทางเลือก A11 ซึ่งเป็นจุดแปรผันในการแสดงรายชื่อวิชา ซึ่งแต่ละโปรแกรมประยุกต์มีการแสดงที่แตกต่างกัน เช่น โปรแกรมประยุกต์เห็นมีการแสดงรหัสวิชา และชื่อวิชา ส่วนโปรแกรมประยุกต์เอ็ดวอเตอร์มีการแสดงรหัสวิชา ชื่อวิชา และคำอธิบายรายวิชา เป็นต้น จึงพัฒนาจุดดังกล่าวเป็นจุดแปรผันเพื่อรองรับการทำงานที่แตกต่างกันของแต่ละโปรแกรมประยุกต์ ผู้ใช้งานเฟรมเวิร์กเพียงแต่พัฒนาในจุดแปรผันดังกล่าว ก็เปลี่ยนแปลงการทำงานให้เป็นแบบที่ผู้ใช้งานเฟรมเวิร์กต้องการได้

จากรูปแบบการเขียนคำอธิบายข้อกำหนดที่ปรับปรุงนี้ จะได้นำไปใช้เป็นเครื่องมือสำหรับการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตต่อไป

3.2 กระบวนการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต

กระบวนการในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตจากการวิเคราะห์โดเมน โดยการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตนั้นจะอาศัยข้อกำหนดของโปรแกรมประยุกต์นำมาพิจารณา ซึ่งสามารถแบ่งลักษณะของโปรแกรมประยุกต์ที่นำมาพิจารณาได้เป็น 2 ลักษณะ คือ ข้อกำหนดจากโปรแกรมประยุกต์ที่มีขนาดใหญ่ ซึ่งมีฟังก์ชันการทำงานที่คล้ายคลึงกันภายในตัวของโปรแกรมประยุกต์เอง และสามารถนำมาสร้างเป็นเฟรมเวิร์กได้ ดังรูปที่ 3.8 (ก) และข้อกำหนดของโปรแกรมประยุกต์หลายโปรแกรมประยุกต์ที่อยู่ภายใต้โดเมนเดียวกัน ดังรูปที่ 3.8 (ข) โดยในการเลือกข้อกำหนดโปรแกรมประยุกต์ที่จะนำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวิร์กนั้น ถ้าเลือกโปรแกรมประยุกต์มากก็จะได้เฟรมเวิร์กที่มีความหลากหลาย แต่ในการวิเคราะห์โปรแกรมประยุกต์จำนวนมากก็สามารถทำได้ยากเช่นกัน ในงานวิจัยนี้ใช้โปรแกรมประยุกต์ในการพิจารณา 3 โปรแกรม [15] แต่สามารถที่จะนำข้อกำหนดโปรแกรมประยุกต์อื่นๆ มา เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการทดสอบภายหลังจากที่สร้างข้อกำหนดเฟรมเวิร์กเสร็จแล้ว และถ้ามีการทำงานใดที่ข้อกำหนดเฟรมเวิร์กไม่รองรับก็สามารถทำการปรับปรุงข้อกำหนดเฟรมเวิร์กดังกล่าวก่อนนำข้อกำหนดเฟรมเวิร์กไปพัฒนาในขั้นตอนการออกแบบ ขั้นตอนการพัฒนา และขั้นตอนการทดสอบ เนื่องจากการปรับปรุงข้อกำหนดก่อนขั้นตอนดังกล่าวสามารถทำได้ง่ายกว่า



รูปที่ 3.8 แสดงการเปรียบเทียบลักษณะของข้อกำหนดโปรแกรมประยุกต์

จากรูปที่ 3.8 (ก) แสดงการรวบรวมข้อกำหนดจากระบบสารสนเทศของมหาวิทยาลัย (University Information System) เพียงระบบเดียว ซึ่งมีฟังก์ชันการทำงาน และขั้นตอนการทำงานภายในที่คล้ายคลึงกัน คือ จัดการนักเรียน (Manage Student) จัดการอาจารย์ (Manage Teacher) และจัดการพนักงาน (Manage Staff) โดยส่วนการจัดการที่มีฟังก์ชันการทำงานที่คล้ายคลึงกัน เช่น ทั้ง 3 ส่วน มีฟังก์ชันในการเก็บรวบรวมข้อมูล ฟังก์ชันในการค้นหา และมีขั้นตอนในการทำงานคล้ายคลึงกัน เป็นต้น จึงสามารถนำข้อกำหนดของโปรแกรมประยุกต์ดังกล่าวมาสร้างข้อกำหนดเฟรมเวิร์กจัดการได้ และจากรูปที่ 3.8 (ข) แสดงการรวบรวมข้อกำหนดจากโปรแกรมประยุกต์ในโดเมนอีเลิร์นนิ่ง 3 โปรแกรมประยุกต์ คือ โปรแกรมประยุกต์เท็น โปรแกรมประยุกต์มูเดิล และโปรแกรมประยุกต์เอติวเตอรฺ์ ซึ่งเป็นโปรแกรมประยุกต์ภายใต้โดเมนอีเลิร์นนิ่งเช่นเดียวกัน และมีขั้นตอนการทำงานที่คล้ายคลึงกัน ซึ่งสามารถนำมาวิเคราะห์เพื่อพัฒนาเป็นเฟรมเวิร์กได้

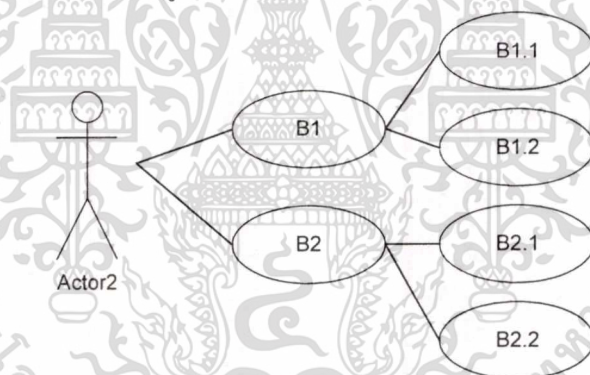
ในกระบวนการพัฒนาข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตในงานวิจัยนี้ได้ยึดหลักในการพัฒนาโปรแกรมประยุกต์ที่แบ่งโปรแกรมประยุกต์ออกเป็น 3 ส่วน คือ ส่วนฐานข้อมูล ส่วนต่อประสานกับผู้ใช้ และโดเมนปัญหา ซึ่งกระบวนการภายในงานวิจัยนี้จะเน้นการสร้างข้อกำหนดในส่วนโดเมนปัญหาเป็นหลัก เนื่องจากในส่วนฐานข้อมูล และส่วนต่อประสานกับผู้ใช้ จะเป็นส่วนที่มีความแปรผันตามแต่ผู้ใช้งานเฟรมเวิร์ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานวิจัยเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อกำหนดโปรแกรมประยุกต์ที่มีอยู่นี้จะถูกนำเข้าสู่กระบวนการวิเคราะห์ และ ออกแบบเพื่อให้ได้มาซึ่งข้อกำหนดเฟรมเลตที่สามารถนำไปพัฒนาเป็นข้อกำหนดของเฟรมเวิร์ก ที่มีประสิทธิภาพได้ โดยกระบวนการดังกล่าวจะประกอบไปด้วย 4 ขั้นตอน คือ การแบ่งข้อกำหนด การปรับแต่งข้อกำหนดย่อย การสร้างข้อกำหนดเฟรมเลต และการสร้างข้อกำหนดเฟรมเวิร์กด้วย เฟรมเลต โดยมีรายละเอียดดังต่อไปนี้

3.2.1 การแบ่งข้อกำหนด

ขั้นตอนการแบ่งข้อกำหนดเป็นการแบ่งข้อกำหนดของโปรแกรมประยุกต์ที่นำมาพิจารณา สร้างเป็นข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต ซึ่ง โดยการแบ่งข้อกำหนดของโปรแกรมประยุกต์นี้ จะแบ่งตามฟังก์ชันการทำงานเป็นหลัก และแบ่งข้อกำหนดลงไปทีละระดับจากฟังก์ชันงานขนาดใหญ่ย่อยลงไปเรื่อยๆ โดยใช้หลักการเดียวกัน แต่จะไม่แบ่งย่อยจนส่วนที่แบ่งย่อยนั้น ไม่มีความหมายกับโดเมนที่พิจารณา เมื่อได้ข้อกำหนดย่อยของ โปรแกรมประยุกต์ที่ไม่สามารถแบ่งย่อย ได้แล้วให้ทำการแบ่งย่อยข้อกำหนดของ โปรแกรมประยุกต์อื่นๆ ทุกโปรแกรมประยุกต์ที่นำมา พิจารณาโดยใช้หลักการเดียวกัน ดังรูปที่ 3.9



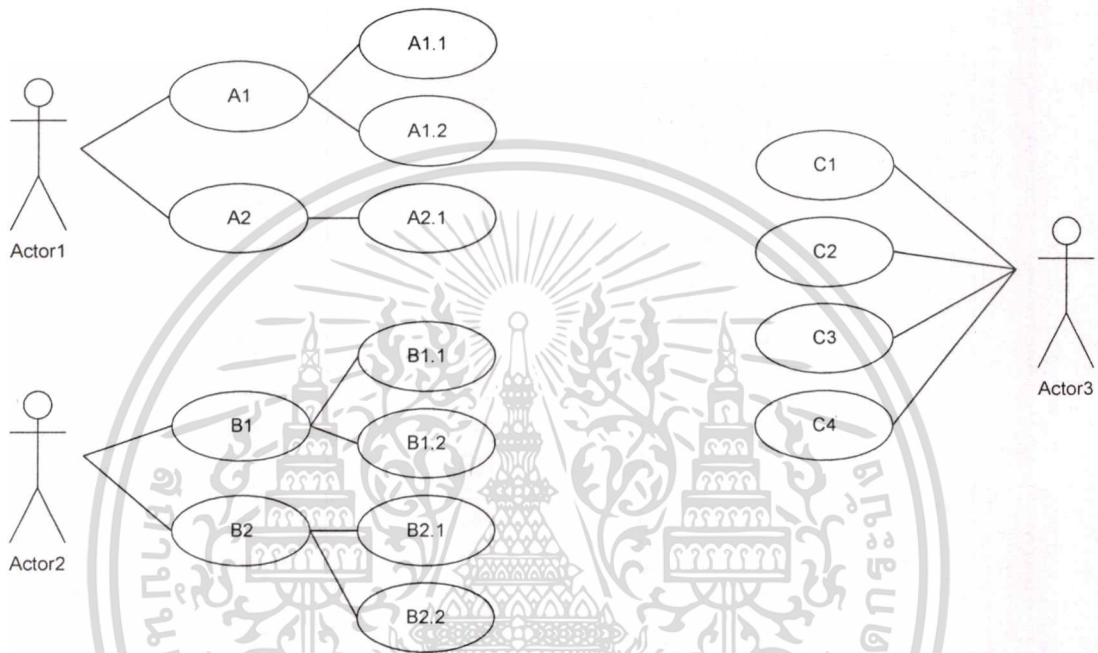
รูปที่ 3.9 แสดงการแบ่งข้อกำหนดโปรแกรมประยุกต์ B

จากรูปที่ 3.9 เป็นการแบ่งข้อกำหนดของ โปรแกรมประยุกต์ B ซึ่งเป็นการแบ่งข้อกำหนด แต่ละโปรแกรมประยุกต์ให้มีขนาดเล็กลง โดยเมื่อแบ่งข้อกำหนดของ โปรแกรมประยุกต์ B แล้วให้ ทำการแบ่งข้อกำหนดโปรแกรมประยุกต์อื่นๆ ที่นำมาพิจารณาต่อไป ซึ่งส่วนที่ได้จากการแบ่ง ข้อกำหนดนี้เมื่อนำมาสร้างเป็นข้อกำหนดเฟรมเลตจะได้เป็นข้อกำหนดเฟรมเลตที่ขึ้นกับโดเมน กล่าวคือเฟรมเลตที่ใช้ได้เฉพาะ โดเมนใด โดเมนหนึ่งเท่านั้น

3.2.2 การปรับแต่งข้อกำหนดย่อย

ขั้นตอนการปรับแต่งข้อกำหนดย่อยเป็นการปรับแต่งข้อกำหนดของแต่ละ โปรแกรมประยุกต์ที่ได้จากขั้นตอนการแบ่งข้อกำหนด ให้ข้อกำหนดของ โปรแกรมประยุกต์ทั้งหมดมีความ สอดคล้องกัน โดยเฉพาะใน โปรแกรมขนาดใหญ่ที่ต้องอาศัยทีมงานในการวิเคราะห์หลายกลุ่ม ซึ่ง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้าน การค้า เป็นไปได้ที่ทีมงานแต่ละกลุ่มจะทำการวิเคราะห์ และจัดแบ่งข้อกำหนดด้วยมุมมองที่แตกต่างกัน ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือในกรณีที่น่าเอาโปรแกรมประยุกต์มากกว่า 1 โปรแกรม มาพิจารณาก็อาจจะเป็นไปได้ที่ลำดับของโปรแกรมประยุกต์ย่อยที่ได้จากขั้นตอนการแบ่งข้อกำหนดจะไม่สอดคล้องกัน เป็นต้น ทำให้การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตทำได้ยาก ตัวอย่างของการปรับแต่งข้อกำหนดย่อย เช่น การจัดรูปแบบของข้อกำหนดย่อยให้มีระดับในการแบ่งที่เหมือนกัน ดังรูป 3.10 แสดงตัวอย่างของการแบ่งข้อกำหนดที่ไม่สอดคล้องกัน

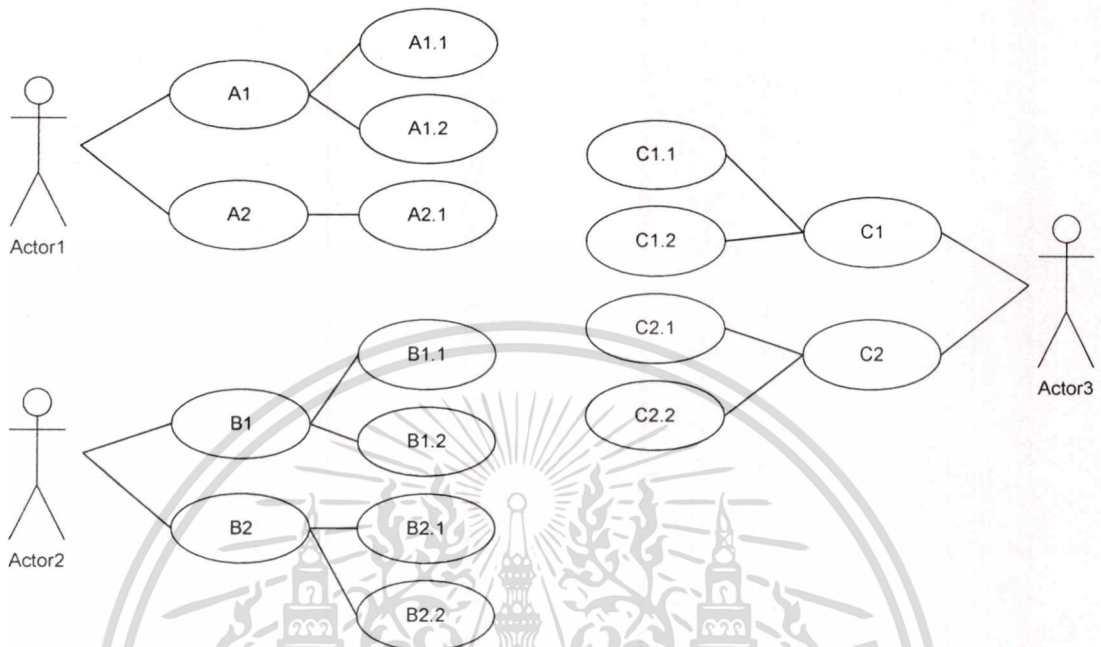


รูปที่ 3.10 แสดงตัวอย่างของการแบ่งข้อกำหนดย่อยที่ไม่สอดคล้องกัน

จากรูปที่ 3.10 โปรแกรมประยุกต์ A มีการแบ่งในระดับแรกเป็น 2 ส่วน คือ A1 A2 แล้วจึงแบ่งในระดับที่สองออกเป็น 3 ส่วนคือ A1.1 A1.2 A2.1 และโปรแกรมประยุกต์ B มีการแบ่งในระดับแรกเป็น 2 ส่วน คือ B1 B2 แล้วจึงแบ่งในระดับที่สองออกเป็น 4 ส่วนคือ B1.1 B1.2 B2.1 B2.2 แต่โปรแกรมประยุกต์ C นั้นแบ่งเป็นระดับเดียวออกเป็น 4 ส่วน คือ C1 C2 C3 C4 ซึ่งจะเห็นได้ว่าในการแบ่งข้อกำหนดโปรแกรมประยุกต์ C นั้น ไม่สอดคล้องกับข้อกำหนดของโปรแกรมประยุกต์ A และโปรแกรมประยุกต์ B โดยการแบ่งข้อกำหนดมีระดับไม่สอดคล้องกันทำให้สามารถนำข้อกำหนดดังกล่าวไปพัฒนาต่อได้ยาก จึงต้องทำการปรับแต่งข้อกำหนดดังกล่าวให้สอดคล้องกับโปรแกรมประยุกต์อื่นๆ โดยในการปรับแต่งข้อกำหนดย่อยในส่วนนี้นั้นควรจัดให้มีการแบ่งเป็นระดับลงมา ดังเช่น โปรแกรมประยุกต์ A และโปรแกรมประยุกต์ B ไม่ควรที่จะแบ่งเป็นส่วนย่อยเลย ดังเช่น โปรแกรมประยุกต์ C เนื่องจากถ้ามีการแบ่งเป็นระดับลงมาจะสามารถลดความซับซ้อนของสายลำดับการทำงานภายในข้อกำหนดเฟรมเวิร์กทั้งระบบได้ เพราะความซับซ้อนจะถูกแบ่งไปอยู่กับสายลำดับการทำงานภายในข้อกำหนดของเฟรมเวิร์กย่อย ทำให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถแยกทำความเข้าใจแต่ละส่วนได้ง่ายยิ่งขึ้น ข้อกำหนดของแต่ละโปรแกรมประยุกต์ที่สอดคล้องกันได้สามารถแสดงได้ ดังรูปที่ 3.11

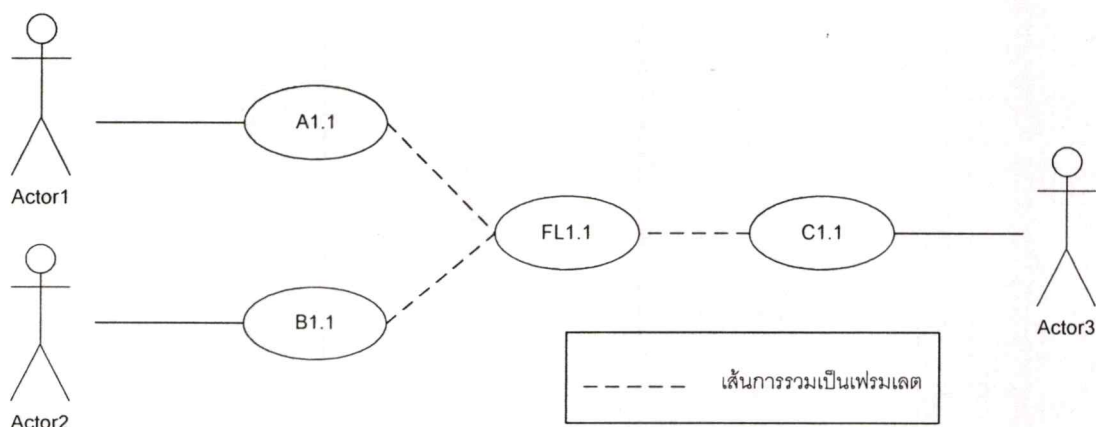


รูปที่ 3.11 แสดงตัวอย่างของการแบ่งข้อกำหนดที่ได้ปรับปรุงให้สอดคล้องกัน

จากรูปที่ 3.11 จะเห็นได้ว่าข้อกำหนดของ โปรแกรมประยุกต์ทั้ง 3 นั้นมีความสอดคล้องกัน เพื่อที่จะนำข้อกำหนดของโปรแกรมประยุกต์ดังกล่าวไปใช้ในขั้นตอนถัดไปได้

3.2.3 การสร้างข้อกำหนดเฟรมเลต

ขั้นตอนการสร้างข้อกำหนดเฟรมเลต เป็นการพัฒนาข้อกำหนดเฟรมเลตจากข้อกำหนดย่อยในส่วนเดียวกันของทุกๆ โปรแกรมประยุกต์มาพิจารณารวมกัน โดยในการสร้างข้อกำหนดเฟรมเลตนั้นจะพิจารณาหาสายลำดับการทำงานร่วมของแต่ละ โปรแกรมประยุกต์ มาสร้างเป็นสายลำดับการทำงานของเฟรมเลต โดยในส่วนการทำงานที่แตกต่างกันจะพัฒนาเป็นจุดสอดสอปอดภายในเฟรมเลต โดยถ้ามีการทำงานที่แตกต่างกันนี้อยู่ในทุก โปรแกรมประยุกต์ที่นำมาวิเคราะห์ ให้พัฒนาเป็นสอดสอปอดสนับสนุน และถ้ามีการทำงานที่แตกต่างกันนี้บาง โปรแกรมประยุกต์ที่นำมาวิเคราะห์ ให้พัฒนาเป็นสอดสอปอดเพิ่มเติม ซึ่งเมื่อผ่านขั้นตอนนี้จะได้ข้อกำหนดเฟรมเลตทั้งหมดที่สามารถนำไปสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตได้ ดังรูปที่ 3.12



รูปที่ 3.12 แสดงการสร้างข้อกำหนดเฟรมเลต

จากรูปที่ 3.12 เป็นการสร้างข้อกำหนดเฟรมเลต 1.1 โดยสร้างจากข้อกำหนดโปรแกรมประยุกต์ A1.1 โปรแกรมประยุกต์ B1.1 และ โปรแกรมประยุกต์ C1.1

โดยเมื่อผ่านขั้นตอนนี้จะ ได้ข้อกำหนดเฟรมเลตทั้งในส่วนเฟรมเลตที่ขึ้นกับ โดเมน และ ข้อกำหนดเฟรมเลตที่ไม่ขึ้นกับ โดเมน ที่สามารถนำไปใช้สร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตต่อไป

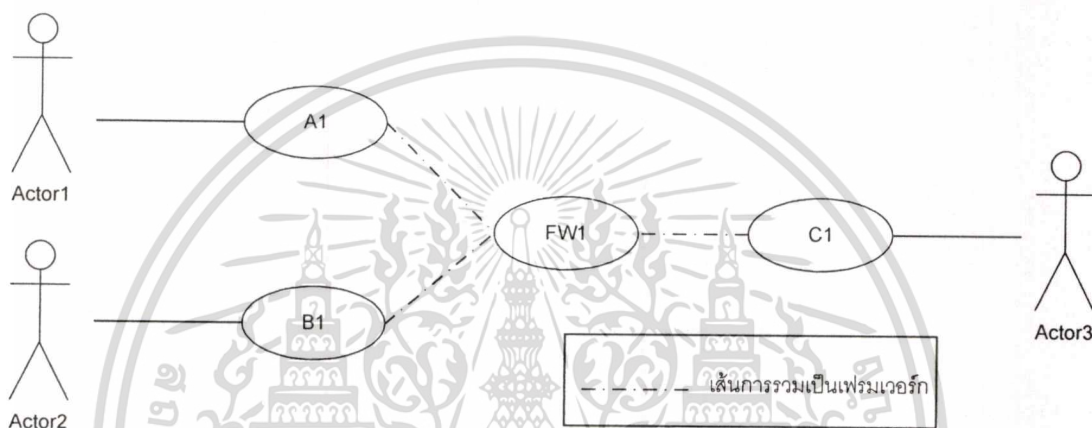
3.2.4 การสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต

เมื่อได้ข้อกำหนดเฟรมเลตจากขั้นตอนก่อนหน้า ในลำดับถัดมาผู้พัฒนาเฟรมเวอร์กจะสามารถนำเอาข้อกำหนดเฟรมเลตดังกล่าวมาสร้างข้อกำหนดเฟรมเวอร์กย้อนกลับขึ้นไปได้ โดยอาศัยจุดแปรผันในการเชื่อมต่อระหว่างข้อกำหนดเฟรมเวอร์กย่อยกับข้อกำหนดเฟรมเลต ซึ่งในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตนั้นจะเป็นการรวมข้อกำหนดเฟรมเลตในส่วนนั้นๆ เป็นเฟรมเวอร์ก ซึ่งในการสร้างข้อกำหนดของเฟรมเวอร์กนั้นจะใช้หลักการเช่นเดียวกับการสร้างข้อกำหนดเฟรมเลต โดยวิเคราะห์สายลำดับการทำงานของแต่ละโปรแกรมประยุกต์เพื่อนำมาสร้างเป็นสายลำดับการทำงานของข้อกำหนดเฟรมเวอร์ก โดยถ้าภายในสายลำดับการทำงานของแต่ละโปรแกรมประยุกต์มีขั้นตอนการทำงานที่สลับกัน แต่ผลลัพธ์การทำงานไม่มีการเปลี่ยนแปลง ควรมีการจัดสายลำดับการทำงานของแต่ละโปรแกรมประยุกต์ให้มีสายลำดับที่เหมือนกันก่อน เพื่อความสะดวกในการสร้างสายลำดับการทำงานของข้อกำหนดเฟรมเวอร์กย่อย เมื่อทำการจัดสายลำดับการทำงานของแต่ละข้อกำหนด โปรแกรมประยุกต์แล้ว ให้ทำการวิเคราะห์ข้อกำหนดของแต่ละโปรแกรมประยุกต์ที่ละชั้นว่ามีฟังก์ชันการทำงานอะไร แล้วทำการค้นหาว่ามีข้อกำหนดเฟรมเลตในส่วนฟังก์ชันการทำงานดังกล่าวที่ได้สร้างไว้ในขั้นตอนก่อนหน้าหรือไม่ ถ้ามีก็จะกำหนดให้จุดดังกล่าวเป็นจุดแปรผันที่เชื่อมต่อกับเฟรมเลตนั้นๆ แต่ถ้าไม่มีเฟรมเลตที่ได้สร้างไว้ในขั้นตอนก่อนหน้าแสดงว่าในส่วนดังกล่าวเป็นขั้นตอนการทำงานในข้อกำหนดเฟรมเวอร์กย่อยนั้นๆ แต่ผู้พัฒนาเฟรมเวอร์กก็สามารถกำหนดให้จุดดังกล่าวเป็นจุดแปรผันโดยทำการสร้างสคริปพื้นฐานไว้ให้ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาก็เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

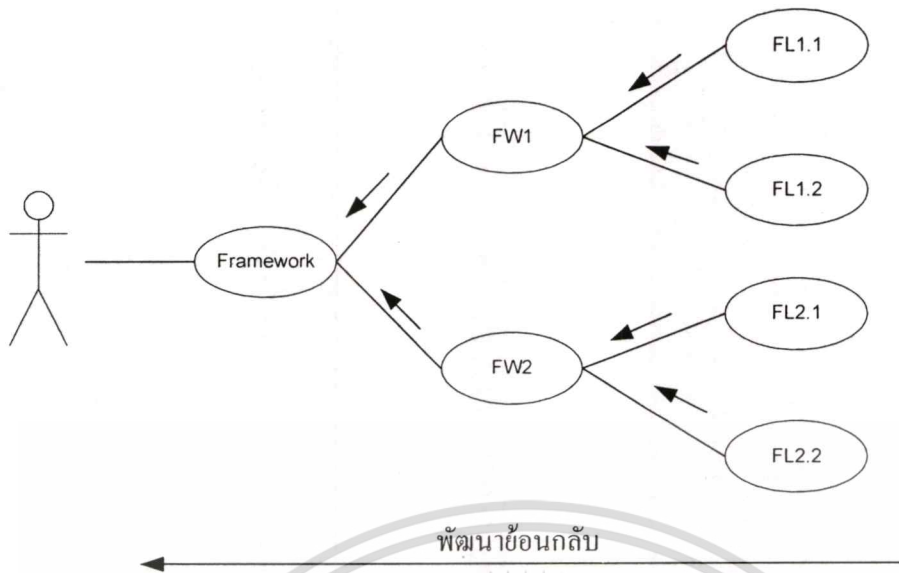
ผู้พัฒนาเฟรมเวิร์กเห็นว่าเป็นส่วนที่น่าจะมีการเปลี่ยนแปลงในภายหน้า และส่วนฟังก์ชันการทำงานที่มีเพียงหนึ่งโปรแกรมประยุกต์ที่นำมาพิจารณานั้นจะไม่มีการสร้างเป็นข้อกำหนดเฟรมเลตในขั้นตอนก่อนหน้า ในการสร้างข้อกำหนดเฟรมเวิร์กย่อยในส่วนนี้จะกำหนดให้เป็นจุดฮอตสปอตที่สามารถสร้างสยุคของฟังก์ชันการทำงานดังกล่าวเข้ามาประกอบได้ แล้วจากนั้นจึงทำการสร้างข้อกำหนดเฟรมเวิร์กในระดับต่อไปย้อนกลับไปทีละระดับจนได้ข้อกำหนดเฟรมเวิร์กที่สมบูรณ์ ดังรูปที่ 3.13 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อย 1



รูปที่ 3.13 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อย 1

จากรูปที่ 3.13 เป็นการสร้างข้อกำหนดเฟรมเวิร์กย่อยจัดการวิชา โดยสร้างจากข้อกำหนดโปรแกรมประยุกต์ A1 โปรแกรมประยุกต์ B1 และ โปรแกรมประยุกต์ C1

เมื่อสร้างข้อกำหนดเฟรมเวิร์กย่อยได้แล้วจึงทำการสร้างข้อกำหนดเฟรมเวิร์กย้อนกลับขึ้นไปทีละระดับ ดังรูปที่ 3.14



รูปที่ 3.14 แสดงการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต

จากรูปที่ 3.14 แสดงการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตโดยการสร้างย้อนกลับขึ้นมาทีละระดับ โดยเริ่มข้อกำหนดเฟรมเลต 1.1 เฟรมเลต 1.2 เฟรมเลต 2.1 และเฟรมเลต 2.2 ซึ่งเป็นส่วนที่ได้จากขั้นตอนการแบ่งข้อกำหนด ต่อมาจึงสร้างข้อกำหนดเฟรมเวิร์กย่อย 1 และเฟรมเวิร์กย่อย 2 สุดท้ายจึงสร้างข้อกำหนดเฟรมเวิร์กทั้งระบบ

โดยเมื่อผ่านขั้นตอนนี้จะได้ข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต ที่สามารถนำไปพัฒนาในกระบวนการออกแบบ กระบวนการพัฒนา และกระบวนการทดสอบ ต่อไปเพื่อให้ได้เฟรมเวิร์กที่สมบูรณ์

จากกระบวนการในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต ดังที่ได้กล่าวมาแล้วข้างต้น สามารถนำเสนอในรูปของแผนภาพได้ ดังรูปที่ 3.15



รูปที่ 3.15 แสดงกระบวนการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต

จากรูปที่ 3.15 เห็นได้ว่าขั้นตอนในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตนั้น แบ่งเป็น 4 ขั้นตอน คือ การแบ่งข้อกำหนดเป็นขั้นตอนในการแบ่งข้อกำหนดของโปรแกรม ประยุกต์ที่ต้องการนำมาพิจารณาให้มีขนาดเล็ก เพื่อนำไปสร้างเป็นข้อกำหนดเฟรมเลต การปรับแต่งข้อกำหนดย่อยเป็นขั้นตอนในการปรับแต่งข้อกำหนดของแต่ละโปรแกรมประยุกต์ที่ได้ จากขั้นตอนการแบ่งข้อกำหนดให้มีความสอดคล้องกัน เพื่อให้สามารถนำข้อกำหนดไปพัฒนาใน ขั้นตอนถัดไปได้สะดวกยิ่งขึ้น การสร้างข้อกำหนดเฟรมเลตเป็นขั้นตอนในการสร้างข้อกำหนด เฟรมเลตจากข้อกำหนดย่อยของแต่ละโปรแกรมประยุกต์ และสุดท้ายการสร้างข้อกำหนดเฟรม เวอร์กด้วยเฟรมเลตเป็นขั้นตอนในการสร้างข้อกำหนดเฟรมเวอร์กโดยใช้ข้อกำหนดของเฟรมเลต ประกอบเข้าด้วยกัน

ประโยชน์ที่ได้จากการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตนั้น คือ ผู้นำข้อกำหนด เฟรมเวอร์กไปใช้สามารถทำความเข้าใจข้อกำหนดเฟรมเวอร์กได้ง่ายขึ้น เนื่องจากโดยปกติ ข้อกำหนดเฟรมเวอร์กจะมีขนาดใหญ่ทำให้สามารถทำความเข้าใจได้ยาก แต่ข้อกำหนดเฟรมเวอร์ก ด้วยเฟรมเลตนั้นเป็นข้อกำหนดเฟรมเวอร์กที่ประกอบด้วยเฟรมเลตภายใน ซึ่งผู้ใช้งานข้อกำหนด เฟรมเวอร์กสามารถทำความเข้าใจในข้อกำหนดเฟรมเลตที่มีขนาดเล็กได้ง่ายกว่าทำความเข้าใจใน ข้อกำหนดเฟรมเวอร์กที่มีขนาดใหญ่ และประโยชน์ที่สำคัญของการสร้างข้อกำหนดเฟรมเวอร์ก ด้วยเฟรมเลตอีกประการหนึ่ง คือ สามารถรองรับการเปลี่ยนแปลงของข้อกำหนดได้ในบางส่วน เนื่องจากข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตนั้นประกอบด้วยเฟรมเลตต่างๆ เมื่อข้อกำหนดมีการ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนแปลงมากจนเกินความสามารถของฮอตสปอตต่างๆ ภายในเฟรมเลต ก็สามารถสร้างข้อกำหนดเฟรมเลตใหม่เข้ามาทดแทนเฟรมเลตเดิมได้ ซึ่งถ้าเป็นการสร้างข้อกำหนดเฟรมเวอร์กแบบเดิมนั้นอาจจำเป็นจะต้องสร้างข้อกำหนดเฟรมเวอร์กใหม่ทั้งระบบ

ในบทต่อไปจะเสนอการประเมินผลการวิจัย โดยแบ่งการประเมินออกเป็น 3 ส่วน คือ ส่วนแรกเสนอการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต โดยใช้ระเบียบวิธีที่ได้จากงานวิจัยนี้ ส่วนที่สองเสนอการนำข้อกำหนดเฟรมเวอร์กที่ได้จากส่วนแรกไปพัฒนาอีกเป็นข้อกำหนดโปรแกรมประยุกต์ และส่วนสุดท้ายเสนอการเปรียบเทียบกระบวนการในการสร้างข้อกำหนดเฟรมเวอร์กจากระเบียบวิธีในงานวิจัยนี้ กับกระบวนการสร้างข้อกำหนดเฟรมเวอร์กแบบเดิม



บทที่ 4

การประเมินผล

ในบทนี้จะเป็นการนำเสนอการประเมินผลระเบียบวิธีการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลต ทั้งนี้เพื่อนำเสนอความสามารถในการพัฒนาข้อกำหนดด้วยระเบียบวิธีการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลตที่ได้นำเสนอไว้ในงานวิจัยนี้ โดยจัดแบ่งโครงสร้างการนำเสนอออกเป็นสามส่วน ส่วนแรกจะกล่าวถึงวิธีการ และขั้นตอนการออกแบบการประเมินผลระเบียบวิธีที่นำเสนอ ส่วนที่สองเป็นการนำเสนอกระบวนการในการประเมินผลตามวิธีการและขั้นตอนที่ได้นำเสนอไว้แล้วในส่วนแรก สำหรับส่วนสุดท้ายเป็นการสรุปผลการประเมินที่ได้

4.1 วิธีการประเมินผล

สำหรับในหัวข้อนี้จะได้นำเสนอขั้นตอน และวิธีการในการประเมินผลความสามารถในการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลต ที่ได้จากระเบียบวิธีสำหรับการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลตที่นำเสนอ โดยแบ่งส่วนการประเมินผลออกเป็น 3 ส่วน ดังนี้

4.1.1 นำระเบียบวิธีที่ได้นำเสนอนี้มาใช้เป็นแนวทางในการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลต เพื่อแสดงถึงความสามารถของระเบียบวิธีที่ได้นำเสนอ ว่าสามารถนำไปใช้สร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลตได้ กรณีศึกษาที่นำมาใช้จะเป็นกลุ่มของโปรแกรมประยุกต์ในโดเมนอีเลิร์นนิ่งจำนวน 3 โปรแกรมประยุกต์ คือ โปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และโปรแกรมประยุกต์เอ็ดิตเตอร์ ซึ่งรายละเอียดได้นำเสนอไว้แล้วในบทที่ 2

4.1.2 ในขั้นตอนถัดมาเป็นการนำเอาข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์คที่ได้ในขั้นตอนแรก มาทดสอบสร้างย้อนกลับเป็นข้อกำหนดของโปรแกรมประยุกต์โดยแบ่งโปรแกรมประยุกต์ออกเป็น 2 ประเภท คือ ประเภทแรกโปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ค คือ โปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และโปรแกรมประยุกต์เอ็ดิตเตอร์ ซึ่งเป็นโปรแกรมประยุกต์ที่ใช้ในการพิจารณาในการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลตจากงานวิจัยนี้ และประเภทที่สองโปรแกรมประยุกต์ภายใต้โดเมนอีเลิร์นนิ่งนอกเหนือจากโปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ค โดยในที่นี้จะทดสอบสร้างข้อกำหนดโปรแกรมประยุกต์เลิร์นแอสควร์

4.1.3 เปรียบเทียบกระบวนการในการสร้างข้อกำหนดเฟรมเวอร์คจากระเบียบวิธีในงานวิจัยนี้ กับกระบวนการสร้างข้อกำหนดเฟรมเวอร์คแบบเดิม โดยปัญหาที่เกิดขึ้นจากการพัฒนาข้อกำหนดเฟรมเวอร์คแบบเดิมมีหลากหลายประเภท โดยปัญหาที่การสร้างข้อกำหนดเฟรมเวอร์คเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยเฟรมเลตสามารถช่วยแก้ไขปัญหาของการสร้างข้อกำหนดเฟรมเวอร์คเดิมประกอบด้วย ปัญหาในการใช้งานเฟรมเวอร์ค และปัญหาในการบำรุงรักษาเฟรมเวอร์ค โดยปัญหาในการใช้งานเฟรมเวอร์ค คือ ปัญหาความเข้าใจในโครงสร้างของเฟรมเวอร์คในโดเมนที่มีขนาดใหญ่ เนื่องจากเฟรมเวอร์คที่มีขนาดใหญ่จะประกอบด้วยสอตสปอต และรายละเอียดภายในจำนวนมาก ทำให้สามารถทำความเข้าใจเฟรมเวอร์คที่มีขนาดใหญ่ได้ยาก และปัญหาในการบำรุงรักษาเฟรมเวอร์ค คือ ปัญหาในการเปลี่ยนแปลงของข้อกำหนดภายในโดเมน โดยในการสร้างเฟรมเวอร์คมีความเป็นไปได้ที่จะไม่รองรับการทำงานทั้งหมดที่ผู้ใช้งานเฟรมเวอร์คต้องการ โดยในการเปลี่ยนแปลงหรือปรับปรุงเฟรมเวอร์คที่มีขนาดใหญ่สามารถทำได้ยากกว่าการเปลี่ยน หรือปรับปรุงเฟรมเลตที่มีขนาดเล็ก

4.2 การประเมินผล

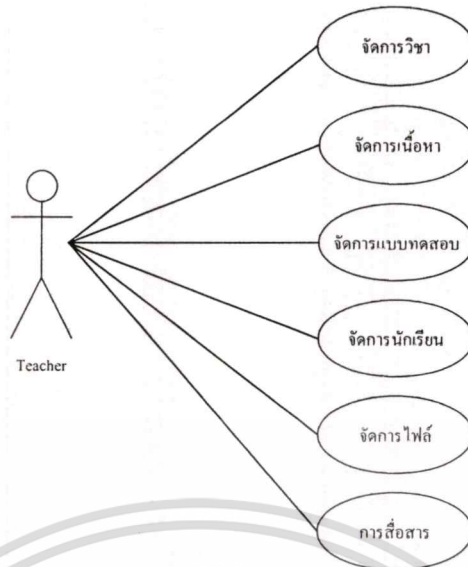
ในส่วนนี้จะอธิบายขั้นตอนการประเมินผลทั้ง 3 ส่วนที่ได้กล่าวไว้แล้วข้างต้น โดยส่วนแรกจะทำการสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลตจากระเบียบวิธีที่ได้ในบทที่ 3 ส่วนที่สองจะทำการทดสอบสร้างข้อกำหนดโปรแกรมประยุกต์จากข้อกำหนดเฟรมเวอร์คที่ได้ในส่วนแรก และส่วนสุดท้ายเสนอการเปรียบเทียบกระบวนการสร้างข้อกำหนดเฟรมเวอร์คที่ได้จากระเบียบวิธีนี้กับกระบวนการสร้างข้อกำหนดเฟรมเวอร์คแบบเดิม โดยมีรายละเอียดการประเมินผลทั้งสามส่วนดังนี้

4.2.1 การสร้างข้อกำหนดเฟรมเวอร์คด้วยเฟรมเลต

สร้างข้อกำหนดอิเล็กทรอนิกส์เฟรมเวอร์คจากระเบียบวิธีที่ได้ ซึ่งพิจารณาจากข้อกำหนดโปรแกรมประยุกต์ ในโดเมนอิเล็กทรอนิกส์ 3 โปรแกรมประยุกต์ คือ โปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และ โปรแกรมประยุกต์เอ็ดิตเตอร์ ซึ่งรายละเอียดได้นำเสนอไว้แล้วในบทที่ 2 โดยใช้ระเบียบวิธีที่ได้แสดงรายละเอียดไว้ในบทที่ 3

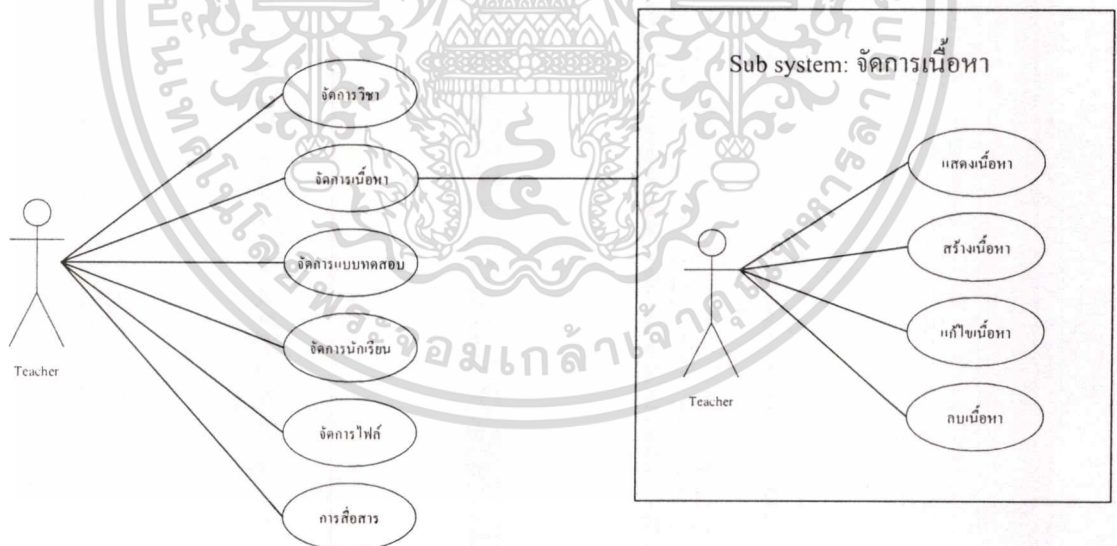
4.2.1.1 การแบ่งข้อกำหนด

แบ่งข้อกำหนดโปรแกรมประยุกต์ทั้ง 3 ที่นำมาพิจารณา ซึ่งในที่นี้จะเริ่มจากการแบ่งข้อกำหนดโปรแกรมประยุกต์เห็นก่อน ดังรูปที่ 4.1



รูปที่ 4.1 แสดงการแบ่งข้อกำหนดโปรแกรมประยุกต์เห็นในระดับแรก

จากรูปที่ 4.1 โปรแกรมประยุกต์เห็นสามารถแบ่งฟังก์ชันการทำงานได้เป็น 6 ส่วน คือ จัดการวิชา จัดการเนื้อหา จัดการแบบทดสอบ จัดการนักเรียน จัดการไฟล์ และการสื่อสาร จากนั้นจึงทำการแบ่งย่อยภายในแต่ละส่วนต่อไป ดังรูปที่ 4.2



รูปที่ 4.2 แสดงการแบ่งข้อกำหนดของเห็นในส่วนจัดการเนื้อหา

จากรูปที่ 4.2 โปรแกรมประยุกต์เห็นในส่วนจัดการวิชา สามารถแบ่งฟังก์ชันการทำงานได้เป็น 4 ส่วน คือ แสดงเนื้อหา สร้างเนื้อหา แก้ไขเนื้อหา และลบเนื้อหา เมื่อได้ส่วนย่อยดังกล่าว จากนั้นจึงทำการแบ่งย่อยภายในแต่ละส่วนต่อไป แต่เนื่องจากถ้าแบ่งย่อยทั้ง 4 ส่วน ดังกล่าวลงไปอีกนั้น จะได้ฟังก์ชันในการค้นหาแต่เนื่องจากการค้นหานั้นเป็นฟังก์ชันการทำงานทั่วไปไม่เป็นเอกสาร ฟังก์ชันการทำงานที่ขึ้นกับโดเมนอิเล็กทรอนิกส์ จึงไม่แบ่งฟังก์ชันการค้นหาออกมาจากส่วนดังกล่าวไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากในขั้นตอนการแบ่งข้อกำหนดนี้นั้นจะไม่แบ่งจนข้อกำหนดย่อยไม่ขึ้นกับโดเมนที่ต้องการพัฒนา เมื่อแบ่งฟังก์ชันการทำงานย่อยต่อไปไม่ได้แล้วจึงทำการแบ่งในส่วนจัดการวิชา จัดการแบบทดสอบ จัดการนักเรียน จัดการไฟล์ และการสื่อสาร ต่อไปโดยใช้หลักการเดียวกัน ซึ่งรายละเอียดได้แสดงไว้ในภาคผนวก ก.

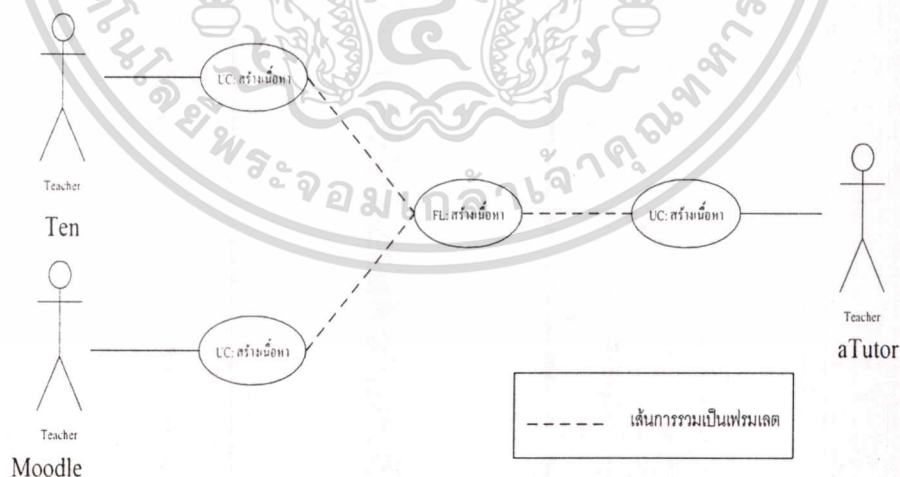
เมื่อแบ่งข้อกำหนดโปรแกรมประยุกต์เห็นแล้ว จึงทำการแบ่งข้อกำหนดโปรแกรมประยุกต์มูเดิล และ โปรแกรมประยุกต์เอ็ดวเเตอร์ โดยใช้หลักการเดียวกัน ซึ่งรายละเอียดได้แสดงไว้ในภาคผนวก ก.

4.2.1.2 การปรับแต่งข้อกำหนดย่อย

จัดรูปแบบข้อกำหนดย่อยของแต่ละโปรแกรมประยุกต์ที่ได้จากขั้นตอนการแบ่งข้อกำหนด ให้ข้อกำหนดของโปรแกรมประยุกต์มีความสอดคล้องกัน ดังที่ได้กล่าวไว้แล้วในบทที่ 3 แต่เนื่องจากในงานวิจัยนี้ในขั้นตอนการแบ่งข้อกำหนดย่อยของแต่ละโปรแกรมประยุกต์นั้นได้แบ่งโดยผู้วิจัยเพียงคนเดียว การแบ่งจึงสอดคล้องกันอยู่แล้ว จึงไม่จำเป็นต้องทำการปรับแต่งข้อกำหนดย่อย

4.2.1.3 การสร้างข้อกำหนดเฟรมเลต

สร้างข้อกำหนดเฟรมเลตโดยพิจารณาจากข้อกำหนดย่อยของแต่ละโปรแกรมประยุกต์ในส่วนฟังก์ชันการทำงานเดียวกัน มาสร้างข้อกำหนดเฟรมเลต ดังรูปที่ 4.3 แสดงการสร้างข้อกำหนดเฟรมเลตสร้างเนื้อหา



รูปที่ 4.3 แสดงการสร้างข้อกำหนดเฟรมเลตสร้างเนื้อหา

จากรูปที่ 4.3 เป็นการสร้างข้อกำหนดเฟรมเลตสร้างเนื้อหา โดยสร้างจากข้อกำหนดสร้างเนื้อหาของแต่ละโปรแกรมประยุกต์ที่ได้จากขั้นตอนการแบ่งข้อกำหนด โดยในการสร้างข้อกำหนดเฟรมเลตนั้นจะวิเคราะห์จากสายลำดับการทำงานของแต่ละโปรแกรมประยุกต์ เพื่อนำมาสร้างเป็นเอกสารเป็นเอกสารที่สวอนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายลำดับการทำงานของข้อกำหนดเฟรมเลต ดังรูปที่ 4.4-4.6 แสดงข้อกำหนดย่อยสร้างเนื้อหาของแต่ละโปรแกรมประยุกต์ที่นำมาพิจารณา

Use Case: สร้างเนื้อหา

Primary Actor: ผู้สอน

Stakeholders and Interests:

— **Teacher:** ผู้สอนสามารถสร้างเนื้อหาที่ต้องการให้นักเรียนเข้ามาเรียนได้

Preconditions: จะต้องมีการสร้างวิชาไว้แล้ว และอยู่ในหน้าหลักวิชา

Postconditions: สร้างเนื้อหาได้

Main Success Scenario (or Basic Flow):

1. เลือกหมวด(มี 3 หมวด ได้แก่ หมวด บท บทย่อย) และประเภทของบทเรียนที่ต้องการสร้าง
2. ใส่รายละเอียดของเนื้อหา(บท ชื่อเรื่อง/ชื่อบท ชื่อเรื่อง/ชื่อบท(แบบย่อ) คำสำคัญเพื่อใช้ในการค้นหา สถานะ)
3. เลือกตกลง

Extensions (or Alternative Flows):

1. เลือกประเภทของบทเรียน
 - 1.1 ประเภทของบทเรียน
 - 1.2 ทดสอบก่อนเรียน
 - 1.3 แบบฝึกหัด
 - 1.4 ข้อสอบ
 - 1.5 งานที่ได้รับมอบหมาย/การบ้าน
 - 1.6 ประเมินผล
3. กรอกเนื้อหาไม่ครบถ้วนตามที่กำหนด
 - 3.1 แสดงข้อความการผิดพลาด

รูปที่ 4.4 แสดงข้อกำหนดสร้างเนื้อหาของโปรแกรมประยุกต์เห็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Use Case: สร้างเนื้อหา

Primary Actor: ผู้สอน

Stakeholders and Interests:

— **Teacher:** ผู้สอนสามารถสร้างเนื้อหาที่ต้องการได้

Preconditions: จะต้องมีการสร้างวิชาไว้แล้ว และอยู่ในหน้าหลักวิชา

Postconditions: สร้างเนื้อหาได้สำเร็จ

Main Success Scenario (or Basic Flow):

1. เลือกประเภทของเนื้อหาที่ต้องการสร้าง
2. ใส่รายละเอียดต่างๆของเนื้อหา (โดยรายละเอียดที่ใส่นั้นขึ้นอยู่กับ ประเภทของเนื้อหาที่เราต้องการสร้างขึ้นด้วย)
3. ยืนยันการสร้างวิชา

Extensions (or Alternative Flows):

1. เลือกประเภทของเนื้อหา
 - 1.1 Chat
 - 1.2 กระดานเสวนา
 - 1.3 การบ้าน
 - 1.4 ตัวเลือก
 - 1.5 วารสาร
 - 1.6 สัมมนา
 - 1.7 แบบทดสอบ
 - 1.8 แบบสำรวจ
 - 1.9 แหล่งข้อมูล
3. กรอกเนื้อหาไม่ครบถ้วนตามที่กำหนด
 - 3.1 แสดงข้อความการผิดพลาด

รูปที่ 4.5 แสดงข้อกำหนดสร้างเนื้อหาของ โปรแกรมประยุกต์มูเดิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Use Case: สร้างเนื้อหา

Primary Actor: ผู้สอน

Stakeholders and Interests:

— **Teacher:** ผู้สอนสามารถสร้างเนื้อหาที่ต้องการได้

Preconditions: จะต้องมีการสร้างวิชาไว้แล้ว และอยู่ในหน้าหลักวิชา

Postconditions: สร้างเนื้อหาได้สำเร็จ

Main Success Scenario (or Basic Flow):

1. เลือกสร้างเนื้อหา
2. ใส่เนื้อหาที่ต้องการสร้าง(ชื่อเรื่อง รูปแบบเนื้อหา ข้อความ ไฟล์ที่ต้องการ)
3. เลือกบันทึก

Extensions (or Alternative Flows):

2. กรอกเนื้อหาไม่ครบถ้วนตามที่กำหนด
 - 2.1 แสดงข้อความการผิดพลาด

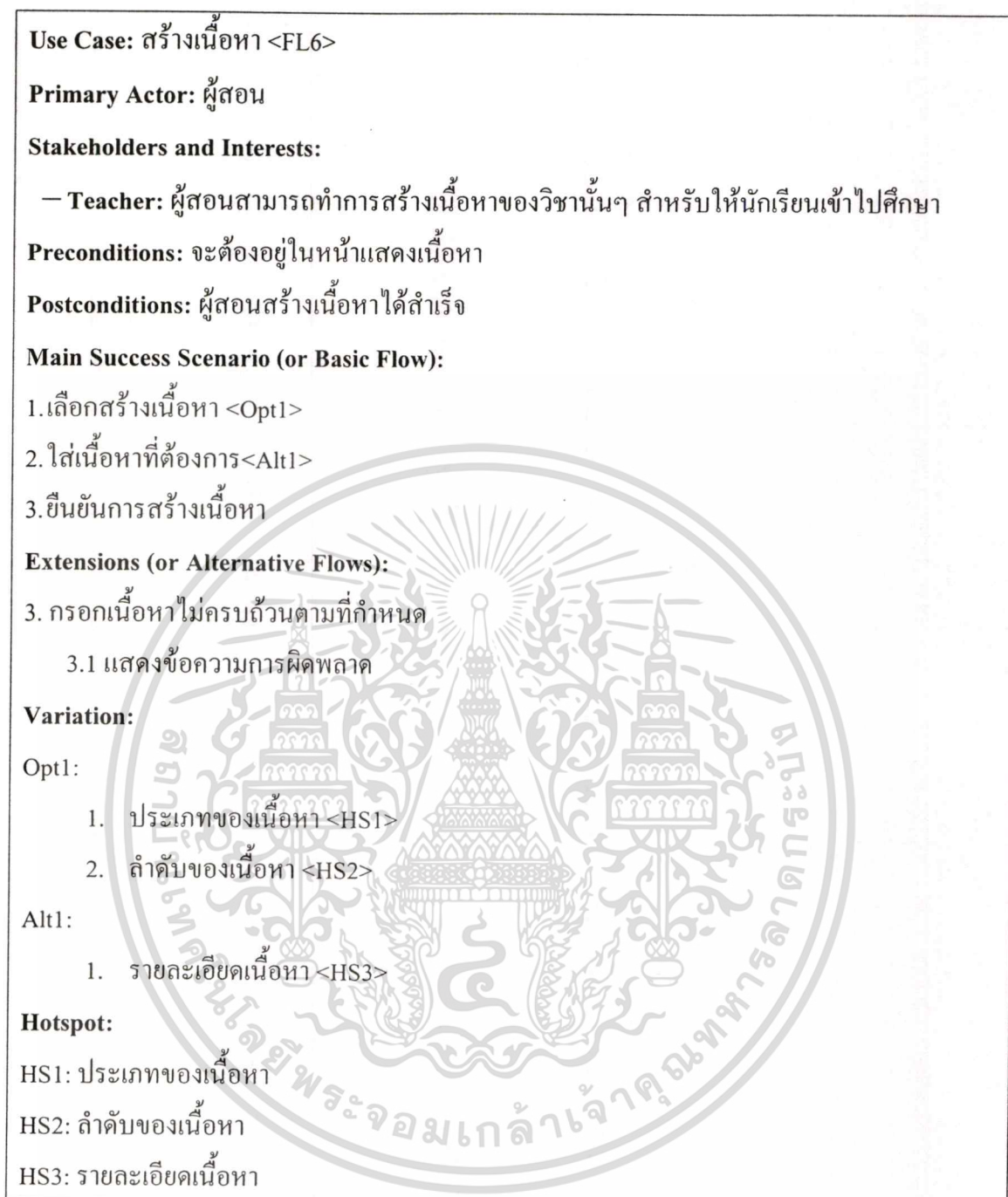
รูปที่ 4.6 แสดงข้อกำหนดสร้างเนื้อหาของ โปรแกรมประยุกต์เอ็ดวเเตอร์

จากรูปที่ 4.4-4.6 จากการวิเคราะห์ข้อกำหนดของ โปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูล และ โปรแกรมประยุกต์เอ็ดวเเตอร์ โดยจะเห็นได้ว่าสายลำดับการทำงานของทั้ง 3 โปรแกรมประยุกต์ มีความคล้ายคลึงกัน ซึ่งในขั้นตอนแรกภายในสายลำดับการทำงานของทั้ง 3 โปรแกรมประยุกต์ จะเป็นการเริ่มสร้างเนื้อหา โดยในขั้นตอนแรกจะมีส่วนที่แตกต่างกัน คือ โปรแกรมประยุกต์เห็นมีการแบ่งลำดับของเนื้อหา และประเภทของเนื้อหา ส่วน โปรแกรมประยุกต์มูลมีแบ่งประเภทของเนื้อหาเพียงอย่างเดียว ส่วน โปรแกรมประยุกต์เอ็ดวเเตอร์ไม่มีการแบ่งทั้ง 2 ส่วน สายลำดับการทำงานภายในข้อกำหนดเฟรมเลตสร้างเนื้อหาจึงกำหนดให้จุดดังกล่าวเป็นจุดแปรผันตัวเลือก Opt1 ที่ใช้เชื่อมต่อกับฮอตสปอตประเภทของเนื้อหา และฮอตสปอตลำดับของเนื้อหา จุดแปรผันดังกล่าวเป็นจุดแปรผันตัวเลือกเนื่องจากมีได้มากกว่า 1 ฮอตสปอตในจุดแปรผันนั้นๆ ในขั้นตอนที่สองภายในสายลำดับการทำงานของทั้ง 3 โปรแกรมประยุกต์เป็นขั้นตอนในการใส่รายละเอียดของเนื้อหา ซึ่งรายละเอียดของเนื้อหาในแต่ละ โปรแกรมประยุกต์ก็มีความแตกต่างกันไป ภายในข้อกำหนดเฟรมเลตสร้างเนื้อหาจึงกำหนดให้จุดดังกล่าวเป็นจุดแปรผันทางเลือก Alt1 ที่ใช้เชื่อมต่อกับฮอตสปอตรายละเอียดเนื้อหา จุดแปรผันดังกล่าวเป็นจุดแปรผันทางเลือกเนื่องจากมีได้เพียง 1 ฮอตสปอตในจุดแปรผันนั้นๆ และในขั้นตอนสุดท้ายภายในสายลำดับการทำงานของทั้ง 3 โปรแกรมประยุกต์เป็นขั้นตอนในการยืนยันการสร้างเนื้อหาเหมือนกันทุกโปรแกรมประยุกต์ภายในข้อกำหนดเฟรมเลตสร้างเนื้อหาจึงไม่มีจุดแปรผันในส่วนนี้ จึงสามารถนำข้อกำหนดดังกล่าว

มาวิเคราะห์สร้างเป็นข้อกำหนดเฟรมเลตสร้างเนื้อหาได้ ดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ การใช้งานเพื่อวัตถุประสงค์อื่นโดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงข้อกำหนดเฟรมเลตสร้างเนื้อหา

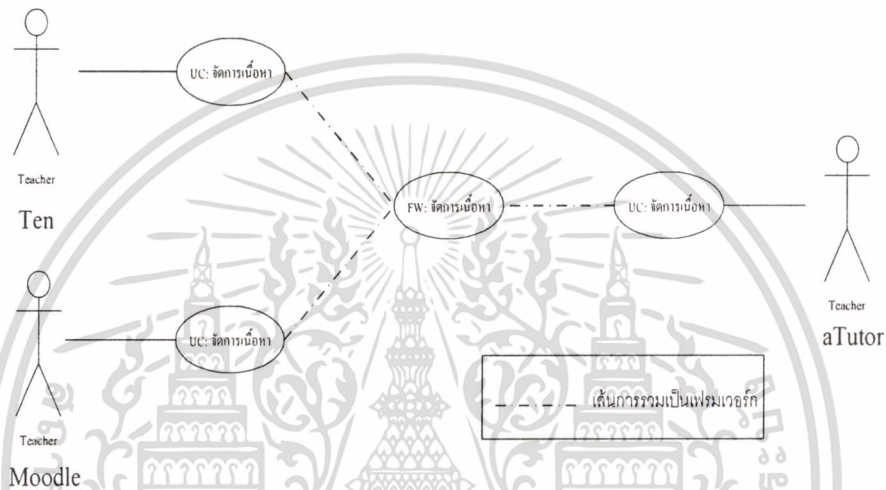
จากรูปที่ 4.7 แสดงข้อกำหนดเฟรมเลตสร้างเนื้อหาที่ได้ ซึ่งจะเห็นได้ว่าภายในสายลำดับการทำงานประกอบไปด้วยจุดแปรผันต่างๆ เพื่อรองรับการทำงานที่หลากหลายของแต่ละโปรแกรมประยุกต์

โดยเมื่อสร้างข้อกำหนดเฟรมเลตสร้างเนื้อหาแล้ว จึงทำการสร้างข้อกำหนดเฟรมเลตในส่วนอื่นๆ โดยใช้หลักการเดียวกัน ซึ่งรายละเอียดได้แสดงไว้ในภาคผนวก ก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.4 การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต

เมื่อได้ข้อกำหนดเฟรมเลตทั้งหมดจากขั้นตอนการสร้างข้อกำหนดเฟรมเลตแล้วให้นำข้อกำหนดเฟรมเลตดังกล่าวมาสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลต โดยทำการสร้างข้อกำหนดเฟรมเวิร์กย้อนกลับขึ้นมาทีละระดับ ซึ่งเฟรมเวิร์กระดับต่ำสุดที่จะสร้างประกอบด้วย 6 ส่วน คือ จัดการวิชา จัดการเนื้อหา จัดการแบบทดสอบ จัดการนักเรียน จัดการไฟล์ และการสื่อสาร ดังรูปที่ 4.8 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหา



รูปที่ 4.8 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหา

จากรูปที่ 4.8 แสดงการสร้างข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหาจากข้อกำหนดของโปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และโปรแกรมประยุกต์เอ็ดวเเตอร์ ซึ่งในการสร้างข้อกำหนดดังกล่าว ให้พิจารณาจากสายลำดับการทำงานร่วมของแต่ละโปรแกรมประยุกต์มาสร้างเป็นสายลำดับการทำงานของเฟรมเวิร์กย่อย ดังรูปที่ 4.9-4.11

Use Case: จัดการเนื้อหา

Primary Actor: Abstract Actor

Stakeholders and Interests:

- **Abstract Actor:** ผู้สอนต้องการจัดการเกี่ยวกับเนื้อหาที่ต้องการได้ อาทิ แสดงเนื้อหา สร้างเนื้อหา ลบเนื้อหา เป็นต้น

Preconditions: ผู้สอนต้องทำการลงทะเบียน และล็อกอินในฐานะผู้สอน

Postconditions: ผู้สอนจัดการเนื้อหาต่างๆ ภายในระบบได้สำเร็จ

Main Success Scenario (or Basic Flow):

1. เข้าสู่หน้าหลักจัดการเนื้อหา และแสดงรายชื่อเนื้อหา
2. เลือกฟังก์ชันการทำงานต่างๆ
 - 2.1 แสดงเนื้อหา
 - 2.2 สร้างเนื้อหา
 - 2.3 แก้ไขเนื้อหา
 - 2.5 ลบเนื้อหา

รูปที่ 4.11 แสดงข้อกำหนด โปรแกรมประยุกต์เอ็ดวเเตอร์ในส่วนจัดการเนื้อหา

จากรูปที่ 4.9-4.11 ที่ได้จากการวิเคราะห์สายลำดับการทำงานในส่วนจัดการเนื้อหาของโปรแกรมประยุกต์ที่นำมาพิจารณาที่มีความคล้ายคลึงกันมาสร้างเป็นสายลำดับการทำงานของข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหา ซึ่งจากการวิเคราะห์สายลำดับการทำงานของข้อกำหนดโปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และโปรแกรมประยุกต์เอ็ดวเเตอร์ ในขั้นตอนแรกภายในสายลำดับการทำงานของทั้ง 3 โปรแกรมประยุกต์เป็นส่วนเข้าสู่หน้าหลักจัดการเนื้อหาซึ่งมีส่วนการแสดงรายชื่อเนื้อหาทุกโปรแกรมประยุกต์ที่นำมาพิจารณา โดยเมื่อทำการค้นหาเฟรมเลตที่ได้สร้างไว้ในขั้นตอนก่อนหน้าจะพบว่าได้ทำการสร้างเฟรมเลตแสดงรายชื่อเนื้อหาไว้แล้ว จึงกำหนดให้จุดดังกล่าวเป็นจุดแปรผันที่เชื่อมต่อกับเฟรมเลตแสดงรายชื่อเนื้อหา โดยกำหนดให้จุดดังกล่าวเป็นจุดแปรผันทางเลือก เนื่องจากสามารถมีการทำงานในส่วนดังกล่าวได้เพียง 1 ฟังก์ชันการทำงาน ในขั้นตอนที่สองของสายลำดับการทำงานของโปรแกรมประยุกต์ทั้ง 3 จะเห็นได้ว่ามีส่วนที่สามารถเลือกฟังก์ชันการทำงานได้ทั้งหมด 4 ส่วนเหมือนกัน คือ แสดงเนื้อหา สร้างเนื้อหา แก้ไขเนื้อหา และลบเนื้อหา โดยเมื่อทำการค้นหาเฟรมเลตที่ได้สร้างไว้ในขั้นตอนก่อนหน้าจะพบว่าได้ทำการสร้างเฟรมเลตทั้ง 4 ส่วนไว้แล้ว จึงกำหนดให้จุดดังกล่าวเป็นจุดแปรผันที่เชื่อมต่อกับเฟรมเลตทั้ง 4 โดยกำหนดให้จุดแปรผันดังกล่าวเป็นจุดแปรผันตัวเลือกเพราะสามารถเลือกได้มากกว่า 1 เฟรมเลต จึงสามารถนำข้อกำหนดดังกล่าวมาสร้างเป็นข้อกำหนดเฟรมเวิร์กย่อยจัดการเนื้อหาได้ ดังรูปที่ 4.12

เอกสารนี้เป็นเอกสารต้นฉบับที่จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Use Case: จัดการเนื้อหา <FW2>

Primary Actor: Abstract Actor

Stakeholders and Interests:

- **Abstract Actor:** ผู้สอนต้องการจัดการเกี่ยวกับเนื้อหาที่ต้องการได้ อาทิ แสดงเนื้อหา สร้างเนื้อหา ลบเนื้อหา เป็นต้น

Preconditions: ผู้สอนต้องทำการลงทะเบียน และล็อกอินในฐานะผู้สอน

Postconditions: ผู้สอนจัดการเนื้อหาต่างๆ ภายในระบบได้สำเร็จ

Main Success Scenario (or Basic Flow):

1. เข้าสู่หน้าหลักจัดการเนื้อหา <Alt1>
2. เลือกฟังก์ชันการทำงานต่างๆ <Opt1>

Extensions (or Alternative Flows):

Variation:

Alt1:

1. แสดงรายชื่อเนื้อหา <FL5>

Opt1:

1. เลือกแสดงเนื้อหา <FL6>
2. เลือกสร้างเนื้อหา <FL7>
3. เลือกแก้ไขเนื้อหา <FL8>
4. เลือกลบเนื้อหา <FL9>

Frameworks:

Framelets:

FL5: แสดงรายชื่อเนื้อหา

FL6: แสดงเนื้อหา

FL7: สร้างเนื้อหา

FL8: แก้ไขเนื้อหา

FL9: ลบเนื้อหา

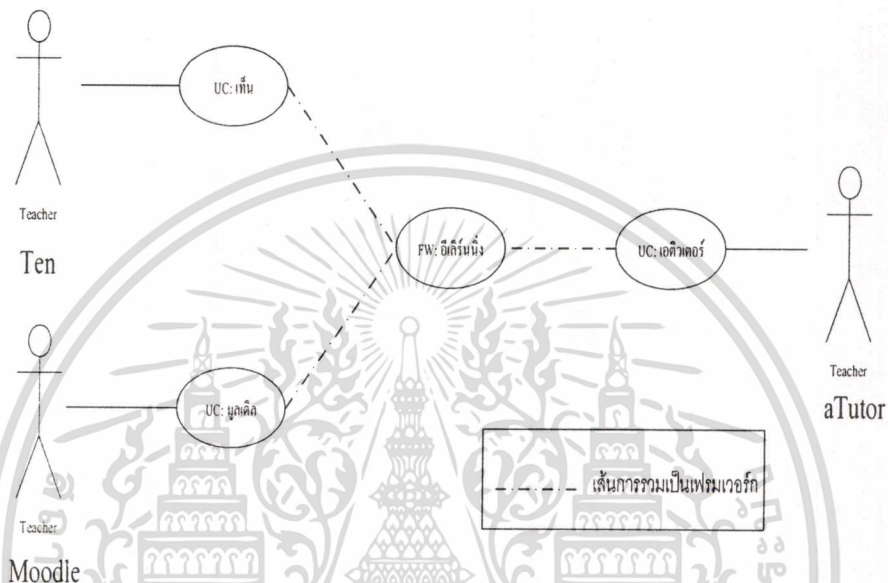
Hotspot:

รูปที่ 4.12 แสดงข้อกำหนดเฟรมเวอร์กย่อยจัดการเนื้อหา

จากรูปที่ 4.12 แสดงข้อกำหนดเฟรมเวอร์กย่อยจัดการเนื้อหา ซึ่งจะเห็นได้ว่าภายในสายลำดับการทำงานประกอบไปด้วยจุดแปรผันต่างๆ เพื่อรองรับการทำงานที่หลากหลายของแต่ละเอกสารโปรแกรมประยุกต์ได้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเมื่อสร้างข้อกำหนดเฟรมเวิร์กจัดการเนื้อหาแล้ว จึงทำการสร้างข้อกำหนดเฟรมเวิร์กย่อยส่วนอื่นๆ ในระดับเดียวกันโดยใช้หลักการเดียวกัน ซึ่งรายละเอียดได้แสดงไว้ในภาคผนวก ก.

เมื่อได้ข้อกำหนดเฟรมเวิร์กย่อยในระดับนี้แล้ว จึงทำการสร้างข้อกำหนดเฟรมเวิร์กย้อนกลับขึ้นไป ซึ่งในที่นี้เป็นระดับที่ใหญ่ที่สุด คือ อีเลิร์นนิ่งเฟรมเวิร์ก ดังรูปที่ 4.13



รูปที่ 4.13 แสดงการสร้างข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์ก

จากรูปที่ 4.13 แสดงการสร้างข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์กจากข้อกำหนดของโปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มอดูล และ โปรแกรมประยุกต์เอ็ดวอเตอร์ ซึ่งในการสร้างข้อกำหนดดังกล่าว ให้พิจารณาจากสายลำดับการทำงานร่วมของแต่ละโปรแกรมประยุกต์มาสร้างเป็นสายลำดับการทำงานของเฟรมเวิร์กย่อย เช่นเดียวกับการสร้างข้อกำหนดเฟรมเวิร์กย่อยข้างต้น ดังรูปที่ 4.14 แสดงข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์กทั้งระบบ

Use Case: อีเลิร์นนิ่ง

Primary Actor: Abstract Actor

Stakeholders and Interests:

— **Abstract Actor:** ผู้สอนสามารถใช้ระบบต่างๆ ภายในระบบอีเลิร์นนิ่งได้

Preconditions: ทำการลงทะเบียนภายในระบบ

Postconditions:

รูปที่ 4.14 แสดงข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์กทั้งระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Main Success Scenario (or Basic Flow):

1. ล็อกอินเข้ามาภายในระบบ
2. แสดงหน้าหลักผู้ล็อกอิน <Alt1> <Opt1>
3. เลือกวิชาที่ต้องการ
4. แสดงหน้าหลักวิชา <Alt2>
5. เลือกฟังก์ชันการทำงานต่างๆ <Opt2>
6. เลือกจัดการทั่วไป <Alt3>
7. เลือกฟังก์ชันการทำงานต่างๆ ภายในส่วนจัดการทั่วไป <Opt3>

Extensions (or Alternative Flows):**Variation:**

Alt1:

1. แสดงรายชื่อวิชา <FL1>

Opt1:

1. สร้างวิชา <HS1>

Alt2:

1. แสดงรายละเอียดวิชา <FL2>
2. แสดงรายชื่อเนื้อหา <FL4>

Opt2:

1. ประกาศ <FL24>
2. สนทนาออนไลน์ <HS4>
3. กระดานข่าว <FL23>
4. แสดงแบบทดสอบ <FL9>
5. แสดงรายละเอียดวิชา <FL2>
6. แก้ไขรายละเอียดวิชา <FL3>
7. จัดการนักเรียน <FW4>
8. จัดการเนื้อหา <FW2>
9. จัดการแบบทดสอบ <FW3>

Alt3:

1. แสดงเนื้อหา <FL5>

รูปที่ 4.14 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Opt3:

1. จัดการไฟล์ <FW5>
2. เมลล์ <HS3>
3. ประกาศ <FL24>
4. กระดานข่าว <FL23>
5. สันทนาออนไลน์ <HS4>
6. จัดการเนื้อหา <FW2>
7. จัดการแบบทดสอบ <FW3>
8. แสดงรายละเอียดวิชา <FL2>
9. แก้ไขรายละเอียดวิชา <FL3>
10. จัดการนักเรียน <FW4>
11. จัดการวิชา <FW1>
12. แสดงเนื้อหา <FL5>
13. ลบวิชา <HS2>

Frameworks:

FW1: จัดการวิชา

FW2: จัดการเนื้อหา

FW3: จัดการแบบทดสอบ

FW4: จัดการนักเรียน

FW5: จัดการไฟล์

Framelets:

FL1: แสดงรายชื่อวิชา

FL2: แสดงรายละเอียดวิชา

FL3: แก้ไขรายละเอียดวิชา

FL4: แสดงรายชื่อเนื้อหา

FL5: แสดงเนื้อหา

FL9: แสดงแบบทดสอบ

FL24: ประกาศ

FL25: กระดานข่าว

รูปที่ 4.14 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hotspot:

HS1: สร้างวิชา

HS2: ลบวิชา

HS3: เมลล์

HS4: สอนทนายออนไลน์

รูปที่ 4.14 (ต่อ)

จากรูปที่ 4.14 แสดงข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์ก ที่ได้จากการวิเคราะห์สายลำดับการทำงานในโดยรวมทั้งระบบของทั้ง 3 โปรแกรมประยุกต์ที่นำมาพิจารณา มาสร้างเป็นสายลำดับการทำงานของข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์ก โดยในการสร้างข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กจะมีขั้นตอนการสร้างเช่นเดียวกับการสร้างข้อกำหนดเฟรมเวอร์กย่อยที่ได้อธิบายไว้แล้วข้างต้น

โดยเมื่อผ่านขั้นตอนนี้แล้วจะได้ข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กที่สามารถนำไปพัฒนาในขั้นตอนการออกแบบเฟรมเวอร์ก ขั้นตอนการพัฒนาเฟรมเวอร์ก และขั้นตอนการทดสอบเฟรมเวอร์กต่อไปได้ โดยรายละเอียดทั้งหมดในการสร้างข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กที่ได้จากงานวิจัยนี้แสดงไว้ในภาคผนวก ก.

4.2.2 การสร้างข้อกำหนดโปรแกรมประยุกต์จากข้อกำหนดเฟรมเวอร์ก

ส่วนนี้จะทดสอบข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กที่ได้จากการทดสอบข้างต้น โดยการสร้างข้อกำหนดโปรแกรมประยุกต์ในโดเมนอีเลิร์นนิ่งจากข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กที่ได้จากระเบียบวิธีในงานวิจัยนี้ โดยโปรแกรมประยุกต์จะแบ่งได้เป็น 2 ประเภท คือ โปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ก เพื่อทดสอบข้อกำหนดเฟรมเวอร์กที่ได้จากระเบียบวิธีจากงานวิจัยนี้ว่าสามารถสร้างกลับเป็นข้อกำหนดโปรแกรมประยุกต์ และข้อกำหนดโปรแกรมประยุกต์ที่ได้สามารถทำงานได้เช่นเดียวกับข้อกำหนดโปรแกรมประยุกต์เดิม และโปรแกรมประยุกต์ภายใต้โดเมนอีเลิร์นนิ่งนอกเหนือจากโปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ก เพื่อทดสอบความสามารถของข้อกำหนดเฟรมเวอร์กที่ได้ว่าสามารถนำไปสร้างข้อกำหนดโปรแกรมประยุกต์อื่นๆ ภายในโดเมนเดียวกันได้ ซึ่งในส่วนแรกจะทำการสร้างข้อกำหนดโปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มุเคิล และ โปรแกรมประยุกต์เอติเวเตอร์ ซึ่งเป็นโปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ก และทำการทดสอบสร้างข้อกำหนดโปรแกรมประยุกต์อีเลิร์นนิ่งแอสควร์ ซึ่งเป็นโปรแกรมประยุกต์ภายใต้โดเมนอีเลิร์นนิ่งนอกเหนือจากโปรแกรมประยุกต์ที่นำมาพิจารณาเป็นข้อกำหนดเฟรมเวอร์ก โดยภายในข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กจะมีสายลำดับการทำงานหลักทั้งหมดไว้แล้ว ผู้ใช้งานเฟรมเวอร์กเพียงแต่พัฒนาซุกเข้าไปประกอบที่จุดแปรผันต่างๆ ภายในข้อกำหนดเฟรมเวอร์ก ข้อกำหนดเฟรมเวอร์ก

เอกสารนี้เป็นเอกสารที่เผยแพร่โดยมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี เพื่อใช้ในการศึกษาวิจัยและพัฒนาเท่านั้น ไม่สามารถนำออกจำหน่าย หรือทำซ้ำโดยไม่ได้รับอนุญาตจากมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อกำหนดโปรแกรมประยุกต์ที่ต้องการได้โดยไม่จำเป็นต้องสร้างข้อกำหนดโปรแกรมประยุกต์ใหม่ทั้งระบบ โดยเมื่อได้ข้อกำหนดโปรแกรมประยุกต์ที่ได้จากข้อกำหนดเฟรมเวิร์กแล้วจะทำการเปรียบเทียบกับข้อกำหนดโปรแกรมประยุกต์เดิมว่ามีขั้นตอนการทำงานที่เหมือนกัน

4.2.2.1 การสร้างข้อกำหนดโปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวิร์ก จากรูปที่ 4.14 การแสดงข้อกำหนดอีเลิร์นนิ่งเฟรมเวิร์กจะเห็นได้ว่าภายในสายลำดับการทำงานจะประกอบด้วยจุดแปรผันทั้งสิ้น 8 จุดแปรผัน ภายในสายลำดับการทำงาน โดยเป็นจุดแปรผันทางเลือก 4 จุด และจุดแปรผันตัวเลือก 2 จุด ซึ่งในการสร้างข้อกำหนดโปรแกรมประยุกต์ที่เห็นนั้นจะทำการพัฒนาสูกเข้าไปประกอบด้วยสอตสล็อตต่างๆ ภายในเฟรมเวิร์ก เฟรมเวิร์กย่อย หรือเฟรมเลต ให้ตรงกับข้อกำหนดโปรแกรมประยุกต์ที่เห็น โดยในที่นี้จะอธิบายในส่วนจุดแปรผันทางเลือก Alt1 จุดแปรผันตัวเลือก Opt1 และจุดแปรผันตัวเลือก Opt2 ซึ่งเป็นจุดแปรผันที่มีลักษณะการใช้งานแตกต่างกัน ดังนี้

4.2.2.1.1 จุดแปรผันทางเลือก Alt1 (ภายในอีเลิร์นนิ่งเฟรมเวิร์ก) เป็นจุดแปรผันที่ใช้เชื่อมต่อกับข้อกำหนดเฟรมเลตแสดงรายชื้อวิชา ผู้ใช้งานเฟรมเวิร์กเพียงแค่พัฒนาสูกเข้ามาประกอบด้วยสอตสล็อตที่ได้เตรียมไว้ในข้อกำหนดเฟรมเลตแสดงรายชื้อวิชา ดังรูปที่ 4.15 แสดงข้อกำหนดเฟรมเลตแสดงรายชื้อวิชา จะเห็นได้ว่าข้อกำหนดเฟรมเลตแสดงรายชื้อวิชาประกอบด้วยจุดแปรผันทั้งหมด 1 จุด คือ จุดแปรผันทางเลือก Alt1 (ภายในเฟรมเลตแสดงรายชื้อวิชา) เป็นจุดแปรผันที่เชื่อมต่อกับสอตสล็อตส่วนต่อประสานผู้ใช้ ผู้ใช้งานเฟรมเวิร์กเพียงแค่ทำการพัฒนาสูกเข้าไปประกอบด้วยสอตสล็อตส่วนต่อประสานผู้ใช้ ดังรูปที่ 4.16 แสดงคำอธิบายสูกในส่วนต่อประสานผู้ใช้ในการแสดงรายชื้อวิชา ให้มีการแสดงเหมือนกับข้อกำหนดโปรแกรมประยุกต์ที่เห็นเดิม โดยในส่วนการเปลี่ยนแปลงจะเป็นสายลำดับการทำงานที่ท่าจะนำไปประกอบกับจุดแปรผัน Alt1 ภายในข้อกำหนดเฟรมเลตแสดงรายชื้อวิชา ซึ่งเมื่อทำการประกอบเสร็จแล้วก็จะได้ข้อกำหนดแสดงรายชื้อวิชาของโปรแกรมประยุกต์ที่เห็นใหม่ ที่มีการทำงานเหมือนกับข้อกำหนดแสดงรายชื้อวิชาของโปรแกรมประยุกต์ที่เห็นเดิม ดังรูปที่ 4.17 แสดงข้อกำหนดแสดงรายชื้อวิชาของโปรแกรมประยุกต์ที่เห็นเดิม

Use Case: แสดงรายชื่อวิชา <FL2>

Primary Actor: ผู้สอน

Stakeholders and Interests:

– **Teacher:** ผู้สอนต้องการดูว่ามีวิชาอะไรบ้างที่ตนเองได้สร้างไว้ก่อนหน้านี้แล้ว

Preconditions: ผู้สอนต้องทำการลงทะเบียนไว้แล้ว และระบบทำงานเป็นปกติ

Postconditions: แสดงรายชื่อวิชาของผู้สอนที่ลือคอินได้สร้างไว้แล้ว

Main Success Scenario (or Basic Flow):

1. ระบบทำการตรวจสอบสถานะของผู้ทำการลือคอิน
2. ทำการค้นหารายชื่อวิชาของผู้ที่ลือคอินทั้งหมด
3. แสดงรายชื่อวิชาทั้งหมดของผู้ลือคอินคนนั้นๆ <Alt1>

Extensions (or Alternative Flows):

1. สถานะของผู้ลือคอิน ไม่ถูกต้อง
 - 1.1 แสดงข้อความการผิดพลาด
2. ค้นหาผิดพลาด
 - 2.1 แสดงข้อความการผิดพลาด
3. ไม่มีวิชาที่ผู้ลือคอินเรียกแสดง
 - 3.1 แสดงหน้าเปล่า

Variation:

Alt1:

1. ส่วนต่อประสานผู้ใช้ <HS1>

Frameworks:

Framelets:

Hotspot:

HS1: ส่วนต่อประสานผู้ใช้

รูปที่ 4.15 แสดงข้อกำหนดแสดงรายชื่อวิชา

Name: สุกส่วนต่อประสานผู้ใช้ในการแสดงรายชื้อวิชา

Requirement: สุกที่ใช้เปลี่ยนแปลงการแสดงผลส่วนต่อประสานผู้ใช้ในส่วนการแสดงผลรายชื้อวิชา ให้เป็นไปตามความต้องการของผู้ใช้งานเฟรมเวิร์ก

Type: Adding

Area: เฟรมเลตแสดงผลรายชื้อวิชา

Changes:

1. แสดงรหัสวิชา หลักสูตร วันที่เปิด เวลาใช้งาน/หมดอายุ สถานะปัจจุบัน และเวลาใช้ระบบ

รูปที่ 4.16 แสดงคำอธิบายสุกในส่วนต่อประสานผู้ใช้ในการแสดงรายชื้อวิชา

Use Case: แสดงรายชื้อวิชา

Primary Actor: ผู้สอน

Stakeholders and Interests:

— **Teacher:** ผู้สอนต้องการแสดงรายชื้อวิชาที่ได้สร้างไว้

Preconditions: ผู้สอนต้องทำการลงทะเบียนไว้แล้ว และล็อกอินในฐานะผู้สอน

Postconditions: แสดงรายชื้อวิชาของผู้สอนที่ล็อกอินได้สร้างไว้แล้ว

Main Success Scenario (or Basic Flow):

1. ระบบทำการตรวจสอบสถานะของผู้ที่ทำการล็อกอิน
2. ทำการค้นหารายชื้อวิชาของผู้ที่ล็อกอินทั้งหมด
3. แสดงรหัสวิชา หลักสูตร วันที่เปิด เวลาใช้งาน/หมดอายุ สถานะปัจจุบัน และเวลาใช้ระบบ

Extensions (or Alternative Flows):

1. สถานะของผู้ล็อกอินไม่ถูกต้อง
 - 1.1 แสดงข้อความการผิดพลาด
2. ค้นหาผิดพลาด
 - 2.1 แสดงข้อความการผิดพลาด
3. ไม่มีวิชาของผู้ล็อกอินคนนั้น
 - 3.1 แสดงหน้าเปล่า

รูปที่ 4.17 แสดงข้อกำหนดแสดงรายชื้อวิชาของโปรแกรมประยุกต์เห็นเดิม

จากรูปที่ 4.15 -4.17 จะเห็นได้ว่าเมื่อนำส่วนการเปลี่ยนแปลงภายในสุกไปประกอบกับจุดแปรผัน A1t1 ภายในข้อกำหนดเฟรมเลตแสดงผลรายชื้อวิชา ก็จะมีขั้นตอนที่ 3 ภายในสายลำดับการทำงาน ที่มีการแสดงรหัสวิชา หลักสูตร วันที่เปิด เวลาใช้งาน/หมดอายุ สถานะปัจจุบัน และเวลาใช้ระบบ เหมือนกับข้อกำหนดแสดงรายชื้อวิชาของโปรแกรมประยุกต์เห็นเดิมไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.1.2 จุดแปรผันตัวเลือก Opt1 (ภายในอีเลิร์นนิ่งเฟรมเวอร์ก) เป็นจุดแปรผันที่เชื่อมต่อกับสอตสพอตสร้างวิชา ซึ่งจุดแปรผันตัวเลือกนั้นเป็นจุดแปรผันที่สามารถเลือกได้มากกว่าหนึ่ง หรือไม่เลือกเลยก็ได้ ซึ่งข้อกำหนดโปรแกรมประยุกต์ที่นั้นไม่สามารถสร้างวิชาในส่วนนี้ จึงไม่ต้องพัฒนาสอตสร้างวิชาเข้ามาประกอบกับสอตสพอตสร้างวิชาดังกล่าว

4.2.2.1.3 จุดแปรผันตัวเลือก Opt2 (ภายในอีเลิร์นนิ่งเฟรมเวอร์ก) เป็นจุดแปรผันที่เชื่อมต่อกับเฟรมเลตต่างๆ ที่สามารถทำการเลือกได้ในส่วนนี้ โดยในส่วนนี้ได้มีฟังก์ชันการทำงานให้เลือกทั้งหมด 9 ส่วน คือ ประกาศ สนทนาออนไลน์ กระดานข่าว แสดงแบบทดสอบ แสดงรายละเอียดวิชา แก้ไขรายละเอียดวิชา จัดการนักเรียน จัดการเนื้อหา และจัดการแบบทดสอบ ซึ่งข้อกำหนดโปรแกรมประยุกต์ที่นั้นเดิมที่กำหนดให้มีฟังก์ชันการทำงานทั้งหมด 6 ส่วน คือ ประกาศ แสดงรายละเอียดวิชา แก้ไขรายละเอียดวิชา จัดการนักเรียน จัดการเนื้อหา และจัดการแบบทดสอบ โดยในการทำงานดังกล่าวจะต้องทำการสร้างสอตเข้าไปประกอบกับสอตสพอตต่างๆ ภายในเฟรมเลต ประกาศ เฟรมเลตแสดงรายละเอียดวิชา เฟรมเลตแก้ไขรายละเอียดวิชา เฟรมเวอร์กย่อยจัดการนักเรียน เฟรมเวอร์กย่อยจัดการเนื้อหา และเฟรมเวอร์กย่อยจัดการแบบทดสอบ โดยในส่วนของเฟรมเลตต่างๆ จะมีลักษณะการทำงานเช่นเดียวกับเฟรมเลตแสดงรายชื่อวิชาที่ได้แสดงรายละเอียดไว้แล้วข้างต้น ซึ่งรายละเอียดของการพัฒนาสอตเข้ามาประกอบกับสอตสพอตต่างๆ ดังกล่าวได้แสดงไว้ในภาคผนวก ข.

ต่อไปจะเป็นการพัฒนาเฟรมเวอร์กย่อย โดยในที่นี้จะยกตัวอย่างการพัฒนาเฟรมเวอร์กย่อยจัดการนักเรียน ซึ่งมีรายละเอียดการทำงานดังนี้

เฟรมเวอร์กย่อยจัดการนักเรียน เป็นส่วนที่รวมเฟรมเลตในส่วนจัดการนักเรียนไว้ทั้งหมด ซึ่งภายในสายลำดับการทำงานจะประกอบไปด้วยจุดแปรผันต่างๆ โดยที่จุดแปรผันต่างๆ นั้น จะเป็นจุดที่ใช้เชื่อมต่อกับเฟรมเลตภายในเฟรมเวอร์กย่อยจัดการนักเรียน ดังรูปที่ 4.18

Main Success Scenario (or Basic Flow):

1. เข้าสู่หน้าหลักจัดการนักเรียน <Alt1>
2. เลือกฟังก์ชันการทำงานต่างๆ <Opt1>

Variation:**Alt1:**

1. แสดงรายชื่อนักเรียน <FL17>

Opt1:

1. เลือกแสดงรายละเอียดนักเรียน <FL18>
2. เลือกลบนักเรียน <FL19>
3. เลือกเพิ่มนักเรียน <HS1>

รูปที่ 4.18 แสดงสายลำดับการทำงาน และจุดแปรผันในส่วนเฟรมเวอร์กย่อยจัดการนักเรียน

จากรูปที่ 4.18 จะเห็นได้ว่าข้อกำหนดเฟรมเวอร์กย่อยจัดการนักเรียนประกอบด้วยจุดแปรผันทั้งหมด 2 จุด คือ จุดแปรผันทางเลือก Alt1 (ภายในเฟรมเวอร์กย่อยจัดการนักเรียน) เป็นจุดแปรผันที่เชื่อมต่อกับเฟรมแสดงรายชื่อจัดการนักเรียน และจุดแปรผันตัวเลือก Opt1 (ภายในเฟรมเวอร์กย่อยจัดการนักเรียน) ซึ่งเป็นจุดแปรผันที่เชื่อมต่อกับเฟรมแสดงรายละเอียดนักเรียน เฟรมลบนักเรียน และเฟรมเพิ่มนักเรียน ซึ่งข้อกำหนดโปรแกรมประยุกต์ที่เดิมนั้นกำหนดให้มีฟังก์ชันการทำงานทั้งหมด 1 ส่วน คือ แสดงรายละเอียดนักเรียน ดังรูปที่ 4.19 แสดงข้อกำหนด โปรแกรมประยุกต์เห็นในส่วนจัดการนักเรียน

Use Case: จัดการนักเรียน**Primary Actor:** Teacher**Stakeholders and Interests:**

- **Abstract Actor:** ผู้สอนต้องการจัดการนักเรียนในวิชา อาทิ แสดงรายชื่อจัดการนักเรียน แสดงรายละเอียดนักเรียน เป็นต้น

Preconditions: ผู้สอนต้องทำการลงทะเบียน และล็อกอินในฐานะผู้สอน**Postconditions:** ผู้สอนจัดการนักเรียนต่างๆ ภายในระบบได้สำเร็จ**Main Success Scenario (or Basic Flow):**

1. เข้าสู่หน้าหลักจัดการนักเรียน และแสดงรายชื่อจัดการนักเรียน
2. เลือกฟังก์ชันการทำงานต่างๆ
 - 2.1 แสดงรายละเอียดนักเรียน

Extensions (or Alternative Flows):

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันวิจัยและพัฒนาเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.18-4.19 จะเห็นได้ว่าข้อกำหนดโปรแกรมประยุกต์เห็นในส่วนจัดการนักเรียน มีสายลำดับการทำงาน 2 ขั้นตอน เช่นเดียวกับสายลำดับการทำงานภายในข้อกำหนดเฟรมเวอร์กย่อยจัดการนักเรียน โดยมีขั้นตอนการพัฒนาสุคเข้ามาประกอบ ดังนี้

1. ในขั้นตอนแรกในสายลำดับการทำงานภายในเฟรมเวอร์กย่อยจัดการนักเรียน จะเป็นการเข้าสู่หน้าหลักจัดการนักเรียนเหมือนกัน โดยในข้อกำหนดโปรแกรมประยุกต์เห็นจะมีส่วนในการแสดงรายชื่อนักเรียน ซึ่งภายในสายลำดับการทำงานของข้อกำหนดเฟรมเวอร์กย่อยจัดการนักเรียน ได้มีจุดแปรผันทางเลือกที่เชื่อมต่อกับข้อกำหนดเฟรมแสดงรายชื่อเรียน ถ้าผู้ใช้งานเฟรมเวอร์กต้องการสร้างข้อกำหนดโปรแกรมประยุกต์เห็นจากอีเลิร์นนิ่งเฟรมเวอร์ก จึงเข้าไปดูที่ข้อกำหนดเฟรมแสดงรายชื่อเรียน และสร้างสุคเข้าไปประกอบกับฮอตสปอตต่างๆ ภายในข้อกำหนดเฟรมแสดงรายชื่อเรียน ดังนี้

เฟรมแสดงรายชื่อเรียน ผู้ใช้งานเฟรมเวอร์กเพียงแต่พัฒนาสุคเข้าไปประกอบกับฮอตสปอตที่ได้เตรียมไว้ในข้อกำหนดเฟรมแสดงรายชื่อเรียน ดังรูปที่ 4.20 แสดงข้อกำหนดเฟรมแสดงรายชื่อเรียน จะเห็นได้ว่าข้อกำหนดเฟรมแสดงรายชื่อเรียน ประกอบด้วยจุดแปรผันทั้งหมด 2 จุด คือ จุดแปรผันทางเลือก Alt1 (ภายในเฟรมแสดงรายชื่อเรียน) เป็นจุดแปรผันที่เชื่อมต่อกับฮอตสปอตส่วนต่อประสานผู้ใช้ ผู้ใช้งานเฟรมเวอร์กเพียงแต่ทำการพัฒนาสุคเข้าไปประกอบกับฮอตสปอตส่วนต่อประสานผู้ใช้ ดังรูปที่ 4.21 แสดงคำอธิบายสุคในส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อเรียน ที่พัฒนาขึ้นเพื่อนำไปประกอบเข้ากับฮอตสปอตภายในเฟรมแสดงรายชื่อ ให้มีการแสดงเหมือนกับข้อกำหนดโปรแกรมประยุกต์เห็นเดิม โดยในส่วนการเปลี่ยนแปลงจะเป็นสายลำดับการทำงานที่ทำงานจะไปประกอบกับจุดแปรผัน Alt1 ภายในข้อกำหนดเฟรมแสดงรายชื่อเรียน ซึ่งเมื่อทำการประกอบเสร็จแล้วก็จะได้ข้อกำหนดแสดงรายชื่อเรียนของ โปรแกรมประยุกต์เห็นใหม่ ที่มีการทำงานเหมือนกับข้อกำหนดแสดงรายชื่อเรียนของ โปรแกรมประยุกต์เห็นเดิม ดังรูปที่ 4.22 แสดงข้อกำหนดแสดงรายชื่อเรียนของ โปรแกรมประยุกต์เห็นเดิม

<p>Use Case: แสดงรายชื่อนักเรียน <FL18></p> <p>Primary Actor: ผู้สอน</p> <p>Stakeholders and Interests:</p> <p>– Teacher: ผู้สอนสามารถเรียกดูรายชื่อนักเรียนที่ได้เข้ามาลงทะเบียนในวิชานั้นๆ ได้</p> <p>Preconditions: จะต้องอยู่ในหน้าหลักของวิชา</p> <p>Postconditions: ผู้สอนรู้ว่ามึนักเรียนคนใดในวิชานั้นๆบ้าง</p> <p>Main Success Scenario (or Basic Flow):</p> <ol style="list-style-type: none"> 1. ระบบทำการค้นหาชื่อนักเรียนของวิชานั้นๆ 2. แสดงรายชื่อนักเรียนของวิชานั้นๆ <Alt1> <p>Extensions (or Alternative Flows):</p> <ol style="list-style-type: none"> 1. ค้นหาผิดพลาด <ol style="list-style-type: none"> 1.1 แสดงข้อความการผิดพลาด 2. ไม่มีแบบทดสอบของวิชานั้นๆ <ol style="list-style-type: none"> 2.1 แสดงหน้าเปล่า <p>Variation:</p> <p>Alt1:</p> <ol style="list-style-type: none"> 1. ส่วนต่อประสานผู้ใช้ <HS1> <p>Frameworks:</p> <p>Framelets:</p> <p>Hotspot:</p> <p>HS1: ส่วนต่อประสานผู้ใช้</p>

รูปที่ 4.20 แสดงข้อกำหนดเฟรมเลตแสดงรายชื่อนักเรียน

<p>Name: สุกส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อนักเรียน</p> <p>Requirement: สุกที่ใช้เปลี่ยนแปลงการแสดงส่วนต่อประสานผู้ใช้ในส่วนการแสดงรายชื่อนักเรียนให้เป็นไปตามความต้องการของผู้ใช้งานเฟรมเวอร์ก</p> <p>Type: Adding</p> <p>Area: เฟรมเลตแสดงรายชื่อนักเรียน</p> <p>Changes:</p> <ol style="list-style-type: none"> 1. แสดงลำดับ ชื่อผู้ใช้ เปิด เวลาใช้งาน/หมดอายุ สถานะปัจจุบัน เวลาใช้ระบบ
--

รูปที่ 4.21 แสดงคำอธิบายสุกในส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อนักเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Use Case: แสดงรายชื่อนักเรียน

Primary Actor: ผู้สอน

Stakeholders and Interests:

— **Teacher:** ผู้สอนสามารถเรียกดูรายชื่อนักเรียนที่เรียนวิชานั้นๆ ได้

Preconditions: จะต้องมีการสร้างวิชาไว้แล้ว และอยู่ในหน้าหลักวิชา

Postconditions: ผู้สอนสามารถทราบได้ว่ามีนักเรียนคนใดมาลงทะเบียนในวิชานั้นๆ

Main Success Scenario (or Basic Flow):

1. ระบบทำการค้นหารายชื่อนักเรียนทั้งหมดที่ได้ลงทะเบียนในวิชานั้นๆ
2. แสดงลำดับ ชื่อผู้ใช้ เปิด เวลาใช้งาน/หมดอายุ สถานะปัจจุบัน เวลาใช้ระบบ

Extensions (or Alternative Flows):

1. ค้นหาผิดพลาด
 - 1.1 แสดงข้อความการผิดพลาด
2. ไม่มีนักเรียนของวิชานั้น
 - 2.1 แสดงหน้าเปล่า

รูปที่ 4.22 แสดงข้อกำหนดโปรแกรมประยุกต์เห็นในส่วนแสดงรายชื่อนักเรียน

จากรูปที่ 4.20-4.22 จะเห็นได้ว่าเมื่อนำสุดไปประกอบกับจุดแปรผัน Alt1 ภายในข้อกำหนดเฟรมแสดงรายชื่อนักเรียน ก็จะมีขั้นตอนที่ 2 ภายในสายลำดับการทำงาน ที่มีการแสดงลำดับ ชื่อผู้ใช้ เปิด เวลาใช้งาน/หมดอายุ สถานะปัจจุบัน และเวลาใช้ระบบ เหมือนกับข้อกำหนดแสดงรายชื่อนักเรียนของโปรแกรมประยุกต์เห็นเดิม

2. ในขั้นตอนที่สองในสายลำดับการทำงานภายในเฟรมเวอร์ก้อยจัดการนักเรียน จะเป็นการเลือกฟังก์ชันการทำงานต่างๆ โดยในข้อกำหนดโปรแกรมประยุกต์เห็นจะมีส่วนฟังก์ชันการทำงาน 1 ส่วน คือ การแสดงรายละเอียดนักเรียน ซึ่งภายในสายลำดับการทำงานข้อกำหนดเฟรมเวอร์ก้อยจัดการนักเรียน มีจุดแปรผันตัวเลือกที่เชื่อมต่อกับข้อกำหนดเฟรมแสดงรายละเอียดนักเรียน ถ้าผู้ใช้งานเฟรมเวอร์กต้องการสร้างข้อกำหนดโปรแกรมประยุกต์เห็นจากอีเลิร์นนิ่งเฟรมเวอร์กก็ทำการเข้าไปดูที่ข้อกำหนดเฟรมแสดงรายละเอียดนักเรียน และสร้างซุกเข้าไปประกอบกับซอตสปอตต่างๆ ภายในข้อกำหนดเฟรมแสดงรายละเอียดนักเรียน โดยมีลักษณะการทำงานเช่นเดียวกับเฟรมแสดงรายชื่อนักเรียนที่ได้แสดงไว้แล้วข้างต้น โดยได้แสดงรายละเอียดทั้งหมดไว้ในภาคผนวก ข.

เมื่อทำการพัฒนาซุกเพื่อเข้าไปประกอบในแต่ละจุดซอตสปอตภายในเฟรมเวอร์ก เฟรมเวอร์ก้อย และเฟรมแสดงทั้งหมดแล้วก็จะได้ข้อกำหนดโปรแกรมประยุกต์เห็น ซึ่งในการสร้างข้อกำหนดโปรแกรมประยุกต์เห็นโดยใช้ข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กนั้น สามารถสร้างได้ง่าย

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กว่าการสร้างข้อกำหนดโปรแกรมประยุกต์เห็นตั้งแต่ต้น เนื่องจากข้อกำหนดอิเล็กทรอนิกส์นิ่งเฟรมเวอร์ก นั้นมีสายลำดับการทำงานหลักทั้งหมดของข้อกำหนดโปรแกรมประยุกต์ในโดเมนอิเล็กทรอนิกส์นิ่งไว้แล้ว ผู้ใช้งานอิเล็กทรอนิกส์นิ่งเฟรมเวอร์กเพียงแค่พัฒนาสูกให้ตรงกับข้อกำหนดโปรแกรมประยุกต์เห็นเข้าไป ประกอบเข้ากับฮอตสปอตต่างๆ ที่ข้อกำหนดเฟรมเวอร์ก ข้อกำหนดเฟรมเวอร์กย่อย หรือ ข้อกำหนดเฟรมเลตเตรียมไว้ให้ โดยรายละเอียดทั้งหมดในการสร้างข้อกำหนดโปรแกรมประยุกต์ เห็น การสร้างข้อกำหนดโปรแกรมประยุกต์มูลิต และการสร้างข้อกำหนดโปรแกรมประยุกต์เอ็ดว เตอร์ จากอิเล็กทรอนิกส์นิ่งเฟรมเวอร์กทั้งหมดได้แสดงไว้ในภาคผนวก ข. ภาคผนวก ค. และภาคผนวก ง. ตามลำดับ

4.2.2.2 การสร้างข้อกำหนดโปรแกรมประยุกต์ภายใต้โดเมนอิเล็กทรอนิกส์นิ่ง นอกเหนือจากโปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ก โดยในการสร้าง ข้อกำหนดดังกล่าวจะมีขั้นตอนการทำงานเช่นเดียวกับการสร้างข้อกำหนดโปรแกรมประยุกต์ที่ นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ก โดยพัฒนาสูกเข้าไปประกอบกับฮอตสปอตต่างๆ ภายในเฟรมเวอร์ก เฟรมเวอร์กย่อย หรือเฟรมเลต เช่น จุดแปรผันทางเลือก All (ภายในอิเล็กทรอนิกส์นิ่ง เฟรมเวอร์ก) เป็นจุดแปรผันที่ใช้เชื่อมต่อกับข้อกำหนดเฟรมเลตแสดงรายชื่อวิชา ผู้ใช้งานเฟรม เวอร์กเพียงแค่พัฒนาสูกเข้ามาประกอบกับฮอตสปอตที่ได้เตรียมไว้ในข้อกำหนดเฟรมเลตแสดง รายชื่อวิชา ซึ่งในการสร้างข้อกำหนดโปรแกรมประยุกต์ดังกล่าวมีความเป็นไปได้ที่ข้อกำหนดเฟรม เวอร์กจะไม่รองรับการทำงานที่ข้อกำหนดโปรแกรมประยุกต์ต้องการ ในส่วนนี้เราสามารถทำการ ปรับปรุงข้อกำหนดเฟรมเวอร์กได้ เนื่องจากในส่วนนี้เป็นเพียงข้อกำหนดที่สามารถปรับปรุงได้ง่าย กว่าการปรับปรุงเฟรมเวอร์กที่ผ่านขั้นตอนการออกแบบ ขั้นตอนการพัฒนา และขั้นตอนการ ทดสอบแล้ว โดยข้อกำหนดเฟรมเวอร์กที่ผ่านการปรับปรุงแล้วจะสามารถรองรับการทำงานได้มาก ยิ่งขึ้นด้วย โดยในการสร้างข้อกำหนดโปรแกรมประยุกต์ภายใต้โดเมนอิเล็กทรอนิกส์นิ่งนอกเหนือจาก โปรแกรมประยุกต์ที่นำมาพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์กจะยกตัวอย่าง โดยสร้าง ข้อกำหนดโปรแกรมประยุกต์อิเล็กทรอนิกส์แสดควร์ ซึ่งสามารถแสดงข้อกำหนดโปรแกรมประยุกต์อิเล็กทรอนิกส์แสดควร์ในส่วนแสดงรายชื่อวิชาได้ดังรูปที่ 4.23 โดยจะนำข้อกำหนดดังกล่าวไปเปรียบเทียบกับ ข้อกำหนดโปรแกรมประยุกต์อิเล็กทรอนิกส์แสดควร์ที่ได้จากอิเล็กทรอนิกส์นิ่งเฟรมเวอร์ก ซึ่งในการสร้างข้อกำหนด โปรแกรมประยุกต์อิเล็กทรอนิกส์แสดควร์จากอิเล็กทรอนิกส์นิ่งเฟรมเวอร์กในส่วนแสดงรายชื่อวิชา ก็ให้ทำการ พัฒนาสูกเข้าไปประกอบกับฮอตสปอตต่างๆ ภายในเฟรมเลตแสดงรายชื่อวิชา ดังรูปที่ 4.15 ให้ตรงกับข้อกำหนดโปรแกรมประยุกต์อิเล็กทรอนิกส์แสดควร์ ดังรูปที่ 4.24 แสดงคำอธิบายสูกในส่วนต่อประสาน ผู้ใช้ในการแสดงรายชื่อวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Use Case: แสดงรายชื่อวิชา

Primary Actor: ผู้สอน

Stakeholders and Interests:

— **Teacher:** ผู้สอนต้องการแสดงรายชื่อวิชาที่ได้สร้างไว้

Preconditions: ผู้สอนต้องทำการลงทะเบียนไว้แล้ว และล็อกอินในฐานะผู้สอน

Postconditions: แสดงรายชื่อวิชาของผู้สอนที่ล็อกอินได้สร้างไว้แล้ว

Main Success Scenario (or Basic Flow):

1. ระบบทำการตรวจสอบสถานะของผู้ที่ทำการล็อกอิน
2. ทำการค้นหารายชื่อวิชาของผู้ที่ล็อกอินทั้งหมด
3. แสดงวันที่เปิดสอน ชื่อวิชา

Extensions (or Alternative Flows):

1. สถานะของผู้ล็อกอิน ไม่ถูกต้อง
 - 1.1 แสดงข้อความการผิดพลาด
2. ค้นหาผิดพลาด
 - 2.1 แสดงข้อความการผิดพลาด
3. ไม่มีวิชาของผู้ล็อกอินคนนั้น
 - 3.1 แสดงหน้าเปล่า

รูปที่ 4.23 แสดงข้อกำหนด โปรแกรมประยุกต์เว็บแสดควร์ในส่วนแสดงรายชื่อวิชา

Name: สุกส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อวิชา

Requirement: สุกที่ใช้เปลี่ยนแปลงการแสดงผลส่วนต่อประสานผู้ใช้ในส่วนการแสดงผลรายชื่อวิชา ให้เป็นไปตามความต้องการของผู้ใช้งานเฟรมเวิร์ก

Type: Adding

Area: เฟรมเลตแสดงรายชื่อวิชา

Changes:

1. แสดงวันที่เปิดสอน ชื่อวิชา

รูปที่ 4.24 แสดงคำอธิบายสุกในส่วนต่อประสานผู้ใช้ในการแสดงรายชื่อวิชา

จากรูปที่ 4.24 แสดงสุกที่ใช้ในการเปลี่ยนแปลงการทำงานของข้อกำหนดเฟรมเลตแสดงผลรายชื่อวิชา ให้มีการแสดงผลเหมือนกับข้อกำหนดโปรแกรมประยุกต์เว็บแสดควร์ โดยในส่วนการเปลี่ยนแปลงจะเป็นสายลำดับการทำงานที่ท่าจะนำไปประกอบกับจุดแปรผัน A11 ภายในข้อกำหนดเฟรมเลตแสดงผลรายชื่อวิชา ซึ่งเมื่อทำการประกอบเสร็จแล้วก็จะได้ข้อกำหนดแสดงรายชื่อ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาของโปรแกรมประยุกต์เลิร์นแสดควร์ เมื่อนำส่วนการเปลี่ยนแปลงภายในสู่ภายนอกกับจุดแปรผัน AI1 ภายในข้อกำหนดเฟรมเลตแสดงรายชื่อวิชา จะมีขั้นตอนที่ 3 ภายในสายลำดับการทำงาน ที่มีการแสดงวันที่เปิดสอน ชื่อวิชา เหมือนกับข้อกำหนดแสดงรายชื่อวิชาของ โปรแกรมประยุกต์เลิร์นแสดควร์

ส่วนต่อไปจะอธิบายในส่วนที่ข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กไม่รองรับการทำงานของข้อกำหนดโปรแกรมประยุกต์ เนื่องจากการพัฒนาข้อกำหนดอีเลิร์นนิ่งดังกล่าวได้พัฒนาจากข้อกำหนดโปรแกรมประยุกต์เพียง 3 โปรแกรม ข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กดังกล่าวจึงอาจมีส่วนที่ไม่รองรับการทำงานของโปรแกรมประยุกต์อื่นๆ ขั้นตอนนี้จึงปรับปรุงข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กให้รองรับการทำงานดังกล่าว เช่น โปรแกรมประยุกต์เลิร์นแสดควร์มีความต้องการให้สามารถสร้างวิชา ลบวิชา และแก้ไขรายละเอียดวิชาได้ ภายในขั้นตอนที่ 2 ของสายลำดับการทำงานอีเลิร์นนิ่งเฟรมเวอร์ก แต่อีเลิร์นนิ่งเฟรมเวอร์กมีการทำงานในส่วนดังกล่าวเพียง 1 การทำงาน คือ สร้างวิชา โดยในส่วนนี้เป็นเพียงข้อกำหนดของเฟรมเวอร์กจึงสามารถปรับข้อกำหนดเฟรมเวอร์กดังกล่าวให้รองรับการทำงานของข้อกำหนดโปรแกรมประยุกต์เลิร์นแสดควร์ได้ ดังรูปที่ 4.25 แสดงสายลำดับการทำงานบางส่วนของโปรแกรมประยุกต์เลิร์นแสดควร์

Main Success Scenario (or Basic Flow):

1. ล็อกอินเข้ามาภายในระบบ
2. ค้นหารายชื่อวิชาทั้งหมดของผู้ล็อกอิน
3. แสดงหน้าหลักผู้ล็อกอิน และแสดงรายชื่อวิชาทั้งหมดที่ค้นหาได้
 - 3.1 เลือกสร้างวิชา
 - 3.2 เลือกลบวิชา
 - 3.3 เลือกแก้ไขรายละเอียดวิชา
4. เลือกวิชาที่ต้องการ

รูปที่ 4.25 แสดงสายลำดับการทำงานบางส่วนของโปรแกรมประยุกต์เลิร์นแสดควร์

จากรูปที่ 4.25 จะเห็นได้ว่าในขั้นตอนการแสดงผลหน้าหลักผู้ล็อกอิน มีส่วนฟังก์ชันการทำงานทั้งหมด 3 ส่วน คือ สร้างวิชา ลบวิชา และแก้ไขรายละเอียดวิชา ซึ่งข้อกำหนดอีเลิร์นนิ่งเฟรมเวอร์กรองรับการทำงานแค่เพียงส่วนเดียว คือ สร้างวิชา ดังรูปที่ 4.26 แสดงสายลำดับการทำงานบางส่วนของอีเลิร์นนิ่งเฟรมเวอร์ก

Main Success Scenario (or Basic Flow):

1. ล็อกอินเข้ามาภายในระบบ
2. แสดงหน้าหลักผู้ล็อกอิน <Alt1> <Opt1>
3. เลือกวิชาที่ต้องการ

Variation:

Alt1:

1. แสดงรายชื่อวิชา <FL2>

Opt1:

1. สามารถสร้างวิชา <HS1>

รูปที่ 4.26 แสดงสายลำดับการทำงานบางส่วนของอีเลิร์นนิ่งเฟรมเวิร์ก

จากรูปที่ 4.25-4.26 จะเห็นได้ว่าข้อกำหนดคือเลิร์นนิ่งเฟรมเวิร์กไม่รองรับทำงานของข้อกำหนดโปรแกรมประยุกต์เลิร์นนิ่งแอสควอร์ในส่วนดังกล่าว ผู้พัฒนาเฟรมเวิร์กจึงต้องทำการปรับปรุงข้อกำหนดคือเลิร์นนิ่งเฟรมเวิร์กให้รองรับการทำงานดังกล่าว เพื่อให้ข้อกำหนดคือเลิร์นนิ่งเฟรมเวิร์กดังกล่าวสามารถรองรับการทำงานต่างๆ ได้มากยิ่งขึ้น โดยในการปรับปรุงข้อกำหนดคือเลิร์นนิ่งเฟรมเวิร์กในส่วนนี้จะไม่มีผลกระทบต่อส่วนอื่นๆ เนื่องจากได้ออกแบบให้จุดแปรผันดังกล่าวเป็นจุดแปรผันทางเลือกไว้แล้ว ดังรูปที่ 4.27

Main Success Scenario (or Basic Flow):

1. ล็อกอินเข้ามาภายในระบบ
2. แสดงหน้าหลักผู้ล็อกอิน <Alt1> <Opt1>
3. เลือกวิชาที่ต้องการ

Variation:

Alt1:

1. แสดงรายชื่อวิชา <FL2>

Opt1:

1. สร้างวิชา <HS1>
2. ลบวิชา <HS2>
3. แก้ไขรายละเอียดวิชา <FL4>

รูปที่ 4.27 แสดงสายลำดับการทำงานบางส่วนของอีเลิร์นนิ่งเฟรมเวิร์กที่ได้รับการปรับปรุง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดการสร้างข้อกำหนด โปรแกรมประยุกต์เลิร์นแอสควร์ทั้งหมดได้แสดงไว้ใน ภาคผนวก จ. โดยจากการสร้างข้อกำหนด โปรแกรมประยุกต์ที่นอกเหนือจาก โปรแกรมประยุกต์ที่ นำมาพิจารณาสร้างเป็นเฟรมเวิร์ก ช่วยให้ข้อกำหนดเฟรมเวิร์กสามารถรองรับการทำงานได้มาก ยิ่งขึ้น เนื่องจากการปรับปรุงข้อกำหนดเฟรมเวิร์กให้รองรับการทำงานของ โปรแกรมประยุกต์ ดังกล่าว ซึ่งในการปรับปรุงข้อกำหนดเฟรมเวิร์กนั้นมีข้อได้เปรียบที่สามารถปรับปรุงได้ง่ายกว่า การปรับปรุงเฟรมเวิร์กที่ผ่านขั้นตอนการออกแบบ ขั้นตอนการพัฒนา และขั้นตอนการทดสอบ แล้ว แต่ในการปรับปรุงเฟรมเวิร์กที่ได้จากเฟรมเลตที่ผ่านขั้นตอนดังกล่าวก็สามารถทำได้ง่ายกว่า การปรับปรุงเฟรมเวิร์กจากกระบวนการเดิม ซึ่งสามารถสร้างเฟรมเลตใหม่เพื่อทดแทนเฟรมเลต เดิมได้ โดยไม่ส่งผลกระทบต่อเฟรมเวิร์กทั้งระบบ

4.2.3 การเปรียบเทียบกระบวนการในการสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตกับการ สร้างข้อกำหนดเฟรมเวิร์กเดิม

เปรียบเทียบกระบวนการในการพัฒนาเฟรมเวิร์กจากระเบียบวิธีในงานวิจัยนี้ กับ กระบวนการพัฒนาเฟรมเวิร์กแบบเดิม โดยปัญหาที่เกิดขึ้นจากการพัฒนาข้อกำหนดเฟรมเวิร์ก เดิมมีหลากหลายประเภท โดยปัญหาที่การสร้างข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตสามารถช่วย แก้ไขปัญหาของการสร้างข้อกำหนดเฟรมเวิร์กเดิมประกอบด้วย ปัญหาที่หนึ่ง คือ ปัญหาความ เข้าใจในโครงสร้างของเฟรมเวิร์กในโดเมนที่มีขนาดใหญ่ และปัญหาที่สอง คือ ปัญหาการ เปลี่ยนแปลงของข้อกำหนดภายในโดเมน โดยมีรายละเอียดทั้ง 2 ส่วน ดังนี้

4.2.3.1 ปัญหาความเข้าใจโครงสร้างของเฟรมเวิร์กในโดเมนที่มีขนาดใหญ่ โดย เฟรมเวิร์กที่มีขนาดใหญ่จะประกอบไปด้วยจุดแปรผันต่างๆ มากมายทำให้ยากต่อการทำความเข้าใจ ซึ่งระเบียบวิธีจากงานวิจัยนี้ได้ออกแบบแนวทางในการแบ่งข้อกำหนดต่างๆ ให้มีขนาดเล็ก และแนวทางในการประกอบข้อกำหนดย่อยเหล่านั้น เพื่อให้ผู้พัฒนาเฟรมเวิร์กสามารถพัฒนา ข้อกำหนดเฟรมเวิร์กด้วยเฟรมเลตได้ง่าย จึงสามารถแยกทำความเข้าใจในแต่ละส่วนได้ง่าย โดย จากกระบวนการพัฒนาเฟรมเวิร์กเดิม [5][10] ไม่ได้มีกระบวนการในการแบ่งข้อกำหนดดังกล่าว เป็นข้อกำหนดย่อยๆ ซึ่งถ้าทำการพัฒนาข้อกำหนดเฟรมเวิร์กตามกระบวนการเดิมจะได้ ข้อกำหนดจัดการวิชา ดังรูปที่ 4.28

Main Success Scenario (or Basic Flow):

1. เข้าสู่หน้าหลักผู้ล็อกอิน
2. ระบบทำการตรวจสอบสถานะของผู้ล็อกอิน
3. ทำการค้นหารายชื่อวิชาของผู้ล็อกอินทั้งหมด
4. แสดงรายชื่อวิชาทั้งหมดของผู้ล็อกอินคนนั้นๆ <HS1>
 - <Opt1> สร้างวิชา
 1. ใส่รายละเอียดวิชา <HS2>
 2. ยืนยันการสร้างวิชา
5. เข้าสู่หน้าหลักวิชา
 - <Alt1> แสดงรายละเอียดวิชา
 1. ค้นหารายละเอียดวิชา
 2. แสดงรายละเอียดวิชา <HS3>
 - <Alt2> แสดงรายชื่อนี้เนื้อหา
 1. ค้นหารายชื่อนี้เนื้อหา
 2. แสดงรายชื่อนี้เนื้อหา <HS4>
6. เลือกฟังก์ชันการทำงานต่างๆ
 - <Opt1> เลือกแก้ไขรายละเอียดวิชา
 1. ค้นหารายละเอียดวิชา
 2. แสดงรายละเอียดวิชา <HS5>
 3. แก้ไขรายละเอียดวิชาตามต้องการ
 4. ยืนยันการแก้ไข
 - <Opt2> เลือกลบวิชา
 1. เลือกวิชาที่ต้องการลบ
 2. ยืนยันการลบ

รูปที่ 4.28 แสดงสายลำดับการทำงานข้อกำหนดเฟรมเวอร์กจัดการวิชาจากกระบวนการเดิม

จากรูปที่ 4.28 จะเห็นได้ว่าสายลำดับการทำงานของข้อกำหนดเฟรมเวอร์กจัดการวิชาจากกระบวนการเดิม ประกอบด้วยจุดแปรผันต่างๆ และมีรายละเอียดภายในจำนวนมาก เนื่องจากมีส่วนการทำงานภายในหลายส่วน เช่น ส่วนการแสดงรายละเอียดวิชา ส่วนการแก้ไขรายละเอียดวิชา โดยถ้าใช้ระเบียบวิธีที่ได้จากงานวิจัยนี้สามารถลดความซับซ้อนในส่วนดังกล่าวได้ เนื่องจากได้แบ่งส่วนการทำงานดังกล่าวออกเป็นส่วนย่อย ดังรูปที่ 4.29 แสดงสายลำดับการทำงานข้อกำหนด

เฟรมเวอร์กจัดการวิชาที่ได้จากงานวิจัยนี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Main Success Scenario (or Basic Flow):

1. เข้าสู่หน้าหลักผู้ล็อกอิน <Alt1> <Opt1>
2. เข้าสู่หน้าหลักวิชา <Alt2>
3. เลือกฟังก์ชันการทำงานต่างๆ <Opt2>

Variation:

Alt1:

1. แสดงรายชื่อวิชา <FL1>

Opt1:

1. สร้างวิชา <HS1>

Alt2:

1. แสดงรายละเอียดวิชา <FL2>
2. แสดงรายชื่อนักเรียน <FL4>

Opt2:

1. แก้ไขรายละเอียดวิชา <FL3>
2. ลบวิชา <HS2>

รูปที่ 4.29 แสดงสายลำดับการทำงานข้อกำหนดเฟรมเวิร์กจัดการวิชาที่ได้จากงานวิจัยนี้

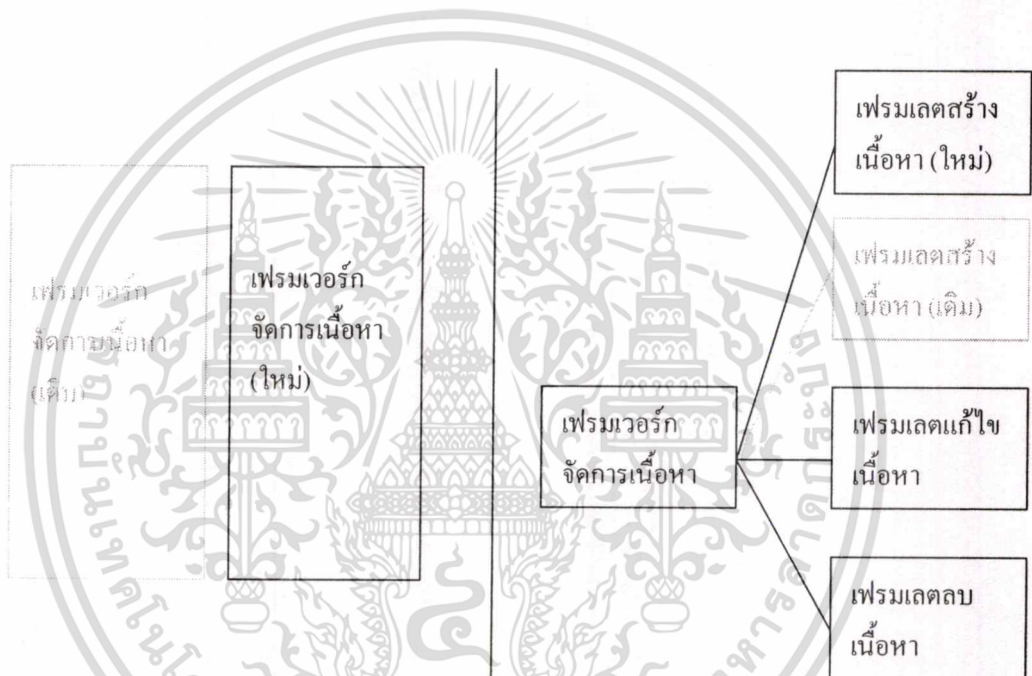
จากรูปที่ 4.28-4.29 สายลำดับการทำงานข้อกำหนดเฟรมเวิร์กจากกระบวนการเดิมจะประกอบด้วยฮอตสปอต และรายละเอียดต่างๆ ซับซ้อนมาก แต่สายลำดับการทำงานข้อกำหนดเฟรมเวิร์กที่ได้จากงานวิจัยนี้จะประกอบด้วยฮอตสปอตที่เชื่อมต่อกับเฟรมเลตต่างๆ ผู้ใช้งานเฟรมเวิร์กจึงสามารถเข้าไปทำความเข้าใจกับฮอตสปอตต่างๆ ภายในเฟรมเลตได้ง่ายกว่า

4.2.3.2 การเปลี่ยนแปลงของข้อกำหนดภายในโดเมน เนื่องจากเฟรมเวิร์กเป็นวิธีหนึ่งในการนำกลับมาใช้ใหม่ ซึ่งเฟรมเวิร์กดังกล่าวจะมีฮอตสปอตต่างๆ ภายในเพื่อรองรับการเปลี่ยนแปลงของโปรแกรมประยุกต์ที่พัฒนาจากเฟรมเวิร์ก แต่ฮอตสปอตต่างๆ ภายในเฟรมเวิร์กอาจจะไม่ครอบคลุมการทำงานทั้งหมดในการสร้าง โปรแกรมประยุกต์ได้ ซึ่งถ้าเป็นการพัฒนาเฟรมเวิร์กแบบเดิมนั้นจำเป็นต้องพัฒนาเฟรมเวิร์กใหม่ทั้งระบบ หรือปรับปรุงเฟรมเวิร์กภายในหลายจุด เนื่องจากเฟรมเวิร์กที่ได้จากกระบวนการเดิมมีขนาดใหญ่ ในการปรับปรุงเพียงเล็กน้อย อาจส่งผลกระทบต่อเฟรมเวิร์กทั้งหมดได้ แต่เฟรมเวิร์กที่ได้จากงานวิจัยนี้สามารถช่วยลดปัญหาดังกล่าว เนื่องจากข้อกำหนดที่ได้จากระเบียบวิธีนี้แบ่งเป็นส่วนย่อยๆ ถ้ามีการเปลี่ยนแปลงที่ไม่ครอบคลุมการทำงานดังกล่าว ก็สามารถพัฒนาเพียงส่วนเฟรมเลต หรือเฟรมเวิร์กย่อยที่ไม่ครอบคลุมการทำงานใหม่ได้ ไม่จำเป็นต้องพัฒนาเฟรมเวิร์กใหม่ทั้งระบบ เช่น ถ้าผู้ใช้งานเฟรมเวิร์กต้องการให้ในการสร้างเนื้อหา มีการใส่รหัสป้องกันการสร้างเนื้อหา ก่อนการยืนยันการสร้าง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้เพื่อการศึกษาเท่านั้น เมื่อผู้ใช้ที่เห็นใบเขียวจะยืนยันการกระทำ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื้อหา ซึ่งผู้พัฒนาเฟรมเวิร์กไม่ได้สร้างสอตสปอตเพื่อรองรับการทำงานดังกล่าว โดยถ้าต้องการให้เฟรมเวิร์กดังกล่าวรองรับการทำงานดังกล่าวจำเป็นจะต้องพัฒนาเฟรมเวิร์กใหม่ หรือปรับปรุงเฟรมเวิร์กภายในหลายจุด แต่ถ้าเป็นเฟรมเวิร์กที่ได้จากระเบียบวิธีนี้ สามารถที่จะพัฒนาเฟรมเลตสร้างเนื้อหาเพื่อรองรับการทำงานดังกล่าวเพียงส่วนเดียวก็สามารถใช้งานได้ โดยในการแก้ไข หรือปรับปรุงเฟรมเวิร์ก ถ้าสามารถแก้ไขหลังจากที่สร้างข้อกำหนดเฟรมเวิร์กแล้ว จะสามารถแก้ไขได้ง่ายกว่าการแก้ไขหลังจากที่ผ่านขั้นตอนการออกแบบ ขั้นตอนการพัฒนา และขั้นตอนการทดสอบแล้ว ดังรูปที่ 4.30 แสดงการสร้างเฟรมเลตสร้างเนื้อหาใหม่เพื่อทดแทนเฟรมเลตสร้างเนื้อหาเดิม



รูปที่ 4.30 แสดงการสร้างเฟรมเลตสร้างเนื้อหาใหม่เพื่อทดแทนเฟรมเลตสร้างเนื้อหาเดิม

จากรูปที่ 4.30 จะเห็นได้ว่าการเปลี่ยนแปลงเฟรมเวิร์กดังกล่าวเพื่อรองรับการทำงานที่เพิ่มขึ้นนั้น เฟรมเวิร์กที่ได้จากระเบียบวิธีในงานวิจัยนี้สามารถสร้างเฟรมเลตสร้างเนื้อหาใหม่ ทดแทนเฟรมเลตสร้างเนื้อหาเดิม เพื่อรองรับการเปลี่ยนแปลงที่ผู้พัฒนาเฟรมเวิร์กเดิมไม่ได้เตรียมไว้ให้ได้ โดยการเปลี่ยนแปลงดังกล่าวจะไม่กระทบกับเฟรมเวิร์ก หรือเฟรมเลตในส่วนอื่นๆ แต่ถ้าเป็นเฟรมเวิร์กที่ได้จากกระบวนการเดิมนั้น ไม่ได้มีการแบ่งออกเป็นส่วนย่อยๆ จำเป็นต้องสร้างเฟรมเวิร์กในส่วนดังกล่าวใหม่ หรือต้องปรับปรุงเฟรมเวิร์กภายในหลายจุด

4.3 สรุปผลการประเมิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการประเมินผลระเบียบวิธีจากงานวิจัยนี้สามารถสร้างข้อกำหนดเฟรมเวิร์ก โดยข้อกำหนดเฟรมเวิร์กที่ได้มีประสิทธิภาพที่ดีกว่าข้อกำหนดเฟรมเวิร์กจากกระบวนการเดิม 2 ส่วน คือ ข้อกำหนดเฟรมเวิร์กที่ได้จากระเบียบวิธีนี้สามารถแยกทำความเข้าใจได้ง่ายกว่า และข้อกำหนดเฟรมเวิร์กที่ได้จากระเบียบวิธีนี้สามารถรองรับการเปลี่ยนแปลงได้มากกว่าการสร้างข้อกำหนดเฟรมเวิร์กจากกระบวนการเดิม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและข้อเสนอแนะ

5.1 สรุป

เฟรมเวอร์กเป็นอีกหนึ่งแนวทางในการพัฒนาโปรแกรมประยุกต์ที่มีประสิทธิภาพ ซึ่งช่วยให้การพัฒนาโปรแกรมประยุกต์ทำได้สะดวกยิ่งขึ้น แต่การพัฒนาเฟรมเวอร์กยากกว่าการพัฒนาโปรแกรมประยุกต์เดิม เนื่องจากเฟรมเวอร์กต้องพิจารณาจากกลุ่มโปรแกรมประยุกต์ที่สัมพันธ์กันมาสร้างเป็น โครงสร้างหลักของเฟรมเวอร์ก ซึ่งส่วนมากแล้วเฟรมเวอร์กจะมีขนาดใหญ่ และมีความซับซ้อนสูง ด้วยเหตุนี้เองแนวคิดเฟรมเลตจึงได้รับการพัฒนาขึ้นเพื่อช่วยลดปัญหาซับซ้อนในการพัฒนาเฟรมเวอร์ก เฟรมเลตเป็นเฟรมเวอร์กขนาดเล็กที่มีหน้าที่การทำงานเฉพาะ จึงมีความซับซ้อนน้อยกว่าเฟรมเวอร์กที่มีขนาดใหญ่ ผู้ใช้งานเฟรมเวอร์กสามารถนำเอาเฟรมเลตแต่ละตัวมาประกอบกันเข้าเป็นเฟรมเวอร์กได้ โดยกระบวนการในการพัฒนาเฟรมเวอร์กโดยใช้เฟรมเลตนั้นมีกระบวนการไม่ต่างจากการพัฒนาเฟรมเวอร์กเดิมมากนัก อย่างไรก็ตามในขั้นตอนการวิเคราะห์ของการพัฒนาเฟรมเวอร์กด้วยเฟรมเลตนั้นจะมีขั้นตอนที่แตกต่างจากการพัฒนาเฟรมเวอร์กเดิม เนื่องจากการพัฒนาเฟรมเวอร์กด้วยเฟรมเลตจะมีขั้นตอนในการแบ่งส่วนข้อกำหนดให้มีขนาดเล็กลงเพื่อลดความซับซ้อนของเฟรมเวอร์ก งานวิจัยนี้ได้นำเสนอระเบียบวิธีในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต โดยแบ่งออกเป็น 2 ส่วน คือ ส่วนแรกทำการปรับปรุงรูปแบบคำอธิบายยูสเคสให้เหมาะสมกับการเก็บข้อกำหนดเฟรมเลต และข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต ส่วนที่สองได้ทำการออกแบบขั้นตอนในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต และทำการทดสอบระเบียบวิธีที่ได้โดยสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต โดยใช้ข้อกำหนดโปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล และโปรแกรมประยุกต์เอ็ดิตเตอร์ ในการพิจารณาสร้างเป็นข้อกำหนดเฟรมเวอร์ก จากนั้นจึงทดสอบข้อกำหนดเฟรมเวอร์กที่ได้โดยการสร้างข้อกำหนดโปรแกรมประยุกต์เห็น โปรแกรมประยุกต์มูเดิล โปรแกรมประยุกต์เอ็ดิตเตอร์ และโปรแกรมประยุกต์เลิร์นแอสควร์ สุดท้ายจึงทำการทดสอบโดยการเปรียบเทียบกระบวนการในการสร้างข้อกำหนดเฟรมเวอร์กจากระเบียบวิธีในงานวิจัยนี้ กับกระบวนการสร้างข้อกำหนดเฟรมเวอร์กแบบเดิม ซึ่งประโยชน์จากระเบียบวิธีในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต มีดังนี้

1. ข้อกำหนดเฟรมเวอร์กที่ได้จากระเบียบวิธี มีความซับซ้อนน้อยลงเนื่องจากได้ทำการแบ่งข้อกำหนดเฟรมเวอร์กที่มีขนาดใหญ่ออกเป็นเฟรมเลตย่อยๆ ทำให้แยกทำความเข้าใจในแต่ละเฟรมเลตที่มีขนาดเล็กได้ง่ายกว่าการทำความเข้าใจเฟรมเวอร์กที่มีขนาดใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตที่ได้จากงานวิจัยนี้ สามารถรองรับการเปลี่ยนแปลงได้มากกว่าข้อกำหนดเฟรมเวอร์กเดิม เนื่องจากข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตจากงานวิจัยนี้ได้ทำการแบ่งออกเป็นเฟรมเลตย่อย เมื่อข้อกำหนดมีการเปลี่ยนแปลงมากจนเกินความสามารถของฮอตสปอตต่างๆ ภายในเฟรมเลต ก็สามารถสร้างข้อกำหนดเฟรมเลตใหม่เข้ามาทดแทนเฟรมเลตเดิมได้ โดยถ้าเป็นการสร้างข้อกำหนดเฟรมเวอร์กเดิมนั้นอาจจำเป็นต้องสร้างข้อกำหนดเฟรมเวอร์กใหม่ทั้งระบบ

5.2 ข้อเสนอแนะ

งานวิจัยนี้นำเสนอระเบียบวิธีในการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลต เป็นการสร้างข้อกำหนดเฟรมเวอร์กด้วยเฟรมเลตจากการวิเคราะห์โปรแกรมประยุกต์ที่สัมพันธ์กัน แนวทางหนึ่งในการวิจัยต่อไป คือ การเสนอระเบียบวิธีในการพัฒนาเฟรมเวอร์กด้วยเฟรมเลตในขั้นตอนการออกแบบ ขั้นตอนการพัฒนา และขั้นตอนการทดสอบ เพื่อให้ได้เป็นระเบียบวิธีในการพัฒนาเฟรมเวอร์กด้วยเฟรมเลตที่สมบูรณ์

เอกสารอ้างอิง

- [1] Bertolino A. and Gnesi S. "PLUTO: A Test Methodology for Product Families" [Online], Available : <http://fmt.isti.cnr.it/WEBPAPER/23-BertGnesi.pdf>
- [2] Douglas K., Marc R. and Murray W. "On the creation of pattern languages for framework reuse" [Online], Available : <http://www.cis.strath.ac.uk/research/efocs/papers/EFoCS-49-2002.pdf>
- [3] Dzenan R. "Database Framework Lite (dbfwLite)." [Online], Available : <http://drdb.fsa.ulaval.ca/dbfwLite/doc/dbfwLite.pdf>
- [4] Fayad E.M., Schmidt C.D. and Johnson E.R. **Building Application Frameworks Object-oriented Foundations of Framework Design**. New York : John Wiley & Sons, 1999.
- [5] Fernandez R.M. and Pisano G.J. "MSTORE: Store Management Framework" [Online], Available : http://www.student.luth.se/~javgon-4/MSTORE_Final_Report.pdf
- [6] Froehlich G., Hoover J., Liu L. and Sorenson P. "Designing Object-Oriented Frameworks." [Online], Available : <http://www.cs.ualberta.ca/~softeng/papers/design12.pdf>
- [7] IBM. "Building Object-Oriented Frameworks" [Online], Available : <http://www.ibm.com/java/education/oobuilding/index.html>.
- [8] Intakosum S., Batanov Dentcho N., Boonjing V., Netisopakul P., and Tanyasit W. "Framelet Based Approach for Framework Development." The 8th World Multiconference on Systemics, Cybernetics and Informatics Proceedings. Volume XIV, Page 152-155, Orlando, Florida, USA, July 18-21,2004.
- [9] Krasner G. and Pope S. "A cookbook for using the model-view-controller user interface paradigm in Smalltalk-80." Journal of Object-Oriented Programming, vol. 3,
- [10] Landin N. and Niklasson A. "Development of Object-Oriented Frameworks" Master Thesis, CODEN:LUTEDX(TETS-5231)/1-146, Department of Communication Systems
- [11] Larman C. **Applying UML and Patterns an Introduction to Object-Oriented Analysis and Design and the Unified Process**. NJ: Prentice Hall PTR. 2002.
- [12] Leung tsoi H. "A Management Framework for Software Project Development." [Online], Available : http://www.journal.au.edu/ijcim/may00/holeung_3.pdf.
- [13] Pree W.,Koskimies K. "Framelets-small and loosely coupled frameworks." [Online]. Available : <http://portal.acm.org/citation.cfm?id=351942>. 2000.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [14] Pree W., Koskimies K. "Framelets-small is beautiful." [Online], Available :
<http://portal.acm.org/citation.cfm?id=952532.952747>. 2003.
- [15] Roberts D. and Johnson R. "Evolving Frameworks : A Pattern Language for Developing Object-Oriented Frameworks" [Online], Available :
<http://citeseer.ist.psu.edu/roberts96evolving.html>
- [16] Saeki M. "Reusing Use Case Descriptions for Requirements Specification: Towards Use Case Patterns." APSEC 1999: 309-316
- [17] Schneider G. and Jason P. Winters. Applying Use Case a Practical Guide. MA: Addison-Wesley. 1998.
- [18] Verpoorten H.J. "An application framework for E-learning" [Online], Available :
<http://www.cs.uu.nl/people/janherm/isa2002.pdf>
- [19] Young Y. J., Kim S. Y., Choi G. J., Cho E. S., Kim C. J. and Kim S. D. "A UML-Based Object-Oriented Framework Development Methodology" Proceedings 5th Asia Pacific Software Engineering Conference (APSEC '98), December 2-4, 1998, Taipei, Taiwan, 1998.
- [20] ศรัณย์ อินทโกสุม, แก่นจันทร์ ธรรมรักษ์ "รูปแบบคำอธิบายยูสเคสสำหรับเฟรมเวิร์ก" การประชุมวิชาการ วิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์แห่งชาติ ครั้งที่ 10 (NCSEC 2006)
- [21] ผศ.ดร.ถนอมพร (ต้นพิพัฒน์) เลขาจรัสแสง **Designing e-Learning หลักการออกแบบและการสร้างเว็บเพื่อการเรียนการสอน**
- [22] รศ.ไพโรจน์ ตีระชนากุล, ผศ.ดร.ไพบุสย์ เกียรติโกมล และ เสกสรร เข้มพินิจ **การออกแบบและการผลิต บทเรียนคอมพิวเตอร์การสอน สำหรับ e-Learning**

ประวัติผู้เขียน

ชื่อ-สกุล นายกรกิต จิรสุนัย
 วัน เดือน ปี เกิด 22 กันยายน 2523
 ที่อยู่ 410 ซอยสุทธิพร 2 ถนนประชาสงเคราะห์ แขวงดินแดง เขตดินแดง
 จังหวัดกรุงเทพฯ 10400
 ประวัติการศึกษา 2545 วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีคอมพิวเตอร์ ภาควิชา
 วิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีราชมงคล
 ประสบการทำงาน
 พ.ศ. 2548 อาจารย์พิเศษ สังกัดคณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัย

กรุงเทพ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้