

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

เว็บซีแมนติกข้อมูลกีฬาฟุตบอล

FOOTBALL SEMANTIC WEB



พ.
๒/๖/๒
๒๕๕๐

เลขหมู่.....
เลขทะเบียน..... 04541
วัน,เดือน,ปี. 19 ส.ย. 2551



H004541

b. 11๙ 2A 123
i.....

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเบื้องต้นเท่านั้น ไม่สามารถให้นำไปใช้ประโยชน์ด้านการค้า
ภาคเรียนที่ 1 ปีการศึกษา 2550
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FOOTBALL SEMANTIC WEB



**A SYSTEM DEVELOPMENT PROJECT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECNOLOGY**

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
1/ 2007
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2007

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการเรียนการสอนเท่านั้น การคัดลอกหรือเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	เว็บซีแมนติกข้อมูลกีฬาฟุตบอล
นักศึกษา	นางสาวปนัดดา รูปแพ
รหัสนักศึกษา	47066646
ปริญญา	วิทยาศาสตร์มหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2550
อาจารย์ที่ปรึกษา	ผศ.ดร.พรฤดี เนติโสภากุล

บทคัดย่อ

โครงการพัฒนาระบบงานฉบับนี้เสนอวิธีการค้นหาข้อมูลโดยใช้ OWL (Web Ontology Language) ในการกำหนดรายละเอียดของข้อมูล (Metadata) ซึ่งรวมถึงความสัมพันธ์ระหว่างข้อมูลใน OWL โดยลักษณะเด่นของวิธีการที่นำเสนอในการพัฒนาระบบงานนี้คือ การกำหนดรายละเอียดของข้อมูลไว้ใน OWL เพื่อสามารถค้นหาข้อมูลได้ดีขึ้นและถูกต้องตามความต้องการมากขึ้น ในโครงการพัฒนาระบบงานเล่มนี้จะใช้ SPARQL Query Language ในการค้นหาข้อมูลที่กำหนดไว้ใน OWL คล้ายกับการค้นหาจากฐานข้อมูลทั่วไป ซึ่งในโครงการพัฒนาระบบงานเล่มนี้ได้แสดงผลของการค้นหาแบบที่ไม่ใช้ OWL ในการค้นหาด้วย เพื่อเปรียบเทียบวิธีการค้นหาแบบเดิม กับการค้นหาโดยที่มี OWL

Title	Football Semantic Web
Student	Ms.Panadda Thooppae
Student ID.	47066646
Degree	Master of Science
Programme	Information Science
Academic Year	2007
Advisor	Assist. Prof. Dr. Pornrudee Netisopakul

ABSTRACT

This project proposes a channel searching method of using OWL (Web Ontology Language) for define data of data (Metadata) and relationship among the data. The salient feature of proposed method is define OWL, which more efficiency of searching and more accuracy of searching. This project implements the SPARQL Query Language to search data in OWL, similar to searching data in normal database. This project presents the various searching to verify the proposed method as comparing with that for the OWL assist searching.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร. พรฤดี เนติโสภาคกุล ซึ่งเป็นอาจารย์ผู้ควบคุมโครงการพัฒนาระบบงาน, ผศ.ดร. ภัทรชัย สถิตโรจน์วงศ์ และ ดร. นล เปรมชัยเรูียร ข้าพเจ้ารู้สึกทราบบซึ่งในความอนุเคราะห์จากท่านอาจารย์ทั้งสามท่าน และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ภาควิชาวิทยาการสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิทยาการสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะเทคโนโลยีสารสนเทศที่ให้ความช่วยเหลือ ในเรื่องต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำโครงการพัฒนาระบบฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากโครงการพัฒนาระบบงานฉบับนี้ ข้าพเจ้าขอบแต่ผู้มีพระคุณทุกท่าน

ปนัดดา ชูปแพ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 สมมติฐานของการศึกษา.....	1
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย.....	2
1.5 การเปรียบเทียบการค้นหาระหว่างวิธีการที่ใช้ OWL และไม่ใช่ OWL.....	2
1.6 ขอบเขตการวิจัย.....	2
1.7 ขั้นตอนการศึกษา.....	2
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในการพัฒนาระบบงาน.....	4
2.1 เทคนิค OWL (Web Ontology Language).....	4
2.1.1 Class Element.....	4
2.1.2 Property Element.....	4
2.1.3 Property Restriction.....	5
2.1.4 Special Properties.....	5
2.1.5 Boolean Combinations.....	5
2.1.6 Enumerations.....	6
2.1.7 Instances.....	6
2.1.8 Data Types.....	6
2.2 XML Query Language (Xquery).....	6
2.2.1 ภายนอกกระบวนการค้นคืน (External Processing).....	6
2.2.2 กระบวนการค้นคืน (Query Processing).....	7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

สารบัญ (ต่อ)

	หน้า
2.3 SPARQL Query Language.....	9
2.4 Jena’s Framework.....	10
2.5 The Protégé-OWL Plugin and CO-ODE Tools.....	18
บทที่ 3 การวิเคราะห์และออกแบบซีแมนติกเว็บเนื้อหากีฬาฟุตบอล.....	29
3.1 การวิเคราะห์เนื้อหากีฬาฟุตบอล.....	29
3.2 การออกแบบ OWL (Web Ontology Language).....	33
3.3 การออกแบบโครงสร้างสถาปัตยกรรมของซีแมนติกเว็บที่มีเนื้อหากีฬาฟุตบอล..	62
บทที่ 4 ทดสอบการค้นหาข้อมูลของซีแมนติกเว็บเนื้อหากีฬาฟุตบอล.....	70
4.1 การค้นหาข้อมูลทั่วไป.....	70
4.2 การค้นหาข้อมูลซีแมนติกเว็บ.....	70
4.3 การทดสอบการค้นหาข้อมูลซีแมนติกเว็บ.....	71
4.4 สรุปผลการค้นหา.....	79
บทที่ 5 บทสรุปโครงการพัฒนาระบบ.....	80
บรรณานุกรม.....	81
ประวัติผู้เขียน.....	82

สารบัญรูป

รูปที่	หน้า
2.1 แสดงภาพรวมกระบวนการค้นหา.....	7
2.2 แสดงการค้นหาข้อมูลแบบง่ายด้วย SPARQL.....	9
2.3 แสดงการค้นหาข้อมูลที่ได้ผลการค้นหาเป็นชุดข้อมูลด้วย SPARQL.....	10
2.4 แสดงการสร้างโปรเจกใหม่ใน Protégé	19
2.5 แสดงการเลือกลักษณะของโปรเจกใหม่ใน Protégé	19
2.6 แสดงตัวเลือกให้ทำการนำเข้าไฟล์ OWL สำหรับโปรเจกใหม่ใน Protégé	20
2.7 แสดงการกำหนดชื่อไฟล์ OWL สำหรับโปรเจกใหม่ใน Protégé	20
2.8 แสดงการเลือกประเภทของ OWL ของโปรเจกใหม่ใน Protégé	21
2.9 แสดงการเลือกมุมมอง OWL สำหรับโปรเจกใหม่ใน Protégé	21
2.10 แสดงการสิ้นสุดเมื่อสร้างโปรเจกใหม่ใน Protégé สำเร็จ.....	22
2.11 แสดงการเลือกเมนูเพื่อกำหนดค่าให้โปรเจกใหม่ใน Protégé	23
2.12 แสดงการกำหนดค่าเพื่อแสดงรูปภาพ OWL ของโปรเจกใหม่ใน Protégé	23
2.13 แสดงหน้ารูปภาพ OWL ของโปรเจกใหม่ใน Protégé	24
2.14 แสดงการเลือกเมนูเพื่อทำการบันทึกโปรเจกใหม่ใน Protégé	24
2.15 แสดงการกำหนดชื่อเพื่อทำการบันทึกโปรเจกใหม่ใน Protégé	25
2.16 แสดงไฟล์โปรเจกและไฟล์ OWL เมื่อทำการบันทึกโปรเจกใหม่ใน Protégé	26
2.17 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างคลาสใน Protégé	26
2.18 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างข้อจำกัดของคลาสใน Protégé	27
2.19 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างความสัมพันธ์ (Property) ใน Protégé	28
2.20 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างคลาสที่ไม่เกี่ยวข้องกัน ใน Protégé	28
3.1 OWL แสดงข้อมูลเว็บซีแมนติกที่เป็นเนื้อหาของกีฬาฟุตบอล.....	33
3.2 แสดง OWL ของข้อมูลคลาส Stadium.....	34
3.3 แสดง OWL ของคลาส Stadium และคลาส TownStadium.....	34
3.4 แสดง OWL ของคลาส ClubStadium.....	35
3.5 แสดง OWL ความสัมพันธ์ของคลาส ClubStadium.....	35
3.6 แสดงความสัมพันธ์โดยรวมของของคลาส Stadium.....	35
3.7 แสดง OWL ของคลาส MatchPlay.....	36
3.8 แสดง OWL ความสัมพันธ์ของคลาส MathPlay.....	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.9 แสดง OWL ของคลาส FaPremierLeague.....	37
3.10 แสดง OWL ของคลาส TheFootballLeagueCup.....	37
3.11 แสดง OWL ของคลาส TheFaCup.....	38
3.12 แสดงความสัมพันธ์โดยรวมของของคลาส MatchPlay.....	38
3.13 แสดง OWL ของคลาส Results.....	39
3.14 แสดง OWL ของคลาส Results.....	39
3.15 แสดง OWL ของคลาส PenaltyKick.....	39
3.16 แสดง OWL ของคลาส Goal.....	39
3.17 แสดง OWL ของคลาส FoulPlay.....	40
3.18 แสดง OWL ของคลาส TurnPlayer.....	40
3.19 แสดงความสัมพันธ์โดยรวมของของคลาส Results.....	41
3.20 แสดง OWL ของคลาส Teamwork.....	42
3.21 แสดง OWL ของคลาส Teamwork.....	42
3.22 แสดง OWL ของคลาส Coach.....	43
3.23 แสดง OWL ของคลาส Manager.....	43
3.24 แสดง OWL ของคลาส Player.....	43
3.25 แสดง OWL ของคลาส ForwardPlayer.....	44
3.26 แสดง OWL ของคลาส MidfieldPlayer.....	45
3.27 แสดง OWL ของคลาส GoalkeeperPlayer.....	46
3.28 แสดง OWL ของคลาส DefensivePlayer.....	47
3.29 แสดงความสัมพันธ์โดยรวมของของคลาส Teamwork.....	48
3.30 แสดง OWL ของคลาส Score.....	48
3.31 แสดง OWL ของคลาส Score.....	49
3.32 แสดงความสัมพันธ์โดยรวมของของคลาส Score.....	49
3.33 แสดง OWL ของคลาส TeamFootball.....	50
3.34 แสดง OWL ของคลาส TeamFootball.....	50
3.35 แสดง OWL ของคลาส TeamFootball.....	51
3.36 แสดง OWL ของคลาส TeamFootball.....	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญรูป (ต่อ)

รูปที่	หน้า
3.37 แสดง OWL ของคลาส TeamFootball.....	51
3.38 แสดง OWL ของคลาส PremierTeam.....	52
3.39 แสดง OWL ของคลาส ChampionshipTeam.....	53
3.40 แสดง OWL ของคลาส LeagueOneTeam.....	53
3.41 แสดง OWL ของคลาส LeagueTwoTeam.....	54
3.42 แสดงความสัมพันธ์โดยรวมของของคลาส Team Football.....	55
3.43 แสดง OWL ของคลาส Club.....	56
3.44 แสดง OWL ของคลาส Club.....	56
3.45 แสดงความสัมพันธ์โดยรวมของของคลาส Club.....	57
3.46 แสดง OWL ของคลาส TeamFootballDivision.....	57
3.47 แสดงความสัมพันธ์โดยรวมของของคลาส TeamFootball Division.....	58
3.48 แสดง OWL ของคลาส PlayerPosition.....	59
3.49 แสดง OWL ของคลาส PlayerPosition.....	59
3.50 แสดงความสัมพันธ์โดยรวมของของคลาส PlayerPosition.....	60
3.51 แสดง OWL ของคลาส PlayerPosition.....	61
3.52 แสดงความสัมพันธ์โดยรวมของของคลาส ShortNamePosition.....	61
3.53 แสดง โครงสร้างสถาปัตยกรรม.....	62
3.54 แสดงภาพรวมในเชิงเทคนิคของส่วนที่ใช้ในการค้นหาข้อมูล.....	63
3.55 แสดงคลาสที่ใช้ในการติดต่อกับฐานข้อมูลและ OWL.....	64
3.56 แสดงส่วนของโปรแกรมที่ทำการอ่านไฟล์ OWL มาเก็บไว้ในหน่วยความจำ.....	65
3.57 แสดงส่วนของโปรแกรมที่เข้าไปค้นหาใน OWL.....	65
3.58 แสดงคลาสที่ใช้ในการค้นหาจาก OWL มาประมวลผล.....	66
3.59 แสดงส่วนของคลาสที่นิยามค่าต่าง ๆ เช่นความสัมพันธ์ใน Ontology.....	66
3.60 แสดง QueryString ที่ใช้ในการค้นหาข้อมูลที่เกี่ยวข้อง ของนักฟุตบอลที่มีชื่อเล่นตามที่กำหนด.....	67
3.61 แสดง QueryString ที่ใช้ในการค้นหาข้อมูลทุกอย่างที่เกี่ยวข้องกับสโมสรฟุตบอลที่มีชื่อเล่น ตามที่กำหนด.....	67

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.62 แสดงคลาสต่าง ๆ ที่นำเอาเงื่อนไขหรือคำแนะนำจาก Ontology ไปค้นหาข้อมูลหรือ ไฟล์ HTML.....	68
4.1 แสดงหน้าจอหลักของซีแมนติกเว็บเนื้อหากีฬาฟุตบอล.....	70
4.2 แสดงหน้าจอเมื่อกรอกคำค้น.....	71
4.3 แสดงหน้าผลการค้นหาเมื่อไม่ใช้ OWL จากคำค้น Attacking-midfielder.....	72
4.4 แสดงหน้าผลการค้นหาเมื่อใช้ OWL จากคำค้น Attacking-midfielder.....	73
4.5 แสดงหน้าผลการค้นหาเมื่อไม่ใช้ OWL จากคำค้น Baby Gold.....	74
4.6 แสดงหน้าผลการค้นหาเมื่อใช้ OWL จากคำค้น Baby Gold.....	75
4.7 แสดงหน้าผลการค้นหาเมื่อกรอกคำค้นสโมสร.....	76
4.8 แสดงหน้าจอเมื่อกรอกคำค้นสโมสรจากชื่ออื่นของสโมสร.....	77
4.9 แสดงหน้าผลการค้นหาเมื่อกรอกคำค้นชื่ออื่นของสโมสร.....	78

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การตอบสนองการใช้งาน internet ในปัจจุบันที่มีข้อมูลอยู่อย่างมากมาย ทำให้ต้องการความรวดเร็วในการค้นหา รวมทั้งความถูกต้องในการค้นหามากยิ่งขึ้น ดังนั้นเพื่อค้นหาข้อมูลให้ได้เนื้อหาตรงกับความต้องการมากยิ่งขึ้นจากปัจจุบัน เราสามารถออกแบบและพัฒนาเว็บให้เป็นซีแมนติกเว็บ

ซีแมนติกเว็บ คือ เว็บที่สามารถตีความหมายได้ว่าเป็นเว็บที่มีเนื้อหาของอะไร เครื่องคอมพิวเตอร์สามารถเข้าใจเนื้อหาของเว็บนั้นๆได้

เว็บมากมายที่เห็นในปัจจุบันนั้น เครื่องคอมพิวเตอร์ไม่สามารถรู้และเข้าใจได้ว่า แต่ละเว็บนั้นเป็นเว็บที่มีเนื้อหาเกี่ยวข้องกับอะไร ในการค้นหาเว็บที่ต้องการนั้น เมื่อระบุคำค้นแล้วคอมพิวเตอร์ก็ทำการค้นหา โดยนำคำค้นไปค้นหาจากข้อความเนื้อหาของเว็บว่า เว็บนั้นๆพบคำที่เป็นคำค้นอยู่หรือไม่ ถ้าพบแสดงว่าตรงกับความต้องการในการค้นหา ซึ่งจากเว็บที่ค้นออกมาได้นั้น จะพบว่าหลายๆเว็บในจำนวนนั้นมีเนื้อหาที่ไม่ตรงกับความต้องการจริงๆ ถึงแม้จะมีคำค้นนั้นอยู่ในเนื้อหาของเว็บก็ตาม ซึ่งกว่าที่จะรู้ว่าเนื้อหาไม่ตรงกับที่ต้องการนั้น ผู้สืบค้นต้องเข้าไปอ่านเนื้อหาเหล่านั้นเอง ทำให้สูญเสียเวลาไปอย่างมาก

ดังนั้นเมื่อพัฒนาเว็บให้เป็นซีแมนติกเว็บแล้ว จะช่วยให้เครื่องคอมพิวเตอร์สามารถทำการค้นหาได้ตรงกับคำค้นที่ต้องการมากขึ้น จากความสามารถที่เครื่องคอมพิวเตอร์สามารถเข้าใจเนื้อหาของเว็บนั้นๆนั่นเอง

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

โครงการพัฒนาระบบงานฉบับนี้มุ่งหวังเพื่อพัฒนาซีแมนติกเว็บที่เป็นเนื้อหากีฬาฟุตบอล เพื่อแสดงการค้นหาข้อมูลในเว็บได้ตรงกับความต้องการของผู้สืบค้นได้มากยิ่งขึ้น ซึ่งซีแมนติกเว็บ เป็นเทคโนโลยีที่ก้าวเข้าสู่ยุคที่สาม ซึ่งเป็นยุครายละเอียดของข้อมูล (Metadata) เพื่อจัดการข้อมูลมากมายที่เพิ่มขึ้นทุกวัน

1.3 สมมติฐานของการศึกษา

ข้อดีของการค้นหาข้อมูลในปัจจุบัน คือการนำคำค้นที่ผู้สืบค้นต้องการ ไปค้นในเนื้อหาของเว็บเท่านั้นเอง ไม่ได้ค้นหาจากความหมายของเนื้อหาจริงๆ ดังนั้นความถูกต้องของการค้นหา

เพื่อให้ตรงกับความต้องการนั้น จึงยังไม่ตรงกับความต้องการจริงๆ เนื่องจากอาจจะค้นพบคำๆนั้น ในเว็บแต่ไม่ได้เป็นเรื่องที่ต้องการค้นหาเลย

การแก้ปัญหาข้างต้นนี้ ถ้าหากเว็บใดๆมีการสร้างรายละเอียดของข้อมูลเพื่อนำมาใช้ในการ ค้นหา ก็จะสามารถค้นหาข้อมูลได้ตรงกับความต้องการได้มากยิ่งขึ้นนั่นเอง ซึ่งในโปรเจกต์นี้จะใช้ OWL (Web Ontology Language) ในการกำหนดรายละเอียดและความสัมพันธ์ของข้อมูล

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

วิธีการกำหนดรายละเอียดและความสัมพันธ์ของข้อมูลเลือกใช้ OWL (Web Ontology Language) ซึ่งก็คือการใช้เทคโนโลยีโครงสร้าง XML (Extension Markup Language) เป็นพื้นฐาน ในรายละเอียดการใช้และหลักการของ OWL จะกล่าวต่อไปในบทที่ 2

เนื่องจากการใช้ OWL ซึ่งมีโครงสร้างเป็น XML มาใช้กำหนดรายละเอียดและความสัมพันธ์ของข้อมูลแล้ว การเข้าถึงรายละเอียดและความสัมพันธ์ดังกล่าวจึงต้องใช้ภาษาในการค้นคืนข้อมูลสำหรับ XML โดยเฉพาะ โปรเจกต์นี้ได้เลือกใช้ SPARQL Query Language ซึ่งมีหลักการและรายละเอียดการใช้ในบทที่ 2 เช่นกัน

1.5 การเปรียบเทียบการค้นหาระหว่างวิธีการที่ใช้ OWL และไม่ใช่ OWL

วิธีการค้นหาข้อมูลโดยใช้ OWL (Web Ontology Language) เมื่อเทียบกับการค้นหาที่ไม่ใช่ OWL แล้ว การค้นหาข้อมูลโดยใช้ OWL นั้น จะให้ผลการค้นหาข้อมูลได้มีความรวดเร็วและถูกต้องในความหมายของคำค้นมากกว่าวิธีการค้นหาข้อมูลแบบไม่ใช่ OWL

1.6 ขอบเขตการวิจัย

ในโครงการพัฒนาระบบงานฉบับนี้ได้พัฒนาซีเมนติกเว็บที่เนื้อหาเป็นกีฬาฟุตบอลโดยใช้ OWL (Web Ontology Language) เพื่อแสดงการค้นหาข้อมูลได้ตรงกับความต้องการ โดยเนื้อหา กีฬาฟุตบอลจะเก็บข้อมูลเฉพาะทีมพรีเมียร์ลีกประเทศอังกฤษเท่านั้น

1.7 ขั้นตอนของการศึกษา

โครงการพัฒนาระบบงานฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของโครงการพัฒนาระบบงาน ความมุ่งหมายและ

วัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในโครงการพัฒนาระบบงาน และพื้นฐานของซีแมนติกเวบ ซึ่งประกอบด้วยเทคนิคการกำหนด OWL (Web Ontology Language) และภาษาในการค้นหารายละเอียดข้อมูลจาก XML ที่ใช้ในโปรเจกต์ SPARQL Query Language

บทที่ 3 กล่าวถึงการวิเคราะห์รายละเอียดของกัฬาฟุตบอล และการออกแบบซีแมนติกเวบที่มีเนื้อหากีฬาฟุตบอล OWL

บทที่ 4 แสดงการค้นหาข้อมูลของซีแมนติกเวบเนื้อหากีฬาฟุตบอล เพื่อแสดงให้เห็นว่าสามารถค้นหาได้ถูกต้องตามความต้องการได้ถูกต้องยิ่งขึ้น โดยการนำ Jena's Framework มาช่วยในการพัฒนาระบบงาน

บทที่ 5 เป็นบทสรุปโครงการพัฒนาระบบงาน



บทที่ 2

ทฤษฎีพื้นฐานและเครื่องมือที่ใช้ในการพัฒนาระบบงาน

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานและเครื่องมือต่างๆ ที่เกี่ยวข้องในการพัฒนาระบบงาน และโครงสร้างพื้นฐานของซีเมนติกเว็บ ซึ่งเนื้อหาทั้งหมดนี้จำเป็นสำหรับการพัฒนาระบบงาน

2.1 เทคนิค OWL (Web Ontology Language)

ซีเมนติกเว็บ มีพื้นฐานการพัฒนาโดยการใช้โครงสร้าง XML (Extension Markup Language) เพื่อถ่ายทอดความเข้าใจของผู้ใช้ที่เป็นคนทั่วไป โดยสามารถใช้ RDF (Resource Description Language) หรือ OWL (Web Ontology Language) ซึ่งในโครงการพัฒนาระบบงานนี้ ได้เลือกใช้ OWL ในการพัฒนาซีเมนติกเว็บเนื้อหาที่หาฟูลบอล ซึ่งมีหลักการดังต่อไปนี้

2.1.1 Class Element

2.1.1.1 Sub class

คือการกำหนดคลาสย่อยของคลาสใดๆ

2.1.1.2 Disjoint class

คือการกำหนดให้คลาสแยกจากกัน

2.1.1.3 Equivalence of class

คือการกำหนดให้คลาสใดๆมีความเหมือนกัน

2.1.2 Property Elements

2.1.2.1 Object properties

คือส่วนที่เป็นความสัมพันธ์ของคลาส ซึ่งจะมีค่าต่างๆ ดังต่อไปนี้

2.1.2.1.1 Domain

คือขอบเขตค่าของคลาสใดๆ

2.1.2.1.2 Range

คือขอบเขตค่าของคลาสใดๆ

2.1.2.1.3 Inverse properties

คือความสัมพันธ์แบบย้อนกลับ

2.1.2.1.4 Sub property

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ **คือ property ย่อยของ property ใดๆ** อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น **2.1.2.2 Data type properties** และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือส่วนที่เป็นข้อมูลของคลาสใดๆ

2.1.2.3 Equivalence of properties

คือ property ที่มีความเหมือนกัน

2.1.3 Property Restrictions

2.1.3.1 All value from

คือการกำหนดให้ค่าของ property มาจากคลาสใดๆ เท่านั้น

2.1.3.2 Has value

คือการกำหนดค่าให้ property ว่ามีค่าเป็นอะไรได้บ้าง

2.1.3.3 Some value from

คือการกำหนดให้ค่าเป็นค่าที่มาจากคลาสใดๆ ได้บ้าง

2.1.3.4 Min cardinality

คือการกำหนดค่าต่ำสุดให้ property นั้นๆ

2.1.3.5 Max cardinality

คือการกำหนดค่าสูงสุดให้ property นั้นๆ

2.1.3.6 Cardinality

คือการกำหนดค่าต่ำสุดและสูงสุดเป็นค่าเดียวกันให้ property นั้นๆ

2.1.4 Special Properties

2.1.4.1 Transitive Property

คือการกำหนดให้ property มีความสัมพันธ์โดยตรงผ่านความสัมพันธ์ตรงกลางได้ เช่น เด็กหญิงมานีเป็นเพื่อนเด็กชายมานะ และเด็กชายมานะเป็นเพื่อนเด็กหญิงชูใจ ดังนั้นความสัมพันธ์ผ่านจะเป็น เด็กหญิงมานีเป็นเพื่อนเด็กหญิงชูใจ เป็นต้น

2.1.4.2 Symmetric Property

คือการกำหนดให้ property มีความสมดุลให้มีผลต่อกันเหมือนกัน เช่น เด็กหญิงมานีเป็นเพื่อนเด็กชายมานะ ดังนั้นความสัมพันธ์ที่สมดุลกันจะเป็น เด็กชายมานะเป็นเพื่อนเด็กหญิงมานี เป็นต้น

2.1.4.3 Functional Property

คือการกำหนดให้มียังมากแต่ค่าเดียวที่จะมีความสัมพันธ์กัน เช่น เด็กหญิงมานีเป็นเพื่อนเด็กชายมานะ เป็นต้น ก็คือจะเป็นเด็กชายมานะคนนี้เป็นเพื่อนกัน เด็กหญิงมานี ความสัมพันธ์แบบนี้เรียกได้อีกอย่างว่า Single Valued Properties

2.1.4.4 Inverse Functional Property

คือการกำหนดความสัมพันธ์ย้อนกลับ เช่น เด็กหญิงมานีเป็นเพื่อนเด็กชายมานะ

ดังนั้นความสัมพันธ์ย้อนกลับคือ เด็กชายมานะมีเพื่อนเป็นเด็กหญิงมานี เป็นต้น

2.1.5 Boolean Combination

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการเรียนการสอนเท่านั้น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5.1 Union

คือค่าที่เกิดจากการรวมกันของคลาสใดๆ

2.1.5.2 Intersection

คือค่าที่เกิดจากการที่มีบางส่วนของคลาสใดๆที่เกี่ยวข้องกัน

2.1.5.3 Complement

คือค่าที่นอกเหนือจากคลาสใดๆ

2.1.6 Enumerations

คือการกำหนดค่าเจาะจงให้คลาสใดๆ

2.1.7 Instances

คือการสร้างตัวอย่างจากคลาสใดๆ

2.1.8 Data Types

คือตัวแปรของคลาสใดๆ

2.2 XML Query Language (Xquery)

XQuery เป็นภาษาที่เหมาะสมจะใช้ในการค้นหาข้อมูลที่จัดเก็บในรูปแบบของ XML ซึ่งนำมาใช้ในการพัฒนาเว็บไซต์เว็บ กระทบการประมวลผลของ XQuery แสดงดังรูปที่ 2.1

จากรูปที่ 2.1 ขอบเขตของการประมวลผลจะอยู่ในกรอบสี่เหลี่ยม จึงขอแบ่งออกเป็น 2 ส่วน คือ ภายนอกกระบวนการค้นคืน (External Processing) และกระบวนการค้นคืน (Query Processing)

2.2.1 ภายนอกกระบวนการค้นคืน (External Processing)

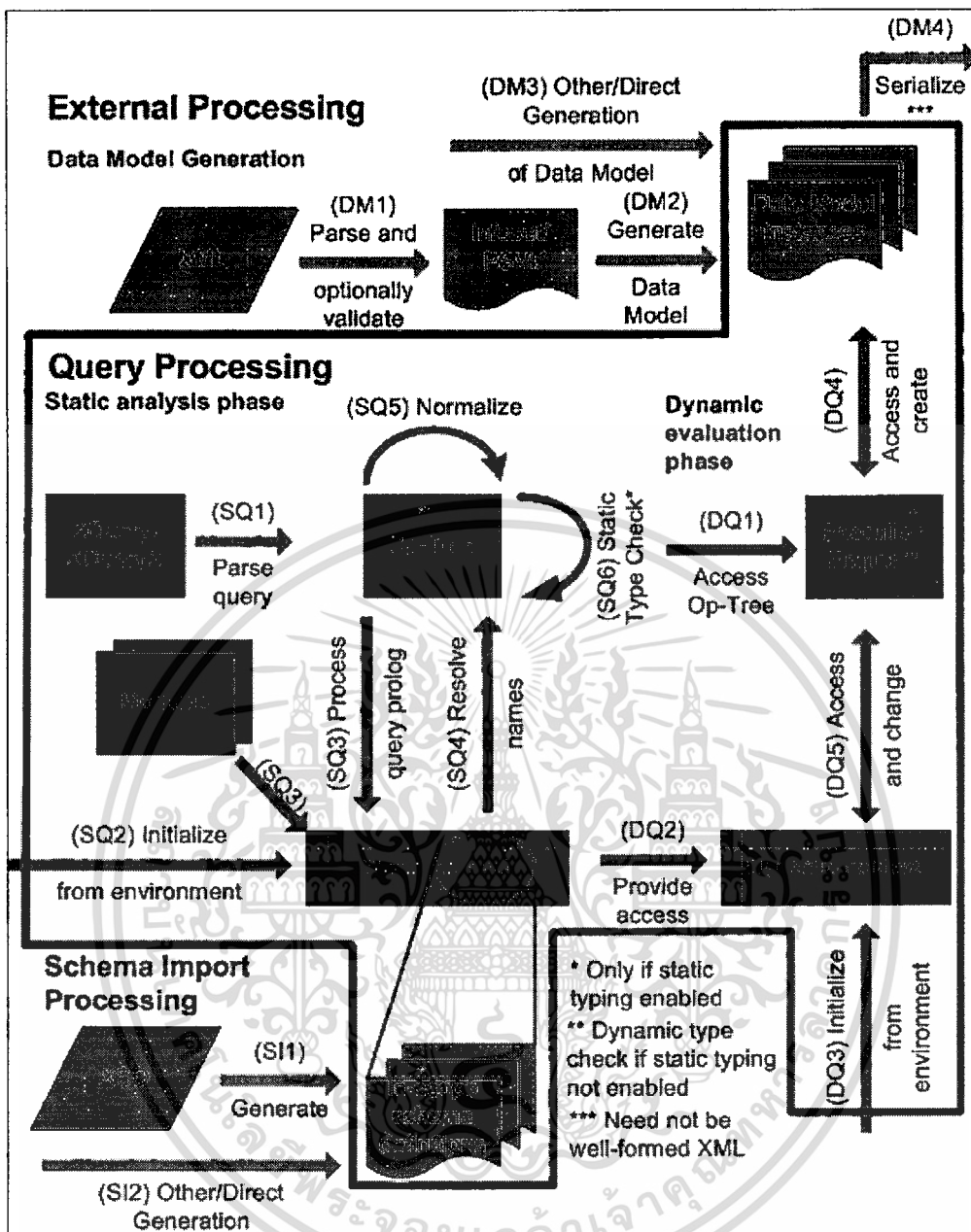
เป็นส่วนที่จะต้องทำก่อนจะทำกระบวนการค้นหา คือการสร้าง XDM (XQuery/XPath Data Model) เป็นส่วนข้อมูลที่จะใช้ในการค้นคืน, กำหนด Constraint จากนั้นจะต้องทำการสร้าง Serialization Parameters ซึ่งแบ่งเป็น 2 ส่วน คือ

1. กระบวนการสร้างข้อมูลที่จะใช้ในการค้นคืน (Data Model Generation) โดยการใช้ XML parser ทำการแปลง XML ให้เป็นข้อมูลที่มีโครงสร้างแบบ PSVI (Post-Schema Validation Infoset) จากนั้นใช้กระบวนการ XDM แปลงเป็น XDM instance ซึ่งก็คือ Data Model Instance ในรูปที่ 2.1 และหากจะมีการทำ Serialization ก็จะทำ Data Model ในส่วนนี้ไปทำต่อซึ่ง XQuery ก็ไม่ได้กำหนดให้ทำ Serialization

2. กระบวนการนำเข้าโครงสร้าง (Schema Import Processing) จะทำการนำเข้า XML หรือ XSD (XML Schema Definition) เป็น In-scope schema definitions ดังรูปที่ 2.1 เพื่อนำไปใช้ใน Static Context

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงภาพรวมกระบวนการค้นหา [1]

2.2.2 กระบวนการค้นคืน (Query Processing)

เป็นส่วนที่จะทำการค้นคืน โดยจะต้องมีการทำการวิเคราะห์คำสั่ง XQuery ทั้ง Static Analysis Phase และ Dynamic Evaluation Phase ซึ่งอาจจะมีข้อผิดพลาดเกิดขึ้นเป็น Static Errors, Dynamic Errors หรือ Type Errors รวมทั้งจะมีการตรวจสอบ Consistency Constraints ดังรายละเอียดต่อไปนี้

1. Static Analysis Phase คือกระบวนการวิเคราะห์ XQuery ซึ่งจะยังไม่เกี่ยวข้องกับข้อมูล เป็นการวิเคราะห์ก่อนที่จะนำไปทำการคำนวณ จากรูปที่ 2.1 จะมีขั้นตอน SQ1-SQ6 ดังรายละเอียดต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- i. SQ1 จะทำการแปลง XQuery เป็น Op-Tree (Operation Tree) ในการแปลงเป็น Op-Tree นี้อาจจะเกิด Static Errors ขึ้น
- ii. SQ2 จาก Initialization จะได้เป็น Static Context
- iii. SQ3 หากขั้นตอน SQ1 ไม่มีข้อผิดพลาดก็จะถูกเปลี่ยนเป็น Static Context ด้วย Prolog
- iv. SQ4 จากนั้นหาก Static Context มีกระบวนการนำเข้าโครงสร้าง ก็จะมี In-scope Schema Definitions และหากมีการนำเข้า การประกาศฟังก์ชัน (Function Declaration) และการประกาศตัวแปร (Variable Declaration) ไว้ด้วยก็จะต้องทำการแปลงประเภทชื่อ (Type Names), ชื่อฟังก์ชัน (Function Names), คำนำหน้าชื่อ (Name-space Prefixes) และชื่อตัวแปร (Variable Names) เป็น Op-Tree ถ้าหากชื่อใน Op-Tree ไม่ตรงกับใน Static Context ก็จะเกิดข้อผิดพลาด Static Errors ขึ้น
- v. SQ5 จากนั้นจะทำการ Normalize ค่าคำสั่งต่างๆรวมทั้งค่าที่จริงต่างๆ ด้วย
- vi. SQ6 กรณีที่มี Static Typing Feature จะทำการตรวจสอบประเภทตัวแปรตาม Static Type

2. Dynamic Evaluation Phase คือกระบวนการหาค่า XQuery หลังจากทำการวิเคราะห์เสร็จเรียบร้อยแล้ว ไม่มีข้อผิดพลาดของ Static Errors เกิดขึ้นเลย จากรูปที่ 2.1 จะมีขั้นตอน DQ1-DQ5 ดังรายละเอียดต่อไปนี้

- i. DQ1 จะทำการดึงค่าจาก Op-Tree มาทำการคำนวณหาค่าที่ Execution Engine หากในขั้นตอนการวิเคราะห์ (Static Analysis Phase) ไม่มี Static Typing Feature ในขั้นตอนนี้ ก็จะทำการตรวจสอบประเภทตัวแปรต่างๆตาม Dynamic Type ด้วย
- ii. DQ2 หาก Dynamic Context ต้องใช้ค่าจาก Static Context ก็จะมีการดึงค่าจาก Static Context มาใช้
- iii. DQ3 ค่าใน Dynamic Context สามารถกำหนดได้จากการ Initialize จากภายนอกได้ด้วย
- iv. DQ4 จะทำการดึงค่าข้อมูลจาก Data Model Instances มาใช้ในการคำนวณที่ Execution Engine เมื่อต้องการข้อมูล และหากมีผลจากการคำนวณที่เปลี่ยนค่าข้อมูลก็จะทำการส่งค่ากลับไป
- v. DQ5 จะทำการดึงค่าจาก Dynamic Context ไปใช้ในการคำนวณที่ Execution Engine และหากมีผลจากการคำนวณที่เปลี่ยนค่าก็จะทำการส่งค่ากลับ เช่นเดียวกับขั้นตอน DQ4

2.3 SPARQL Query Language

SPARQL เป็นภาษาที่ใช้ในการค้นหาข้อมูลที่จัดเก็บในรูปแบบของ XML ในโครงการพัฒนาระบบซีแมนติกเว็บฟุตบอล โดยมีหลักการพื้นฐานดังต่อไปนี้

พื้นฐานการค้นหาข้อมูลจาก OWL ดังรูปที่ 2.2 แบ่งเป็น 3 ส่วน คือ ส่วนของข้อมูลที่มีอยู่ จากนั้นเป็นส่วนแสดงคำสั่งการค้นหาข้อมูล และสุดท้ายเป็นส่วนแสดงผลการค้นหา

Data:

```
<http://example.org/book/book1> <http://purl.org/dc/elements/1.1/title> "SPARQL Tutorial" .
```

Query:

```
SELECT ?title
WHERE
{
  <http://example.org/book/book1> <http://purl.org/dc/elements/1.1/title> ?title .
}
```

Query Result:

title
"SPARQL Tutorial"

รูปที่ 2.2 แสดงการค้นหาข้อมูลแบบง่าย [2]

จากการค้นหาข้อมูลแบบง่าย เพื่อได้ผลการค้นหาเฉพาะข้อมูลใดๆ เราสามารถค้นหาข้อมูลที่ได้ผลการค้นหาเป็นชุดของข้อมูลได้ดังรูปที่ 2.3 จากรูปแบ่งการแสดงผลไว้ 3 ส่วนเช่นกัน ส่วนแรกเป็นข้อมูลที่มีอยู่ใน OWL จากนั้นแสดงคำสั่งในการค้นหาข้อมูลจาก OWL และสุดท้ายแสดงผลการค้นหา ซึ่งได้ผลเป็นชุดของข้อมูล

Data:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
_:a foaf:name "Johnny Lee Outlaw" .
_:a foaf:mbox <mailto:jlow@example.com> .
_:b foaf:name "Peter Goodguy" .
_:b foaf:mbox <mailto:peter@example.org> .
_:c foaf:mbox <mailto:carol@example.org> .
```

Query:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?mbox
WHERE
{ ?x foaf:name ?name .
  ?x foaf:mbox ?mbox }
```

Query Result:

name	mbox
"Johnny Lee Outlaw"	<mailto:jlow@example.com>
"Peter Goodguy"	<mailto:peter@example.org>

รูปที่ 2.3 แสดงการค้นหาข้อมูลที่ได้ผลการค้นหาเป็นชุดข้อมูลด้วย SPARQL [2]

2.4 Jena's Framework

Jena เป็น Framework สำหรับพัฒนาโปรแกรมประยุกต์สำหรับ Semantic Web โดยได้ถูกพัฒนาโดย HP Labs Semantic Web Research ในลักษณะของโอเพ่นซอสด้วยภาษา Java Jena ได้จัดเตรียมสภาพแวดล้อมสำหรับการพัฒนาโปรแกรมเพื่อติดต่อกับ RDF, RDFS และ OWL สนับสนุนการค้นหาข้อมูลใน OWL ด้วยภาษา SPARQL และสนับสนุนการวิเคราะห์ตามกฎที่จัดตั้งไว้

สำหรับ Framework Jena นั้นจะประกอบไปด้วย

1. A RDF API
2. Reading and writing RDF in RDF/XML, N3 and N-Triples
3. An OWL API
4. In-memory and persistent storage
5. SPARQL query engine

เนื่องจาก ในการพัฒนานั้น จะเป็นการใช้งาน Jena ร่วมกับ OWL ผ่านทาง API จึงจะขอ

กล่าวในรายละเอียดเฉพาะในส่วนของ API ที่ได้ใช้ในการพัฒนาเท่านั้น สำหรับรายละเอียดส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
อื่น ๆ สามารถศึกษาเพิ่มเติมได้ที่ <http://jena.sourceforge.net>

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Jena นั้นจะมี API สำหรับภาษา Java ที่ใช้ในการจัดการกับ RDF graphs โดยได้สร้างวัตถุเพื่อใช้แทน graphs, resources, properties และ literals ในรูปของ interface ที่ชื่อว่า Resource, Property และ Literal สำหรับ graphs นั้น Jena จะมองในชื่อของ model และได้สร้างวัตถุในรูปของ interface Model เพื่อใช้แทน graphs นั้นเอง

พิจารณาตัวอย่างการสร้าง graph หรือ model อย่างง่าย ๆ ดังนี้

```
// some definitions
static String personURI = "http://somewhere/JohnSmith";
static String fullName = "John Smith";

// create an empty Model
Model model = ModelFactory.createDefaultModel();

// create the resource
Resource johnSmith = model.createResource(personURI);

// add the property
johnSmith.addProperty(VCARD.FN, fullName);
```

จากตัวอย่าง โปรแกรมเริ่มการทำงานโดยสร้าง model เปล่า ๆ โดยใช้ ModelFactory เมทอด createDefaultModel() เพื่อสร้าง model แล้วจัดเก็บไว้ใน memory จากนั้นจึงสร้าง resource John Smith และเพิ่มในส่วนของ property เข้าไป โดย property นั้นจะถูกจัดเก็บในลักษณะของค่า constant ในคลาส VCARD ซึ่งภายในจะเต็มไปด้วย วัตถุต่าง ๆ ซึ่งมีรายละเอียดและความหมายตรงกับโครงสร้างของ VCARD schema นั้นเอง

เมื่อเราต้องการแสดง RDF ออกมาในรูปของ XML เราสามารถทำได้ง่าย ๆ โดยการเรียกใช้งานเมทอดหนึ่งของ model ในชื่อว่า write ดังตัวอย่างต่อไปนี้ จะเป็นการแสดง RDF ในรูปแบบของ XML ผ่านทางหน้าจอกอนโซล

```
// now write the model in XML form to a file
```

```
model.write(System.out);
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำซ้ำโดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งเมื่อโปรแกรมทำงานจะได้ผลลัพธ์ในลักษณะดังนี้

```
<rdf:RDF
  xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
  xmlns:vcard='http://www.w3.org/2001/vcard-rdf/3.0#'
>
  <rdf:Description rdf:about='http://somewhere/JohnSmith'>
    <vcard:FN>John Smith</vcard:FN>
    <vcard:N rdf:nodeID="A0"/>
  </rdf:Description>
  <rdf:Description rdf:nodeID="A0">
    <vcard:Given>John</vcard:Given>
    <vcard:Family>Smith</vcard:Family>
  </rdf:Description>
</rdf:RDF>
```

นอกจากการเขียนดังกล่าวแล้ว Jena API ได้รองรับการอ่านจากไฟล์ที่เป็น OWL ด้วยเช่นกัน โดยจะสามารถเรียกใช้งานผ่านเมทอด read ของคลาส model ดังตัวอย่างต่อไปนี้

```
// create an empty model
Model model = ModelFactory.createDefaultModel();

// use the FileManager to find the input file
InputStream in = FileManager.get().open( inputFileName );
if (in == null) {
  throw new IllegalArgumentException(
    "File: " + inputFileName + " not found");
}

// read the RDF/XML file
model.read(in, "");
```

```
// write it to standard out
model.write(System.out);
```

ในการอ่าน OWL จากแหล่งข้อมูลต่าง ๆ นั้น Jena ไม่เพียงสนับสนุนเฉพาะการอ่านจากไฟล์เท่านั้น Jena ยังสนับสนุนการอ่านจาก Relational Database อย่าง Oracle, MySQL และอื่น ๆ อีกด้วย โดยสามารถศึกษารายละเอียดของ API พร้อมทั้งตัวอย่างเพิ่มเติมได้ที่ <http://jena.sourceforge.net/DB/index.html>

สำหรับการสืบค้นข้อมูลต่าง ๆ ใน OWL นั้น Jena จะกระทำผ่านคลาส model โดยสามารถไล่สืบค้นภายใน model ได้ ดังตัวอย่าง เป็นการแสดง subject ทั้งหมดที่มี property VCARD.FN อยู่ โดยผลลัพธ์ที่ได้จะส่งออกมาในรูปแบบของ interface Iterator ของ java นั่นเอง

```
// list vcards
ResIterator iter = model.listSubjectsWithProperty(VCARD.FN);
while (iter.hasNext()) {
    Resource r = iter.nextResource();
    ...
}
```

อย่างไรก็ตาม ในการค้นหาลักษณะนี้ จะพบว่า หากข้อมูลมีขนาดใหญ่ การค้นหาจะใช้เวลาและทรัพยากรระบบค่อนข้างสูง วิธีการเช่นนี้ย่อมไม่ใช่ทางเลือกที่ดีแน่ แต่หากว่า การสืบค้นข้อมูลใน OWL นั้นสามารถที่จะใช้ภาษาที่สื่อความหมายและเหมาะสมอย่างเช่น SPARQL ได้ ย่อมจะเป็นทางเลือกที่ดีกว่าแน่นอน ซึ่ง Jena เองนั้นได้สนับสนุนการส่ง คำค้นในรูปแบบของ SPARQL ไปสืบค้นข้อมูลใน model ได้ โดยจะสามารถกระทำผ่าน Jena API ในส่วนของ ARQ นั่นเอง

ARQ API นั้นจะมีส่วนคลาสหลัก ๆ อยู่ที่ package `com.hp.hpl.jena.query` ได้แก่

- คลาส Query เป็นคลาสที่จัดเก็บรายละเอียดต่าง ๆ ของการสืบค้น
- คลาส QueryExecution เป็นคลาสที่เป็นตัวแทนของ 1 คำสั่งค้นหา โดยสามารถสร้างจากเมทอด QueryExecutionFactory
- คลาส DatasetFactory ใช้สำหรับสร้างกลุ่มข้อมูล
- คลาส QuerySolution เป็นผลลัพธ์ของการสืบค้น 1 ระเบียบ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คลาส ResultSet เป็นกลุ่มของผลลัพธ์ของการสืบค้น โดยจะอยู่ในรูปของ interface Iterator ของ java
- คลาส ResultSetFormatter เป็นรูปแบบของผลลัพธ์ของการสืบค้นว่าต้องการให้อยู่ในรูปของข้อความธรรมดาหรือให้อยู่ในรูปของ RDF graph ซึ่งก็คือคลาส Model ใน Jena นั่นเอง

พิจารณาตัวอย่างในการใช้งาน ARQ API ดังต่อไปนี้

```
import com.hp.hpl.jena.query.* ;

Model model = ... ;
String queryString = " .... " ;
Query query = QueryFactory.create(queryString) ;
QueryExecution qexec = QueryExecutionFactory.create(query, model) ;
try {
    ResultSet results = qexec.execSelect() ;
    for ( ; results.hasNext() ; ) {
        QuerySolution soln = results.nextSolution() ;
        RDFNode x = soln.get("varName") ; // Get a result variable by name.
        Resource r = soln.getResource("VarR") ; // Get a result variable - must be a
        resource
        Literal l = soln.getLiteral("VarL") ; // Get a result variable - must be a literal
    }
} finally { qexec.close() ; }
```

จากตัวอย่าง การค้นหาจะเริ่มโดย QueryFactory นำเอาคำค้นที่อยู่ในรูปของ String ไปทำการสร้าง query ขึ้นมา จากนั้นจึงนำเอา query และ model ไปสร้างเป็นคำสั่งด้วยคลาส QueryExecutionFactory เมื่อได้คำสั่ง (qexec) แล้ว จึงสั่งให้เริ่มค้นหาผ่านเมทอด execSelect ได้ผลลัพธ์ออกมาในรูปของ ResultSet จากนั้นจึงไล่อ่านข้อมูลแต่ละระเบียนตามลักษณะของการ

เอกสารเขียนแบบ Iterator ใน java ต่อไป
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถกำหนดรูปแบบของผลลัพธ์จากการสืบค้นได้ โดยเรียกใช้งานคลาส `ResultSetFormatter` ดังตัวอย่างต่อไปนี้ ซึ่งเป็นการกำหนดให้ผลลัพธ์ที่ได้อยู่ในรูปแบบของข้อความธรรมดา

```
ResultSetFormatter fmt = new ResultSetFormatter(results, query) ;
fmt.printAll(System.out) ;
```

หรือ

```
ResultSetFormatter.out(System.out, results, query) ;
```

ในการสืบค้นในบางครั้งนั้น เราอาจไม่ได้ต้องการผลลัพธ์มาประมวลผลมากมาย อาจเพียงต้องการตรวจสอบว่ามีรูปแบบที่กำหนด อยู่ใน OWL หรือไม่ เราสามารถที่จะสอบถามผ่าน ARQ API ได้ โดยการเปลี่ยนไปเรียกเมทอด `execAsk` แทน ดังตัวอย่าง

```
Query query = QueryFactory.create(queryString) ;
QueryExecution qexec = QueryExecutionFactory.create(query, model) ;
boolean result = qexec.execAsk() ;
qexec.close() ;
```

อย่างไรก็ตาม จาก ARQ API นั้นจะพบว่า API ยังสนับสนุนเฉพาะการสืบค้นธรรมดา ๆ เท่านั้น หากต้องการเพิ่มเงื่อนไขบางอย่างร่วมในการสืบค้นด้วย จะยังไม่สามารถทำได้ เช่นการสืบค้นในลักษณะ ขึ้นต้นด้วย ลงท้ายด้วย หรือประกอบด้วย เป็นต้น นอกจากนี้แล้ว API ยังไม่สนับสนุนการสืบค้นในลักษณะนับ (Count) การจัดกลุ่ม หรือการกรองผลลัพธ์ที่ได้จากการสืบค้น อีกด้วย ดังนั้น เพื่อเป็นการแก้ปัญหาดังกล่าว เราจึงต้องใช้งาน ARQ ร่วมกับ Lucene

Lucene นั้น ถูกพัฒนาขึ้นโดยนักพัฒนาจาก Apache ด้วยภาษา Java จัดเป็นเครื่องมือสำหรับนักพัฒนาที่ใช้สำหรับการค้นหาข้อความที่มีประสิทธิภาพสูง รายละเอียดเพิ่มเติมสามารถศึกษาต่อได้ที่ <http://lucene.apache.org/java/docs/index.html>

ในการใช้งาน ARQ ร่วมกับ Lucene นั้น จะต้องสร้างดัชนีสำหรับการค้นหาขึ้นมาก่อน โดยสามารถสร้างจากคลาส `IndexBuilderString` หรือ `IndexBuilderSubject` ดังตัวอย่าง

```
// -- Read and index all literal strings.
IndexBuilderString larqBuilder = new IndexBuilderString() ;

// -- Index statements as they are added to the model.
model.register(larqBuilder) ;

FileManager.get().readModel(model, datafile) ;

// -- Finish indexing
larqBuilder.closeForWriting() ;
model.unregister(larqBuilder) ;

// -- Create the access index
IndexLARQ index = larqBuilder.getIndex() ;
```

ภายหลังจากสร้างดัชนีเรียบร้อยแล้ว จึงนำไปใช้งานร่วมกับ ARQ ดังตัวอย่าง

```
QueryExecution qExec = QueryExecutionFactory.create(query, model) ;
// -- Make available to this query execution only
LARQ.setDefaultIndex(qExec.getContext(), index) ;
```

จากนั้น เมื่อต้องการสืบค้นโดยใช้ Lucene ช่วย เราสามารถเรียกใช้ฟังก์ชัน pf:textMatch พร้อมกลับคำสั่งสืบค้นได้เลย ตัวอย่างเช่น

```
PREFIX pf: <http://jena.hpl.hp.com/ARQ/property#>
SELECT ?doc
{
  ?lit pf:textMatch '+text' .
  ?doc ?p ?lit
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำเอาไปใช้งานผ่านทาง API จะมีลักษณะดังต่อไปนี้

```
String queryString = StringUtils.join("\n", new String[]{
    "PREFIX pf: <http://jena.hpl.hp.com/ARQ/property#>",
    "SELECT * {",
    "  ?lit pf:textMatch '+text'",
    "}"
});

Query query = QueryFactory.create(queryString);
QueryExecution qExec = QueryExecutionFactory.create(query, model);
ResultSetFormatter.out(System.out, qExec.execSelect(), query);
```

หากต้องการแสดงคำค้นพร้อมกับคะแนน เราสามารถแก้ไขคำสั่งค้น เป็น

```
PREFIX pf: <http://jena.hpl.hp.com/ARQ/property#>
SELECT ?doc ?score
{
  (?lit ?score ) pf:textMatch '+text' .
  ?doc ?p ?lit
}
```

และหากต้องการจำกัดจำนวนผลลัพธ์ที่ได้ เราสามารถแก้ไขคำสั่งค้น เป็น

```
?lit pf:textMatch ( '+text' 100 ) .    # Limit to at most 100 hits
?lit pf:textMatch ( '+text' 0.5 ) .    # Limit to Lucene scores of 0.5 and over.
?lit pf:textMatch ( '+text' 0.5 100 ) . # Limit to scores of 0.5 and limit to 100 hits
```

นอกจากการใช้ Lucene ผ่านทางคำสั่งคั่นคั้งที่ผ่านมาแล้วนั้น เราสามารถเรียกใช้งานผ่านทาง API ได้อีกด้วย ดังตัวอย่างเช่น

```
// -- Create the access index
IndexLARQ index = larqBuilder.getIndex();

NodeIterator nIter = index.searchModelByIndex("+text");

for ( ; nIter.hasNext(); )
{
    // if it's an index storing literals ...
    Literal lit = (Literal)nIter.nextNode();
}
```

นอกจากสนับสนุนในการสืบค้นข้อความดังลักษณะที่กล่าวมาแล้ว ยังสนับสนุนในส่วนของการกรองผลลัพธ์จากการค้นหา, การนับจำนวน หรือการจัดกลุ่ม โดยสามารถศึกษาเพิ่มเติมได้ที่ <http://jena.sourceforge.net/ARQ/extension.html#valueFunctions>

2.5 The Protégé-OWL Plugin and CO-ODE Tools

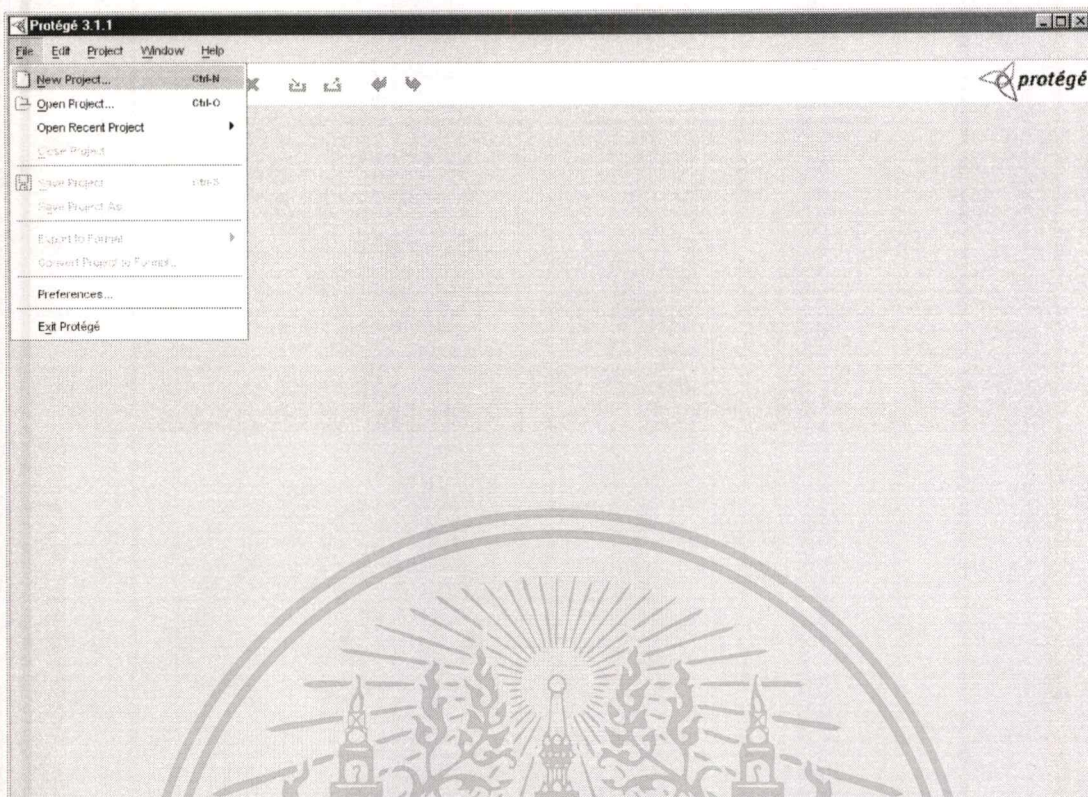
Protégé เป็นเครื่องมือที่ใช้ช่วยในการออกแบบ OWL โดยในการพัฒนาระบบซีแมนติกเวบฟุตบอลได้ใช้ Protégé Edition 1.0 ซึ่งเป็นลิขสิทธิ์ของมหาวิทยาลัยแมนเชสเตอร์ (The University Of Manchester) มาช่วยในการออกแบบ OWL โดยมีการใช้งานพื้นฐานดังต่อไปนี้

2.4.1 การสร้างโปรเจกใหม่

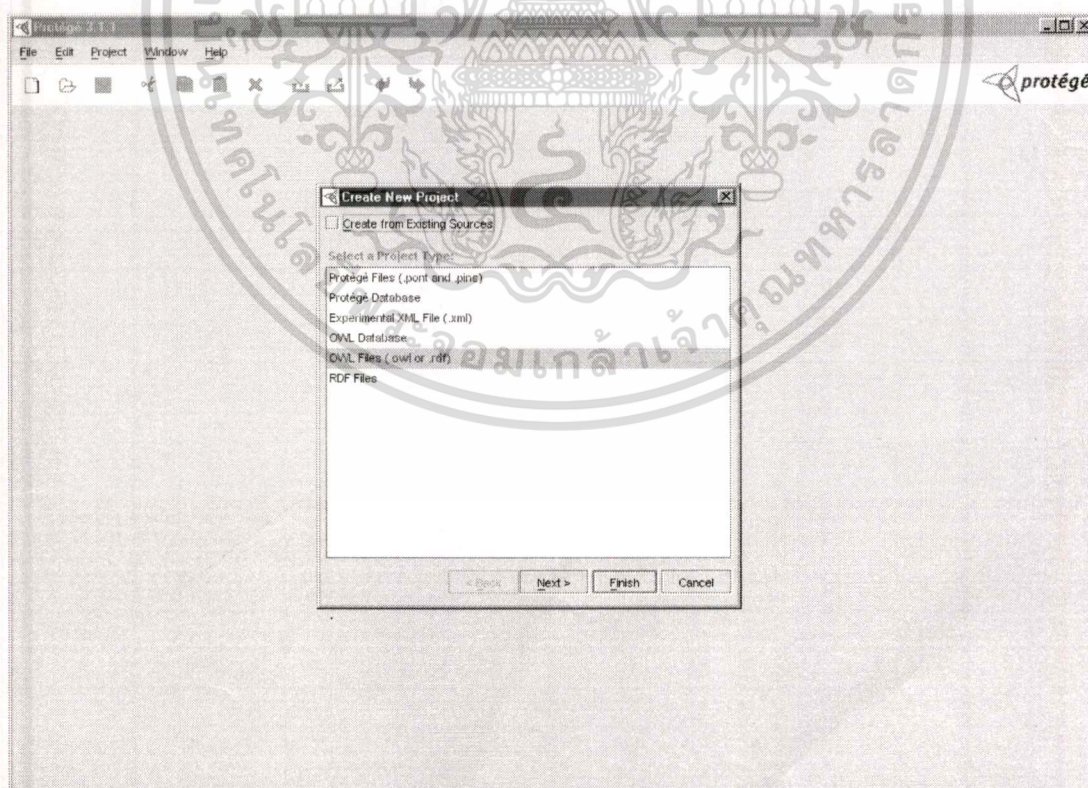
ให้เลือกสร้างโปรเจกใหม่ (New Project...) ที่เมนูของเครื่องมือ ดังรูปที่ 2.4 จากนั้นกดปุ่มเพื่อไปหน้าถัดไป (Next>) เพื่อกำหนดลักษณะของโปรเจกดังรูปที่ 2.5 เมื่อกดปุ่มเพื่อไปหน้าถัดไป (Next>) เพื่อกำหนดการนำเข้าไฟล์ OWL ในกรณีที่มีไฟล์ OWL อยู่แล้ว ดังรูปที่ 2.6 จากนั้นให้กดปุ่มเพื่อไปหน้าถัดไป (Next>) เพื่อกำหนดชื่อของ OWL ดังรูปที่ 2.7 เมื่อกำหนดชื่อของ OWL แล้วให้กดปุ่มเพื่อไปหน้าถัดไป (Next>) ให้เลือกประเภทของ OWL ที่ต้องการใช้ ดังรูปที่ 2.8 จากนั้นให้ไปหน้าถัดไป (Next>) เพื่อเลือกมุมมองของ OWL ดังรูปที่ 2.9 จากนั้นก็สิ้นสุดการสร้างโปรเจกใหม่ (Finish) จะได้โปรเจกใหม่ดังรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามนำไปเผยแพร่หรือทำซ้ำโดยไม่ได้รับอนุญาต หากฝ่าฝืนจะดำเนินการตามกฎหมายที่เกี่ยวข้อง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

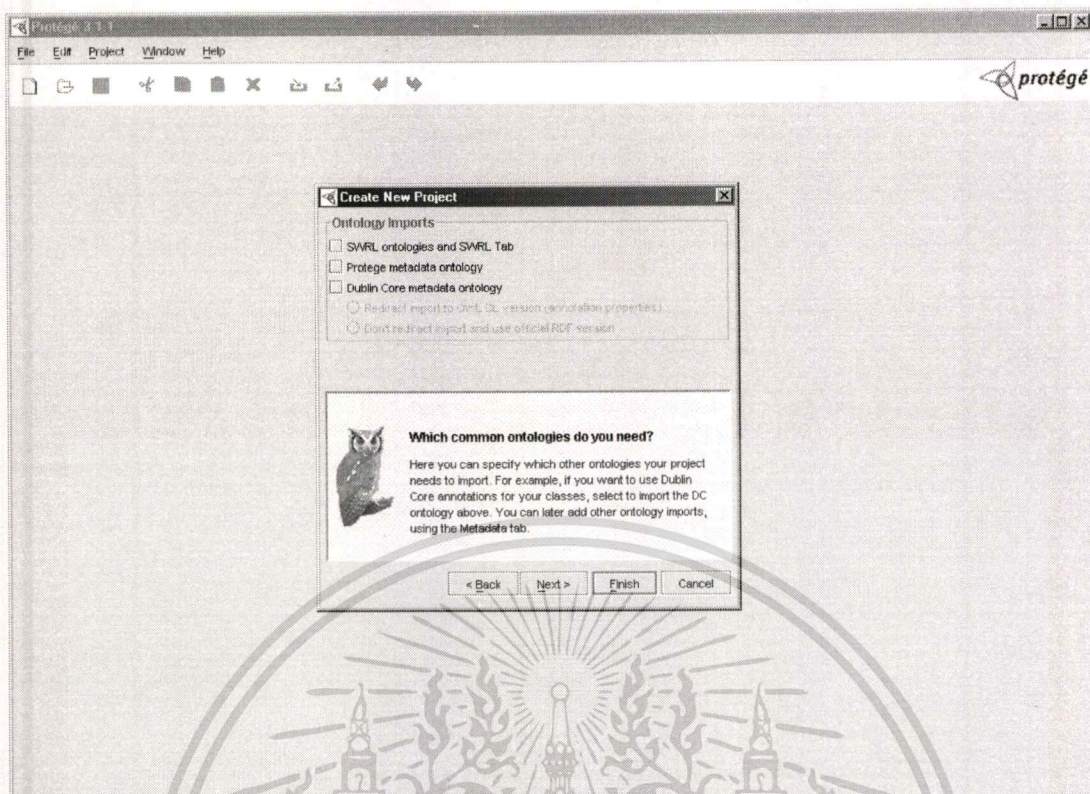


รูปที่ 2.4 แสดงการสร้างโปรเจกใหม่ใน Protégé

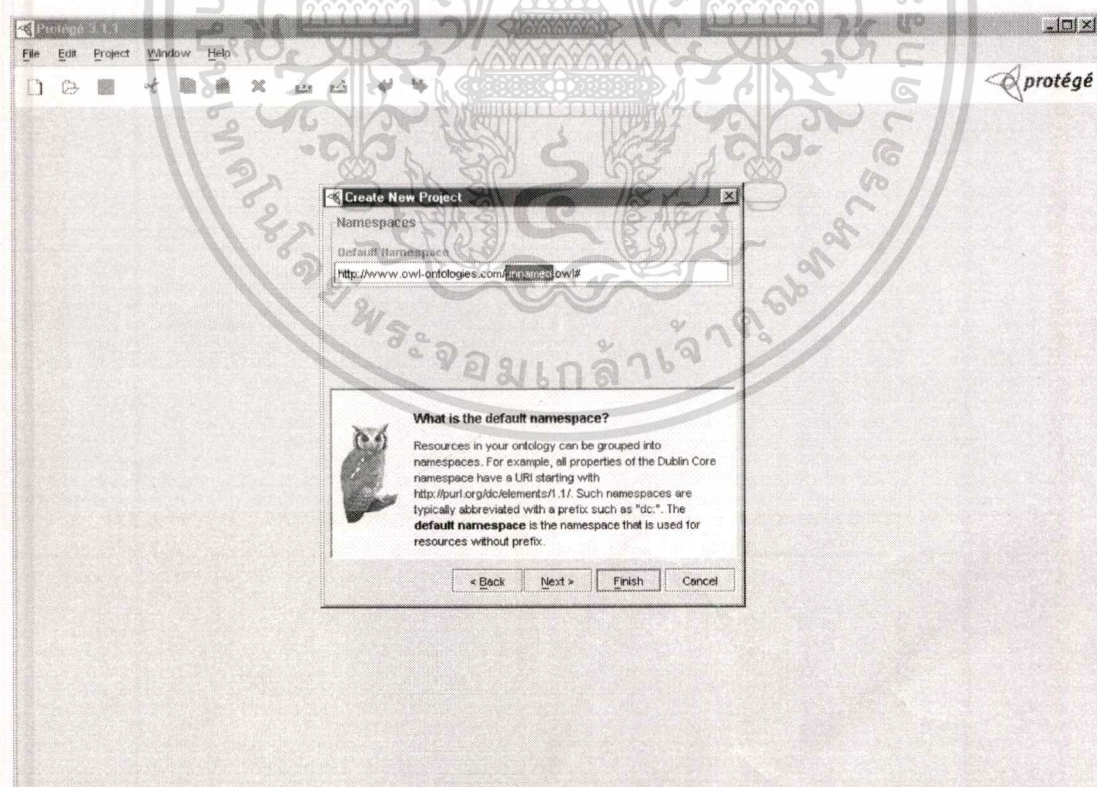


รูปที่ 2.5 แสดงการเลือกลักษณะของโปรเจกใหม่ใน Protégé

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

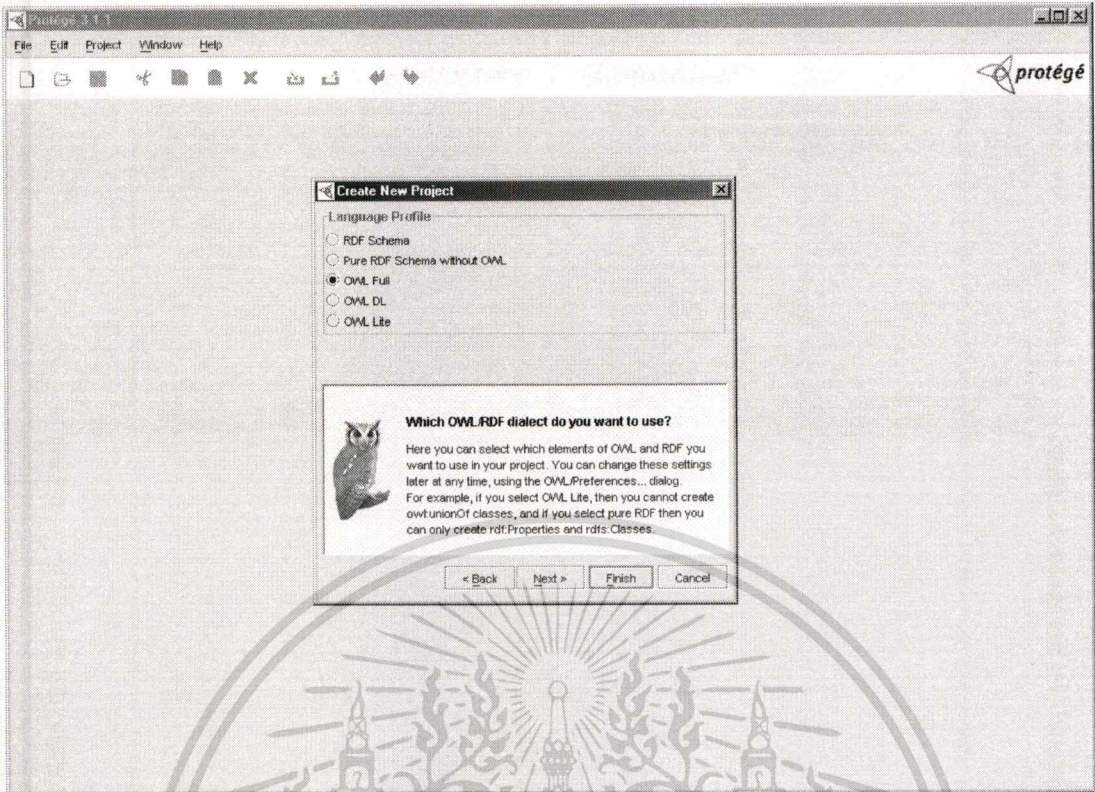


รูปที่ 2.6 แสดงตัวเลือกให้ทำการนำเข้าไฟล์ OWL สำหรับโปรเจกต์ใหม่ใน Protégé

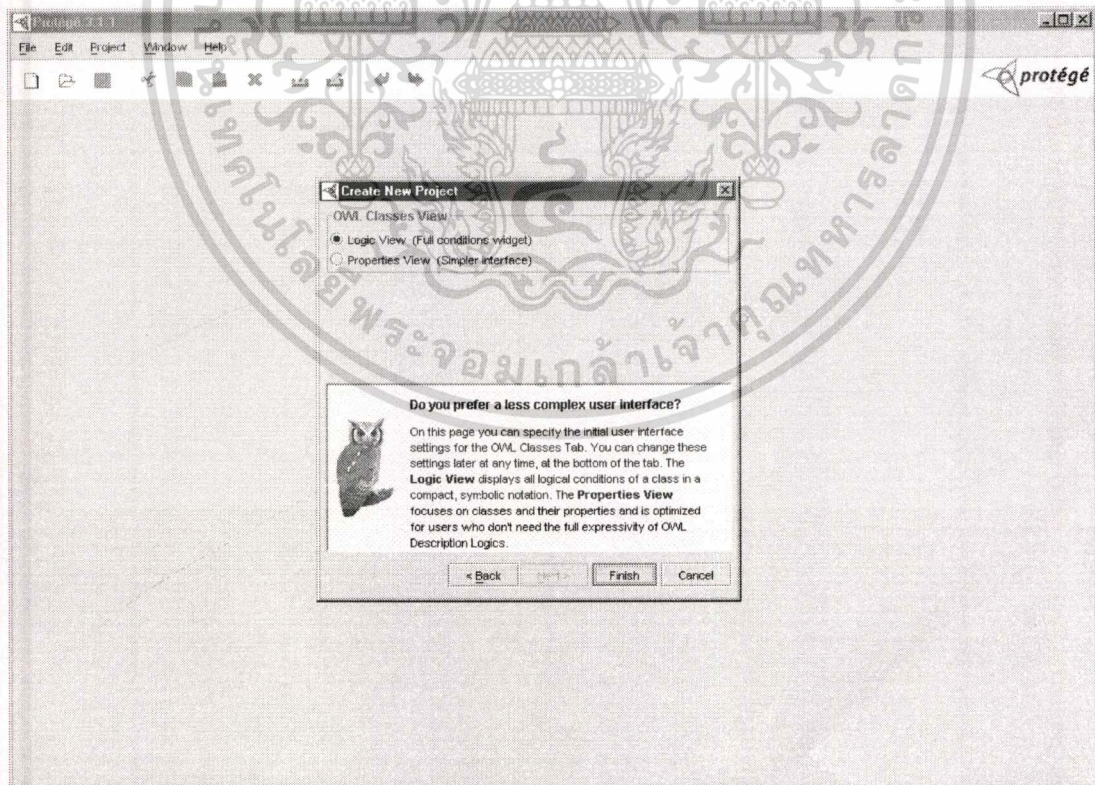


รูปที่ 2.7 แสดงการกำหนดชื่อไฟล์ OWL สำหรับโปรเจกต์ใหม่ใน Protégé

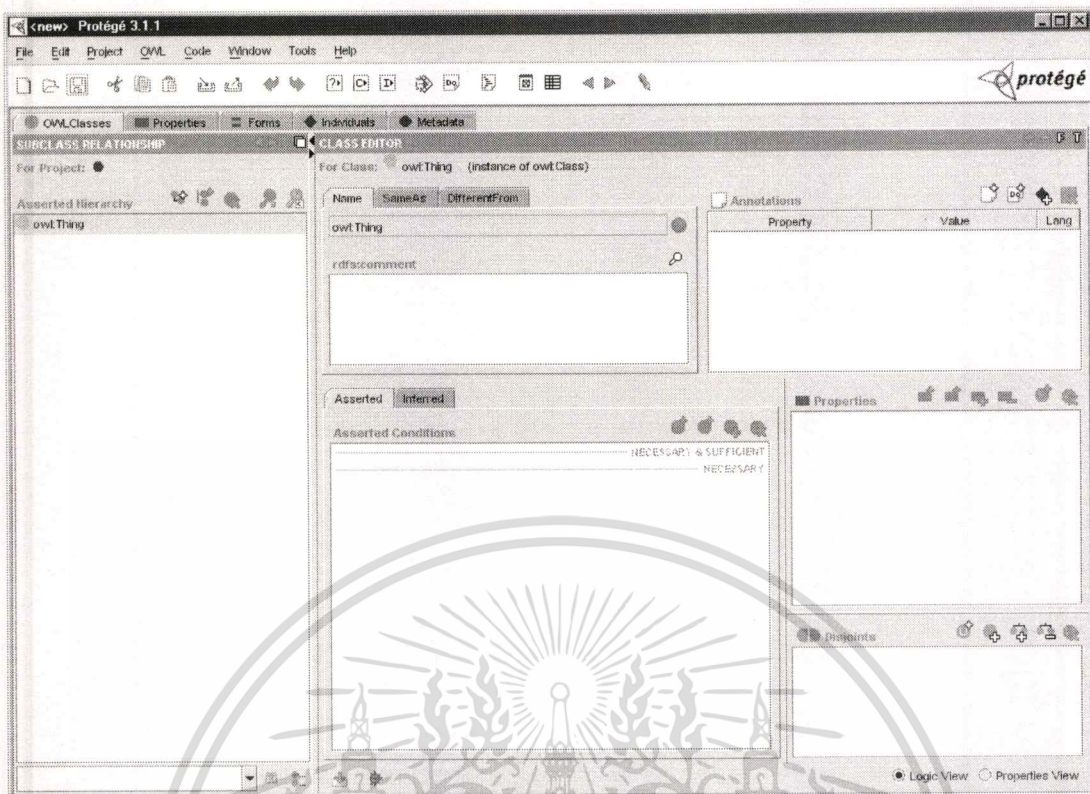
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงการเลือกประเภทของ OWL ของโปรเจกใหม่ใน Protégé

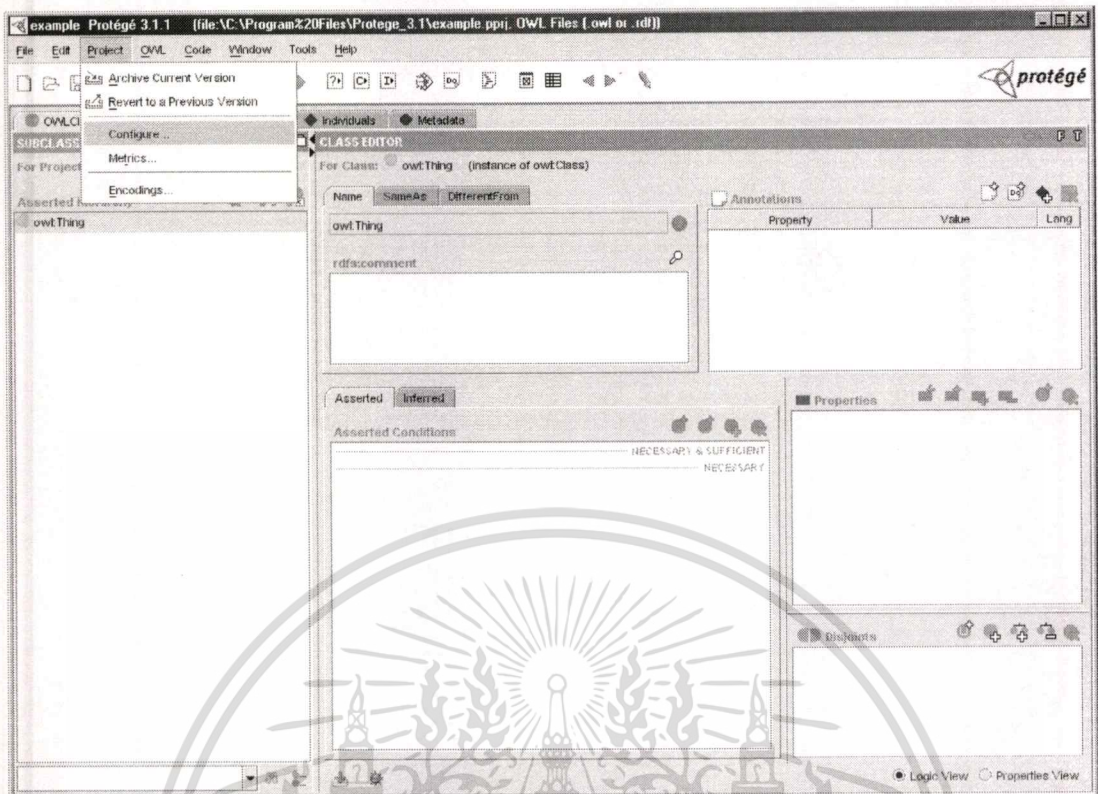


รูปที่ 2.9 แสดงการเลือกมุมมอง OWL สำหรับโปรเจกใหม่ใน Protégé

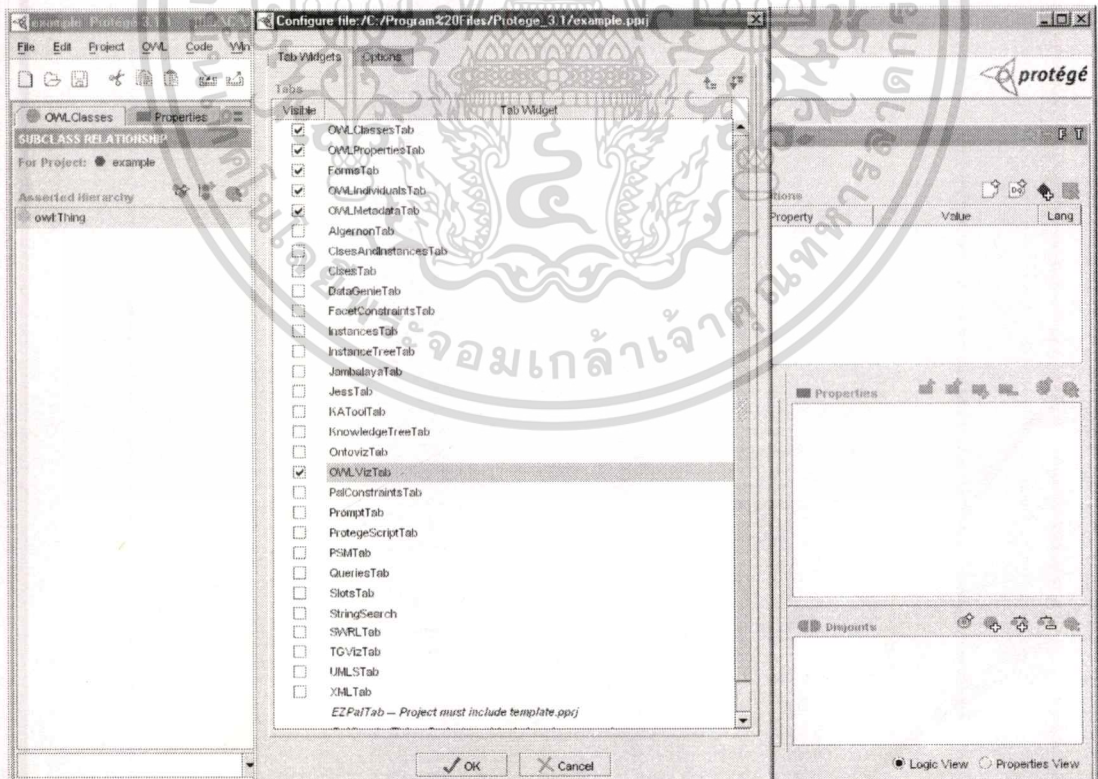


รูปที่ 2.10 แสดงการสิ้นสุดเมื่อสร้างโปรเจกใหม่ใน Protégé สำเร็จ

เมื่อทำการสร้าง โปรเจกแล้วยังมีส่วนที่ต้องทำการกำหนดให้กับโปรเจก ส่วนสำคัญคือการกำหนดให้แสดงรูปภาพของ OWL ที่สร้างเพื่อดูภาพรวมของ OWL นั้นเอง โดยให้เลือกที่เมนู โปรเจก (Project) และเลือกกำหนดค่า (Configure...) ดังรูปที่ 2.11 จากนั้นให้เลือกกล่องหน้ารายการ OWLVizTab ดังรูปที่ 2.12 แล้วเลือกตกลง (OK) เพื่อยืนยันการกำหนดค่า จากนั้นจะเห็นได้ว่าการเพิ่มหน้าแสดงรูปภาพดังรูปที่ 2.13

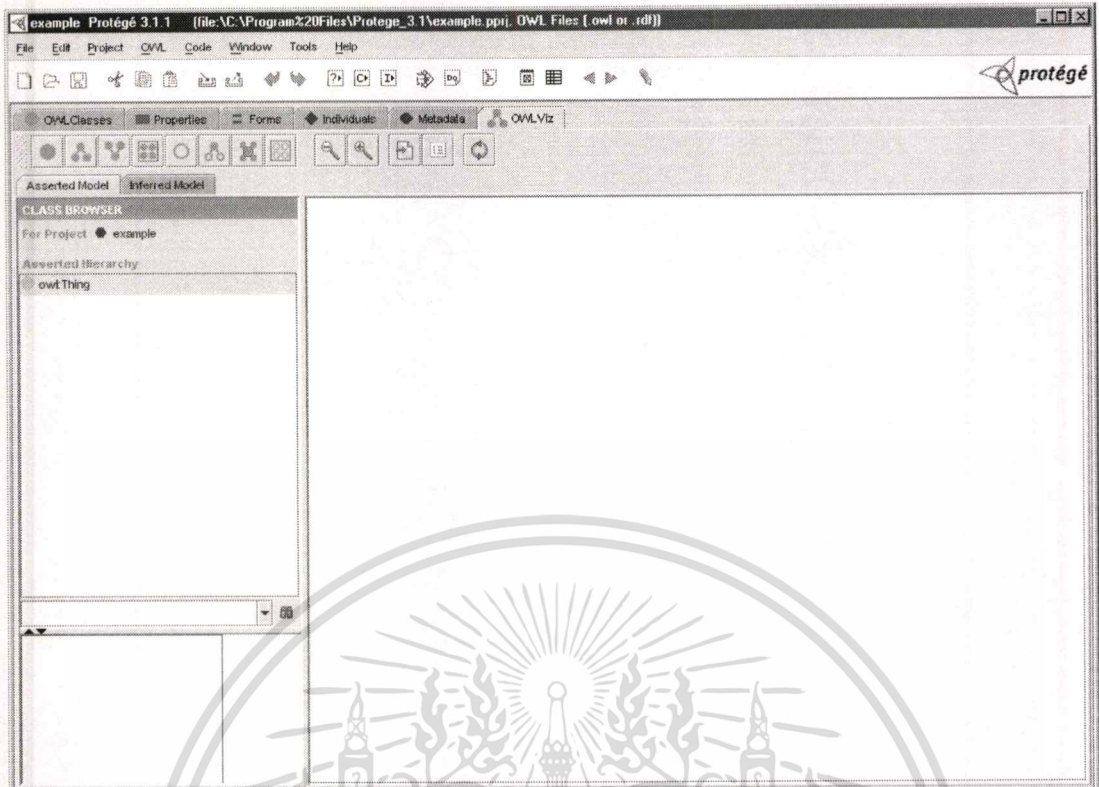


รูปที่ 2.11 แสดงการเลือกเมนูเพื่อกำหนดค่าให้โปรเจกใหม่ใน Protégé



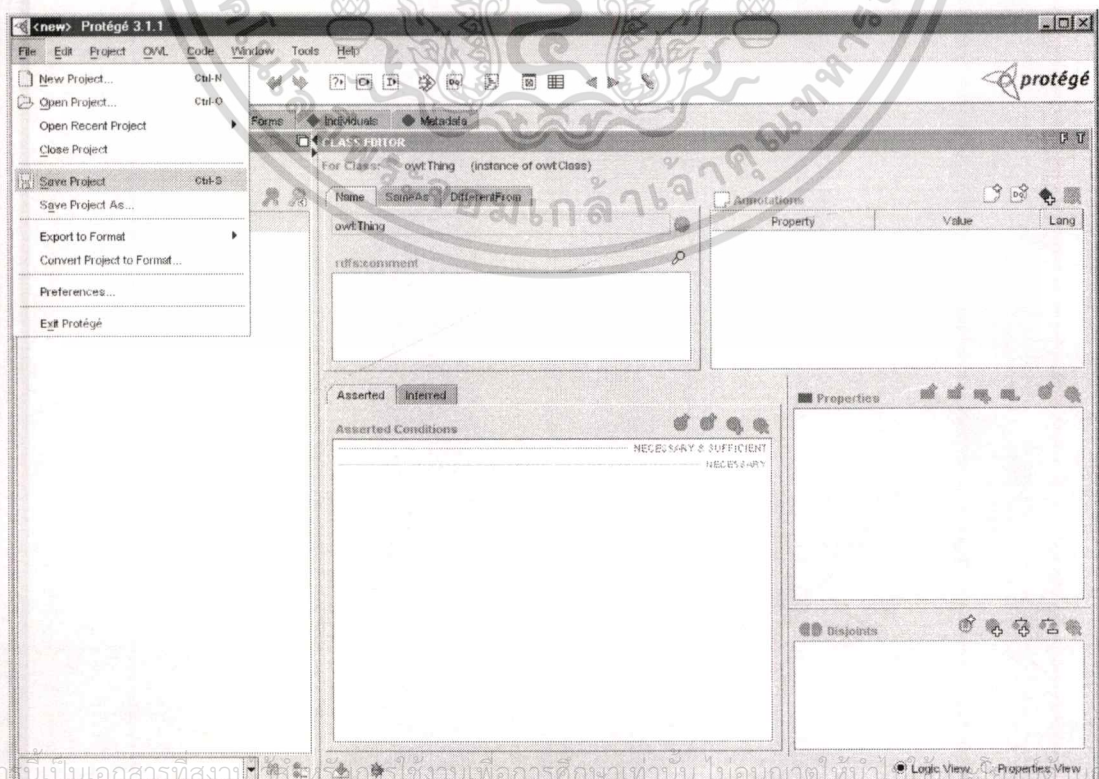
รูปที่ 2.12 แสดงการกำหนดค่าเพื่อแสดงรูปภาพ OWL ของโปรเจกใหม่ใน Protégé

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



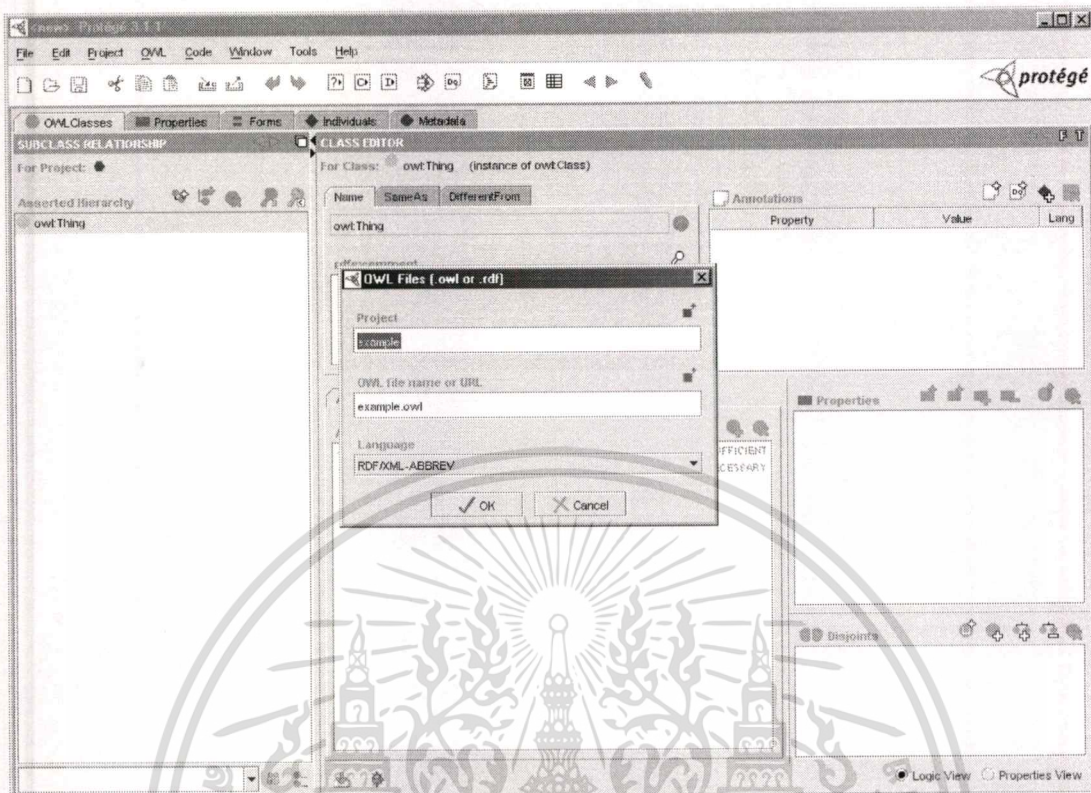
รูปที่ 2.13 แสดงหน้ารูปภาพ OWL ของโปรเจกใหม่ใน Protégé

จากนั้นก็ทำการบันทึกโปรเจกใหม่ดังรูปที่ 2.14 แล้วกำหนดชื่อโปรเจกที่เพิ่งสร้างดังรูปที่ 2.15



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือทำซ้ำโดยไม่ได้รับอนุญาต

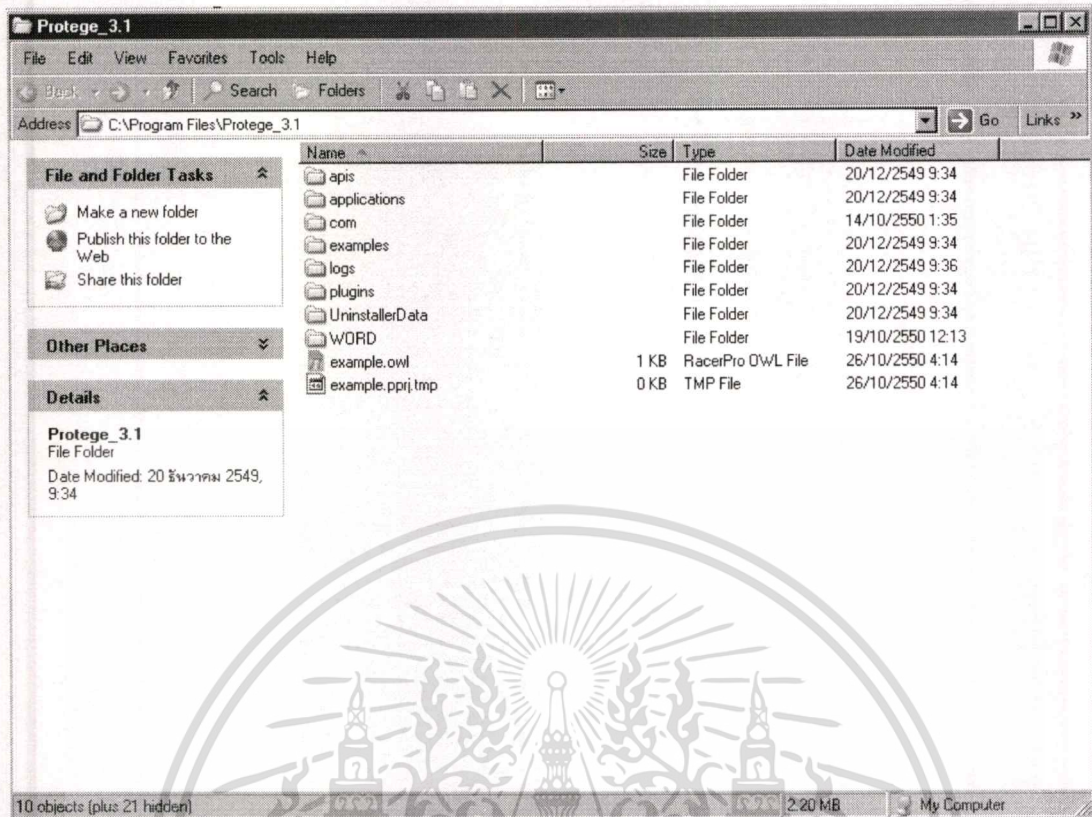
รูปที่ 2.14 แสดงการเลือกเมนูเพื่อทำการบันทึกโปรเจกใหม่ใน Protégé



รูปที่ 2.15 แสดงการกำหนดชื่อเพื่อทำการบันทึกโปรเจกใหม่ใน Protégé

เมื่อทำการบันทึกชื่อโปรเจกใหม่แล้วจะเห็นได้ว่าการสร้างโปรเจกไฟล์ และ
OWL ไฟล์ใหม่ ดังรูปที่ 2.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

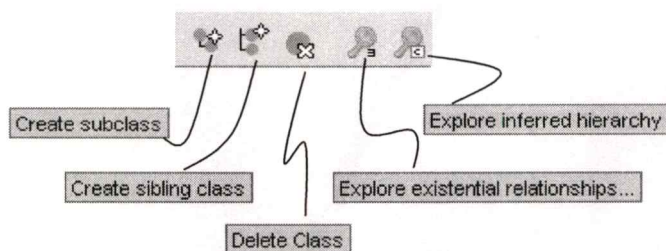


รูปที่ 2.16 แสดงไฟล์โปรเจกต์และไฟล์ OWL เมื่อทำการบันทึกโปรเจกต์ใหม่ใน Protégé

2.4.2 การสร้างคลาส (Named Classes)

การสร้างคลาสมีปุ่มต่างๆให้ใช้ดังรูปที่ 2.17 ดังรายละเอียดต่อไปนี้

- ปุ่มสร้างคลาสย่อย (Create subclass)
- ปุ่มสร้างคลาสในระดับเดียวกัน (Create sibling class)
- ปุ่มลบคลาส (Delete Class)
- ปุ่มดูรายการความสัมพันธ์ทั้งหมดที่มี (Explore existential relationships...)
- ปุ่มแสดงลำดับชั้น (Explore inferred hierachy)



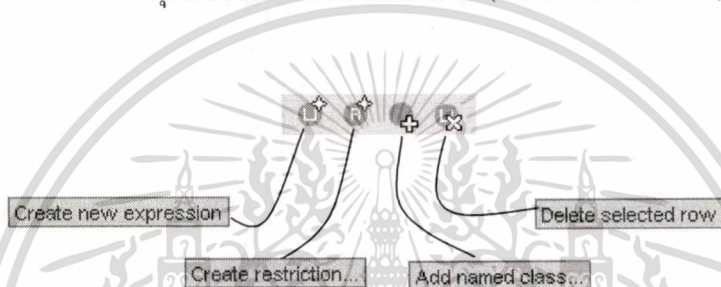
รูปที่ 2.17 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างคลาสใน Protégé

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 การกำหนดข้อจำกัดของคลาส OWL (Property Restriction)

การสร้างข้อจำกัดของคลาส OWL มีปุ่มต่างๆ ให้ใช้ดังรูปที่ 2.18 โดยมีรายละเอียดดังต่อไปนี้

- ปุ่มสร้างสมการ (Create new expression)
ใช้ในการกำหนดค่าต่างๆ เช่น All Value From, Some Value From เป็นต้น
- ปุ่มสร้างข้อจำกัด (Create restriction...)
- ปุ่มเพิ่มคลาสให้เป็นข้อจำกัด (Add named class...)
- ปุ่มลบรายการข้อจำกัดที่เลือก (Delete selected row)

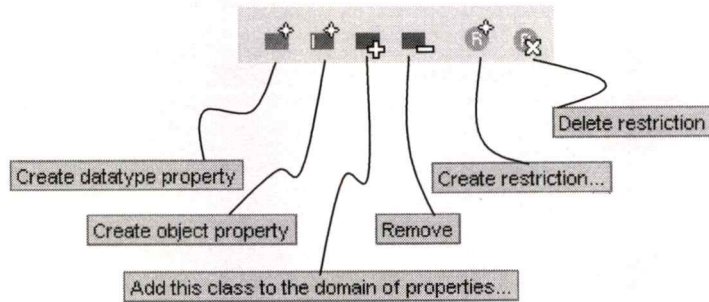


รูปที่ 2.18 แสดงรายละเอียดปุ่มต่างๆ สำหรับสร้างข้อจำกัดของคลาสใน Protégé

2.4.4 การสร้างความสัมพันธ์ของ OWL (Properties)

การสร้างความสัมพันธ์มีปุ่มต่างๆ ให้ใช้ดังรูป 2.19 โดยมีรายละเอียดดังต่อไปนี้

- ปุ่มสร้างความสัมพันธ์ข้อมูล (Create datatype property)
- ปุ่มสร้างความสัมพันธ์ระหว่างคลาส (Create object property)
- ปุ่มเพิ่มคลาสให้เป็นค่าหลักของความสัมพันธ์ (Add this class to the domain of properties...)
- ปุ่มลบรายการความสัมพันธ์ที่เลือก (Remove)
- ปุ่มสร้างข้อจำกัดของความสัมพันธ์ (Create restriction...)
- ปุ่มลบข้อจำกัดของความสัมพันธ์ (Delete restriction)

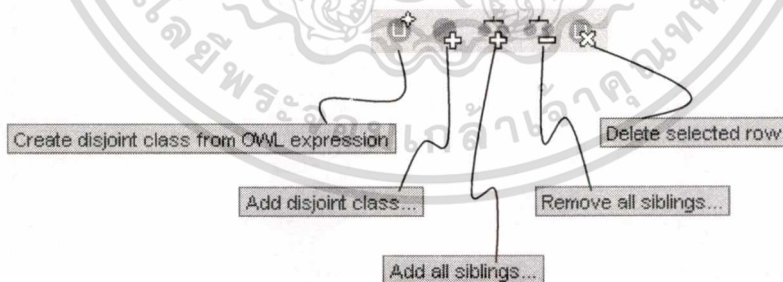


รูปที่ 2.19 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างความสัมพันธ์ (Property) ใน Protégé

2.4.5 การกำหนดคลาสที่ไม่เกี่ยวข้องกัน (Disjoint Classes)

การสร้างกำหนดที่ไม่เกี่ยวข้องกันมีปุ่มต่างๆให้ใช้ดังรูปที่ 2.20 โดยมีรายละเอียดดังต่อไปนี้

- ปุ่มกำหนดสมการสำหรับคลาสที่ไม่เกี่ยวข้องกัน (Create disjoint class from OWL expression)
- ปุ่มกำหนดคลาสที่ไม่เกี่ยวข้องกัน (Add disjoint class...)
- ปุ่มกำหนดเลือกคลาสในระดับเดียวกันทั้งหมด (Add all siblings...)
- ปุ่มลบคลาสทั้งหมดในระดับเดียวกันที่กำหนดไว้เป็นคลาสที่ไม่เกี่ยวข้องกัน (Remove all siblings...)
- ปุ่มลบรายการคลาสที่ไม่เกี่ยวข้องกัน (Delete selected row)



รูปที่ 2.20 แสดงรายละเอียดปุ่มต่างๆสำหรับสร้างคลาสที่ไม่เกี่ยวข้องกันใน Protégé

บทที่ 3

การวิเคราะห์และการออกแบบซีเมนติกเวบ

เนื้อหากีฬาฟุตบอล

การวิเคราะห์และออกแบบซีเมนติกเวบฟุตบอล ในส่วนที่เป็นข้อมูลที่เกี่ยวข้องกับระบบ ได้แก่ ข้อมูลทีมฟุตบอล, ข้อมูลนักฟุตบอล และข้อมูลการแข่งขัน ได้ออกแบบเป็นออนโทโลยี OWL ดังรูปที่ 3.1 จากรายละเอียดดังต่อไปนี้

3.1 การวิเคราะห์เนื้อหากีฬาฟุตบอล

หัวข้อนี้จะแสดงรายละเอียดกีฬาฟุตบอลซึ่งจะใช้ในการพัฒนาระบบงานซีเมนติกเวบฟุตบอล มีรายละเอียดข้อมูลดังต่อไปนี้

3.1.1 ระดับทีมฟุตบอล (Division)

กีฬาฟุตบอลมีการจัดแบ่งทีมฟุตบอลออกเป็นระดับต่างๆ ตามความสามารถของทีมฟุตบอล ออกเป็นกลุ่มๆ เรียงจกทีมที่มีความสามารถในระดับสูงลงไป โดยจะสามารถจัดแบ่งได้ดังต่อไปนี้

3. ระดับพรีเมียร์ (Premier Division)
4. ระดับแชมเปียนชิพ (Championship Division)
5. ระดับลีกหนึ่ง (League One Division)
6. ระดับลีกสอง (League Two Division)

3.1.2 การแข่งขัน (Match Play)

การแข่งขันภายในประเทศอังกฤษของทีมฟุตบอลระดับพรีเมียร์ จะมีการแข่งขันอยู่ 3 การแข่งขัน ดังต่อไปนี้

1. เอฟเอคัพ (The FA Cup)

เป็นการแข่งขันของทีมฟุตบอลภายในประเทศอังกฤษ เพื่อชิงถ้วยของสมาคมฟุตบอลอังกฤษ (FA Cup) โดยจัดการแข่งขันในรูปแบบแพ้คัดออกจนกระทั่งเหลือทีมที่ไม่เคยแพ้เพียงทีมเดียวเป็นผู้ชนะเลิศต่อไป

2. เอฟเอพรีเมียร์ลีก (FA Premier League)

เป็นการแข่งขันเฉพาะทีมฟุตบอลระดับพรีเมียร์ 20 ทีมเท่านั้น โดยใช้รูปแบบ

เอกสารนี้เป็นการนำเสนอเพียงแนวทางเบื้องต้นเพื่อให้เห็นภาพรวมเท่านั้น เมื่อผู้จัดทำเห็นประโยชน์และจำเป็นต่อการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 คะแนน ส่วนทีมแพ้จะไม่ได้คะแนน เมื่อครบ 1 ฤดูกาลแข่งขันแล้วจะนำเอาคะแนนมาเปรียบเทียบหาผู้ที่ได้คะแนนสูง สุดเป็นผู้ชนะเลิศประจำฤดูกาลนั้นไป

3. เดอะฟุตบอลคัพ (The Football League Cup)

เป็นการแข่งขันเฉพาะทีมฟุตบอลที่มีระดับเท่านั้น คือระดับพรีเมียร์ลีก, ระดับแชมเปี้ยนชิพ, ระดับลีกหนึ่ง และระดับลีกสอง แข่งขันกัน โดยใช้รูปแบบการแข่งขันแบบแพ้คัดออก ในช่วงรอบแรก ๆ ส่วนรอบรองชนะเลิศจะใช้วิธีแข่ง 2 นัดแบบเหย้าเยือนหาผู้ชนะเลิศชิงชนะเลิศต่อไป

รายละเอียดการแข่งขันฟุตบอลแต่ละนัด มีดังต่อไปนี้

1. สนามที่ใช้แข่งขัน (Stadium)
2. วันเวลาที่แข่งขัน (Date)
3. เป็นการแข่งขันแบบเหย้า หรือเยือน (Away-Home)
4. ผลการแข่งขัน (Score)

ซึ่งประกอบด้วยจำนวนประตูที่เกิดขึ้น, ชื่อผู้เล่น, ทีมฟุตบอล, เวลาที่ทำประตู, ลักษณะการทำประตู เช่นจากการเล่นแบบเปิด, การยิงลูกโทษ หรือการยิงลูกตั้งเตะ เป็นต้น และระยะเวลาในการแข่งขันจริง (Time Play)

5. รายชื่อผู้เล่นที่ลงสนาม
6. การเปลี่ยนตัวผู้เล่นระหว่างการแข่งขัน (Turn Player)

จะประกอบไปด้วย ผู้เล่นที่เปลี่ยนตัวเข้า (Turn-in Player), ผู้เล่นที่เปลี่ยนตัวออก (Turn-out Player) และนาทีที่เปลี่ยนตัว (Turn Minute)

7. การฟาล์วที่เกิดขึ้นระหว่างการแข่งขัน (Foul)

ประกอบด้วย ผู้เล่นที่ทำฟาล์ว (Foul Player), ได้รับโทษเป็นใบเหลืองหรือใบแดง (Red Card, Yellow Card) และนาทีที่เกิดการฟาล์ว (Foul Minute)

3.1.3 สโมสรของทีมฟุตบอล (Club)

เป็นต้นสังกัดของทีมฟุตบอลแต่ละทีม ทุกๆ ทีมที่ทำการแข่งขันในระดับพรีเมียร์ลีก จะต้อง มีสโมสรสังกัดเสมอ โดยจะมีรายละเอียดดังต่อไปนี้

1. ชื่อสโมสร (Club Name)
2. สนาม (Stadium)

ในส่วนของสนามแยกเป็น สนามซ้อมและสนามแข่ง ซึ่งแต่ละสนามอาจเป็นของสโมสรเอง หรือเป็นสนามกลางซึ่งถือเป็นสนามประจำเมืองได้

3. สถานที่ของสโมสร (Club Address)

3.1.4 ผู้เล่นหรือนักฟุตบอล (Player)

มีรายละเอียดส่วนตัวของแต่ละคนในทีม ดังต่อไปนี้

1. ชื่อนักฟุตบอล (Firstname)
2. ชื่อกลางนักฟุตบอล (Middlename)
3. นามสกุลนักฟุตบอล (Lastname)
4. หมายเลขเสื้อนักฟุตบอล (Shirt Number)
5. วันเกิด (Birthdate)
6. สัญชาติ (Nationality)
7. ตำแหน่งผู้เล่น (Position)

ตำแหน่งผู้เล่นแบ่งได้เป็น 4 ตำแหน่งหลักและมีตำแหน่งโดยละเอียดได้ดังนี้ [6]

- i. ผู้รักษาประตู (Goalkeeper/GK)
- ii. กองหลัง (Defensive)
 1. กองหลังตัวกลางหรือตัวหยุด (Centre Back/CB/Stopper)
 2. กองหลังตัวสุดท้ายหรือตัวกวาด (Sweeper/SW/Libero)
 3. กองหลัง (Back)
 - a. กองหลังซ้าย (Left Back)
 - b. กองหลังขวา (Right Back)
 - c. กองหลังคุมทั้งซ้ายและขวา (Full Back)
 4. ปีกหลัง (Wing Back)
 - a. ปีกหลังซ้าย (Left Wing Back)
 - b. ปีกหลังขวา (Right Wing Back)
 5. กองกลาง (Midfield)
 - a. กองกลางตัวกลาง (Centre Midfield)
 - b. กองกลางตัวรับ (Defensive Midfield)
 - c. กองกลางตัวรุก (Attacking Midfield)
 - d. ปีกหรือตัวทำเกมริมเส้น (Winger)
 - i. ปีกหรือตัวทำเกมริมเส้นด้านซ้าย (Left Winger)
 - ii. ปีกหรือตัวทำเกมริมเส้นด้านขวา (Right Winger)
 - e. กองหน้า (Attacker)
 - i. ศูนย์หน้าตัวเป้า (Centre Forward)
 - ii. สไตรเกอร์ (Striker)
 - iii. ตัวหน้าต่ำ (Deep-lying Forward)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5 โค้ช (Coach)

ซึ่งมีข้อมูล (Datatype) ดังนี้

1. ชื่อต้นโค้ช (Firstname)
2. ชื่อกลางโค้ช (Middlename)
3. นามสกุลโค้ช (Lastname)
4. วันเกิด (Birthdate)
5. สัญชาติ (Nationality)

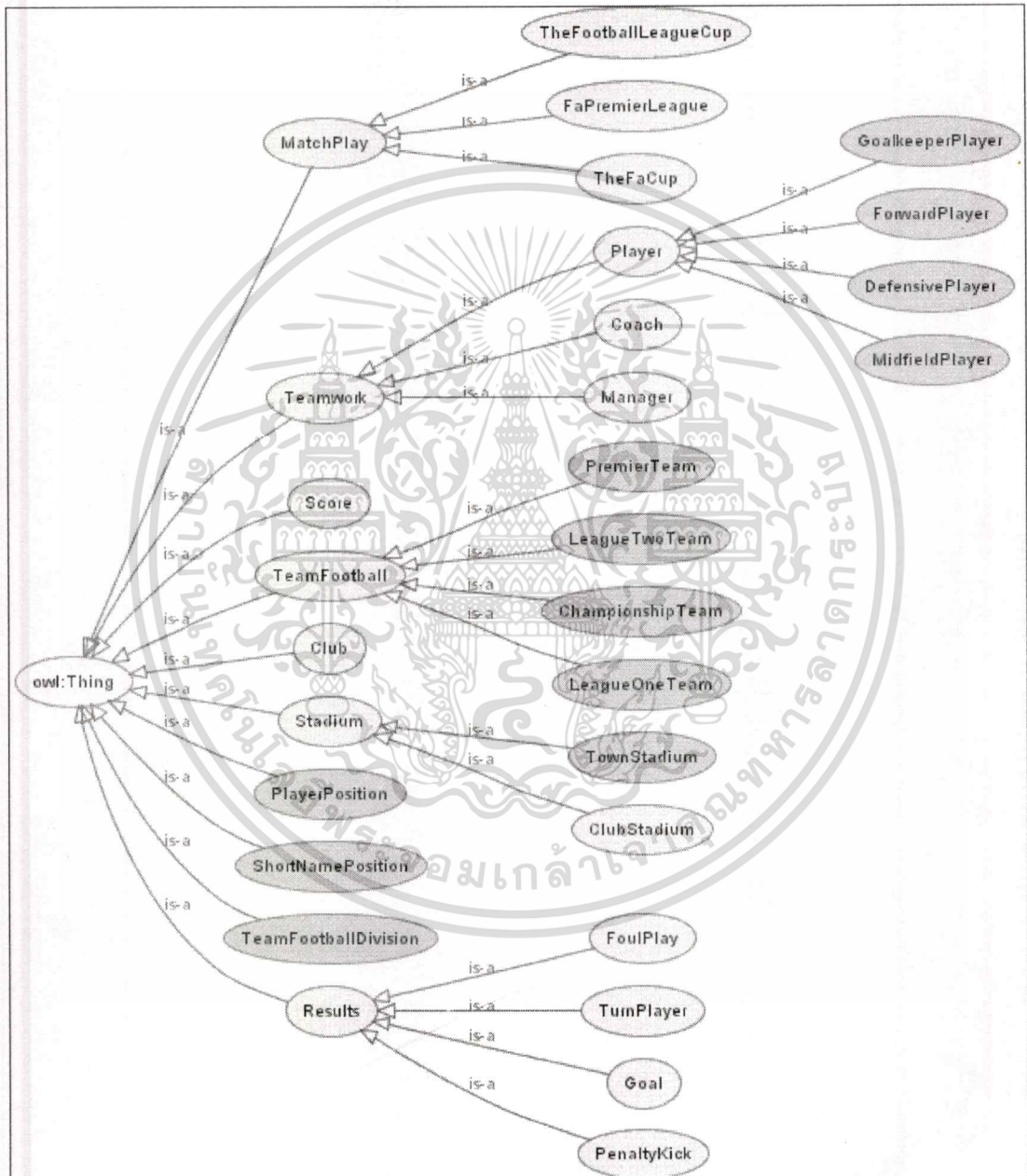
3.1.6 ผู้จัดการทีม (Manager)

ซึ่งมีข้อมูล (Datatype) ดังนี้

1. ชื่อต้นผู้จัดการทีม (Firstname)
2. ชื่อกลางผู้จัดการทีม (Middlename)
3. นามสกุลผู้จัดการทีม (Lastname)
4. วันเกิด (Birthdate)
5. สัญชาติ (Nationality)

3.2 การออกแบบ OWL (Web Ontology Language)

ในการออกแบบ OWL จากรายละเอียดกีฬาฟุตบอลดังกล่าวมาแล้ว จึงได้ OWL ดังรูปที่ 3.1 เป็นคลาสทั้งหมด 29 คลาส ซึ่งกำหนดให้อยู่ภายใต้คลาส OWL โดยแต่ละคลาสมีรายละเอียด คลาส, ความสัมพันธ์ระหว่างคลาส และข้อมูลของคลาสดังจะกล่าวต่อไป



รูปที่ 3.1 OWL แสดงข้อมูลเว็บซีแมนติกที่เป็นเนื้อหาของกีฬาฟุตบอล

จากคลาสทั้งหมด 29 คลาส มีรายละเอียดคลาส, ความสัมพันธ์ระหว่างคลาส และข้อมูลของคลาส ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 Stadium Class

เป็นคลาสของสนามฟุตบอลที่ใช้ในการแข่งขันโดยจะเก็บข้อมูลดังต่อไปนี้ ดัง OWL รูปที่ 3.2

1. ชื่อสนามฟุตบอล (Stadium Name)
2. ที่ตั้งสนามฟุตบอล (Stadium Address)

```
<owl:Class rdf:ID="Stadium"/>
<owl:DatatypeProperty rdf:ID="stadiumName">
  <rdfs:domain rdf:resource="#Stadium"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="stadiumAddress">
  <rdfs:domain rdf:resource="#Stadium"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

รูปที่ 3.2 แสดง OWL ของข้อมูลคลาส Stadium

แบ่งออกเป็น 2 คลาสย่อยดังต่อไปนี้

3.2.1.1 Town Stadium Class

เป็นคลาสของสนามฟุตบอลที่เป็นสนามกลางของเมืองใดๆ ดัง OWL รูปที่ 3.3

```
<owl:Class rdf:ID="TownStadium">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Stadium"/>
        <owl:Restriction>
          <owl:onProperty>
            <owl:InverseFunctionalProperty rdf:about="#isMatchOf"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class>
              <owl:complementOf rdf:resource="#ClubStadium"/>
            </owl:Class>
          </owl:someValuesFrom>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

รูปที่ 3.3 แสดง OWL ของคลาส Stadium และคลาส TownStadium

3.2.1.2 Club Stadium Class

เป็นคลาสของสนามฟุตบอลที่เป็นสนามของสโมสรใดๆ ซึ่งสโมสรก็จะมาจาก
 คลาส Club เท่านั้น ดัง OWL รูปที่ 3.4
 ไม่ว่าใครเป็นผู้เขียนเอกสารนี้ก็ตาม แต่ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<owl:Class rdf:ID="ClubStadium">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="isStadiumOf"/>
      </owl:onProperty>
      <owl:allValuesFrom>
        <owl:Class rdf:ID="Club"/>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#Stadium"/>
</owl:Class>

```

รูปที่ 3.4 แสดง OWL ของคลาส ClubStadium

และคลาส ClubStadium มีความสัมพันธ์กับคลาส Club เป็น Range ให้มีความสัมพันธ์เป็น isStadiumOf ดัง OWL รูปที่ 3.5

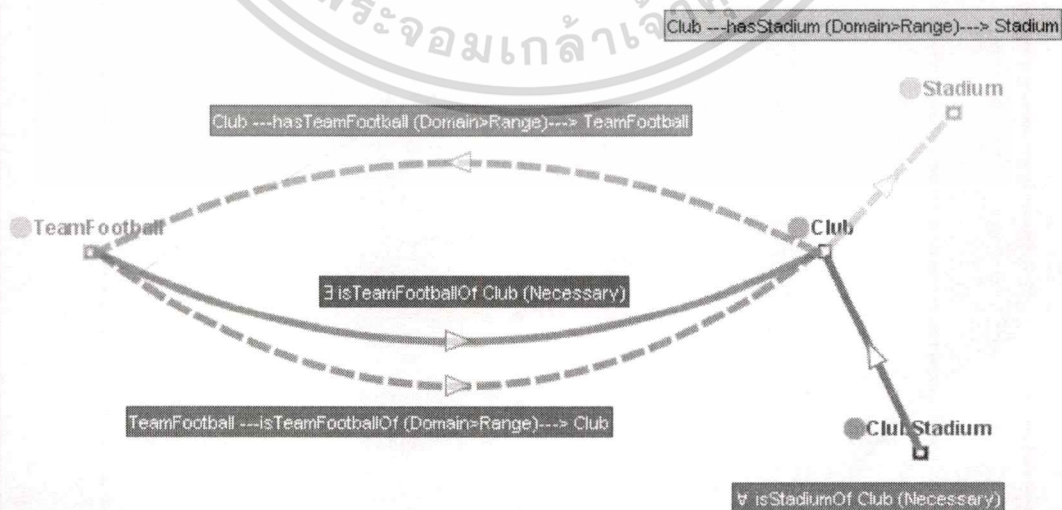
```

<owl:ObjectProperty rdf:about="#isStadiumOf">
  <rdfs:domain rdf:resource="#ClubStadium"/>
</owl:ObjectProperty>

```

รูปที่ 3.5 แสดง OWL ความสัมพันธ์ของคลาส ClubStadium

จากรายละเอียดของคลาส Stadium ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์โดยรวมได้ดังรูปที่ 3.6



รูปที่ 3.6 แสดงความสัมพันธ์โดยรวมของของคลาส Stadium

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 Match Play Class

เป็นคลาสของการแข่งขันภายในประเทศอังกฤษของทีมฟุตบอลระดับพรีเมียร์ลีก เก็บข้อมูลดังต่อไปนี้ ดัง OWL รูปที่ 3.7 และรูปที่ 3.8

1. วันที่ทำการแข่งขัน (Date)
2. เวลาที่ทำการแข่งขัน (Playtime)
3. ทีมแข่งขันที่เป็นทีมเหย้า หรือทีมเจ้าบ้าน (Home Team)
4. ทีมแข่งขันที่เป็นทีมเยือน (Away Team)

```

<owl:Class rdf:about="#MatchPlay">
  <owl:disjointWith rdf:resource="#TeamFootball"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Teamwork"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Club"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="date">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#date"/>
  <rdfs:domain rdf:resource="#MatchPlay"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="awayTeam">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#MatchPlay"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="homeTeam">
  <rdfs:domain rdf:resource="#MatchPlay"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="playtime">
  <rdfs:domain rdf:resource="#MatchPlay"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>

```

รูปที่ 3.7 แสดง OWL ของคลาส MatchPlay

และมีความสัมพันธ์กับคลาสต่างๆ เป็น Range ดังต่อไปนี้ ดัง OWL รูปที่ 3.8

- TeamFootball Class ให้มีความสัมพันธ์เป็น isMatchOf
- Stadium Class ให้มีความสัมพันธ์เป็น playAt
- Score Class ให้มีความสัมพันธ์เป็น hasScore

```

<owl:InverseFunctionalProperty rdf:about="#isMatchOf">
  <rdfs:range rdf:resource="#TeamFootball"/>
  <rdfs:subPropertyOf rdf:resource="#isDetailsOf"/>
  <owl:inverseOf rdf:resource="#hasMatch"/>
  <rdfs:domain rdf:resource="#MatchPlay"/>
  <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
</owl:InverseFunctionalProperty>

<owl:ObjectProperty rdf:ID="playAt">
  <rdfs:range rdf:resource="#Stadium"/>
  <rdfs:domain rdf:resource="#MatchPlay"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="hasScore">
  <rdfs:domain rdf:resource="#MatchPlay"/>
  <rdfs:range rdf:resource="#Score"/>
  <owl:inverseOf>
    <owl:InverseFunctionalProperty rdf:ID="isScoreOf"/>
  </owl:inverseOf>
</owl:ObjectProperty>

```

รูปที่ 3.8 แสดง OWL ความสัมพันธ์ของคลาส MathPlay

แบ่งเป็น 3 คลาสย่อยดังต่อไปนี้

3.2.2.1 FaPremierLeague Class

เป็นคลาสการแข่งขันพรีเมียร์ลีก ดัง OWL รูปที่ 3.9

```

<owl:Class rdf:ID="FaPremierLeague">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="isMatchOf"/>
      </owl:onProperty>
      <owl:allValuesFrom rdf:resource="#PremierTeam"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#MatchPlay"/>
</owl:Class>

```

รูปที่ 3.9 แสดง OWL ของคลาส FaPremierLeague

3.2.2.2 TheFootballLeagueCup Class

เป็นคลาสการแข่งขันถ้วยลีก ดัง OWL รูปที่ 3.10

```

<owl:Class rdf:ID="TheFootballLeagueCup">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="MatchPlay"/>
  </rdfs:subClassOf>
</owl:Class>

```

รูปที่ 3.10 แสดง OWL ของคลาส TheFootballLeagueCup

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

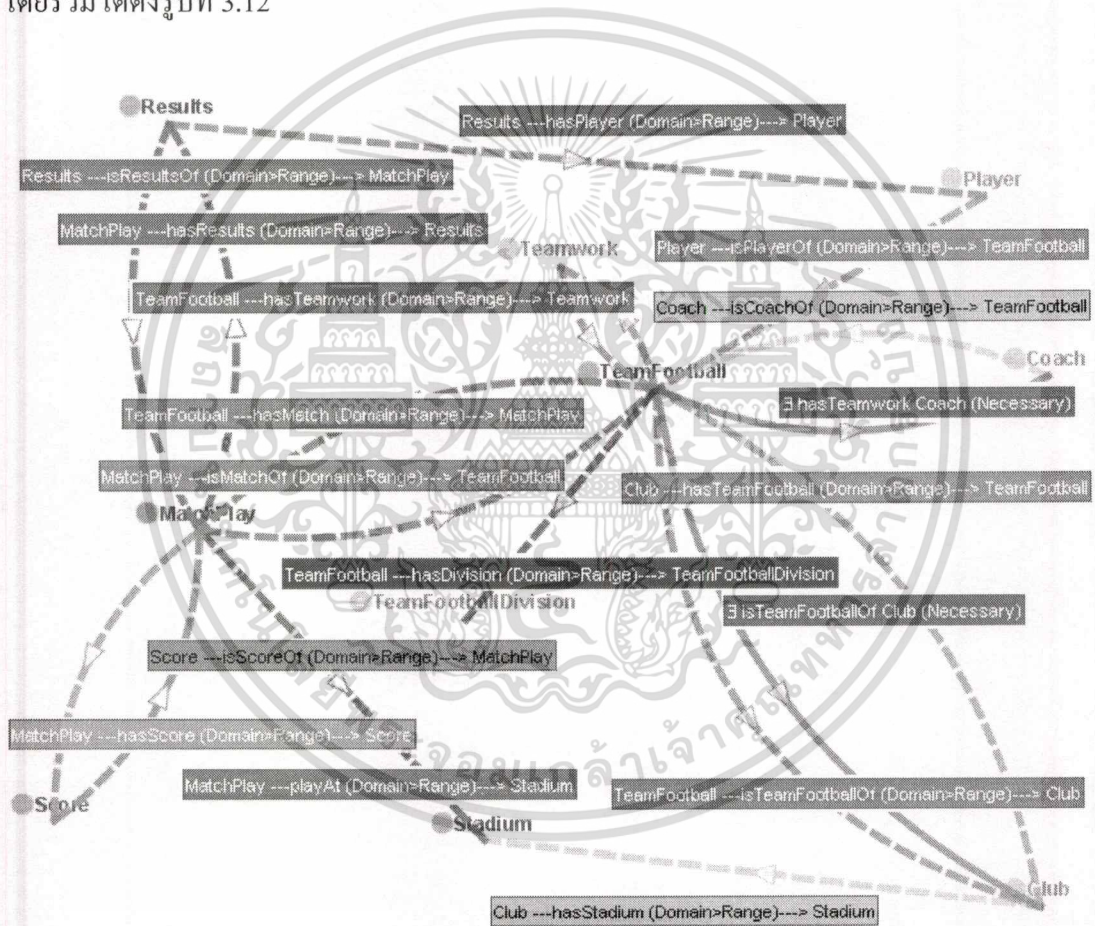
3.2.2.3 TheFaCup Class

เป็นคลาสการแข่งขันถ้วยเอฟเอ ดัง OWL รูปที่ 3.11

```
<owl:Class rdf:ID="TheFaCup">
  <rdfs:subClassOf rdf:resource="#MatchPlay"/>
</owl:Class>
```

รูปที่ 3.11 แสดง OWL ของคลาส TheFaCup

จากรายละเอียดของคลาส MatchPlay ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์โดยรวมได้ดังรูปที่ 3.12



รูปที่ 3.12 แสดงความสัมพันธ์โดยรวมของของคลาส MatchPlay

3.2.3 Result Class

เป็นคลาสเก็บรายละเอียดการแข่งขัน ทั้งที่เป็นการเตะจุดโทษ (Penalty Kick), การทำประตู(Goal), การทำฟาล์ว(Foul Play) และการเปลี่ยนตัวผู้เล่นระหว่างการแข่งขัน (Turn Player) ซึ่งมีเก็บข้อมูลดังนี้ ดัง OWL รูปที่ 3.13

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<owl:DatatypeProperty rdf:ID="atMinute">
  <rdfs:domain rdf:resource="#Results"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>
```

รูปที่ 3.13 แสดง OWL ของคลาส Results

และมีความสัมพันธ์กับคลาสต่างๆ เป็น Range ดังนี้ และได้ OWL ดังรูปที่ 3.14

- MatchPlay Class
ให้มีความสัมพันธ์เป็น isResultOf
- Player Class
ให้มีความสัมพันธ์เป็น hasPlayer

```
<owl:ObjectProperty rdf:about="#isResultsOf">
  <rdfs:domain rdf:resource="#Results"/>
  <rdfs:range rdf:resource="#MatchPlay"/>
  <owl:inverseOf rdf:resource="#hasResults"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="hasPlayer">
  <rdfs:domain rdf:resource="#Results"/>
  <rdfs:range rdf:resource="#Player"/>
</owl:ObjectProperty>
```

รูปที่ 3.14 แสดง OWL ของคลาส Results

แบ่งเป็น 4 คลาสย่อยดังนี้

3.2.3.1 PenaltyKick Class

เป็นคลาสของการเตะจุดโทษ ได้ OWL ดังรูปที่ 3.15

```
<owl:Class rdf:ID="PenaltyKick">
  <rdfs:subClassOf rdf:resource="#Results"/>
</owl:Class>
```

รูปที่ 3.15 แสดง OWL ของคลาส PenaltyKick

3.2.3.2 Goal Class

เป็นคลาสของการทำประตู ได้ OWL ดังรูปที่ 3.16

```
<owl:Class rdf:ID="Goal">
  <rdfs:subClassOf rdf:resource="#Results"/>
</owl:Class>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.16 แสดง OWL ของคลาส Goal ญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.3 FoulPlay Class

เป็นคลาสของการทำฟาล์วมีข้อมูลดังนี้ ดัง OWL รูปที่ 3.17

- ใบแดง/ใบเหลือง (Card)

```
<owl:Class rdf:ID="FoulPlay">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Results"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="card">
  <rdfs:domain rdf:resource="#FoulPlay"/>
  <rdfs:range>
    <owl:DataRange>
      <owl:oneOf rdf:parseType="Resource">
        <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
          yellow</rdf:first>
        <rdf:rest rdf:parseType="Resource">
          <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/>
          <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            red</rdf:first>
        </rdf:rest>
      </owl:oneOf>
    </owl:DataRange>
  </rdfs:range>
</owl:DatatypeProperty>
```

รูปที่ 3.17 แสดง OWL ของคลาส FoulPlay

3.2.3.4 TurnPlayer Class

เป็นคลาสของการเปลี่ยนตัวผู้เล่นมีข้อมูลดังนี้ ดัง OWL รูปที่ 3.18

- ผู้เล่นที่เปลี่ยนตัวเข้า (Turn-in Player)
- ผู้เล่นที่เปลี่ยนตัวออก (Turn-out Pla-yer)

```
<owl:Class rdf:ID="TurnPlayer">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Results"/>
  </rdfs:subClassOf>
</owl:Class>

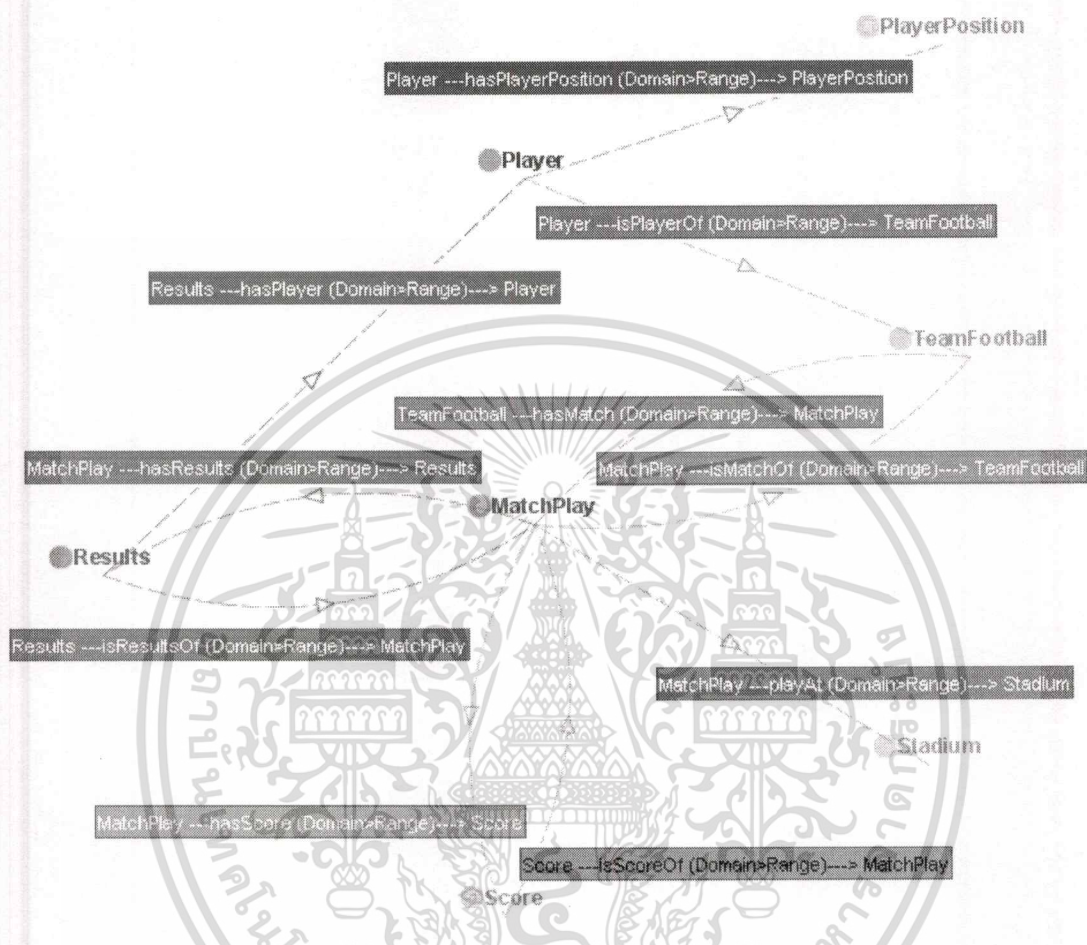
<owl:DatatypeProperty rdf:ID="turnOutPlayer">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#TurnPlayer"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="turnInPlayer">
  <rdfs:domain rdf:resource="#TurnPlayer"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.18 แสดง OWL ของคลาส TurnPlayer ให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรายละเอียดของคลาส Results ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์ โดยรวมได้ดังรูปที่ 3.19



รูปที่ 3.19 แสดงความสัมพันธ์โดยรวมของของคลาส Results

3.2.4 Teamwork Class

เป็นคลาสผู้ร่วมทีมฟุตบอล ทั้งที่เป็นโค้ชผู้ฝึก (Coach), ผู้จัดการทีมฟุตบอล (Manager) และนักกีฬาฟุตบอล (Player) โดยจะเก็บข้อมูลดังต่อไปนี้ ดัง OWL รูปที่ 3.20

1. ฉายา, ชื่อเล่น (Nickname)
2. ชื่อกลาง (Middlename)
3. นามสกุล (Lastname)
4. ชื่อต้น (Firstname)
5. สัญชาติ (Nationality)
6. วันเกิด (Birthdate)
7. อายุ (Age)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<owl:Class rdf:about="#Teamwork">
  <owl:disjointWith rdf:resource="#TeamFootball"/>
  <owl:disjointWith rdf:resource="#MatchPlay"/>
  <owl:disjointWith rdf:resource="#Club"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="nickName">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Teamwork"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="midname">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#Teamwork"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="lastname">
  <rdfs:domain rdf:resource="#Teamwork"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="firstname">
  <rdfs:domain rdf:resource="#Teamwork"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="nationality">
  <rdfs:domain rdf:resource="#Teamwork"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="birthdate">
  <rdfs:domain rdf:resource="#Teamwork"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#date"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="age">
  <rdfs:domain rdf:resource="#Teamwork"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>

```

รูปที่ 3.20 แสดง OWL ของคลาส Teamwork

และมีความสัมพันธ์กับคลาส TeamFootball เป็น isTeamworkOf ดัง OWL รูปที่ 3.21

```

<owl:ObjectProperty rdf:ID="isTeamworkOf">
  <rdfs:domain rdf:resource="#Teamwork"/>
  <rdfs:range rdf:resource="#TeamFootball"/>
</owl:ObjectProperty>

```

รูปที่ 3.21 แสดง OWL ของคลาส Teamwork

แบ่งเป็น 3 คลาสย่อยดังต่อไปนี้

3.2.4.1 Coach Class

เป็นคลาสของผู้ฝึก ซึ่งมีความสัมพันธ์กับคลาส TeamFootball เป็น isCoachOf
ดัง OWL รูปที่ 3.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<owl:Class rdf:ID="Coach">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Teamwork"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:InverseFunctionalProperty rdf:about="#isCoachOf">
  <owl:inverseOf rdf:resource="#hasTeamwork"/>
  <rdfs:domain rdf:resource="#Coach"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:range rdf:resource="#TeamFootball"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  <rdfs:subPropertyOf rdf:resource="#isDetailsOf"/>
</owl:InverseFunctionalProperty>

```

รูปที่ 3.22 แสดง OWL ของคลาส Coach

3.2.4.2 Coach Class

เป็นคลาสของผู้จัดการทีม ดัง OWL รูปที่ 3.23

```

<owl:Class rdf:ID="Manager">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Teamwork"/>
  </rdfs:subClassOf>
</owl:Class>

```

รูปที่ 3.23 แสดง OWL ของคลาส Manager

3.2.4.3 Player Class

เป็นคลาสของนักกีฬาฟุตบอล มีข้อมูลดังต่อไปนี้ ดัง OWL รูปที่ 3.24

- หมายเลขเสื้อกีฬา (Shirt Number)

```

<owl:Class rdf:about="#Player">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Teamwork"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:DatatypeProperty rdf:ID="shirtNumber">
  <rdfs:domain rdf:resource="#Player"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:DatatypeProperty>

<owl:InverseFunctionalProperty rdf:ID="isPlayerOf">
  <rdfs:domain rdf:resource="#Player"/>
  <rdfs:subPropertyOf rdf:resource="#isDetailsOf"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  <rdfs:range rdf:resource="#TeamFootball"/>
</owl:InverseFunctionalProperty>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.24 แสดง OWL ของคลาส Player ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และแบ่งผู้เล่นออกเป็น 4 คลาสย่อยดังนี้

3.2.4.3.1 ForwardPlayer Class

เป็นคลาสของผู้เล่นกองหน้า ดัง OWL รูปที่ 3.25

```

<owl:Class rdf:ID="ForwardPlayer">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Player"/>
        <owl:Restriction>
          <owl:someValuesFrom>
            <owl:Class>
              <owl:oneOf rdf:parseType="Collection">
                <PlayerPosition rdf:ID="Centre-forward"/>
                <ShortNamePosition rdf:ID="CF"/>
                <PlayerPosition rdf:ID="Deep-lying-forward"/>
                <ShortNamePosition rdf:ID="S"/>
                <ShortNamePosition rdf:ID="SS"/>
                <PlayerPosition rdf:ID="Striker"/>
              </owl:oneOf>
            </owl:Class>
          </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#hasPlayerPosition"/>
        </owl:onProperty>
      </owl:Restriction>
    </owl:intersectionOf>
  </owl:Class>
</owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.25 แสดง OWL ของคลาส ForwardPlayer

3.2.4.3.2 MidfieldPlayer Class

เป็นคลาสของผู้เล่นกองกลาง ดัง OWL รูปที่ 3.26

```

<owl:Class rdf:ID="MidfieldPlayer">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Player"/>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#hasPlayerPosition"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class>
              <owl:oneOf rdf:parseType="Collection">
                <ShortNamePosition rdf:ID="AM"/>
                <PlayerPosition rdf:about="#Attacking-midfielder"/>
                <PlayerPosition rdf:about="#Centre-midfield"/>
                <ShortNamePosition rdf:ID="CM"/>
                <PlayerPosition rdf:about="#Defensive-midfielder"/>
                <ShortNamePosition rdf:ID="DM"/>
                <PlayerPosition rdf:about="#Left-midfielder"/>
                <ShortNamePosition rdf:ID="LM"/>
                <PlayerPosition rdf:about="#Right-midfielder"/>
                <ShortNamePosition rdf:ID="RM"/>
                <PlayerPosition rdf:about="#Side-midfielder"/>
              </owl:oneOf>
            </owl:Class>
          </owl:someValuesFrom>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.26 แสดง OWL ของคลาส MidfieldPlayer

3.2.4.3.3 GoalkeeperPlayer Class

เป็นคลาสของผู้รักษาประตู ดัง OWL รูปที่ 3.27

```

<owl:Class rdf:ID="GoalkeeperPlayer">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#hasPlayerPosition"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class>
              <owl:oneOf rdf:parseType="Collection">
                <PlayerPosition rdf:about="#Goalkeeper"/>
                <ShortNamePosition rdf:ID="GK"/>
              </owl:oneOf>
            </owl:Class>
          </owl:someValuesFrom>
        </owl:Restriction>
        <owl:Class rdf:about="#Player"/>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.27 แสดง OWL ของคลาส GoalkeeperPlayer



3.2.4.3.4 DefensivePlayer Class

เป็นคลาสของผู้รักษาประตู ค้าง OWL รูปที่ 3.28

```

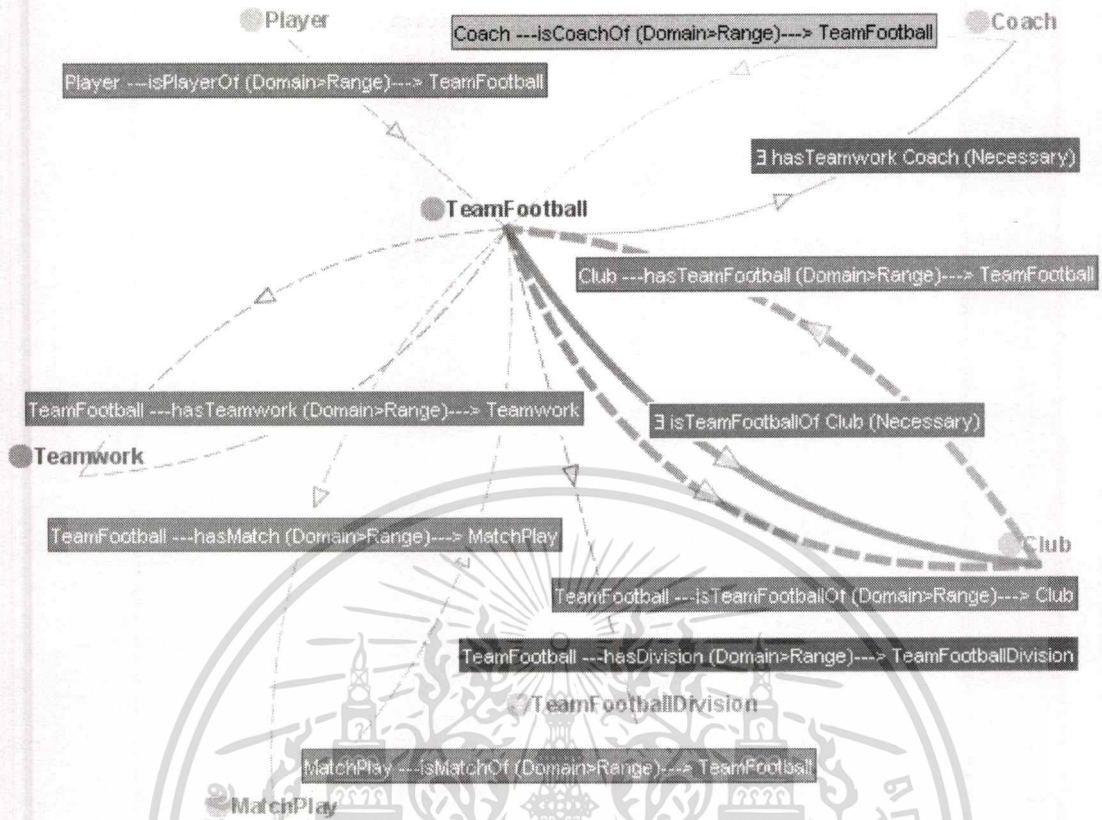
<owl:Class rdf:ID="DefensivePlayer">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="Player"/>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:ID="hasPlayerPosition"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class>
              <owl:oneOf rdf:parseType="Collection">
                <ShortNamePosition rdf:ID="CB"/>
                <PlayerPosition rdf:ID="Centre-back"/>
                <ShortNamePosition rdf:ID="FB"/>
                <PlayerPosition rdf:ID="Full-back"/>
                <ShortNamePosition rdf:ID="LB"/>
                <PlayerPosition rdf:ID="Left-back"/>
                <ShortNamePosition rdf:ID="Libero"/>
                <ShortNamePosition rdf:ID="RB"/>
                <PlayerPosition rdf:ID="Right-back"/>
                <ShortNamePosition rdf:ID="Stopper"/>
                <ShortNamePosition rdf:ID="SW"/>
                <PlayerPosition rdf:ID="Sweeper"/>
                <PlayerPosition rdf:ID="Left-wingback"/>
                <ShortNamePosition rdf:ID="LWB"/>
                <PlayerPosition rdf:ID="Right-wingback"/>
                <ShortNamePosition rdf:ID="RWB"/>
                <ShortNamePosition rdf:ID="WB"/>
                <PlayerPosition rdf:ID="Wingback"/>
              </owl:oneOf>
            </owl:Class>
          </owl:someValuesFrom>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.28 แสดง OWL ของคลาส DefensivePlayer

จากรายละเอียดของคลาส Teamwork ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์
โดยรวมได้ดังรูปที่ 3.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.29 แสดงความสัมพันธ์โดยรวมของของคลาส Teamwork

3.2.5 Score Class

เป็นคลาสเก็บผลการแข่งขันการทำประตู โดยจะเก็บข้อมูลดังต่อไปนี้ ดัง OWL รูปที่

3.30

1. ทีมที่แข่งขัน (MatchTeam)
2. ผลการแข่งขัน (Score)
3. ทีมแข่งขันที่เป็นทีมเหย้า หรือทีมเจ้าบ้าน (Home Team)
4. ทีมแข่งขันที่เป็นทีมเยือน (Away Team)

```
<owl:Class rdf:ID="Score"/>
```

```
<owl:DatatypeProperty rdf:ID="score">
```

```
<rdfs:domain rdf:resource="#Score"/>
```

```
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
```

```
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="matchTeam">
```

```
<rdfs:domain rdf:resource="#Score"/>
```

```
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
```

```
</owl:DatatypeProperty>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 3.30 แสดง OWL ของคลาส Score

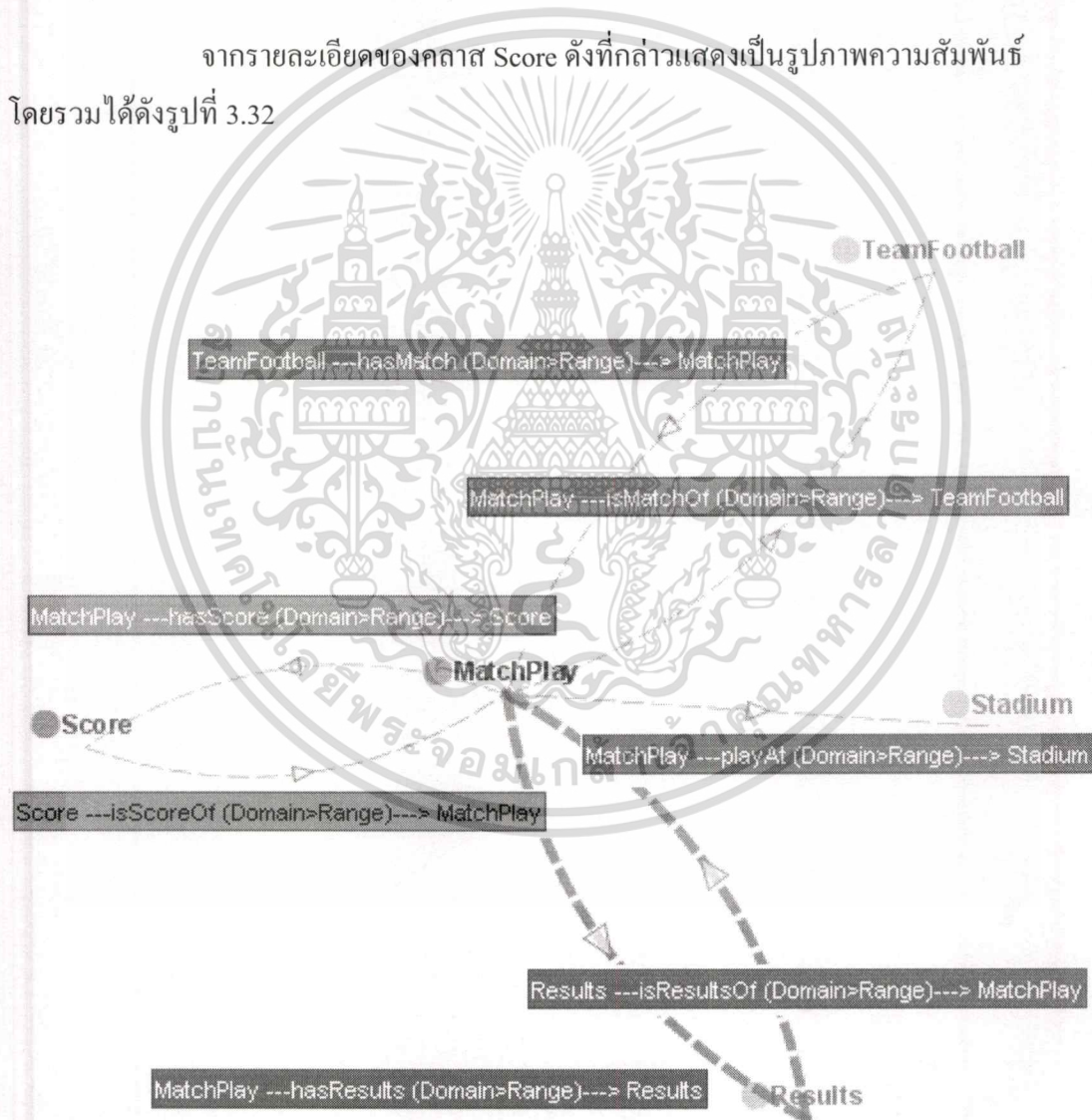
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีความสัมพันธ์กับคลาส MatchPlay เป็น isScoreOf ดัง OWL รูปที่ 3.31

```
<owl:InverseFunctionalProperty rdf:about="#isScoreOf">
  <rdfs:range rdf:resource="#MatchPlay"/>
  <rdfs:domain rdf:resource="#Score"/>
  <owl:inverseOf rdf:resource="#hasScore"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
</owl:InverseFunctionalProperty>
```

รูปที่ 3.31 แสดง OWL ของคลาส Score

จากรายละเอียดของคลาส Score ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์ โดยรวมได้ดังรูปที่ 3.32



รูปที่ 3.32 แสดงความสัมพันธ์โดยรวมของของคลาส Score

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 Team Football Class

เป็นคลาสของทีมฟุตบอลในประเทศไทยซึ่งแบ่งระดับไว้เป็น 4 ระดับ (Division) หลักๆ โดยจะเก็บข้อมูลดังต่อไปนี้ ดัง OWL รูปที่ 3.33

1. ชื่อทีมฟุตบอล (Team Name)

```
<owl:Class rdf:about="#TeamFootball">
  <owl:disjointWith rdf:resource="#Club"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="hasTeamwork"/>
      </owl:onProperty>
      <owl:someValuesFrom rdf:resource="#Coach"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="isTeamFootballOf"/>
      </owl:onProperty>
      <owl:someValuesFrom rdf:resource="#Club"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <owl:disjointWith>
    <owl:Class rdf:about="#Teamwork"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#MatchPlay"/>
  </owl:disjointWith>
</owl:Class>

<owl:DatatypeProperty rdf:ID="teamName">
  <rdfs:domain rdf:resource="#TeamFootball"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
```

รูปที่ 3.33 แสดง OWL ของคลาส TeamFootball

และมีความสัมพันธ์กับคลาสต่างๆ เป็น Range ดังต่อไปนี้

- TeamFootballDivision Class

ให้มีความสัมพันธ์เป็น hasDivision ดัง OWL รูปที่ 3.34

```
<owl:ObjectProperty rdf:about="#hasDivision">
  <rdfs:domain rdf:resource="#TeamFootball"/>
  <rdfs:range rdf:resource="#TeamFootballDivision"/>
  <rdfs:subPropertyOf>
    <owl:TransitiveProperty rdf:about="#hasDetails"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
```

รูปที่ 3.34 แสดง OWL ของคลาส TeamFootball

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Club Class

ให้มีความสัมพันธ์เป็น isTeamFootballOf ดัง OWL รูปที่ 3.35

```
<owl:ObjectProperty rdf:about="#isTeamFootballOf">
  <rdfs:domain rdf:resource="#TeamFootball"/>
  <rdfs:range rdf:resource="#Club"/>
  <rdfs:subPropertyOf>
    <owl:TransitiveProperty rdf:ID="hasDetails"/>
  </rdfs:subPropertyOf>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="hasTeamFootball"/>
  </owl:inverseOf>
</owl:ObjectProperty>
```

รูปที่ 3.35 แสดง OWL ของคลาส TeamFootball

- Teamwork Class

ให้มีความสัมพันธ์เป็น hasTeamwork ดัง OWL รูปที่ 3.36

```
<owl:ObjectProperty rdf:about="#hasTeamwork">
  <rdfs:range rdf:resource="#Teamwork"/>
  <owl:inverseOf>
    <owl:InverseFunctionalProperty rdf:ID="isCoachOf"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#TeamFootball"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#InverseFunctionalProperty"/>
  <rdfs:subPropertyOf>
    <owl:TransitiveProperty rdf:about="#hasDetails"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
```

รูปที่ 3.36 แสดง OWL ของคลาส TeamFootball

- MatchPlay Class

ให้มีความสัมพันธ์เป็น hasMatch ดัง OWL รูปที่ 3.37

```
<owl:ObjectProperty rdf:ID="hasMatch">
  <rdfs:range rdf:resource="#MatchPlay"/>
  <rdfs:domain rdf:resource="#TeamFootball"/>
  <owl:inverseOf>
    <owl:InverseFunctionalProperty rdf:about="#isMatchOf"/>
  </owl:inverseOf>
  <rdfs:subPropertyOf>
    <owl:TransitiveProperty rdf:about="#hasDetails"/>
  </rdfs:subPropertyOf>
</owl:ObjectProperty>
```

รูปที่ 3.37 แสดง OWL ของคลาส TeamFootball

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และแบ่งทีมฟุตบอลออกเป็น 4 คลาสย่อยตามระดับ ดังต่อไปนี้ ดัง OWL รูปที่ 3.38, รูปที่ 3.39, รูปที่ 3.40 และรูปที่ 3.41

3.2.6.1 PremierTeam Class

เป็นทีมฟุตบอลที่อยู่ระดับพรีเมียร์ ซึ่งแยกจากกันกับระดับอื่น ดัง OWL รูปที่ 3.38

```

<owl:Class rdf:ID="PremierTeam">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:ID="TeamFootball"/>
        <owl:Restriction>
          <owl:hasValue>
            <TeamFootballDivision rdf:ID="Premier"/>
          </owl:hasValue>
          <owl:onProperty>
            <owl:ObjectProperty rdf:ID="hasDivision"/>
          </owl:onProperty>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
  <owl:disjointWith>
    <owl:Class rdf:ID="ChampionshipTeam"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:ID="LeagueOneTeam"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:ID="LeagueTwoTeam"/>
  </owl:disjointWith>
</owl:Class>

```

รูปที่ 3.38 แสดง OWL ของคลาส PremierTeam

3.2.6.2 ChampionshipTeam Class

เป็นคลาสทีมฟุตบอลระดับแชมป์เปียนชิพ ซึ่งแยกกับระดับอื่น ดัง OWL รูปที่ 3.39

```

<owl:Class rdf:about="#ChampionshipTeam">
  <owl:disjointWith rdf:resource="#LeagueTwoTeam"/>
  <owl:disjointWith rdf:resource="#PremierTeam"/>
  <owl:disjointWith rdf:resource="#LeagueOneTeam"/>
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#TeamFootball"/>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#hasDivision"/>
          </owl:onProperty>
          <owl:hasValue rdf:resource="#Championship"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.39 แสดง OWL ของคลาส ChampionshipTeam

3.2.6.3 LeagueOneTeam Class

เป็นคลาสทีมฟุตบอลระดับลีกหนึ่ง ซึ่งแยกกับระดับอื่น ดัง OWL รูปที่ 3.40

```

<owl:Class rdf:about="#LeagueOneTeam">
  <owl:disjointWith>
    <owl:Class rdf:about="#ChampionshipTeam"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#PremierTeam"/>
  <owl:disjointWith rdf:resource="#LeagueTwoTeam"/>
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#TeamFootball"/>
        <owl:Restriction>
          <owl:hasValue rdf:resource="#One"/>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#hasDivision"/>
          </owl:onProperty>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.40 แสดง OWL ของคลาส LeagueOneTeam

3.2.6.4 LeagueTwoTeam Class

เป็นคลาสทีมฟุตบอลระดับลีกสองซึ่งแยกกับระดับอื่น ดัง OWL รูปที่ 3.41

```

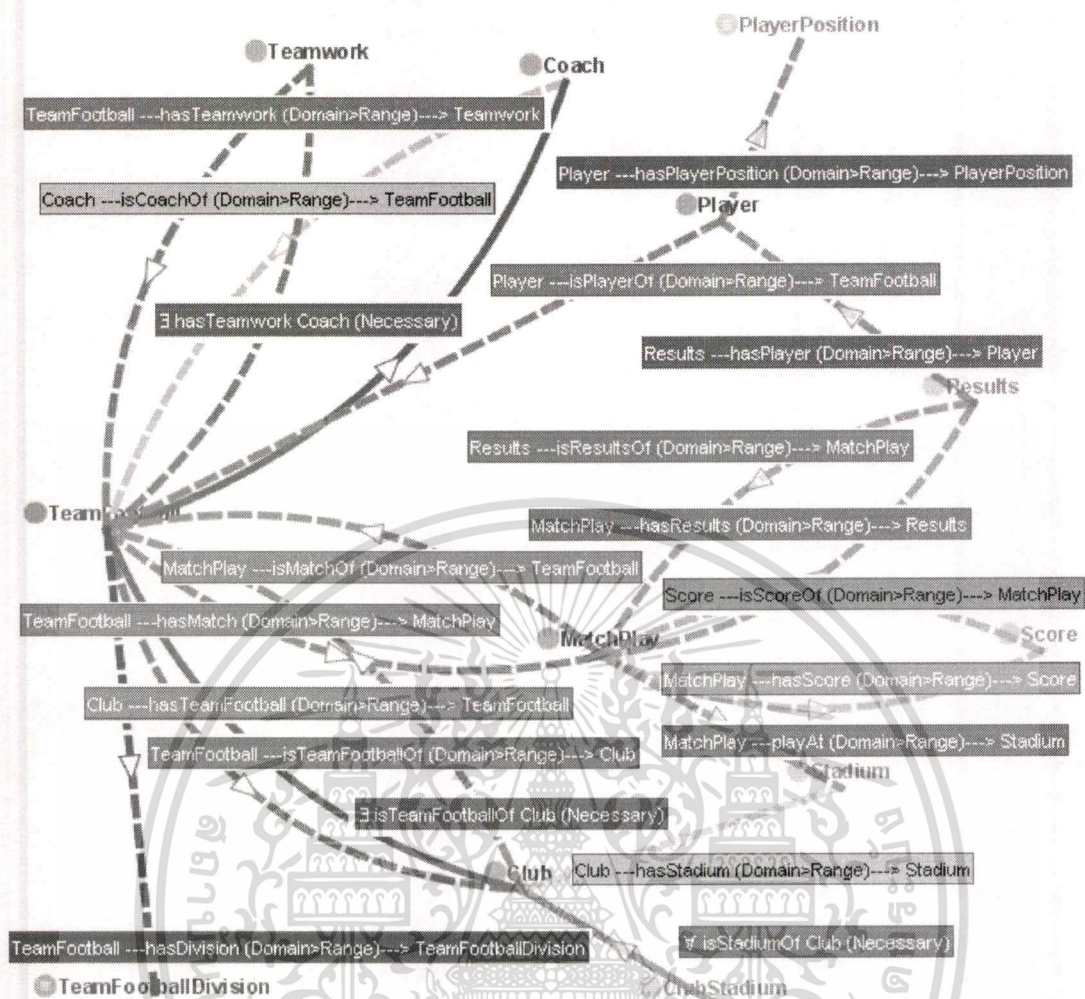
<owl:Class rdf:about="#LeagueTwoTeam">
  <owl:disjointWith>
    <owl:Class rdf:about="#LeagueOneTeam"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#PremierTeam"/>
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#TeamFootball"/>
        <owl:Restriction>
          <owl:hasValue rdf:resource="#Two"/>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#hasDivision"/>
          </owl:onProperty>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
  <owl:disjointWith>
    <owl:Class rdf:about="#ChampionshipTeam"/>
  </owl:disjointWith>
</owl:Class>

```

รูปที่ 3.41 แสดง OWL ของคลาส LeagueTwoTeam

จากรายละเอียดของคลาส Score ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์

โดยรวมได้ดังรูปที่ 3.42



รูปที่ 3.42 แสดงความสัมพันธ์โดยรวมของของคลาส Team Football

3.2.7 Club Class

เป็นคลาสสโมสรของทีมฟุตบอล โดยจะเก็บข้อมูลดังต่อไปนี้ ดัง OWL รูปที่ 3.43 และ รูปที่ 3.44

1. ชื่อสโมสร (Club Name)
2. อายุสโมสร (Club Age)
3. ที่อยู่สโมสร (Club Address)
4. ขนาดของสโมสร (Club Size)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<owl:Class rdf:ID="Club">
  <owl:disjointWith>
    <owl:Class rdf:ID="Teamwork"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:ID="TeamFootball"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:ID="MatchPlay"/>
  </owl:disjointWith>
</owl:Class>

```

รูปที่ 3.43 แสดง OWL ของคลาส Club

```

<owl:DatatypeProperty rdf:ID="clubName">
  <rdfs:domain rdf:resource="#Club"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="clubAge">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdfs:domain rdf:resource="#Club"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="clubAddress">
  <rdfs:domain rdf:resource="#Club"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

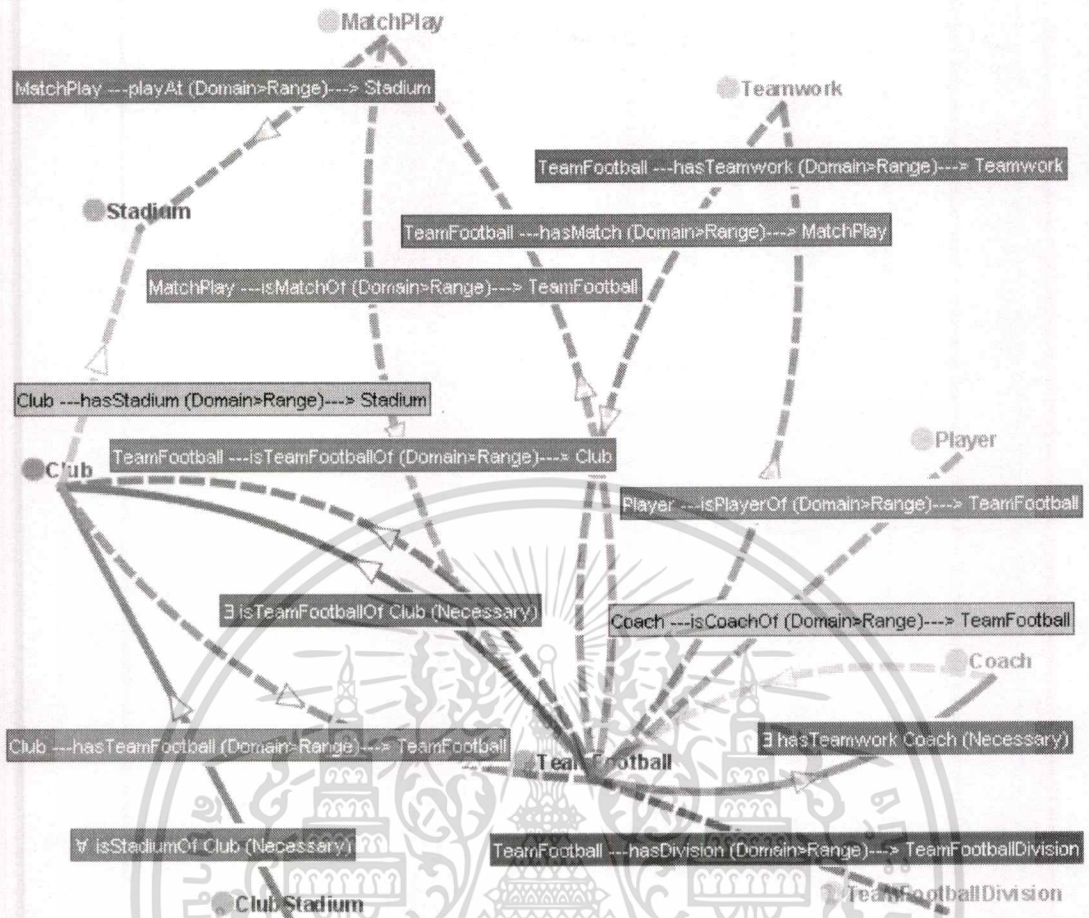
<owl:DatatypeProperty rdf:ID="clubSize">
  <rdfs:range>
    <owl:DataRange>
      <owl:oneOf rdf:parseType="Resource">
        <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
          >1</rdf:first>
        <rdf:rest rdf:parseType="Resource">
          <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
            >2</rdf:first>
          <rdf:rest rdf:parseType="Resource">
            <rdf:rest rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#nil"/>
            <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
              >3</rdf:first>
          </rdf:rest>
        </rdf:rest>
      </owl:oneOf>
    </owl:DataRange>
  </rdfs:range>
  <rdfs:domain rdf:resource="#Club"/>
</owl:DatatypeProperty>

```

รูปที่ 3.44 แสดง OWL ของคลาส Club

จากรายละเอียดของคลาส Club ดังที่กล่าวแสดงเป็นรูปภาพความสัมพันธ์
โดยรวมได้ดังรูปที่ 3.45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.45 แสดงความสัมพันธ์โดยรวมของของคลาส Club

3.2.8 Team Football Division

เป็นคลาสที่กำหนดค่าของระดับทีมฟุตบอล ซึ่งมี 4 ระดับ คือ Premier, Championship, One, Two ดัง OWL รูปที่ 3.46

```

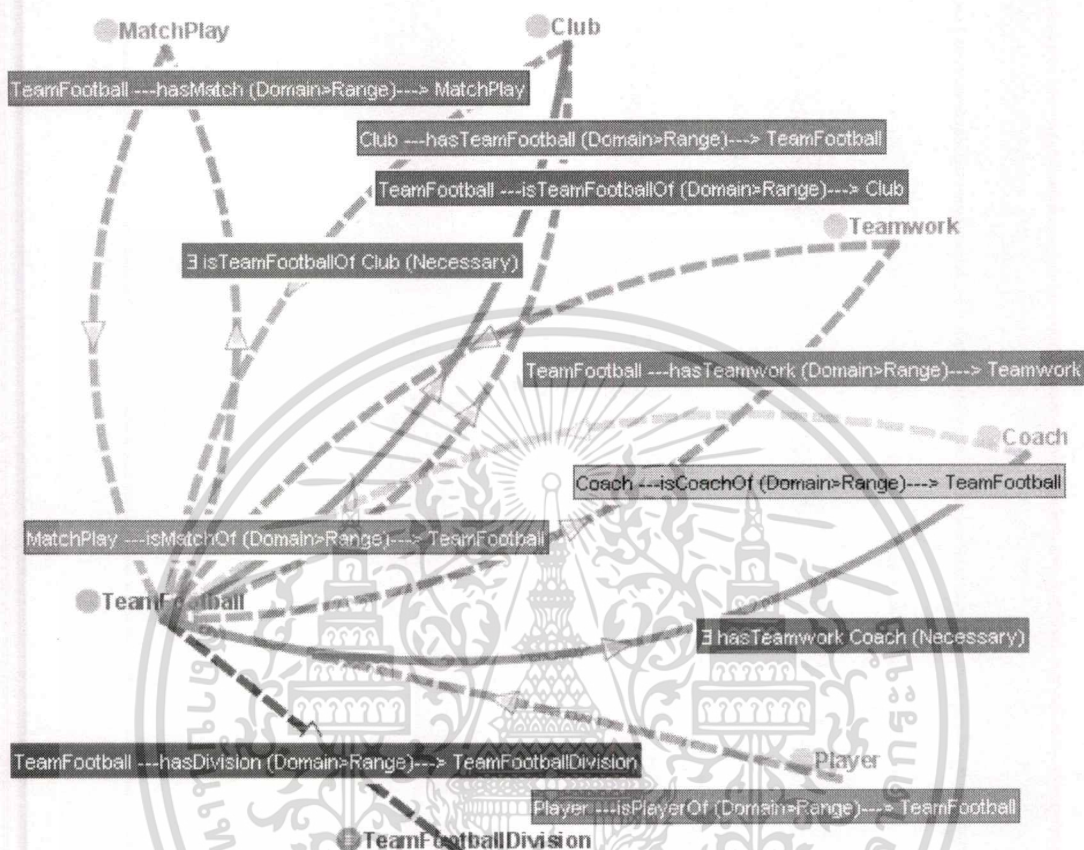
<owl:Class rdf:ID="TeamFootballDivision">
  <owl:equivalentClass>
    <owl:Class>
      <owl:oneOf rdf:parseType="Collection">
        <TeamFootballDivision rdf:about="#Premier"/>
        <TeamFootballDivision rdf:ID="Championship"/>
        <TeamFootballDivision rdf:ID="Two"/>
        <TeamFootballDivision rdf:ID="One"/>
      </owl:oneOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.46 แสดง OWL ของคลาส TeamFootballDivision

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรายละเอียดของคลาส TeamFootball Division ดังที่กล่าวแสดงเป็นรูปภาพ ความสัมพันธ์โดยรวมได้ดังรูปที่ 3.47



รูปที่ 3.47 แสดงความสัมพันธ์โดยรวมของของคลาส TeamFootball Division

3.2.9 Player Position Class

เป็นคลาสของตำแหน่งผู้เล่นในทีมฟุตบอล ซึ่งจะมีค่าตำแหน่งผู้เล่นทั้งหมดที่มี ดัง OWL

รูปที่ 3.48

```

<owl:ObjectProperty rdf:about="#hasPlayerPosition">
  <rdfs:range rdf:resource="#PlayerPosition"/>
  <rdfs:domain rdf:resource="#Player"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="PlayerPosition">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:someValuesFrom>
        <owl:Class>
          <owl:oneOf rdf:parseType="Collection">
            <PlayerPosition rdf:ID="Attacking-midfielder"/>
            <PlayerPosition rdf:about="#Centre-back"/>
            <PlayerPosition rdf:about="#Centre-forward"/>
            <PlayerPosition rdf:ID="Centre-midfield"/>
            <PlayerPosition rdf:about="#Deep-lying-forward"/>
            <PlayerPosition rdf:ID="Defensive-midfielder"/>
            <PlayerPosition rdf:about="#Full-back"/>
            <PlayerPosition rdf:ID="Goalkeeper"/>
            <PlayerPosition rdf:about="#Left-back"/>
            <PlayerPosition rdf:ID="Left-midfielder"/>
            <PlayerPosition rdf:about="#Left-wingback"/>
            <PlayerPosition rdf:ID="Left-winger"/>
            <PlayerPosition rdf:about="#Liberi"/>
            <PlayerPosition rdf:about="#Right-back"/>
            <PlayerPosition rdf:ID="Right-midfielder"/>
            <PlayerPosition rdf:about="#Right-wingback"/>
            <PlayerPosition rdf:ID="Right-winger"/>
            <PlayerPosition rdf:ID="Side-midfielder"/>
            <PlayerPosition rdf:about="#Stopper"/>
            <PlayerPosition rdf:about="#Striker"/>
            <PlayerPosition rdf:about="#Sweeper"/>
            <PlayerPosition rdf:about="#Wingback"/>
            <PlayerPosition rdf:ID="Winger"/>
          </owl:oneOf>
        </owl:Class>
      </owl:someValuesFrom>
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>

```

รูปที่ 3.48 แสดง OWL ของคลาส PlayerPosition

และมีความสัมพันธ์กับคลาสย่อยของตำแหน่งผู้เล่นในทีมฟุตบอล ดัง OWL รูปที่ 3.49

```

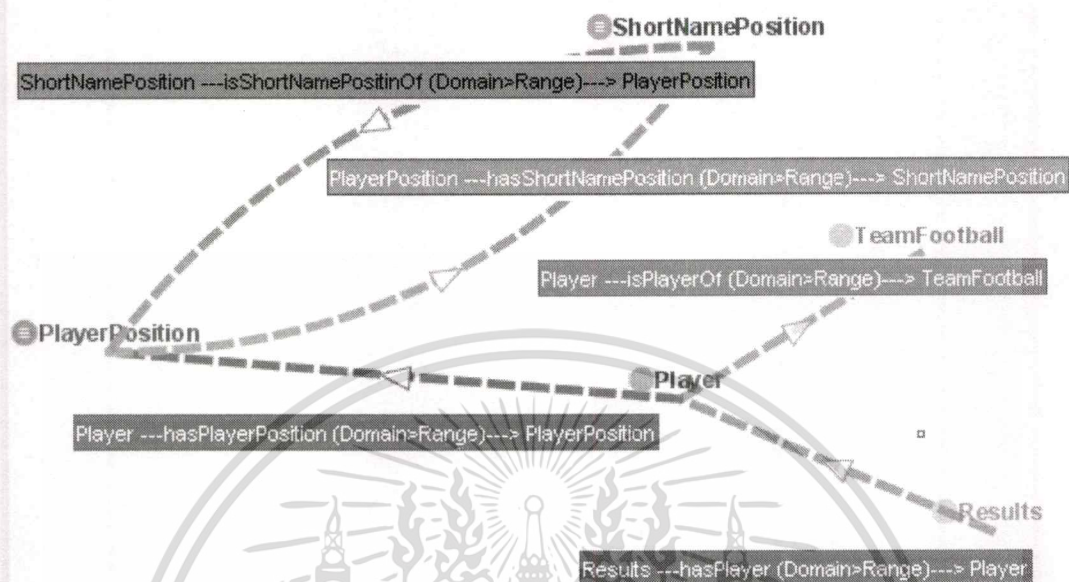
<owl:ObjectProperty rdf:about="#hasShortNamePosition">
  <rdfs:domain rdf:resource="#PlayerPosition"/>
  <rdfs:range rdf:resource="#ShortNamePosition"/>
  <owl:inverseOf rdf:resource="#isShortNamePosition"/>
</owl:ObjectProperty>

```

รูปที่ 3.49 แสดง OWL ของคลาส PlayerPosition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรายละเอียดของคลาส PlayerPosition ดังที่กล่าวแสดงเป็นรูปภาพ ความสัมพันธ์โดยรวมได้ดังรูปที่ 3.50



รูปที่ 3.50 แสดงความสัมพันธ์โดยรวมของของคลาส PlayerPosition

3.2.10 Short Name Position Class

เป็นคลาสชื่อย่อของตำแหน่งผู้เล่นในทีมฟุตบอล ซึ่งจะมีค่าชื่อย่อของตำแหน่งผู้เล่นทั้งหมดที่มี ดัง OWL รูปที่ 3.51

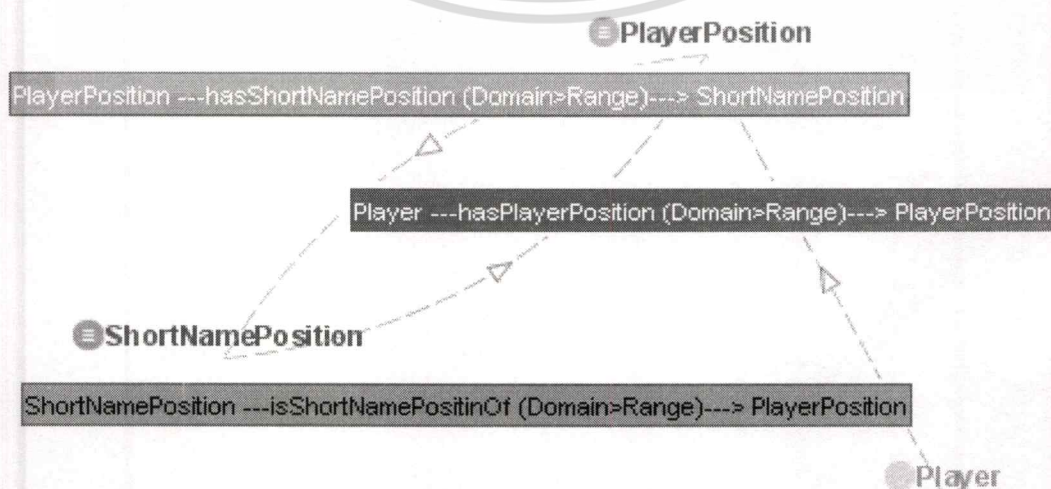
```

<owl:Class rdf:ID="ShortNamePosition">
  <owl:equivalentClass>
    <owl:Class>
      <owl:oneOf rdf:parseType="Collection">
        <ShortNamePosition rdf:about="#AM"/>
        <ShortNamePosition rdf:about="#CB"/>
        <ShortNamePosition rdf:about="#CF"/>
        <ShortNamePosition rdf:about="#CM"/>
        <ShortNamePosition rdf:about="#DM"/>
        <ShortNamePosition rdf:about="#FB"/>
        <ShortNamePosition rdf:about="#GK"/>
        <ShortNamePosition rdf:about="#LB"/>
        <ShortNamePosition rdf:about="#LM"/>
        <ShortNamePosition rdf:ID="LW"/>
        <ShortNamePosition rdf:about="#LWB"/>
        <ShortNamePosition rdf:about="#RB"/>
        <ShortNamePosition rdf:about="#RM"/>
        <ShortNamePosition rdf:ID="RW"/>
        <ShortNamePosition rdf:about="#RWB"/>
        <ShortNamePosition rdf:about="#S"/>
        <ShortNamePosition rdf:about="#SS"/>
        <ShortNamePosition rdf:about="#SW"/>
        <ShortNamePosition rdf:about="#WB"/>
      </owl:oneOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
<owl:ObjectProperty rdf:ID="isShortNamePositionOf">
  <rdf:range rdf:resource="#PlayerPosition"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="hasShortNamePosition"/>
  </owl:inverseOf>
  <rdf:domain rdf:resource="#ShortNamePosition"/>
</owl:ObjectProperty>

```

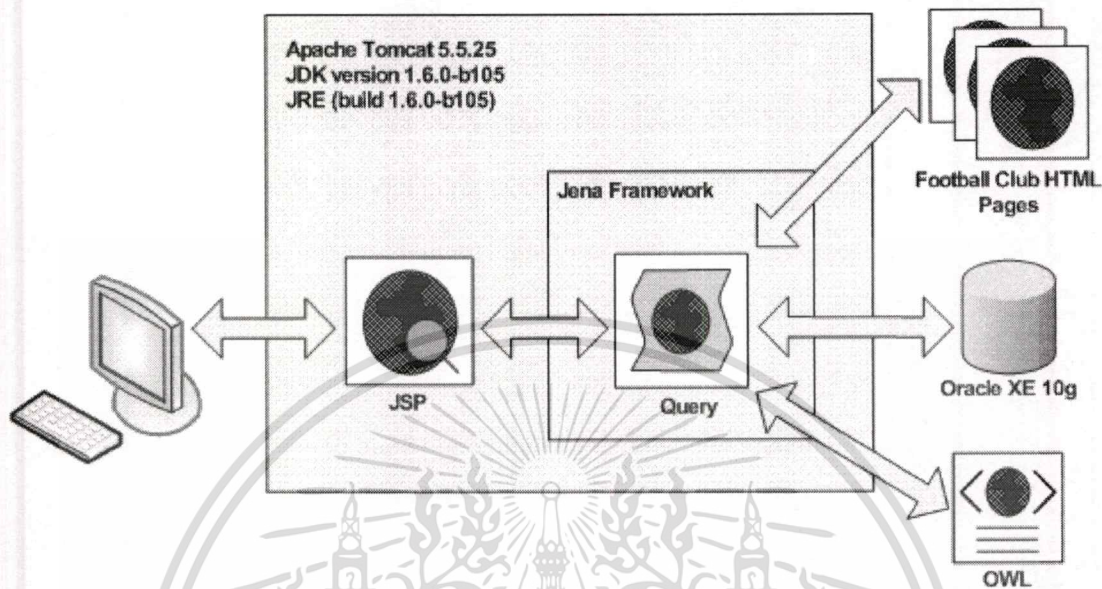
รูปที่ 3.51 แสดง OWL ของคลาส PlayerPosition

จากรายละเอียดของคลาส ShortNamePosition ดังที่กล่าวแสดงเป็นรูปภาพ ความสัมพันธ์โดยรวมได้ดังรูปที่ 3.52



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆที่ผู้อื่นทำซ้ำโดยไม่ขออนุญาต หรือเผยแพร่โดยไม่ได้รับอนุญาตของเจ้าของลิขสิทธิ์ที่มีการนำไปใช้

3.3 การออกแบบโครงสร้างสถาปัตยกรรมของซีแมนติกเว็บที่มีเนื้อหากีฬาฟุตบอล (Football Semantic Web Architecture)



รูปที่ 3.53 แสดงโครงสร้างสถาปัตยกรรม

จากรูปที่ 3.53 แสดงให้เห็นว่าระบบได้ออกแบบและพัฒนาในลักษณะของโปรแกรมประยุกต์บนเว็บ โดยทำงานภายใต้สภาพแวดล้อมของเซิร์ฟเวอร์ Apache Tomcat 5.5.25 และ JRE 1.6.0-b105 ตัวโปรแกรมประยุกต์สามารถแบ่งออกเป็นส่วน ๆ ได้แก่

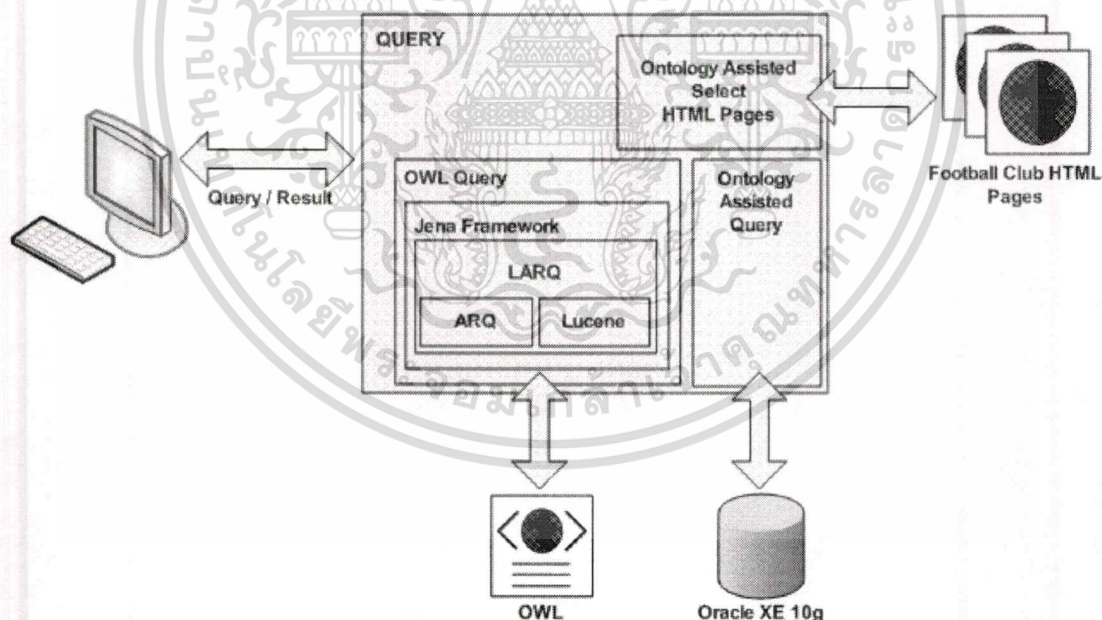
1. ส่วนติดต่อผู้ใช้งาน (User Interface) ได้จัดทำเป็นหน้าเว็บด้วยภาษา JSP ทำหน้าที่รับเอาเงื่อนไขหรือคำค้นที่ผู้ใช้งานต้องการ ไปส่งต่อให้กับส่วนที่ใช้ในการค้นหา และรับเอาผลลัพธ์ที่ได้จากส่วนที่ใช้ในการค้นหามาจัดแสดงให้กับผู้ใช้งาน
2. ส่วนที่ใช้ในการค้นหาข้อมูลที่เกี่ยวข้องกับฟุตบอล เป็นส่วนที่รับเอาเงื่อนไขหรือคำค้นจากส่วนที่ 1 เข้ามาประมวลผล และค้นหาข้อมูลจากทั้งฐานข้อมูลหรือไฟล์ HTML ที่เกี่ยวข้อง จากนั้นจึงส่งผลลัพธ์ออกไป เพื่อนำเอาไปแสดงผลยังหน้าเว็บต่อไป
3. ส่วนจัดเก็บข้อมูล สามารถแบ่งย่อยได้เป็น 3 ส่วนได้แก่
 - a. ไฟล์ HTML เป็นไฟล์ HTML ที่เก็บข้อมูลของสโมสรฟุตบอลในประเทศอังกฤษ
 - b. ฐานข้อมูล ใช้จัดเก็บข้อมูลที่เกี่ยวข้องกับฟุตบอลในประเทศอังกฤษ
 - c. ไฟล์ OWL เป็นไฟล์ที่จัดเก็บข้อมูลและองค์ความรู้ (Ontology) ที่เกี่ยวกับฟุตบอลในประเทศอังกฤษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับรายละเอียดในการพัฒนาส่วนที่ 1 นั้น เนื่องจากการพัฒนาหน้าเว็บตามมาตรฐาน JSP ดังนั้นจะไม่ขอกล่าวรายละเอียดในการพัฒนาในที่นี้ โดยสามารถศึกษาเพิ่มเติมได้จาก <http://java.sun.com/products/jsp/>

ในการพัฒนาส่วนที่ใช้ในการค้นหาข้อมูลที่เกี่ยวข้องกับฟุตบอลนั้น จะใช้ภาษา Java ในการพัฒนา โดยมีการนำเอา Jena เข้ามาช่วยในการพัฒนาเนื่องจากว่า Jena นั้นถูกพัฒนาขึ้นมาสำหรับการทำ Semantic Web โดยเฉพาะ มี API ที่เหมาะสมและยืดหยุ่นสูง ทั้งยังสนับสนุนการค้นหาข้อมูลใน OWL ด้วยภาษาที่ใช้สำหรับการค้นหาอย่าง SPARQL โดยสามารถใช้งานผ่านทาง API ใน package ARQ

แม้ว่าเราจะสามารถค้นหาผ่านทาง API ใน ARQ ได้ก็ตาม แต่จะพบว่า ยังคงมีข้อจำกัดในการค้นหาบางเงื่อนไข เช่นการค้นหาค่าขึ้นต้นด้วย หรือลงท้ายด้วย หรือการค้นหาข้อความใน OWL เป็นต้น แต่อย่างไรก็ตาม ผู้พัฒนาในส่วนของ ARQ นั้นได้พัฒนาให้สามารถรองรับ plug-ins เข้ามาช่วยเพิ่มขีดความสามารถให้มากยิ่งขึ้น นั่นคือ ARQ ได้นำเอาเทคโนโลยี Lucene เข้ามาช่วยเหลือ ทำให้ลดข้อจำกัดเหล่านั้นลงไปได้



รูปที่ 3.54 แสดงภาพรวมในเชิงเทคนิคของส่วนที่ใช้ในการค้นหาข้อมูล

จากรูปจะพบว่า ส่วนของการค้นหาข้อมูลจะประกอบด้วย 3 ส่วนหลักได้แก่

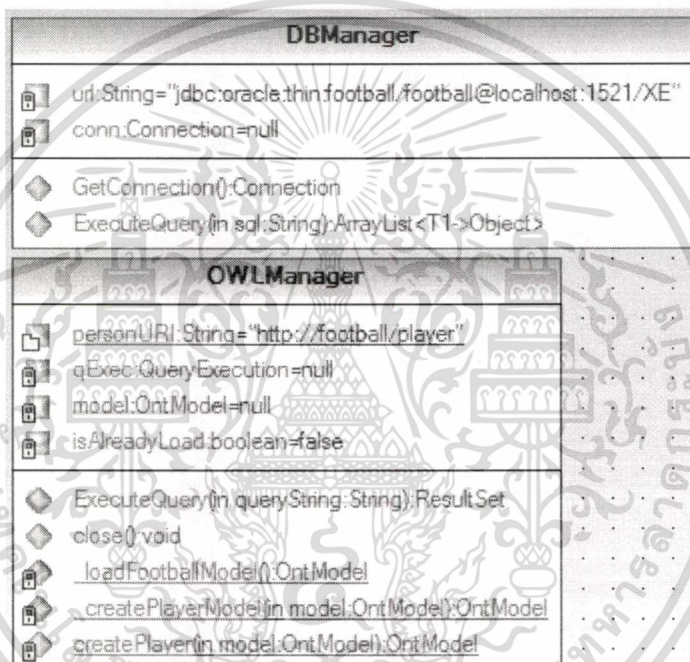
1. OWL Query เป็นส่วนที่ใช้ในการติดต่อกับ Ontology ซึ่งได้ถูกจัดเก็บอยู่ในรูปของไฟล์ OWL เพื่อทำการค้นหาข้อมูลฟุตบอลที่อยู่ใน Ontology จากนั้นจึงนำเอาผลลัพธ์ที่ได้มาประมวลผล สร้างเงื่อนไขและแนวทาง เพื่อนำเอาไปใช้ในการค้นหาข้อมูลที่เกี่ยวข้องทั้งในฐานข้อมูลหรือไฟล์ HTML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือใช้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Ontology Assisted Query เป็นส่วนที่ใช้ในการติดต่อไปค้นหาข้อมูลฟุตบอลที่ถูกจัดเก็บอยู่ในฐานข้อมูล โดยการค้นหาจะทำการค้นหาตามเงื่อนไขและแนวทางที่ Ontology ได้ประมวลผลออกมา
3. Ontology Assisted Select HTML Pages เป็นส่วนที่ใช้ในการดึงเอาหน้า HTML มาแสดงตามเงื่อนไขและแนวทางที่ Ontology ได้ประมวลผลออกมา

เมื่อนำเอาแนวคิดดังกล่าวมาพัฒนา จะได้คลาสดังนี้

1. คลาสที่ใช้เป็นเครื่องมือจัดการข้อมูลในฐานข้อมูลและ OWL



รูปที่ 3.55 แสดงคลาสที่ใช้ในการติดต่อกับฐานข้อมูลและ OWL

DBManager เป็นการเข้าถึงฐานข้อมูลผ่าน jdbc driver ธรรมดา จึงจะไม่ขอก้าวในรายละเอียดในที่นี้ สำหรับ OWLManager นั้นจะทำการอ่านข้อมูล OWL จากไฟล์ผ่าน API ของ Jena ขึ้นมาเก็บไว้ในหน่วยความจำ จากนั้น จะทำการค้นหาข้อมูลต่างๆ ตามเงื่อนไขที่ส่งมา ผ่านทาง API ของ ARQ + Lucene

```

103 private static OntModel _loadFootballModel() {
104     OntModel m = ModelFactory.createOntologyModel();
105     OntDocumentManager dm = m.getDocumentManager();
106     dm.addAltEntry(
107         "http://www.xfront.com/owl/ontologies/football/",
108         "file:C:/Football.owl");
109     m.read( "http://www.xfront.com/owl/ontologies/football/" );
110     return m;
111 }

```

รูปที่ 3.56 แสดงส่วนของโปรแกรมที่ทำการอ่านไฟล์ OWL มาเก็บไว้ในหน่วยความจำ

```

30 public ResultSet ExecuteQuery(String queryString) {
31
32     if(!isAlreadyLoad){
33         model = _loadFootballModel() ;
34         model = _createPlayerModel( model ) ;
35         isAlreadyLoad = true ;
36         model.write( System.out); System.out.println();
37     }
38
39     IndexBuilderString larqBuilder = new IndexBuilderString() ;
40     larqBuilder.indexStatements( model.listStatements() ) ;
41
42     larqBuilder.closeForWriting() ;
43     model.unregister( larqBuilder ) ;
44     // ---- Create the access index
45     IndexLARQ index = larqBuilder.getIndex() ;
46     LARQ.setDefaultIndex(index) ;
47
48     Query query = QueryFactory.create(queryString) ;
49     query.serialize(System.out) ;
50
51     qExec = QueryExecutionFactory.create(query, model) ;
52     LARQ.setDefaultIndex(qExec.getContext(), index) ;
53
54     ResultSet resultSet = qExec.execSelect() ;
55
56     return resultSet ;
57 }

```

รูปที่ 3.57 แสดงส่วนของโปรแกรมที่เข้าไปค้นหาใน OWL

2. คลาสที่ใช้ในการส่งคำค้นไปค้นหาใน OWL และนำเอาผลลัพธ์ที่ได้มาประมวลผลเพื่อสร้างเงื่อนไขหรือคำแนะนำในการไปค้นหาข้อมูลในฐานข้อมูลหรือไฟล์ HTML ต่าง ๆ

Club
◆ SearchByName(in name:String):ArrayList<T1->String>
Player
◆ GetPosition Short Name(in position:String):String
◆ SearchByName(in name:String):ArrayList<T1->String>
Nickname
◆ Get Teamwork Nickname(in nickname:String):ArrayList<T1->String>
◆ Get Teamwork(in nickname:String):String

รูปที่ 3.58 แสดงคลาสที่ใช้ในการค้นหาจาก OWL มาประมวลผล

Football
◆ <<final>> uri:String="http://www.owl-ontologies.com/football.owl#"
◆ m:Model=ModelFactory.createDefaultModel()
◆ <<final>> CONTAIN:Property=m.createProperty(uri, "CONTAIN")
◆ <<final>> HASSHORTNAMEPOSITION:Property=m.createProperty(uri, "has Short Name Position")
◆ <<final>> SAMEASPOSITION:Property=m.createProperty(uri, "Same As Position")
◆ <<final>> PLAYER:Property=m.createProperty(uri, "player")
◆ <<final>> FullName:Property=m.createProperty(uri, "Full Name")
◆ <<final>> FirstName:Property=m.createProperty(uri, "First Name")
◆ <<final>> LastName:Property=m.createProperty(uri, "Last Name")
◆ <<final>> NICKNAME:Property=m.createProperty(uri, "nickname")
◆ <<final>> Has Team:Property=m.createProperty(uri, "Has Team")
◆ <<final>> Has Player:Property=m.createProperty(uri, "Has Player")
◆ <<final>> ClubID:Property=m.createProperty(uri, "ClubID")
◆ <<final>> Club Web URI:Property=m.createProperty(uri, "Club Web URI")
◆ <<final>> DerbyMatchWith:Property=m.createProperty(uri, "Derby Match With")
◆ getURI():String

รูปที่ 3.59 แสดงส่วนของคลาสที่นิยามค่าต่าง ๆ เช่นความสัมพันธ์ใน Ontology

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

50 String queryString = StringUtils.join("\n", new String[] {
51     "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> ",
52     "PREFIX owl: <http://www.w3.org/2002/07/owl#> ",
53     "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> ",
54     "PREFIX daml: <http://www.daml.org/2001/03/daml+oil#> ",
55     "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> ",
56     "PREFIX football: <http://www.owl-ontologies.com/football.owl#> ",
57     "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> ",
58     "PREFIX pf: <http://jena.hpl.hp.com/ARQ/property#> ",
59     "PREFIX : <.> ",
60     "SELECT * { ",
61     "  { ?lit pf:textMatch '" + name + "' . ",
62     "    ?x ?hasNickname ?lit . ",
63     "    ?a ?hasProperty ?x . ",
64     "  OPTIONAL { ?x football:FirstName ?firstname } ",
65     "  OPTIONAL { ?x football:LastName ?lastname } ",
66     "  OPTIONAL { ?a football:ClubID ?clubid } ",
67     "  OPTIONAL { ?a football:ClubWebURI ?clubweburi } } ",
68     "  UNION ",
69     "  { ?x football:FirstName '"+name+"' . ",
70     "    ?x football:LastName ?lastname . } ",
71     "  UNION ",
72     "  { ?x football:LastName '"+name+"' . ",
73     "    ?x football:FirstName ?firstname . } ",
74     " }" });

```

รูปที่ 3.60 แสดง QueryString ที่ใช้ในการค้นหาข้อมูลที่เกี่ยวข้อง
ของนักฟุตบอลที่มีชื่อเล่นตามที่กำหนด

```

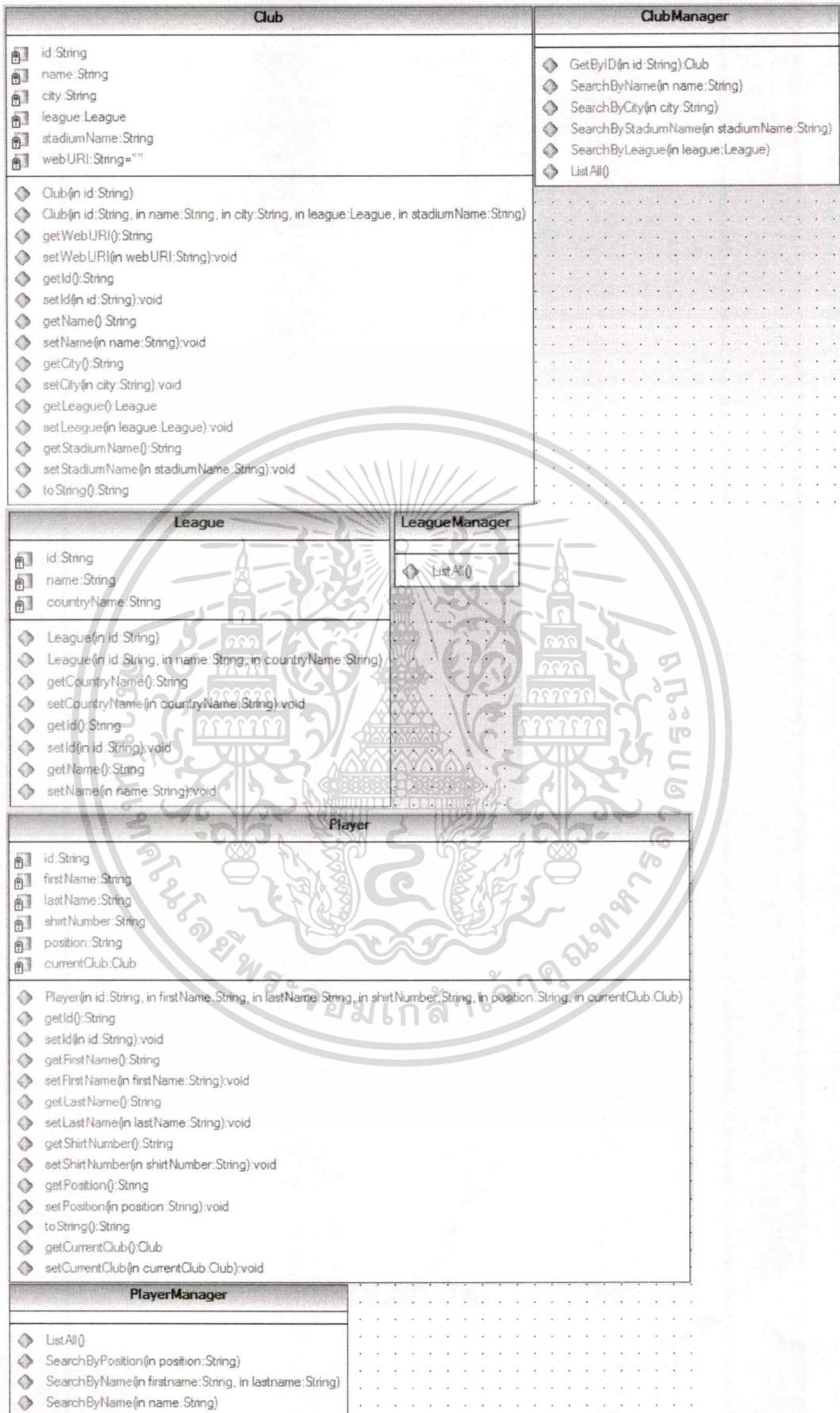
String queryString = StringUtils.join("\n", new String[] {
    "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> ",
    "PREFIX owl: <http://www.w3.org/2002/07/owl#> ",
    "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> ",
    "PREFIX daml: <http://www.daml.org/2001/03/daml+oil#> ",
    "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> ",
    "PREFIX football: <http://www.owl-ontologies.com/football.owl#> ",
    "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> ",
    "PREFIX pf: <http://jena.hpl.hp.com/ARQ/property#> ",
    "PREFIX : <.> ",
    "SELECT * { ",
    "  { ?lit pf:textMatch '" + name + "' . ",
    "    ?x football:nickname ?lit . ",
    "    OPTIONAL { ?x football:FullName ?fullname } ",
    "    OPTIONAL { ?x football:ClubID ?clubid } ",
    "    OPTIONAL { ?x football:ClubWebURI ?clubweburi } } ",
    "  UNION ",
    "  { ?lit pf:textMatch '" + name + "' . ",
    "    ?x ?hasNickname ?lit . ",
    "    OPTIONAL { ?x football:DerbyMatchWith ?derbymatchwith . ",
    "      OPTIONAL { ?derbymatchwith football:ClubID ?derbyclubid } ",
    "      OPTIONAL { ?derbymatchwith football:ClubWebURI ?derbycluburi } } } ",
    " }" });

```

รูปที่ 3.61 แสดง QueryString ที่ใช้ในการค้นหาข้อมูลทุกอย่างที่เกี่ยวข้องกับ
สโมสรฟุตบอลที่มีชื่อเล่นตามที่กำหนด

3. คลาสที่ใช้ในการค้นหาข้อมูลจากฐานข้อมูลหรือไฟล์ HTML ตามเงื่อนไขหรือคำแนะนำจาก Ontology คลาสต่าง ๆ ในส่วนนี้จะแบ่งออกเป็น 2 ประเภทใหญ่ ๆ ได้แก่ ประเภทที่

ใช้เป็น Entity ของข้อมูล เช่น สโมสร, นักฟุตบอล และประเภทที่ใช้ในการจัดการ เช่น การค้า
ไม่ว่ากรณีใดก็ตาม ข้อมูลที่ค้นหาข้อมูลตามที่กำหนดจากฐานข้อมูล เป็นต้น ทุกครั้งที่มีการนำไปใช้



รูปที่ 3.62 แสดงคลาสต่าง ๆ ที่นำเอาเงื่อนไขหรือคำแนะนำจาก Ontology

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า

ไปค้นหาลงฐานข้อมูลหรือ ไฟล์ HTML

ไม่ว่ากรณีใดๆ กรุณาแจ้งที่ขงมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาส่วนจัดเก็บข้อมูล

สำหรับการพัฒนาในส่วนนี้นั้น จะไม่ขอกล่าวรายละเอียดในส่วนของการจัดการข้อมูลของไฟล์ HTML เนื่องจากการจัดเก็บหน้า URL เชื่อมโยงไปยังหน้า HTML เพื่อใช้ทดสอบกรณีที่ต้องการค้นหาจากแหล่งข้อมูลที่ไม่สามารถค้นหาได้โดยง่าย ดังเช่นฐานข้อมูลเชิงสัมพันธ์เท่านั้น

ในการพัฒนาออกแบบในส่วนการจัดการข้อมูลบนฐานข้อมูลนั้น ได้ออกแบบให้ง่าย ไม่ซับซ้อน เนื่องจาก ต้องการทดสอบและเปรียบเทียบ การค้นหาด้วยวิธีต่างๆ ไปกับการค้นหาข้อมูลโดยใช้ OWL ช่วยเหลือนั่นเอง

เนื่องจาก การพัฒนาในส่วนนี้ พัฒนาขึ้นมาเพื่อทดสอบและเปรียบเทียบดังกล่าวข้างต้นเท่านั้น ดังนั้นคลาสที่ถูกพัฒนาจึงมีเฉพาะส่วนที่ใช้ในการค้นหาข้อมูลเท่านั้น (ซึ่งได้แสดงรายละเอียดก่อนหน้านี้) ในส่วนของการพัฒนาส่วนที่ใช้ในการค้นหาข้อมูลที่เกี่ยวข้องกับฟุตบอล) สำหรับการจัดการด้านอื่น ๆ เช่น เพิ่ม ปรับปรุง จะกระทำผ่านเครื่องมือที่มาพร้อมกับฐานข้อมูล Oracle แทน

บทที่ 4

ทดสอบการค้นหาข้อมูลของซีแมนติกเว็บเนื้อหากีฬาฟุตบอล

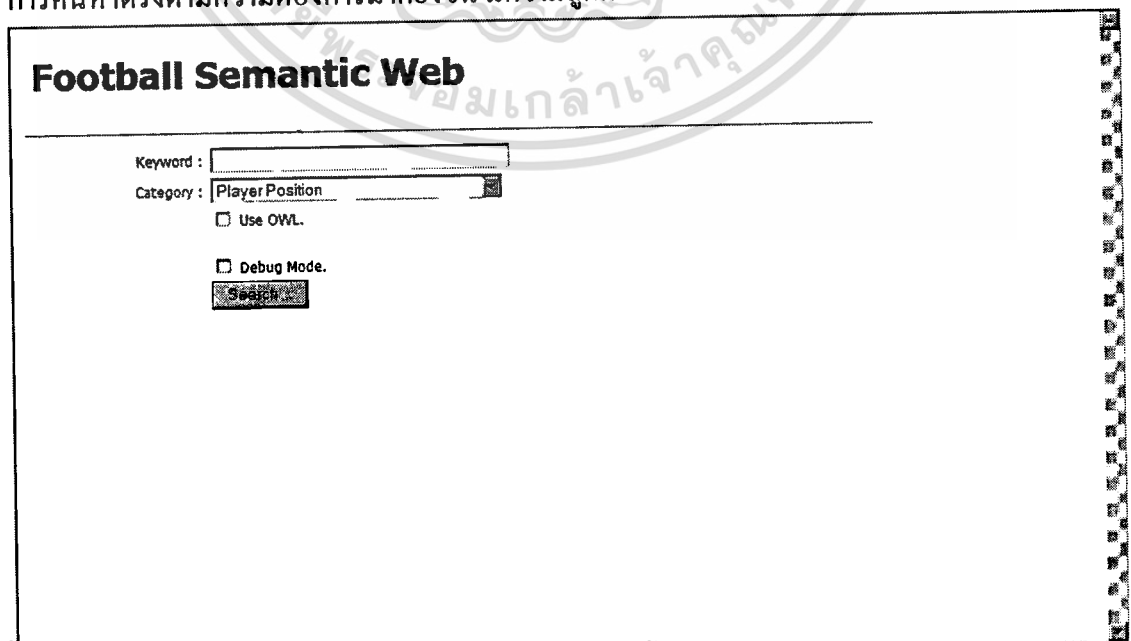
ในบทนี้จะกล่าวถึงการค้นหาข้อมูลในซีแมนติกเว็บเนื้อหากีฬาฟุตบอล และผลการค้นหาข้อมูล เพื่อแสดงให้เห็นประสิทธิภาพในการค้นหาโดยใช้หลักการดังที่กล่าวไว้ในบทที่ 2 และดังที่ได้วิเคราะห์ออกแบบในบทที่ 3 และบทที่ 4 นั้นทำให้การค้นหาข้อมูลในซีแมนติกเว็บนั้นตรงตามความต้องการมากกว่าเว็บโดยทั่วไป

4.1 การค้นหาข้อมูลโดยทั่วไป

ในการค้นหาข้อมูลทั่วไปเป็นการนำคำค้นที่ต้องการค้นหา ไปค้นหาในเอกสารใดๆหรือในฐานข้อมูลที่ทำกรเก็บข้อมูลไว้ โดยการนำคำค้นไปไล่เทียบหา ซึ่งผลการค้นหานั้น อาจจะเป็นข้อมูลที่ไม่ตรงหรือเกินความต้องการ เนื่องจาก คำค้นบางคำอาจจะมีหลายความหมาย หรือคำค้นบางคำอาจจะมีหลายความหมายตรงกับคำค้นอื่น แต่ระบบไม่ทราบ จึงไม่ได้ใช้คำค้นอื่น ๆ เข้าร่วมค้นหา ซึ่งกรณีหลังนี้ จะทำให้ผลการค้นหาคำค้นนั้นขาดหายไป

4.2 การค้นหาข้อมูลซีแมนติกเว็บ

ในการค้นหาข้อมูลในซีแมนติกเว็บเนื้อหากีฬาฟุตบอลนั้น จะใช้ OWL ช่วยเหลือในส่วนของความหมายของคำค้นตลอดจนคำค้นอื่น ๆ ที่มีความหมายเดียวกัน เพื่อทำการค้นหาให้ได้ผลการค้นหาตรงตามความต้องการมากยิ่งขึ้น มีความถูกต้องตามความหมายที่ต้องการมากยิ่งขึ้น



Football Semantic Web

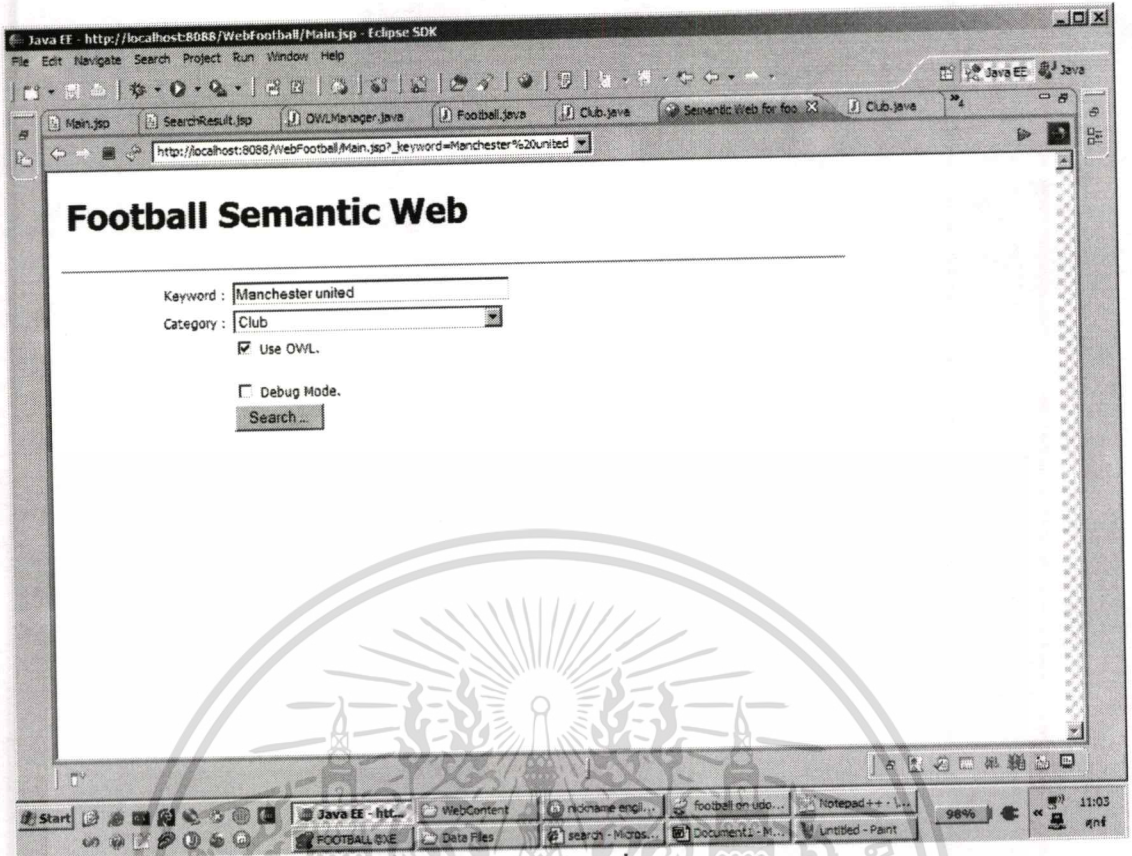
Keyword :

Category :

Use OWL.

Debug Mode.

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้เพื่อวัตถุประสงค์เท่านั้น ไม่สามารถเผยแพร่หรือใช้ประโยชน์ด้านการค้า
รูปที่ 4.1 แสดงหน้าจอหลักของซีแมนติกเว็บเนื้อหากีฬาฟุตบอล
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงหน้าจอเมื่อกรอกคำค้น

4.3 การทดสอบการค้นหาข้อมูลซีแมนติกเว็บ

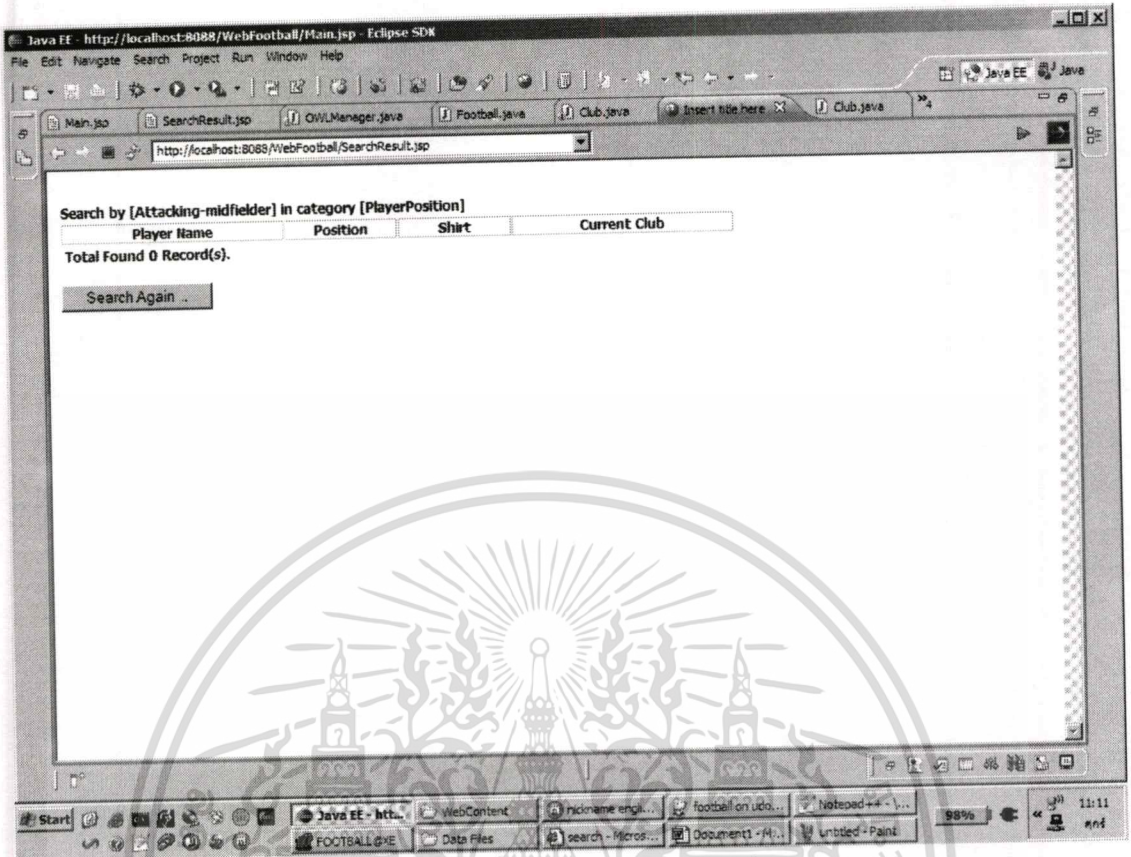
ในการทดสอบการค้นหาข้อมูลในซีแมนติกเว็บเนื้อหากีฬาฟุตบอลนั้น ผู้พัฒนาได้ ออกแบบสถานการณ์ในการค้นหาออกเป็นสถานการณ์ต่าง ๆ ดังต่อไปนี้

1. การค้นหานักฟุตบอลจากตำแหน่ง ด้วยการใช้คำค้นที่เป็นตำแหน่งของผู้เล่น โดยไม่ใช่ OWL ร่วมค้นหาเทียบกับการค้นหาโดยใช้ OWL ร่วมค้นหา
2. การค้นหาข้อมูลของนักฟุตบอล ด้วยการค้นหาจากชื่อเล่นหรือฉายาของนักฟุตบอล โดยไม่ใช่ OWL ร่วมค้นหาเทียบกับการค้นหาโดยใช้ OWL ร่วมค้นหา
3. ที่เป็นส่วนหนึ่งของชื่อหรือชื่อเล่นหรือชื่อฉายาของสโมสร เพื่อทดสอบว่า OWL สามารถค้นหาความสัมพันธ์อื่น ๆ ที่รู้จัก และแนะนำให้ระบบนำเอาความสัมพันธ์อื่น ๆ ที่พบไปค้นหาข้อมูลอื่น ๆ ที่เกี่ยวข้องเพิ่มเติมได้

จากการทำสอบตามการออกแบบข้างต้นนั้น จะได้ผลการทดสอบดังนี้

1. การค้นหานักฟุตบอลจากตำแหน่ง ด้วยการใช้คำค้นที่เป็นตำแหน่งของผู้เล่น โดยไม่ใช่ OWL ร่วมค้นหาเทียบกับการค้นหาโดยใช้ OWL ร่วมค้นหา
- ทำการทดสอบสถานการณ์นี้โดยการค้นหาข้อมูลจากตำแหน่งการเล่น 'Attacking-

midfielder' ค้นหาโดยไม่ใช่ OWL ช่วยค้นหา จะได้ผลลัพธ์ดังภาพ ประโยชน์ด้านการค้า
ไม่ว่าการนี้ใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงหน้าผลการค้นหาเมื่อไม่ใช้ OWL จากคำค้น Attacking-midfielder และเมื่อนำเอาคำค้นเดียวกันไปค้นหา โดยใช้ OWL ช่วยเหลือ จะได้ผลลัพธ์ดังภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Search by [Attacking-midfielder] in category [PlayerPosition]

Player Name	Position	Shirt	Current Club
Seol Ki-Hyeon	MF	7	Fulham
Alexey Smertin	MF	8	Fulham
Steven Davis	MF	10	Fulham
Michael Timin	MF	16	Fulham
Lee Cook	MF	17	Fulham
Jimmy Bullard	MF	21	Fulham
Clint Dempsey	MF	23	Fulham
Simon Davies	MF	25	Fulham
Danny Murphy	MF	27	Fulham
Robert Milson	MF	28	Fulham
Wayne Brown	MF	30	Fulham
Kalifa Ciss?	MF	4	Reading
Brynjar Gunnarsson	MF	6	Reading
Glen Little	MF	7	Reading
Stephen Hunt	MF	10	Reading
John Oster	MF	11	Reading
James Harper	MF	15	Reading
Bobby Convey	MF	17	Reading
Emrese Fa?	MF	20	Reading
James Henry	MF	34	Reading
Jem Karacan	MF	40	Reading
Oliver Bozanic	MF	42	Reading
Papa Bouba Diop	MF	8	Portsmouth
Sulley Muntari	MF	11	Portsmouth
Matthew Taylor	MF	14	Portsmouth
Arnold Mvuemba	MF	18	Portsmouth
Niko Kranjcar	MF	19	Portsmouth
Damien Johnson	MF	22	Birmingham City
Olivier Kapo	MF	23	Birmingham City
Fabrice Muamba	MF	26	Birmingham City
Wilson Palacios	MF	28	Birmingham City
Borja Oubiña	MF	29	Birmingham City
Jordan Mutch	MF	31	Birmingham City
Antona Siberski	MF	6	Wigan Athletic
Kevin Kibane	MF	8	Wigan Athletic
Jason Koumas	MF	10	Wigan Athletic
Michael Brown	MF	11	Wigan Athletic
Denny Landzaat	MF	14	Wigan Athletic
Antonio Valencia	MF	16	Wigan Athletic
Paul Schärner	MF	18	Wigan Athletic
Tomasz Cywka	MF	21	Wigan Athletic
Josip Skoko	MF	24	Wigan Athletic
Rachid Bouadouzan	MF	26	Wigan Athletic
Peter Moore	MF	28	Wigan Athletic
Lewis Montrose	MF	29	Wigan Athletic
Julio Arca	MF	3	Middlesbrough
Gary O'Neil	MF	4	Middlesbrough
George Boateng	MF	7	Middlesbrough
F7bio Rochemback	MF	10	Middlesbrough
Mohamed Shawky	MF	15	Middlesbrough
Stewart Downing	MF	19	Middlesbrough
Lee Cattermole	MF	27	Middlesbrough
Graeme Owens	MF	35	Middlesbrough
Josh Walker	MF	41	Middlesbrough

Total Found 209 Record(s).

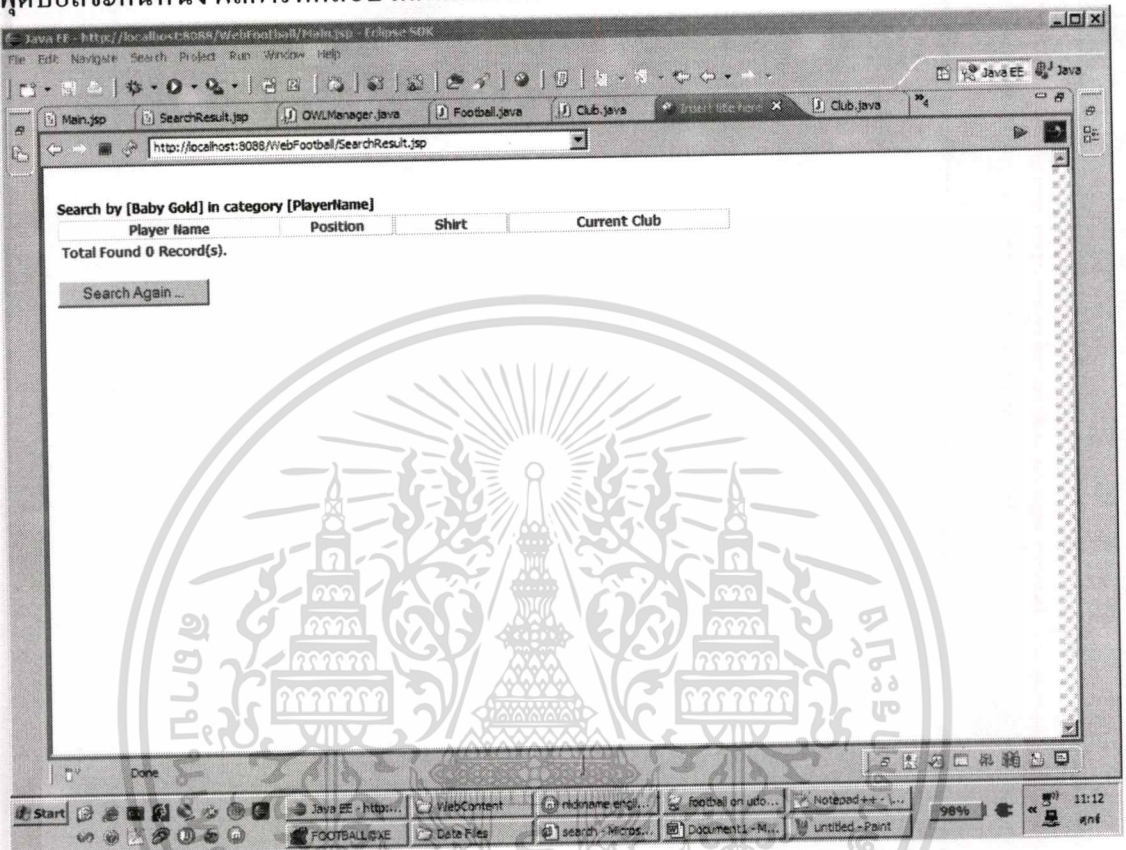
Search Again...

รูปที่ 4.4 แสดงหน้าผลการค้นหาเมื่อใช้ OWL จากคำค้น Attacking-midfielder

จากรูปผลการค้นหา จะพบว่า เมื่อไม่ใช้ OWL ช่วยเหลือ ระบบจะไม่สามารถค้นหาข้อมูลของนักฟุตบอลได้ เนื่องจาก ในข้อมูลของนักฟุตบอลนั้นจะไม่รู้จักตำแหน่งการเล่น Attacking-midfielder จึงทำให้ไม่สามารถค้นหาข้อมูลได้ แต่หากใช้ OWL ช่วยเหลือ จะพบว่า OWL จะรู้จักตำแหน่ง Attacking-midfielder และแนะนำคำค้นอื่น ๆ ที่มีความหมายเดียวกับกับระบบ ทำให้ระบบสามารถค้นหาข้อมูลจากตำแหน่ง โดยใช้คำค้นนี้ได้

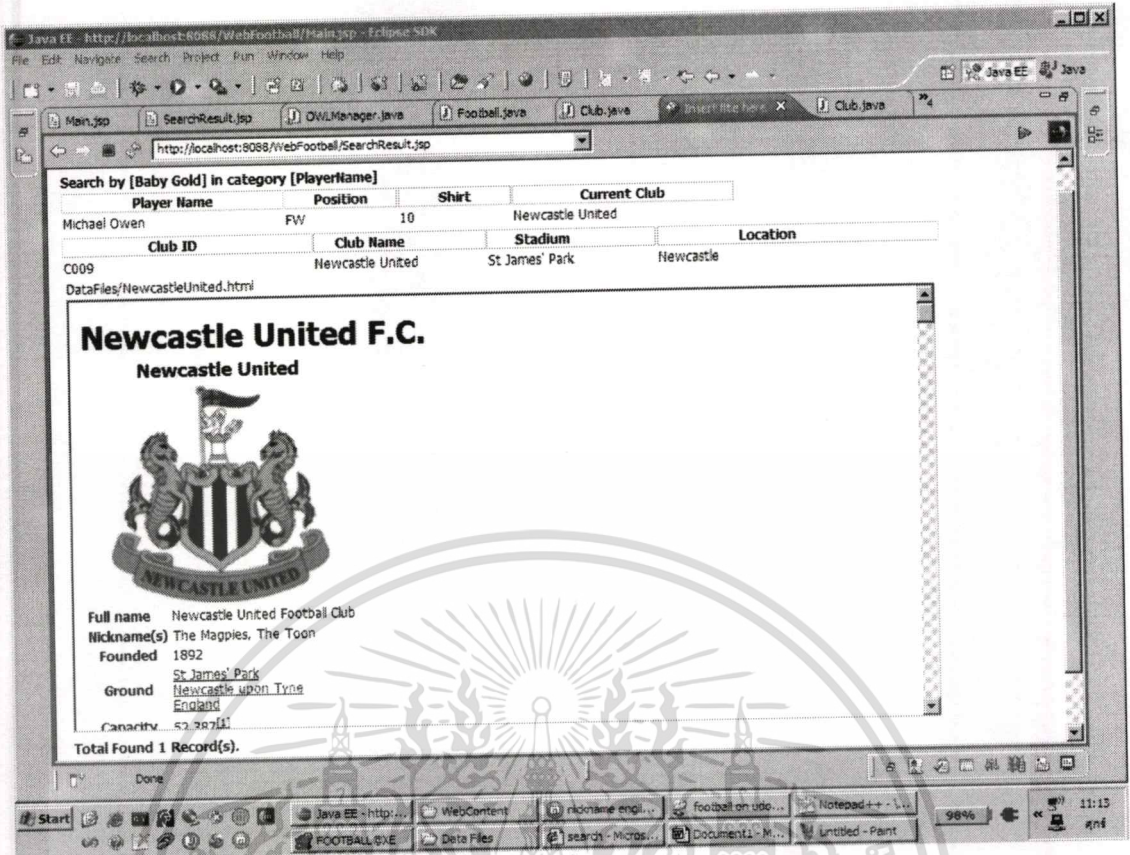
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การค้นหาข้อมูลของนักฟุตบอล ด้วยการค้นหาจากชื่อเล่นหรือฉายาของนักฟุตบอล โดยไม่ใช่ OWL ร่วมค้นหาเทียบกับการค้นหาโดยใช้ OWL ร่วมค้นหา ในการทดสอบสถานะการณ์นี้ จะใช้คำค้นว่า 'Baby Gold' ซึ่งหมายถึงฉายาของนักฟุตบอลจะคนหนึ่ง ผลการทดสอบได้ผลลัพธ์ดังภาพ



รูปที่ 4.5 แสดงหน้าผลการค้นหาเมื่อไม่ใช่ OWL จากคำค้น Baby Gold

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงหน้าผลการค้นหาเมื่อใช้ OWL จากคำค้น Baby Gold

จากผลการทดสอบในสถานการณ์นี้ จะพบว่า หากไม่มีการใช้ OWL ช่วยเหลือในการค้นหา ระบบจะไม่ได้สามารถค้นหาได้ เนื่องจาก ระบบไม่รู้จัก Baby Gold นั้นเอง แต่หากใช้ OWL ช่วยเหลือในการค้นหาครั้งนี้ จะพบว่า OWL จะรู้จัก Baby Gold โดยทราบว่าเป็นฉายาของนักฟุตบอลคนหนึ่ง พร้อมทั้งยังพบความสัมพันธ์อื่น ๆ ของนักฟุตบอลเจ้าของฉายานี้อีกด้วย และเมื่อระบบทำการค้นหาจากคำแนะนำของ OWL แล้ว จึงทำให้สามารถค้นหานักฟุตบอลผู้นี้ พร้อมกับข้อมูลอื่น ๆ ที่เกี่ยวข้อง เช่น ข้อมูลในเว็บของสโมสรที่นักฟุตบอลผู้นี้สังกัดอยู่

3. การค้นหาข้อมูลของสโมสรจากคำค้น ที่เป็นส่วนหนึ่งของชื่อหรือชื่อเล่นหรือชื่อฉายาของสโมสร เพื่อทดสอบว่า OWL สามารถค้นหาความสัมพันธ์อื่น ๆ ที่รู้จัก และแนะนำให้ระบบนำเอาความสัมพันธ์อื่น ๆ ที่พบไปค้นหาข้อมูลอื่น ๆ ที่เกี่ยวข้องเพิ่มเติมได้

สำหรับผลการทดสอบสถานการณ์นี้ จะทำการทดสอบด้วยการใช้คำค้น 2 คำ ได้แก่ 'Manchester united' ซึ่งเป็นชื่อเต็มของสโมสรฟุตบอลแห่งหนึ่ง และคำว่า 'Red' ซึ่งเป็นส่วนหนึ่งของฉายาสโมสรฟุตบอลหลาย ๆ แห่ง และเมื่อนำเอาคำ 'Manchester united' ไปค้นหาข้อมูลจะได้ผลลัพธ์ดังภาพ


Java EE - http://localhost:8088/WebFootball/Main.jsp - Eclipse SDK
File Edit Navigate Search Project Run Window Help



http://localhost:8088/WebFootball/SearchResult.jsp

Search by [Manchester united] in category (Club)

Club ID	Club Name	Stadium	Location
C002	Manchester United	Old Trafford	Manchester

DataFiles/ManchesterUnited.html




Full name Manchester United Football Club
Nickname(s) The Red Devils, Man United, United
Founded 1878, as *Newton Heath L&YR F.C.*
Ground [Old Trafford](#)
Capacity 76,212
Chairman  Joel & Avram Glazer
Manager  Sir Alex Ferguson
League [Premier League](#)
2006-07 [Premier League, 1st](#)

Manchester United Football Club is an [English football club](#), based at the [Old Trafford stadium](#) in [Trafford, Greater Manchester](#), and is arguably the most popular football club in the world, with over 50 million supporters worldwide;^{[1][2]} average attendances at the club have been higher than any other team in [English football](#) for all but six seasons since [1964-65](#).^[3] The club is also one of the most successful in English football; for over twenty years, since the [1986-87 season](#), they have won 18 major honours, which is more than any other [Premier League club](#).^[4] They are the Premier League's reigning champions, and have won the [Premier League/Football League](#) 16 times. In 1968, they became the first English club to win the [European Cup](#), beating [S.L. Benfica](#) 4-1, and they won a second European Cup in 1999. They also hold the record for the most FA Cup titles with 11.^[5] Since the late 1990s, the club has been one of the richest in the world, and until recently had the highest revenue of any football club for several years running. As of 2007, the club has the fourth largest turnover in club football,^[6] but remains the most profitable club for several years running.

C003 Manchester City City of Manchester Stadium Manchester
DataFiles/ManchesterCity.html

Manchester City F.C.
Manchester City



Full name Manchester City Football Club
Nickname(s) The Citizens, The Blues, City, Man City
Founded 1880, as *St Mark's (West Gorton)*

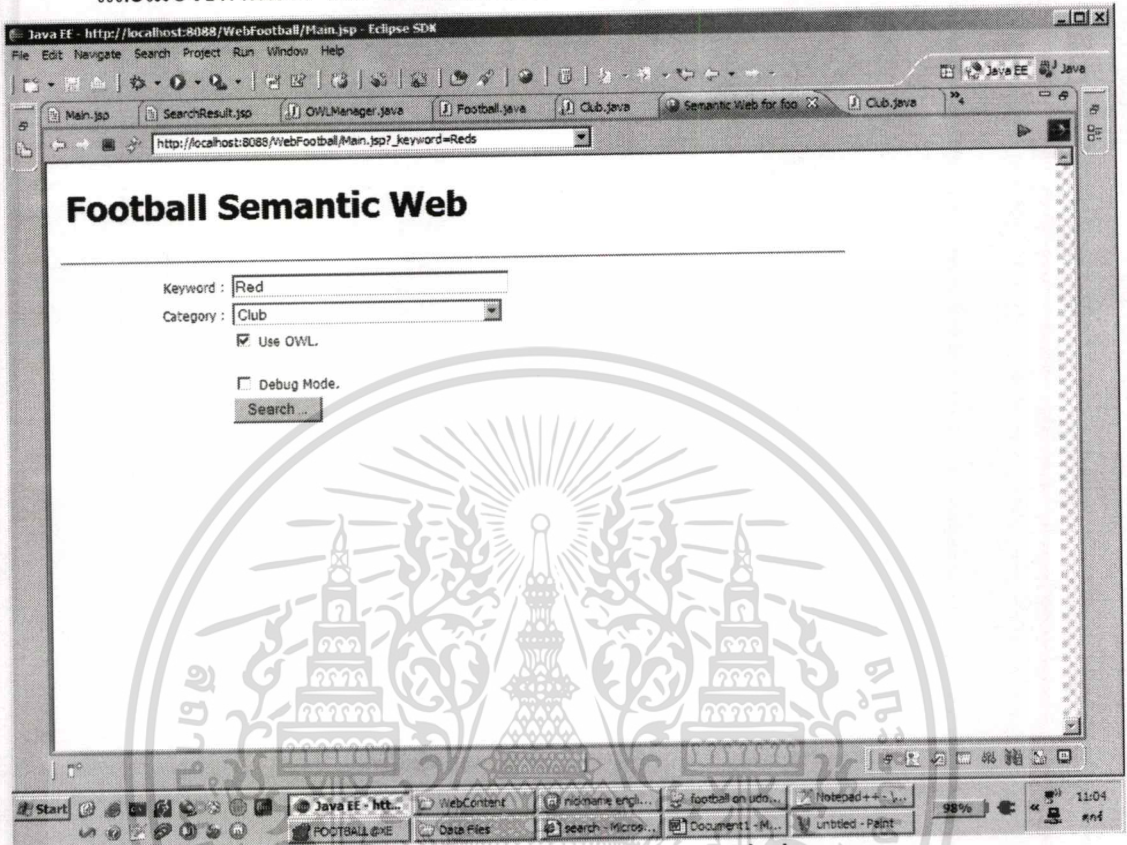
Total Found 2 Record(s).

Start Java EE - Http... WebContent nickname engl... football on udo... Notepad++... 98% 11:00 AM

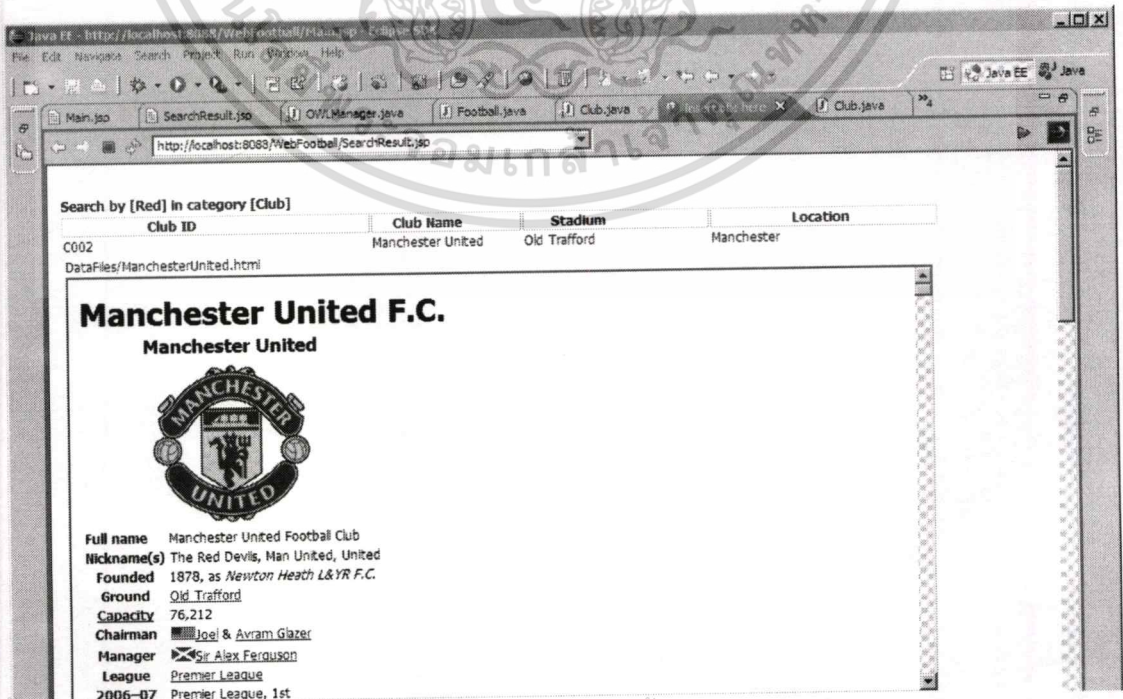
รูปที่ 4.7 แสดงหน้าผลการค้นหาเมื่อกรอกคำค้นสโมสร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

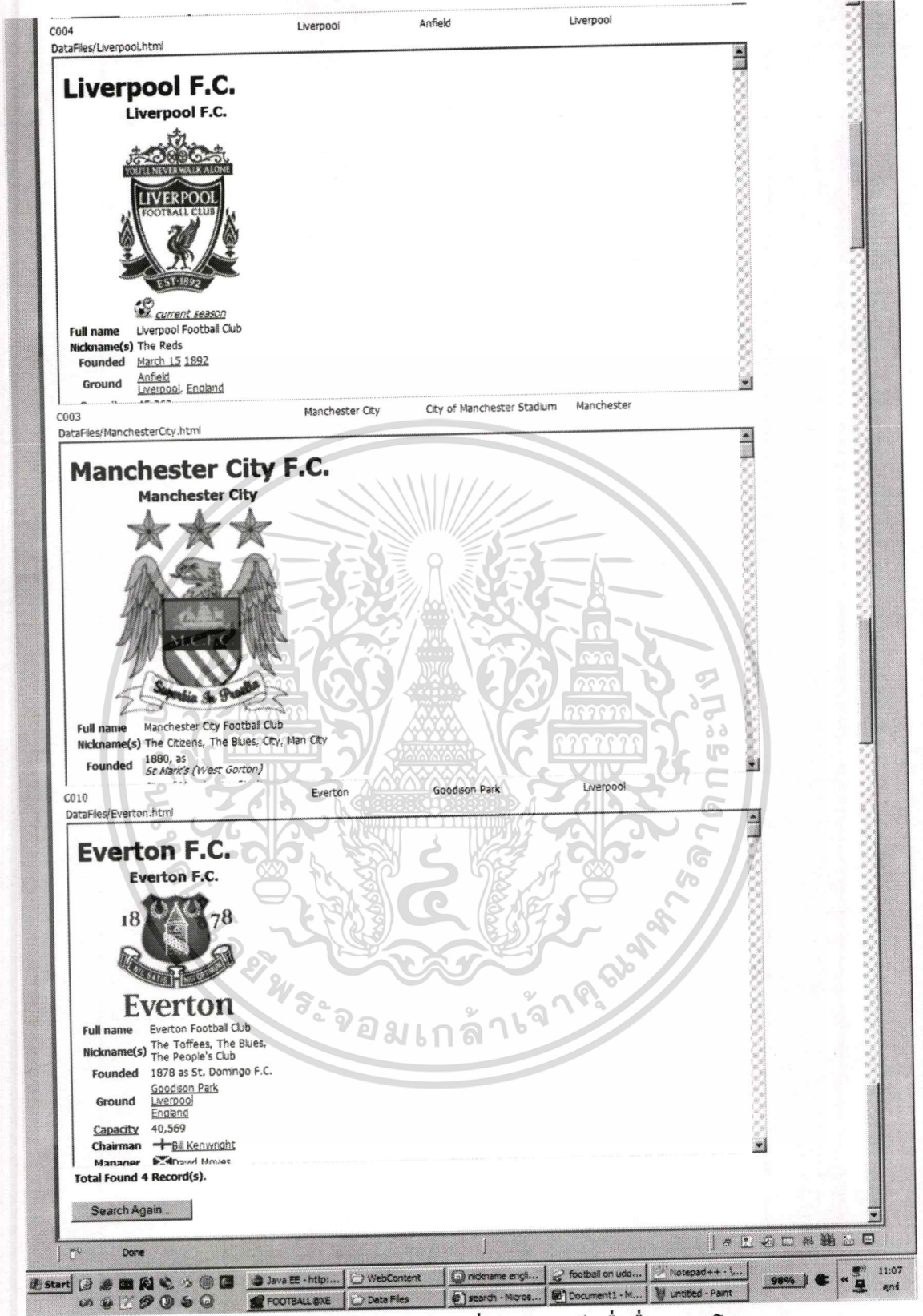
จากผลการทดสอบที่ได้ จะพบว่า ระบบสามารถค้นหาสโมสรเจอ พร้อมทั้งยังพบสโมสรคู่แข่งที่ตั้งอยู่ร่วมเมืองเดียวกันออกมาได้ และเมื่อใช้คำค้นหา 'red' จะได้ผลการทดสอบดังภาพ



รูปที่ 4.8 แสดงหน้าจอเมื่อกรอกคำค้นหาสโมสรจากชื่ออักษรของสโมสร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงหน้าผลการค้นหาเมื่อกรอกคำค้นชื่ออื่นของสโมสร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลลัพธ์ที่ได้นั้น จะพบว่าระบบสามารถค้นหาข้อมูลสโมสรฟุตบอลจากชื่อเล่นหรือฉายาของสโมสรพบ 2 แห่งนั้นคือ The Red Devils และ The Reds และระบบยังพบอีกว่านอกจาก 2 สโมสรดังกล่าวแล้ว แต่ละสโมสรยังมีคู่แข่งร่วมเมืองเดียวกันอีกด้วย ระบบจึงได้นำเอาสโมสรคู่แข่งร่วมเมืองขึ้นมาแสดง

4.4 สรุปผลการค้นหา

จากการทดสอบจะพบว่า ในการค้นหาข้อมูลโดยทั่วไปนั้น อาจจะได้ผลลัพธ์ที่ไม่ตรงกับความต้องการ อาจจะขาดหรือเกินความต้องการ เนื่องจากระบบจะทำการค้นหาด้วยการไล่เทียบคำค้นนั้น กับเนื้อหาที่มีอยู่ ซึ่งบางแหล่งข้อมูล เช่นบางเว็บ อาจจะทำลอกระบบค้นหาด้วยการบรรจุคำค้นไว้ ไม่ตรงกับเนื้อหาข้อมูลจริง เพื่อลอกระบบให้แสดงหน้าเว็บของตน และเมื่อผู้ค้นหาเข้าไปดูจะได้ลอกขยายโฆษณาชวนเชื่อต่าง ๆ แต่หากระบบค้นหานั้น มีการนำเอา OWL ไปช่วยเหลือในการค้นหาด้วย จะทำให้ระบบสามารถค้นหาข้อมูลได้แม่นยำ และยังสามารถหาข้อมูลอื่น ๆ ที่เกี่ยวข้องกับคำค้นหาเพิ่มเติมได้อีก โดยไม่เป็นการเกินความต้องการเนื่องจาก OWL มีการตรวจสอบความสัมพันธ์ก่อนที่จะแนะนำให้ลองค้นหาจากข้อมูลที่เกี่ยวข้อง ทำให้ได้ข้อมูลที่ถูกต้องครบถ้วนกว่าแบบทั่วไป

บทที่ 5

บทสรุปโครงการพัฒนาระบบ

ข้อมูลในปัจจุบันมีอยู่มากมายและเพิ่มมากขึ้นทุกวัน การจัดการข้อมูลจึงเป็นเรื่องที่จำเป็นมาก การจัดเก็บที่เป็นระเบียบไม่ซ้ำซ้อน, การแบ่งปันข้อมูล และการค้นคืนข้อมูลให้มีประสิทธิภาพจึงเป็นเรื่องที่มีการพัฒนาอย่างต่อเนื่อง

การเก็บข้อมูลในฐานข้อมูล เป็นการจัดการข้อมูลที่ใช้กันทั่วไปอย่างแพร่หลาย เนื่องจากการจัดการความซ้ำซ้อน และการสร้าง index เพื่อช่วยทำให้การค้นหามีความรวดเร็วมากขึ้น ทำให้ค้นคืนข้อมูลได้มีประสิทธิภาพอย่างเป็นที่ยอมรับกันทั่วไป แต่การค้นหาเป็นการค้นหาจากคำค้นนั้นคือหากพบคำค้นก็ให้คืนค่าเป็นผลการค้นหาให้ผู้ใช้ ซึ่งผลการค้นหาดังกล่าวจึงยังมีความหมายไม่ตรงกับความหมายที่ผู้ใช้ต้องการค้นหาเท่าที่ควร เนื่องจากการค้นหาดังกล่าวไม่ได้ค้นหาจากความหมายของคำค้นเลย ดังนั้นผลการค้นหาที่ได้จึงเป็นผลการค้นหาที่ไม่ตรงความหมาย ไม่ตรงกับเรื่องที่ต้องการค้นหาซะทีเดียว

การนำ OWL (Web Ontology Language) มาใช้ในการกำหนดรายละเอียดและความสัมพันธ์ของข้อมูลก็เป็นเทคนิคหนึ่งในการระบุความหมายของข้อมูล ช่วยในการจัดเก็บข้อมูลและช่วยในการค้นคืนข้อมูลให้มีประสิทธิภาพมากยิ่งขึ้น ช่วยให้คำค้นคืนข้อมูลได้ถูกต้องตามความหมายที่ต้องการค้นหามากยิ่งขึ้น ทำให้เวลาในการค้นหาข้อมูลที่มีอยู่มากมายในปัจจุบันนั้นลดลง เพิ่มความสะดวกสบายให้กับผู้ใช่มากยิ่งขึ้น

ในโครงการพัฒนาระบบฉบับนี้ได้นำเสนอวิธีการกำหนดความหมายให้กับข้อมูลโดยนำ OWL มาใช้กำหนดรายละเอียดและความสัมพันธ์ให้กับข้อมูล ซึ่งเป็นเพียงวิธีการหนึ่งในการเพิ่มประสิทธิภาพในการค้นหาให้ตรงกับความหมายที่ผู้ใช้ต้องการมากขึ้น

บรรณานุกรม

HP Labs Semantic Web Research. 2007 **Jena Semantic Web Framework** [Online] Available.

<http://jena.sourceforge.net/index.html>

The University Of Manchester. 2007 **The Protégé-Owl Plugin and CO-ODE Tools** [Online]

Available. <http://protege.stanford.edu/plugins/owl/api/guide.html>

W3C Organization. 2007 **Web Ontology Language (OWL)** [Online] Available.

http://www.w3schools.com/rdf/rdf_owl.asp

W3C Organization. 2007 **XQuery** [Online] Available. <http://www.w3.org/XML/Query/>



ประวัติผู้เขียน

ชื่อ-นามสกุล นางสาวปนัดดา ฐูปแพ
วันเดือนปีเกิด 5 ตุลาคม 2524
ประวัติการศึกษา จบปริญญาตรี จากมหาวิทยาลัยเกษตรศาสตร์ คณะวิทยาศาสตร์ สาขา
วิทยาการคอมพิวเตอร์ เมื่อปี พ.ศ.2546
ปัจจุบันศึกษาต่อ ปริญญาโทที่มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าคุณ
ทหารลาดกระบัง คณะเทคโนโลยีสารสนเทศ
ประวัติการทำงาน ปี พ.ศ. 2546 เข้าทำงานบริษัทเอกชน กลุ่มบริษัทซอฟต์แวร์
ตำแหน่ง นักพัฒนาซอฟต์แวร์
ปัจจุบัน เป็นพนักงานบริษัททีเอสที อินเทอร์เน็ตซันนอล
ตำแหน่งนักวิเคราะห์และพัฒนาโปรแกรม

นางสาวปนัดดา ฐูปแพ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้