

การพัฒนาเครื่องมือสร้างข้อมูลทดสอบเพื่อสนับสนุน
การทดสอบซอฟต์แวร์จากสคีมาของฐานข้อมูล



นายกฤษณะ พิริยะกิจไพบุลย์

สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2547

ISBN 974-17-6185-6

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

DEVELOPMENT OF A TEST DATA GENERATING TOOL TO SUPPORT
SOFTWARE TESTING FROM DATABASE SCHEMA

Mr. Kritsana Piriyakitpaiboon



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2004

ISBN 974-17-6185-6

หัวข้อวิทยานิพนธ์	การพัฒนาเครื่องมือสร้างข้อมูลทดสอบเพื่อสนับสนุนการทดสอบซอฟต์แวร์จากสคีมาของฐานข้อมูล
โดย	นาย กฤษณะ พิริยะกิจไพบูลย์
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. ธาราทิพย์ สุวรรณศาสตร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. ดิเรก ลาวัญย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. พรศิริ หมั่นไชยศิริ)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร. ธาราทิพย์ สุวรรณศาสตร์)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. วิวัฒน์ วัฒนาวุฒิ)

..... กรรมการ
(อาจารย์ ดร. อาทิตย์ ทองทักษ์)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

กฤษฎณะ พิริยะกิจไพบูลย์: การพัฒนาเครื่องมือสร้างข้อมูลทดสอบเพื่อสนับสนุนการทดสอบซอฟต์แวร์จากสคีมาของฐานข้อมูล. (DEVELOPMENT OF A TEST DATA GENERATING TOOL TO SUPPORT SOFTWARE TESTING FROM DATABASE SCHEMA) อ. ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร. ธาธาทิพย์ สุวรรณศาสตร์, 164 หน้า. ISBN 974-17-6185-6.

การสร้างข้อมูลทดสอบเพื่อทดสอบซอฟต์แวร์เป็นงานที่เป็นภาระหนักอย่างหนึ่งที่ต้องใช้เวลาและแรงงานเป็นอย่างมากสำหรับผู้ทดสอบซอฟต์แวร์เนื่องจากข้อมูลทดสอบที่สร้างออกมานั้นจำเป็นต้องมีความคล้ายคลึงกับข้อมูลที่ได้จากการใช้งานจริงให้มากที่สุดทั้งในแง่ของปริมาณและความหมาย วิทยานิพนธ์จึงนำเสนอเครื่องมือที่ช่วยผู้ทดสอบสร้างข้อมูลทดสอบได้ง่ายขึ้น โดยเครื่องมือจะสุ่มสร้างข้อมูลทดสอบตามเงื่อนไขที่ได้มาจากผู้ทดสอบและฐานข้อมูลต้นทาง เงื่อนไขจะประกอบด้วยสคีมาของฐานข้อมูล การขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูล ความคงสภาพในการอ้างอิงของฐานข้อมูล ปริมาณระเบียบข้อมูลที่ต้องการสร้างวิธีการสร้างข้อมูลทดสอบ และรูปแบบการบันทึกข้อมูลทดสอบ เป้าหมายของเครื่องมือที่พัฒนาขึ้นนี้คือการสร้างข้อมูลทดสอบที่มีความใกล้เคียงกับข้อมูลที่ได้จากการใช้งานจริงของซอฟต์แวร์ที่กำลังถูกทดสอบให้มากที่สุด ซึ่งจากการทดสอบการใช้งานเครื่องมือกับระบบงานจริงพบว่าเครื่องมือสามารถสร้างข้อมูลทดสอบได้ตรงกับคุณสมบัติของข้อมูลที่มีอยู่จริง โดยข้อมูลทดสอบที่สร้างได้จะมีความถูกต้องตามการขึ้นต่อกันเชิงตรรกะและความคงสภาพในการอ้างอิงของสคีมาของฐานข้อมูลด้วย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา.....วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อนิสิต.....
สาขาวิชา.....วิทยาวิศวกรรมคอมพิวเตอร์.. ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา...2547..... ลายมือชื่ออาจารย์ที่ปรึกษาร่วม.....

4570208321 : MAJOR COMPUTER SCIENCE

KEYWORD: SOFTWARE TESTING / TEST DATA GENERATOR

KRITSANA PIRIYAKITPAIBOON : DEVELOPMENT OF A TEST DATA
GENERATING TOOL TO SUPPORT SOFTWARE TESTING FROM DATABASE
SCHEMA.

THESIS ADVISOR : ASSISTANT PROFESSOR TARATIP SUWANNASART, PH.D.,
164 pp. ISBN 974-17-6185-6.

Test data generation is one of the tasks that consumes a lot of cost and effort for software testers because the generated test data need to be similar to the actual data in both quantity and meaning. This thesis presents a tool to automate test data generation for testers. The tool randomly generates test data according to the criteria supplied by testers and source database. The criteria consist of database schema, logical dependencies between fields in each table, referential integrity of database, quantity of generated records, method of generation, and output format. The goal of this tool is to generate test data that is meaningful as the actual data of the software under test. The result of the testing of this tool with real software indicates that the tool could generate test data that is meaningful to the software under test.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department.....Computer Engineering..... Student's signature.....
Field of study.....Computer Science..... Advisor's signature.....
Academic year...2004..... Co-advisor's signature.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลืออย่างดียิ่งของบุคคลหลายๆ ท่าน ข้าพเจ้าขอขอบพระคุณบุคคลดังต่อไปนี้เป็นอย่างสูง

ท่านผู้ช่วยศาสตราจารย์ ดร. ธราทิพย์ สุวรรณศาสตร์ อาจารย์ที่ปรึกษา วิทยานิพนธ์ ขอกราบขอบพระคุณท่านอาจารย์ที่ให้คำแนะนำ คำปรึกษา และความเมตตาต่อ ข้าพเจ้าอย่างดีเสมอมา

ขอกราบขอบพระคุณท่านคณะกรรมการ ซึ่งได้แก่ ผู้ช่วยศาสตราจารย์ ดร. วิวัฒน์ วัฒนานุกูล ผู้ช่วยศาสตราจารย์ ดร. พรศิริ หมั่นไชยศรี และอาจารย์ ดร. อาทิตย์ ทองทักษ์ที่กรุณา สละเวลาให้คำแนะนำต่างๆ

ขอกราบขอบพระคุณบิดามารดาของข้าพเจ้าที่คอยให้ความหวังใจ ความเข้าใจ และกำลังใจทุกครั้งที่ข้าพเจ้าท้อแท้และเหน็ดเหนื่อยเสมอมา



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ	ช
สารบัญตาราง.....	ญ
สารบัญภาพ	ฐ
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการวิจัย.....	2
1.3 ขอบเขตของการวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.5 ขั้นตอนการวิจัย	3
2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	5
2.1 ทฤษฎีที่เกี่ยวข้อง	5
2.1.1 การทดสอบซอฟต์แวร์	5
2.1.2 เอ็กซ์เอ็มแอล (XML – eXtensible Markup Language) [4].....	6
2.1.3 เอ็กซ์เอ็มแอลสคีมา (XML Schemas).....	8
2.1.4 คำศัพท์ที่สำคัญ.....	10
2.2 งานวิจัยที่เกี่ยวข้อง.....	11
2.2.1 Datatect 1.6.....	11
2.2.2 วิทยานิพนธ์เรื่อง “การพัฒนาเครื่องมือซอฟต์แวร์สำหรับสร้างข้อมูลทดสอบ (Development of a Software Tool for Generating Test Data)”	11
2.2.3 TurboData	12

สารบัญ (ต่อ)

บทที่	หน้า
3. การวิเคราะห์และออกแบบเครื่องมือ	13
3.1 โครงสร้างการทำงานของเครื่องมือ	13
3.2 การวิเคราะห์และออกแบบเครื่องมือ	16
3.2.1 แผนภาพยูสเคส.....	16
3.2.2 แผนภาพคลาส	31
3.2.3 แผนภาพซีควเอนซ์และแผนภาพคอลลาบอเรชัน	35
3.2.4 แผนภาพลำดับกิจกรรม.....	44
4. การพัฒนาเครื่องมือ	65
4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ	65
4.2 โครงสร้างฐานข้อมูลของเครื่องมือ.....	65
4.3 โครงสร้างของเครื่องมือ	67
4.3.1 ส่วนประกอบในการนำเครื่องมือไปใช้งาน.....	67
4.3.2 ส่วนประกอบของโครงสร้างหน้าจอของเครื่องมือ.....	68
5. การทดสอบเครื่องมือ	81
5.1 สภาพแวดล้อมที่ใช้ในการทดสอบ	81
5.2 การทดสอบเครื่องมือ.....	81
5.2.1 การทดสอบการใช้เครื่องมือกับระบบงานจริง	81
5.2.2 การทดสอบเพื่อวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบ	83
6. สรุปผลการวิจัย	88
6.1 สรุปผลการวิจัย.....	88
6.2 ข้อจำกัดของเครื่องมือสร้างข้อมูลทดสอบ	89
6.3 แนวทางในการพัฒนาต่อ.....	89
6.4 ผลงานที่เกี่ยวข้องกับการวิจัย.....	90
รายการอ้างอิง.....	91

สารบัญ (ต่อ)

บทที่	หน้า
ภาคผนวก.....	92
ภาคผนวก ก พจนานุกรมข้อมูลของฐานข้อมูลของเครื่องมือ	93
ภาคผนวก ข ตัวอย่างผลการทดสอบเครื่องมือสร้างข้อมูลทดสอบกับระบบสารสนเทศ ผู้ปวยนอกของสถานพยาบาลรัฐวิสาหกิจ	97
ภาคผนวก ค ตัวอย่างผลการทดสอบเครื่องมือสร้างข้อมูลทดสอบกับระบบช่วยตัดสินใจ ในการเลือกสถาบันเพื่อศึกษาต่อ.....	122
ภาคผนวก ง ตัวอย่างผลการทดสอบเครื่องมือสร้างข้อมูลทดสอบวัดเวลาตอบสนอง (Response Time) ในการสร้างข้อมูลทดสอบ	128
ภาคผนวก จ คู่มือการใช้งานเครื่องมือสร้างข้อมูลทดสอบ.....	129
ภาคผนวก ฉ ผลงานที่ตีพิมพ์.....	155
ประวัติผู้เขียนวิทยานิพนธ์.....	164

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 3.1 รายละเอียดของยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย.....	17
ตารางที่ 3.2 รายละเอียดของยูสเคสกำหนดชื่อสคีมามาของฐานข้อมูล.....	19
ตารางที่ 3.3 รายละเอียดของยูสเคสสคีมามาของฐานข้อมูลซ้ำซ้อน.....	19
ตารางที่ 3.4 รายละเอียดของยูสเคสสร้างสคีมามาของฐานข้อมูล.....	20
ตารางที่ 3.5 รายละเอียดของยูสเคสสร้างสคีมามาของฐานข้อมูลด้วยมือ.....	20
ตารางที่ 3.6 รายละเอียดของยูสเคสสร้างสคีมามาของฐานข้อมูลจากฐานข้อมูลต้นทาง.....	21
ตารางที่ 3.7 รายละเอียดของยูสเคสเปิดสคีมามาของฐานข้อมูล.....	22
ตารางที่ 3.8 รายละเอียดของยูสเคสลบสคีมามาของฐานข้อมูล.....	22
ตารางที่ 3.9 รายละเอียดของยูสเคสเลือกสคีมามาของฐานข้อมูล.....	23
ตารางที่ 3.10 รายละเอียดของยูสเคสแก้ไขสคีมามาของฐานข้อมูล.....	23
ตารางที่ 3.11 รายละเอียดของยูสเคสปิดสคีมามาของฐานข้อมูล.....	24
ตารางที่ 3.12 รายละเอียดของยูสเคสสร้างข้อมูลทดสอบ.....	24
ตารางที่ 3.13 รายละเอียดของยูสเคสกำหนดรายละเอียดสำหรับการสร้างข้อมูลทดสอบ.....	25
ตารางที่ 3.14 รายละเอียดของยูสเคสสร้างผลลัพธ์.....	26
ตารางที่ 3.15 รายละเอียดของยูสเคสสร้างเพิ่มข้อความที่มีตัวค้น.....	26
ตารางที่ 3.16 รายละเอียดของยูสเคสสร้างเพิ่มเอกสารอิเล็กทรอนิกส์อีเมล.....	27
ตารางที่ 3.17 รายละเอียดของยูสเคสเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมาย.....	27
ตารางที่ 3.18 รายละเอียดของยูสเคสเพิ่มกลุ่มค่าทั่วไปใหม่.....	28
ตารางที่ 3.19 รายละเอียดของยูสเคสแก้ไขกลุ่มค่าทั่วไป.....	29
ตารางที่ 3.20 รายละเอียดของยูสเคสลบกลุ่มค่าทั่วไป.....	29
ตารางที่ 3.21 รายละเอียดของยูสเคสเลือกกลุ่มค่าทั่วไป.....	30
ตารางที่ 3.22 รายละเอียดของยูสเคสกลุ่มค่าทั่วไปซ้ำซ้อน.....	30
ตารางที่ 5.1 รายละเอียดของแต่ละเขตข้อมูลในตารางข้อมูล.....	84
ตารางที่ 5.2 สรุปผลการทดสอบวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบ.....	86
ตารางที่ ก-1 พจนานุกรมข้อมูลของตาราง DatabaseSchema.....	93
ตารางที่ ก-2 พจนานุกรมข้อมูลของตาราง DbTable.....	93
ตารางที่ ก-3 พจนานุกรมข้อมูลของตาราง DbField.....	93

สารบัญตาราง

ตาราง	หน้า
ตารางที่ ก-4 พจนานุกรมข้อมูลของตาราง FieldType	95
ตารางที่ ก-5 พจนานุกรมข้อมูลของตาราง NormalDependency.....	95
ตารางที่ ก-6 พจนานุกรมข้อมูลของตาราง SQLDependency	95
ตารางที่ ก-7 พจนานุกรมข้อมูลของตาราง Category	96
ตารางที่ ก-8 พจนานุกรมข้อมูลของตาราง CategoryValue	96
ตารางที่ ข-1 ตารางสถานพยาบาล	97
ตารางที่ ข-2 ตารางกลุ่มโรค	97
ตารางที่ ข-3 ตารางการแพทย์	98
ตารางที่ ข-4 ตารางการแพ้สารอื่น	98
ตารางที่ ข-5 ตารางประเภทการรักษา.....	99
ตารางที่ ข-6 ตารางรายการจ่ายยาย่อย	99
ตารางที่ ข-7 ตารางระดับสั่งเพิ่มยา	99
ตารางที่ ข-8 ตารางผู้จ่ายยา	100
ตารางที่ ข-9 ตารางเจ้าหน้าที่เวชระเบียน.....	101
ตารางที่ ข-10 ตารางคลังยา.....	101
ตารางที่ ข-11 ตารางเดือน	101
ตารางที่ ข-12 ตารางกำหนดเลขที่ใบสั่งยา.....	102
ตารางที่ ข-13 ตารางประเภทบุคคล.....	102
ตารางที่ ข-14 ตารางความสัมพันธ์.....	103
ตารางที่ ข-15 ตารางค่าน้ำหนัก.....	103
ตารางที่ ข-16 ตารางสถานภาพสมรส.....	103
ตารางที่ ข-17 ตารางกลุ่มผู้ใช้โปรแกรม	104
ตารางที่ ข-18 ตารางผู้ใช้โปรแกรม	104
ตารางที่ ข-19 ตารางกลุ่มผู้ใช้ระบบ.....	105
ตารางที่ ข-20 ตารางประเภทผู้ใช้ระบบ	105
ตารางที่ ข-21 ตารางข้อมูลโปรแกรม	106
ตารางที่ ข-22 ตารางยาของสถานพยาบาล.....	107
ตารางที่ ข-23 ตารางรายการจ่ายยา	108

สารบัญตาราง

ตาราง	หน้า
ตารางที่ ข-24 ตารางการใช้ยา	109
ตารางที่ ข-25 ตารางแพทย์.....	110
ตารางที่ ข-26 ตารางการนัดแพทย์.....	111
ตารางที่ ข-27 ตารางยา.....	112
ตารางที่ ข-28 ตารางประวัติการรักษา	113
ตารางที่ ข-29 ตารางใบสั่งยา.....	114
ตารางที่ ข-30 ตารางลงทะเบียนผู้ป่วย.....	115
ตารางที่ ข-31 ตารางพนักงานและลูกจ้าง	116
ตารางที่ ข-32 ตารางครอบครัว.....	118
ตารางที่ ข-33 ตารางผู้เกษียณ	119
ตารางที่ ข-34 ตารางบุคคลภายนอก	120
ตารางที่ ข-35 ตารางที่อยู่บุคคลภายนอก.....	121
ตารางที่ ค-1 ตาราง T_CITYSPEC.....	122
ตารางที่ ค-2 ตาราง T_NEARCITY	122
ตารางที่ ค-3 ตาราง T_RANK	123
ตารางที่ ค-4 ตาราง T_CODEU	123
ตารางที่ ค-5 ตาราง T_CODEFIELD.....	124
ตารางที่ ค-6 ตาราง T_STATE	124
ตารางที่ ค-7 ตาราง T_FIELD	125
ตารางที่ ค-8 ตาราง T_UDETAIL	126
ตารางที่ ค-9 ตาราง T_MARK.....	127
ตารางที่ ค-10 ตาราง T_ADMIN	127
ตารางที่ ง-1 ตารางแสดงข้อมูลทดสอบที่ได้จากการทดสอบวัดเวลาตอบสนองในการสร้าง ข้อมูลทดสอบ	128

สารบัญภาพ

ภาพประกอบ	หน้า
รูปที่ 2.1 ตัวอย่างเอกสารอิเล็กทรอนิกส์เอ็มแอล.....	7
รูปที่ 2.2 ตัวอย่างอิเล็กทรอนิกส์เอ็มแอลสคีมา	9
รูปที่ 3.1 โครงสร้างการทำงานของเครื่องมือ	13
รูปที่ 3.2 แผนภาพยูสเคสโดยรวม.....	17
รูปที่ 3.3 ยูสเคสภายในแพ็คเกจการจัดการสคีมาของฐานข้อมูล	18
รูปที่ 3.4 ยูสเคสภายในแพ็คเกจการสร้างข้อมูลทดสอบ	24
รูปที่ 3.5 ยูสเคสภายในแพ็คเกจการจัดการกลุ่มค่าทั่วไป.....	28
รูปที่ 3.6 แผนภาพคลาสของเครื่องมือสร้างข้อมูลทดสอบ	31
รูปที่ 3.7 คลาส TestDataGenerator.....	32
รูปที่ 3.8 คลาส DatabaseSchema	32
รูปที่ 3.9 คลาส Table.....	33
รูปที่ 3.10 คลาส Field.....	34
รูปที่ 3.11 คลาส LogicalDependency.....	34
รูปที่ 3.12 คลาส SQLDependency	34
รูปที่ 3.13 คลาส Category.....	35
รูปที่ 3.14 คลาส DBConnection	35
รูปที่ 3.15 แผนภาพซีควเอนซ์แสดงการกำหนดชื่อสคีมาของฐานข้อมูล.....	36
รูปที่ 3.16 แผนภาพซีควเอนซ์แสดงการเพิ่มข้อมูลให้กับสคีมาของฐานข้อมูล	37
รูปที่ 3.17 แผนภาพซีควเอนซ์แสดงการกำหนดการขึ้นต่อกันเชิงตรรกะแบบปกติ	38
รูปที่ 3.18 แผนภาพซีควเอนซ์แสดงการกำหนดการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล.....	38
รูปที่ 3.19 แผนภาพซีควเอนซ์แสดงการกำหนดความคงสภาพในการอ้างอิง	39
รูปที่ 3.20 แผนภาพคอลลาบอเรชั่นแสดงการสร้างสคีมาของฐานข้อมูลจากฐานข้อมูล ต้นทาง.....	40
รูปที่ 3.21 แผนภาพซีควเอนซ์แสดงการเปิดสคีมาของฐานข้อมูล	40
รูปที่ 3.22 แผนภาพซีควเอนซ์แสดงการปิดสคีมาของฐานข้อมูล	41
รูปที่ 3.23 แผนภาพซีควเอนซ์แสดงการลบสคีมาของฐานข้อมูล	42
รูปที่ 3.24 แผนภาพซีควเอนซ์แสดงการกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย.....	42
รูปที่ 3.25 แผนภาพซีควเอนซ์แสดงการเพิ่มกลุ่มค่าทั่วไปใหม่.....	43

สารบัญภาพ (ต่อ)

ภาพประกอบ	หน้า
รูปที่ 3.26 แผนภาพซีควเอนซ์แสดงการลบกลุ่มค่าทั่วไป	44
รูปที่ 3.27 แผนภาพลำดับกิจกรรมแสดงการกำหนดรายละเอียดการสร้างข้อมูลทดสอบ	45
รูปที่ 3.28 แผนภาพลำดับกิจกรรมแสดงการกำหนดรูปแบบการบันทึกข้อมูลทดสอบ	45
รูปที่ 3.29 แผนภาพลำดับกิจกรรมแสดงกระบวนการสร้างข้อมูลทดสอบ	46
รูปที่ 3.30 แผนภาพลำดับกิจกรรมแสดงการตรวจสอบตารางข้อมูลในการสร้างข้อมูล ทดสอบ.....	47
รูปที่ 3.31 แผนภาพลำดับกิจกรรมของ Table.canGenerateTestData()	47
รูปที่ 3.32 แผนภาพลำดับกิจกรรมแสดงการสร้างข้อมูลทดสอบ	48
รูปที่ 3.33 แผนภาพลำดับกิจกรรมของเมธอด Table.generateTestData()	49
รูปที่ 3.34 แผนภาพลำดับกิจกรรมของเมธอด Table.generateTableData()	51
รูปที่ 3.35 แผนภาพลำดับกิจกรรมของเมธอด Field.generateTestData()	52
รูปที่ 3.36 แผนภาพลำดับกิจกรรมของเมธอด Field.deriveNewData().....	53
รูปที่ 3.37 แผนภาพลำดับกิจกรรมของเมธอด Table.generateFKTable()	54
รูปที่ 3.38 แผนภาพลำดับกิจกรรมของเมธอด possibleToGenerate().....	55
รูปที่ 3.39 แผนภาพลำดับกิจกรรมของเมธอด Table.generateFKTableData().....	57
รูปที่ 3.40 แผนภาพลำดับกิจกรรมแสดงการสร้างเพิ่มข้อความที่มีตัวค้น	58
รูปที่ 3.41 แผนภาพลำดับกิจกรรมแสดงการสร้างเพิ่มเอกสารอิเล็กทรอนิกส์อีเมล	59
รูปที่ 3.42 แผนภาพลำดับกิจกรรมของเมธอด deriveMySQLXMLFile().....	61
รูปที่ 3.43 แผนภาพลำดับกิจกรรมของเมธอด deriveOracleXMLFiles().....	62
รูปที่ 3.44 แผนภาพลำดับกิจกรรมของเมธอด deriveAccessXMLFiles()	63
รูปที่ 3.45 แผนภาพลำดับกิจกรรมแสดงการเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมาย	64
รูปที่ 4.1 โครงสร้างฐานข้อมูลของเครื่องมือ	66
รูปที่ 4.2 แผนภาพส่วนประกอบแสดงส่วนประกอบในการนำเครื่องมือไปใช้งาน.....	67
รูปที่ 4.3 แผนภาพส่วนประกอบแสดงโครงสร้างหน้าจอของเครื่องมือ	68
รูปที่ 4.4 หน้าจอหลัก.....	69
รูปที่ 4.5 หน้าจอสร้างสคีมาของฐานข้อมูลใหม่	70
รูปที่ 4.6 หน้าจอสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง	70
รูปที่ 4.7 หน้าจอเปิดสคีมาของฐานข้อมูล	71

สารบัญภาพ (ต่อ)

ภาพประกอบ	หน้า
รูปที่ 4.8 หน้าจอลบสคีมาของฐานข้อมูล	71
รูปที่ 4.9 หน้าจอแก้ไขสคีมาของฐานข้อมูล	72
รูปที่ 4.10 หน้าจอแก้ไขตารางข้อมูล	72
รูปที่ 4.11 หน้าจอกำหนดการขึ้นต่อกันเชิงตรรกะ	73
รูปที่ 4.12 หน้าจอแก้ไขเขตข้อมูล	74
รูปที่ 4.13 หน้าจอกำหนดความคงสภาพในการอ้างอิง	74
รูปที่ 4.14 หน้าจอกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย	75
รูปที่ 4.15 หน้าจอกำหนดรายละเอียดการสร้างข้อมูลทดสอบ	76
รูปที่ 4.16 หน้าจอเลือกกลุ่มค่าทั่วไป	77
รูปที่ 4.17 หน้าจอกำหนดรูปแบบการบันทึกข้อมูลทดสอบ	77
รูปที่ 4.18 หน้าจอเพิ่มกลุ่มค่าทั่วไป	78
รูปที่ 4.19 หน้าจอแก้ไขค่าข้อมูล	78
รูปที่ 4.20 หน้าจอลบกลุ่มค่าทั่วไป	79
รูปที่ 4.21 หน้าจอแสดงรายการกลุ่มค่าทั่วไป	79
รูปที่ 4.22 หน้าจอแก้ไขกลุ่มค่าทั่วไป	80
รูปที่ ๑-1 การสร้างสคีมาของฐานข้อมูลด้วยตัวเอง	129
รูปที่ ๑-2 การกำหนดชื่อสคีมาของฐานข้อมูล	129
รูปที่ ๑-3 ข้อความแจ้งเตือนชื่อสคีมาของฐานข้อมูลซ้ำ	130
รูปที่ ๑-4 หน้าแก้ไขสคีมาของฐานข้อมูล	130
รูปที่ ๑-5 การเพิ่มตารางข้อมูล	130
รูปที่ ๑-6 การเลือกลบตารางข้อมูล	131
รูปที่ ๑-7 หน้าขอคำยืนยันในการลบตารางข้อมูล	131
รูปที่ ๑-8 การเลือกแก้ไขตารางข้อมูล	131
รูปที่ ๑-9 หน้าแก้ไขตารางข้อมูล	132
รูปที่ ๑-10 การเพิ่มเขตข้อมูล	132
รูปที่ ๑-11 การเลือกลบเขตข้อมูล	132
รูปที่ ๑-12 หน้าขอคำยืนยันในการลบเขตข้อมูล	133
รูปที่ ๑-13 การเลือกแก้ไขเขตข้อมูล	133

สารบัญภาพ (ต่อ)

ภาพประกอบ	หน้า
รูปที่ ๑-14 หน้าแก้ไขคุณสมบัติของเขตข้อมูล	134
รูปที่ ๑-15 การเลือกกำหนดการขึ้นต่อกันเชิงตรรกะ	135
รูปที่ ๑-16 หน้ากำหนดการขึ้นต่อกันเชิงตรรกะ	135
รูปที่ ๑-17 การเพิ่มเงื่อนไขการขึ้นต่อกันเชิงตรรกะ	136
รูปที่ ๑-18 การแก้ไขเงื่อนไขการขึ้นต่อกันเชิงตรรกะ	136
รูปที่ ๑-19 การลบเงื่อนไขการขึ้นต่อกันเชิงตรรกะ	137
รูปที่ ๑-20 หน้าขอคำยืนยันในการลบเงื่อนไขการขึ้นต่อกันเชิงตรรกะ	137
รูปที่ ๑-21 การกำหนดการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล	137
รูปที่ ๑-22 การเลือกกำหนดคีย์อ้างอิง	138
รูปที่ ๑-23 หน้ากำหนดคีย์อ้างอิง	138
รูปที่ ๑-24 การสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลที่มีอยู่	139
รูปที่ ๑-25 หน้ากำหนดการเชื่อมต่อกับฐานข้อมูลเพื่อสร้างสคีมาของฐานข้อมูล	140
รูปที่ ๑-26 การเปิดสคีมาของฐานข้อมูล	140
รูปที่ ๑-27 หน้าเปิดสคีมาของฐานข้อมูล	141
รูปที่ ๑-28 การแก้ไขสคีมาของฐานข้อมูล	141
รูปที่ ๑-29 การปิดสคีมาของฐานข้อมูล	142
รูปที่ ๑-30 การลบสคีมาของฐานข้อมูล	143
รูปที่ ๑-31 หน้าลบสคีมาของฐานข้อมูล	143
รูปที่ ๑-32 หน้าขอคำยืนยันในการลบสคีมาของฐานข้อมูล	144
รูปที่ ๑-33 การเชื่อมต่อกับฐานข้อมูลเป้าหมาย	144
รูปที่ ๑-34 หน้ากำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย	145
รูปที่ ๑-35 การสร้างข้อมูลทดสอบ	145
รูปที่ ๑-36 หน้ากำหนดรายละเอียดการสร้างข้อมูลทดสอบ	146
รูปที่ ๑-37 การเลือกตารางข้อมูลที่ต้องการสร้างข้อมูลทดสอบ	146
รูปที่ ๑-38 การเลือกฐานข้อมูลเป้าหมาย	147
รูปที่ ๑-39 การกำหนดจำนวนระเบียบข้อมูลทดสอบที่ต้องการสร้าง	147
รูปที่ ๑-40 การกำหนดวิธีการสร้างข้อมูลทดสอบ	147
รูปที่ ๑-41 การกำหนดให้เครื่องมือสร้างข้อมูลทดสอบอัตโนมัติ	147

สารบัญญภาพ (ต่อ)

ภาพประกอบ	หน้า
รูปที่ ๑-42 หน้ากำหนดช่วงของค่าที่ต้องการสุ่มสร้างข้อมูลทดสอบ.....	148
รูปที่ ๑-43 การกำหนดให้ใช้ค่าคงที่.....	148
รูปที่ ๑-44 การกำหนดให้ใช้ค่าว่าง.....	148
รูปที่ ๑-45 การกำหนดให้ใช้กลุ่มค่าทั่วไป.....	148
รูปที่ ๑-46 หน้ารายการกลุ่มค่าทั่วไป.....	149
รูปที่ ๑-47 หน้าเลือกรูปแบบการบันทึกข้อมูลทดสอบ.....	150
รูปที่ ๑-48 การเพิ่มกลุ่มค่าทั่วไป.....	150
รูปที่ ๑-49 หน้าเพิ่มข้อมูลกลุ่มค่าทั่วไป.....	151
รูปที่ ๑-50 การแก้ไขกลุ่มค่าทั่วไป.....	151
รูปที่ ๑-51 การเลือกกลุ่มทั่วไปที่ต้องการแก้ไข.....	152
รูปที่ ๑-52 หน้าแก้ไขกลุ่มค่าทั่วไป.....	152
รูปที่ ๑-53 การลบกลุ่มค่าทั่วไป.....	153
รูปที่ ๑-54 การเลือกกลุ่มค่าทั่วไปที่ต้องการลบ.....	153
รูปที่ ๑-55 หน้าขอคำยืนยันในการลบกลุ่มค่าทั่วไป.....	154

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การทดสอบซอฟต์แวร์ (Software Testing) เป็นขั้นตอนหนึ่งที่สำคัญมากในกระบวนการพัฒนาซอฟต์แวร์ และยังเป็นขั้นตอนที่ต้องใช้แรงงาน (Effort) และค่าใช้จ่ายสูงอีกด้วย เนื่องจากการทดสอบซอฟต์แวร์เป็นการประมวลผล (Execute) โปรแกรมเพื่อหาว่าโปรแกรมนั้นมีข้อผิดพลาด (Error) หรือไม่ จะได้วิเคราะห์หาสาเหตุและทำการแก้ไขโปรแกรมให้ถูกต้องต่อไป ในปัจจุบันซอฟต์แวร์ถูกนำไปประยุกต์ใช้กับงานหลายๆ อย่างที่ต้องเน้นความถูกต้องและความแม่นยำของการทำงานเป็นสิ่งสำคัญ เช่น การใช้ซอฟต์แวร์ช่วยงานทางการแพทย์ หรือควบคุมเครื่องจักรที่มีมูลค่ามหาศาล เป็นต้น จะเห็นได้ว่าถ้ามีความผิดพลาดเกิดขึ้นในการทำงานเหล่านี้ จะนำมาซึ่งการสูญเสียและความเสียหายที่มากมายมหาศาลเลยทีเดียว ดังนั้นการทดสอบซอฟต์แวร์จึงต้องมีประสิทธิภาพเพียงพอที่จะช่วยลดความเสี่ยงที่อาจจะเกิดขึ้นระหว่างการใช้งานซอฟต์แวร์ได้

การทดสอบซอฟต์แวร์จำเป็นต้องใช้กรณีทดสอบ (Test Cases) มาทดสอบการทำงานในด้านต่างๆ ของซอฟต์แวร์ และนอกจากกรณีทดสอบแล้ว ผู้ทดสอบยังต้องเตรียมข้อมูลทดสอบเอาไว้ในฐานข้อมูล หรือเพิ่มข้อมูลต่างๆ ที่ซอฟต์แวร์ที่กำลังถูกทดสอบจะมีการเรียกใช้งานด้วย โดยการเตรียมข้อมูลสามารถทำได้จากการป้อนด้วยมือ คัดลอกมาจากข้อมูลเก่า หรือใช้เครื่องมือซอฟต์แวร์ (Software tools) มาช่วยสร้างข้อมูลอย่างอัตโนมัติ แต่หลังจากที่ซอฟต์แวร์มีความซับซ้อนมากขึ้นและมีความต้องการใช้ข้อมูลในปริมาณที่มากขึ้น การเตรียมข้อมูลด้วยวิธีการป้อนด้วยมือ และคัดลอกมาจากข้อมูลเก่าจึงไม่ใช่วิธีการที่มีประสิทธิภาพและเหมาะสมอีกต่อไป การสร้างข้อมูลอย่างอัตโนมัติโดยใช้เครื่องมือซอฟต์แวร์นั้น จะช่วยลดค่าใช้จ่ายและเวลาในการพัฒนาและทดสอบซอฟต์แวร์ไปได้มากเลยทีเดียว

ในปัจจุบันมีเครื่องมือที่ช่วยสร้างข้อมูลทดสอบอยู่มากมาย โดยเครื่องมือเหล่านี้จะยึดรูปแบบชนิดข้อมูลของแต่ละเขตข้อมูลและโครงสร้างของตารางข้อมูลเป็นหลักในการสร้างข้อมูลทดสอบ แต่เครื่องมือส่วนใหญ่ไม่มีการคำนึงถึงการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูล และความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลที่อยู่ในฐานข้อมูล ซึ่งเป็นสิ่งสำคัญอย่างหนึ่งของฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ทำให้ข้อมูลทดสอบที่สร้าง

มาได้นั้นไม่เหมือนกับข้อมูลที่ได้จากการใช้งานจริงส่งผลให้สภาวะแวดล้อมของการทดสอบซอฟต์แวร์ไม่เหมือนการใช้งานจริงไปด้วย สำหรับรูปแบบการบันทึกข้อมูลทดสอบที่สร้างขึ้นมานั้นโดยทั่วไปเครื่องมือจะบันทึกข้อมูลได้ 2 รูปแบบคือบันทึกเป็นแฟ้มข้อความที่มีตัวคั่น (Delimited Text file) และบันทึกลงในฐานข้อมูล แต่ในปัจจุบันมีรูปแบบการบันทึกข้อมูลแบบใหม่ที่ได้รับคามนิยมอย่างมากคือเอ็กซ์เอ็มแอล (XML) โดยข้อมูลที่อยู่ในรูปแบบเอ็กซ์เอ็มแอลจะไม่ขึ้นต่อซอฟต์แวร์หรือระบบปฏิบัติการใดๆ ทำให้สามารถนำข้อมูลไปใช้งานกับซอฟต์แวร์คนละตัวหรือใช้งานข้ามระบบปฏิบัติการได้ทันทีโดยไม่จำเป็นต้องแปลงรูปข้อมูลก่อน ด้วยเหตุนี้เอ็กซ์เอ็มแอลจึงเป็นรูปแบบการบันทึกข้อมูลที่น่าสนใจมากแต่เครื่องมือสร้างข้อมูลทดสอบส่วนใหญ่ยังไม่สามารถบันทึกข้อมูลในรูปแบบเอ็กซ์เอ็มแอลได้

วิทยานิพนธ์นี้จึงออกแบบและพัฒนาเครื่องมือช่วยสร้างข้อมูลทดสอบที่สามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูล และความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลภายในฐานข้อมูลได้ ซึ่งจะทำให้ได้ข้อมูลทดสอบที่มีความใกล้เคียงกับข้อมูลที่ได้จากการใช้งานจริงมากขึ้น โดยข้อมูลทดสอบที่สร้างขึ้นสามารถบันทึกลงในฐานข้อมูล หรือบันทึกให้อยู่ในรูปแบบแฟ้มข้อความที่มีตัวคั่นหรือแฟ้มเอกสารเอ็กซ์เอ็มแอลได้

1.2 วัตถุประสงค์ของการวิจัย

เพื่อออกแบบและพัฒนาเครื่องมือช่วยสร้างข้อมูลทดสอบที่สามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูล และความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลภายในฐานข้อมูลได้

1.3 ขอบเขตของการวิจัย

1. ผู้ใช้เครื่องมือสามารถกำหนดชนิดข้อมูลให้กับแต่ละเขตข้อมูลได้ 7 ชนิด ได้แก่ ตัวอักษร (Character) จำนวนเต็ม (Integer) จำนวนทศนิยม (Double) ค่าตรรกะ (Boolean) ค่าวันเดือนปี (Date) ค่าเวลา (Time) และค่าตราเวลา (Timestamp)
2. ผู้ใช้เครื่องมือสามารถกำหนดวิธีการสร้างข้อมูลให้กับแต่ละเขตข้อมูลได้ 4 วิธี ได้แก่ กำหนดให้เครื่องมือสุ่มสร้างให้เองโดยอัตโนมัติ กำหนดให้เป็นค่าคงที่ กำหนดให้เป็นค่าว่าง และกำหนดให้ใช้กลุ่มค่าทั่วไป
3. ผู้ใช้สามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลที่ต้องการจะสร้างข้อมูลได้ โดยจะต้องเป็นเขตข้อมูลที่อยู่ภายในตารางข้อมูลเดียวกันและจำนวนเขตข้อมูลที่เกี่ยวข้องในแต่ละความสัมพันธ์จะมีได้ 2 เขตข้อมูลเท่านั้น

4. ผู้ใช้สามารถกำหนดความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลภายในสคีมาของฐานข้อมูลที่ต้องการจะสร้างข้อมูลได้ โดยจะต้องเป็นความสัมพันธ์แบบสองทางระหว่างเขตข้อมูล 2 เขตข้อมูลเท่านั้น
5. ผู้ใช้สามารถเลือกตารางข้อมูลที่ต้องการสร้างข้อมูลทดสอบจากตารางข้อมูลทั้งหมดภายในสคีมาของฐานข้อมูลเดียวกันได้
6. ถ้าเป็นการบันทึกข้อมูลทดสอบลงในฐานข้อมูลเป้าหมาย ภายในฐานข้อมูลเป้าหมายต้องมีตารางข้อมูลที่มีโครงสร้างตรงกับตารางข้อมูลในสคีมาของฐานข้อมูลที่เครื่องมือสร้างข้อมูลทดสอบไว้
7. เครื่องมือสามารถบันทึกข้อมูลทดสอบที่สร้างได้ 3 รูปแบบ คือ
 - 7.1 บันทึกข้อมูลเป็นแฟ้มข้อความแบบที่มีตัวคั่น
 - 7.2 บันทึกข้อมูลเป็นแฟ้มเอกสารเอ็กซ์เอ็มแอลตามรูปแบบของฐานข้อมูลเป้าหมาย
 - 7.3 บันทึกข้อมูลลงในฐานข้อมูลเป้าหมาย โดยถ้าฐานข้อมูลเป้าหมายเป็นฐานข้อมูลมายเอสคิวแอลหรือฐานข้อมูลออราเคิลจะทำการบันทึกโดยตรง แต่ถ้าเป็นฐานข้อมูลไมโครซอฟท์แอ็คเซสจะบันทึกผ่านทาง การเชื่อมต่อโอดีบีซี (ODBC: Open Database Connectivity)
8. เครื่องมือสามารถใช้งานบนระบบปฏิบัติการไมโครซอฟท์วินโดวส์ (Microsoft Windows) ได้
9. เครื่องมือที่พัฒนาเสร็จแล้วถูกนำไปทดสอบกับระบบงานจริง 2 ระบบ โดยนำไปใช้สร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลของระบบนั้น และพิจารณาข้อมูลทดสอบที่ได้ว่าเป็นไปตามการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูลและความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลหรือไม่

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ข้อมูลทดสอบที่ได้จากเครื่องมือที่มีความใกล้เคียงกับข้อมูลที่ได้จากการใช้งานจริง
2. ช่วยลดเวลาในการสร้างข้อมูลทดสอบให้น้อยลง
3. ช่วยลดเวลารวมที่ใช้ในกระบวนการทดสอบซอฟต์แวร์ให้น้อยลง

1.5 ขั้นตอนการวิจัย

1. สสำรวจและศึกษาเครื่องมือสร้างข้อมูลทดสอบที่มีอยู่ในปัจจุบัน
2. ศึกษารูปแบบภาษาเอสคิวแอลของฐานข้อมูลที่เครื่องมือจะต้องติดต่อกับ

3. ศึกษาวิธีการเขียนและการใช้งานภาษาเอ็กซ์เซลแมค
4. กำหนดขอบเขตความสามารถที่เครื่องมือจะทำได้
5. ออกแบบหน้าจอและวิธีการสร้างข้อมูลทดสอบของเครื่องมือที่จะพัฒนา
6. พัฒนาเครื่องมือสร้างข้อมูลทดสอบตามที่ได้ออกแบบไว้
7. ทดสอบเครื่องมือที่พัฒนาขึ้นตามขอบเขตที่ได้กำหนดไว้
8. สรุปผลการวิจัย และข้อเสนอแนะ
9. จัดทำรายงานวิทยานิพนธ์



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับการทำวิทยานิพนธ์มีดังนี้

2.1.1 การทดสอบซอฟต์แวร์

การทดสอบซอฟต์แวร์ [1] หมายถึง กระบวนการในการใช้งานหรือประเมินค่าซอฟต์แวร์ทั้งการทำได้ด้วยมือ หรือทำอย่างอัตโนมัติ เพื่อที่จะตรวจสอบว่าซอฟต์แวร์เป็นไปตามความต้องการของซอฟต์แวร์ (Software requirements) หรือเพื่อที่จะระบุความแตกต่างระหว่างผลลัพธ์ที่คาดหวังกับผลลัพธ์ที่ได้จริงจากซอฟต์แวร์

เป้าหมายของการทดสอบซอฟต์แวร์ [2] มีดังนี้

1. สามารถป้องกันความผิดพลาด (Bug) ไม่ให้เกิดขึ้นกับซอฟต์แวร์ได้
2. ถ้าการทดสอบไม่สามารถป้องกันความผิดพลาดไม่ให้เกิดขึ้นได้ ก็ควรจะ สามารถบอกได้ว่าความผิดพลาดนั้นจะส่งผลกระทบต่อซอฟต์แวร์อย่างไร
3. การทดสอบควรจะบอกแนวทางแก้ไขความผิดพลาดที่ชัดเจน เพื่อให้สามารถแก้ไขความผิดพลาดได้อย่างง่ายดาย

การทดสอบซอฟต์แวร์แบ่งเป็นระดับได้ 4 ระดับ [3] ดังนี้

1. การทดสอบหน่วยโปรแกรม (Unit Testing) คือการทดสอบหน่วยโปรแกรมหลังจากที่เขียนโปรแกรมในหน่วยนั้นเสร็จแล้ว
2. การทดสอบแบบบูรณาการ (Integration Testing) คือการทดสอบซอฟต์แวร์ โดยนำแต่ละหน่วยโปรแกรมที่ผ่านการทดสอบหน่วยโปรแกรมมาแล้วมาทำงานร่วมกัน
3. การทดสอบระบบ (System Testing) คือการทดสอบว่าซอฟต์แวร์ที่พัฒนาได้นั้นทำงานได้ตรงกับข้อกำหนดความต้องการของลูกค้า (Customer requirements) หรือไม่
4. การทดสอบการตรวจรับ (Acceptance Testing) คือการนำเอาซอฟต์แวร์ไปทดสอบกับสิ่งแวดล้อมทางปฏิบัติที่ซอฟต์แวร์จะถูกนำไปใช้งานจริงเพื่อยืนยันว่าซอฟต์แวร์นั้นสามารถทำงานได้ตามความต้องการที่กำหนดไว้จริงๆ

2.1.2 เอกซ์เอ็มแอล (XML – eXtensible Markup Language) [4]

เอกซ์เอ็มแอล เป็นภาษาที่ใช้เป็นมาตรฐานในการอธิบายข้อมูลได้ทุกชนิด เอกซ์เอ็มแอลกำเนิดขึ้นมาจากสาเหตุที่ว่า ในปัจจุบันการจัดเก็บข้อมูลที่มีโครงสร้างเป็นลำดับชั้น (Hierarchy) ถูกนำไปใช้ในการสื่อสารระหว่างโปรแกรมต่างๆ บนอินเทอร์เน็ตมากขึ้น และเนื่องจากข้อมูลเหล่านี้มักจะผูกติดกับซอฟต์แวร์ที่สร้างข้อมูลหรือระบบปฏิบัติการที่ใช้สร้างข้อมูลนั้น จึงทำให้การส่งผ่านข้อมูลระหว่างโปรแกรมที่อยู่ต่างระบบปฏิบัติการนั้นค่อนข้างทำได้ยาก ด้วยเหตุนี้จึงจำเป็นต้องมีการกำหนดภาษากลางขึ้นมาเพื่อเป็นมาตรฐานที่จะใช้ในการนิยามและอธิบายข้อมูล และภาษากลางนั้นก็คือนามเอกซ์เอ็มแอลนั่นเอง การนำเอกซ์เอ็มแอลไปใช้งานนั้น นอกจากจะใช้ตัวภาษาเอกซ์เอ็มแอลในการนิยามและอธิบายข้อมูลแล้ว ยังต้องมีการประกาศใช้ดีทีดี (DTD – Document Type Declarations) เพื่อบริหารโครงสร้างของข้อมูลในเอกสารเอกซ์เอ็มแอลด้วย

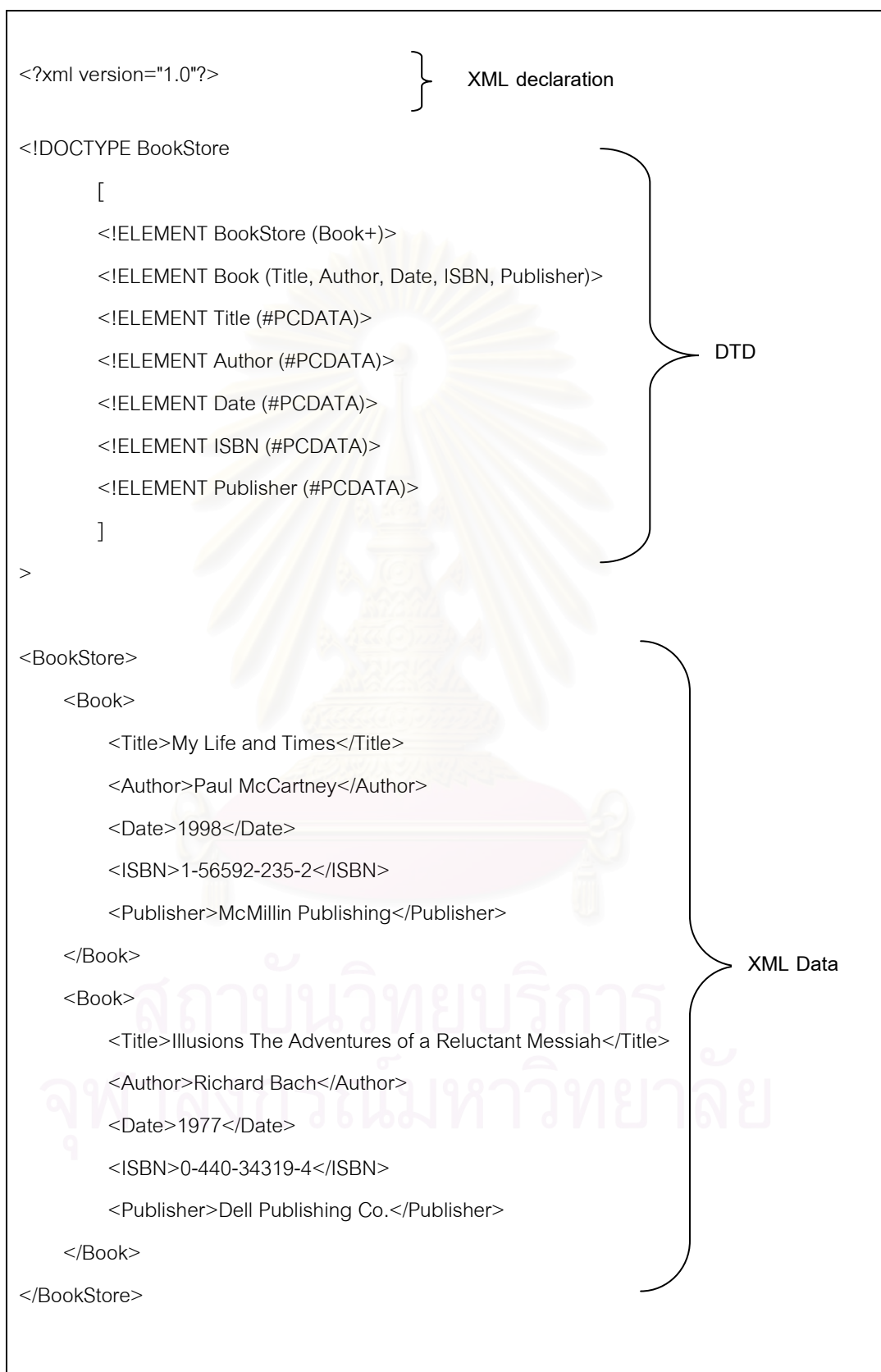
ภาษาเอกซ์เอ็มแอลเป็นภาษามาร์คอัพ (Markup Language) ดังนั้นข้อมูลภายในเอกสารเอกซ์เอ็มแอลจึงประกอบไปด้วยอีลีเมนต์ (Elements) และแอททริบิวต์ (Attribute) ต่างๆ ของแต่ละอีลีเมนต์ที่ใช้ในการอธิบายข้อมูล โดยแต่ละอีลีเมนต์จะเขียนแทนด้วยแท็ก (Tag) ซึ่งเมื่อจบการใช้งานอีลีเมนต์ต้องมีแท็กปิดเสมอ เอกสารเอกซ์เอ็มแอลนั้นสามารถแบ่งได้เป็น 2 ชนิดคือ

1. เอกสารเอกซ์เอ็มแอลที่มีรูปแบบถูกต้อง (Well-Formed XML Document) คือเอกสารเอกซ์เอ็มแอลที่มีคุณสมบัติถูกต้องตามกฎหมายข้อบังคับพื้นฐานของการเขียนเอกสารเอกซ์เอ็มแอล โดยทุกๆ เอกสารเอกซ์เอ็มแอลจะต้องมีรูปแบบถูกต้องเป็นอย่างน้อย สำหรับกฎหมายข้อบังคับพื้นฐานของการเขียนเอกสารเอกซ์เอ็มแอลมีดังนี้

- เอกสารต้องมีอีลีเมนต์ระดับบนสุดเพียงอีลีเมนต์เดียวเท่านั้น (ซึ่งเรียกว่าอีลีเมนต์เอกสาร (Document Element) หรือ อีลีเมนต์ราก (Root Element)) โดยอีลีเมนต์อื่นๆ ทั้งหมดจะต้องซ้อนอยู่ภายใน เช่น จากรูปที่ 2.1 อีลีเมนต์รากคือ <BookStore>

- อีลีเมนต์ต้องซ้อนอยู่อย่างถูกต้อง นั่นหมายความว่าถ้ามีอีลีเมนต์ซ้อนอยู่ในอีลีเมนต์อื่นๆ ต้องปิดอีลีเมนต์นั้นภายในอีลีเมนต์เดียวกับที่อีลีเมนต์นั้นซ้อนอยู่ เช่น จากรูปที่ 2.1 อีลีเมนต์ <Book> ซ้อนอยู่ในอีลีเมนต์ <BookStore> อย่างถูกต้อง

- แต่ละอีลีเมนต์ต้องมีแท็กเริ่มต้นและแท็กปิดท้าย ถ้าแท็กไม่ครบคู่จะทำให้บราวเซอร์ (Browser) ไม่เข้าใจคำสั่งของอีลีเมนต์นั้น



รูปที่ 2.1 ตัวอย่างเอกสารเอ็กซ์เอ็มแอล

- ชื่อของอีลีเมนต์ในแท็กเริ่มต้นจะต้องมีชื่อตรงกับชื่อของแท็กปิดท้ายเท่านั้น
- ชื่อของอีลีเมนต์มีลักษณะเป็น Case-sensitive (การแยกระหว่างตัวอักษรใหญ่และตัวอักษรเล็ก) ซึ่งอันที่จริงข้อความทั้งหมดภายในเอกสารจะมีลักษณะเป็น Case-sensitive ทั้งหมด

2. เอกสารเอ็กซ์เอ็มแอลที่ถูกต้องสมบูรณ์ (Valid XML Document) คือเอกสารเอ็กซ์เอ็มแอลที่มีรูปแบบถูกต้องที่มีการเพิ่มความต้องการ (Requirement) ที่เป็นส่วนเสริม 2 ข้อเข้าไปในเอกสารคือ

- ต้องมีการประกาศการใช้ดีทีดีสำหรับเอกสาร ซึ่งข้อมูลภายในดีทีดีจะเป็นตัวกำหนดโครงสร้างของเอกสารนั้น เช่น จากรูปที่ 2.1 อีลีเมนต์ <!DOCTYPE> คืออีลีเมนต์ที่ประกาศการใช้ดีทีดี โดยภายในดีทีดีอธิบายว่าอีลีเมนต์ <BookStore> ต้องประกอบด้วยอีลีเมนต์ <Book> ตั้งแต่ 1 อีลีเมนต์ขึ้นไป และอีลีเมนต์ <Book> ต้องประกอบด้วยอีลีเมนต์ <Title> อีลีเมนต์ <Author> อีลีเมนต์ <Date> อีลีเมนต์ <ISBN> และอีลีเมนต์ <Publisher> โดยทั้ง 5 อีลีเมนต์จะต้องมีเนื้อหาภายในอีลีเมนต์เป็นข้อมูลแบบอักขระเท่านั้น เป็นต้น

- ส่วนต่างๆ ภายในเอกสารต้องถูกต้องตามโครงสร้างที่กำหนดในดีทีดี จากรูปที่ 2.1 จะเห็นได้ว่า เนื้อหาภายในอีลีเมนต์ <BookStore> นั้นเป็นไปตามดีทีดี

2.1.3 เอ็กซ์เอ็มแอลสคีมา (XML Schemas) [5]

เอ็กซ์เอ็มแอลสคีมาเป็นรูปแบบคำสั่งที่ใช้นิยามและอธิบายโครงสร้างข้อมูลในเอกสารเอ็กซ์เอ็มแอลเหมือนกับดีทีดี เอ็กซ์เอ็มแอลสคีมาถูกสร้างขึ้นมาจากสาเหตุที่ว่าดีทีดีนั้นยังไม่สามารถใช้อธิบายโครงสร้างข้อมูลในเอกสารเอ็กซ์เอ็มแอลได้อย่างมีประสิทธิภาพ เนื่องจากไวยากรณ์ที่ใช้อธิบายข้อมูลของดีทีดีนั้นแตกต่างจากภาษาเอ็กซ์เอ็มแอล ทำให้ผู้ใช้ต้องจำไวยากรณ์ถึง 2 อย่างเพื่อใช้งานเอ็กซ์เอ็มแอล และดีทีดีนั้นกำหนดชนิดข้อมูลมาให้ใช้งานน้อยมากเพียง 10 ชนิดเท่านั้น ทำให้การใช้งานเอ็กซ์เอ็มแอลไม่มีความยืดหยุ่นเท่าที่ควร ดังนั้นเอ็กซ์เอ็มแอลสคีมาจึงถูกสร้างขึ้นมากำจัดข้อบกพร่องเหล่านี้ให้หมดไป โดยเอ็กซ์เอ็มแอลสคีมาจะมีความสามารถที่น่าสนใจดังนี้

1. สนับสนุนชนิดข้อมูลทั้งหมด 44 ชนิด และให้ผู้ใช้สามารถสร้างชนิดข้อมูลเพิ่มเติมเองได้
2. ใช้ไวยากรณ์แบบเดียวกับกับภาษาเอ็กซ์เอ็มแอล
3. มีความเป็นเชิงวัตถุ (Object-oriented) คือสามารถสร้างชนิดข้อมูลใหม่โดยสืบทอด (Extend) หรือจำกัด (Restrict) รูปแบบการใช้งานของชนิดข้อมูลเดิมที่มีอยู่ได้

4. สามารถแสดงข้อมูลในแบบเซต (Set) ได้
5. สามารถกำหนดให้อีลีเมนต์หลายๆ อันมีชื่อเดียวกัน แต่มีเนื้อหาที่ต่างกันได้

ตัวอย่างของเอ็กซ์เอ็มแอลสคีมาที่อธิบายโครงสร้างของอีลีเมนต์ <BookStore> แบบเดียวกับที่ดีที่สุดในรูปที่ 2.1 แสดงได้ดังรูปที่ 2.2

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xsd:element name="BookStore">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Book" minOccurs="1" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Book">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Title" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Author" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Date" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="ISBN" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Publisher" minOccurs="1" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Title" type="xsd:string"/>
  <xsd:element name="Author" type="xsd:string"/>
  <xsd:element name="Date" type="xsd:string"/>
  <xsd:element name="ISBN" type="xsd:string"/>
  <xsd:element name="Publisher" type="xsd:string"/>
</xsd:schema>
```

รูปที่ 2.2 ตัวอย่างเอ็กซ์เอ็มแอลสคีมา

2.1.4 คำศัพท์ที่สำคัญ

1. การขึ้นต่อกันเชิงตรรกะ (Logical Dependency) คือความสัมพันธ์ระหว่างเขตข้อมูลภายในแต่ละตารางข้อมูล โดยความสัมพันธ์จะมีลักษณะเป็นเงื่อนไขระหว่างค่าข้อมูลในเขตข้อมูล เช่น ถ้าค่าข้อมูลในเขตข้อมูลเพศเป็น “ชาย” แล้วค่าข้อมูลในเขตข้อมูลค่านำหน้าชื่อจะต้องเป็น “ด.ช.” หรือ “นาย” เท่านั้น หรือ ถ้าค่าข้อมูลในเขตข้อมูลประเทศเป็น “ประเทศไทย” แล้วค่าข้อมูลในเขตข้อมูลจังหวัดก็ต้องเป็นจังหวัดที่อยู่ในประเทศไทยด้วย เป็นต้น
2. คีย์อ้างอิง (Foreign Key) [6] คือเขตข้อมูลหรือกลุ่มของเขตข้อมูลในตารางข้อมูลหนึ่ง ที่อ้างอิงถึงเขตข้อมูลหรือกลุ่มของเขตข้อมูลที่เป็นคีย์หลัก (Primary Key) ในตารางข้อมูลอื่น
3. ความคงสภาพในการอ้างอิง (Referential Integrity) [6] คือความสัมพันธ์ระหว่างตารางข้อมูลในฐานข้อมูลเชิงสัมพันธ์ โดยจะมีคีย์อ้างอิงเป็นตัวแทนของความสัมพันธ์ที่บังคับให้ข้อมูลระหว่าง 2 ตารางข้อมูลมีความสอดคล้องกัน ซึ่งคีย์อ้างอิงต้องมีค่าสอดคล้องกับเงื่อนไขต่อไปนี้
 - 3.1 คีย์อ้างอิงต้องมีค่าสอดคล้อง (เท่ากับ) กับค่าของระเบียบข้อมูลระเบียบในใดระเบียบหนึ่งของคีย์หลักในตารางข้อมูลที่อ้างอิงถึง หรือ
 - 3.2 ถ้าไม่สอดคล้องกับค่าของระเบียบข้อมูลระเบียบในใดระเบียบหนึ่งของคีย์หลักในตารางข้อมูลที่อ้างอิง จะต้องมีค่าเป็นค่าว่าง (Null)
 ตัวอย่างเช่น ถ้าตารางข้อมูลพนักงานมีเขตข้อมูล “รหัสแผนก” ที่เป็นคีย์อ้างอิงไปยังเขตข้อมูล “รหัสแผนก” ซึ่งเป็นคีย์หลักของตารางข้อมูลแผนก ดังนั้นค่าข้อมูลในเขตข้อมูล “รหัสแผนก” ของตารางข้อมูลพนักงานจะต้องสอดคล้องค่าข้อมูลในเขตข้อมูล “รหัสแผนก” ของตารางข้อมูลแผนกด้วย เพื่อให้ถูกต้องตามความคงสภาพในการอ้างอิง
4. ความสัมพันธ์แบบสองทาง (Binary relationship) คือความสัมพันธ์ที่มีสิ่งที่เกี่ยวข้องเพียง 2 สิ่งเท่านั้น เช่น ความสัมพันธ์แบบสองทางระหว่างเขตข้อมูล จะมีเขตข้อมูลที่สัมพันธ์กันเพียง 2 เขตข้อมูลเท่านั้น

2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการทำวิทยานิพนธ์มีดังนี้

2.2.1 Datatect 1.6 [7]

Datatect 1.6 เป็นเครื่องมือสร้างข้อมูลทดสอบที่พัฒนาขึ้นโดยบริษัท Banner Software เครื่องมือนี้จะช่วยผู้ใช้ในการสร้างข้อมูลทดสอบ โดยให้ผู้ใช้งานกำหนดคุณสมบัติต่างๆ ของแต่ละเขตข้อมูลที่ต้องการจะสร้างข้อมูล แล้วกำหนดรูปแบบของผลลัพธ์ (Output specification) เพื่อระบุว่าต้องการเขตข้อมูลใดบ้างและปริมาณระเบียบข้อมูลที่ต้องการสร้างเป็นเท่าใด จากนั้นเครื่องมือจะสร้างข้อมูลทดสอบด้วยวิธีการสุ่มตามคุณสมบัติของเขตข้อมูลและรูปแบบของผลลัพธ์ที่กำหนด ข้อมูลทดสอบที่ได้มานั้นจะเป็นอิสระกันในแต่ละเขตข้อมูลและข้อมูลทดสอบในแต่ละตารางข้อมูลก็เป็นอิสระจากกันด้วย เนื่องจากเครื่องมือนี้ไม่สามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างแต่ละเขตข้อมูล และความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลได้ เครื่องมือนี้สามารถสร้างข้อมูลทดสอบได้ครั้งละ 1 ตารางและสามารถบันทึกข้อมูลทดสอบได้ 2 รูปแบบคือ บันทึกข้อมูลเป็นแฟ้มข้อความที่มีตัวค้น และบันทึกข้อมูลลงในฐานข้อมูลที่เชื่อมต่ออยู่ทางไอดีบีซี

2.2.2 วิทยานิพนธ์เรื่อง “การพัฒนาเครื่องมือซอฟต์แวร์สำหรับสร้างข้อมูลทดสอบ

(Development of a Software Tool for Generating Test Data)” [8]

เครื่องมือสร้างข้อมูลทดสอบขึ้นนี้ถูกพัฒนาขึ้นโดยนายภพพงศ์ สกุลพิพัฒนศิลป์ เครื่องมือสามารถช่วยผู้สร้างข้อมูลทดสอบ โดยผู้ใช้งานต้องสร้างเขตข้อมูลที่ต้องการสร้างข้อมูล แล้วกำหนดคุณสมบัติต่างๆ ให้เขตข้อมูลนั่นเอง ไม่สามารถอ่านโครงสร้างเหล่านี้ขึ้นมาจากฐานข้อมูลได้ เครื่องมือนี้สามารถกำหนดการขึ้นต่อกันเชิงตรรกะให้กับแต่ละเขตข้อมูลที่ต้องการได้ ซึ่งเครื่องมือจะเรียกว่า เงื่อนไขของเขตข้อมูลที่ขึ้นแก่กัน (Condition of the Dependent Fields) โดยจะให้ผู้ใช้งานกำหนดว่าต้องการสร้างความสัมพันธ์ที่ความสัมพันธ์ แล้วจึงให้ผู้ใช้งานกำหนดความสัมพันธ์ระหว่างแต่ละเขตข้อมูลที่ต้องการ ข้อมูลทดสอบที่ได้จากเครื่องมือนี้จะมีความใกล้เคียงกับข้อมูลจากการใช้งานจริงในระดับหนึ่งแต่ยังไม่สมบูรณ์นัก เนื่องจากยังไม่สามารถกำหนดความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลได้ ทำให้ข้อมูลทดสอบในแต่ละตารางข้อมูลไม่สอดคล้องตามกฎความคงสภาพในการอ้างอิง เครื่องมือนี้สามารถสร้างข้อมูลทดสอบได้ครั้งละ 1 ตารางและสามารถบันทึกข้อมูลทดสอบได้ 2 รูปแบบคือ บันทึกข้อมูลเป็นแฟ้มข้อความที่มีตัวค้น และบันทึกข้อมูลลงในฐานข้อมูลไมโครซอฟท์แอ็คเซส

2.2.3 TurboData [9]

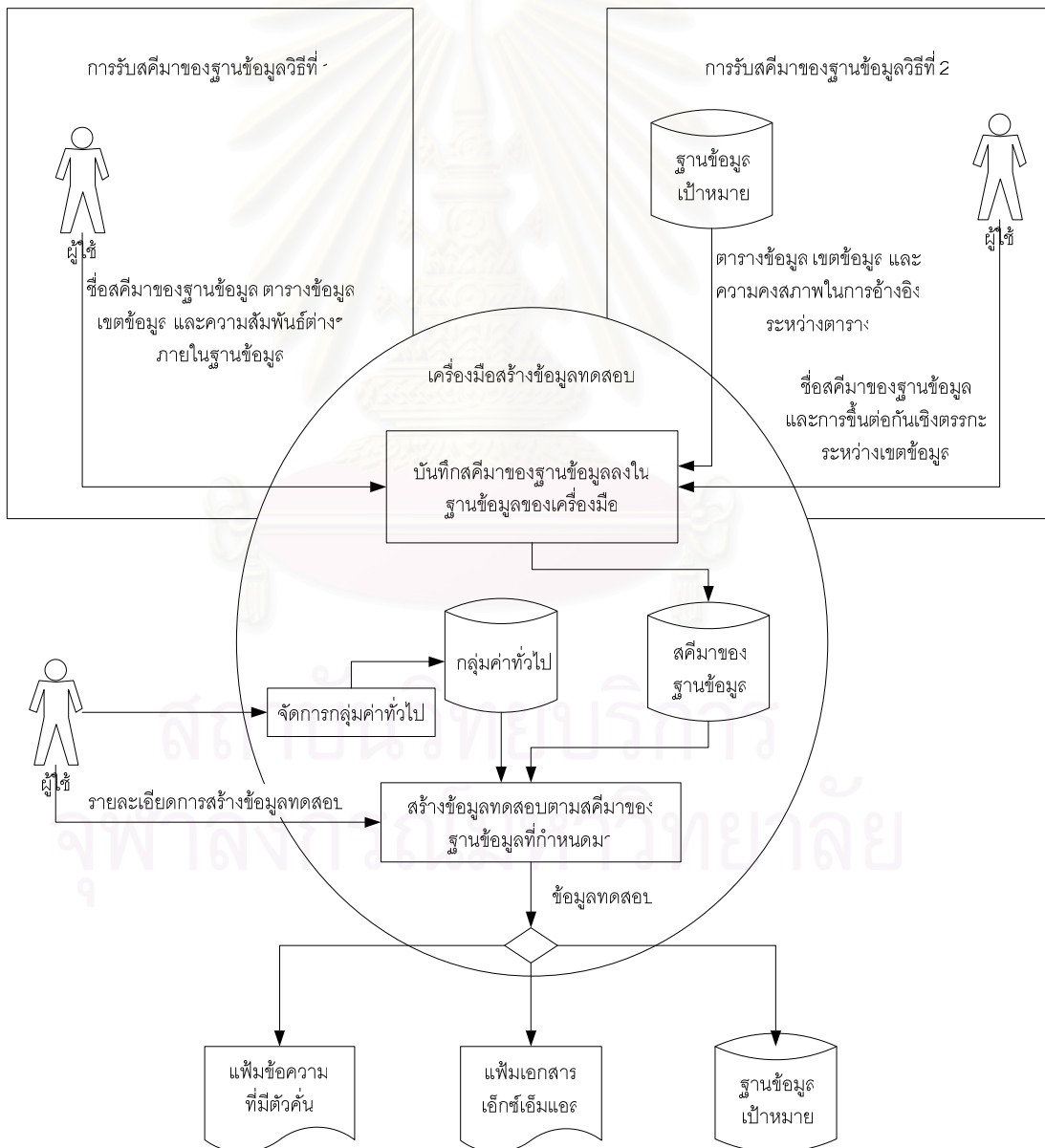
TurboData เป็นเครื่องมือสร้างข้อมูลทดสอบที่พัฒนาขึ้นโดยบริษัท CANAM Software Labs เครื่องมือนี้จะยึดรูปแบบสคีมาของฐานข้อมูล (Database schema) เป็นหลักในการสร้างข้อมูลทดสอบ โดยจะให้ผู้เลือกใช้เลือกได้ว่าจะกำหนดรูปแบบสคีมาของฐานข้อมูลขึ้นมาเอง หรือจะให้เครื่องมืออ่านรูปแบบสคีมาของฐานข้อมูลจากฐานข้อมูลที่กำลังเชื่อมต่ออยู่ สำหรับรูปแบบสคีมาของฐานข้อมูลนั้น เครื่องมือจะให้ผู้ใช้งานกำหนดความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลภายในฐานข้อมูลได้ แต่ไม่สามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในตารางข้อมูลเดียวกันได้ จึงทำให้ข้อมูลทดสอบที่ได้นั้นไม่มีความสมบูรณ์นักในเรื่องความหมายของข้อมูลเมื่อเทียบกับการใช้งานจริง เครื่องมือนี้สามารถสร้างข้อมูลทดสอบให้กับทุกๆ ตารางข้อมูลที่อยู่ในฐานข้อมูลได้ และสามารถบันทึกข้อมูลทดสอบได้ 2 รูปแบบคือ บันทึกข้อมูลเป็นแฟ้มข้อความที่มีตัวคั่น และบันทึกข้อมูลลงในฐานข้อมูลที่เชื่อมต่ออยู่โดยตรง ซึ่งฐานข้อมูลที่เครื่องมือสนับสนุน ได้แก่ ดิบีทูยูนิเวอร์แซล (DB2 Universal) ไมโครซอฟท์แอ็คเซส ไมโครซอฟท์เอสคิวแอลเซิร์ฟเวอร์ (MS/SQL Server) นอนสตอปป์เอสคิวแอล (NonStop SQL) ออราเคิลไซเบส (Sybase) และเอสคิวแอลเอนีแวร์ (SQL Anywhere)

บทที่ 3

การวิเคราะห์และออกแบบเครื่องมือ

3.1 โครงสร้างการทำงานของเครื่องมือ

วิทยานิพนธ์นี้ได้พัฒนาเครื่องมือสร้างข้อมูลทดสอบ โดยข้อมูลทดสอบที่ได้มานั้น เป็นไปตามการขึ้นต่อกันเชิงตรรกะและความคงสภาพในการอ้างอิงของข้อมูลตามผู้ทดสอบ กำหนดไว้ โดยก่อนที่จะอธิบายในส่วนการวิเคราะห์และออกแบบเครื่องมือ นั้น จะขออธิบาย โครงสร้างการทำงานของเครื่องมือก่อน โดยจะแบ่งอธิบายเป็น 2 ส่วน ดังนี้



รูปที่ 3.1 โครงสร้างการทำงานของเครื่องมือ

จากรูปที่ 3.1 โครงสร้างการทำงานของเครื่องมือเริ่มจากการรับสคีมาของฐานข้อมูล โดยเครื่องมือสามารถเลือกรับสคีมาของฐานข้อมูลได้ 2 วิธีคือ

1. รับสคีมาของฐานข้อมูลจากผู้ใช้ทั้งหมด วิธีนี้เครื่องมือจะให้ผู้ใช้กำหนดข้อมูลเกี่ยวกับสคีมาของฐานข้อมูลเองทั้งหมด ดังนี้

1.1 ชื่อของสคีมาของฐานข้อมูล

1.2 ตารางข้อมูลในสคีมาของฐานข้อมูล

1.3 เขตข้อมูลในแต่ละตารางข้อมูล พร้อมทั้งกำหนดคุณสมบัติของเขตข้อมูลด้วย ซึ่งคุณสมบัติของเขตข้อมูลจะประกอบด้วย

- ชนิดข้อมูล (Data Type) ได้แก่ ตัวอักษร (Character) จำนวนเต็ม (Integer) จำนวนทศนิยม (Double) ค่าตรรกะ (Boolean) ค่าวันเดือนปี (Date) ค่าเวลา (Time) และค่าตราเวลา (Timestamp)

- ขนาดของเขตข้อมูล (Size) หรือความเที่ยงตรงของจำนวน (Numeric Precision) ขึ้นอยู่กับชนิดข้อมูล ถ้าเป็นตัวอักษรจะกำหนดเป็นขนาดของเขตข้อมูล แต่ถ้าเป็นจำนวนเต็มหรือจำนวนทศนิยม จะกำหนดเป็นความเที่ยงตรงของจำนวน ซึ่งหมายถึงจำนวนหลักที่เป็นไปได้มากที่สุดของค่าตัวเลข โดยนับรวมจำนวนหลักของทศนิยมด้วยนั่นเอง

- ขนาดของทศนิยม (Numeric Scale) คือจำนวนหลักของทศนิยมสามารถกำหนดได้เฉพาะเขตข้อมูลที่มีชนิดเป็นจำนวนทศนิยมเท่านั้น

- ตัวระบุ (Identifier) หรือคีย์หลัก (Primary Key) จะเป็นการระบุว่าเขตข้อมูลนั้นเป็นคีย์หลักหรือเป็นส่วนหนึ่งของคีย์หลักที่ใช้อ้างอิงถึงแต่ละระเบียนข้อมูลในตารางข้อมูลหรือไม่

- ตัวเลือก (Optional) จะเป็นการระบุว่าเขตข้อมูลนั้นจำเป็นต้องมีค่าข้อมูลในทุกๆ ระเบียนของเขตข้อมูลหรือไม่

1.4 การขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูล โดยผู้ใช้สามารถกำหนดได้ 2 แบบคือ

- การขึ้นต่อกันเชิงตรรกะแบบปกติ (Normal Logical Dependency) คือระบุงการขึ้นต่อกันเชิงตรรกะเป็นเงื่อนไขระหว่าง 2 เขตข้อมูลว่า ถ้าเขตข้อมูลหนึ่งมีค่าเท่ากับค่าๆ หนึ่งแล้ว ค่าของเขตข้อมูลอีกอันหนึ่งจะต้องมีค่าที่เท่ากับค่าที่กำหนดไว้ เช่น มีการกำหนดเงื่อนไขระหว่าง เขต

ข้อมูล "title" กับเขตข้อมูล "sex" ว่า ถ้าเขตข้อมูล "title" มีค่าเท่ากับ "Mr." แล้วเขตข้อมูล "sex" ต้องมีค่าเท่ากับ "Male" เป็นต้น

- การขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล (SQL Logical Dependency) คือระบุงการขึ้นต่อกันเชิงตรรกะเป็นคำสั่งเอสคิวแอล เช่น `select * from employee where if(prefix='Mr.', gender='Male', gender<>'Male')`

1.5 ความคงสภาพในการอ้างอิงระหว่างแต่ละตารางข้อมูลภายในฐานข้อมูล เป็นการกำหนดคีย์อ้างอิง (Foreign Key) ให้กับตารางข้อมูลว่ามีเขตข้อมูลใดบ้างที่อ้างอิงเขตข้อมูลในตารางข้อมูลอื่นๆ ภายในสคีมาของฐานข้อมูล

2. รับสคีมาของฐานข้อมูลจากฐานข้อมูลเป้าหมายและผู้ใช้ วิธีนี้เครื่องมือจะอ่านสคีมาของฐานข้อมูลผ่านทางารเชื่อมต่อกับฐานข้อมูลเป้าหมาย (ฐานข้อมูลที่เครื่องมือจะบันทึกข้อมูลทดสอบที่สร้างได้) ที่กำหนดโดยผู้เข้ามาเก็บไว้ โดยเครื่องมือสามารถอ่านสคีมาของฐานข้อมูลจากฐานข้อมูล 3 ชนิดคือ ฐานข้อมูลมายเอสคิวแอล (MySQL database) ฐานข้อมูลออราเคิล (Oracle database) และฐานข้อมูลไมโครซอฟท์แอคเซส (Microsoft Access) แต่เนื่องจากสคีมาของฐานข้อมูลที่ได้จากฐานข้อมูลเป้าหมายเหล่านั้นนั้นจะมีเพียงรายการตารางข้อมูล เขตข้อมูลในแต่ละตาราง และความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลเท่านั้น ดังนั้นผู้ใช้จะต้องกำหนดชื่อของสคีมาของฐานข้อมูลและการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางข้อมูลเพิ่มเติม

ข้อมูลเกี่ยวกับสคีมาของฐานข้อมูลทั้งหมดจะถูกบันทึกไว้ในฐานข้อมูลของเครื่องมือ เมื่อผู้ใช้ต้องการสร้างข้อมูลทดสอบ ผู้ใช้ต้องเลือกตารางข้อมูลที่จะสร้างข้อมูลทดสอบ และกำหนดจำนวนระเบียบข้อมูลทดสอบที่ต้องการสร้างให้กับตารางข้อมูลเหล่านั้น จากนั้นจึงกำหนดวิธีการสร้างข้อมูลให้กับแต่ละเขตข้อมูลในแต่ละตารางข้อมูลเพื่อใช้อ้างอิงในการสร้างข้อมูลทดสอบให้กับเขตข้อมูลนั้น โดยวิธีการสร้างข้อมูลจะมีอยู่ 4 วิธีคือ

1. เครื่องมือสุ่มสร้างให้เองโดยอัตโนมัติ ทุกๆ เขตข้อมูลจะถูกกำหนดให้ใช้วิธีนี้เป็นค่าเริ่มต้น (Default)
2. ค่าคงที่ (Constant value) คือกำหนดให้ทุกระเบียนข้อมูลของเขตข้อมูลนี้มีค่าเท่ากับค่าที่กำหนดเหมือนกันทั้งหมด
3. ค่าว่าง (Null value) คือกำหนดให้ไม่ต้องมีการสร้างข้อมูลให้กับเขตข้อมูลนี้ โดยจะใช้ได้กับเขตข้อมูลที่มีคุณสมบัติเป็นตัวเลือกเท่านั้น

4. ค่าทั่วไป (Generic value) คือกำหนดให้ใช้กลุ่มค่าทั่วไปซึ่งเป็นกลุ่มของค่าข้อมูลทั่วไปที่เครื่องมือได้จัดเตรียมไว้ให้ เช่น ชื่อ นามสกุล เพศ ชื่อประเทศ และสี เป็นต้น ในการสร้างข้อมูลให้กับเขตข้อมูล โดยเครื่องมือจะสุ่มค่าขึ้นมาจากกลุ่มค่าทั่วไปที่เลือกไว้

หลังจากนั้นผู้ใช้ต้องเลือกรูปแบบการบันทึกข้อมูลทดสอบที่ต้องการ โดยรูปแบบการบันทึกข้อมูลทดสอบจะมีอยู่ 3 แบบคือ

- บันทึกข้อมูลทดสอบเป็นแฟ้มข้อความที่มีตัวคั่น โดยผู้ใช้สามารถกำหนดตัวคั่นระหว่างเขตข้อมูลและระเบียบข้อมูลได้ และผู้ใช้ต้องกำหนดที่อยู่ของแฟ้มข้อความที่จะบันทึกข้อมูลทดสอบด้วย

- บันทึกข้อมูลทดสอบเป็นแฟ้มเอกสารเอ็กซ์เอ็มแอล โดยผู้ใช้ต้องกำหนดที่อยู่ของแฟ้มเอกสารเอ็กซ์เอ็มแอลที่จะบันทึกข้อมูลทดสอบด้วย

- บันทึกข้อมูลทดสอบลงในฐานข้อมูลเป้าหมาย โดยเครื่องมือจะบันทึกข้อมูลทดสอบที่สร้างได้ลงในฐานข้อมูลเป้าหมายที่กำลังเชื่อมต่ออยู่

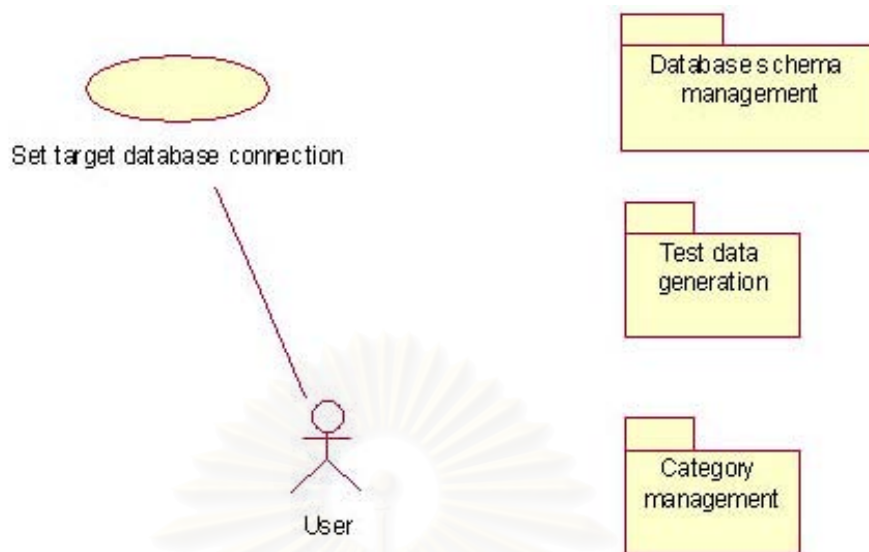
เมื่อกำหนดรายละเอียดในการสร้างข้อมูลทดสอบเสร็จแล้ว เครื่องมือจะสร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลและรายละเอียดที่ผู้ใช้กำหนดมา

3.2 การวิเคราะห์และออกแบบเครื่องมือ

สำหรับการวิเคราะห์และออกแบบเครื่องมือนั้นจะอธิบายด้วยแผนภาพต่างๆ ในภาษายูเอ็มแอล (UML: Unified Modeling Language) ซึ่งเป็นสัญลักษณ์มาตรฐานที่ใช้ในการวิเคราะห์และออกแบบเชิงวัตถุ (Object Oriented Analysis and Design) โดยแผนภาพที่เลือกใช้ ได้แก่ แผนภาพยูสเคส (Use case diagram) แผนภาพคลาส (Class diagram) แผนภาพซีควেনซ์ (Sequence diagram) แผนภาพคอลลาบอเรชัน (Collaboration diagram) และแผนภาพลำดับกิจกรรม (Activity diagram) ดังนี้

3.2.1 แผนภาพยูสเคส

แผนภาพยูสเคสจะแสดงว่าผู้ใช้ระบบสามารถทำอะไรได้บ้าง ในที่นี้ผู้ใช้หมายถึงผู้ทดสอบและระบบหมายถึงเครื่องมือที่พัฒนาขึ้นมา โดยแผนภาพยูสเคสโดยรวมของเครื่องมือเป็นดังนี้



รูปที่ 3.2 แผนภาพยูสเคสโดยรวม

จากรูปที่ 3.2 แผนภาพยูสเคสโดยรวมจะประกอบด้วยแพ็คเกจ (Package) 3 แพ็คเกจคือ แพ็คเกจการจัดการสคีมาของฐานข้อมูล (Database schema management) แพ็คเกจการสร้างข้อมูลทดสอบ (Test data generation) และแพ็คเกจการจัดการกลุ่มค่าทั่วไป (Category management) นอกจากนี้ในแผนภาพยังประกอบด้วยยูสเคสอีก 1 ยูสเคสคือ ยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย ซึ่งหมายความว่าผู้ใช้สามารถกำหนดให้เครื่องมือทำการเชื่อมต่อกับฐานข้อมูลเป้าหมายที่ต้องการได้ รายละเอียดของยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมายจะเป็นดังตารางที่ 3.1

ตารางที่ 3.1 รายละเอียดของยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

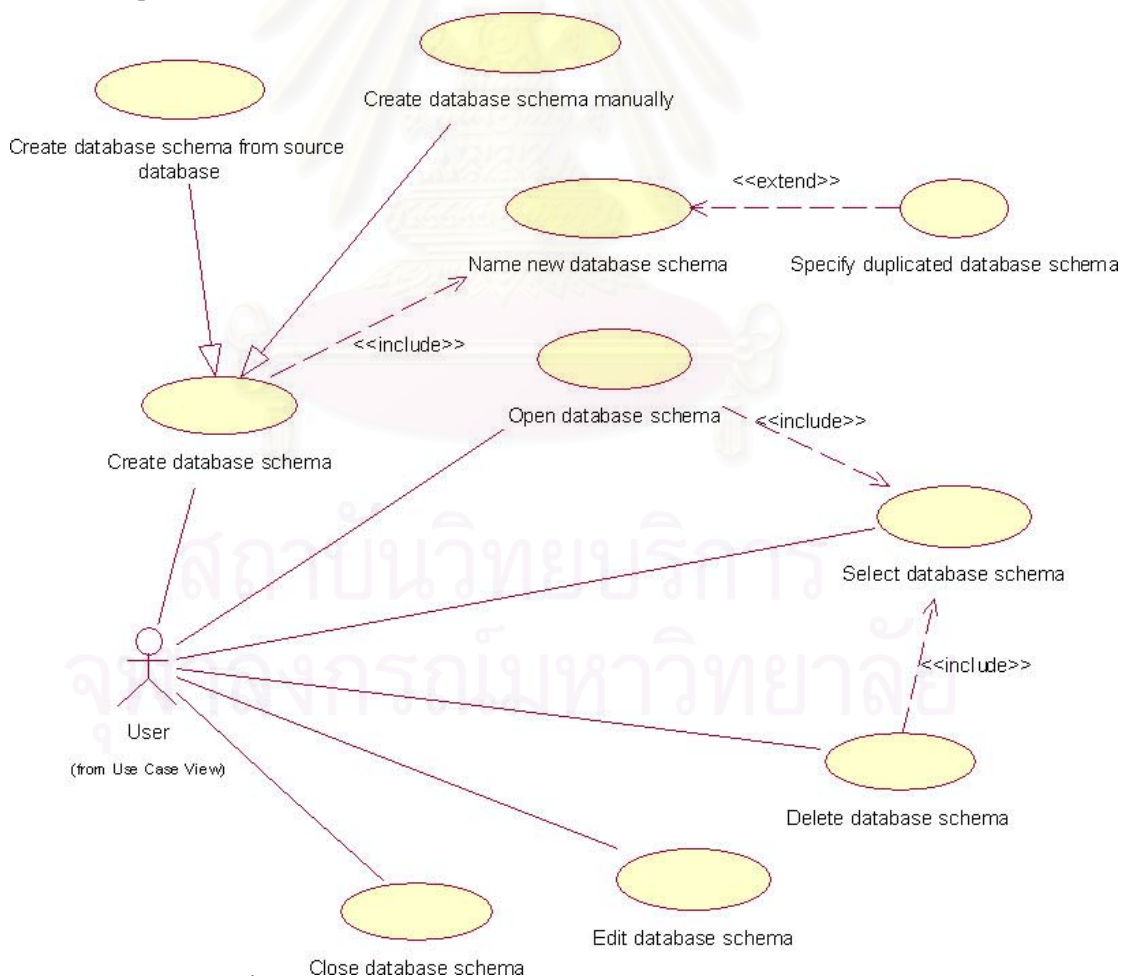
Use case:	Set target database connection
Actors:	User
Goal:	เพื่อกำหนดให้เครื่องมือเชื่อมต่อกับฐานข้อมูลเป้าหมายที่ต้องการจะบันทึกข้อมูลทดสอบหลังจากที่สร้างข้อมูลทดสอบเสร็จแล้ว
Related use cases:	-
Preconditions:	ฐานข้อมูลเป้าหมายที่ต้องการเชื่อมต่อนั้นต้องเปิดให้บริการอยู่
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกกำหนดรายละเอียดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย 2. เครื่องมือแสดงหน้าต่างรับข้อมูลรายละเอียดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

ตารางที่ 3.1 รายละเอียดของยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย (ต่อ)

	3. ผู้ใช้เลือกชนิดของฐานข้อมูลที่ต้องการเชื่อมต่อ 4. ผู้ใช้กำหนดพารามิเตอร์ที่ใช้ในการเชื่อมต่อ 5. ผู้ใช้ยืนยันข้อมูลที่ใช้ในการเชื่อมต่อ 6. เครื่องมือตอบรับการเชื่อมต่อกับฐานข้อมูลเป้าหมาย
Postconditions:	เครื่องมือทำการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

สำหรับแพ็คเกจทั้ง 3 แพ็คเกจที่ปรากฏในแผนภาพยูสเคสนั้นเป็นแพ็คเกจที่รวมเอายูสเคสที่มีความเกี่ยวข้องในกลุ่มงานเดียวกันไว้ด้วยกัน โดยรายละเอียดภายในแต่ละแพ็คเกจนั้นเป็นดังนี้

1. แพ็คเกจการจัดการสคีมาของฐานข้อมูล รายละเอียดของแพ็คเกจนี้สามารถแสดงได้ดังรูปที่ 3.3



รูปที่ 3.3 ยูสเคสภายในแพ็คเกจการจัดการสคีมาของฐานข้อมูล

จากรูปที่ 3.3 แพ็คเคจการจัดการสคีมาของฐานข้อมูลจะประกอบด้วยยูสเคสทั้งหมด 10 ยูสเคส ซึ่งแต่ละยูสเคสจะเกี่ยวข้องกับการจัดการสคีมาของฐานข้อมูลทั้งหมด โดยรายละเอียดของแต่ละยูสเคสจะเป็นดังตารางที่ 3.2 ถึงตารางที่ 3.11

ตารางที่ 3.2 รายละเอียดของยูสเคสกำหนดชื่อสคีมาของฐานข้อมูล

Use case:	Name new database schema
Actors:	User
Goal:	เพื่อกำหนดชื่อและคำอธิบายให้กับสคีมาของฐานข้อมูลใหม่ที่จะสร้าง
Related use cases:	-
Preconditions:	ผู้ใช้เลือกสร้างสคีมาของฐานข้อมูล
Steps:	<ol style="list-style-type: none"> 1. เครื่องมือแสดงหน้าต่างเพื่อรับข้อมูลชื่อและคำอธิบายสคีมาของฐานข้อมูล 2. ผู้ใช้กรอกชื่อและคำอธิบายของสคีมาของฐานข้อมูลที่ต้องการสร้าง 3. ผู้ใช้ยืนยันการกำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูล 4. เครื่องมือตอบรับการกำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูล
Postconditions:	-

ตารางที่ 3.3 รายละเอียดของยูสเคสสคีมาของฐานข้อมูลซ้ำซ้อน

Use case:	Specify duplicated database schema
Actors:	User
Goal:	เพื่อไม่ให้มีการสร้างสคีมาของฐานข้อมูลที่ซ้ำซ้อนกัน
Related use cases:	Extensions of: Name new database schema
Preconditions:	ผู้ที่ยืนยันการกำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูล
Steps:	<ol style="list-style-type: none"> 1. เครื่องมือแจ้งว่าชื่อของสคีมาของฐานข้อมูลที่ผู้ใช้ต้องการสร้างนั้นซ้ำซ้อนกับชื่อของสคีมาของฐานข้อมูลที่มีอยู่ในฐานข้อมูลของเครื่องมือ 2. เครื่องมือปฏิเสธชื่อสคีมาของฐานข้อมูล
Postconditions:	-

ตารางที่ 3.4 รายละเอียดของยูสเคสสร้างสคีมาของฐานข้อมูล

Use case:	Create database schema
Actors:	User
Goal:	เพื่อสร้างสคีมาของฐานข้อมูลใหม่เก็บไว้ในฐานข้อมูลของเครื่องมือ สำหรับการสร้างข้อมูลทดสอบต่อไป
Related use cases:	Generalization of: <ul style="list-style-type: none"> ❑ Create database schema manually ❑ Create database schema from source database Includes: Name new database schema
Preconditions:	-
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกสร้างสคีมาของฐานข้อมูล 2. ผู้ใช้กำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูล (Use case: Name new database schema) 3. ผู้ใช้กำหนดรายละเอียดของสคีมาของฐานข้อมูล 4. เครื่องมือตอบรับการสร้างสคีมาของฐานข้อมูล
Postconditions:	เครื่องมือสร้างสคีมาของฐานข้อมูลใหม่ตามรายละเอียดที่ผู้ใช้กำหนดลงในฐานข้อมูลของเครื่องมือ

ตารางที่ 3.5 รายละเอียดของยูสเคสสร้างสคีมาของฐานข้อมูลด้วยมือ

Use case:	Create database schema manually
Actors:	User
Goal:	เพื่อสร้างสคีมาของฐานข้อมูลใหม่เก็บไว้ในฐานข้อมูลของเครื่องมือ โดยผู้ใช้เป็นผู้ออกรายละเอียดของสคีมาของฐานข้อมูลเองทั้งหมด
Related use cases:	Specialization of: Create database schema
Preconditions:	-
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกสร้างสคีมาของฐานข้อมูลด้วยมือ 2. ผู้ใช้กำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูล 3. เครื่องมือแสดงหน้าต่างเพื่อรับรายละเอียดของสคีมาของฐานข้อมูล 4. ผู้ใช้กรอกชื่อตารางข้อมูลที่มีในสคีมาของฐานข้อมูล

ตารางที่ 3.5 รายละเอียดของยูสเคสสร้างสคีมาของฐานข้อมูลด้วยมือ (ต่อ)

	<ol style="list-style-type: none"> 5. ผู้ใช้กำหนดเขตข้อมูลของแต่ละตารางในสคีมาของฐานข้อมูล 6. ผู้ใช้กำหนดความคงสภาพในการอ้างอิงระหว่างตารางที่ต้องการ 7. ผู้ใช้กำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลที่ต้องการภายในแต่ละตาราง 8. ผู้ใช้ยืนยันการบันทึกข้อมูลรายละเอียดของสคีมาของฐานข้อมูล 9. เครื่องมือตอบรับการสร้างสคีมาของฐานข้อมูล
Postconditions:	เครื่องมือสร้างสคีมาของฐานข้อมูลใหม่ตามรายละเอียดที่ผู้ใช้กำหนดลงในฐานข้อมูลของเครื่องมือ

ตารางที่ 3.6 รายละเอียดของยูสเคสสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง

Use case:	Create database schema from source database
Actors:	User
Goal:	เพื่อสร้างสคีมาของฐานข้อมูลใหม่เก็บไว้ในฐานข้อมูลของเครื่องมือ โดยอ่านมาจากฐานข้อมูลต้นทางผ่านทางทการเชื่อมต่อ
Related use cases:	Specialization of: Create database schema
Preconditions:	ฐานข้อมูลต้นทางที่ต้องการเชื่อมต่อนั้นต้องเปิดให้บริการอยู่
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง 2. ผู้ใช้กำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูล 3. เครื่องมือแสดงหน้าต่างรับข้อมูลรายละเอียดการเชื่อมต่อกับฐานข้อมูลต้นทาง 4. ผู้ใช้เลือกชนิดของฐานข้อมูลที่ต้องการเชื่อมต่อ 5. ผู้ใช้กำหนดพารามิเตอร์ (Parameters) ที่ใช้ในการเชื่อมต่อ 6. ผู้ใช้ยืนยันข้อมูลที่ใช้ในการเชื่อมต่อ 7. เครื่องมือเชื่อมต่อกับฐานข้อมูลต้นทาง 8. เครื่องมืออ่านรายละเอียดของสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทางและแสดงผลการอ่านข้อมูล 9. เครื่องมือตอบรับการสร้างสคีมาของฐานข้อมูล

ตารางที่ 3.6 รายละเอียดของยูสเคสสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง (ต่อ)

Postconditions:	เครื่องมือสร้างสคีมาของฐานข้อมูลใหม่ตามรายละเอียดที่ได้จากฐานข้อมูลต้นทางซึ่งจะมีเพียงรายการตารางข้อมูล รายการเขตข้อมูลในแต่ละตาราง และความคงสภาพในการอ้างอิงระหว่างตารางเท่านั้น
-----------------	---

ตารางที่ 3.7 รายละเอียดของยูสเคสเปิดสคีมาของฐานข้อมูล

Use case:	Open database schema
Actors:	User
Goal:	เพื่อเปิดใช้สคีมาของฐานข้อมูลที่เคยสร้างไว้ในฐานข้อมูลของเครื่องมือแล้ว
Related use cases:	Includes: Select database schema
Preconditions:	-
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกเปิดสคีมาของฐานข้อมูล 2. ผู้ใช้เลือกสคีมาของฐานข้อมูลที่ต้องการ (Use case: Select database schema) 3. เครื่องมือตอบรับการเปิดสคีมาของฐานข้อมูลที่ผู้ใช้เลือกมา
Postconditions:	เครื่องมือทำการเปิดสคีมาของฐานข้อมูลที่ผู้ใช้เลือกมา

ตารางที่ 3.8 รายละเอียดของยูสเคสลบสคีมาของฐานข้อมูล

Use case:	Delete database schema
Actors:	User
Goal:	เพื่อลบสคีมาของฐานข้อมูลที่เคยสร้างไว้ออกจากฐานข้อมูลของเครื่องมือ
Related use cases:	Includes: Select database schema
Preconditions:	สคีมาของฐานข้อมูลที่ต้องการลบจะต้องไม่ถูกเปิดใช้อยู่
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกลบสคีมาของฐานข้อมูล 2. ผู้ใช้เลือกสคีมาของฐานข้อมูลที่ต้องการ (Use case: Select database schema) 3. เครื่องมือตอบรับการลบสคีมาของฐานข้อมูลที่ผู้ใช้เลือกมา

ตารางที่ 3.8 รายละเอียดของยูสเคสลบสคีมาของฐานข้อมูล (ต่อ)

Postconditions:	เครื่องมือทำการลบสคีมาของฐานข้อมูลที่ผู้ใช้เลือกมาออกจากฐานข้อมูลของเครื่องมือ
-----------------	--

ตารางที่ 3.9 รายละเอียดของยูสเคสเลือกสคีมาของฐานข้อมูล

Use case:	Select database schema
Actors:	User
Goal:	เพื่อเลือกสคีมาของฐานข้อมูลที่เคยสร้างไว้ในการเปิดหรือลบสคีมาของฐานข้อมูล
Related use cases:	-
Preconditions:	ผู้ใช้เลือกเปิดสคีมาของฐานข้อมูล หรือ ผู้ใช้เลือกลบสคีมาของฐานข้อมูล
Steps:	<ol style="list-style-type: none"> 1. เครื่องมือแสดงหน้าต่างที่แสดงรายชื่อสคีมาของฐานข้อมูลที่มีอยู่ในฐานข้อมูลของเครื่องมือ 2. ผู้ใช้เลือกสคีมาของฐานข้อมูลที่ต้องการ 3. ผู้ใช้ยืนยันการเลือกสคีมาของฐานข้อมูล
Postconditions:	เครื่องมือทำการเลือกสคีมาของฐานข้อมูลที่ผู้ใช้ต้องการ

ตารางที่ 3.10 รายละเอียดของยูสเคสแก้ไขสคีมาของฐานข้อมูล

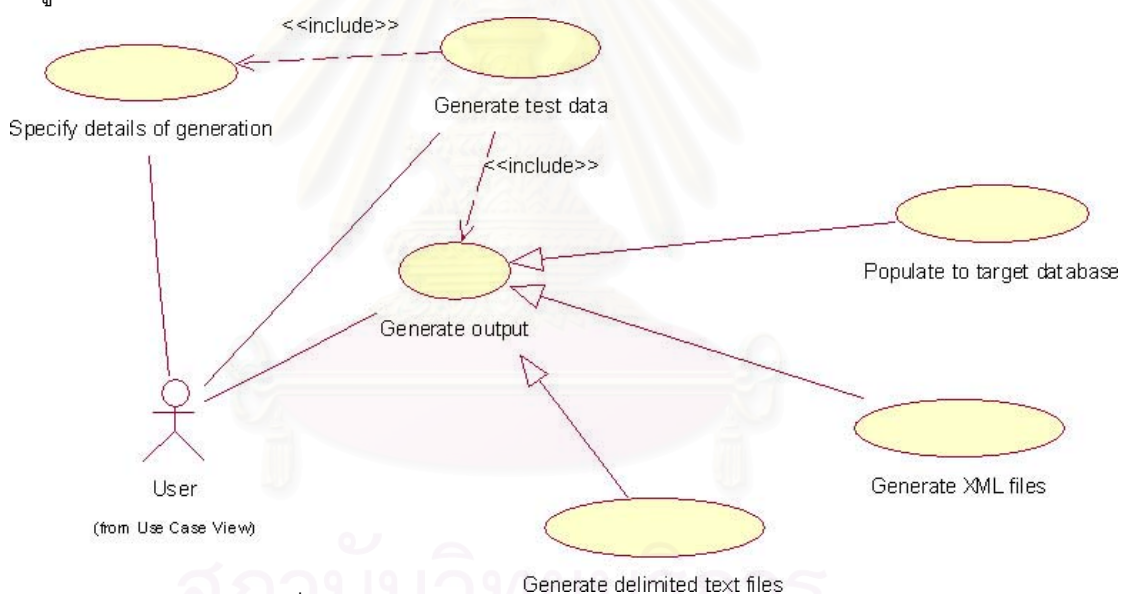
Use case:	Edit database schema
Actors:	User
Goal:	เพื่อแก้ไขสคีมาของฐานข้อมูลที่กำลังเปิดใช้อยู่
Related use cases:	-
Preconditions:	ผู้ใช้ออกเปิดสคีมาของฐานข้อมูลที่ต้องการแก้ไขไว้แล้ว
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกแก้ไขสคีมาของฐานข้อมูล 2. เครื่องมือแสดงหน้าต่างจัดการข้อมูลภายในสคีมาของฐานข้อมูล 3. ผู้ใช้แก้ไขข้อมูลที่ต้องการภายในสคีมาของฐานข้อมูล 4. ผู้ใช้ยืนยันการแก้ไขสคีมาของฐานข้อมูล
Postconditions:	เครื่องมือทำการแก้ไขข้อมูลของสคีมาของฐานข้อมูลตามที่ผู้ใช้กำหนด

ตารางที่ 3.11 รายละเอียดของยูสเคสปิดสคีมามาของฐานข้อมูล

Use case:	Close database schema
Actors:	User
Goal:	เพื่อปิดสคีมามาของฐานข้อมูลที่กำลังเปิดใช้อยู่
Related use cases:	-
Preconditions:	ผู้ใช้งานต้องเปิดสคีมามาของฐานข้อมูลที่จะปิดไว้แล้ว
Steps:	1. ผู้ใช้เลือกปิดสคีมามาของฐานข้อมูล 2. เครื่องมือตอบรับการปิดสคีมามาของฐานข้อมูล
Postconditions:	เครื่องมือทำการปิดสคีมามาของฐานข้อมูลที่กำลังเปิดไว้

2. แพ็คเกจการสร้างข้อมูลทดสอบ รายละเอียดของแพ็คเกจนี้สามารถแสดงได้

ดังรูปที่ 3.4



รูปที่ 3.4 ยูสเคสภายในแพ็คเกจการสร้างข้อมูลทดสอบ

จากรูปที่ 3.4 แพ็คเกจการสร้างข้อมูลทดสอบจะประกอบด้วยยูสเคสทั้งหมด 6 ยูสเคส ซึ่งแต่ละยูสเคสจะเกี่ยวข้องกับการสร้างข้อมูลทดสอบทั้งหมด โดยรายละเอียดของแต่ละยูสเคสจะเป็นดังตารางที่ 3.12 ถึงตารางที่ 3.17

ตารางที่ 3.12 รายละเอียดของยูสเคสสร้างข้อมูลทดสอบ

Use case:	Generate test data
Actors:	User

ตารางที่ 3.12 รายละเอียดของยูสเคสสร้างข้อมูลทดสอบ (ต่อ)

Goal:	เพื่อสร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลที่กำลังเปิดให้อยู่
Related use cases:	Includes: <ul style="list-style-type: none"> ■ Specify details of generation ■ Generate output
Preconditions:	ผู้ใช้งานต้องเปิดสคีมาของฐานข้อมูลที่ต้องการสร้างข้อมูลทดสอบไว้แล้ว
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกสร้างข้อมูลทดสอบ 2. ผู้ใช้กำหนดรายละเอียดของการสร้างข้อมูล (Use case: Specify details of generation) 3. ผู้ใช้ยืนยันการสร้างข้อมูลทดสอบ 4. เครื่องมือสร้างข้อมูลทดสอบ 5. เครื่องมือบันทึกข้อมูลทดสอบตามรูปแบบที่ผู้ใช้กำหนดมา (Use case: Generate output) 6. เครื่องมือแสดงผลการสร้างข้อมูลทดสอบ
Postconditions:	เครื่องมือทำการสร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลและรายละเอียดที่ผู้ใช้กำหนดมา

ตารางที่ 3.13 รายละเอียดของยูสเคสกำหนดรายละเอียดสำหรับการสร้างข้อมูลทดสอบ

Use case:	Specify details of generation
Actors:	User
Goal:	เพื่อกำหนดรายละเอียดของการสร้างข้อมูลทดสอบ
Related use cases:	-
Preconditions:	ผู้ใช้งานต้องเลือกสร้างข้อมูลทดสอบ
Steps:	<ol style="list-style-type: none"> 1. เครื่องมือแสดงหน้าต่างรับรายละเอียดการสร้างข้อมูลทดสอบ 2. ผู้ใช้เลือกตารางข้อมูลที่ต้องการสร้างข้อมูลทดสอบ 3. ผู้ใช้กำหนดชนิดของฐานข้อมูลเป้าหมายที่จะนำข้อมูลทดสอบไปใช้ 4. ผู้ใช้กำหนดจำนวนระเบียบข้อมูลที่ต้องการสร้างให้กับแต่ละตาราง

ตารางที่ 3.13 รายละเอียดของยูสเคสกำหนดรายละเอียดสำหรับการสร้างข้อมูลทดสอบ (ต่อ)

	5. ผู้ใช้กำหนดวิธีการสร้างข้อมูลให้กับแต่ละเขตข้อมูลของแต่ละตาราง 6. ผู้ใช้กำหนดรูปแบบการบันทึกข้อมูลทดสอบ 7. ผู้ใช้ยืนยันรายละเอียดของการสร้างข้อมูลทดสอบ
Postconditions:	เครื่องมือรับรายละเอียดของการสร้างข้อมูลทดสอบ

ตารางที่ 3.14 รายละเอียดของยูสเคสสร้างผลลัพธ์

Use case:	Generate output
Actors:	User
Goal:	เพื่อบันทึกข้อมูลทดสอบที่สร้างเสร็จแล้วตามรูปแบบที่ผู้ใช้กำหนดมา
Related use cases:	Generalization of: <ul style="list-style-type: none"> ■ Generate delimited text files ■ Generate XML files ■ Update to target database
Preconditions:	เครื่องมือสร้างข้อมูลทดสอบเสร็จแล้ว
Steps:	1. เครื่องมือบันทึกข้อมูลทดสอบตามรูปแบบที่ผู้ใช้กำหนดมา
Postconditions:	-

ตารางที่ 3.15 รายละเอียดของยูสเคสสร้างเพิ่มข้อความที่มีตัวค้น

Use case:	Generate delimited text files
Actors:	User
Goal:	เพื่อบันทึกข้อมูลทดสอบเป็นเพิ่มข้อความที่มีตัวค้น
Related use cases:	Specialization of: Generate output
Preconditions:	เครื่องมือสร้างข้อมูลทดสอบเสร็จแล้ว
Steps:	1. เครื่องมือบันทึกข้อมูลทดสอบเป็นเพิ่มข้อความที่มีตัวค้นตามที่ผู้ใช้กำหนด โดยบันทึกเป็นตารางละ 1 เพิ่ม
Postconditions:	-

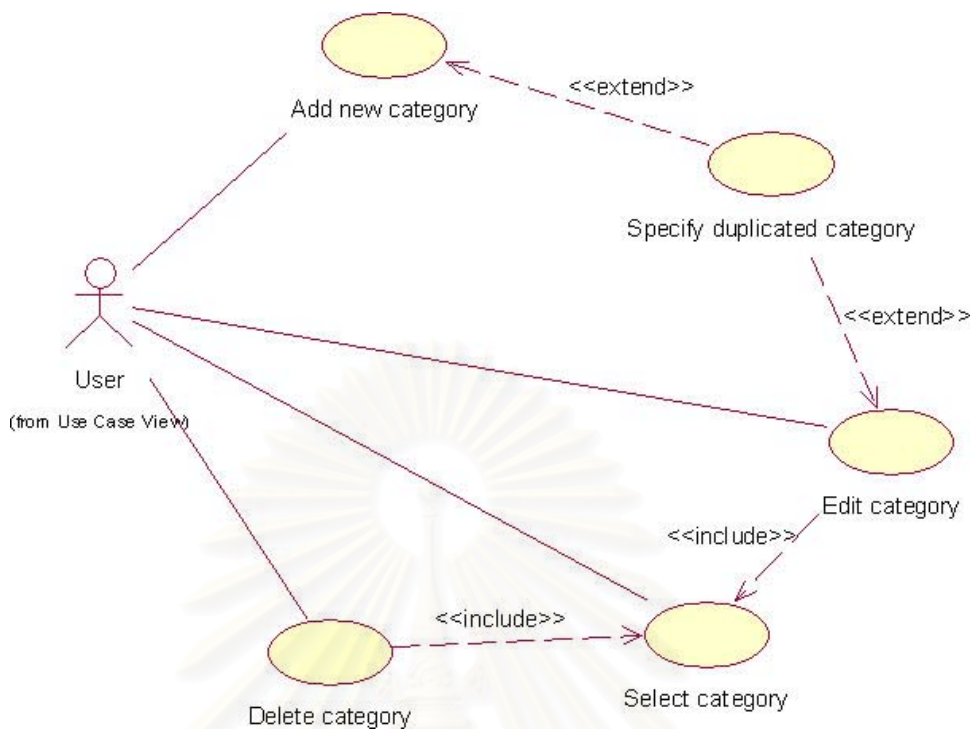
ตารางที่ 3.16 รายละเอียดของยูสเคสสร้างแฟ้มเอกสารอิเล็กทรอนิกส์เอ็มแอล

Use case:	Generate XML files
Actors:	User
Goal:	เพื่อบันทึกข้อมูลทดสอบเป็นแฟ้มเอกสารอิเล็กทรอนิกส์เอ็มแอล
Related use cases:	Specialization of: Generate output
Preconditions:	เครื่องมือสร้างข้อมูลทดสอบเสร็จแล้ว
Steps:	1. เครื่องมือบันทึกข้อมูลทดสอบเป็นแฟ้มเอกสารอิเล็กทรอนิกส์เอ็มแอลตารางละ 1 แฟ้ม โดยรูปแบบของเอกสารจะเป็นไปตามชนิดของฐานข้อมูลเป้าหมายที่ผู้ใช้กำหนด
Postconditions:	-

ตารางที่ 3.17 รายละเอียดของยูสเคสเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมาย

Use case:	Populate to target database
Actors:	User
Goal:	เพื่อบันทึกข้อมูลทดสอบลงในฐานข้อมูลเป้าหมายที่มีการเชื่อมต่ออยู่
Related use cases:	Specialization of: Generate output
Preconditions:	1. เครื่องมือสร้างข้อมูลทดสอบเสร็จแล้ว 2. ผู้ใช้ทำการเชื่อมต่อกับฐานข้อมูลปลายทางไว้แล้ว
Steps:	1. เครื่องมือบันทึกข้อมูลทดสอบลงในฐานข้อมูลเป้าหมายที่กำลังเชื่อมต่ออยู่
Postconditions:	-

3. แพ็คเกจการจัดการกลุ่มค่าทั่วไป รายละเอียดของแพ็คเกจนี้สามารถแสดงได้ดังรูปที่ 3.5 แพ็คเกจการจัดการกลุ่มค่าทั่วไปจะประกอบด้วยยูสเคสทั้งหมด 5 ยูสเคส ซึ่งแต่ละยูสเคสจะเกี่ยวข้องกับการจัดการกลุ่มค่าทั่วไปทั้งหมด โดยรายละเอียดของแต่ละยูสเคสจะเป็นดังตารางที่ 3.18 ถึงตารางที่ 3.22



รูปที่ 3.5 ยูสเคสภายในแพ็คเกจการจัดการกลุ่มค่าทั่วไป

ตารางที่ 3.18 รายละเอียดของยูสเคสเพิ่มกลุ่มค่าทั่วไปใหม่

Use case:	Add new category
Actors:	User
Goal:	เพื่อสร้างกลุ่มค่าทั่วไปกลุ่มใหม่ที่ผู้ใช้ต้องการลงในฐานข้อมูลของเครื่องมือ
Related use cases:	-
Preconditions:	-
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกเพิ่มกลุ่มค่าทั่วไป 2. เครื่องมือแสดงหน้าต่างจัดการข้อมูลของกลุ่มค่าทั่วไป 3. ผู้ใช้กำหนดชื่อและชนิดข้อมูลของกลุ่มค่าทั่วไป 4. ผู้ใช้เพิ่มค่าข้อมูลให้กับกลุ่มค่าทั่วไป 5. ผู้ใช้ยืนยันการสร้างกลุ่มค่าทั่วไป 6. เครื่องมือตอบรับการสร้างกลุ่มค่าทั่วไป
Postconditions:	เครื่องมือทำการสร้างกลุ่มค่าทั่วไปกลุ่มใหม่และเพิ่มเข้าสู่รายการกลุ่มค่าทั่วไปของเครื่องมือ

ตารางที่ 3.19 รายละเอียดของยูสเคสแก้ไขกลุ่มค่าทั่วไป

Use case:	Edit category
Actors:	User
Goal:	เพื่อแก้ไขข้อมูลของกลุ่มค่าทั่วไปที่ผู้ใช้เลือกมา
Related use cases:	Includes: Select category
Preconditions:	-
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกแก้ไขกลุ่มค่าทั่วไป 2. ผู้ใช้เลือกกลุ่มค่าทั่วไปที่ต้องการแก้ไข (Use case: Select category) 3. เครื่องมือแสดงหน้าต่างจัดการข้อมูลของกลุ่มค่าทั่วไป 4. ผู้ใช้แก้ไขข้อมูลของกลุ่มค่าทั่วไป 5. ผู้ใช้ยืนยันการแก้ไขกลุ่มค่าทั่วไป 6. เครื่องมือตอบรับการแก้ไขกลุ่มค่าทั่วไป
Postconditions:	เครื่องมือทำการแก้ไขข้อมูลของกลุ่มค่าทั่วไปที่ผู้ใช้กำหนดมา

ตารางที่ 3.20 รายละเอียดของยูสเคสลบกลุ่มค่าทั่วไป

Use case:	Delete category
Actors:	User
Goal:	เพื่อลบกลุ่มค่าทั่วไปที่ผู้ใช้เลือกมา
Related use cases:	Includes: Select category
Preconditions:	-
Steps:	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกลบกลุ่มค่าทั่วไป 2. ผู้ใช้เลือกกลุ่มค่าทั่วไปที่ต้องการลบ (Use case: Select category) 3. ผู้ใช้ยืนยันการลบกลุ่มค่าทั่วไป 4. เครื่องมือตอบรับการลบกลุ่มค่าทั่วไป
Postconditions:	เครื่องมือทำการลบกลุ่มค่าทั่วไปออกจากฐานข้อมูลของเครื่องมือ

ตารางที่ 3.21 รายละเอียดของยูสเคสเลือกกลุ่มค่าทั่วไป

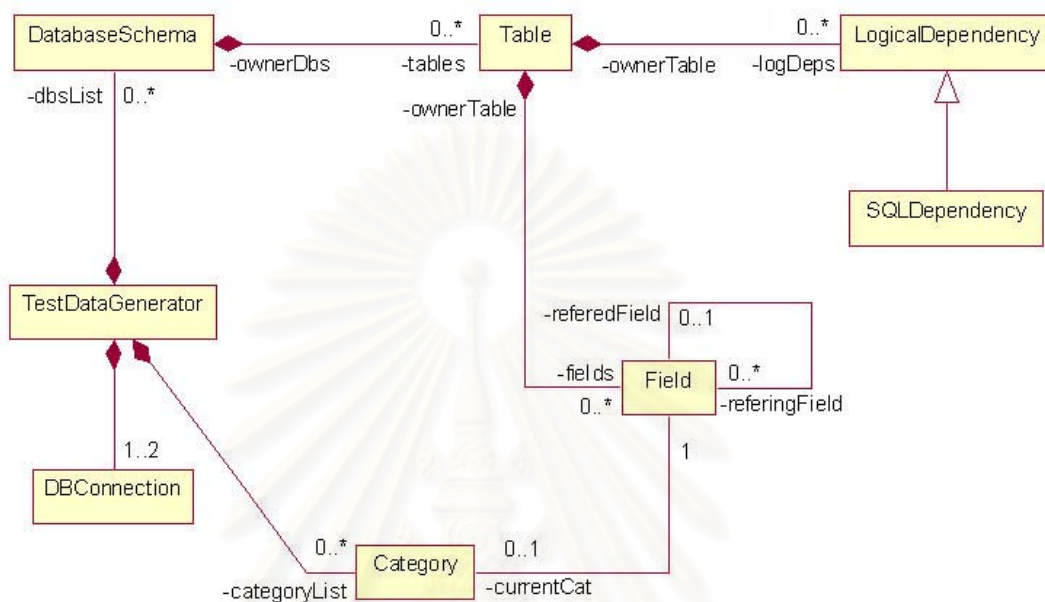
Use case:	Select category
Actors:	User
Goal:	เพื่อเลือกกลุ่มค่าทั่วไปในการแก้ไขหรือลบกลุ่มค่าทั่วไป
Related use cases:	-
Preconditions:	ผู้ใช้เลือกแก้ไขกลุ่มค่าทั่วไป หรือ ผู้ใช้เลือกลบกลุ่มค่าทั่วไป
Steps:	<ol style="list-style-type: none"> 1. เครื่องมือแสดงหน้าต่างที่แสดงรายชื่อกลุ่มค่าทั่วไปที่มีอยู่ในฐานข้อมูลของเครื่องมือ 2. ผู้ใช้เลือกกลุ่มค่าทั่วไปที่ต้องการ 3. ผู้ใช้ยืนยันการเลือกกลุ่มค่าทั่วไป
Postconditions:	เครื่องมือทำการเลือกกลุ่มค่าทั่วไปที่ผู้ใช้ต้องการ

ตารางที่ 3.22 รายละเอียดของยูสเคสกลุ่มค่าทั่วไปซ้ำซ้อน

Use case:	Specify duplicated category
Actors:	User
Goal:	เพื่อไม่ให้มีการกำหนดชื่อของกลุ่มค่าทั่วไปที่ซ้ำซ้อนกับชื่อของกลุ่มค่าทั่วไปที่มีอยู่
Related use cases:	Extensions of: <ul style="list-style-type: none"> <input type="checkbox"/> Add new category <input type="checkbox"/> Edit category
Preconditions:	ผู้ใช้ยืนยันการสร้างกลุ่มค่าทั่วไป หรือ ผู้ใช้ยืนยันการแก้ไขกลุ่มค่าทั่วไป
Steps:	<ol style="list-style-type: none"> 1. เครื่องมือแจ้งว่าชื่อของกลุ่มค่าทั่วไปนั้นซ้ำซ้อนกับชื่อของกลุ่มค่าทั่วไปที่มีอยู่ในฐานข้อมูลของเครื่องมือ 2. เครื่องมือปฏิเสธชื่อของกลุ่มค่าทั่วไป
Postconditions:	-

3.2.2 แผนภาพคลาส

แผนภาพคลาสจะแสดงถึงคลาสต่างๆ ที่มีอยู่ในระบบ และความสัมพันธ์ระหว่างคลาสเหล่านี้ สำหรับแผนภาพคลาสของเครื่องมือสร้างข้อมูลทดสอบจะเป็นดังรูปที่ 3.6



รูปที่ 3.6 แผนภาพคลาสของเครื่องมือสร้างข้อมูลทดสอบ

จากรูปที่ 3.6 เครื่องมือสร้างข้อมูลทดสอบจะประกอบด้วยคลาสทั้งหมด 8 คลาสที่สำคัญ โดยรายละเอียดของคลาสแต่ละคลาสเป็นดังนี้

1. คลาส TestDataGenerator คือคลาสหลักที่เป็นตัวแทนของเครื่องมือในการติดต่อกับผู้ใช้ จะทำหน้าที่รับคำสั่งทุกอย่างจากผู้ใช้เพื่อนำมาประมวลผลและส่งผลลัพธ์กลับไปยังผู้ใช้ รายละเอียดของคลาส TestDataGenerator จะเป็นดังรูปที่ 3.7

2. คลาส DatabaseSchema คือคลาสที่เป็นตัวแทนของสคีมาของฐานข้อมูลที่มีการใช้งานในเครื่องมือ คลาสนี้จะเก็บข้อมูลของตารางข้อมูลทั้งหมดที่มีอยู่ในแต่ละสคีมาของฐานข้อมูล รายละเอียดของคลาส DatabaseSchema จะเป็นดังรูปที่ 3.8

3. คลาส Table คือคลาสที่เป็นตัวแทนของตารางข้อมูลในแต่ละสคีมาของฐานข้อมูล คลาสนี้จะเก็บข้อมูลของเขตข้อมูลทั้งหมดที่มีอยู่ในตารางและทำหน้าที่หลักในสร้างข้อมูลทดสอบของตารางข้อมูล รายละเอียดของคลาส Table จะเป็นดังรูปที่ 3.9

TestDataGenerator
<ul style="list-style-type: none"> currentDbs : DatabaseSchema targetConnection : DBConnection testDataDbConnection : DBConnection
<ul style="list-style-type: none"> createTextFile() createOracleLoadFile() createXMLFile() selectDataFromTestDataDb() writeDataToTextFile() deriveMySQLXMLFile() deriveOracleXMLFiles() deriveAccessXMLFiles() deriveInsertSQL() insertToTargetDb() <ul style="list-style-type: none"> createDbSchema() checkDuplicateDbs() generateTestData() alertDuplicatedDbs() connectSourceDB() connectTargetDB() openDbs() setCurrentDbs() findDbs() deleteDbs() removeDbsFromList() closeDbs() removeCurrentDbs() createCategory() checkDuplicateCategory() alertDuplicatedCategory() deleteCategory() findCategory() removeCategoryFromList()

รูปที่ 3.7 คลาส TestDataGenerator

DatabaseSchema
<ul style="list-style-type: none"> name : String desc : String
<ul style="list-style-type: none"> create() : DatabaseSchema addTable(name : String) readDbsDetail(sourceConn : DBConnection) setName(dbsName : String) setDesc(dbsDesc : String) getTable(name : String) : Table getName() : String getDesc() : String

รูปที่ 3.8 คลาส DatabaseSchema

Table
<pre> name : String genAmt : Integer depMode : String generated : Boolean getPkFields() : Vector generateTableData(pkFields : Vector, amount : Integer) : Vector validateNormalDep(tableData : Vector) : Vector validateSQLDep(tableData : Vector) : Vector insertToTestDataOf(tableData : Vector) setGenerated/generated : Boolean getPKRecord(pkFields : Vector) : Vector checkExistence(pkRecord : Vector, tableData : Vector) : Boolean insertRecordToTableData : Vector, newRecord : Vector addData(tableData : Vector, fieldIndex : Integer, newData : String) getParentTables() : Vector groupPKFields() : Vector groupFKFields() : Vector getSelfReferencedFields() : Vector getNormalPKFields() : Vector getNormalFields() : Vector possibleToGenerate(groupedPKFields : Vector, normalPKFields : Vector) : Boolean generateFKTableData(amount : Integer, groupedPKFields : Vector, groupedFKFields : Vector, selfRefFields : Vector, normalPKFields : Vector, normalFields : Vector) : Vector getNoOfUniqueRecords(fieldList : Vector) : Integer randomParentRecordOf(fieldList : Vector) : Vector getNoOfRecordsOfTableData : Vector) : Long appendData(dataList : Vector, new dataList : Vector) create() : Table addField(name : String, type : String, size : Integer, scale : Integer, pk : Boolean, optional : Boolean) addLogDepOfField : Field, condValue : String, destField : Field, resultValue : String) setName(name : String) setOwner/Owner : DatabaseSchema) addFieldName : String) : Field setNormalDepMode() setSQLDepMode() addSQLDepOfGenAmt : Integer) setGenAmt/genAmt : Integer) canGenerateTestData() : Boolean generateTestData() generateFKTables() getName() : String getFields() : Vector wasGenerated() : Boolean </pre>

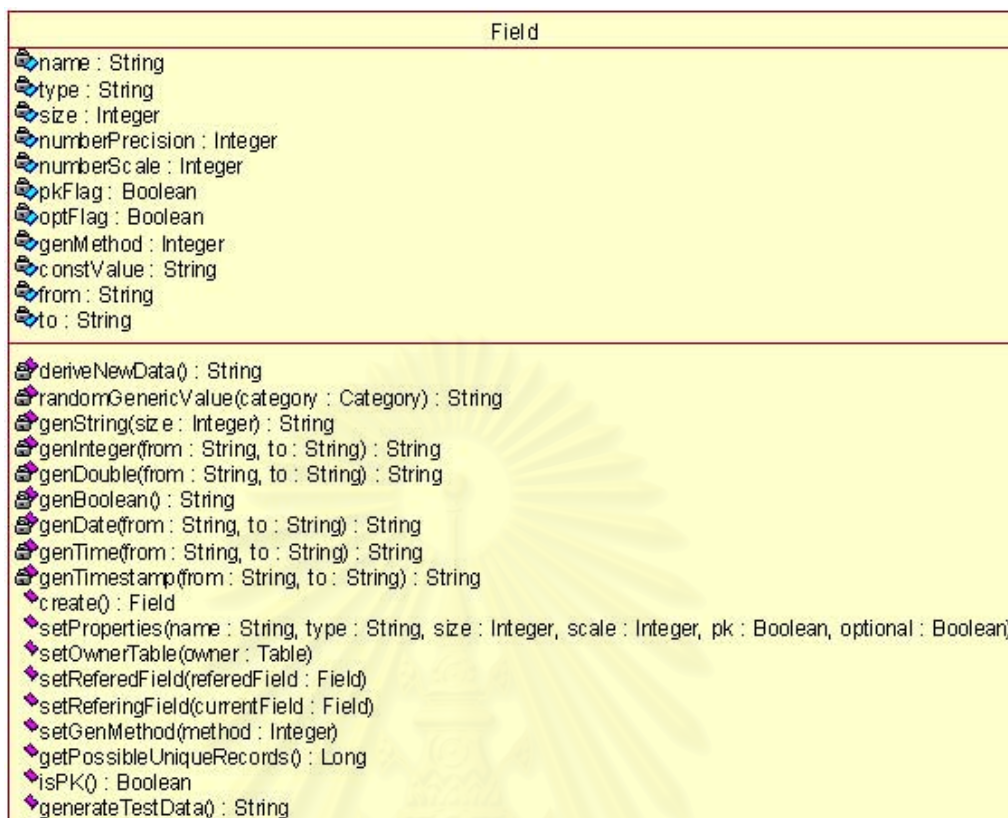
รูปที่ 3.9 คลาส Table

4. คลาส Field คือคลาสที่เป็นตัวแทนของเขตข้อมูลในแต่ละตารางข้อมูล คลาสนี้จะทำหน้าที่หลักในการสุ่มสร้างข้อมูลทดสอบตามคุณสมบัติของเขตข้อมูลเพื่อใช้เป็นระเบียบข้อมูลสำหรับเขตข้อมูลนั้น รายละเอียดของคลาส Field จะเป็นดังรูปที่ 3.10

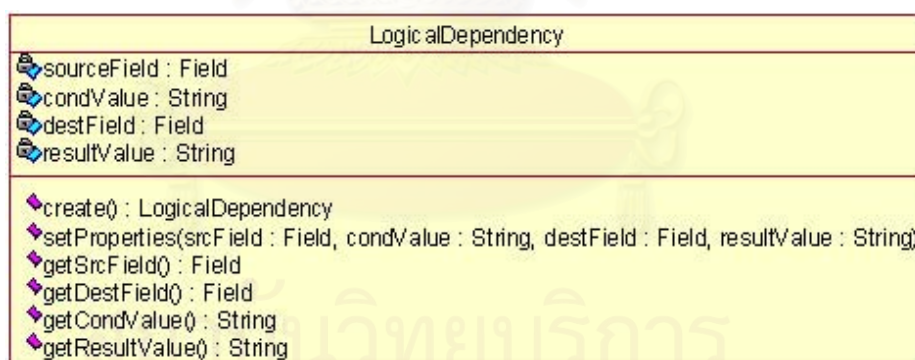
5. คลาส LogicalDependency คือคลาสที่เป็นตัวแทนของการขึ้นต่อกันเชิงตรรกะแบบปกติในแต่ละตารางข้อมูล คลาสนี้จะเก็บรายละเอียดของเงื่อนไขการขึ้นต่อกันเชิงตรรกะเพื่อนำไปใช้ในพิจารณาพร้อมในการสร้างข้อมูลทดสอบ รายละเอียดของคลาส LogicalDependency จะเป็นดังรูปที่ 3.11

6. คลาส SQLDependency คือคลาสที่เป็นตัวแทนของการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอลในแต่ละตารางข้อมูล คลาสนี้จะเก็บคำสั่งเอสคิวแอลที่กำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลเพื่อนำไปใช้ในพิจารณาพร้อมในการสร้างข้อมูลทดสอบ รายละเอียดของคลาส SQLDependency จะเป็นดังรูปที่ 3.12

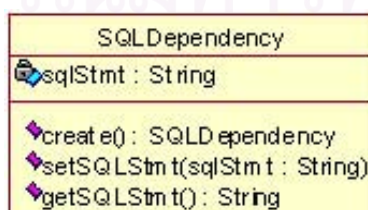
7. คลาส Category คือคลาสที่เป็นตัวแทนของกลุ่มค่าทั่วไปที่จะเก็บรายการข้อมูลค่าทั่วไป เพื่อนำไปใช้ร่วมในการสร้างข้อมูลทดสอบ รายละเอียดของคลาส Category จะเป็นดังรูปที่ 3.13



รูปที่ 3.10 คลาส Field



รูปที่ 3.11 คลาส LogicalDependency

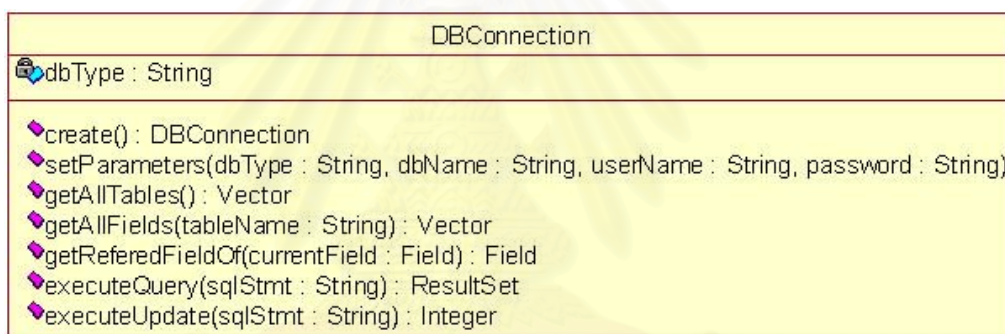


รูปที่ 3.12 คลาส SQLDependency



รูปที่ 3.13 คลาส Category

8. คลาส DBConnection คือคลาสที่เป็นตัวแทนการเชื่อมต่อกับฐานข้อมูลต่างๆ ที่จำเป็นต้องใช้ในการทำงานของเครื่อง เช่น ฐานข้อมูลต้นทางที่ให้ข้อมูลในการสร้างสคีมาของข้อมูล และฐานข้อมูลเป้าหมายที่จะบันทึกข้อมูลทดสอบ จะทำหน้าที่รับคำสั่งทุกอย่างที่ต้องทำกับฐานข้อมูลไปประมวลผล รายละเอียดของคลาส DBConnection จะเป็นดังรูปที่ 3.14



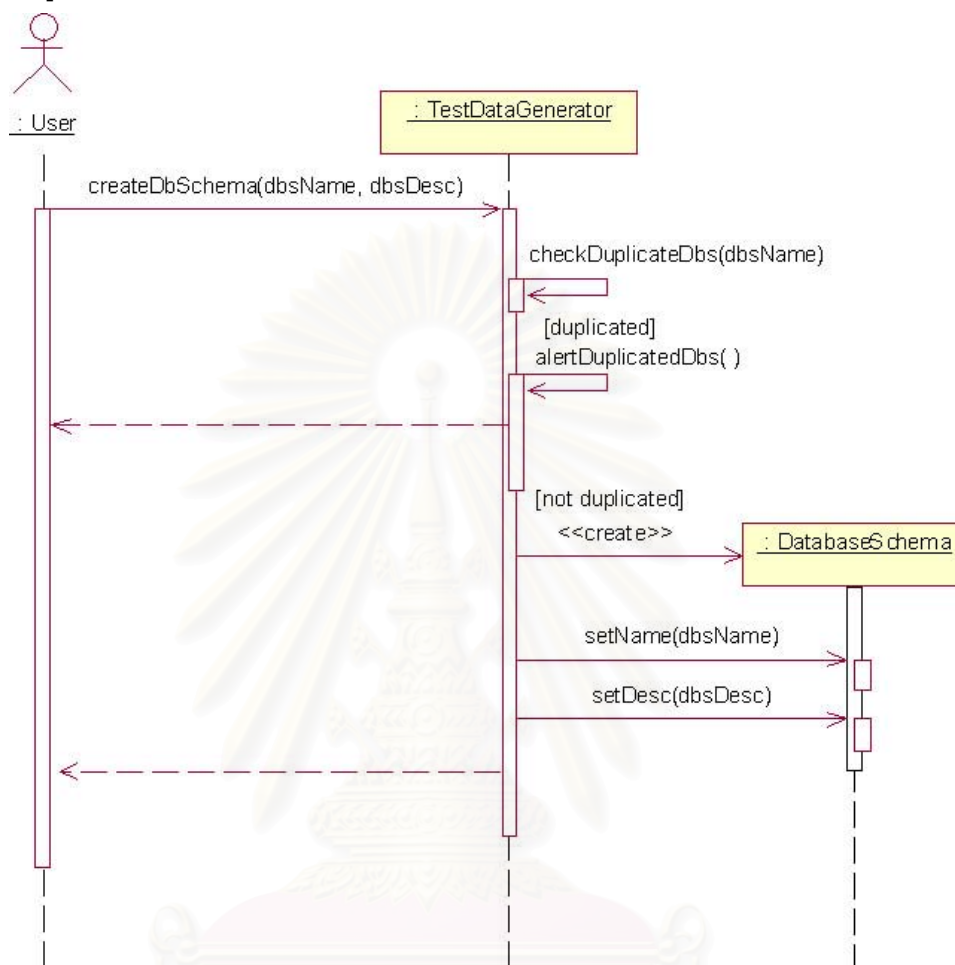
รูปที่ 3.14 คลาส DBConnection

3.2.3 แผนภาพซีควেনซ์และแผนภาพคอลลาบอเรชัน

แผนภาพซีควেনซ์และแผนภาพคอลลาบอเรชันเป็นแผนภาพที่แสดงการเกิด

ปฏิสัมพันธ์กันระหว่างวัตถุภายในระบบ การปฏิสัมพันธ์คือการส่งสาร (Message) กันระหว่างวัตถุ เพื่อให้บรรลุเป้าหมายในการทำงานหนึ่งๆ แผนภาพซีควেনซ์และแผนภาพคอลลาบอเรชันของเครื่องมือสามารถแบ่งอธิบายเป็นกลุ่มๆ ตามยูสเคส ได้แก่ ยูสเคสกำหนดชื่อสคีมาของฐานข้อมูล ยูสเคสสร้างสคีมาของฐานข้อมูลด้วยมือ ยูสเคสสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง ยูสเคสเปิดสคีมาของฐานข้อมูล ยูสเคสปิดสคีมาของฐานข้อมูล ยูสเคสลบสคีมาของฐานข้อมูล ยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย ยูสเคสเพิ่มกลุ่มค่าทั่วไปใหม่ และยูสเคสลบกลุ่มค่าทั่วไป ดังนี้

1. แผนภาพซีควเอนซ์ของยูสเคสกำหนดชื่อสคีมาของฐานข้อมูล แผนภาพซีควเอนซ์จะเป็นดังรูปที่ 3.15

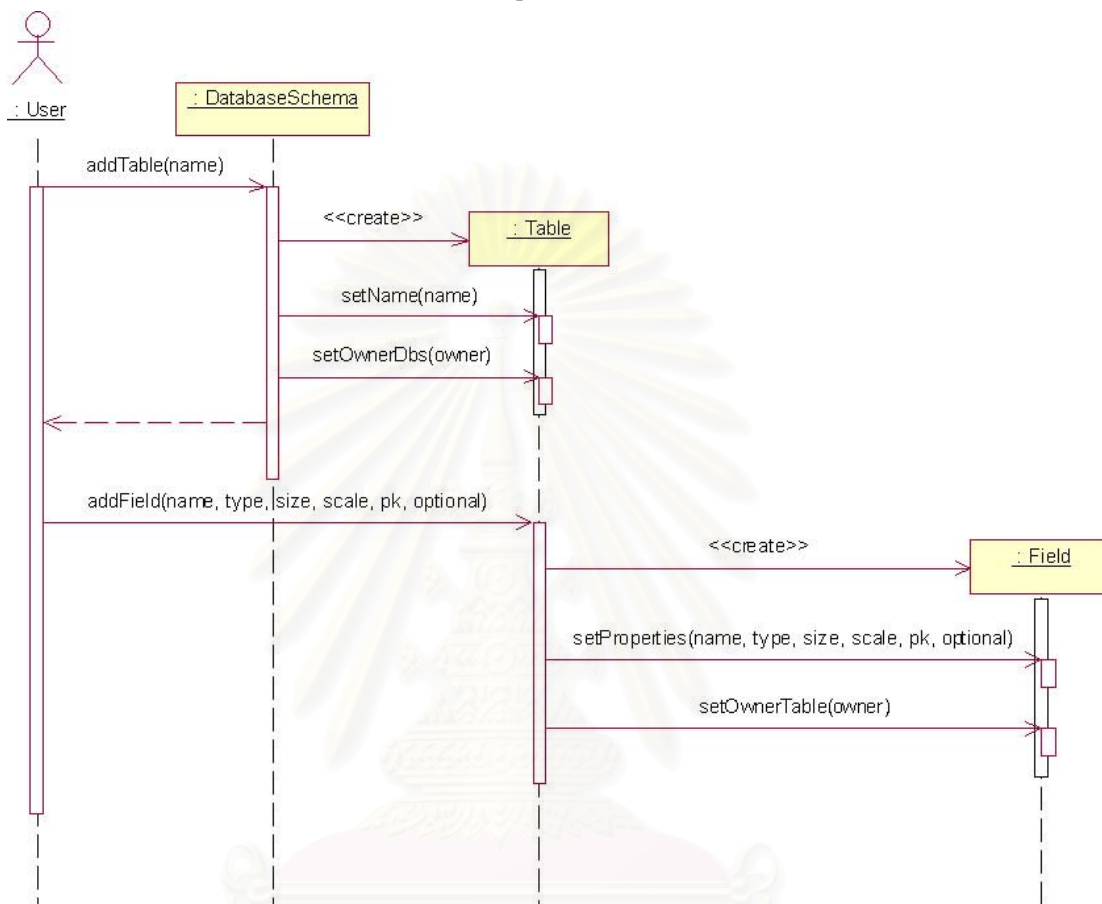


รูปที่ 3.15 แผนภาพซีควเอนซ์แสดงการกำหนดชื่อสคีมาของฐานข้อมูล

จากรูปที่ 3.15 การทำงานจะเริ่มจากผู้ใช้กำหนดชื่อและคำอธิบายของสคีมาของฐานข้อมูลเพื่อให้เครื่องมือ (คลาส TestDataGenerator) สร้างเป็นสคีมาของฐานข้อมูลใหม่ จากนั้นเครื่องมือจะนำชื่อสคีมาของฐานข้อมูลที่ผู้ใช้กำหนดมาไปตรวจสอบกับรายการสคีมาของฐานข้อมูลที่มีอยู่ว่าซ้ำซ้อนกันหรือไม่ หากพบว่าซ้ำซ้อนเครื่องมือจะแจ้งกลับไปยังผู้ใช้ให้ทราบ แต่ถ้าไม่มีการซ้ำซ้อนเครื่องมือจะสร้างวัตถุของคลาส DatabaseSchema ขึ้นมาแล้วกำหนดชื่อและคำอธิบายให้กับวัตถุนั้น

2. แผนภาพซีควเอนซ์ของยูสเคสสร้างสคีมาของฐานข้อมูลด้วยมือ สำหรับยูสเคสนี้จะเริ่มจากการเพิ่มข้อมูลให้กับสคีมาของฐานข้อมูลดังรูปที่ 3.16 โดยในขั้นแรกผู้ใช้จะเพิ่มตารางข้อมูลให้กับสคีมาของฐานข้อมูลก่อน จากนั้นผู้ใช้จะเพิ่มเขตข้อมูลให้กับตารางข้อมูล พร้อมทั้งกำหนดคุณสมบัติให้กับเขตข้อมูลนั้นด้วย เมื่อผู้ใช้กำหนดตารางข้อมูลและเขตข้อมูลให้กับ

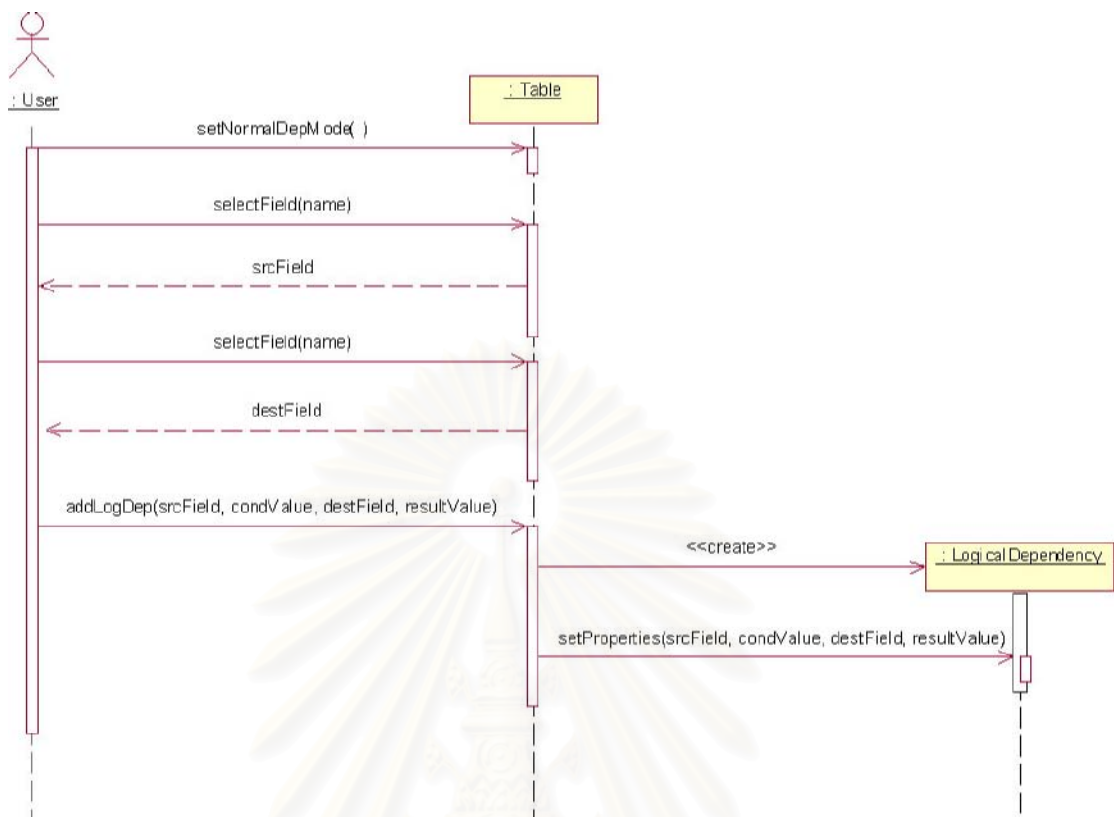
สคีมาของฐานข้อมูลแล้ว ผู้ใช้จะต้องกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลภายในตาราง ซึ่งผู้ใช้สามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลได้ 2 แบบดังที่กล่าวมาแล้ว คือ แบบปกติและแบบเอสคิวแอล ดังรูปที่ 3.17 และ 3.18 ตามลำดับ



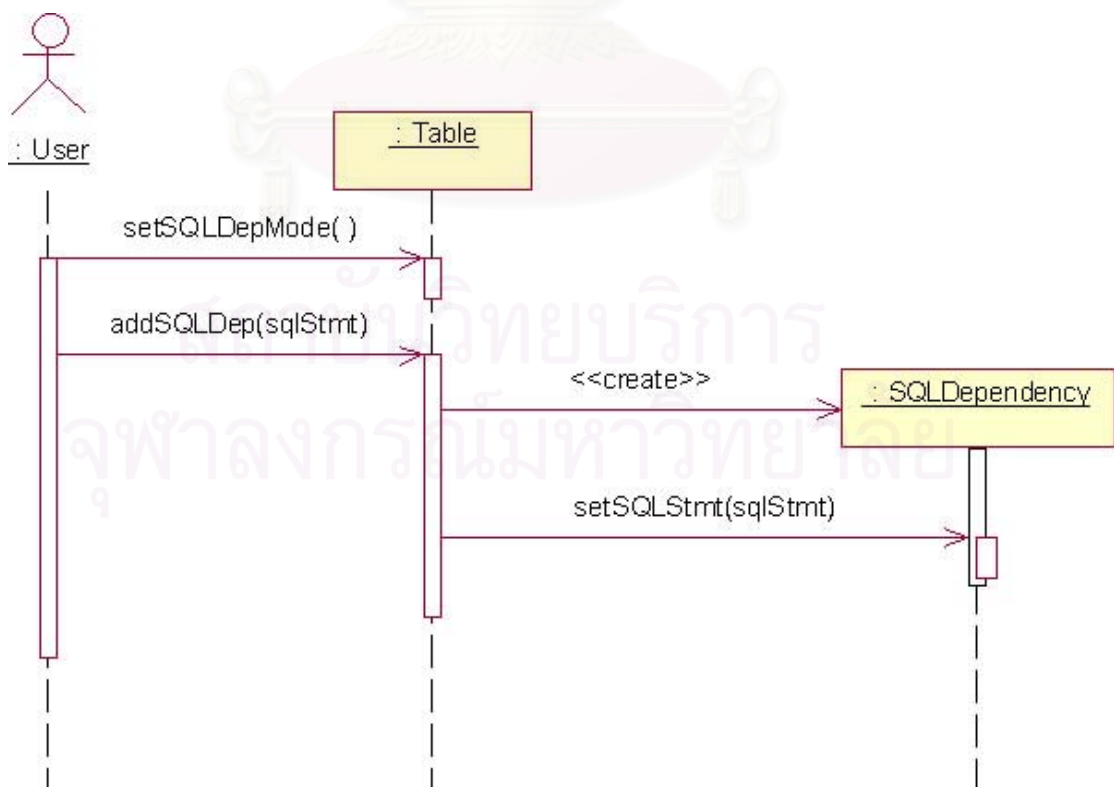
รูปที่ 3.16 แผนภาพซีควเอนซ์แสดงการเพิ่มข้อมูลให้กับสคีมาของฐานข้อมูล

จากรูปที่ 3.17 การกำหนดการขึ้นต่อกันเชิงตรรกะแบบปกติจะเริ่มจากผู้ใช้ทำการกำหนดให้วัตถุที่แทนตารางข้อมูลใช้การขึ้นต่อกันเชิงตรรกะแบบปกติ จากนั้นผู้ใช้จะเลือกเขตข้อมูลต้นทาง (Source Field) และเขตข้อมูลปลายทาง (Destination Field) ของเงื่อนไข พร้อมทั้งกำหนดค่าข้อมูลเงื่อนไข (Condition value) และค่าผลลัพธ์ (Result value) ของเงื่อนไข แล้วผู้ใช้จึงสั่งเพิ่มเงื่อนไขให้กับวัตถุที่แทนตารางข้อมูล ซึ่งวัตถุนี้จะนำเงื่อนไขที่กำหนดมาไปสร้างเป็นวัตถุของคลาส LogicalDependency เก็บไว้ในตัวเอง

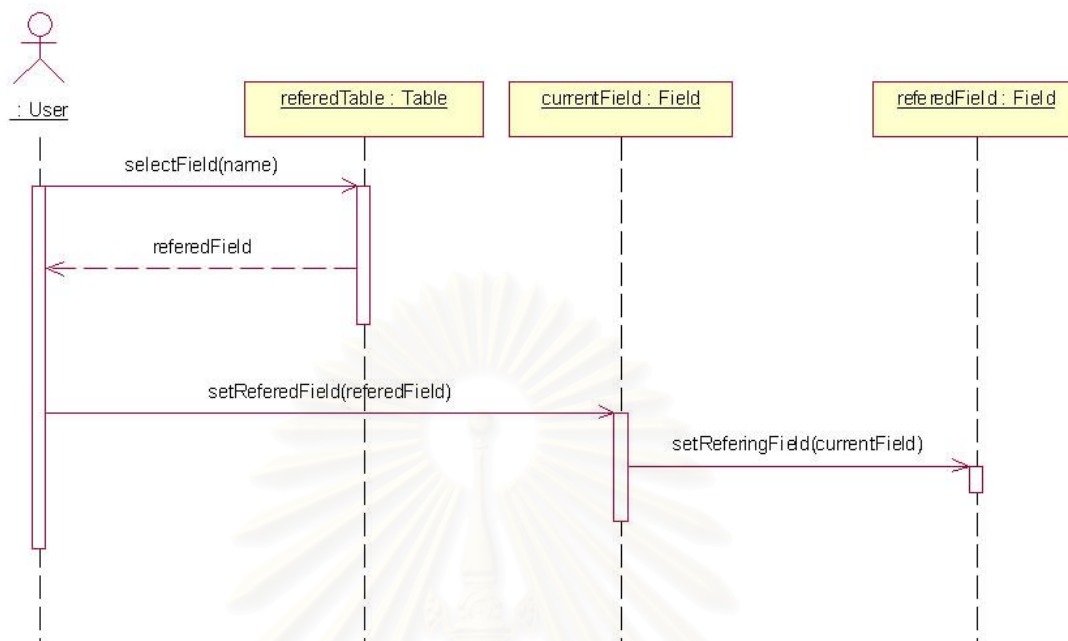
จากรูปที่ 3.18 การกำหนดการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอลจะเริ่มจากผู้ใช้ทำการกำหนดให้ตารางข้อมูลใช้การขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล จากนั้นผู้ใช้จะเพิ่มคำสั่งเอสคิวแอลที่ต้องการให้กับตารางข้อมูล ซึ่งตารางข้อมูลจะนำคำสั่งเอสคิวแอลไปสร้างเป็นวัตถุของคลาส SQLDependency เก็บไว้ในตาราง



รูปที่ 3.17 แผนภาพซีควเอนซ์แสดงการกำหนดการขึ้นต่อกันเชิงตรรกะแบบปกติ



รูปที่ 3.18 แผนภาพซีควเอนซ์แสดงการกำหนดการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล

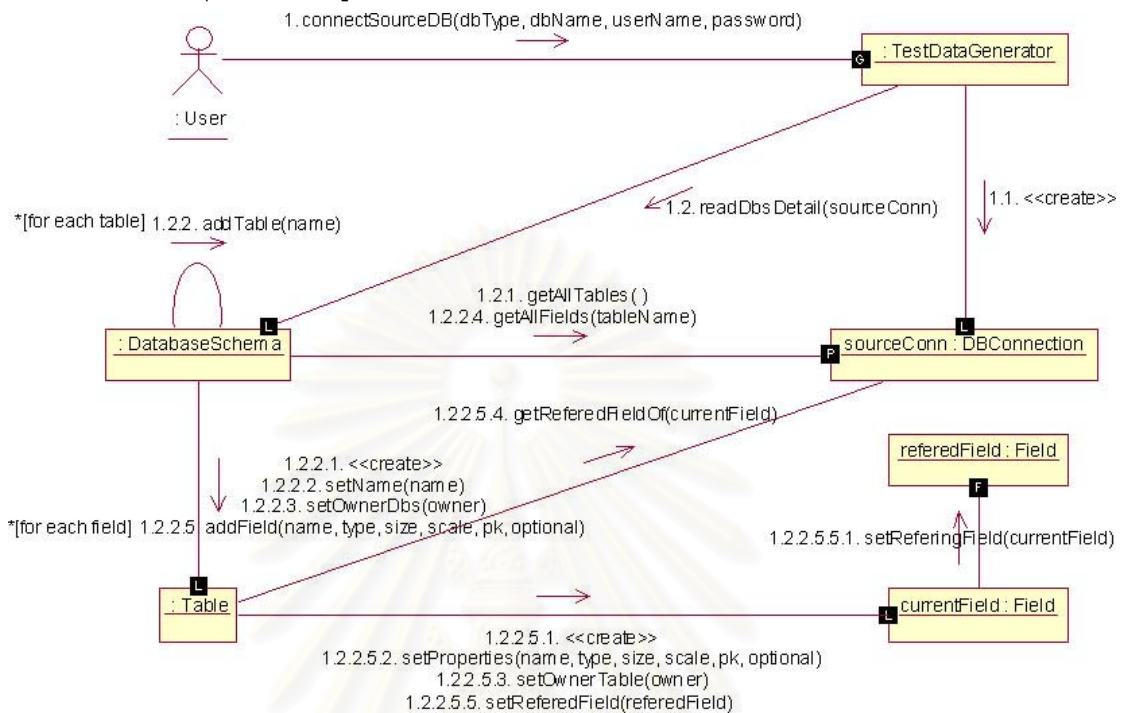


รูปที่ 3.19 แผนภาพซีควเอนซ์แสดงการกำหนดความคงสภาพในการอ้างอิง

เมื่อผู้ใช้กำหนดการขึ้นต่อกันเชิงตรรกะให้กับตารางข้อมูลแล้ว ผู้ใช้จะต้องกำหนดความคงสภาพในการอ้างอิงให้กับตารางข้อมูลด้วย โดยการกำหนดความคงสภาพในการอ้างอิงจะเป็นดังรูปที่ 3.19 เริ่มจากผู้ใช้เลือกเขตข้อมูลที่จะถูกอ้างถึงจากตารางข้อมูลที่เป็นเจ้าของเขตข้อมูลนั้น แล้วนำเขตข้อมูลนั้นมากำหนดให้เขตข้อมูลปัจจุบัน พร้อมทั้งกำหนดให้เขตข้อมูลที่ถูกอ้างถึงรู้ด้วยว่าถูกอ้างถึงโดยเขตข้อมูลปัจจุบัน

3. แผนภาพคอลลาบอเรชันของยูสเคสสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง แผนภาพคอลลาบอเรชันจะเป็นดังรูปที่ 3.20 การสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทางจะเริ่มจากผู้ใช้สั่งให้เครื่องมือเชื่อมกับฐานข้อมูลต้นทางโดยส่งพารามิเตอร์สำหรับการเชื่อมต่อมาให้ ซึ่งเครื่องมือจะเชื่อมต่อกับฐานข้อมูลต้นทางโดยการสร้างวัตถุ sourceConn จากคลาส DBConnection เพื่อใช้เป็นตัวแทนการเชื่อมต่อ จากนั้นเครื่องมือจะสั่งให้สคีมาของฐานข้อมูลอ่านรายละเอียดของสคีมาของฐานข้อมูลต้นทางเพื่อดึงข้อมูลมาเก็บไว้ ในขั้นแรกสคีมาของฐานข้อมูลจะอ่านรายการตารางข้อมูลทั้งหมดในฐานข้อมูลต้นทางมาสร้างเป็นวัตถุของคลาส Table เก็บไว้เป็นในสคีมาของฐานข้อมูล โดยในระหว่างการสร้างวัตถุของแต่ละตารางข้อมูลนั้น จะมีการอ่านรายการเขตข้อมูลทั้งหมดของตารางข้อมูลนั้นๆ ขึ้นมาเพื่อสร้างเป็นวัตถุของคลาส Field เก็บไว้ในแต่ละตารางข้อมูลด้วย และในทำนองเดียวกัน ในระหว่างการสร้างวัตถุของแต่ละ

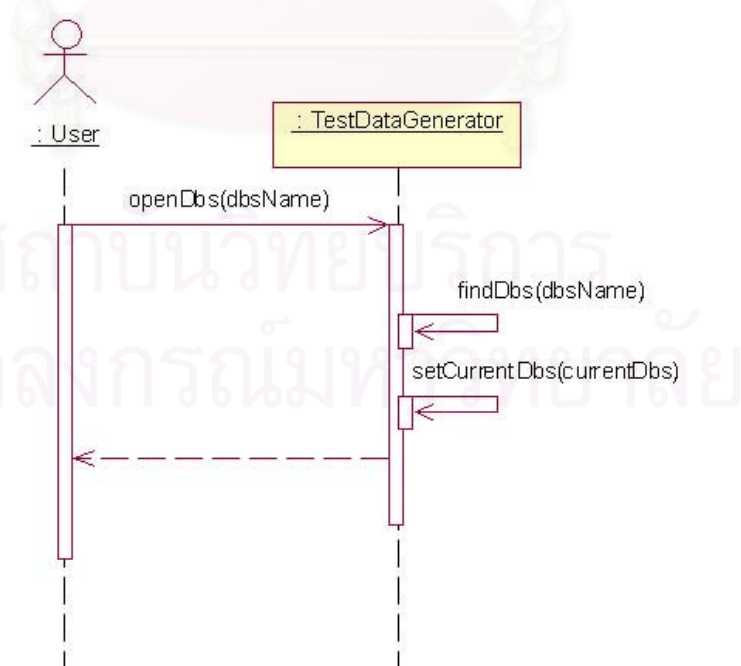
เขตข้อมูลนั้น จะมีการอ่านข้อมูลเกี่ยวกับการอ้างอิงถึงเขตข้อมูลอื่นของเขตข้อมูลนั้นขึ้นมาเพื่อกำหนดให้กับวัตถุของเขตข้อมูลนั้นด้วย



รูปที่ 3.20 แผนภาพคอลลาบอเรชันแสดงการสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง

4. แผนภาพซีควเอนซ์ของยูสเคสเปิดสคีมาของฐานข้อมูล แผนภาพซีควเอนซ์จะ

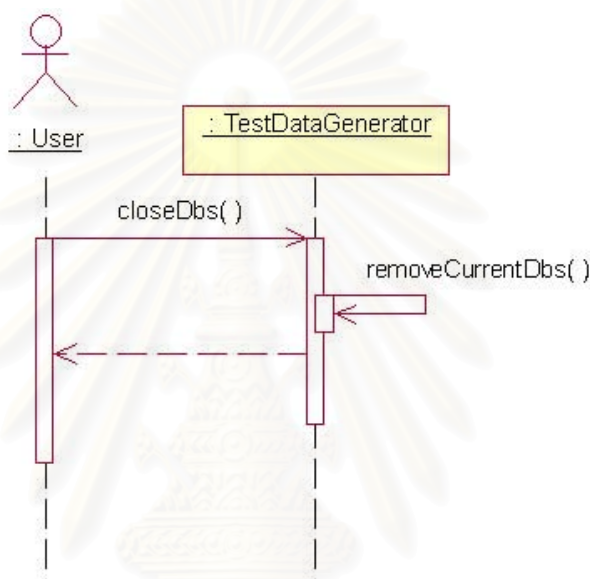
เป็นดังรูปที่ 3.21



รูปที่ 3.21 แผนภาพซีควเอนซ์แสดงการเปิดสคีมาของฐานข้อมูล

จากรูปที่ 3.21 การเปิดสคีมาของฐานข้อมูลจะเริ่มจากผู้ใช้งานสั่งให้เครื่องมือเปิดสคีมาของฐานข้อมูลที่มีชื่อตรงกับที่ผู้ใช้งานกำหนด จากนั้นเครื่องมือจะนำชื่อสคีมาของฐานข้อมูลไปค้นหาในรายการสคีมาของฐานข้อมูลทั้งหมด แล้วจะนำเอาสคีมาของฐานข้อมูลที่พบมากำหนดให้เป็นสคีมาของฐานข้อมูลปัจจุบันที่จะใช้งาน

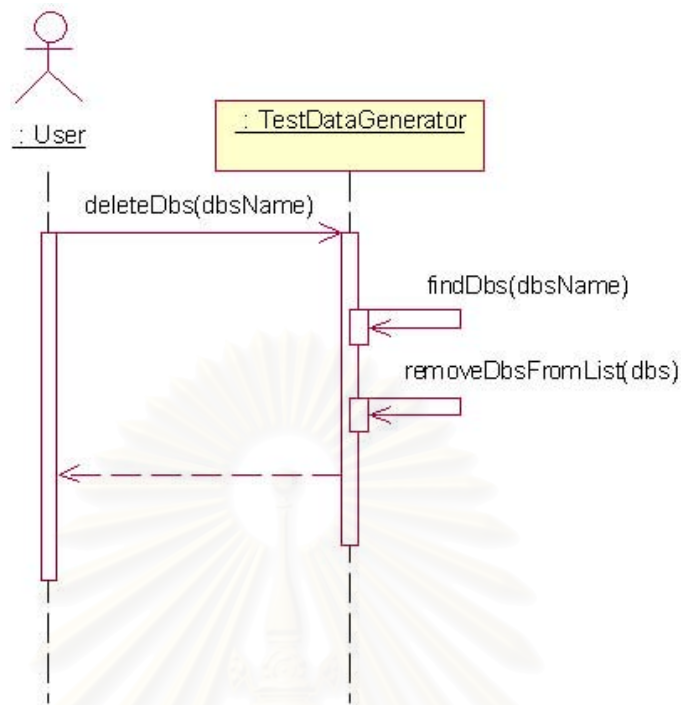
5. แผนภาพซีควเอนซ์ของยูสเคสเปิดสคีมาของฐานข้อมูล แผนภาพซีควเอนซ์จะเป็นดังรูปที่ 3.22



รูปที่ 3.22 แผนภาพซีควเอนซ์แสดงการปิดสคีมาของฐานข้อมูล

จากรูปที่ 3.22 การปิดสคีมาของฐานข้อมูลจะเริ่มจากผู้ใช้งานสั่งให้เครื่องมือปิดสคีมาของฐานข้อมูลปัจจุบันที่กำลังใช้งานอยู่ จากนั้นเครื่องมือจะเอาสคีมาของฐานข้อมูลปัจจุบันออกเพื่อเป็นการปิดสคีมาของฐานข้อมูลปัจจุบัน

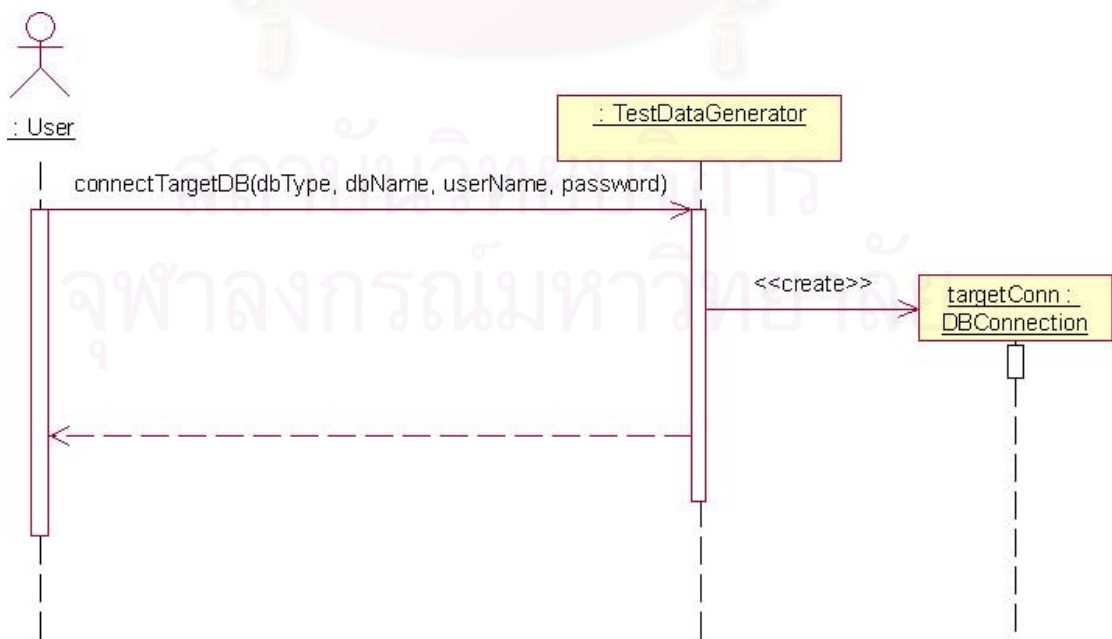
6. แผนภาพซีควเอนซ์ของยูสเคสลบสคีมาของฐานข้อมูล แผนภาพซีควเอนซ์จะเป็นดังรูปที่ 3.23 การลบสคีมาของฐานข้อมูลจะเริ่มจากผู้ใช้งานสั่งให้เครื่องมือลบสคีมาของฐานข้อมูลที่มีชื่อตรงกับที่ผู้ใช้งานกำหนด จากนั้นเครื่องมือจะนำชื่อสคีมาของฐานข้อมูลไปค้นหาในรายการสคีมาของฐานข้อมูลทั้งหมด แล้วจะเอาสคีมาของฐานข้อมูลที่พบออกจากรายการสคีมาของฐานข้อมูล



รูปที่ 3.23 แผนภาพซีควเอนซ์แสดงการลบสคีมาของฐานข้อมูล

7. แผนภาพซีควเอนซ์ของยูสเคสกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

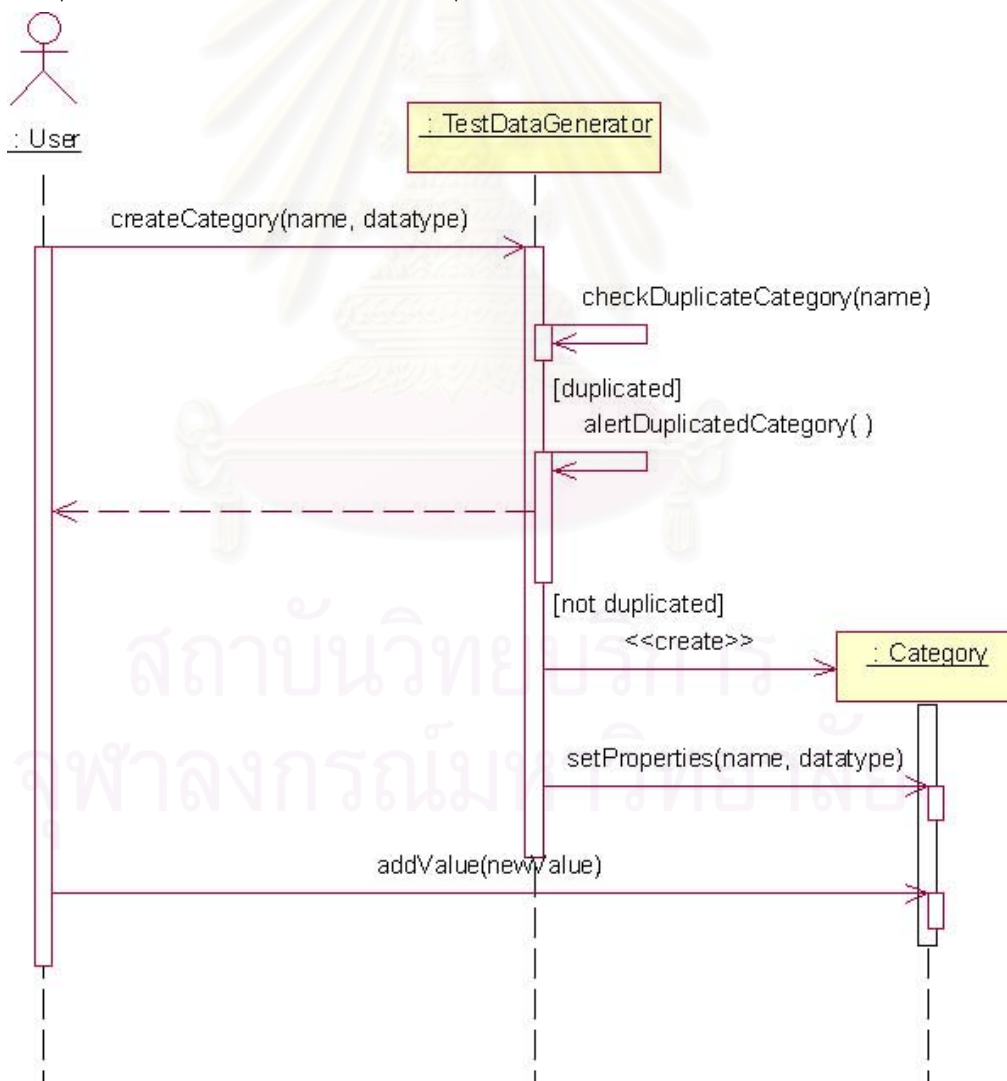
แผนภาพซีควเอนซ์จะเป็นดังรูปที่ 3.24 การกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมายจะเริ่มจากผู้ใช้สั่งให้เครื่องมือเชื่อมกับฐานข้อมูลเป้าหมายโดยส่งพารามิเตอร์สำหรับการเชื่อมต่อมาให้ ซึ่งเครื่องมือจะเชื่อมต่อกับฐานข้อมูลเป้าหมายโดยการสร้างวัตถุ targetConn จากคลาส DBConnection เพื่อใช้เป็นตัวแทนการเชื่อมต่อแล้วเก็บไว้ใช้ในภายหลัง



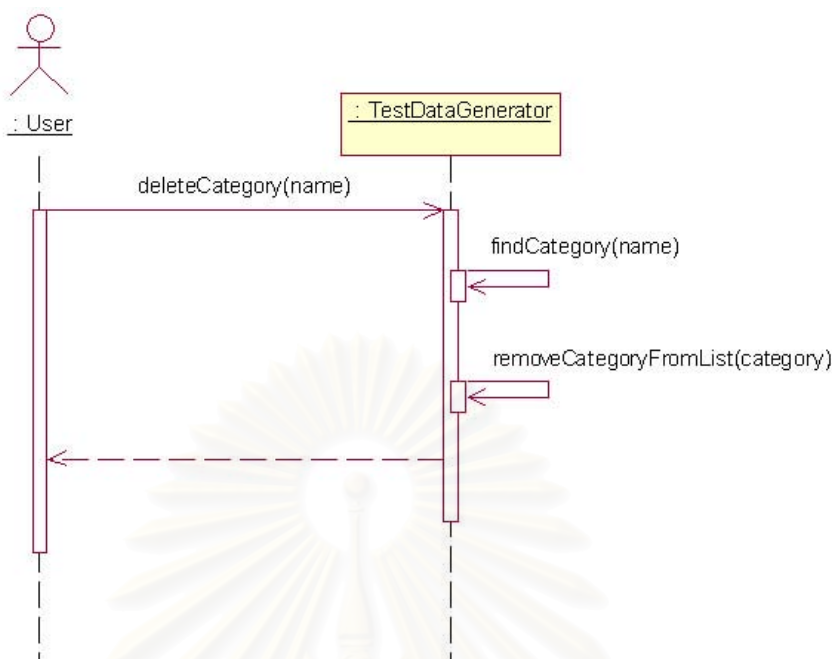
รูปที่ 3.24 แผนภาพซีควเอนซ์แสดงการกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

8. แผนภาพซีควเอนซ์ของยูสเคสเพิ่มกลุ่มค่าทั่วไปใหม่ แผนภาพซีควเอนซ์จะเป็นดังรูปที่ 3.25 การเพิ่มกลุ่มค่าทั่วไปใหม่จะเริ่มจากผู้ใช้สั่งให้เครื่องสร้างกลุ่มค่าทั่วไปโดยกำหนดชื่อและชนิดข้อมูลมาให้ จากนั้นเครื่องมือจะนำชื่อกลุ่มค่าทั่วไปที่ผู้ใช้กำหนดมาไปตรวจสอบกับรายการกลุ่มค่าทั่วไปที่มีอยู่ว่าซ้ำซ้อนกันหรือไม่ หากพบว่าซ้ำซ้อนเครื่องมือจะแจ้งกลับไปยังผู้ใช้ให้ทราบ แต่ถ้าไม่มีการซ้ำซ้อนเครื่องมือจะสร้างวัตถุของคลาส Category ขึ้นมาแล้วกำหนดชื่อและชนิดข้อมูลให้กับวัตถุนั้น สุดท้ายผู้ใช้จึงทำการเพิ่มค่าข้อมูลให้กับวัตถุนั้นตามต้องการ

9. แผนภาพซีควเอนซ์ของยูสเคสลบกลุ่มค่าทั่วไป แผนภาพซีควเอนซ์จะเป็นดังรูปที่ 3.26 การลบกลุ่มค่าทั่วไปจะเริ่มจากผู้ใช้สั่งให้เครื่องมือลบกลุ่มค่าทั่วไปที่มีชื่อตรงกับที่ผู้ใช้กำหนด จากนั้นเครื่องมือจะนำชื่อของกลุ่มค่าทั่วไปไปค้นหาในรายการกลุ่มค่าทั่วไปทั้งหมด แล้วจะเอากลุ่มค่าทั่วไปที่พบออกจากรายการกลุ่มค่าทั่วไป



รูปที่ 3.25 แผนภาพซีควเอนซ์แสดงการเพิ่มกลุ่มค่าทั่วไปใหม่

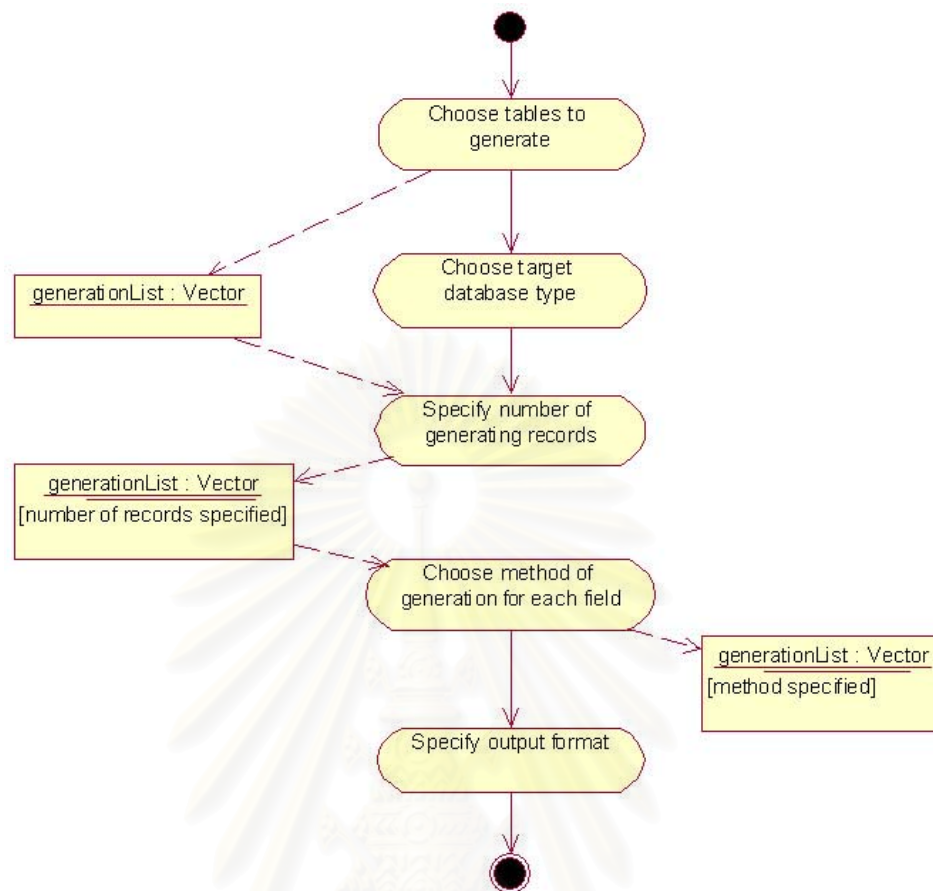


รูปที่ 3.26 แผนภาพซีควเอนซ์แสดงการลบกลุ่มค่าทั่วไป

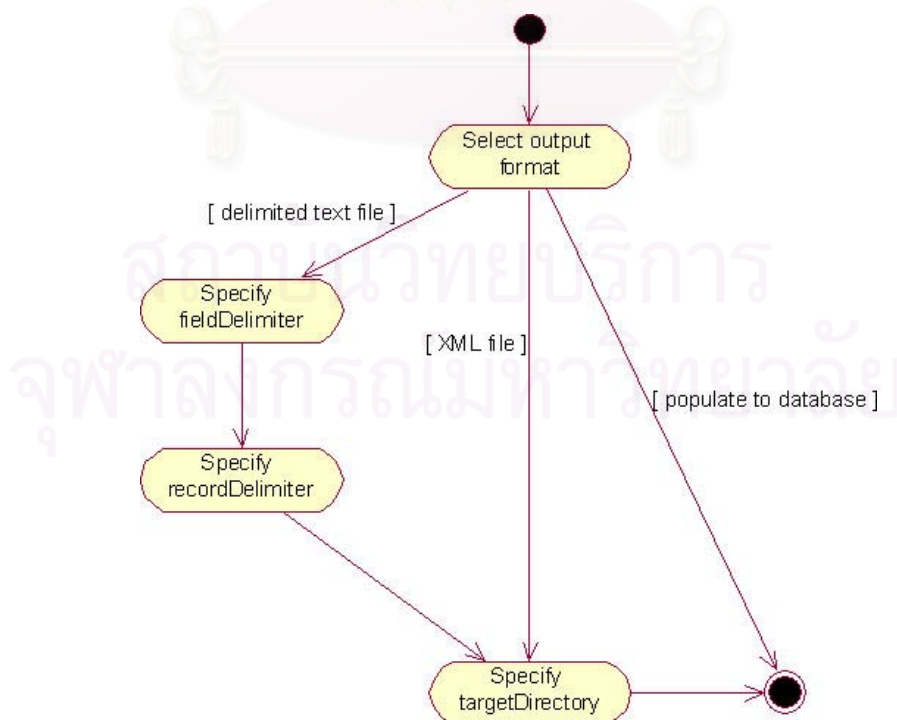
3.2.4 แผนภาพลำดับกิจกรรม

แผนภาพลำดับกิจกรรมเป็นแผนภาพที่แสดงลำดับขั้นตอนการทำงานจากกิจกรรมหนึ่งไปยังอีกกิจกรรมเพื่อให้บรรลุเป้าหมายของการทำงานอย่างใดอย่างหนึ่งของระบบ แผนภาพลำดับกิจกรรมของเครื่องมือนั้นสามารถแบ่งอธิบายเป็นกลุ่มๆ ตามยูสเคส ได้แก่ ยูสเคส กำหนดรายละเอียดการสร้างข้อมูลทดสอบ ยูสเคสสร้างข้อมูลทดสอบ ยูสเคสสร้างเพิ่มความที่มีตัวค้น ยูสเคสเพิ่มเอกสารอิเล็กทรอนิกส์อีเมล และยูสเคสเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมาย ดังนี้

1. แผนภาพลำดับกิจกรรมของยูสเคสกำหนดรายละเอียดการสร้างข้อมูลทดสอบ
 แผนภาพลำดับกิจกรรมจะเป็นดังรูปที่ 3.27 การกำหนดรายละเอียดการสร้างข้อมูลทดสอบจะเริ่มจากผู้ใช้เลือกตารางข้อมูลที่ต้องการสร้างข้อมูลทดสอบ ซึ่งจะได้ออกมาเป็นรายการตารางข้อมูลที่จะสร้างข้อมูลทดสอบ (generationList) จากนั้นผู้ใช้ต้องเลือกชนิดของฐานข้อมูลเป้าหมายที่จะนำข้อมูลทดสอบไปใช้ ต่อมาผู้ใช้จะต้องกำหนดจำนวนของระเบียบข้อมูลที่ต้องการสร้างข้อมูลทดสอบให้กับแต่ละตารางข้อมูล พร้อมทั้งเลือกวิธีการสร้างข้อมูลทดสอบให้กับแต่ละเขตข้อมูลของแต่ละตารางข้อมูลด้วย และสุดท้ายผู้ใช้ต้องกำหนดรูปแบบการบันทึกข้อมูลทดสอบดังรูปที่ 3.28 จึงเป็นอันเสร็จสิ้นการกำหนดรายละเอียดการสร้างข้อมูลทดสอบ

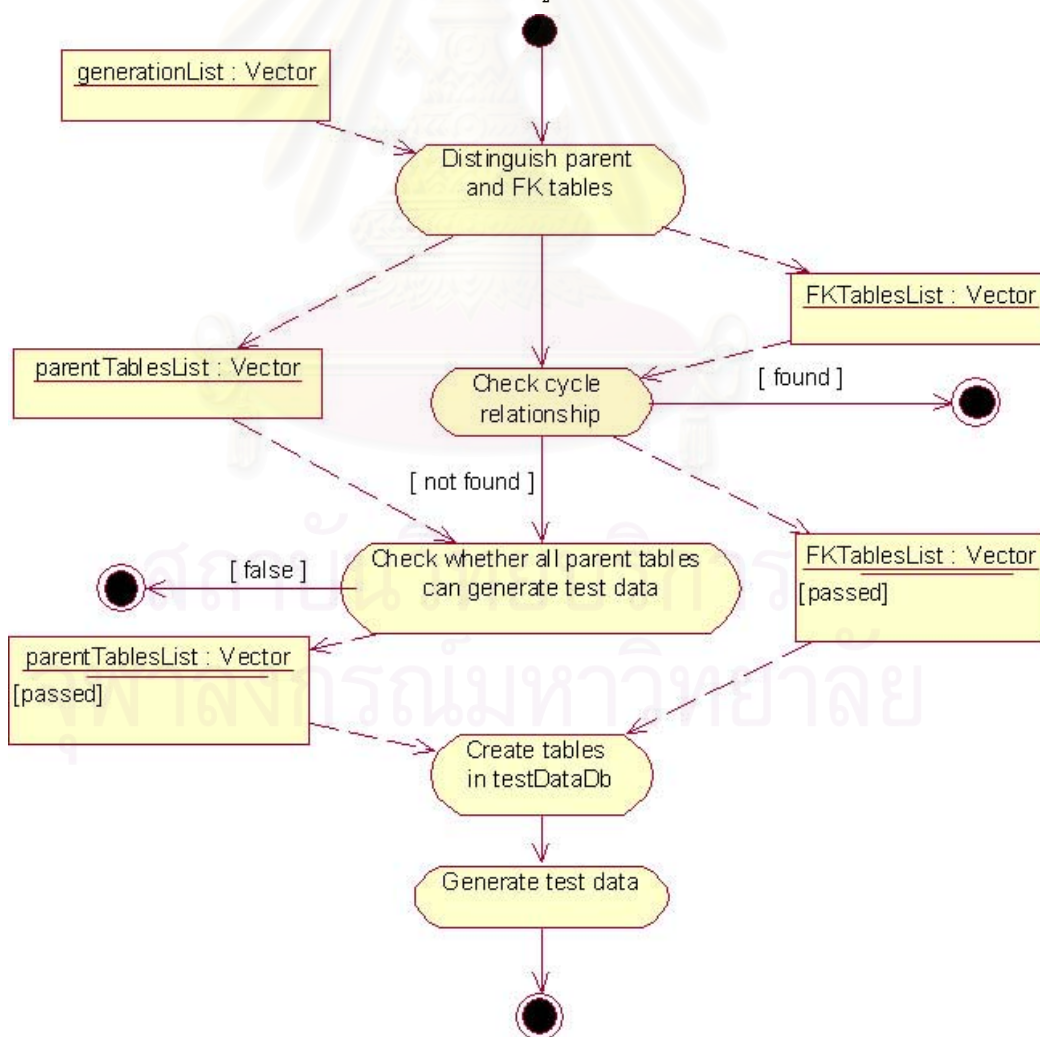


รูปที่ 3.27 แผนภาพลำดับกิจกรรมแสดงการกำหนดรายละเอียดการสร้างข้อมูลทดสอบ

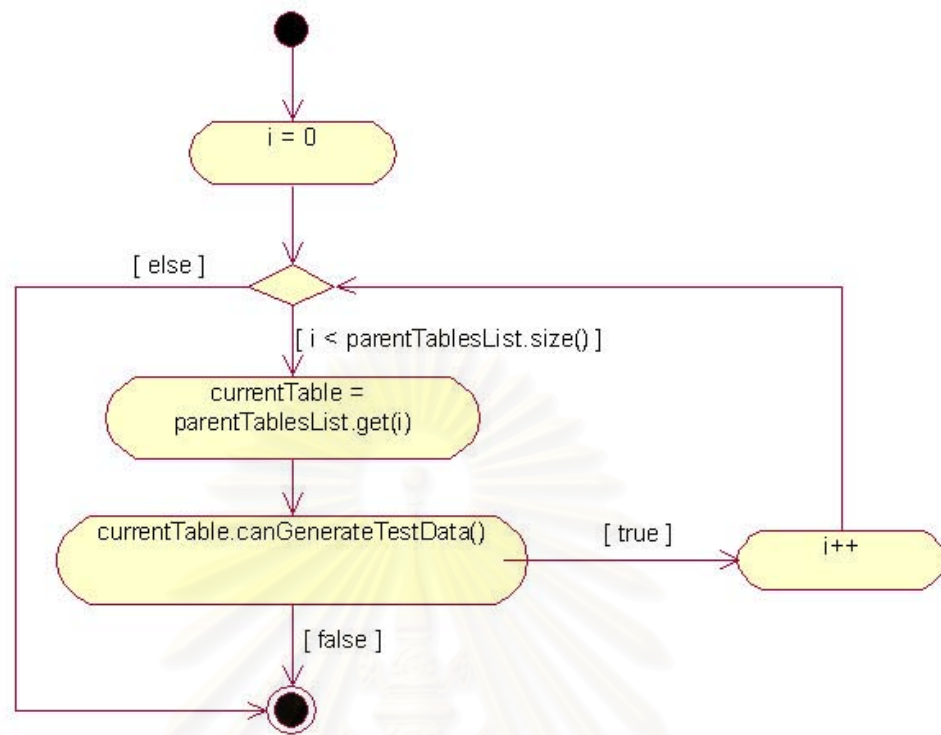


รูปที่ 3.28 แผนภาพลำดับกิจกรรมแสดงการกำหนดรูปแบบการบันทึกข้อมูลทดสอบ

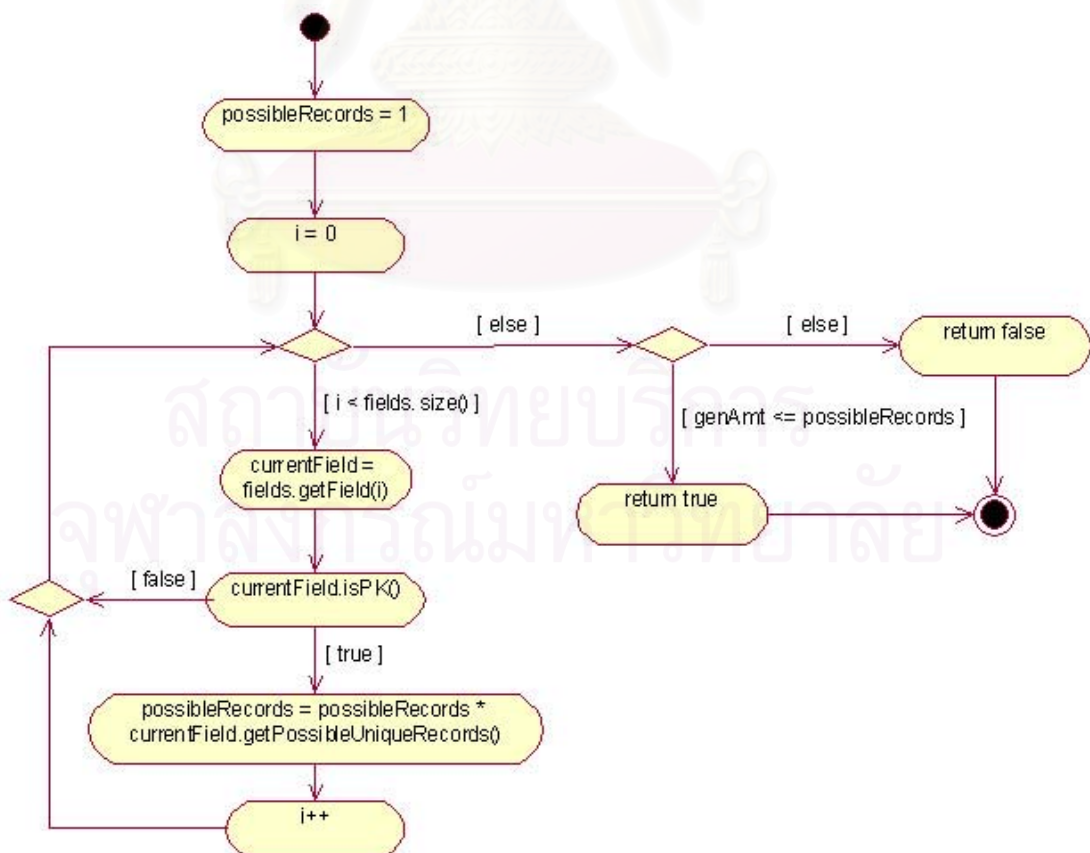
2. แผนภาพลำดับกิจกรรมของยูสเคสสร้างข้อมูลทดสอบ แผนภาพลำดับกิจกรรมจะเป็นดังรูปที่ 3.29 กระบวนการสร้างข้อมูลทดสอบจะเริ่มจากเครื่องมือทำการแยกรายการตารางข้อมูลที่จะสร้างข้อมูลทดสอบออกเป็น 2 รายการคือ รายการตารางข้อมูลที่ไม่มีเขตข้อมูลอ้างอิงถึงเขตข้อมูลในตารางข้อมูลอื่นๆ แทนด้วย parentTablesList และรายการตารางข้อมูลที่มีเขตข้อมูลอ้างอิงถึงเขตข้อมูลในตารางข้อมูลอื่นๆ แทนด้วย FKTablesList จากนั้นจะนำ FKTablesList ไปทำการตรวจสอบหาความสัมพันธ์แบบวงกลม (Cycle relationship) ซึ่งเป็นจะเกิดขึ้นเมื่อมีตารางข้อมูลอ้างอิงต่อกันไปเป็นวงกลม หากพบความสัมพันธ์แบบนี้เครื่องมือจะหยุดกระบวนการสร้างข้อมูลทดสอบทันที แต่ถ้าไม่พบเครื่องมือจะนำ parentTablesList ไปทำการตรวจสอบว่าทุกๆ ตารางที่อยู่ในรายการสามารถสร้างข้อมูลทดสอบตามจำนวนระเบียบข้อมูลที่ใช้กำหนดมาได้หรือไม่ หากพบว่าไม่มีตารางข้อมูลอันใดอันหนึ่งที่ไม่สามารถสร้างข้อมูลทดสอบได้ เครื่องมือจะหยุดกระบวนการสร้างข้อมูลทดสอบเช่นกัน โดยรายละเอียดการตรวจสอบสามารถแสดงได้ดังรูปที่ 3.30 และ 3.31



รูปที่ 3.29 แผนภาพลำดับกิจกรรมแสดงกระบวนการสร้างข้อมูลทดสอบ



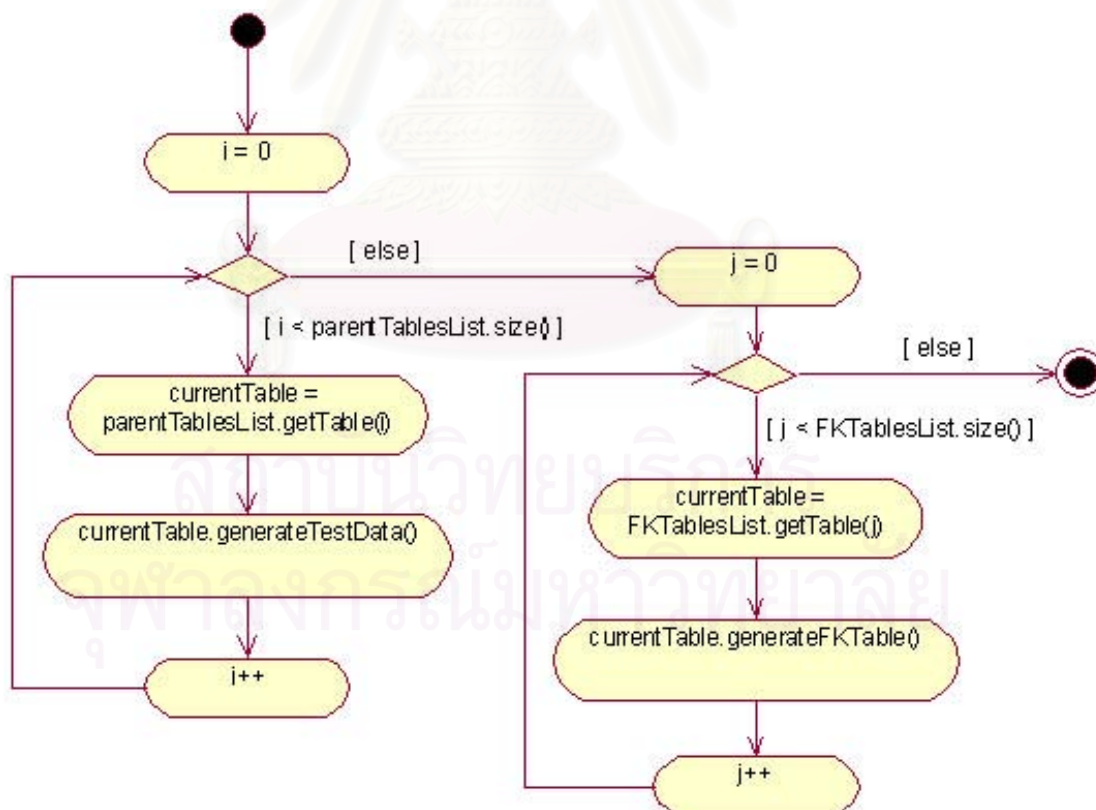
รูปที่ 3.30 แผนภาพลำดับกิจกรรมแสดงการตรวจสอบตารางข้อมูลในการสร้างข้อมูลทดสอบ



รูปที่ 3.31 แผนภาพลำดับกิจกรรมของ Table.canGenerateTestData()

จากรูปที่ 3.30 และ 3.31 แต่ละตารางข้อมูลที่อยู่ใน parentTablesList จะถูกนำมาตรวจสอบว่าสามารถสร้างข้อมูลทดสอบได้หรือไม่ด้วยการเรียกเมทอด (Method) canGenerateTestData() ของแต่ละตารางซึ่งการทำงานของเมทอดนี้จะทำการคำนวณจำนวนระเบียบข้อมูลที่เป็นไปได้ในการสร้างข้อมูลทดสอบ (possibleRecords) ของตารางข้อมูล โดยให้นำเอาจำนวนระเบียบข้อมูลที่ไม่ซ้ำกันที่เป็นไปได้ (getPossibleUniqueRecords()) ของเขตข้อมูลที่เป็นคีย์หลักมาคูณกัน จากนั้นจะตรวจสอบว่าจำนวนระเบียบข้อมูลที่เป็นไปได้ที่คำนวณได้นั้นน้อยกว่าหรือเท่ากับจำนวนระเบียบข้อมูลที่ใช้กำหนดมาหรือไม่ ถ้าใช่แสดงว่าตารางข้อมูลสามารถสร้างข้อมูลทดสอบได้ แต่ถ้าไม่ใช่แสดงว่าตารางข้อมูลนี้จะไม่สามารถสร้างข้อมูลทดสอบได้

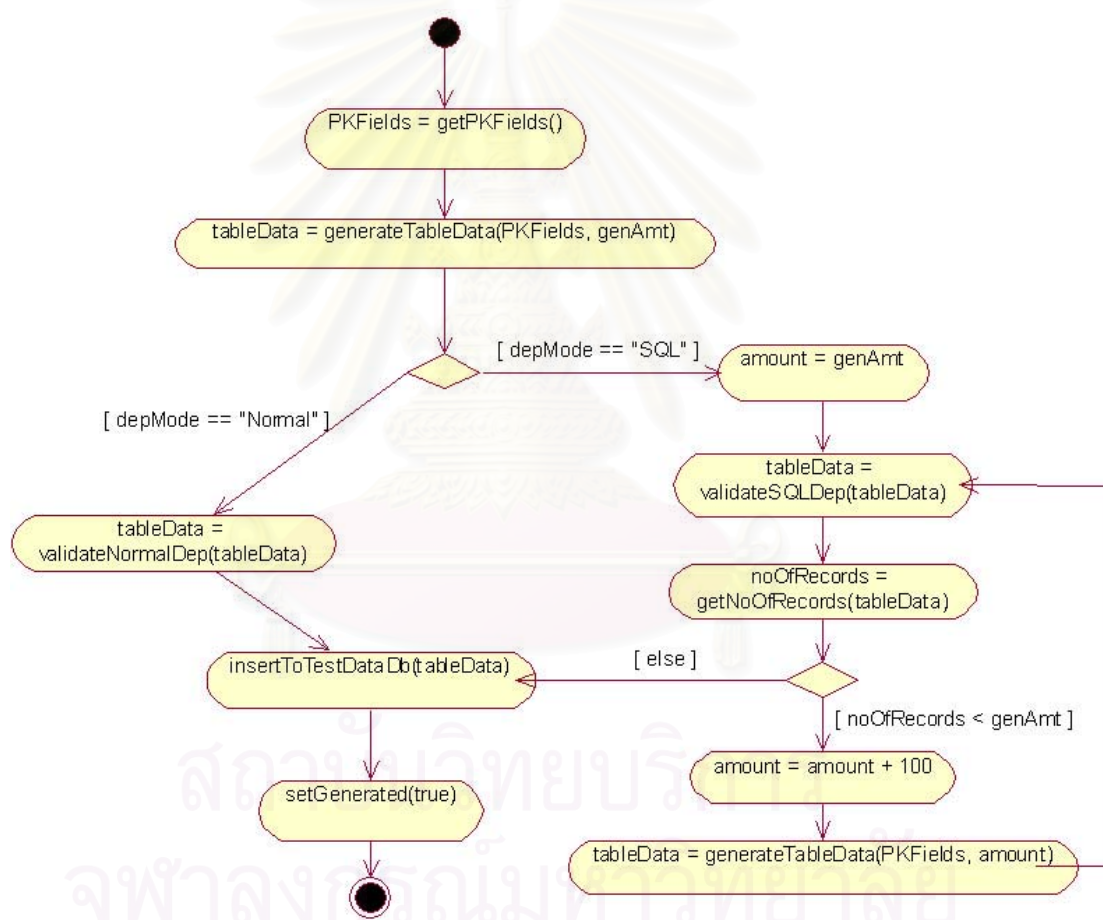
เมื่อผ่านกระบวนการตรวจสอบที่กล่าวมาแล้วเครื่องมือจะสร้างตารางข้อมูลทั้งหมดลงใน testDataDb ซึ่งเป็นฐานข้อมูลชั่วคราวที่ใช้สำหรับเก็บข้อมูลทดสอบของแต่ละตาราง ในระหว่างกระบวนการสร้างข้อมูลทดสอบ จากนั้นการสร้างข้อมูลทดสอบให้กับแต่ละตารางจะเริ่มขึ้น โดยรายละเอียดการสร้างข้อมูลทดสอบ จะเป็นดังรูปที่ 3.32



รูปที่ 3.32 แผนภาพลำดับกิจกรรมแสดงการสร้างข้อมูลทดสอบ

การสร้างข้อมูลทดสอบจะเริ่มจาก parentTablesList ก่อน โดยจะสร้างข้อมูลทดสอบให้ที่ละตารางใน parentTablesList ด้วยการเรียกเมทอด generateTestData() ของแต่ละตาราง เมื่อสร้างข้อมูลทดสอบให้ครบทุกตารางใน parentTablesList แล้ว จึงจะสร้างข้อมูลทดสอบให้กับแต่ละตารางใน FKTablesList ด้วยการเรียกเมทอด generateFKTable() ของแต่ละตารางจึงถือเป็นการสิ้นสุดการสร้างข้อมูลทดสอบ สำหรับรายละเอียดการทำงานของเมทอด generateTestData() และ generateFKTable() สามารถแสดงด้วยแผนภาพลำดับกิจกรรมดังนี้

2.1 แผนภาพลำดับกิจกรรมของเมทอด generateTestData() สามารถแสดงได้ดังรูปที่ 3.33



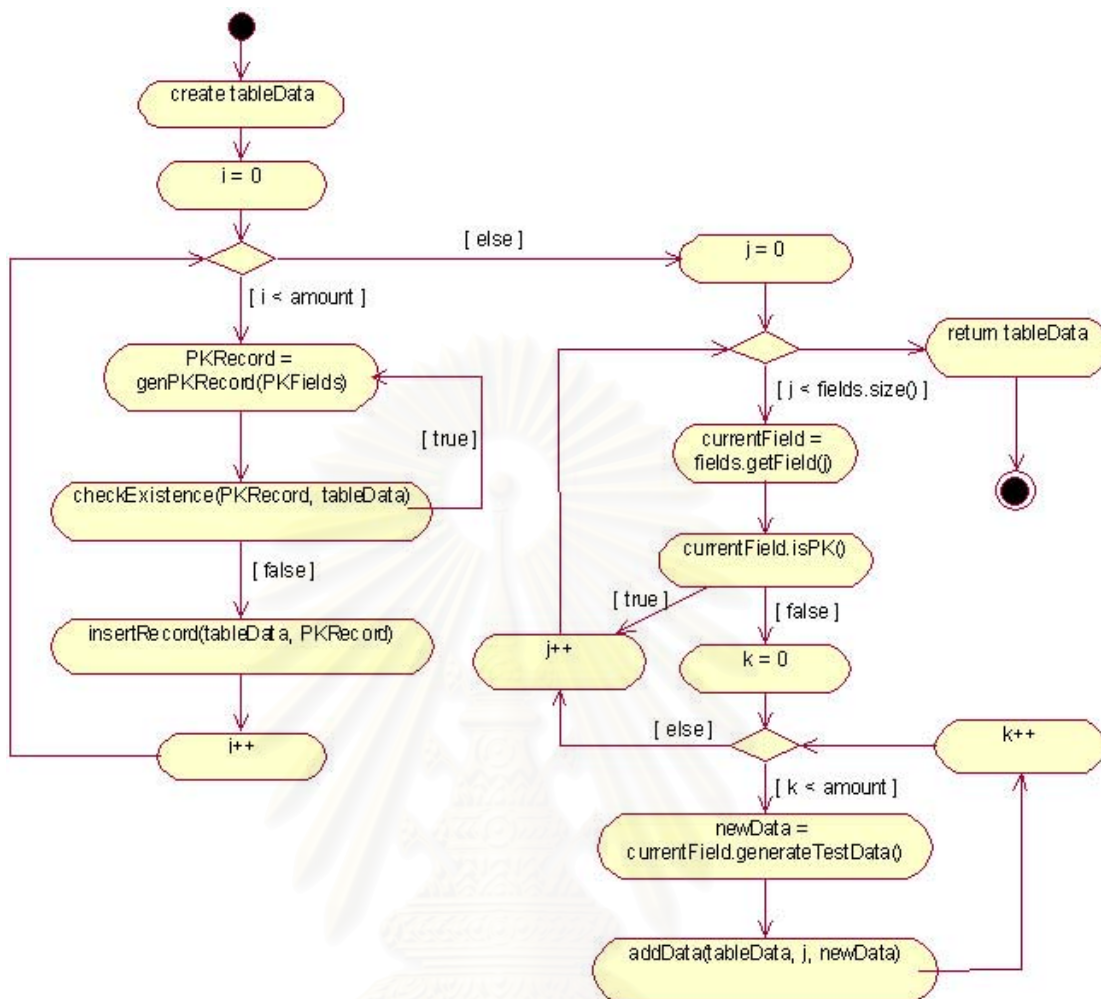
รูปที่ 3.33 แผนภาพลำดับกิจกรรมของเมทอด Table.generateTestData()

จากรูปที่ 3.33 การทำงานของเมทอด generateTestData() จะเริ่มจากการค้นหาเขตข้อมูลที่เป็นคีย์หลักของตารางข้อมูล (getPKFields()) เพื่อแยกออกมาเป็นรายการของเขตข้อมูลคีย์หลักเก็บไว้ใน PKFields จากนั้นจะสร้างข้อมูลทดสอบตาม PKFields และจำนวนระเบียบข้อมูลที่กำหนดมาด้วยการเรียกเมทอด generateTableData(PKFields,

genAmt) แล้วเก็บข้อมูลทดสอบไว้ใน tableData ต่อมาจะตรวจสอบเงื่อนไขการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลที่กำหนดไว้กับ tableData โดยจะแบ่งการตรวจสอบเป็น 2 แบบดังนี้

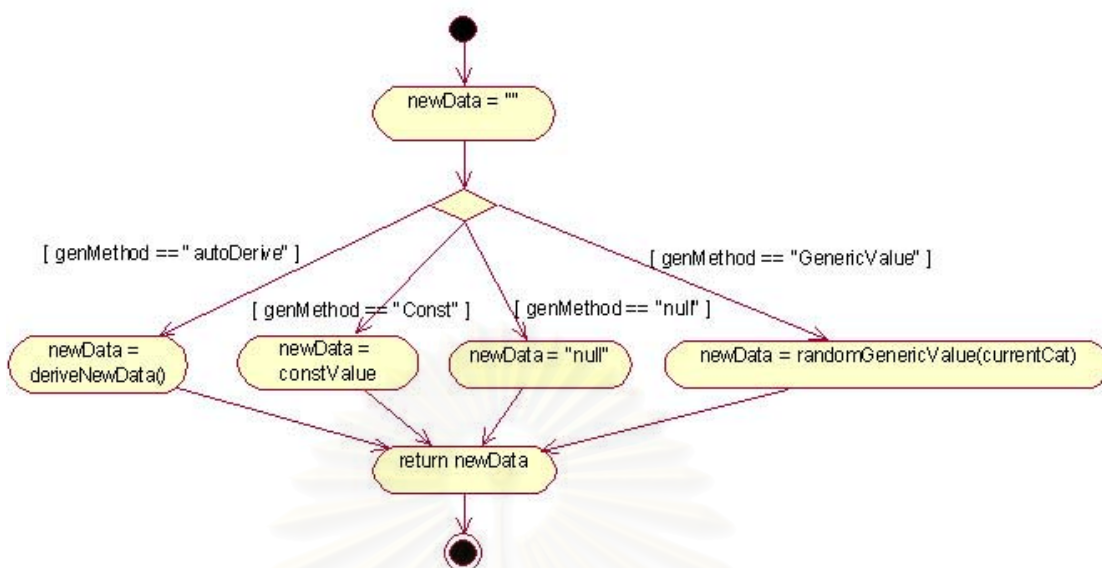
- 1) การตรวจสอบการขึ้นต่อกันเชิงตรรกะแบบปกติ หรือเมทอด validateNormalDep() จะตรวจสอบระเบียบข้อมูลใน tableData กับเงื่อนไขที่กำหนดไว้ที่ระเบียบข้อมูลจนครบทุกระเบียบ โดยการตรวจสอบในแต่ละเงื่อนไขจะตรวจสอบค่าข้อมูลเงื่อนไขในเขตข้อมูลต้นทางของเงื่อนไขกับค่าข้อมูลในเขตข้อมูลของระเบียบข้อมูลที่ตรงกับเขตข้อมูลต้นทางไปที่ระเบียบ ถ้าพบว่าค่าข้อมูลเงื่อนไขตรงกับค่าข้อมูลในเขตข้อมูลของระเบียบข้อมูลปัจจุบัน จึงจะเปลี่ยนค่าข้อมูลที่อยู่ในเขตข้อมูลที่ตรงกับเขตข้อมูลเป้าหมายของเงื่อนไขเป็นค่าผลลัพธ์ตามที่กำหนดไว้ และทำการตรวจสอบลักษณะนี้ไปจนครบทุกเงื่อนไข
- 2) การตรวจสอบการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล หรือเมทอด validateSQLDep() เป็นการตรวจสอบระเบียบข้อมูลใน tableData กับคำสั่งเอสคิวแอลที่กำหนดไว้ โดยนำระเบียบข้อมูลทั้งหมดบันทึกลงในฐานข้อมูล testDataDb จากนั้นทำการสอบถาม (Query) ระเบียบข้อมูลเหล่านี้ขึ้นมาโดยใช้คำสั่งเอสคิวแอลที่กำหนดไว้ เมื่อได้ระเบียบข้อมูลที่ผ่านการสอบถามขึ้นมา ก็จะตรวจสอบว่าจำนวนระเบียบข้อมูลที่ผ่านการสอบถามน้อยกว่าจำนวนระเบียบข้อมูลที่ต้องการสร้างให้ตารางข้อมูลนั้นหรือไม่ ถ้าน้อยกว่าจะสร้างข้อมูลทดสอบเพิ่มขึ้นทีละ 100 ระเบียบ แล้วทำการสอบถามระเบียบข้อมูลเหล่านี้โดยใช้คำสั่งเอสคิวแอลเดิมอีกครั้ง และทำกระบวนการในลักษณะนี้ไปจนกว่าจำนวนระเบียบข้อมูลที่ผ่านการตรวจสอบจะไม่น้อยกว่าจำนวนระเบียบข้อมูลที่ต้องการสร้าง

เมื่อได้ tableData ที่ผ่านการตรวจสอบการขึ้นต่อกันเชิงตรรกะมาแล้ว จะบันทึกข้อมูลทดสอบใน tableData ลงในฐานข้อมูล testDataDb (insertToTestDataDb()) เพื่อนำข้อมูลไปใช้ในภายหลัง และสุดท้ายจะเรียกเมทอด setGenerated(true) เพื่อกำหนดให้รู้ว่าตารางข้อมูลนี้ผ่านการสร้างข้อมูลทดสอบแล้ว สำหรับรายละเอียดการสร้างข้อมูลทดสอบของเมทอด generateTableData() จะเป็นดังรูปที่ 3.34



รูปที่ 3.34 แผนภาพลำดับกิจกรรมของเมทอด Table.generateTableData()

จากรูปที่ 3.34 การทำงานของเมทอด generateTableData() จะเริ่มจากการสร้าง tableData เพื่อเก็บข้อมูลทดสอบ จากนั้นจะสร้างข้อมูลให้ PKFields ก่อน โดยจะสร้างข้อมูลออกมาทีละระเบียน (genPKRecord(PKFields)) แล้วตรวจสอบระเบียนข้อมูลนั้นว่าซ้ำซ้อนกับระเบียนข้อมูลที่มีอยู่เดิมใน tableData หรือไม่ (checkExistence(PKRecord, tableData)) ถ้าซ้ำซ้อน จะทำการสร้างระเบียนข้อมูลใหม่จนกว่าจะไม่ซ้ำ ถ้าระเบียนข้อมูลไม่ซ้ำซ้อน จะเพิ่มระเบียนข้อมูลนั้นเข้าไปใน tableData (insertRecord(tableData, PKRecord)) และทำกระบวนการเดียวกันนี้จนได้จำนวนระเบียนข้อมูลครบตาม amount ที่ส่งมา ต่อมาจะสร้างข้อมูลให้กับเขตข้อมูลที่เหลือ (ที่ไม่ใช่คีย์หลัก) ในทำนองเดียวกันไปที่ละเขตข้อมูลจนครบตามจำนวนระเบียนข้อมูลและครบทุกเขตข้อมูล แล้วส่ง tableData กลับออกไปจากเมทอด สำหรับการสร้างข้อมูลทดสอบให้กับแต่ละเขตข้อมูลนั้นจะใช้เมทอด generateTestData() ของคลาส Field ซึ่งมีรายละเอียดการทำงานดังรูปที่ 3.35



รูปที่ 3.35 แผนภาพลำดับกิจกรรมของเมทอด Field.generateTestData()

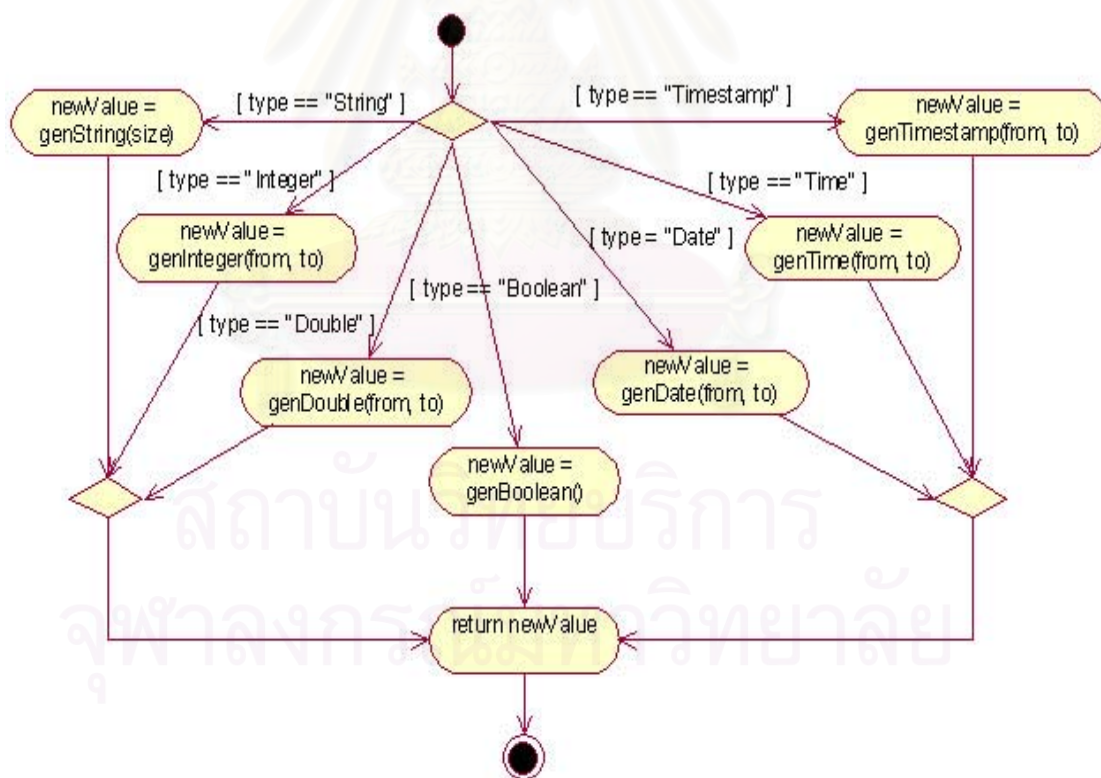
การสร้างข้อมูลให้กับแต่ละเซตข้อมูล จะเริ่มจากการตรวจสอบว่าเซตข้อมูลนั้นถูกกำหนดให้สร้างข้อมูลทดสอบด้วยวิธีใด ซึ่งจะได้ข้อมูลทดสอบที่แตกต่างกันดังนี้

- 1) ถ้ากำหนดให้เครื่องมือสุ่มสร้างให้โดยอัตโนมัติ จะใช้เมทอด deriveNewData() ซึ่งจะสุ่มสร้างข้อมูลทดสอบตามคุณสมบัติของเซตข้อมูลและช่วงที่กำหนดมา ดังรูปที่ 3.36
- 2) ถ้ากำหนดให้ใช้ค่าคงที่ จะนำเอาค่าคงที่ที่ผู้ใช้กำหนดมาเป็นข้อมูลทดสอบใหม่ทันที
- 3) ถ้ากำหนดให้เป็นค่าว่าง จะกำหนดค่าว่างเป็นข้อมูลทดสอบใหม่ทันที
- 4) ถ้าเลือกใช้กลุ่มค่าทั่วไป จะสุ่มค่าจากกลุ่มค่าทั่วไปที่ผู้ใช้เลือกไว้มา กำหนดเป็นข้อมูลทดสอบใหม่

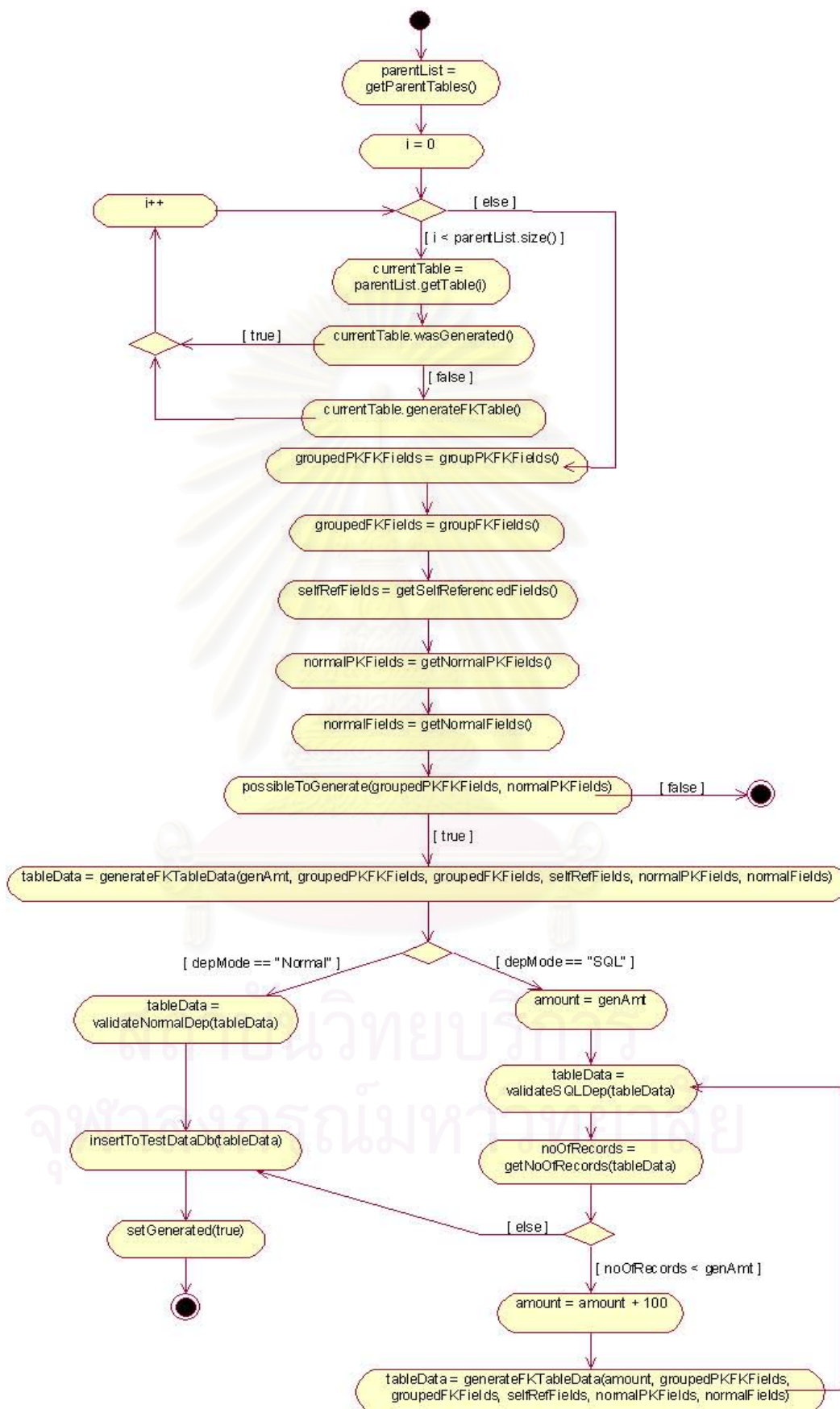
2.2 แผนภาพลำดับกิจกรรมของเมทอด generateFKTable() สามารถแสดงได้ดังรูปที่ 3.37 การทำงานของเมทอด generateFKTable() จะเริ่มจากการค้นหาตารางข้อมูลที่มีเซตข้อมูลที่ถูกอ้างอิงถึงโดยเซตข้อมูลของตารางข้อมูลปัจจุบันที่กำลังจะสร้างข้อมูลทดสอบ (getParentTables()) เก็บไว้เป็นรายการใน parentList ซึ่งตารางข้อมูลที่ถูกอ้างอิงถึงจะต้องอยู่ในรายการตารางข้อมูลที่สร้างข้อมูลทดสอบและไม่ใช่ตารางข้อมูลปัจจุบันด้วย จากนั้นจะทำการสร้างข้อมูลทดสอบให้กับตารางข้อมูลที่อยู่ใน parentList ก่อน โดยจะสร้างข้อมูลให้เฉพาะ

ตารางข้อมูลที่ยังไม่ได้สร้างข้อมูลทดสอบเท่านั้น ต่อมาจะแบ่งรายการเขตข้อมูลของตารางข้อมูลปัจจุบันออกเป็น 5 รายการดังนี้

- 1) groupedPKFKFields คือรายการเขตข้อมูลที่เป็นทั้งคีย์หลักและคีย์อ้างอิงที่จัดกลุ่มย่อยตามตารางข้อมูลที่ถูกอ้างอิงถึงโดยแต่ละเขตข้อมูลในรายการ
- 2) groupedFKFields คือรายการเขตข้อมูลที่เป็นคีย์อ้างอิงที่จัดกลุ่มย่อยตามตารางข้อมูลที่ถูกอ้างอิงถึงโดยแต่ละเขตข้อมูลในรายการเหมือนกับ groupedPKFKFields
- 3) selfRefFields คือรายการเขตข้อมูลที่เป็นคีย์อ้างอิงที่อ้างอิงถึงเขตข้อมูลภายในตารางข้อมูลปัจจุบันเอง
- 4) normalPKFields คือรายการเขตข้อมูลที่เป็นคีย์หลักของตารางข้อมูล
- 5) normalFields คือรายการเขตข้อมูลปกติที่ไม่ได้เป็นทั้งคีย์หลักหรือคีย์อ้างอิง

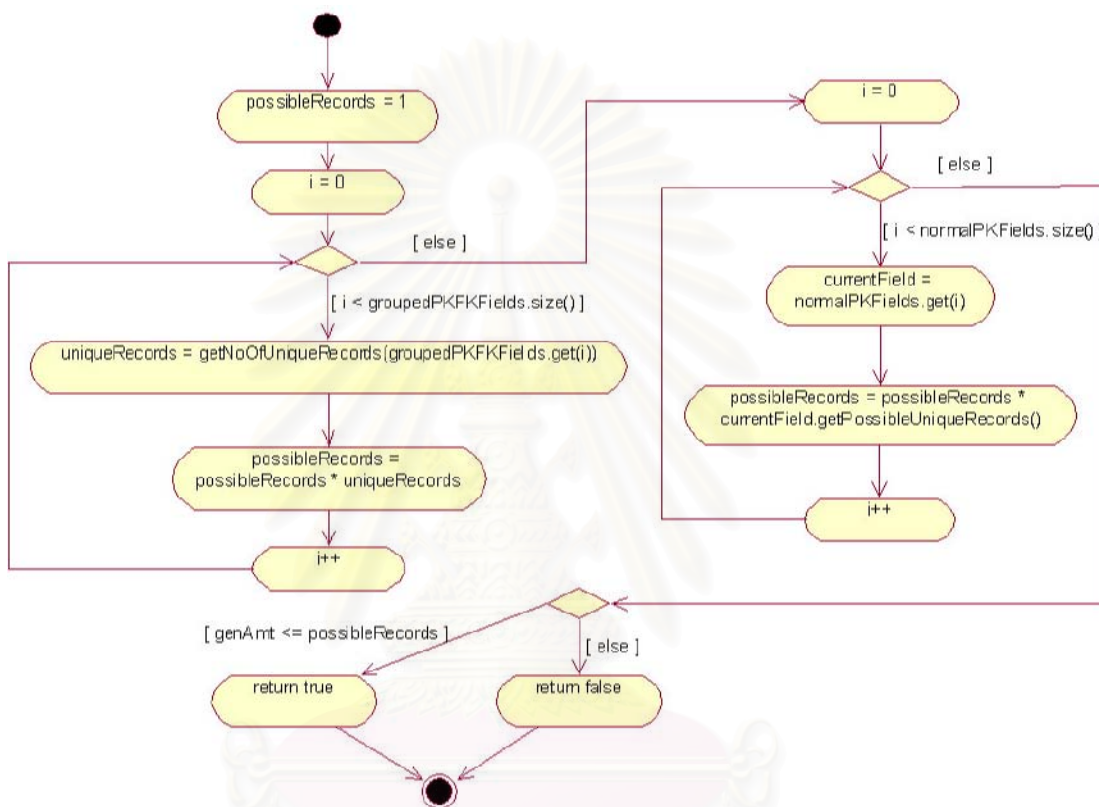


รูปที่ 3.36 แผนภาพลำดับกิจกรรมของเมทอด Field.deriveNewData()



รูปที่ 3.37 แผนภาพลำดับกิจกรรมของเมทอด `Table.generateFKTable()`

เมื่อแบ่งรายการเขตข้อมูลเสร็จแล้ว จะตรวจสอบว่าสามารถสร้างข้อมูลทดสอบให้กับตารางข้อมูลปัจจุบันได้ตามจำนวนระเบียบข้อมูลที่ใช้กำหนดมาได้หรือไม่ โดยเรียกใช้เมทอด `possibleToGenerate()` ถ้าพบว่าตารางข้อมูลปัจจุบันไม่สามารถสร้างข้อมูลทดสอบได้ เครื่องมือจะหยุดกระบวนการสร้างข้อมูลทดสอบทันที รายละเอียดของเมทอด `possibleToGenerate()` จะเป็นดังรูปที่ 3.38

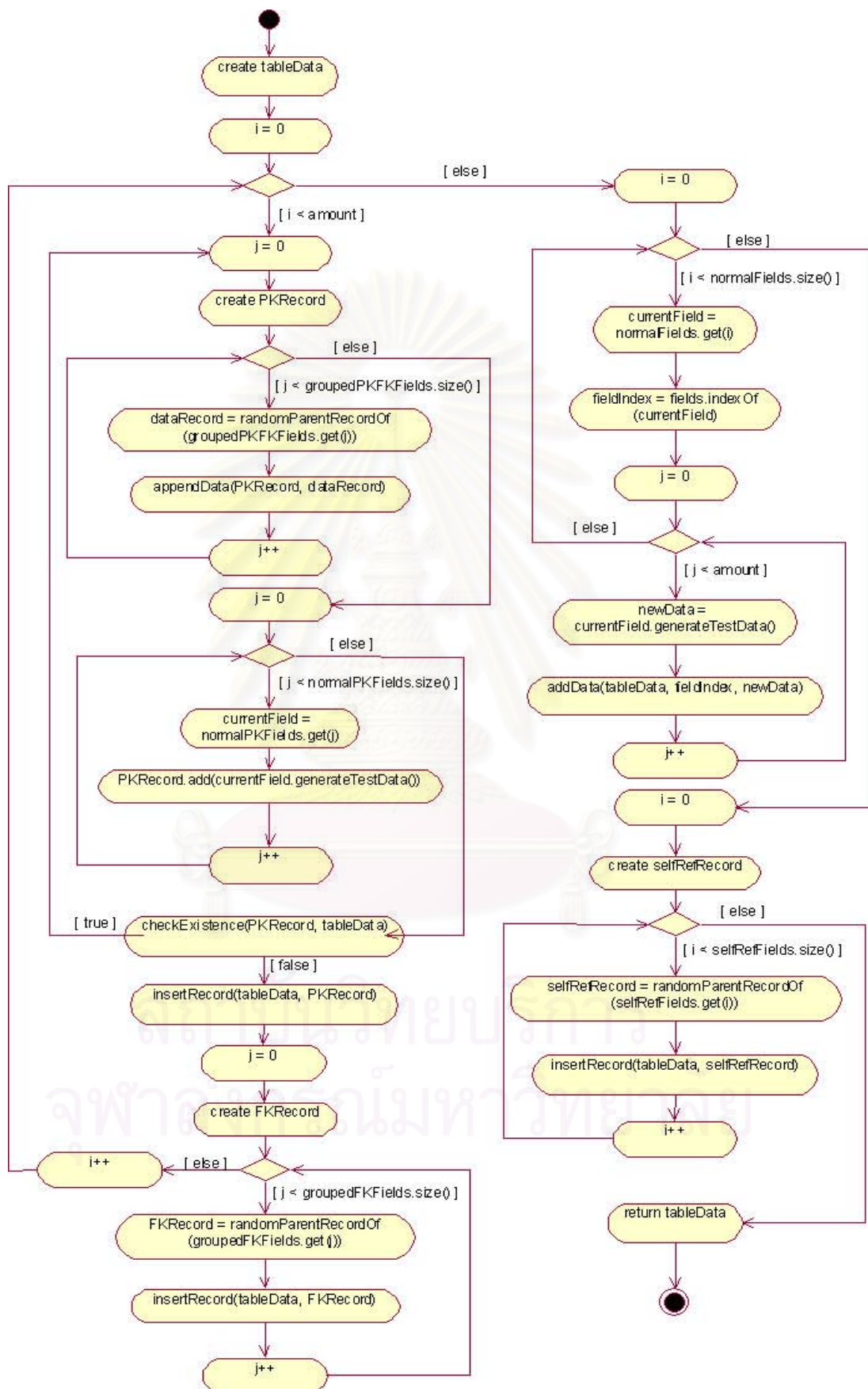


รูปที่ 3.38 แผนภาพลำดับกิจกรรมของเมทอด `possibleToGenerate()`

จากรูปที่ 3.38 การตรวจสอบว่าตารางปัจจุบันสามารถสร้างข้อมูลทดสอบได้หรือไม่นั้น จะเริ่มจากการคำนวณจำนวนระเบียบข้อมูลที่เป็นไปได้ในการสร้างข้อมูลทดสอบ (`possibleRecords`) ของตารางข้อมูล โดยการนำเอาจำนวนระเบียบข้อมูลที่เป็นไปได้ของเขตข้อมูลที่เป็นคีย์หลักมาคูณกันซึ่งเขตข้อมูลที่เป็นคีย์หลักของตารางข้อมูลปัจจุบันนั้นแบ่งเป็น 2 กลุ่มคือ `groupedPKFKFields` และ `normalPKFields` ดังนั้นในขั้นแรกจะทำการหาจำนวนระเบียบข้อมูลที่ไม่ซ้ำกัน (`getNoOfUniqueRecords()`) ของแต่ละตารางข้อมูลที่ถูกอ้างอิงถึงโดยแต่ละรายการย่อยของเขตข้อมูลใน `groupedPKFKFields` มาคูณกันเก็บไว้ใน `possibleRecords` ก่อน จากนั้นจะทำการหาจำนวนระเบียบข้อมูลที่ไม่ซ้ำกันที่เป็นไปได้ (`getPossibleUniqueRecords()`) ของแต่ละเขตข้อมูลในรายการ `normalPKFields` มาคูณกับ `possibleRecords` ก็จะได้จำนวนระเบียบข้อมูลที่เป็นไปได้ของตารางข้อมูลปัจจุบัน สุดท้ายจะ

ตรวจสอบว่าจำนวนระเบียบข้อมูลที่เป็นไปได้ที่คำนวณได้นั้นน้อยกว่าหรือเท่ากับจำนวน
 ระเบียบข้อมูลที่ใช้กำหนดมาหรือไม่ ถ้าใช่แสดงว่าตารางข้อมูลปัจจุบันสามารถสร้างข้อมูล
 ทดสอบได้ แต่ถ้าไม่ใช่แสดงว่าตารางข้อมูลปัจจุบันจะไม่สามารถสร้างข้อมูลทดสอบได้

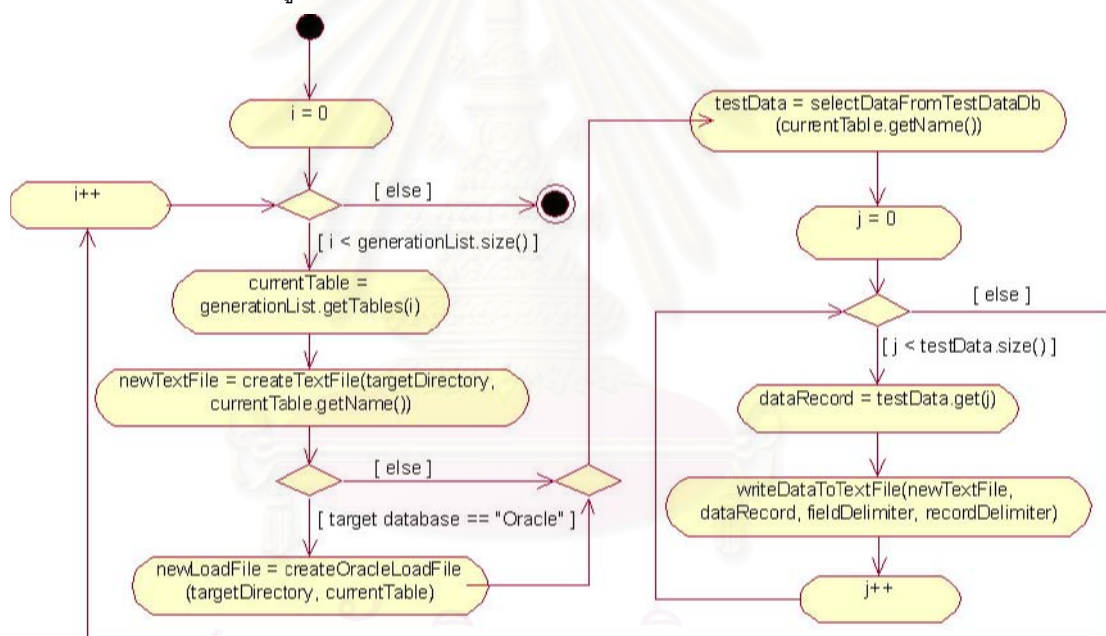
เมื่อผ่านการตรวจสอบที่กล่าวมาแล้ว จึงจะเริ่มการสร้างข้อมูลทดสอบ
 ให้กับตารางข้อมูลปัจจุบัน จากรูปที่ 3.37 การสร้างข้อมูลทดสอบทำได้โดยเรียกใช้เมธอด
`generateFKTableData()` แล้วเก็บข้อมูลทดสอบไว้ใน `tableData` จากนั้นจะตรวจสอบเงื่อนไขการ
 ขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลที่กำหนดไว้กับ `tableData` แล้วบันทึกข้อมูลทดสอบลงใน
 ฐานข้อมูล `testDataDb` ซึ่งกระบวนการเหล่านี้จะเหมือนกับกระบวนการที่กล่าวมาแล้วในหัวข้อที่
 2.1 (แผนภาพลำดับกิจกรรมของเมธอด `generateTestData()`) สำหรับรายละเอียดการทำงานของ
 เมธอด `generateFKTableData()` จะเป็นดังรูป 3.39 การทำงานของเมธอด
`generateFKTableData()` จะเริ่มจากการสร้าง `tableData` เพื่อเก็บข้อมูลทดสอบ จากนั้นจะเป็น
 กระบวนการสร้างระเบียบข้อมูลสำหรับเขตข้อมูลที่เป็นคีย์หลักก่อนคือ `groupedPKFKFields`
 และ `normalPKFields` โดยระเบียบข้อมูลที่จะสร้างขึ้นใหม่จะเก็บไว้ใน `PKRecord` การสร้างข้อมูลให้กับ
`groupedPKFKFields` จะเป็นการสุ่มระเบียบข้อมูลมาจากแต่ละกลุ่มของเขตข้อมูลที่ถูกอ้างอิงถึง
 โดยแต่ละรายการย่อยของเขตข้อมูลใน `groupedPKFKFields` โดยใช้เมธอด
`randomParentRecordOf()` เมื่อได้ระเบียบข้อมูลมาก็นำไปต่อท้ายข้อมูลใน `PKRecord` ต่อมา
 จะเป็นการสร้างข้อมูลให้กับแต่ละเขตข้อมูลใน `normalPKFields` โดยใช้เมธอด
`generateTestData()` ของคลาส `Field` (ดังรูปที่ 3.35) แล้วนำมาต่อท้าย `PKRecord` เช่นกัน เมื่อ
 ได้ `PKRecord` ที่ครบสมบูรณ์ จึงจะตรวจสอบ `PKRecord` ว่าซ้ำซ้อนกับระเบียบข้อมูลที่มีอยู่เดิม
 ใน `tableData` หรือไม่ (`checkExistence(PKRecord, tableData)`) ถ้าซ้ำซ้อน จะสร้าง `PKRecord`
 ใหม่จนกว่าจะไม่ซ้ำ แต่ถ้า `PKRecord` ไม่ซ้ำซ้อน ก็จะเพิ่ม `PKRecord` เข้าไปใน `tableData`
 (`insertRecord(tableData, PKRecord)`) ต่อมาจะเป็นกระบวนการสร้างระเบียบข้อมูลสำหรับเขต
 ข้อมูลที่เป็นคีย์อ้างอิงคือ `groupedFKFields` โดยระเบียบข้อมูลที่จะสร้างขึ้นใหม่จะเก็บไว้ใน `FKRecord`
 การสร้างข้อมูลให้กับ `groupedFKFields` จะเป็นการสุ่มระเบียบข้อมูลมาจากแต่ละกลุ่มของเขต
 ข้อมูลที่ถูกอ้างอิงถึงโดยแต่ละรายการย่อยของเขตข้อมูลใน `groupedFKFields` โดยใช้เมธอด
`randomParentRecordOf()` เหมือนกับการสร้างระเบียบข้อมูลของ `groupedPKFKFields`
 หลังจากนั้น `FKRecord` ใหม่ที่สร้างมาจะถูกเพิ่มเข้าไปใน `tableData` กระบวนการสร้าง
 ระเบียบข้อมูลทั้งหมดที่กล่าวมาจะถูกทำไปจนกว่าจะได้จำนวนระเบียบข้อมูลครบตาม `amount` ที่
 ส่งมา



รูปที่ 3.39 แผนภาพลำดับกิจกรรมของเมทอด Table.generateFKTableData()

จากนั้นจะทำการสร้างระเบียบข้อมูลให้กับเขตข้อมูลปกติใน normalFields จนครบตามจำนวน amount และครบทุกเขตข้อมูล และสุดท้ายจะทำการสร้างระเบียบข้อมูลให้กับเขตข้อมูลใน selfRefFields โดยระเบียบข้อมูลที่สร้างใหม่จะเก็บไว้ใน selfRefRecord การสร้างข้อมูลให้กับ selfRefFields จะใช้เมทอด randomParentRecordOf() สุ่มระเบียบข้อมูลมาจากแต่ละกลุ่มของเขตข้อมูลที่ถูกอ้างอิงถึงโดยแต่ละรายการย่อยของเขตข้อมูลใน selfRefFields สำหรับ selfRefRecord ที่ได้มาจะถูกเพิ่มเข้าไปใน tableData แล้ววนสร้าง selfRefRecord จนครบตามจำนวน amount จึงถือเป็นการสิ้นสุดการสร้างข้อมูล พร้อมทั้งส่ง tableData กลับออกจากเมทอด

3. แผนภาพลำดับกิจกรรมของยูสเคสสร้างเพิ่มข้อความที่มีตัวค้น แผนภาพลำดับกิจกรรมจะเป็นดังรูปที่ 3.40

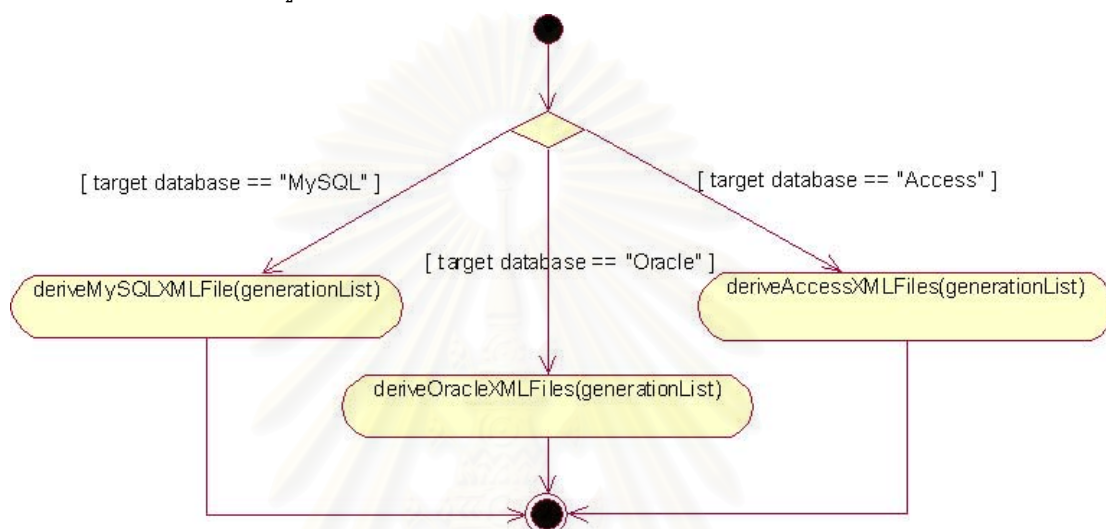


รูปที่ 3.40 แผนภาพลำดับกิจกรรมแสดงการสร้างเพิ่มข้อความที่มีตัวค้น

จากรูปที่ 3.40 การสร้างเพิ่มข้อความที่มีตัวค้น จะเริ่มจากการสร้างเพิ่มข้อความเปล่าๆ (createTextFile()) ขึ้นมาก่อน เพื่อเตรียมไว้สำหรับเขียนข้อมูลทดสอบลงไป การสร้างเพิ่มข้อความจะสร้างที่ไดเรกทอรี (Directory) เป้าหมายที่กำหนดมา และตั้งชื่อเพิ่มตามชื่อตารางข้อมูลที่เป็นเจ้าของข้อมูลทดสอบ แต่ถ้าฐานข้อมูลเป้าหมายที่กำหนดมาเป็นฐานข้อมูลออราเคิลจะต้องมีการสร้างโหลดไฟล์ (createOracleLoadFile()) ซึ่งเป็นเพิ่มข้อความที่กำหนดคุณสมบัติการโหลดข้อมูลเข้าสู่ตารางในฐานข้อมูลด้วย เมื่อสร้างเพิ่มข้อความเสร็จแล้ว จะทำการดึงข้อมูลทดสอบของตารางข้อมูลขึ้นมาจากฐานข้อมูล testDataDb แล้วนำข้อมูลทดสอบมาเขียน

ลงในแฟ้มข้อความที่ละระเบียน (writeDataToTextFile()) โดยคั่นแต่ละเขตข้อมูลในระเบียนข้อมูลด้วยตัวคั่นระหว่างเขตข้อมูลและคั่นแต่ละระเบียนด้วยตัวคั่นระหว่างระเบียนข้อมูลที่ใช้กำหนดมา และจะทำกระบวนการนี้ไปจนครบทุกตารางข้อมูลในรายการ generationList

4. แผนภาพลำดับกิจกรรมของยูสเคสสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอล แผนภาพลำดับกิจกรรมจะเป็นดังรูปที่ 3.41



รูปที่ 3.41 แผนภาพลำดับกิจกรรมแสดงการสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอล

จากรูปที่ 3.41 การสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลจะมีการตรวจสอบฐานข้อมูลเป้าหมายที่ผู้ใช้กำหนดมาก่อน ซึ่งการสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลของแต่ละฐานข้อมูลนั้นจะเรียกใช้เมทอดต่างกันคือ ฐานข้อมูลมายเอสคิวแอลใช้เมทอด deriveMySQLXMLFile() ฐานข้อมูลออราเคิลใช้เมทอด deriveOracleXMLFiles() และฐานข้อมูลไมโครซอฟท์แอ็คเซสใช้เมทอด deriveAccessXMLFiles() สำหรับรายละเอียดการทำงานของแต่ละเมทอดจะเป็นดังรูปที่ 3.42 3.43 และ 3.44 ตามลำดับ

จากรูปที่ 3.42 การสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลของฐานข้อมูลมายเอสคิวแอล จะเริ่มจากการสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลเปล่าเตรียมรับข้อมูลทดสอบ โดยตั้งชื่อแฟ้มเอกสารตามชื่อสคีมาของฐานข้อมูลที่สร้างข้อมูลทดสอบ จากนั้นในส่วนการบันทึกข้อมูลทดสอบที่ดึงขึ้นมาจากฐานข้อมูล testDataDb เครื่องมือจะแปลงข้อมูลทดสอบให้อยู่ในรูปแบบของอีลีเมนต์ต่างๆ ดังนี้

- อีลีเมนต์ที่แทนสคีมามาของฐานข้อมูล จะเป็นอีลีเมนต์ที่มีชื่อเหมือนกับชื่อของสคีมามาของฐานข้อมูลเขียนแทนด้วย <ชื่อสคีมามาของฐานข้อมูล> ซึ่งอีลีเมนต์ที่แทนสคีมามาของฐานข้อมูลจะถือเป็นอีลีเมนต์รากของแฟ้มเอกสารเอ็กซ์เอ็มแอล

- อีลีเมนต์ที่แทนตารางข้อมูล จะเป็นอีลีเมนต์ที่ซ่อนอยู่ภายในอีลีเมนต์ที่แทนสคีมามาของฐานข้อมูล โดยจะมีชื่อเหมือนกับชื่อของแต่ละตารางข้อมูลในรายการ generationList เขียนแทนด้วย <ชื่อตารางข้อมูล>

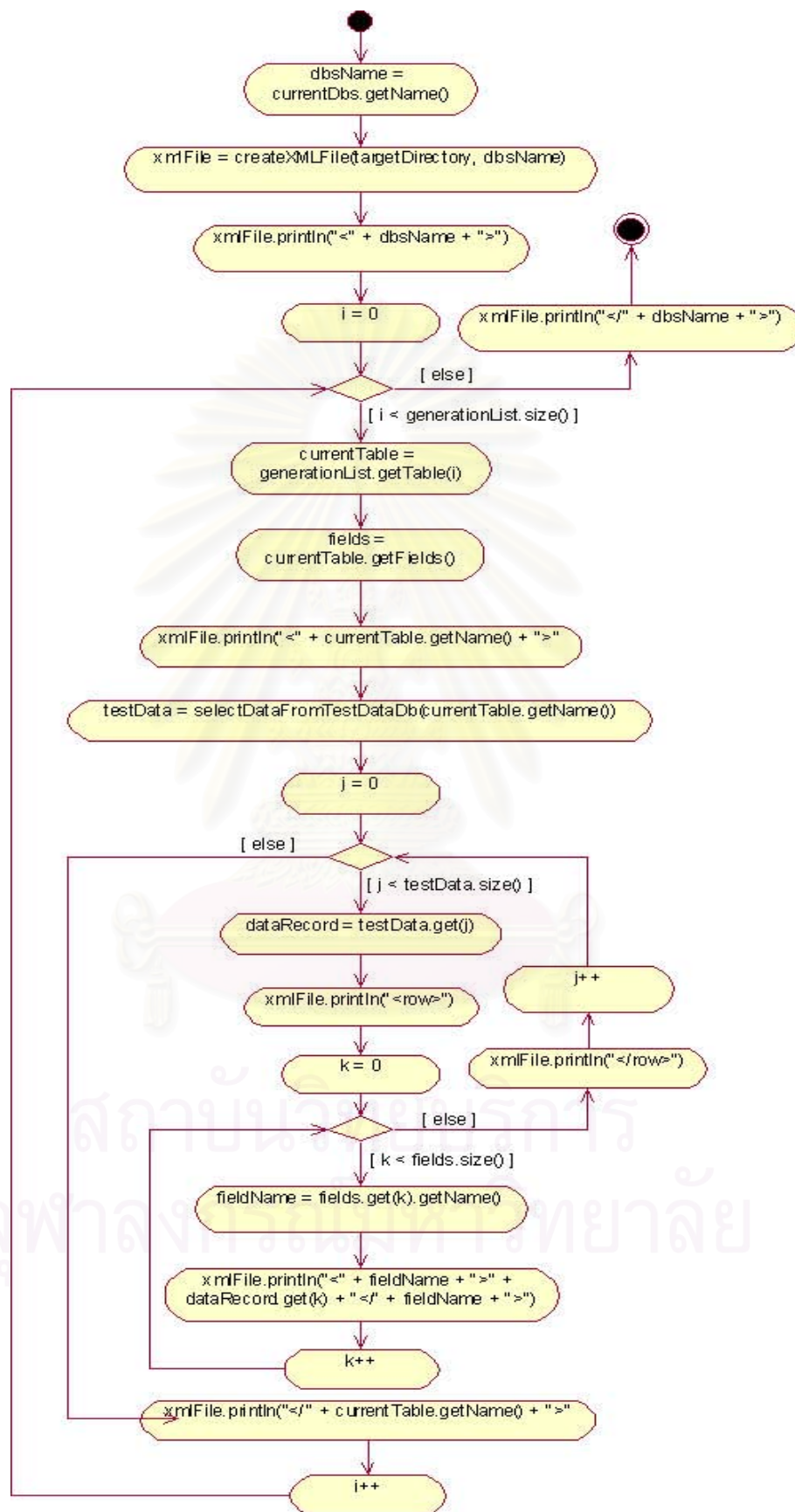
- อีลีเมนต์ที่แทนระเบียบข้อมูล จะเป็นอีลีเมนต์ที่ซ่อนอยู่ภายในอีลีเมนต์ที่แทนตารางข้อมูลที่จะแทนแต่ละระเบียบในตารางข้อมูลหนึ่งๆ โดยอีลีเมนต์มีชื่อว่า "row" เขียนแทนด้วย <row>

- อีลีเมนต์ที่แทนเขตข้อมูล จะเป็นอีลีเมนต์ที่ซ่อนอยู่ภายในอีลีเมนต์ที่แทนระเบียบข้อมูล โดยจะมีชื่อเหมือนกับชื่อของแต่ละเขตข้อมูลภายในตารางข้อมูลนั้นๆ เขียนแทนด้วย <ชื่อเขตข้อมูล>

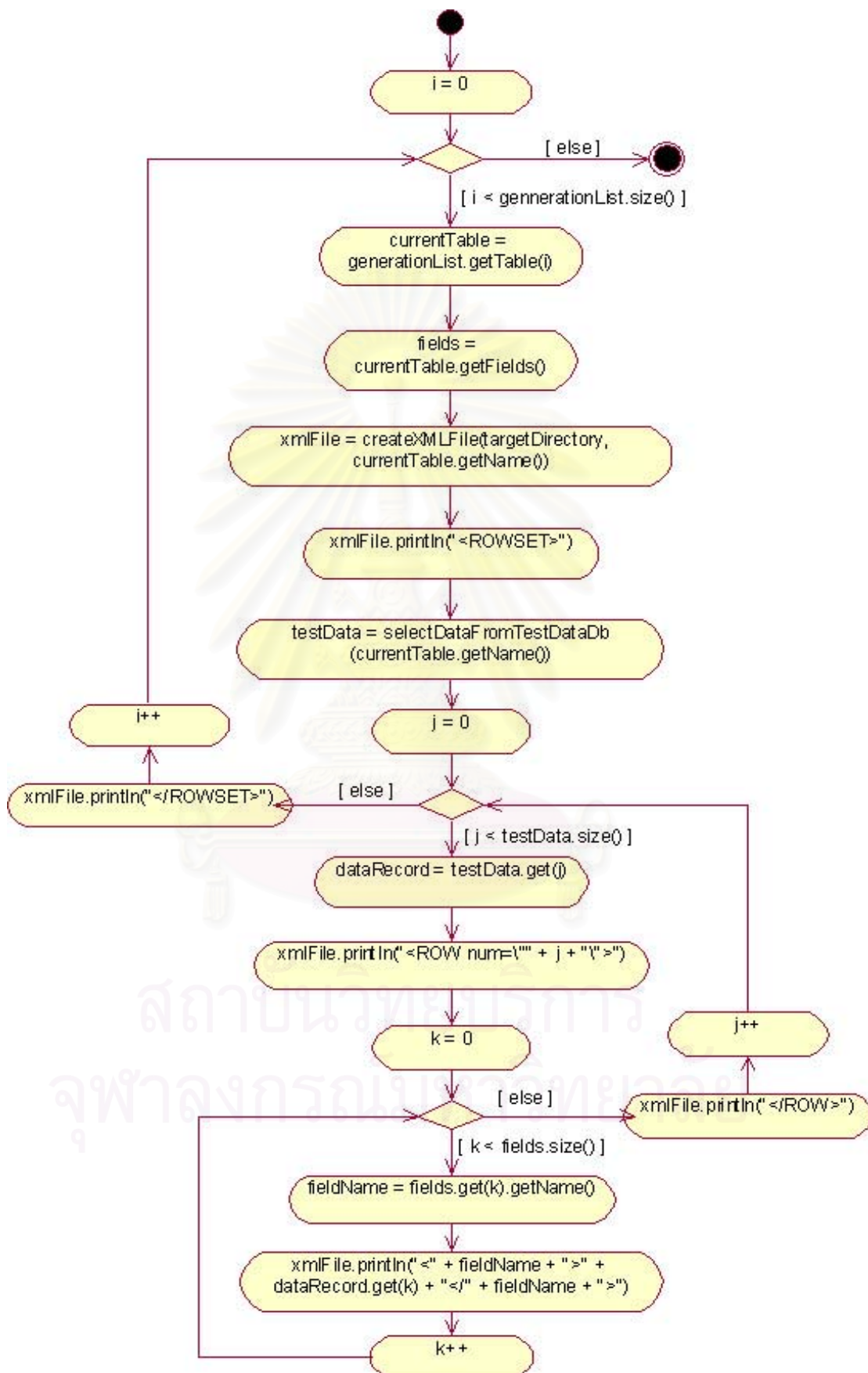
- เนื้อหาภายในอีลีเมนต์ที่แทนเขตข้อมูล จะเป็นค่าข้อมูลทดสอบที่ปรากฏในแต่ละเขตข้อมูลของระเบียบข้อมูลนั้นๆ การเขียนจะครอบค่าข้อมูลด้วยอีลีเมนต์ที่แทนเขตข้อมูลดังนี้ <ชื่อเขตข้อมูล>ค่าข้อมูล</ชื่อเขตข้อมูล>

จากรูปที่ 3.43 การสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลของฐานข้อมูลออราเคิลจะแตกต่างจากฐานข้อมูลมายเอสคิวแอลคือจะได้แฟ้มเอกสารเอ็กซ์เอ็มแอลออกมาตารางละ 1 แฟ้ม แต่สำหรับฐานข้อมูลมายเอสคิวแอลจะได้แฟ้มเอกสารเอ็กซ์เอ็มแอลออกมาเพียง 1 แฟ้มเท่านั้น ในการสร้างแต่ละแฟ้มเอกสารเอ็กซ์เอ็มแอลนั้น จะเริ่มจากการสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลเปล่าเตรียมรับข้อมูลทดสอบ โดยตั้งชื่อแฟ้มเอกสารตามชื่อตารางข้อมูลที่เป็นเจ้าของข้อมูลทดสอบ จากนั้นจะบันทึกข้อมูลทดสอบที่ดึงขึ้นมาฐานข้อมูล testDataDb ลงในแฟ้ม โดยอีลีเมนต์รากที่ใช้จะถือเป็นอีลีเมนต์ชื่อ "ROWSET" เขียนแทนด้วย <ROWSET> ในส่วนของระเบียบข้อมูลจะเขียนแทนด้วยอีลีเมนต์ <ROW num="ลำดับของระเบียบข้อมูล"> สำหรับค่าข้อมูลทดสอบนั้นจะเขียนแทนด้วยอีลีเมนต์แบบเดียวกับแฟ้มเอกสารเอ็กซ์เอ็มแอลของฐานข้อมูลมายเอสคิวแอลคือ <ชื่อเขตข้อมูล>ค่าข้อมูล</ชื่อเขตข้อมูล>

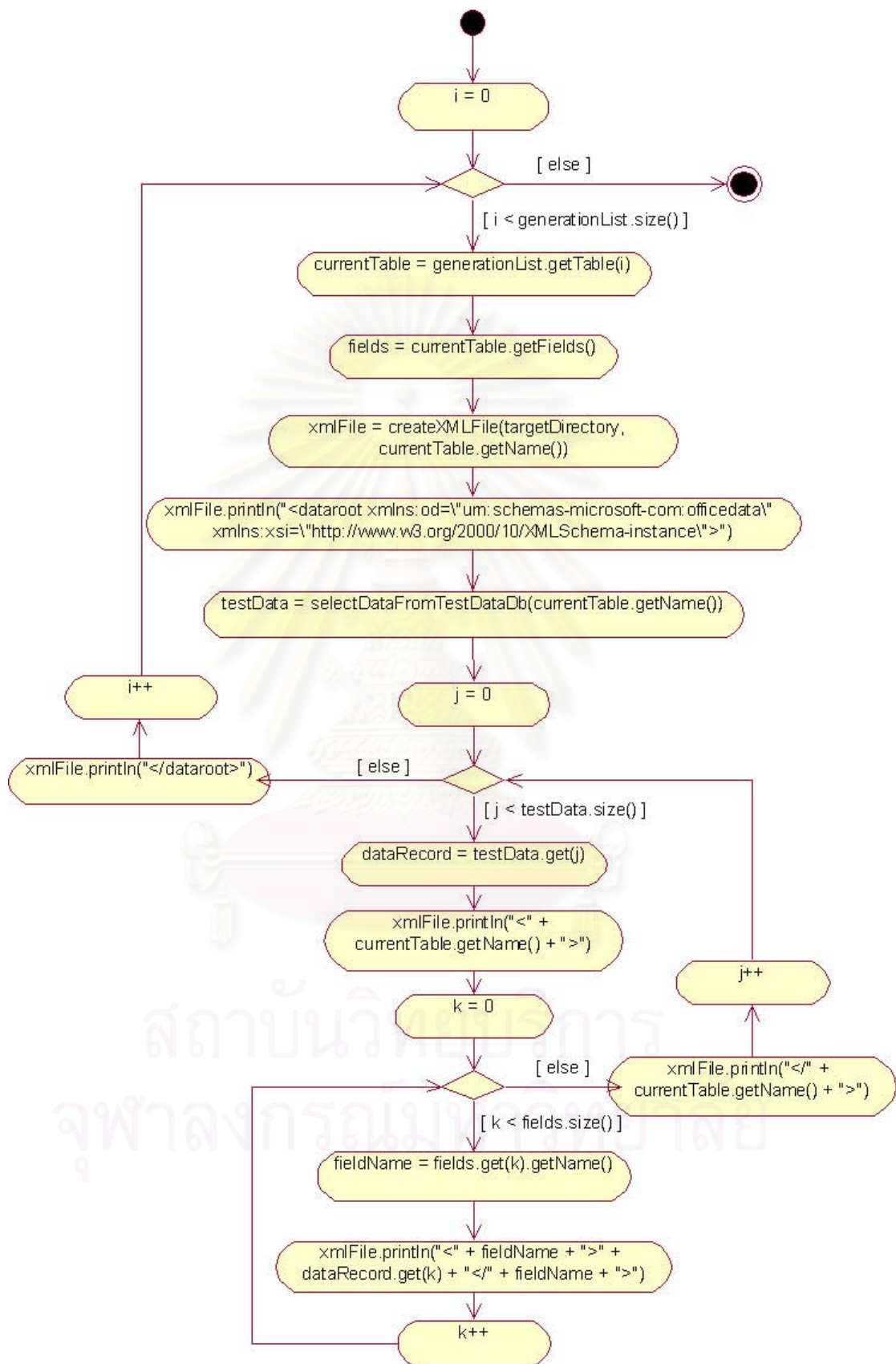
จากรูปที่ 3.44 การสร้างแฟ้มเอกสารเอ็กซ์เอ็มแอลสำหรับฐานข้อมูลไมโครซอฟท์แอ็คเซสชันนั้นเหมือนกับฐานข้อมูลออราเคิล แต่ฐานข้อมูลไมโครซอฟท์แอ็คเซสชันจะใช้อีลีเมนต์ <dataroot> เป็นอีลีเมนต์ราก และเปลี่ยนอีลีเมนต์ที่แทนแต่ละระเบียบข้อมูลไปใช้อีลีเมนต์ที่มีชื่อเหมือนกับตารางข้อมูลแทนคือ <ชื่อตารางข้อมูล>



รูปที่ 3.42 แผนภาพลำดับกิจกรรมของเมทอด deriveMySQLXMLFile()



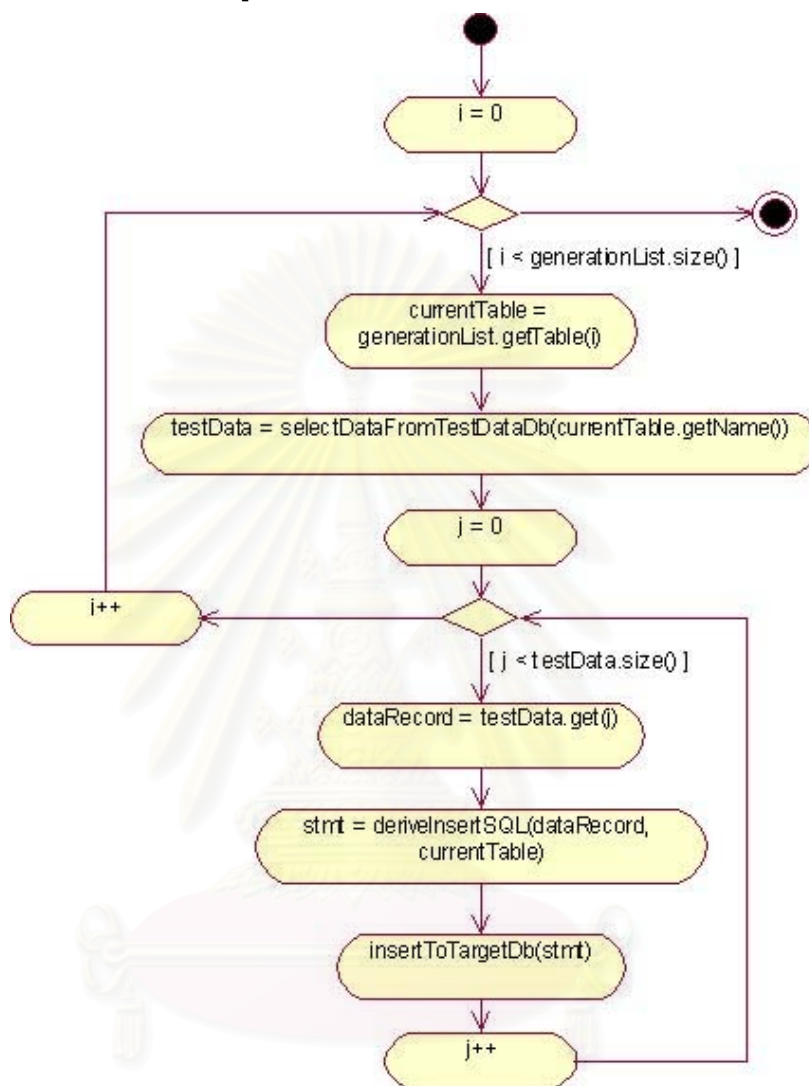
รูปที่ 3.43 แผนภาพลำดับกิจกรรมของเมทอด deriveOracleXMLFiles()



รูปที่ 3.44 แผนภาพลำดับกิจกรรมของเมทอด `deriveAccessXMLFiles()`

5. แผนภาพลำดับกิจกรรมของยูสเคสเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมาย

แผนภาพลำดับกิจกรรมจะเป็นดังรูปที่ 3.45



รูปที่ 3.45 แผนภาพลำดับกิจกรรมแสดงการเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมาย

จากรูปที่ 3.45 การเพิ่มข้อมูลเข้าสู่ฐานข้อมูลเป้าหมายจะทำไปที่ตารางในรายการ generationList โดยขั้นแรกจะดึงข้อมูลทดสอบขึ้นมาจากฐานข้อมูล testDataDb เก็บไว้ใน testData จากนั้นจะทำการสร้างคำสั่งเอสคิวแอลแบบแทรกข้อมูล (deriveInsertSQL()) ให้กับระเบียนข้อมูลใน testData ทีละระเบียน พร้อมทั้งนำคำสั่งนั้นไปประมวลผลเพื่อเพิ่มข้อมูลต่อท้าย (Append) ข้อมูลที่มีอยู่ในตารางข้อมูลภายในฐานข้อมูลเป้าหมาย (insertToTargetDb()) และกระบวนการนี้ไปจนครบทุกระเบียนข้อมูลและตารางข้อมูลใน generationList

บทที่ 4

การพัฒนาเครื่องมือ

การอธิบายการพัฒนาเครื่องมือจะแบ่งเป็น 3 ส่วนดังนี้

4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ

สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือมีรายละเอียดดังต่อไปนี้

1. ฮาร์ดแวร์ (Hardware)

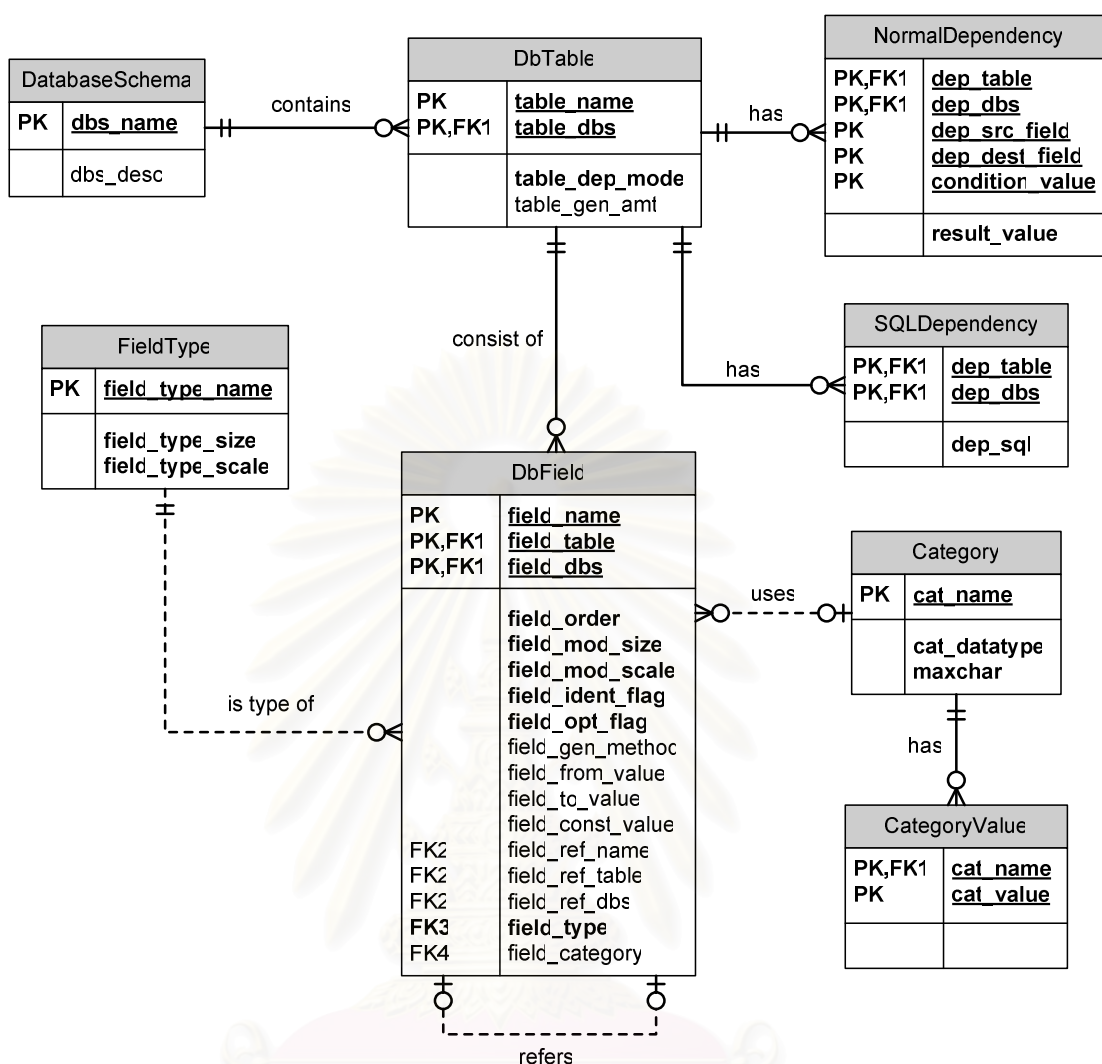
- เครื่องคอมพิวเตอร์แบบพีซี (PC) หน่วยประมวลผลอินเทลเพนเทียมโฟร์ 3.0 กิกะเฮิรท์ซ (Intel Pentium IV 3.0c GHz)
- หน่วยความจำสำรอง (RAM) 1 กิกะไบต์ (1 GB)
- ฮาร์ดดิสก์ (Harddisk) 120 กิกะไบต์ (120 GB)

2. ซอฟต์แวร์ (Software)

- ระบบปฏิบัติการ (Operating system) ไมโครซอฟท์วินโดวส์เอ็กซ์พีโพรเฟชันแนล (Microsoft Windows XP Professional)
- เจ2เอสดีเค เวอร์ชัน 1.4.2_01 (J2SDK 1.4.2_01)
- ฟอर्टเตฟอว์จาวา เวอร์ชัน 3.0 (Forte for Java 3.0 Community Edition)
- ฐานข้อมูลมายเอสคิวแอล เวอร์ชัน 4.0.18 (MySQL database 4.0.18)
- ฐานข้อมูลออราเคิล เวอร์ชัน 8.1.7 (Oracle8i 8.1.7)
- ฐานข้อมูลไมโครซอฟท์แอ็กเซส 2003 (Microsoft Access 2003)

4.2 โครงสร้างฐานข้อมูลของเครื่องมือ

โครงสร้างของฐานข้อมูลของเครื่องมือจะอธิบายด้วยแผนภาพความสัมพันธ์ระหว่างเอนทิตีระดับกายภาพ (Physical Entity-Relationship Diagram) ดังรูปที่ 4.1 เอนทิตีที่ปรากฏอยู่ในแผนภาพนั้นคือตารางข้อมูลในฐานข้อมูลของเครื่องมือ โดยรายละเอียดของแต่ละตารางเป็นดังนี้



รูปที่ 4.1 โครงสร้างฐานข้อมูลของเครื่องมือ

1. ตาราง DatabaseSchema เป็นตารางที่เก็บข้อมูลชื่อและคำอธิบายสคีมาของฐานข้อมูลที่เคยสร้างไว้โดยผู้ใช้
2. ตาราง DbTable เป็นตารางที่เก็บรายละเอียดของแต่ละตารางข้อมูลที่อยู่ในสคีมาของฐานข้อมูล
3. ตาราง DbField เป็นตารางที่เก็บรายละเอียดของแต่ละเขตข้อมูลที่อยู่ในตารางข้อมูลต่างๆ
4. ตาราง FieldType เป็นตารางที่เก็บรายละเอียดของชนิดข้อมูลต่างๆ ที่จะถูกใช้กำหนดให้กับเขตข้อมูล
5. ตาราง NormalDependency เป็นตารางที่เก็บรายละเอียดของการขึ้นต่อกันเชิงตรรกะแบบปกติระหว่างเขตข้อมูลในตารางข้อมูลต่างๆ

6. ตาราง SQLDependency เป็นตารางที่เก็บรายละเอียดของการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอลในตารางข้อมูลต่างๆ

7. ตาราง Category เป็นตารางที่เก็บรายละเอียดของกลุ่มค่าทั่วไปที่จะใช้ในการสร้างข้อมูลทดสอบ

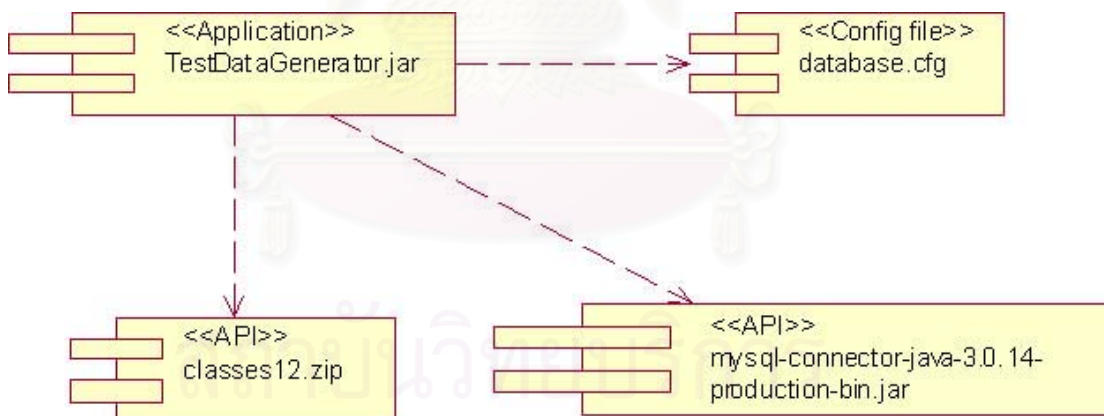
8. ตาราง CategoryValue เป็นตารางที่เก็บค่าข้อมูลของกลุ่มค่าทั่วไป

สำหรับพจนานุกรมข้อมูล (Data dictionary) ของแต่ละตารางสามารถดูได้ที่ภาคผนวก ก

4.3 โครงสร้างของเครื่องมือ

ส่วนโครงสร้างของเครื่องมือจะสามารถอธิบายได้ด้วยแผนภาพส่วนประกอบ (Component Diagram) ซึ่งเป็นแผนภาพที่แสดงส่วนประกอบต่างๆ ของเครื่องมือและความสัมพันธ์ระหว่างส่วนประกอบเหล่านั้น โดยจะแบ่งอธิบายเป็น 2 ส่วนคือส่วนประกอบในการนำเครื่องมือไปใช้งาน และส่วนประกอบของโครงสร้างหน้าจอของเครื่องมือ ดังนี้

4.3.1 ส่วนประกอบในการนำเครื่องมือไปใช้งาน



รูปที่ 4.2 แผนภาพส่วนประกอบแสดงส่วนประกอบในการนำเครื่องมือไปใช้งาน

จากรูปที่ 4.2 ในการนำเครื่องมือไปใช้งานนั้นจะมีส่วนประกอบที่เกี่ยวข้อง 4 ส่วน ดังนี้

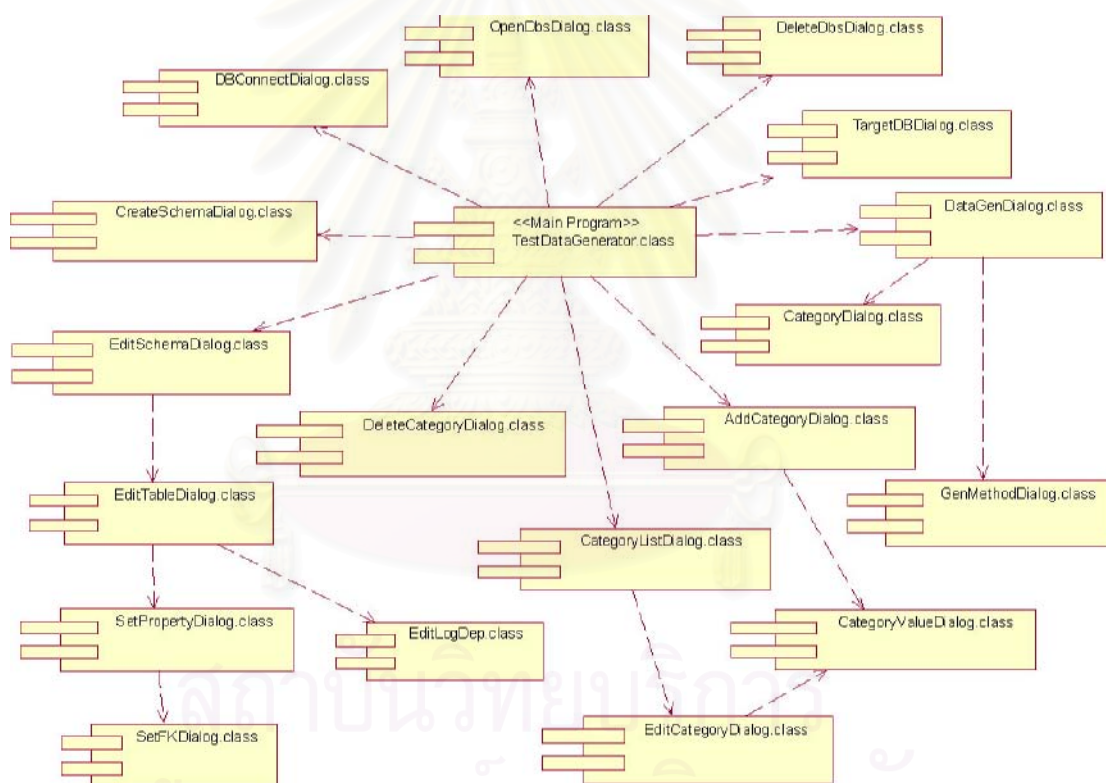
1. TestDataGenerator.jar เป็นส่วนประกอบที่เป็นตัวแทนของเครื่องมือสร้างข้อมูลทดสอบ โดยจะเป็นแฟ้มรูปแบบ .jar (เป็นแฟ้มที่ทำหน้าที่เก็บรายการของแฟ้มข้อมูลประเภท .class ที่มีข้อมูลในรูปแบบไบนารีโค้ด (Byte code) สำหรับการประมวลด้วยจาวาเวอร์ชวลแมชชีน (Java Virtual Machine) ที่เก็บรายการแฟ้ม .class ของเครื่องมือสร้างข้อมูลทดสอบ

2. database.cfg เป็นส่วนประกอบที่เป็นแฟ้มข้อความสำหรับเก็บพารามิเตอร์ต่างๆ ในการเชื่อมต่อกับฐานข้อมูลของเครื่องมือ ได้แก่ ชื่อฐานข้อมูลหลักของเครื่องมือ ชื่อฐานข้อมูล testDataDb ชื่อผู้ใช้ และรหัสผ่าน

3. classes12.zip เป็นส่วนประกอบที่เป็นเจตีสี้ดริเวอร์ (JDBC Driver) ที่เครื่องมือใช้ในการติดต่อกับฐานข้อมูลออรากเคิล

4. mysql-connector-java-3.0.14-production-bin.jar เป็นส่วนประกอบที่เป็นเจตีสี้ดริเวอร์ที่เครื่องมือใช้ในการติดต่อกับฐานข้อมูลมายเอสคิวแอล

4.3.2 ส่วนประกอบของโครงสร้างหน้าจอของเครื่องมือ



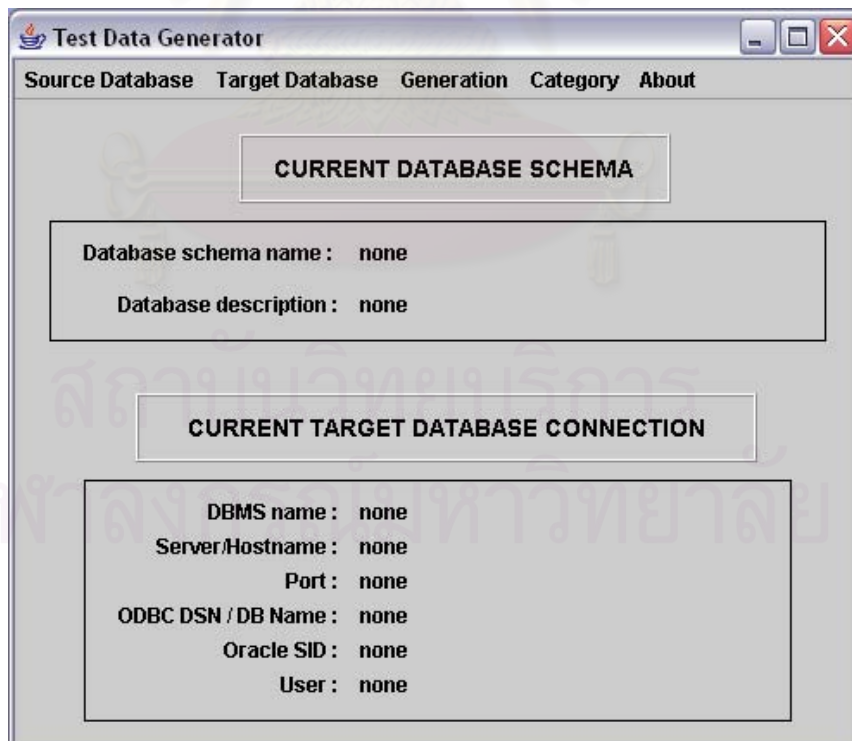
รูปที่ 4.3 แผนภาพส่วนประกอบแสดงโครงสร้างหน้าจอของเครื่องมือ

จากรูปที่ 4.3 เพิ่ม .class แต่ละแฟ้มจะแทนแต่ละหน้าจอของเครื่องมือ สำหรับรายละเอียดของแต่ละหน้าจอของเครื่องมือเป็นดังนี้

1. TestDataGenerator.class เป็นหน้าจอหลักของเครื่องมือมีลักษณะดังรูปที่ 4.4 หน้าจอหลักจะประกอบด้วยแถบรายการ (Menu Bar) ซึ่งประกอบด้วยรายการ 5 รายการดังนี้

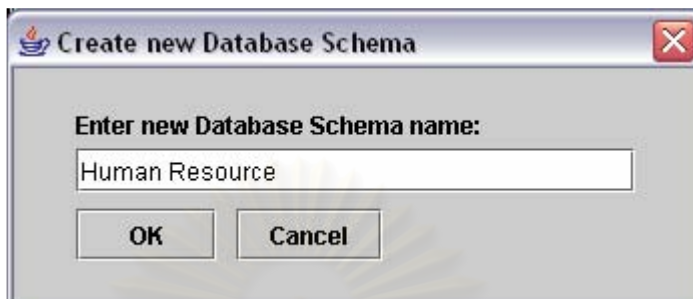
- 1.1 Source Database เป็นรายการที่ใช้ในการจัดการข้อมูลเกี่ยวกับสคีมาของฐานข้อมูลที่จะสร้างข้อมูลทดสอบ ได้แก่ การสร้างสคีมาของฐานข้อมูล การเปิดหรือปิดสคีมาของฐานข้อมูล การแก้ไขสคีมาของฐานข้อมูล และการลบสคีมาของฐานข้อมูล รวมทั้งการออกจากโปรแกรมด้วย
- 1.2 Target Database เป็นรายการที่ใช้กำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมายที่ข้อมูลทดสอบจะถูกบันทึกลงไป
- 1.3 Generation เป็นรายการที่ใช้ในการสร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลที่เลือกไว้
- 1.4 Category เป็นรายการที่ใช้ในการจัดการกลุ่มทั่วไปของเครื่องมือ ได้แก่ การเพิ่มกลุ่มค่าทั่วไป การแก้ไขกลุ่มค่าทั่วไป และการลบกลุ่มค่าทั่วไป
- 1.5 About เป็นรายการที่ใช้แสดงข้อมูลเกี่ยวกับเครื่องมือ

นอกจากแถบรายการแล้ว หน้าจอหลักยังประกอบด้วยส่วนแสดงผลอีก 2 ส่วน คือส่วนแสดงข้อมูลของสคีมาของฐานข้อมูลปัจจุบันที่ใช้งานอยู่ และส่วนแสดงรายละเอียดการเชื่อมต่อกับฐานข้อมูลเป้าหมายที่เครื่องมือกำลังเชื่อมต่ออยู่



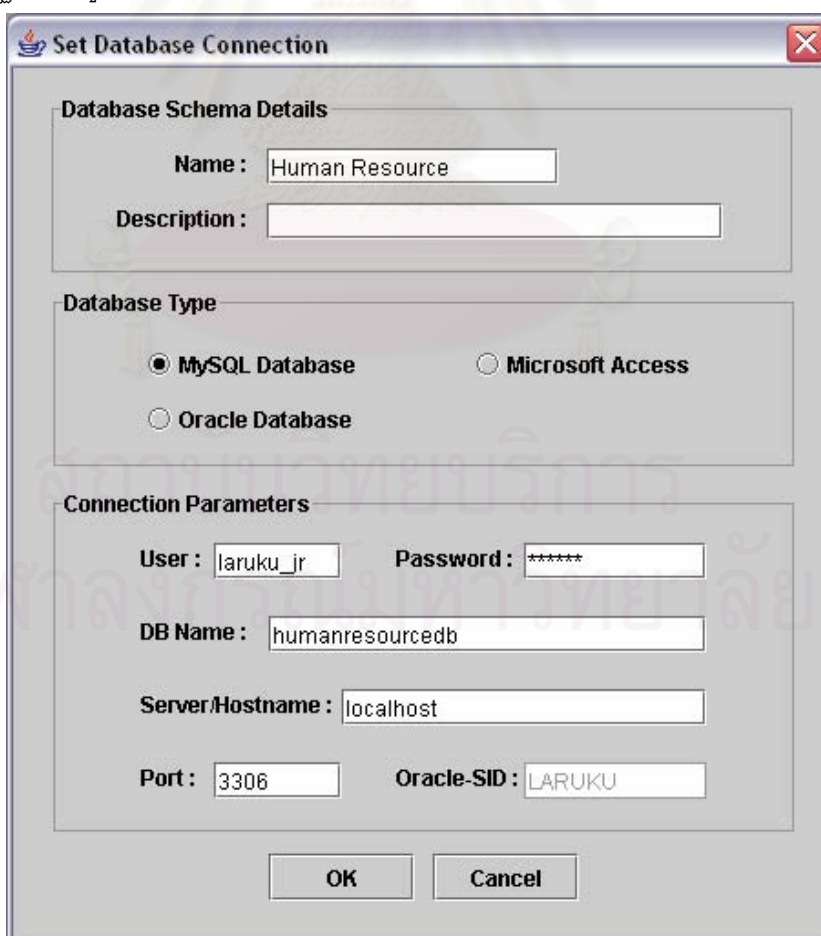
รูปที่ 4.4 หน้าจอหลัก

2. CreateSchemaDialog.class เป็นหน้าจอสร้างสคีมาของฐานข้อมูลใหม่มีลักษณะดังรูปที่ 4.5 หน้าจอนี้จะรับชื่อสคีมาของฐานข้อมูลเพื่อนำไปสร้างเป็นสคีมาของฐานข้อมูลใหม่



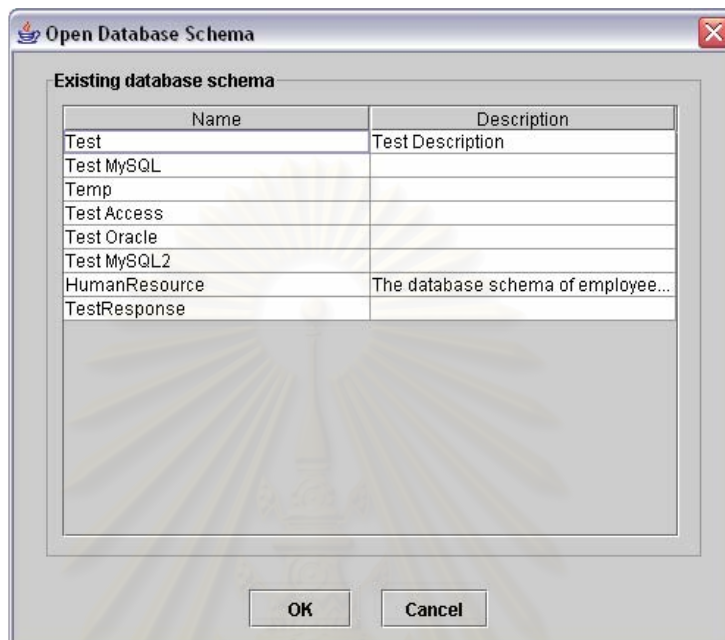
รูปที่ 4.5 หน้าจอสร้างสคีมาของฐานข้อมูลใหม่

3. DBConnectDialog.class เป็นหน้าจอสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทางมีลักษณะดังรูปที่ 4.6 หน้าจอจะรับข้อมูลชื่อและคำอธิบายของสคีมาของฐานข้อมูลใหม่ พร้อมทั้งชนิดของฐานข้อมูลและพารามิเตอร์ (Parameters) ต่างๆ ที่จะใช้ในการเชื่อมต่อกับฐานข้อมูลต้นทาง



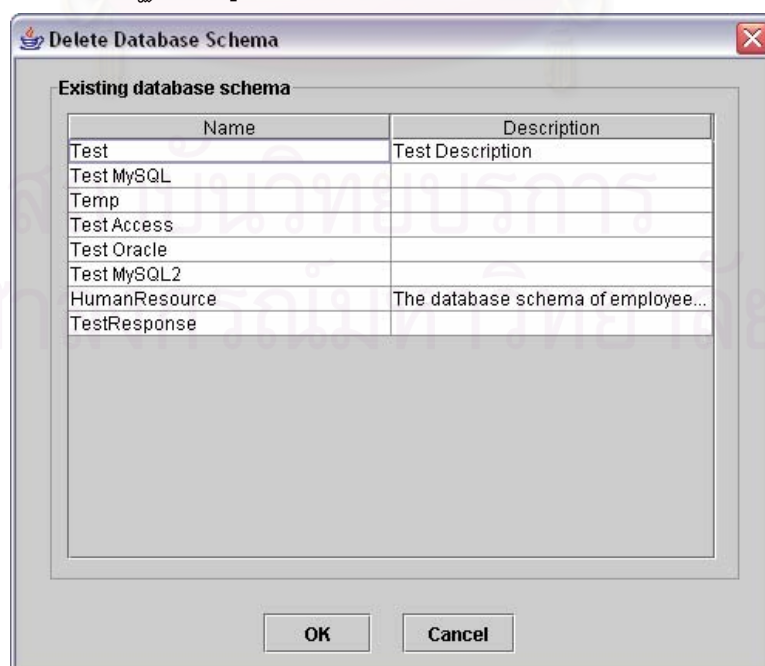
รูปที่ 4.6 หน้าจอสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลต้นทาง

4. OpenDbsDialog.class เป็นหน้าจอเปิดสคีมาของฐานข้อมูลมีลักษณะดังรูปที่ 4.7 หน้าจอจะแสดงรายการสคีมาของฐานข้อมูลที่มีอยู่ในเครื่องมือเพื่อให้ผู้ใช้เลือกเปิดได้ตามต้องการ



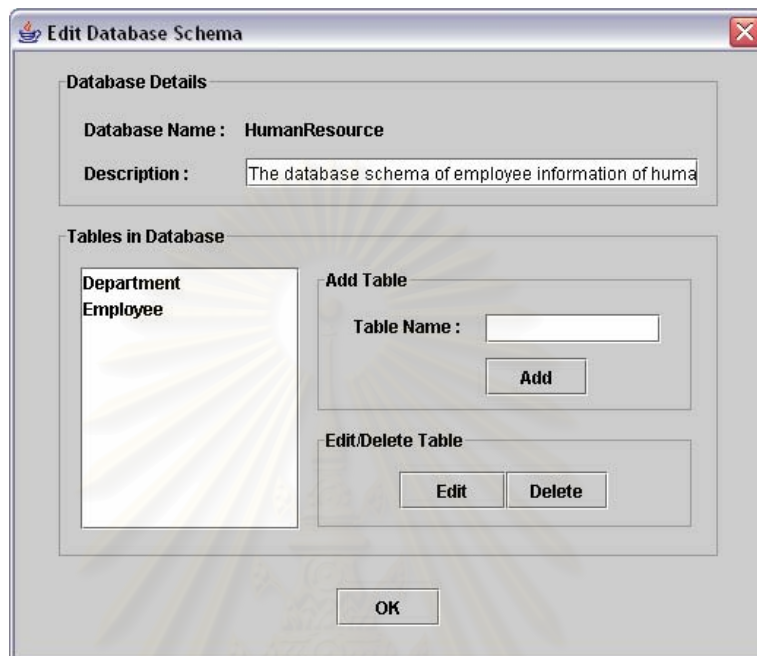
รูปที่ 4.7 หน้าจอเปิดสคีมาของฐานข้อมูล

5. DeleteDbsDialog.class เป็นหน้าจอลบสคีมาของฐานข้อมูลมีลักษณะดังรูปที่ 4.8 หน้าจอนี้จะเหมือนกับหน้าจอเปิดสคีมาของฐานข้อมูล เพียงแต่จะเป็นการเลือกสคีมาของฐานข้อมูลเพื่อลบออกจากฐานข้อมูลของเครื่องมือตนเอง



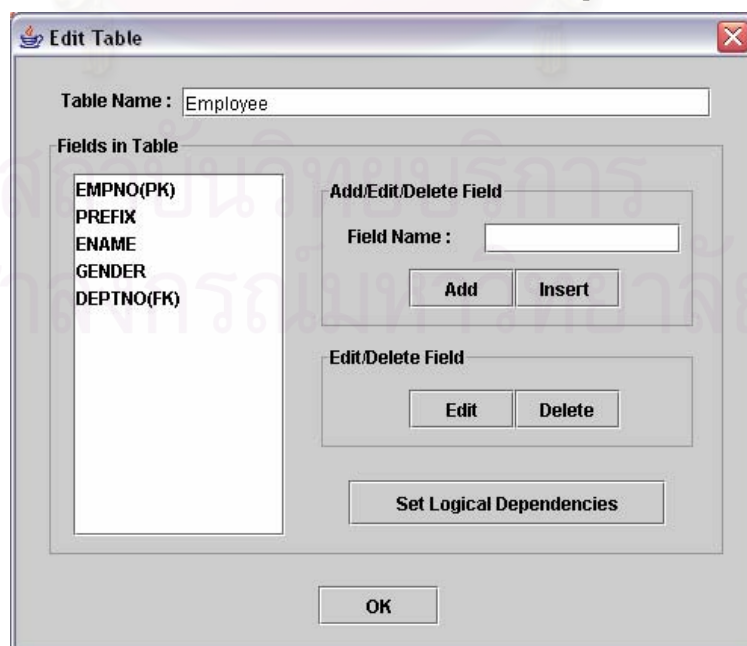
รูปที่ 4.8 หน้าจอลบสคีมาของฐานข้อมูล

6. EditSchemaDialog.class เป็นหน้าจอแก้ไขสคีมาของฐานข้อมูลมีลักษณะดังรูปที่ 4.9 หน้าจอนี้จะใช้ในการเพิ่มและลบตารางข้อมูลที่มีอยู่ในสคีมาของฐานข้อมูล รวมทั้งการเลือกตารางข้อมูลเพื่อทำการแก้ไขด้วย



รูปที่ 4.9 หน้าจอแก้ไขสคีมาของฐานข้อมูล

7. EditTableDialog.class เป็นหน้าจอแก้ไขตารางข้อมูลมีลักษณะดังรูปที่ 4.10 หน้าจอนี้จะใช้ในการเพิ่มและลบเขตข้อมูลที่มีอยู่ในตารางข้อมูล การเลือกเขตข้อมูลเพื่อทำการแก้ไข และการเลือกกำหนดการขึ้นต่อกันเชิงตรรกะให้กับตารางข้อมูลด้วย



รูปที่ 4.10 หน้าจอแก้ไขตารางข้อมูล

8. `EditLogDep.class` เป็นหน้าจอกำหนดการขึ้นต่อกันเชิงตรรกะมีลักษณะดังรูปที่ 4.11 หน้าจอนี้จะใช้กำหนดการต่อกันเชิงตรรกะของตารางข้อมูล โดยผู้ใช้สามารถเพิ่ม แก้ไข และลบเงื่อนไขของการขึ้นต่อกันเชิงตรรกะแบบปกติ รวมทั้งสามารถกำหนดคำสั่งเอสคิวแอลของการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอลได้ด้วย

Edit Logical Dependencies

Mode of Logical Dependency

Normal Logical Dependency SQL Logical Dependency

Add/Edit/Delete Normal Logical Dependencies

If the field are

then the field are

Current Normal Logical Dependencies

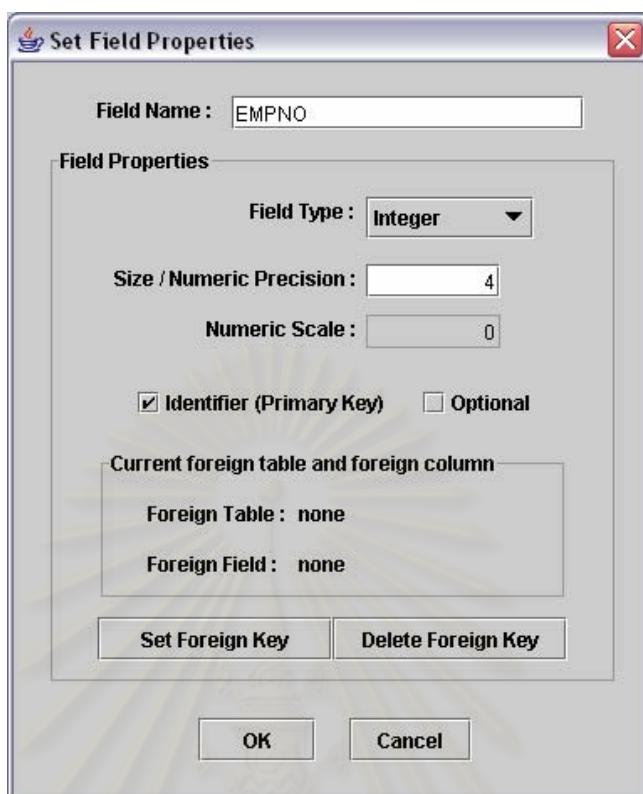
Source Field	Condition value	Destination Field	Result value
PREFIX	Mr.	GENDER	Male
PREFIX	Mrs. Miss	GENDER	Female

Specify SQL Logical Dependency

SQL statement :

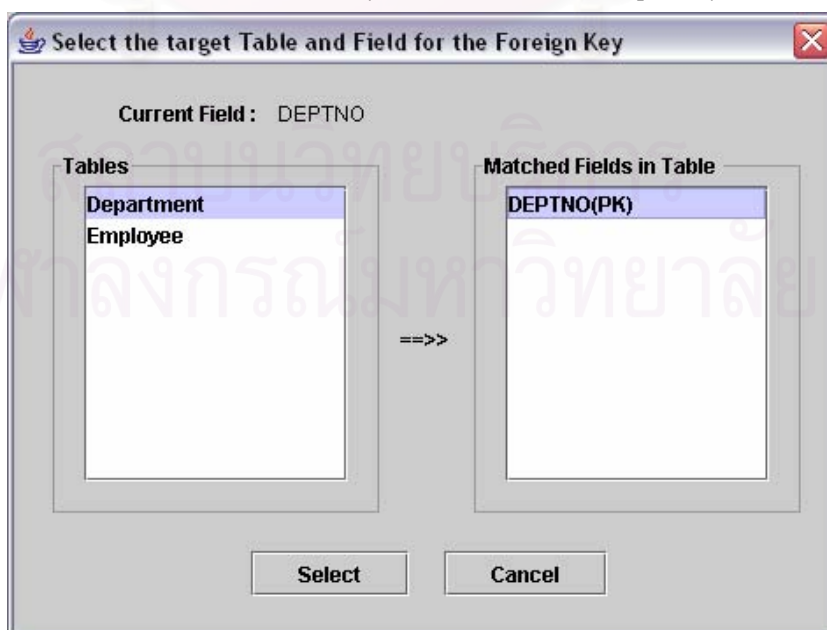
รูปที่ 4.11 หน้าจอกำหนดการขึ้นต่อกันเชิงตรรกะ

9. `SetPropertyDialog.class` เป็นหน้าจอแก้ไขเขตข้อมูลมีลักษณะดังรูปที่ 4.12 หน้าจอนี้จะใช้แก้ไขคุณสมบัติต่างๆ ของเขตข้อมูล ได้แก่ ชื่อเขตข้อมูล ชนิดข้อมูล ขนาด(หรือความเที่ยงตรงของจำนวน) ขนาดของทศนิยม ตัวระบุ และตัวเลือก รวมทั้งการเลือกกำหนดความคงสภาพในการอ้างอิงของเขตข้อมูลและลบความคงสภาพในการอ้างอิงของเขตข้อมูลที่กำหนดไว้ได้ด้วย



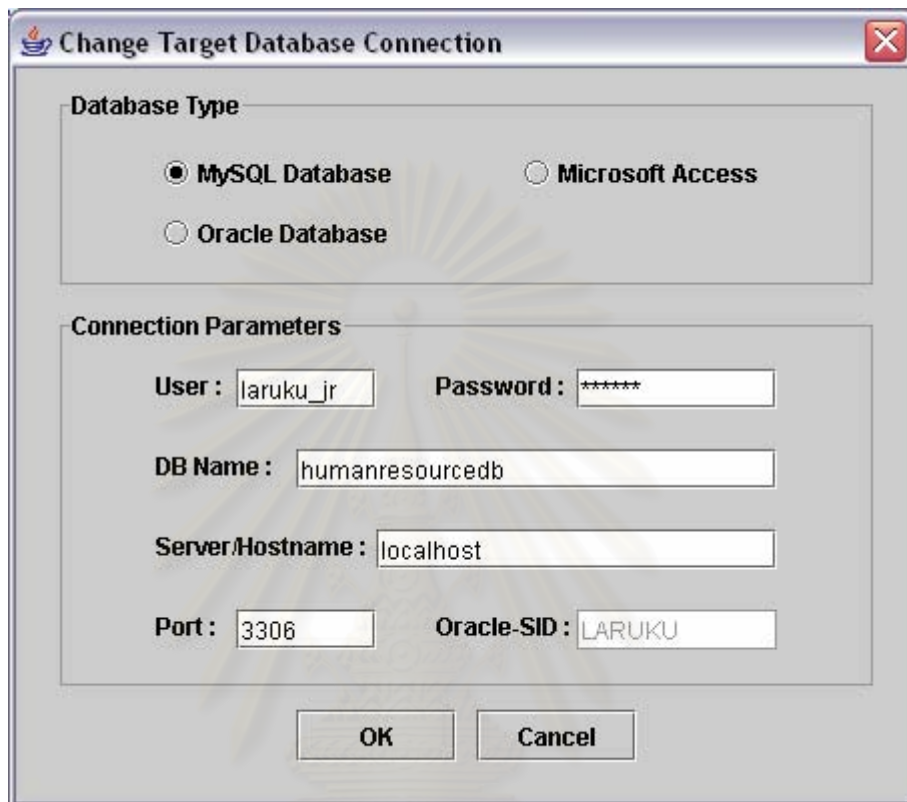
รูปที่ 4.12 หน้าจอแก้ไขเขตข้อมูล

10. SetFKDialog.class เป็นหน้าจอกำหนดความคงสภาพในการอ้างอิงมีลักษณะดังรูปที่ 4.13 หน้าจอนี้จะให้ผู้ใช้เลือกตารางข้อมูลและเขตข้อมูลที่ต้องการให้เขตข้อมูลปัจจุบันอ้างอิงถึงเพื่อเป็นการกำหนดความคงสภาพในการอ้างอิง โดยหน้าจอจะแสดงเฉพาะเขตข้อมูลที่เป็นคีย์หลักของตารางที่เลือกและมีคุณสมบัติตรงกับเขตข้อมูลปัจจุบันเท่านั้น



รูปที่ 4.13 หน้าจอกำหนดความคงสภาพในการอ้างอิง

11. TargetDBDialog.class เป็นหน้าจอกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมายมีลักษณะดังรูปที่ 4.14 หน้าจอนี้จะรับชนิดของฐานข้อมูลและพารามิเตอร์เพื่อใช้ในการเชื่อมต่อกับฐานข้อมูลเป้าหมาย



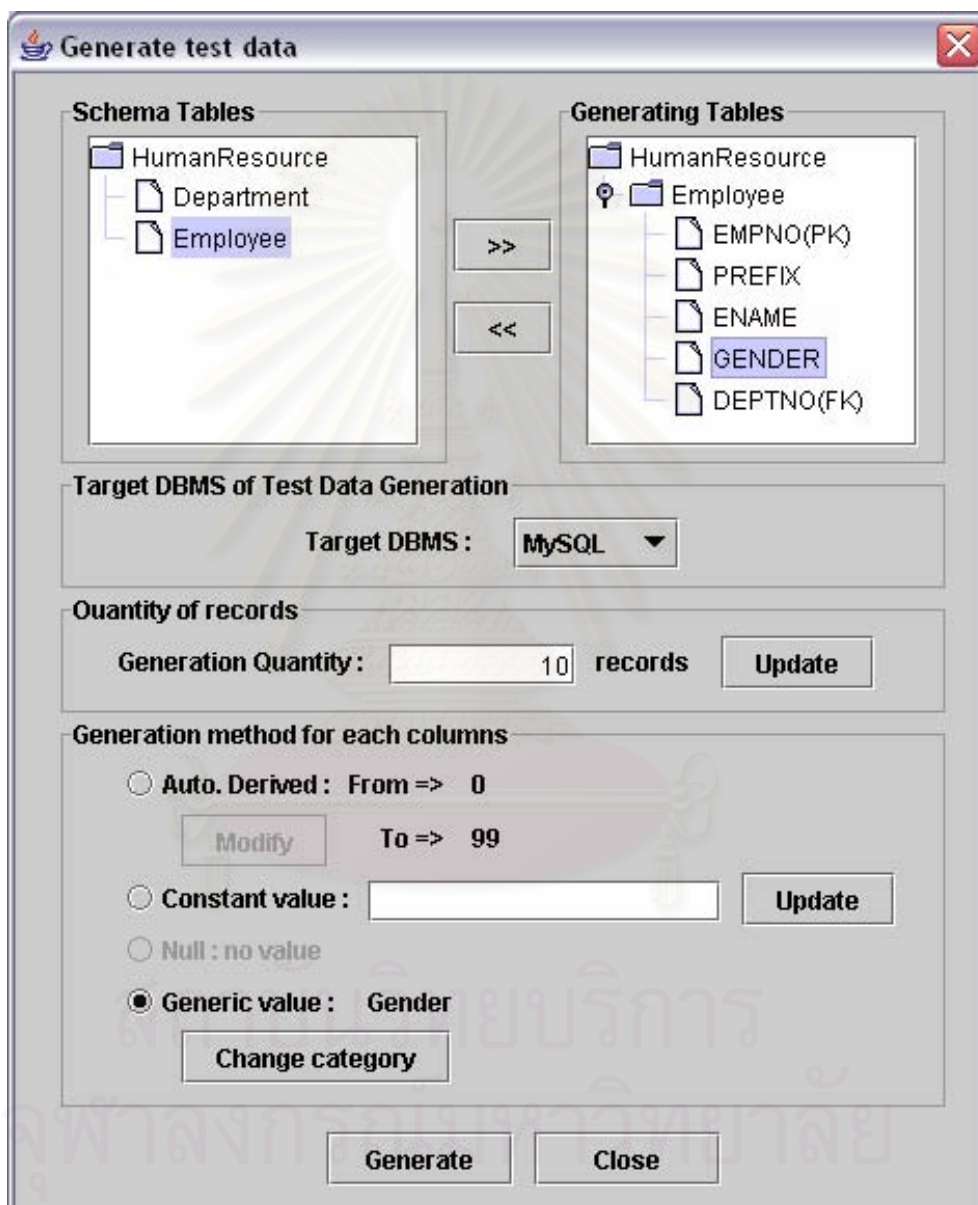
รูปที่ 4.14 หน้าจอกำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

12. DataGenDialog.class เป็นหน้าจอกำหนดรายละเอียดการสร้างข้อมูลทดสอบมีลักษณะดังรูปที่ 4.15 หน้าจอนี้จะใช้กำหนดรายละเอียดการสร้างข้อมูลทดสอบประกอบด้วยรายการตารางข้อมูลที่จะสร้างข้อมูลทดสอบ ชนิดของฐานข้อมูลเป้าหมายที่จะนำข้อมูลทดสอบไปใช้ จำนวนระเบียบข้อมูลที่จะสร้างให้กับแต่ละตาราง และวิธีการสร้างข้อมูลของแต่ละเขตข้อมูล

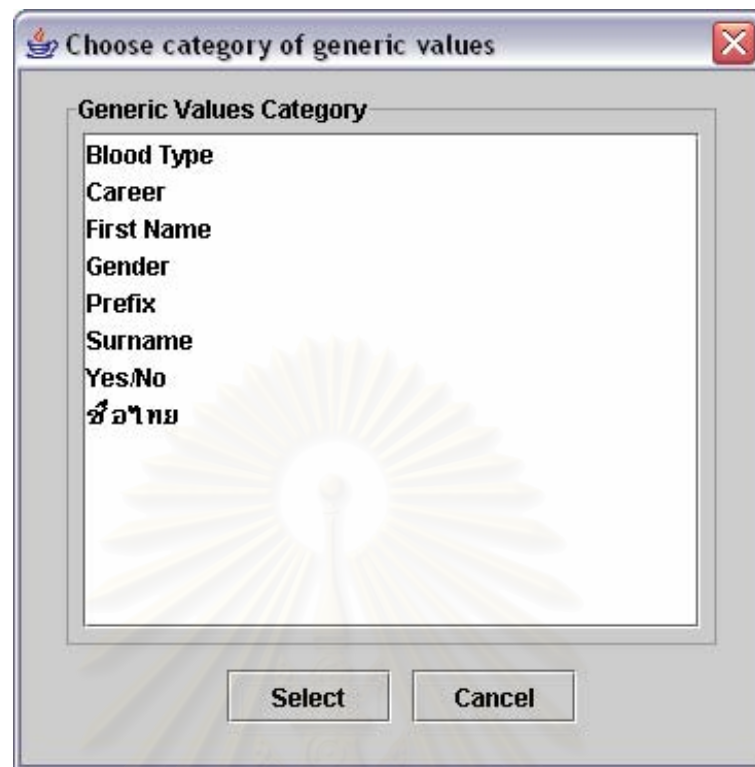
13. CategoryDialog.class เป็นหน้าจอเลือกกลุ่มค่าทั่วไปมีลักษณะดังรูปที่ 4.16 หน้าจอนี้ใช้เลือกกลุ่มค่าทั่วไปที่จะใช้ในการสร้างข้อมูลทดสอบให้กับเขตข้อมูลในกรณีที่เลือกใช้วิธีสร้างข้อมูลทดสอบแบบค่าทั่วไป

14. GenMethodDialog.class เป็นหน้าจอกำหนดรูปแบบการบันทึกข้อมูลทดสอบมีลักษณะดังรูปที่ 4.17 หน้าจอนี้จะใช้กำหนดรูปแบบการบันทึกข้อมูลทดสอบ โดยถ้าเลือกบันทึกเป็นแฟ้มข้อความที่มีตัวคั่น ผู้ใช้สามารถกำหนดตัวคั่นระหว่างเขตข้อมูลและตัวคั่น

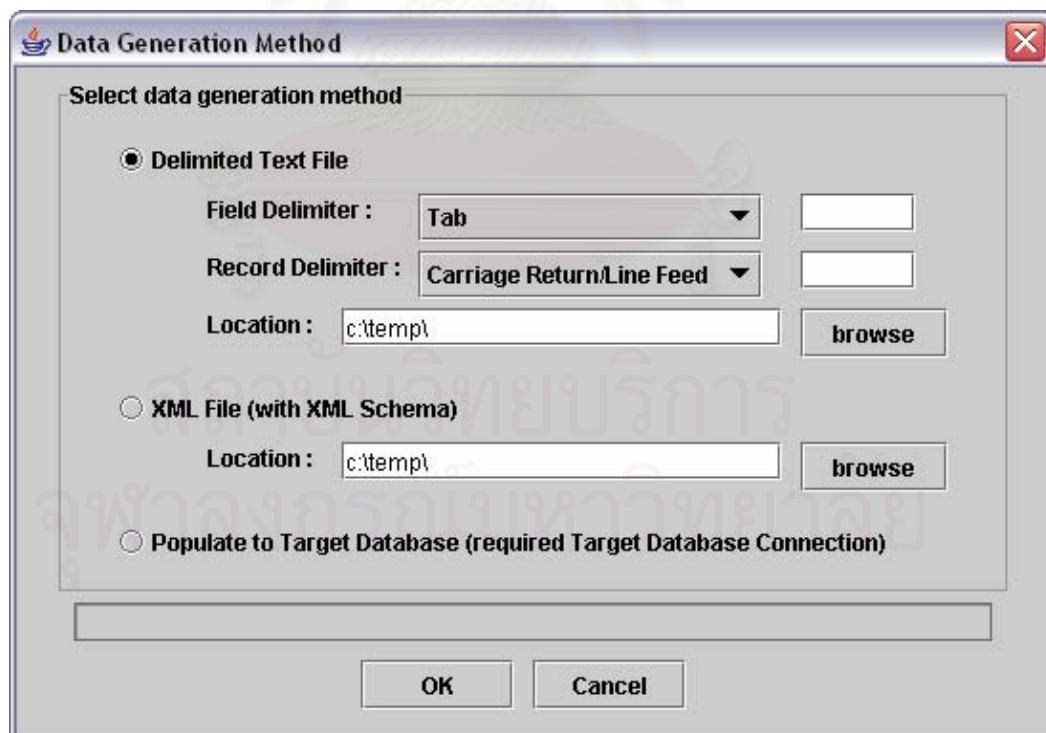
ระเบียบข้อมูล พร้อมทั้งไดเรกทอรีที่จะบันทึกเพิ่มข้อมูลได้ แต่ถ้าเลือกบันทึกเป็นแฟ้มเอกสาร เอ็กซ์เอ็มแอล ผู้ใช้จะกำหนดแค่ไดเรกทอรีที่จะบันทึกเพิ่มข้อมูลเท่านั้น และผู้ใช้สามารถเลือกบันทึกข้อมูลทดสอบลงฐานข้อมูลเป้าหมายที่กำลังเชื่อมต่ออยู่ได้



รูปที่ 4.15 หน้าจอกำหนดรายละเอียดการสร้างข้อมูลทดสอบ

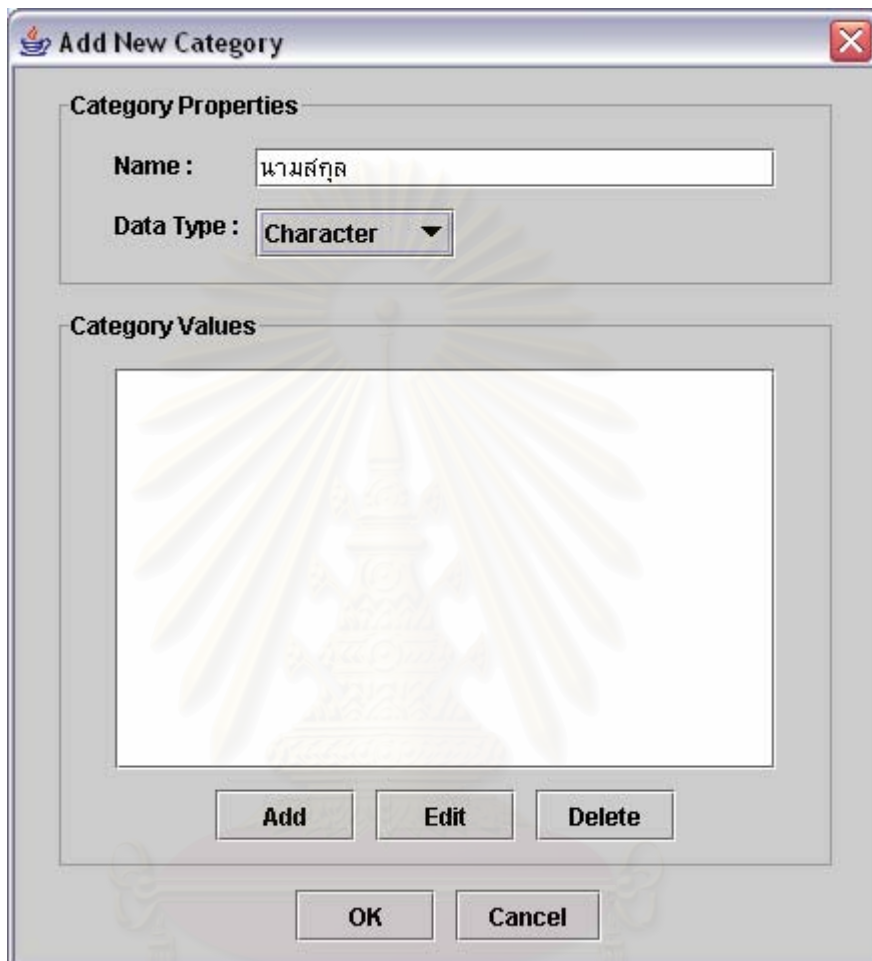


รูปที่ 4.16 หน้าจอเลือกกลุ่มค่าทั่วไป



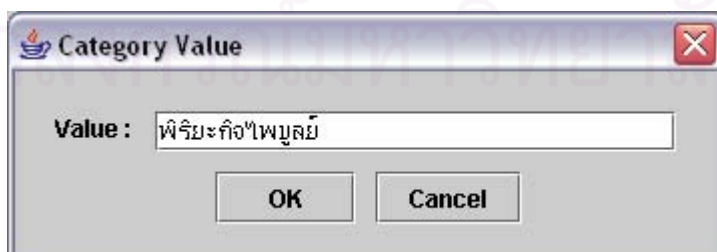
รูปที่ 4.17 หน้าจอกำหนดรูปแบบการบันทึกข้อมูลทดสอบ

15. `AddCategoryDialog.class` เป็นหน้าจอเพิ่มกลุ่มค่าทั่วไปมีลักษณะดังรูปที่ 4.18 หน้าจอนี้ใช้สร้างกลุ่มค่าข้อมูลใหม่ โดยผู้ใช้ต้องกำหนดชื่อ ชนิดข้อมูลและค่าข้อมูลให้กับกลุ่มค่าทั่วไป



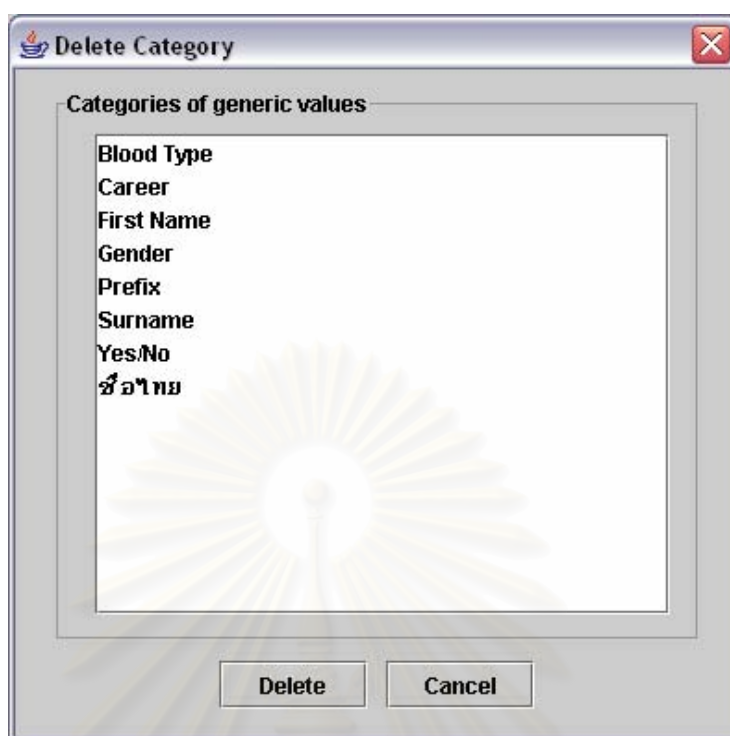
รูปที่ 4.18 หน้าจอเพิ่มกลุ่มค่าทั่วไป

16. `CategoryValueDialog.class` เป็นหน้าจอแก้ไขค่าข้อมูลมีลักษณะดังรูปที่ 4.19 หน้าจอนี้สามารถใช้เพิ่มค่าข้อมูลและแก้ไขค่าข้อมูลที่มีอยู่แล้วในกลุ่มค่าทั่วไปได้



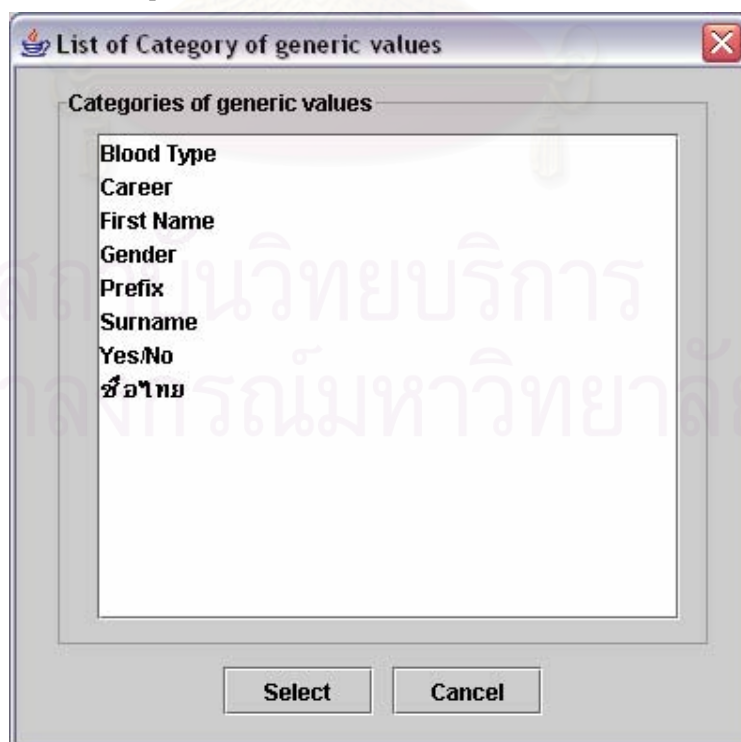
รูปที่ 4.19 หน้าจอแก้ไขค่าข้อมูล

17. `DeleteCategoryDialog.class` เป็นหน้าจอลบกลุ่มค่าทั่วไปมีลักษณะดังรูปที่ 4.20 หน้าจอนี้จะแสดงรายการกลุ่มค่าทั่วไปที่มีอยู่ในเครื่องมือเพื่อให้ผู้ใช้เลือกลบได้ตามต้องการ



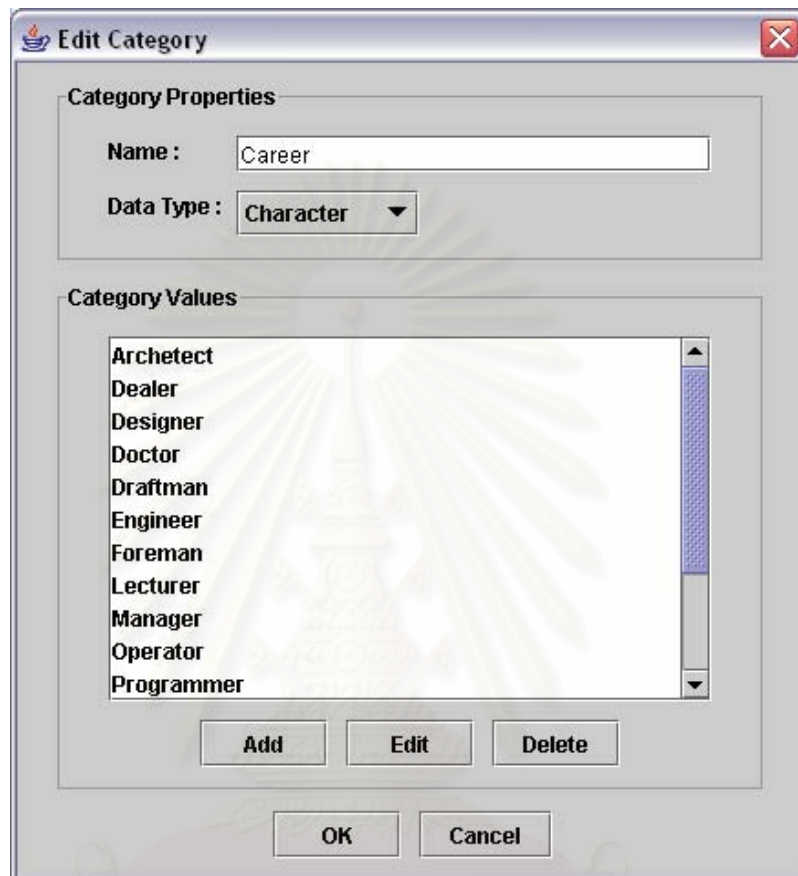
รูปที่ 4.20 หน้าจอลบกลุ่มค่าทั่วไป

18. `CategoryListDialog.class` เป็นหน้าจอแสดงรายการกลุ่มค่าทั่วไปมีลักษณะดังรูปที่ 4.21 หน้าจอนี้มีลักษณะเหมือนกับหน้าจอลบกลุ่มค่าทั่วไป เพียงแต่จะแสดงรายการกลุ่มค่าทั่วไปเพื่อให้ผู้ใช้เลือกแก้ไขเท่านั้น



รูปที่ 4.21 หน้าจอแสดงรายการกลุ่มค่าทั่วไป

19. `EditCategoryDialog.class` เป็นหน้าจอแก้ไขกลุ่มค่าทั่วไปมีลักษณะดังรูปที่ 4.22 หน้าจอนี้จะใช้แก้ไขข้อมูลของกลุ่มค่าทั่วไป ได้แก่ ชื่อและชนิดข้อมูล พร้อมทั้งสามารถเพิ่ม ลบ และแก้ไขค่าข้อมูลของกลุ่มค่าทั่วไปได้



รูปที่ 4.22 หน้าจอแก้ไขกลุ่มค่าทั่วไป

บทที่ 5

การทดสอบเครื่องมือ

5.1 สภาพแวดล้อมที่ใช้ในการทดสอบ

สภาพแวดล้อมที่ใช้ในการทดสอบเครื่องมือมีรายละเอียดดังนี้

1. ฮาร์ดแวร์

- เครื่องคอมพิวเตอร์แบบพีซี หน่วยประมวลผลอินเทลเพนเทียมโฟร์ 3.0 กิกะเฮิรท์
- หน่วยความจำสำรอง 1 กิกะไบต์
- ฮาร์ดดิสก์ 120 กิกะไบต์

2. ซอฟต์แวร์

- ระบบปฏิบัติการ ไมโครซอฟท์วินโดวส์เอ็กซ์พีโพรเฟชันแนล
- เจ2เอสดีเค เวอร์ชัน 1.4.2_01
- ฟอर्टเตพอร์จาวา เวอร์ชัน 3.0
- ฐานข้อมูลมายเอสคิวแอล เวอร์ชัน 4.0.18
- ฐานข้อมูลออราเคิล เวอร์ชัน 8.1.7
- ฐานข้อมูลไมโครซอฟท์แอ็คเซส 2003

5.2 การทดสอบเครื่องมือ

การทดสอบเครื่องมือประกอบด้วย การทดสอบ 2 ส่วน คือ การทดสอบการใช้เครื่องมือกับระบบงานจริง และการทดสอบเพื่อวัดเวลาตอบสนอง (Response Time) ในการสร้างข้อมูลทดสอบ โดยแต่ละการทดสอบมีรายละเอียดดังนี้

5.2.1 การทดสอบการใช้เครื่องมือกับระบบงานจริง

การทดสอบนี้เป็นการทดสอบเพื่อตรวจสอบว่าเครื่องมือสามารถสร้างข้อมูลตามคุณสมบัติของตารางข้อมูลต่างๆ ในระบบงานที่ใช้เป็นกรณีทดสอบได้ถูกต้องหรือไม่ โดยรายละเอียดของการทดสอบเป็นดังนี้

1. กรณีทดสอบ กรณีทดสอบที่ใช้ทดสอบเครื่องมือจะเป็นระบบงานที่ใช้ฐานข้อมูลเป็นหลักในเก็บข้อมูลทั้งหมดของระบบ ซึ่งประกอบด้วยระบบงาน 2 ระบบดังนี้

1.1 ระบบสารสนเทศผู้ป่วยนอกของสถานพยาบาลรัฐวิสาหกิจ [10] เป็นระบบที่เกี่ยวข้องกับการบริหารงานและการบริการรักษาพยาบาลของสถานพยาบาลรัฐวิสาหกิจ ได้แก่ งานเวชระเบียน งานห้องตรวจโรค และงานห้องจ่ายยา เป็นต้น ดังนั้นข้อมูลที่ใช้งานในระบบนี้จึงเป็นข้อมูลประเภททะเบียนประวัติ ข้อมูลคลังยา และข้อมูลที่ใช้ในการบริหารงานให้บริการรักษาพยาบาล การทดสอบกรณีทดสอบนี้เป็นการสร้างข้อมูลทดสอบประเภทประวัติบุคคล ข้อมูลที่อยู่ ข้อมูลเกี่ยวกับส่วนผู้ใช้งานระบบ และข้อมูลรหัสและรายละเอียดที่เกี่ยวข้องกับยา

1.2 ระบบช่วยตัดสินใจในการเลือกสถาบันเพื่อศึกษาต่อ [11] เป็นระบบที่พัฒนาขึ้นสำหรับช่วยตัดสินใจในการเลือกสถาบันเพื่อศึกษาต่อระดับปริญญาตรี ปริญญาโท และปริญญาเอกในประเทศสหรัฐอเมริกา ดังนั้นข้อมูลที่ใช้งานในระบบนี้จึงเป็นข้อมูลของมหาวิทยาลัยในประเทศสหรัฐอเมริกา ข้อมูลเกณฑ์ตัดสินใจต่าง ๆ ที่ใช้ในการพิจารณาเข้าศึกษาต่อของมหาวิทยาลัย ข้อมูลรายละเอียดของเมืองในประเทศสหรัฐอเมริกา ข้อมูลรหัสวิชา และข้อมูลรหัสรัฐและเมืองในประเทศสหรัฐอเมริกา และข้อมูลน้ำหนักต่าง ๆ ที่จำเป็นต่อการสร้างแบบจำลอง เป็นต้น การทดสอบกรณีทดสอบนี้เป็นการสร้างข้อมูลทดสอบประเภทข้อมูลศึกษาต่อ ประเทศสหรัฐอเมริกา เช่น ข้อมูลคะแนนต่าง ๆ ที่ใช้เป็นเกณฑ์ตัดสินใจต่าง ๆ ที่ใช้ในการเลือกนักศึกษาเข้าศึกษาต่อของมหาวิทยาลัย ข้อมูลของมหาวิทยาลัยที่ใช้ในการพิจารณาเลือกสถาบันเพื่อศึกษาต่อสำหรับนักศึกษา เช่น อัตราส่วนอาจารย์ต่อนักเรียน อัตราส่วนนักเรียนไทย และสภาพอากาศของเมืองที่มหาวิทยาลัยตั้งอยู่ เป็นต้น นอกจากนี้ ข้อมูลที่จะต้องสร้างเพื่อทดสอบระบบอีกประเภท คือ ค่าน้ำหนักต่าง ๆ ที่ใช้ในการสร้างแบบจำลองเพื่อการตัดสินใจ

2. การดำเนินการทดสอบ จะประกอบด้วยขั้นตอนดังนี้

- 2.1 ศึกษาระบบที่จะทดสอบเกี่ยวกับคุณสมบัติของเซตข้อมูลในแต่ละตารางข้อมูล และความหมายของข้อมูลของระบบ
- 2.2 เปิดใช้เครื่องมือสร้างข้อมูลทดสอบ
- 2.3 สร้างสคีมาของฐานข้อมูลตามข้อมูลของแต่ละระบบที่ได้ศึกษามา พร้อมทั้งเพิ่มข้อมูลการขึ้นต่อกันเชิงตรรกะให้กับสคีมาของฐานข้อมูลด้วย
- 2.4 สร้างกลุ่มค่าทั่วไปเพิ่มเติมเพื่อให้ข้อมูลทดสอบที่สร้างออกมามีความใกล้เคียงกับข้อมูลที่ได้จากการใช้งานจริง

2.5 สร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลของระบบ

2.6 วิเคราะห์และสรุปผลการทดลอง

3. ผลการทดสอบ สามารถดูได้ที่ภาคผนวก ข และ ค

4. สรุปผลการทดสอบ จากผลการทดสอบเครื่องมือในการสร้างข้อมูลทดสอบให้กับระบบทั้ง 2 ระบบที่ได้กล่าวมา สามารถสรุปได้ว่าเครื่องมือสามารถสร้างข้อมูลทดสอบได้ตรงกับคุณสมบัติของข้อมูลที่มีอยู่จริง โดยข้อมูลทดสอบที่สร้างได้จะมีความถูกต้องตามการขึ้นต่อกันเชิงตรรกะและความคงสภาพในการอ้างอิงของสคีมาของฐานข้อมูลด้วย

5.2.2 การทดสอบเพื่อวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบ

การทดสอบนี้เป็นการทดสอบเพื่อวัดเวลาตอบสนองที่เครื่องมือใช้ในการสร้างข้อมูลทดสอบ (รวมเวลาที่ใช้ในการบันทึกข้อมูลทดสอบด้วย) เวลาตอบสนองในการสร้างข้อมูลทดสอบนั้นถือเป็นปัจจัยที่สำคัญมากในการสร้างข้อมูลทดสอบ เนื่องจากระเบียบข้อมูลที่อยู่ในฐานข้อมูลระบบงานจริงๆ นั้นมีปริมาณมาก การสร้างข้อมูลทดสอบให้ใกล้เคียงหรือเหมือนกับข้อมูลของระบบงานจริงจึงต้องมีปริมาณที่มากตามไปด้วย ดังนั้นเครื่องมือสร้างข้อมูลทดสอบที่ดีจะต้องสามารถสร้างข้อมูลทดสอบได้ภายในเวลาที่สมเหตุสมผลด้วย รายละเอียดของการทดสอบเป็นดังนี้

1. กรณีทดสอบ กรณีการทดสอบที่ออกแบบไว้จะคำนึงถึงปัจจัยที่ส่งผลกระทบต่อเวลาตอบสนองในการกระบวนการสร้างข้อมูล ซึ่งปัจจัยที่ผู้ทดสอบสนใจมีอยู่ 3 ปัจจัยคือ จำนวนตารางข้อมูล ความคงสภาพในการอ้างอิงระหว่างตารางข้อมูล และรูปแบบการบันทึกข้อมูลทดสอบ โดยแต่ละปัจจัยจะมีค่าเปลี่ยนไปในกรณีทดสอบดังนี้

1.1 จำนวนตารางข้อมูล มีอยู่ 2 ค่าคือ 5 ตารางและ 10 ตาราง

1.2 ความคงสภาพในการอ้างอิงระหว่างตารางข้อมูล มีอยู่ 3 รูปแบบคือ

- ไม่มีความคงสภาพในการอ้างอิงระหว่างตารางข้อมูล
- มีความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลที่เชื่อมถึงกันทุกตาราง
- มีความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลที่เชื่อมถึงกันทุกตารางพร้อมทั้งมีความคงสภาพในการอ้างอิงภายในตารางข้อมูลด้วย

1.3 รูปแบบการบันทึกข้อมูล มีอยู่ 5 รูปแบบคือ

- บันทึกเป็นแฟ้มข้อความที่มีตัวค้น
- บันทึกเป็นแฟ้มเอกสารอิเล็กทรอนิกส์เอ็มแอล
- บันทึกลงฐานข้อมูลมายเอสคิวแอล
- บันทึกลงฐานข้อมูลออราเคิล
- บันทึกลงฐานข้อมูลไมโครซอฟท์แอ็คเซส

สำหรับปัจจัยอื่นในแต่ละกรณีทดสอบนั้นจะเป็นปัจจัยควบคุมที่มีค่าเหมือนกันทุกกรณีทดสอบได้แก่ จำนวนเขตข้อมูลในแต่ละตารางและจำนวนระเบียนข้อมูลที่สร้างให้กับแต่ละตาราง โดยทุกๆ ตารางจะมีเขตข้อมูล 10 เขตข้อมูล และทุกๆ ตารางจะได้รับการสร้างระเบียนข้อมูล 1000 ระเบียนเท่ากันทั้งหมด สำหรับรายละเอียดคุณสมบัติของแต่ละเขตข้อมูลในตารางข้อมูลจะเป็นดังตารางที่ 5.1

ตารางที่ 5.1 รายละเอียดของแต่ละเขตข้อมูลในตารางข้อมูล

เขตข้อมูล	ชนิดข้อมูล	การสร้างข้อมูลทดสอบ
EMPNO(PK)	จำนวนเต็ม	เครื่องมือสุ่มสร้างอัตโนมัติจาก 0 ถึง 9999
PREFIX	ตัวอักษร	ใช้กลุ่มค่าทั่วไปของคำนำหน้า (Prefix)
ENAME	ตัวอักษร	ใช้กลุ่มค่าทั่วไปของชื่อ (First Name)
GENDER	ตัวอักษร	ใช้กลุ่มค่าทั่วไปของเพศ (Gender)
JOB	ตัวอักษร	ใช้กลุ่มค่าทั่วไปของอาชีพ (Career)
MGR	จำนวนเต็ม	เครื่องมือสุ่มสร้างอัตโนมัติจาก 0 ถึง 9999
HIREDATE	ค่าวันเดือนปี	เครื่องมือสุ่มสร้างอัตโนมัติจาก 01/01/2004 ถึง 01/12/2004
SAL	จำนวนทศนิยม	เครื่องมือสุ่มสร้างอัตโนมัติจาก 0.0 ถึง 99999.99
COMM	จำนวนทศนิยม	เครื่องมือสุ่มสร้างอัตโนมัติจาก 0.0 ถึง 99999.99
DEPT	จำนวนเต็ม	เครื่องมือสุ่มสร้างอัตโนมัติจาก 0 ถึง 99

2. การดำเนินการทดสอบ มีขั้นตอนดังนี้

- 2.1 เปิดใช้เครื่องมือสร้างข้อมูลทดสอบ
- 2.2 สร้างสคีมาของฐานข้อมูลพร้อมทั้งตารางข้อมูลที่จะใช้ทดสอบ
- 2.3 สร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลของระบบ
- 2.4 วิเคราะห์และสรุปผลการทดลอง

3. ผลการทดสอบ สามารถดูได้ที่ภาคผนวก ง
4. สรุปผลการทดสอบ การทดสอบเครื่องมือในการสร้างข้อมูลทดสอบตามกรณีทดสอบที่กล่าวมา สามารถสรุปได้ดังตารางที่ 5.2 และจากตารางที่ 5.2 สามารถสรุปผลการทดสอบได้ดังนี้
 - 4.1 เมื่อมีจำนวนตารางข้อมูลที่สร้างข้อมูลทดสอบมากขึ้น เวลาตอบสนองในการสร้างข้อมูลจะมากขึ้นอย่างชัดเจน
 - 4.2 เมื่อไม่มีความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลที่สร้างข้อมูลทดสอบ เวลาตอบสนองในการสร้างข้อมูลทดสอบน้อยมาก
 - 4.3 เมื่อมีความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลที่สร้างข้อมูลทดสอบ เวลาตอบสนองในการสร้างข้อมูลทดสอบมากกว่าเมื่อไม่มีความคงสภาพในการอ้างอิงระหว่างตารางข้อมูลอย่างเห็นได้ชัด
 - 4.4 ความคงสภาพในการอ้างอิงภายในตารางข้อมูลไม่ส่งผลกระทบต่อเวลาตอบสนองในการสร้างข้อมูลทดสอบมากนัก
 - 4.5 การสร้างข้อมูลทดสอบที่บันทึกข้อมูลทดสอบลงในฐานข้อมูลออราเคิลใช้เวลาตอบสนองมากที่สุดเมื่อเทียบกับรูปแบบการบันทึกข้อมูลทดสอบแบบอื่นๆ
 - 4.6 เวลาตอบสนองในการสร้างข้อมูลทดสอบด้วยเครื่องมือที่พัฒนาขึ้นมา นั้นถือว่ายอมรับได้และสมเหตุสมผล

ตารางที่ 5.2 สรุปผลการทดสอบวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบ

กรณีทดสอบ	จำนวนตารางข้อมูล	จำนวนเขตข้อมูล/ตาราง	จำนวนระเบียบข้อมูล/ตาราง	ความคงสภาพในการอ้างอิง	รูปแบบการบันทึกข้อมูลทดสอบ	เวลาตอบสนอง (ชม:นา:วินาที)
1	5	10	1000	ไม่มี	เพิ่มข้อความที่มีตัวค้น	00:00:12
2	5	10	1000	ไม่มี	เพิ่มเอกสารอิเล็กทรอนิกส์อีเมล	00:00:11
3	5	10	1000	ไม่มี	ฐานข้อมูลมายเอสคิวแอล	00:00:12
4	5	10	1000	ไม่มี	ฐานข้อมูลออราเคิล	00:00:30
5	5	10	1000	ไม่มี	ฐานข้อมูลไมโครซอฟท์แอ็คเซส	00:00:17
6	10	10	1000	ไม่มี	เพิ่มข้อความที่มีตัวค้น	00:00:21
7	10	10	1000	ไม่มี	เพิ่มเอกสารอิเล็กทรอนิกส์อีเมล	00:00:23
8	10	10	1000	ไม่มี	ฐานข้อมูลมายเอสคิวแอล	00:00:23
9	10	10	1000	ไม่มี	ฐานข้อมูลออราเคิล	00:00:55
10	10	10	1000	ไม่มี	ฐานข้อมูลไมโครซอฟท์แอ็คเซส	00:00:33
11	5	10	1000	4 FK	เพิ่มข้อความที่มีตัวค้น	00:01:23
12	5	10	1000	4 FK	เพิ่มเอกสารอิเล็กทรอนิกส์อีเมล	00:01:21
13	5	10	1000	4 FK	ฐานข้อมูลมายเอสคิวแอล	00:01:25
14	5	10	1000	4 FK	ฐานข้อมูลออราเคิล	00:01:36
15	5	10	1000	4 FK	ฐานข้อมูลไมโครซอฟท์แอ็คเซส	00:01:27
16	10	10	1000	9 FK	เพิ่มข้อความที่มีตัวค้น	00:02:58

ตารางที่ 5.2 สรุปผลการทดสอบวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบ (ต่อ)

กรณีทดสอบ	จำนวนตารางข้อมูล	จำนวนเขตข้อมูล/ตาราง	จำนวนระเบียบข้อมูล/ตาราง	ความคงสภาพในการอ้างอิง	รูปแบบการบันทึกข้อมูลทดสอบ	เวลาตอบสนอง (ชม:นา:วินาที)
17	10	10	1000	9 FK	เพิ่มเอกสารอิเล็กทรอนิกส์	00:02:51
18	10	10	1000	9 FK	ฐานข้อมูลมายเอสคิวแอล	00:02:50
19	10	10	1000	9 FK	ฐานข้อมูลออราเคิล	00:03:25
20	10	10	1000	9 FK	ฐานข้อมูลไมโครซอฟท์แอ็คเซส	00:03:02
21	5	10	1000	4 FK, 1 Self	เพิ่มข้อความที่มีตัวค้น	00:01:20
22	5	10	1000	4 FK, 1 Self	เพิ่มเอกสารอิเล็กทรอนิกส์	00:01:20
23	5	10	1000	4 FK, 1 Self	ฐานข้อมูลมายเอสคิวแอล	00:01:21
24	5	10	1000	4 FK, 1 Self	ฐานข้อมูลออราเคิล	00:01:28
25	5	10	1000	4 FK, 1 Self	ฐานข้อมูลไมโครซอฟท์แอ็คเซส	00:01:25
26	10	10	1000	9 FK, 1 Self	เพิ่มข้อความที่มีตัวค้น	00:03:02
27	10	10	1000	9 FK, 1 Self	เพิ่มเอกสารอิเล็กทรอนิกส์	00:02:53
28	10	10	1000	9 FK, 1 Self	ฐานข้อมูลมายเอสคิวแอล	00:03:08
29	10	10	1000	9 FK, 1 Self	ฐานข้อมูลออราเคิล	00:03:26
30	10	10	1000	9 FK, 1 Self	ฐานข้อมูลไมโครซอฟท์แอ็คเซส	00:02:57

หมายเหตุ FK คือจำนวนเขตข้อมูลที่เป็นคีย์อ้างอิงที่อ้างอิงถึงเขตข้อมูลของตารางข้อมูลอื่นที่อยู่ในกลุ่มตารางข้อมูลการสร้างข้อมูลทดสอบ

Self คือจำนวนเขตข้อมูลที่เป็นคีย์อ้างอิงที่อ้างอิงถึงเขตข้อมูลภายในตารางข้อมูลเดียวกัน

บทที่ 6

สรุปผลการวิจัย

จากการศึกษา ออกแบบและพัฒนาเครื่องมือสร้างข้อมูลทดสอบ สามารถสรุปผลการวิจัย ประโยชน์ที่ได้รับ ข้อจำกัดของเครื่องมือ และแนวทางในการพัฒนาต่อได้ดังนี้

6.1 สรุปผลการวิจัย

วิทยานิพนธ์นี้ได้พัฒนาเครื่องมือสร้างข้อมูลทดสอบตามสคีมาของฐานข้อมูลที่กำหนดมา โดยผู้ใช้งานสามารถสร้างสคีมาของฐานข้อมูลขึ้นมาเองหรือกำหนดให้เครื่องมืออ่านสคีมาของฐานข้อมูลที่มีอยู่แล้วขึ้นมาจากรูปร่างข้อมูลที่ต้องการได้ ในการกำหนดชนิดข้อมูลในกับแต่ละเขตข้อมูลของตารางข้อมูลนั้น ผู้ใช้งานสามารถเลือกกำหนดชนิดข้อมูลได้ 7 ชนิด ได้แก่ ตัวอักษรระจำนวนเต็ม จำนวนทศนิยม ค่าตรรกะ ค่าวันเดือนปี ค่าเวลา และค่าช่วงเวลา ผู้ใช้งานสามารถกำหนดการขึ้นต่อกันเชิงตรรกะระหว่างเขตข้อมูลในแต่ละตารางได้ 2 แบบคือแบบปกติที่กำหนดเป็นเงื่อนไขระหว่างเขตข้อมูล และแบบเอสคิวแอลที่กำหนดเป็นคำสั่งเอสคิวแอล นอกจากนี้ผู้ใช้งานยังสามารถกำหนดความสัมพันธ์ในอ้างอิงระหว่างตารางข้อมูลได้ด้วยการกำหนดคีย์อ้างอิงของแต่ละตารางข้อมูล ในการสร้างข้อมูลทดสอบผู้ใช้งานสามารถกำหนดวิธีการสร้างข้อมูลทดสอบให้กับแต่ละเขตข้อมูลได้ 4 วิธี ได้แก่ กำหนดให้เครื่องมือสุ่มสร้างให้เองโดยอัตโนมัติ กำหนดให้เป็นค่าคงที่ กำหนดให้เป็นค่าว่าง และกำหนดให้ใช้กลุ่มค่าทั่วไป และเครื่องมือสามารถบันทึกข้อมูลทดสอบที่สร้างเสร็จแล้วได้ 3 รูปแบบคือบันทึกเป็นแฟ้มข้อความที่มีตัวค้น บันทึกเป็นแฟ้มเอกสารอิเล็กทรอนิกส์ และเพิ่มข้อมูลลงฐานข้อมูลเป้าหมายที่เชื่อมต่อกับเครื่องมือโดยตรง

ในส่วนการทดสอบเครื่องมือสร้างข้อมูลทดสอบที่พัฒนาขึ้นนี้ ผู้ทดสอบได้ทำการทดสอบสร้างข้อมูลทดสอบให้กับระบบงานจริง 2 ระบบคือ ระบบสารสนเทศผู้ป่วยนอกของสถานพยาบาลรัฐวิสาหกิจและระบบช่วยตัดสินใจในการเลือกสถาบันเพื่อศึกษาต่อ ซึ่งผลการทดสอบแสดงให้เห็นว่าเครื่องมือสามารถสร้างข้อมูลทดสอบได้ตรงตามคุณสมบัติของข้อมูลในระบบงานจริงและมีความถูกต้องตามการขึ้นต่อกันเชิงตรรกะและความคงสภาพในการอ้างอิงของสคีมาของฐานข้อมูลด้วย นอกจากนี้ผู้ทดสอบยังได้ทดสอบเวลาตอบสนองในการสร้างข้อมูลทดสอบของเครื่องมือด้วย ซึ่งเครื่องมือสามารถสร้างข้อมูลทดสอบโดยใช้เวลาตอบสนองที่เหมาะสมและยอมรับได้ สำหรับจำนวนเงื่อนไขของการขึ้นต่อกันเชิงตรรกะในแต่ละตารางข้อมูลนั้นจะไม่มีผลต่อจำนวนระเบียบข้อมูลทดสอบที่สร้างให้กับตารางข้อมูลนั้น เนื่องจากเครื่องมือจะ

นำข้อมูลทดสอบที่สร้างเสร็จแล้วมาตรวจสอบกับเงื่อนไขและเปลี่ยนค่าไปที่ละระเบียบข้อมูล โดยไม่มีการสร้างระเบียบข้อมูลเพิ่มขึ้น ดังนั้นจำนวนระเบียบข้อมูลที่สร้างให้แต่ละตารางข้อมูลจะเท่ากับจำนวนระเบียบที่ผู้ใช้กำหนดมาเสมอ

6.2 ข้อจำกัดของเครื่องมือสร้างข้อมูลทดสอบ

1. ผู้ใช้สามารถกำหนดชนิดข้อมูลให้กับแต่ละเขตข้อมูลได้เพียง 7 ชนิดที่กล่าวมาเท่านั้น
2. เครื่องมือสร้างข้อมูลทดสอบสามารถติดต่อกับฐานข้อมูลได้เพียง 3 ชนิดเท่านั้นคือ ฐานข้อมูลมายเอสคิวแอล ฐานข้อมูลออรากเคิล และฐานข้อมูลไมโครซอฟท์แอ็คเซส
3. เครื่องมือสร้างข้อมูลทดสอบไม่สามารถสร้างตารางข้อมูลในฐานข้อมูลเป้าหมายได้เอง หากผู้ใช้ต้องการบันทึกข้อมูลทดสอบลงในฐานข้อมูลเป้าหมาย ภายในฐานข้อมูลเป้าหมายต้องมีตารางข้อมูลที่มีโครงสร้างตรงกันกับตารางข้อมูลในสคีมาของฐานข้อมูลที่เครื่องมือสร้างข้อมูลทดสอบไว้แล้ว

6.3 แนวทางในการพัฒนาต่อ

1. ปรับปรุงเครื่องมือสร้างข้อมูลทดสอบให้สนับสนุนชนิดข้อมูลหลายรูปแบบมากขึ้น เนื่องจากในปัจจุบันฐานข้อมูลเชิงสัมพันธ์ที่มีอยู่ในตลาดต่างสนับสนุนชนิดข้อมูลใหม่ๆ หลายชนิด เช่น CLOB, BLOB, BFILE, ROWID, SET และ ENUM เป็นต้น
2. พัฒนาเครื่องมือสร้างข้อมูลทดสอบให้สามารถติดต่อกับฐานข้อมูลเชิงสัมพันธ์ได้มากกว่าที่มีอยู่ เช่น Microsoft SQL Sever, DB2 Universal และ Sybase เป็นต้น
3. เพิ่มความสามารถให้เครื่องมือสร้างข้อมูลทดสอบสามารถจะจัดการข้อมูลในฐานข้อมูลเป้าหมายได้โดยตรง เช่น การสร้างตารางข้อมูล การสอบถามระเบียบข้อมูล และการลบระเบียบข้อมูล เป็นต้น เพื่อไม่ให้ผู้ใช้เครื่องมือต้องเสียเวลาทำงานเหล่านี้ผ่านเครื่องมือของแต่ละฐานข้อมูลนั้นๆ
4. ปรับปรุงเครื่องมือสร้างข้อมูลทดสอบให้พิจารณาข้อบังคับ (Constraints) อื่นๆ ที่มีอยู่ในฐานข้อมูลเชิงสัมพันธ์ เช่น การกำหนดโดเมน (Domain) ให้เขตข้อมูลสามารถมีค่าตั้งแต่ 100 ถึง 200 เป็นต้น
5. ปรับปรุงเครื่องมือสร้างข้อมูลทดสอบให้ผู้ใช้สามารถเลือกได้ว่าต้องการเพิ่ม (Insert) หรือปรับปรุง (Update) ข้อมูลทดสอบในฐานข้อมูลเป้าหมาย ในกรณีที่ใช้เลือกบันทึกข้อมูลทดสอบลงฐานข้อมูลเป้าหมายโดยตรง

6.4 ผลงานที่เกี่ยวข้องกับการวิจัย

งานวิจัยนี้ได้รับคัดเลือกให้นำเสนอในงานประชุมวิชาการและตีพิมพ์ในเอกสาร “Proceedings of the Second International Conference on Information and Communication Technologies (ICT 2004)” ในระหว่างวันที่ 18-19 พฤศจิกายน พ.ศ.2547 โดยมีหัวข้องานวิจัยชื่อ “RealGen : A Test Data Generation Tool to Support Software Testing” สำหรับผลงานที่ตีพิมพ์ สามารถดูได้ที่ภาคผนวก ข



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

- [1] IEEE Computer Society. IEEE Standard Glossary of Software Engineering Terminology. 1983. ANSI/IEEE Std 729-1983.
- [2] Boris Beizer. Software Testing Techniques Second Edition. New York: Van Nostrand Reinhold, 1990.
- [3] วรรณชิต มาลัยวงศ์. พจนานุกรม CMM version 2. มีนาคม 2544, Available from: <http://www.swpark.or.th/cmm/>
- [4] ชวลิต จีระทีปดิสุนทร. XML Step by Step ฉบับภาษาไทย. กรุงเทพมหานคร:DLS, 2543.
- [5] Roger L. Costello. XML Schema Tutorial. September 2003, Available from: <http://www.xfront.com>
- [6] C. J. Date. An Introduction to Database Systems Sixth Edition. United States of America: Addison-Wesley Publishing Company, 1994.
- [7] Banner Software. Datatect 1.6. Available from: <http://www.datatect.com>
- [8] ภพพงศ์ สกุลพิพัฒนศิลป์. การพัฒนาเครื่องมือซอฟต์แวร์สำหรับสร้างข้อมูลทดสอบ (Development of a Software Tool for Generating Test Data). วิทยานิพนธ์ปริญญาามหาบัณฑิต สาขาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2544.
- [9] CANAM Software Labs. TurboData. Available from: <http://www.turbodata.ca>
- [10] สุภา ฉายแสง. การออกแบบและพัฒนาระบบสารสนเทศผู้ป่วยนอกของสถานพยาบาลรัฐวิสาหกิจ. วิทยานิพนธ์ปริญญาามหาบัณฑิต สาขาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2543.
- [11] พรพิมล ไทจันทร์. การออกแบบระบบช่วยตัดสินใจในการเลือกสถาบันเพื่อศึกษาต่อ. วิทยานิพนธ์ปริญญาามหาบัณฑิต สาขาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2539.



ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก

พจนานุกรมข้อมูลของฐานข้อมูลของเครื่องมือ

ภาคผนวกนี้แสดงพจนานุกรมข้อมูลของแต่ละตารางในฐานข้อมูลของเครื่องมือ

ตารางที่ ก-1 พจนานุกรมข้อมูลของตาราง DatabaseSchema

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
dbs_name	ชื่อสคีมาของฐานข้อมูล	varchar	100	ไม่	PK	Human Resource
dbs_desc	คำอธิบายสคีมาของฐานข้อมูล	varchar	255	ใช่		The database schema of employee information of human resource department

ตารางที่ ก-2 พจนานุกรมข้อมูลของตาราง DbTable

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
table_name	ชื่อตารางข้อมูล	varchar	100	ไม่	PK	Employee
table_dbs	ชื่อสคีมาของฐานข้อมูล	varchar	100	ไม่	PK,FK	Human Resource
table_dep_mode	รูปแบบการขึ้นต่อกันเชิงตรรกะ	varchar	10	ไม่		normal
table_gen_amt	จำนวนระเบียบข้อมูลที่ต้องการสร้าง	int		ใช่		1000

ตารางที่ ก-3 พจนานุกรมข้อมูลของตาราง DbField

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
field_name	ชื่อเขตข้อมูล	varchar	100	ไม่	PK	EMPNO
field_table	ชื่อตารางข้อมูล	varchar	100	ไม่	PK,FK	Employee
field_dbs	ชื่อสคีมาของฐานข้อมูล	varchar	100	ไม่	PK,FK	Human Resrc.

ตารางที่ ก-3 พจนานุกรมข้อมูลของตาราง DbField (ต่อ)

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
field_order	ลำดับของเขตข้อมูลในตาราง	int		ไม่		1
field_mod_size	ขนาดของเขตข้อมูล	int		ไม่		4
field_mod_scale	ขนาดของทศนิยม	int		ไม่		0
field_ident_flag	ค่าที่ชี้ว่าเขตข้อมูลเป็นคีย์หลักหรือเป็นส่วนหนึ่งของคีย์หลักของตารางข้อมูลหรือไม่ (0=ไม่, 1=ใช่)	int		ไม่		1
field_opt_flag	ค่าที่ชี้ว่าเขตข้อมูลจำเป็นต้องมีข้อมูลหรือไม่ (0=ใช่, 1=ไม่)	int		ไม่		0
field_gen_method	วิธีการสร้างข้อมูลทดสอบ (1=สุ่มสร้างอัตโนมัติ, 2=ค่าคงที่, 3=ค่าว่าง, 4=ค่าทั่วไป)	int		ใช่		1
field_from_value	ค่าเริ่มต้นของช่วงที่ใช้สุ่มข้อมูลทดสอบ	varchar	30	ใช่		1
field_to_value	ค่าสิ้นสุดของช่วงที่ใช้สุ่มข้อมูลทดสอบ	varchar	30	ใช่		9999
field_const_value	ค่าคงที่ที่ใช่เป็นค่าข้อมูลทดสอบ	varchar	255	ใช่		1000
field_category	กลุ่มค่าทั่วไปที่ใช้สร้างข้อมูลทดสอบ	varchar	200	ใช่	FK	employee no.
field_ref_name	ชื่อเขตข้อมูลที่ถูกรอ้างอิงถึง	varchar	100	ใช่	FK	EMPNO
field_ref_table	ชื่อตารางข้อมูล	varchar	100	ใช่	FK	Employee
field_ref_dbs	ชื่อสคีมาของฐานข้อมูล	varchar	100	ใช่	FK	Human Resrc.
Field_type	ชนิดข้อมูล	varchar	20	ไม่	FK	Integer

ตารางที่ ก-4 พจนานุกรมข้อมูลของตาราง FieldType

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
field_type_name	ชื่อตารางข้อมูล	varchar	20	ไม่	PK	Integer
field_type_size	ชื่อสคีมามาของฐานข้อมูล	int		ไม่		1
field_type_scale	รูปแบบการขึ้นต่อกันเชิงตรรกะ	int		ไม่		0

ตารางที่ ก-5 พจนานุกรมข้อมูลของตาราง NormalDependency

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
dep_table	ชื่อตารางข้อมูล	varchar	100	ไม่	PK,FK	Employee
dep_dbs	ชื่อสคีมามาของฐานข้อมูล	varchar	100	ไม่	PK,FK	Human Resource
dep_src_field	เขตข้อมูลต้นทางของเงื่อนไข	varchar	100	ไม่	PK	PREFIX
dep_dest_field	เขตข้อมูลปลายทางของเงื่อนไข	varchar	100	ไม่	PK	GENDER
condition_value	ค่าเงื่อนไข	varchar	100	ไม่	PK	Mr.
result_value	ค่าผลลัพธ์	varchar	100	ไม่		Male

ตารางที่ ก-6 พจนานุกรมข้อมูลของตาราง SQLDependency

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
dep_table	ชื่อตารางข้อมูล	varchar	100	ไม่	PK,FK	Employee
dep_dbs	ชื่อสคีมามาของฐานข้อมูล	varchar	100	ไม่	PK,FK	Human Resource
dep_sql	คำสั่งเอสคิวแอล	varchar	255	ไม่		select * from employee where if(prefix='Mr.', gender='Male, gender<>'Male')

ตารางที่ ก-7 พจนานุกรมข้อมูลของตาราง Category

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
cat_name	ชื่อกลุ่มค่าทั่วไป	varchar	200	ไม่	PK	First Name
cat_datatype	ชนิดข้อมูล	varchar	10	ไม่		Character
maxchar	ขนาดของค่าข้อมูลที่ยาวที่สุดในกลุ่มค่าทั่วไป	int		ไม่		9

ตารางที่ ก-8 พจนานุกรมข้อมูลของตาราง CategoryValue

ชื่อเขตข้อมูล	คำอธิบาย	ชนิดข้อมูล	ความยาว	ว่างเปล่า (Null?)	หมายเหตุ	ตัวอย่างข้อมูล
cat_name	ชื่อกลุ่มค่าทั่วไป	varchar	200	ไม่	PK,FK	First Name
cat_value	ค่าข้อมูล	varchar	255	ไม่	PK	Kritsana

ภาคผนวก ข

ตัวอย่างผลการทดสอบเครื่องมือสร้างข้อมูลทดสอบกับระบบสารสนเทศ
ผู้ป่วยนอกของสถานพยาบาลรัฐวิสาหกิจ

ตัวอย่างข้อมูลทดสอบที่สร้างโดยเครื่องมือสร้างข้อมูลทดสอบสำหรับระบบ
สารสนเทศผู้ป่วยนอกของสถานพยาบาลรัฐวิสาหกิจมีดังต่อไปนี้

ตารางที่ ข-1 ตารางสถานพยาบาล

CLINIC_CODE	CLINIC_NAME	U\$_NAME	LAST_UPDATE
8	สพบ. 50	i	2004-10-04 18:21:27
9	สพบ. 4	Z12iGJR4LY	2003-11-25 00:09:39
3	สพบ. 412	iDAskXolHam	2002-02-10 23:58:20
10	สพบ. 456	UJE37jXZeQ	2001-03-20 03:21:25
5	สพบ. 55	b	2002-05-02 15:17:50
1	สพบ. 2	XdcmPdCXC2LRF	2002-01-25 22:27:33
6	สพบ. 3	QMOEQDviGpwE73	2003-04-28 05:41:26
4	สพบ. 187	xJDXE952V	2002-12-15 11:44:56
2	สพบ. 713	zRtQz3JU2gBb	2004-04-16 06:15:29
7	สพบ. 180	yXbJo6	2002-07-24 22:29:07

ตารางที่ ข-2 ตารางกลุ่มโรค

DISEASE_CODE	DISEASE_NAME	U\$_NAME	LAST_UPDATE
0	โรคระบบประสาทและ อวัยวะสัมผัส	ISeLTXXB2dmRJ4d	2000-03-05 20:00:17
9	รูปวิปริตแต่กำเนิด	2XrTAB5hvw8btofG	2003-07-26 20:35:09
4	สาเหตุทางพยาธิสภาพและ มฤตภาพในการคลอด	J0zRcf8jpPQqFMb	2002-07-27 17:29:27
2	ภาวะแทรกซ้อนในการ ตั้งครรภ์และการคลอด	o6bHj2qxn	2003-03-17 12:26:23
7	โรคติดเชื้อและปรสิต	DYy2L3V4P	2000-11-30 09:57:45
3	ภาวะแปรปรวนทางจิต	ca8	2000-08-28 00:40:54
6	วางแผนครอบครัว	sbR9aUT5O8cO9C	2000-06-10 12:07:37
1	โรคผิวหนังและเนื้อเยื่อใต้ ผิวหนัง	WdQZZXnGk	2002-03-01 08:10:55

ตารางที่ ข-2 ตารางกลุ่มโรค (ต่อ)

DISEASE_CODE	DISEASE_NAME	U\$_NAME	LAST_UPDATE
5	รูปวิปริตแต่กำเนิด	014ZsfYbFoZpsC20B	2003-01-25 17:44:43
8	อื่นๆ	aTFZ5mGBQw	2003-10-31 09:22:34

ตารางที่ ข-3 ตารางการแพ้ยา

PERSON_ID	MEDI_CODE	U\$_NAME	LAST_UPDATE
E18	ZAS423	Y8XvGyBt0BMCQ9JU9PIK	2001-05-30 18:55:50
E08	MIU158	OasKESy	2002-04-17 22:41:02
E32	AES587	7zNw1DojY8KQ	2002-07-17 02:45:54
E75	CFR784	nekwBbwpAlUzexO	2002-01-15 23:38:32
E17	CFR784	P5	2004-01-30 01:46:17
E11	CFR784	dtW619MqPK	2002-11-25 10:16:42
E09	ZAS423	PftN6uZI92AXp	2002-06-09 18:05:14
E03	ZAS423	FtVT	2002-07-19 14:53:39

ตารางที่ ข-4 ตารางการแพ้สารอื่น

PERSON_ID	DESCRIPTION	U\$_NAME	LAST_UPDATE
G24	qgPMkYpj4mnCeJU	XqbuFkHL	2002-04-07 17:26:07
G65	Hmy92eocFexPzDt	Clr7hyhd4	2000-05-10 09:34:07
G11	2u14EC3nuymeg6m	9h8EpU2Lv8ieyuYn PQkJ	2001-03-29 11:54:15
G22	bNPAgtPZUQ70	bNPAgtPZUQ70	2002-12-28 19:27:33
G41	iJygUMLZU2OMuMu	g45dKw77zP0I20gn	2002-09-17 18:15:48
G23	sGB9ab3Ff2juNbA4Ni 9i0IIQN	GP5wme	2003-01-11 09:50:04
G01	D	Lulib8K	2002-11-14 17:45:05
G78	Xw5HKqsbxCA4XrCAn	MuuDJ9Ny	2003-05-15 10:35:39

ตารางที่ ข-5 ตารางประเภทการรักษา

REMEDY_CODE	REMEDY_NAME	U\$_NAME	LAST_UPDATE
4	ชอยยา	sOCS1MgKmoO17sDa	2001-07-05 11:56:23
3	โรคเฉพาะทาง	lpiML5Qy6fC44P	2003-06-28 09:14:32
1	โรคทั่วไป	pWcUIREEHq2edF6	1994-11-13 19:32:52
2	ทันตกรรม	DWYkr	2002-07-23 15:25:58

ตารางที่ ข-6 ตารางรายการจ่ายยา

PRESCRIPTION_CODE	MEDI_CODE	QUANTITY	INVOICE_NO	GET_DATE	U\$_NAME	LAST_UPDATE
04-0000010/47	MIU158	7099	04-0000032/47	2004-03-30	ChMAtsCM	2002-08-18 06:05:10
04-0000007/47	MIU158	8097	04-0000032/47	2004-03-30	UWei9ogm Pt	2001-04-20 16:39:52
04-0000010/47	MIU158	3484	04-0000032/47	2004-04-01	GK	2003-11-02 06:20:46
04-0000010/47	MIU158	7903	04-0000001/47	2004-10-03	OMx	1999-07-08 07:34:48
04-0000020/47	MIU158	2587	04-0000032/47	2004-04-01	JEra5UrEK LuTANoXsr	1999-03-28 13:12:59
04-0000020/47	MIU158	2404	04-0000032/47	2004-07-26	mx79C4rN6 W9C	2002-12-18 09:13:39
04-0000032/47	VCD511	7884	04-0000032/47	2004-04-01	Q	2000-03-30 10:31:17

ตารางที่ ข-7 ตารางระดับสั่งเพิ่มยา

MEDI_CODE	CLINIC_CODE	ADD_POINT	U\$_NAME	LAST_UPDATE
ZAS423	9	67192	GO3NuSD5L	2003-04-02 05:55:29
VCD511	10	42421	mM7uM	2002-08-03 21:43:38
AES587	2	40210	DrYHmntXqs	2000-01-24 20:20:30
VCD511	10	41400	vXFyHDDhq	2002-11-27 04:36:46
VCD511	9	56596	lrGiM	2000-09-25 19:52:18

ตารางที่ ข-7 ตารางระดับสั่งเพิ่มยา (ต่อ)

MEDI_CODE	CLINIC_CODE	ADD_POINT	U\$_NAME	LAST_UPDATE
ZAS423	5	7165	xRa9JMO	2000-07-15 16:15:42
ZAS423	5	32079	zzTYpX	2001-03-24 00:19:18

ตารางที่ ข-8 ตารางผู้จ่ายยา

PHARMACIST_CODE	PREFIX_CODE	FIRST_NAME_THAI	LAST_NAME_THAI	CLINIC_CODE	U\$_NAME	LAST_UPDATE
9	1	เซาวลิต	โสพรรณนกร	2	NZyLS9LW	2002-05-12 20:50:17
5	1	ณิศรา	รังษีวงศ์	8	9F5DQ3t	2000-10-10 19:19:51
6	4	เซาวลิต	จินทร์สด	9	e52iO9	1998-10-07 08:44:00
2	1	ศรชัย	จินดาสวัสดิ์	4	p4lG8Urq	1999-07-10 22:46:14
10	1	ธวัชชัย	วิตา	2	Od7zXKZn	1996-11-23 14:48:30
3	1	เอกรัตน์	จินดาสวัสดิ์	5	QENloNX	2002-01-16 18:42:37
4	1	ชินวร	พงษ์เทพนิวัติ	7	4rPoOqM	1999-06-25 07:02:06
7	1	เฉลิมทรัพย์	พละศักดิ์	8	wUNBSD	1996-10-03 17:48:43
1	4	ศักดิ์ดา	ทองเหลี่ยมนาถ	1	jOPN44G	2003-03-19 18:32:59
8	1	เฉลิมทรัพย์	วิตา	9	6mvcGjl	1999-04-23 15:46:48

ตารางที่ ข-9 ตารางเจ้าหน้าที่เวชระเบียน

RECORDER_CODE	PREFIX_CODE	FIRST_NAME_THAI	LAST_NAME_THAI	CLINIC_CODE	U\$_NAME	LAST_UPDATE
1	1	ทักษพล	จินดาสวัสดิ์	9	8vm1ma8dxy	2003-03-31 11:16:24
2	1	อนุวัตร	โสพรรณนากร	1	bC5BNS0iRI	2004-06-08 07:25:24
3	4	สุชาดา	วิตา	6	Do6sdStyK	2003-06-17 00:47:30
4	1	ทักษพล	อุตะเดช	5	wzX4Hk0y	2003-05-15 07:01:17

ตารางที่ ข-10 ตารางคลังยา

WAREHOUSE_CODE	WAREHOUSE_NAME	U\$_NAME	LAST_UPDATE
1	j7vLeQ9or9cbNFtZC	Bmkp4bSHY	2004-06-12 12:06:03
3	qbByg3XIkPSMk	yvG2fLeyB2sl	2004-01-26 02:02:51
4	VMmVhzauQzkx	X0k6o6SMEG	2004-04-11 16:29:43
2	79zNElqv18B5XI86	JgfaGIZTnS	2004-01-25 17:58:58

ตารางที่ ข-11 ตารางเดือน

MONTH_CODE	MONTH_NAME
1	January
2	February
3	March
4	April
5	May
6	June
7	July
8	August

ตารางที่ ข-11 ตารางเดือน (ต่อ)

MONTH_CODE	MONTH_NAME
9	September
10	October
11	November
12	December

ตารางที่ ข-12 ตารางกำหนดเลขที่ใบสั่งยา

RUN_NO	CLINIC_CODE	YEAR
04-0000011/47	9	2547
04-0000020/47	8	2547
04-0000011/47	1	2547
04-0000003/47	1	2547
04-0000020/47	4	2547
04-0000025/47	7	2547
04-0000011/47	7	2547
04-0000002/47	10	2547
04-0000010/47	8	2547
04-0000002/47	3	2547

ตารางที่ ข-13 ตารางประเภทบุคคล

PERSON_TYPE_CODE	PERSON_TYPE_NAME	U\$_NAME	LAST_UPDATE
4	ผู้เกษียณ	OiHGD1	2003-12-07 13:55:12
1	พนักงาน	7h2siYf	2003-10-17 16:39:01
3	ครอบครัว	qePBeEO	2003-04-12 06:18:32
2	ลูกจ้าง	hwm7	2003-08-28 22:25:31
5	บุคคลภายนอก	NwPsmE4	2003-03-22 17:39:50

ตารางที่ ข-14 ตารางความสัมพันธ์

RELATION_CODE	RELATION_NAME	U\$_NAME	LAST_UPDATE
3	คู่สมรส	KdnAIYE	2002-05-15 20:04:03
2	มารดา	bRnDI5	2001-04-15 11:02:09
4	บุตร	Fdbe1Uus8TPDV	1997-04-22 05:42:53
1	บิดา	0HGA2ewzh	2003-10-07 19:17:11
0	ตนเอง	YmFGbufxrXxAa8	2001-06-30 13:56:29

ตารางที่ ข-15 ตารางคำนำหน้า

PREFIX_CODE	PREFIX_NAME_FULL	PREFIX_NAME_ABBR	SEX_CODE
4	นาง	นาง	F
1	เด็กชาย	ดช.	M
2	นาย	นาย	M
5	เด็กหญิง	ดญ.	F
3	นางสาว	น.ส.	F

ตารางที่ ข-16 ตารางสถานภาพสมรส

MARRY_CODE	MARRY_NAME
1	ม่าย
5	หย่า
3	ไม่ระบุ
0	โสด
4	นอกสมรส
2	สมรส

ตารางที่ ข-17 ตารางกลุ่มผู้ใช้โปรแกรม

GROUP_CODE	PROGRAM_CODE	ACTIVE_STATUS
264	KArb93LE2	0
107	CNmwnblCY	1
18	LMMnrlrvC2	1
471	qjre0FrwJGl	1
72	tpKr	0
666	w6tYGeUsOON77RVM	0
946	tNbDMftWt2bBJHj7l4p	1
422	y2Fcx9pALM7t35df5YY9XniZ	0
829	2MTwKJ	0
795	4mma	1

ตารางที่ ข-18 ตารางผู้ใช้โปรแกรม

USER_NAME	PASSWORD	TYPE	USER_CODE	CLINIC_CODE	ACTIVE_STATUS
Koo	BCI4gzlu	Student	21	7	1
Robert	V8tYEI0	Lecturer	44	9	1
Nancy	IvyL10d	Foreman	98	4	1
Andrew	8GkYpjEc	Designer	80	2	1
Margaret	682rQ	Draftman	2	9	1
Laura	VHn0lbhv	System Analyst	4	1	0
Manny	tWV1LEzf	Salesman	42	10	1
Anne	yt89	Architect	25	8	0
Janet	oxpDnlSu	Foreman	94	3	0
Lorretta	eGB45	Engineer	67	4	0
Michael	60Ck	Doctor	66	3	0
Marcos	oCwQYhzB	Researcher	52	5	0
Steven	msGxADx	Dealer	50	4	1

ตารางที่ ข-18 ตารางผู้ใช้โปรแกรม (ต่อ)

USER_NAME	PASSWORD	TYPE	USER_CODE	CLINIC_CODE	ACTIVE_STATUS
Kritsana	bxRBHoQj	Draftman	51	10	1
Paullette	g2K4ZsYW	Designer	5	4	0

ตารางที่ ข-19 ตารางกลุ่มผู้ใช้ระบบ

GROUP_CODE	GROUP_NAME	ACTIVE_STATUS
3	Manager	1
9	Student	0
2	Salesman	1
6	System Analyst	1
5	Draftman	1
12	Dealer	1
1	Manager	1
7	Engineer	1
4	Doctor	1
8	Researcher	1
10	Student	1
11	Lecturer	1

ตารางที่ ข-20 ตารางประเภทผู้ใช้ระบบ

USER_NAME	GROUP_CODE	ACTIVE_STATUS
Janet	12	0
Andrew	9	0
Nancy	5	1
Marcos	2	1
Kritsana	8	1
Laura	5	1
Michael	1	0

ตารางที่ ข-20 ตารางประเภทผู้ใช้ระบบ (ต่อ)

USER_NAME	GROUP_CODE	ACTIVE_STATUS
Paullette	12	1
Manny	6	0
Margaret	11	0
Anne	9	0
Robert	4	0
Koo	4	1

ตารางที่ ข-21 ตารางข้อมูลโปรแกรม

PROGRAM_CODE	PROGRAM_NAME	MENU_CODE	ACTIVE_STATUS
sv824e0PNvW	ข้อมูลสถานพยาบาล	084nnpnhWoW	1
WeSLY	ข้อมูลกลุ่มโรค	RHi58zXGgKMoZlrG2E	0
kL0E025UYGUpdZ	ข้อมูลบุคคล	MPKfXII	1
wZGyvT5JuO6VuxfN	ข้อมูลคลังยา	uAm2uZNg8i	1
VpT0Sq	ข้อมูลสถานพยาบาล	a5axl0opr5Pp	0
3HdlX9zPUje	ข้อมูลพื้นฐาน	bA1aRjWCO	0
gQ8jpAnXp0xx	ข้อมูลยา	NZKpyZSPHF3g	1
UsCHPcYWLo0GR	ข้อมูลแพทย์	ZExu	1
ogtsQywGkxQVnN	ข้อมูลเจ้าหน้าที่	IW5M5ga1fTWU	0
yWGTTwbs701on2	ข้อมูลสถานพยาบาล	27bGr	0
mWswboQDLp3CRT	ข้อมูลคลังยา	DBpO1NGsj	1
oUVdumozA9lu	ข้อมูลบุคคล	pyKA4caOH	1

ตารางที่ ข-22 ตารางยาของสถานพยาบาล

INVOICE_NO	GET_DATE	MEDI_CODE	CLINIC_CODE	INVENT_CODE	ORDER_NO	LOT_NO	MFG_DATE	EXPI_DATE
04-0000032/47	2004-04-01	AES587	68	31	02/2547	XeEnguGoEeo	1997-10-02	2008-07-02
04-0000032/47	2004-07-26	MIU158	14	54	63/2547	QLt2XIB9E	1926-03-03	2002-01-16
04-0000032/47	2004-03-30	AES587	84	41	63/2547	QIX9bR8OP	1968-05-13	2001-05-27
04-0000045/47	2004-09-29	CFR784	16	65	78/2547	hG8PS1K	1920-09-28	2005-04-17
04-0000011/47	2004-11-19	VCD511	84	11	44/2547	dQFwTYU5H56D	1971-06-07	2008-09-12
04-0000011/47	2004-10-03	ZAS423	39	83	89/2547	t50IYhIv9xAc	1949-08-04	2001-09-10

ตารางที่ ข-22 ตารางยาของสถานพยาบาล (ต่อ)

MEDI_GET	MEDI_BALANCE	AMOUNT	AVG_PRICE	WAREHOUSE_CODE	U\$ _NAME	LAST_UPDATE
8394	3775.49	44943.27	3315.21	1	d4MoheYjA	2003-05-31 14:57:13
6210	3364.09	17312.03	4972.89	1	hBwG2G	2003-03-27 10:59:05
9182	4680.38	31005.81	4421.14	1	uRqvdBJ	2003-10-06 12:42:02
9405	3518.38	7792.10	4812.06	3	oqy5Gg	2003-12-04 10:45:36
7819	4923.93	9362.26	9362.26	2	Q7qNc73	2003-04-19 10:44:31
4196	3549.02	29865.39	2951.02	1	nMJrYIB9	2003-08-24 07:29:17

ตารางที่ ข-23 ตารางรายการจ่ายยา

PRESCRIPTION _CODE	MEDI _CODE	QUANTITY	MEDI _UNIT	UNIT	UNIT _USE	USAGE _CODE	USAGE	AMOUNT	STATU S	U\$ _NAME	LAST _UPDATE
04-0000032/47	VCD511	9838	50 ml.	1544	60 gm.	hOn	อม	98332.64	Yes	WkBc4b	2000-08-15 08:46:58
04-0000020/47	CFR784	7463	30 ml.	890	30 ml.	rr58	ทาบริเวณใบหน้า	60870.41	No	xnoC1W	2000-05-18 03:39:39
04-0000007/47	MIU158	6186	DROP	1049	DROP	r9	ทาน	86004.67	Yes	HvPS	2000-04-07 23:51:20
04-0000010/47	MIU158	5859	50 ml.	440	5 gm.	Z9I5mL	หยอด	94641.32	No	BWtu6H	2002-10-15 14:57:55
04-0000032/47	MIU158	5558	DROP	402	TABLET	jaq	หยอด	7240.98	Yes	xTDW1	2002-03-21 05:08:40

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-24 ตารางการใช้ยา

MEDI_USE_CODE	MEDI_USE_NAME_FULL	USAGE	QUANTITY	UNIT	U\$_NAME	LAST_UPDATE
8	ขัดบริเวณผม	ใช้หลังจากตื่น	79	50 gm.	d9Bsc9F	2004-02-14 13:47:42
rhX	ทาบริเวณคอ	ใช้ก่อนอาหารทุกมื้อ	72	50 ml.	8kx6MW	2004-10-08 19:22:52
643	ทาบริเวณคอ	ใช้เวลาคั้น	51	30 ml.	yYaQ5cULG	2003-01-15 15:52:05
FF	ทาบริเวณขา	ใช้ก่อนนอน	83	10 gm.	oGNnKaif	2003-11-29 22:49:36
ilR	ทาบริเวณขอบตา	ใช้หลังจากตื่น	37	60 ml.	oCqc7	2003-05-30 15:02:57
TsH	ทาบริเวณคอ	ใช้ก่อนอาหารทุกมื้อ	87	SPOON	cdgHx	2004-02-16 02:47:52

ตารางที่ ข-25 ตารางแพทย์

DOCTOR_CODE	PREFIX_CODE	FIRST_NAME_THAI	LAST_NAME_THAI	STATUS	REMEDY_CODE	LICENSE_CODE	SPECIALIZE_NAME
4	1	รัศมีทิพย์	ยอดมงคล	1	2	61600	จุมูก
3	3	สุชาติ	ตู้จินดา	1	4	70725	ผิวหนัง
1	2	ชวรณ	ทองเลี่ยมนาค	1	2	24669	ประสาท
2	4	จิรัชยา	ศิริพรกุลทรัพย์	1	1	58888	คอ
5	4	นงเยาว์	พงษ์เทพนิวัติ	2	2	15206	ผิวหนัง

ตารางที่ ข-25 ตารางแพทย์ (ต่อ)

CLINIC_CODE	WORK_DAY	ISSUE_DATE	EXPI_DATE	U\$_NAME	LAST_UPDATE
7	ทุกวันพฤหัสบดี	2000-03-09	2004-04-29	77H4Axxg	2004-01-31 22:14:16
8	ทุกวันพุธ	2001-08-29	2004-04-15	dKM	2004-06-26 21:10:41
10	ทุกวันอังคาร	2001-11-02	2004-11-19	VD	2004-09-01 21:24:34
10	ทุกวันศุกร์	2001-06-05	2004-08-10	lp1AJSbl	2004-07-07 13:20:09
3	ทุกวันจันทร์	2000-02-11	2004-07-17	Zagr	2004-01-07 11:46:18

จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-26 ตารางการนัดแพทย์

APPOINT _DATE	PERSON _CODE	PERSON _ID	PERSON _TYPE _CODE	RELATION _CODE	CLINIC _CODE	DOCTOR _CODE	REMEDY _CODE	REMARK	U\$ _NAME	LAST _UPDATE
2001-05-07	70265	E15	1	2	3	F	3	รอผลวินิจฉัย	ZLsJ9OQ	2001-11-08 05:41:25
2002-06-14	96586	E20	5	2	2	i0	4	รอผลวินิจฉัย	G	2000-02-03 12:49:59
2002-10-09	25560	E18	5	4	3	F	1	เลื่อนนัด ออกไป 1 เดือน	yidja5r	1999-12-14 23:35:29
2002-06-11	80864	E02	3	1	6	2w	2	เลื่อนนัด ออกไป 1 สัปดาห์	W	1999-07-19 19:44:19
2001-04-26	70265	E15	1	2	4	2w	1	รักษากรณี เร่งด่วน	YIUriHxm	1999-06-16 23:12:22
2004-01-01	25560	E18	5	4	6	T5	2	รักษากรณี เร่งด่วน	BzcXc	1999-02-26 05:55:29

ตารางที่ ๗-27 ตารางยา

MEDI _CODE	MEDI _NAME	GENERIC _NAME	MEDI _UNIT	MEDI _USE_CODE	SYMPTOM	WAREHOUSE _CODE	INVENT _CODE	U\$ _NAME	LAST _UPDATE
ZAS423	SALBUTAMOL SYRUP	ยาธาตุ	25 gm.	iIR	หืด	4	BNeFfdb4z	LQseiAX	1999-10-28 17:52:08
AES587	ARACAF SYRUP	ยาแก้ไอ	DROP	rhX	หืด	3	3xCp1vB	NLAju	1999-02-16 15:16:20
MIU158	EFERINE 5 mg.	ยาแก้ไอ	5 gm.	FF	พ่น	2	C	bk5svo	2002-01-27 08:54:59
VCD511	ANTA 0.5 mg. TAB	ยาน้ำ	30 ml.	643	แก้ท้องอืด	4	BBfJRlbXA	PD3idt3	2000-09-22 02:03:33
CFR784	HERBIE 30 mg. TAB	ยาแดง	60 gm.	FF	อาเจียน	4	8taSnEFv		2000-03-31 06:55:48

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-28 ตารางประวัติการรักษา

PERSON _ID	SEQ _NO	REMEDY _TYPE	PERSON _CODE	PERSON _TYPE_CODE	SEX _CODE	PRESCRIPTION _CODE	RELATION _CODE	DIAG_DATE
E17	2	2	98092	4	M	04-0000010/47	1	2000-01-14
E10	1	1	80864	3	M	04-0000032/47	1	2003-05-03
E07	7	1	25560	5	F	04-0000010/47	4	2002-01-08
E10	3	3	98092	4	M	04-0000002/47	1	2002-06-06
E19	4	4	25560	5	F	04-0000002/47	4	2003-08-26

ตารางที่ ข-28 ตารางประวัติการรักษา (ต่อ)

DIAG_INFO	DISEASE_CODE	DOCTOR_CODE	CLINIC_CODE	FLAG	U\$ _NAME	LAST_UPDATE
bVtkf1r8s9vjX	8	i0	10	0	Vawew	2004-01-14 04:51:02
dPs	10	2w	7	1	TyoDcK	2002-08-21 16:22:08
xpp7WuW4t	7	F	5	0	4jdiymAD	2004-03-14 07:44:18
3VHIJggBEeb	8	i0	9	0	Huz	2003-10-17 01:25:52
krOZqEPzYYjB	7	F	6	1	DZ04o5oBzV	2002-02-10 04:22:21

จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-29 ตารางใบสั่งยา

PRESCRIPTION_CODE	DISEASE_CODE	DOCTOR_CODE	PHARMACIST_CODE	CLINIC_CODE	REGIS_DATE
04-0000020/47	10	i0	6	3	1923-04-02
04-0000010/47	2	i0	8	10	1943-05-15
04-0000020/47	2	i0	3	3	1995-10-31
04-0000020/47	6	z	6	5	1906-03-26
04-0000020/47	2	T5	6	10	1914-04-18

ตารางที่ ข-29 ตารางใบสั่งยา (ต่อ)

MEDICINE_FEE	DOCTOR_FEE	OTHER_FEE	STATUS	PRINT_STATUS	U\$_NAME	LAST_UPDATE
96.33	380.04	238.70	1	Yes	Cekwa1d	2004-01-09 12:11:43
73.33	248.76	363.31	1	Yes	oGCGtm	2003-02-24 00:00:37
55.90	330.11	276.66	2	Yes	JA10nZch0	2003-03-14 10:17:12
61.04	229.41	278.98	1	Yes	M	2003-01-23 00:58:11
88.34	394.87	334.08	2	Yes	yj3Grvlq	2003-06-27 23:05:39

จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-30 ตารางลงทะเบียนผู้ป่วย

PRESCRIPTION _CODE	REGIS _DATE	PERSON _CODE	PERSON _ID	PATIENT _NAME	BIRTH _DATE	PERSON _TYPE_CODE	RELATION _CODE	SEX _CODE	REMEDY _CODE
04-0000011/47	2003-06-16	91616	E15	กั้น	1958-02-07	1	3	F	3
04-0000007/47	2004-01-30	96586	E20	ธวัชชัย	1976-05-22	5	2	F	3
04-0000032/47	2003-09-20	25560	E18	อนุวัตร	1927-08-03	5	4	F	3
04-0000020/47	2003-04-19	70265	E15	นันท์ทา	1985-05-29	1	2	M	2
04-0000020/47	2004-04-28	98092	E06	กฤษฎา	1904-04-29	4	1	M	3

ตารางที่ ข-30 ตารางลงทะเบียนผู้ป่วย (ต่อ)

CLINIC _CODE	DIV_LINE _NAME _ABBR	DIV_COWORK _NAME _ABBR	DIVISION _NAME _ABBR	SECTION _NAME _ABBR	SUBSECTION _NAME _ABBR	QUEUE	STATUS	U\$_NAME	LAST _UPDATE
5	ผฝกน	รผฝกธ	สนงฟฟส	กปท	สดพ	57	2	36kpAgh	2001-04-19 00:02:43
1	รทน	รผฝกน	สนงปปส	กพน	สดพ	84	4	kGTGgaZ	2000-01-19 00:54:46
8	ผฝกธ	รผฝบ	สนงฟฟส	กฟภ	สดพ	63	2	zuMYzC48	2000-05-07 09:47:53
6	ผฝกน	รผฝกน	สนงปปย	กปน	สบรง	72	1	fc3HcZ2	2002-01-21 20:49:03
9	ผฝกน	รผฝกน	สนงปปส	กก	สบรง	62	1	EYZ5	2001-09-25 22:04:21

ตารางที่ ข-31 ตารางพนักงานและลูกจ้าง

PERSON_ID	PERSON_CODE	PERSON_TYPE_CODE	PREFIX_CODE_THAI	PREFIX_NAME_FULL	FIRST_NAME_THAI	LAST_NAME_THAI	SEX_CODE	HEIGHT	WEIGHT
E19	66614	2	1	เด็กหญิง	ศราวุธ	พละศักดิ์	F	160.13	72.69
E44	62071	4	1	เด็กหญิง	คมสัน	วิตา	F	169.21	73.32
E05	71011	1	1	เด็กหญิง	เอกรัตน์	พงษ์เทพนิวัติ	F	172.09	65.38
E12	82294	2	1	เด็กชาย	ธิษณา	พละศักดิ์	M	180.16	62.34
E01	62258	3	1	นาย	นฤชัย	ศิริพรกุลทรัพย์	M	178.54	67.46

ตารางที่ ข-31 ตารางพนักงานและลูกจ้าง (ต่อ)

BIRTH_DATE	MARRY_CODE	BLOOD_GROUP	MINISTRY_CODE	MINISTRY_NAME_FULL	MINISTRY_NAME_ABBR	DEPTMENT_CODE	DEPARTMENT_NAME_FULL	DEPARTMENT_NAME_ABBR
1981-11-05	5	A	98550	กระทรวงคมนาคม	กค	53167	กรมที่ดิน	กฟภ
1994-11-15	1	B	81358	กระทรวงแรงงานฯ	กว	54838	การไฟฟ้าส่วนท้องถิ่น	กปน
1914-08-22	1	A	86118	กระทรวงคมนาคม	กพ	57742	การประปาส่วนภูมิภาค	กปท
1976-04-23	0	B	46657	กระทรวงคมนาคม	กว	77074	การประปาส่วนภูมิภาค	กปท
1973-11-10	0	O	63879	กระทรวงศึกษาธิการ	กค	46828	การไฟฟ้าส่วนภูมิภาค	กย

ตารางที่ ข-31 ตารางพนักงานและลูกจ้าง (ต่อ)

DIV_LINE_CODE	DIV_LINE_NAME_FULL	DIV_LINE_NAME_ABBR	DIV_CODE	DIV_COWORK_NAME_FULL	DIV_COWORK_NAME_ABBR	DIVISION_CODE	DIVISION_NAME_FULL	DIVISION_NAME_ABBR
65698	ผู้ช่วยฝ่ายบริหาร	ผฝภค	80097	รองผู้ช่วยฝ่ายการเงิน	รผฝภค	56974	สำนักงานการประปาสาขา	สนงปปย
48609	รองหัวหน้า	ผฝบ	22843	รองผู้ช่วยฝ่ายบริหาร	รผฝบ	30302	สำนักงานการประปาย่อย	สนงปปส
75359	รองหัวหน้า	ผฝบ	66433	รองผู้ช่วยฝ่ายบริหาร	รผฝภค	36070	สำนักงานการประปาย่อย	สนงฟฟย
48573	ผู้ช่วยฝ่ายการเงิน	หน	99116	รองผู้ช่วยฝ่ายการธุรการ	รผฝภค	62461	สำนักงานการประปาย่อย	สนงฟฟย
57997	ผู้ช่วยฝ่ายการเงิน	ผฝภค	91493	รองผู้ช่วยฝ่ายการเงิน	รผฝบ	80510	สำนักงานการประปาย่อย	สนงฟฟย

ตารางที่ ข-31 ตารางพนักงานและลูกจ้าง (ต่อ)

SECTION_CODE	SECTION_NAME_FULL	SECTION_NAME_ABBR	SUBSECTION_CODE	SUBSECTION_NAME_FULL	SUBSECTION_NAME_ABBR	U\$ _NAME	LAST_DATE
83035	กองงบประมาณ	กง	56252	ส่วนดูแลพัสดุ	สบร	vr41Zx	2003-10-23 02:13:51
78059	กองเครื่องจักรกล	กบ	74364	ส่วนรักษาความปลอดภัย	สบร	l4fBHPoK	2002-06-03 02:45:01
64009	กองดูแลสวัสดิการ	กค	25458	ส่วนบริการงานโรงงาน	สดพ	pIMRME	2002-08-29 05:58:21
68955	กองดูแลสวัสดิการ	กค	67754	ส่วนดูแลพัสดุ	สบร	C4Hira	2003-01-09 18:27:50
85574	กองเครื่องจักรกล	กค	93128	ส่วนบริการงานโรงงาน	สดพ	5K6O6tUQ	2001-04-13 11:31:12

ตารางที่ ข-32 ตารางครอบครัว

PERSONAL _ID	PERSONAL _CODE	PERSON _TYPE _CODE	PREFIX _CODE _THAI	FIRST _NAME_THAI	LAST _NAME_THAI	SEX _CODE	BIRTH _DATE	PARENT _NAME	SON _STATUS	MARRY _STATUS
G11	E18	1	1	ทศพร	ศิริพรกุลทรัพย์	M	1915-08-29	ชินวร	1	3
G14	E44	2	4	กรรณิการ์	ชินอานุภาพ	F	1947-05-19	นงเยาว์	3	1
G78	E12	1	1	สุวิสาข์	ทองเลี่ยมนาค	M	1987-07-06	วิษณา	2	5
G22	E03	4	1	นฤชัย	ชมเดช	M	1948-06-22	ทศพร	8	4
G77	E19	3	1	นงเยาว์	แก้วชนะ	M	1929-02-16	ศักดิ์ดา	8	3

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-33 ตารางผู้เกษียณ

PERSONAL_ID	PERSONAL_CODE	PERSON_TYPE_CODE	PREFIX_CODE_THAI	FIRST_NAME_THAI	LAST_NAME_THAI	SEX_CODE
G45	R12	5	1	ธิษณา	โสพรรณากร	M
G22	R56	5	1	วรวิฑู	พงษ์เทพนิวัติ	M
G01	R77	2	1	ราชันย์	ทองเลี่ยมนาค	M
G14	R58	2	4	สุชาดา	รังษีวงศ์	F
G14	R45	1	1	สุวิสาข์	โสพรรณากร	M

ตารางที่ ข-33 ตารางผู้เกษียณ (ต่อ)

BIRTH_DATE	DATE_RETIRED	DATE_OUT	MARRY_CODE	HEIGHT	WEIGHT
1930-09-08	1940-06-01	1995-12-26	1	166.24	62.51
1954-03-09	1971-12-12	1931-01-19	4	175.12	63.74
1954-07-08	1963-06-29	1971-09-30	3	187.12	73.20
1914-01-05	1995-04-05	1994-08-05	5	179.67	72.22
1974-12-21	1904-05-14	1929-06-15	4	165.49	68.66

จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-34 ตารางบุคคลภายนอก

PERSON_ID	PERSON_CODE	PERSON_TYPE_CODE	PREFIX_CODE_THAI	FIRST_NAME_THAI	LAST_NAME_THAI	SEX_CODE
G41	G45	4	1	ทิมพร	จันทร์สด	M
G11	G11	2	1	ทิมพร	พลศักดิ์	M
G14	G77	3	1	สุดชดา	วิตา	F
G41	G78	5	4	เมลานี	ตู้จินดา	F
G65	G65	3	1	กิตติกร	วิตา	F

ตารางที่ ข-34 ตารางบุคคลภายนอก (ต่อ)

HEIGHT	WEIGHT	BIRTH_DATE	MARRY_CODE	BLOOD_GROUP	U\$_NAME	LAST_UPDATE
175.40	76.16	1976-12-05	4	A	qkmEvuv6l	1999-12-12 23:07:42
167.49	57.81	1988-07-28	2	B	p3Dy	1996-04-20 18:31:44
174.72	50.56	1942-11-04	2	A	l9jnXZzVTw	1999-07-06 13:07:39
164.26	68.16	1986-03-25	4	AB	WHDPPrqwk8rvh	1999-06-02 00:55:09
160.83	52.64	1918-05-21	1	AB	OFichjaG8gH	2000-01-24 08:55:22

จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ข-35 ตารางที่อยู่บุคคลภายนอก

PERSON_CODE	ADDRESS	STREET	SOI	MOO	SUB_DISTRICT
G45	55	ถนนสาลี	ซอย 20	13	RACHA
G11	77	ถนนจมีนศรี	ซอย 2	9	KAI
G65	9	ถนนจมีนศรี	ซอย 3	2	NOI
G77	22	ถนนยักษ์ใหญ่	ซอย 15	2	KAI
G01	52	ถนนจรัญสนิทวงศ์	ซอย 1	1	SINGTOO

ตารางที่ ข-35 ตารางที่อยู่บุคคลภายนอก (ต่อ)

DISTRICT	PROVINCE	POSTCODE	TELEPHONE	U\$_NAME	LAST_UPDATE
AUTHONG	อุบลราชธานี	79485	045321518	21TyGNbb9	2002-08-02 18:11:24
BANGPHAMA	เชียงใหม่	57739	045369852	NTcJQfvbOGZa	2002-05-12 10:55:56
SRIPRAJAN	กาญจนบุรี	79485	045321518	53VbQw2bBA	2002-06-12 01:53:49
AUTHONG	มุกดาหาร	19053	032852417	7Pdp4q4ZE	2003-12-30 02:49:40
SONGPINONG	ปทุมธานี	57739	045321518	yaN84iwQ4Mg	2002-01-09 09:13:59

จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ค

ตัวอย่างผลการทดสอบเครื่องมือสร้างข้อมูลทดสอบกับระบบช่วยตัดสินใจใน การเลือกสถาบันเพื่อศึกษาต่อ

ตัวอย่างข้อมูลทดสอบที่สร้างโดยเครื่องมือสร้างข้อมูลทดสอบสำหรับระบบช่วย
ตัดสินใจในการเลือกสถาบันเพื่อศึกษาต่อมีดังต่อไปนี้

ตารางที่ ค-1 ตาราง T_CITYSPEC

CITYCODE	CITYNAME	URBAN	WEATHER
7304	LINCOLN	3	3
7587	CAMDEN	3	3
4028	LOWELL	4	1
7700	BOWIE	5	4
8144	CAMBRIGE	5	4
1217	LIVONIA	4	2
4009	DULUTH	2	1
6494	LANHAM	5	1
5039	SALEM	1	2
8944	BROOKLINE	0	2

ตารางที่ ค-2 ตาราง T_NEARCITY

CITYCODE	NEARCODE	NEARORDER
8944	8144	1
8144	7304	2
5039	1217	3
1217	8144	3
7700	4028	1
6494	4009	1
4009	6494	1

ตารางที่ ค-3 ตาราง T_RANK

MAJOR	FIELDID	ORDER	UNAME	STATENAME
Other	83549	40	United State University	New South Wales
Science	57104	36	Canton University	Georgia
Science	99851	74	Mary University	Georgia
Art	68178	76	United State University	Victoria
Society	25130	11	Hopkin University	Alaska
Business	13793	74	University of Luton	Idaho
Humanity	15525	95	Montael University	Hawaii
Humanity	53130	0	Bridgewater University	Alabama
Humanity	43961	63	University of Loyland	Queensland
Other	93433	89	University of Martin	Georgia
Humanity	39095	35	New University	Connecticut
Other	79565	95	Stockport University	Hawaii

ตารางที่ ค-4 ตาราง T_CODEU

UCODE	UABBR	UNAME
3021	C	University of Cornell
9118	Jb	Bridgewater University
9866	Olw	Temple University
9143	L	Hopkin University
1656	Qvar7	Montael University
7107	x2	Boston University
3889	sB	Ada University
3429	I	Liengh University
4691	ZsdWZ	University of Loyland
4777	ezMc	Paul University
3888	I	Jackson University
2068	WDb7OI	Stanford University

ตารางที่ ค-5 ตาราง T_CODEFIELD

FIELDID	FIELDNAME
13793	Higher Education
15525	Anatomy
53130	Information System
68178	Computer Engineering
83549	Optic
75278	Information System
93433	Information System
43961	Theory
79565	Higher Education
99851	Organizational Behavior
25130	Wood Sciene
57104	Elementary Education
39095	Accounting

ตารางที่ ค-6 ตาราง T_STATE

ABBRNAME	STATENAME	CITIES	NEARSTATE
Rj	Illinois	7304	e
g7	Alaska	4028	SV
Ks	Connecticut	7304	SV
j	Illinois	4028	g7
6	Connecticut	7304	g7
w	Alaska	5039	v
GR	Tasmania	5039	SV
e	Connecticut	4009	Ks
V	Delaware	7700	GR
SV	Victoria	8944	HA
2	Queensland	1217	g7
HA	Idaho	4028	6

ตารางที่ ค-7 ตาราง T_FIELD

LEVEL	FIELDID	UCODE	CITYCODE	RANKING	ST_RATIO	FINISH	GRADUATE	GET	TOEFL_LOW	TOEFL_HIGH
0	25130	8144	9143	14	53.84	39.19	GRADUATE	10.98	525	718
2	39095	1217	3429	2	30.28	31.47	UNDERGRADUATE	12.83	522	743
2	57104	8144	9143	83	78.84	30.93	UNDERGRADUATE	96.17	574	750
1	39095	8144	3021	79	47.68	89.78	GRADUATE	68.61	593	747
0	83549	7700	3888	32	36.21	85.36	UNDERGRADUATE	2.42	534	755

ตารางที่ ค-7 ตาราง T_FIELD (ต่อ)

GMAT_LOW	GMAT_HIGH	GRE_A_LOW	GRE_A_HIGH	GRE_Q_LOW	GRE_Q_HIGH	GRE_V_LOW	GRE_V_HIGH	GPA_LOW	GPA_HIGH
277	494	480	749	432	735	411	720	2.14	3.98
257	716	468	739	463	708	468	733	2.62	3.87
251	809	437	753	481	742	483	708	2.13	3.25
277	737	434	768	455	713	456	780	2.72	3.44
230	425	453	756	498	766	457	798	2.56	3.09

จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ ค-8 ตาราง T_UDETAIL

UCODE	CITYCODE	TERMTYPE	COST	BLACKSTUD	ASIASTUD	THAISTUD	ADDRESS1
3888	7700	S	6649393.67	53.78	43.34	23.47	41/2 Ploenchit Rd.
3429	1217	Q	1465602.51	63.93	45.80	25.14	55/8 Jarunsanitwong Rd.
4691	7587	S	8286510.74	51.19	45.73	22.41	12/2 Oriental Avenue
3021	8144	T	6471830.00	59.28	37.57	10.57	34/1 Jarunsanitwong Rd.
3889	7700	T	4811159.02	41.86	47.07	18.36	57/8 Chert Wudthakas Rd.
9143	8144	S	4178444.76	52.39	41.91	12.49	9/2 Jarunsanitwong Rd.

ตารางที่ ค-8 ตาราง T_UDETAIL (ต่อ)

ADDRESS2	ADDRESS3	TEL	E-MAIL	URL	DETAIL1	DETAIL2	DETAIL3
Chanthaburi	Thailand	036254789	kgundum@yahoo.com	esREHrA	c15yUO3Q	FkcfXBVdfp	vM073VFz
Bureerum	Thailand	045369852	tam_cs@hotmail.com	DDeGcH	JJAm5u	Ht5dQ	hw7adh
Rayong	Thailand	042247612	BSL@thaimail.com	XEwRw	zhY2Dj	Ckdx54	4hiHyXISx
Ranong	Thailand	032852417	keechiang@hotmail.com	CNFr	MqF	7KKT2	0nQCnVApX
Pijit	Thailand	032852417	pray14@hotmail.com	WeQtYU	4XlJAbNu	498gxNYM	bRXk7IbWx
Yala	Thailand	045323048	judy_jing@yahoo.com	ZAPouiw	vX8brBR	2lLxwaYo	khYDeq7V

ตารางที่ ค-9 ตาราง T_MARK

FIELD ID	UCODE	CITY CODE	M _COST	M _GRADUATE	M _EXAM	M _FIELD	M _TERM	M _NATION	M _STAT	M _RANK	M _LOC	M _URBAN	M _TOTAL
39095	1217	3429	19	43	62	67	93	24	2	23	71	31	75
57104	8144	9143	37	99	53	54	61	61	61	83	76	30	55
25130	8144	9143	69	77	94	81	70	1	82	53	62	57	58
83549	7700	3888	20	83	43	86	53	77	61	92	27	31	99
39095	8144	3021	49	57	72	61	57	68	15	48	44	27	55

ตารางที่ ค-10 ตาราง T_ADMIN

PASSWORD	W_COST	W_TOEFL	W_GPA	W_GMAT	W_GRE	W_GRADUATE
M1HXGxPzdvxXXik7	81	49	18	79	92	76
ERcy6Rd5LjAxY	80	4	16	52	76	27
fGWb	27	6	93	1	82	14
C	34	42	75	70	1	54

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ง

ตัวอย่างผลการทดสอบเครื่องมือสร้างข้อมูลทดสอบวัดเวลาตอบสนอง (Response Time) ในการสร้างข้อมูลทดสอบ

ตัวอย่างข้อมูลทดสอบที่สร้างโดยเครื่องมือสร้างข้อมูลทดสอบสำหรับตารางที่ใช้ทดสอบเพื่อวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบมีดังต่อไปนี้

ตารางที่ ง-1 ตารางแสดงข้อมูลทดสอบที่ได้จากการทดสอบวัดเวลาตอบสนองในการสร้างข้อมูลทดสอบ

EMPNO	PREFIX	ENAME	GENDER	JOB	MGR	HIREDATE	SAL	COMM	DEPT
2106	Mrs.	Steven	Female	Archetect	5075	2004-01-12	82473.45	77927.33	69
1744	Mrs.	Robert	Female	Foreman	7837	2004-11-13	88809.98	64420.76	4
1718	Mr.	Robert	Male	Programmer	7837	2004-09-02	97017.44	28275.02	57
4396	Mr.	Robert	Male	Doctor	1744	2004-01-14	8302.25	88237.89	36
4114	Mrs.	Grover	Female	Lecturer	1744	2004-08-09	16901.21	13033.94	55
5075	Mrs.	Anne	Female	Programmer	1718	2004-03-07	80183.09	10257.54	70
3236	Miss	Koo	Female	Programmer	4114	2004-09-08	15207.15	21880.29	42
7837	Miss	Nancy	Female	Researcher	4396	2004-11-23	83808.87	98380.48	63
2102	Mr.	Andrew	Male	Student	4114	2004-02-03	18219.07	10262.86	38

ภาคผนวก จ

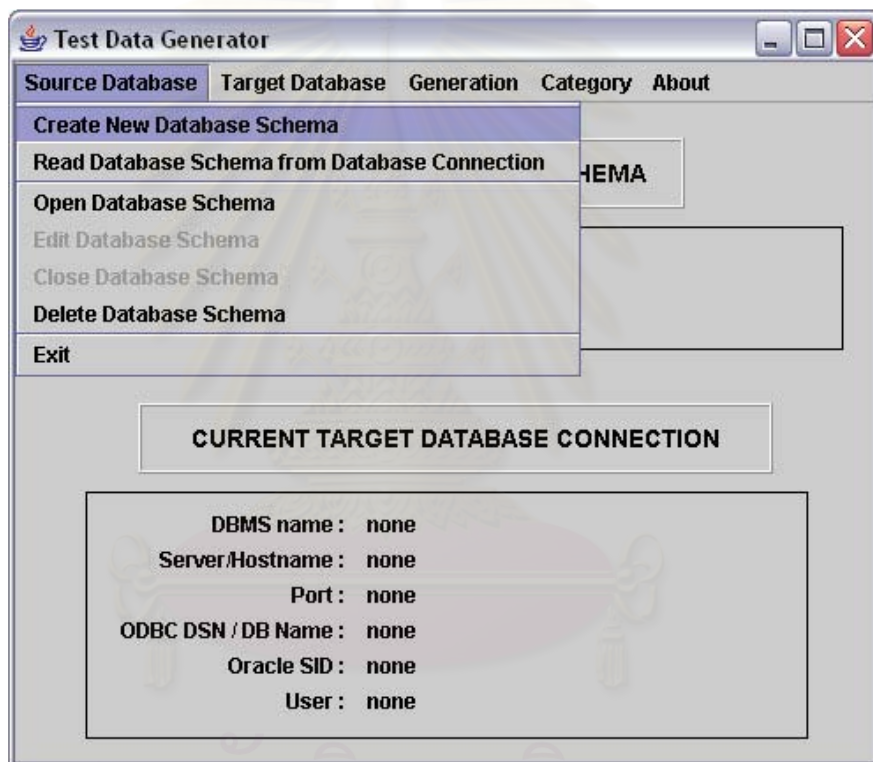
คู่มือการใช้งานเครื่องมือสร้างข้อมูลทดสอบ

การใช้งานเครื่องมือสร้างข้อมูลแบ่งออกเป็น 4 ส่วนดังนี้

1. การจัดการสคีมาของฐานข้อมูล

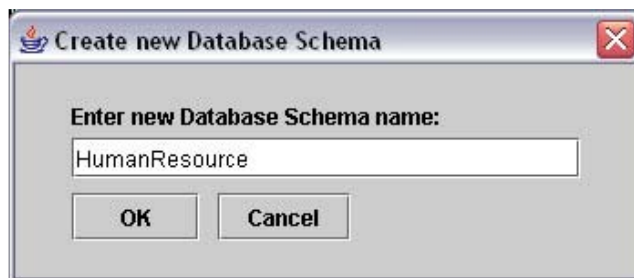
1.1 การสร้างสคีมาของฐานข้อมูลด้วยตัวเอง จะมีขั้นตอนดังนี้

1.1.1 เลือกที่เมนู Source Database และเลือกที่เมนูย่อย Create New Database Schema ดังรูปที่ จ-1



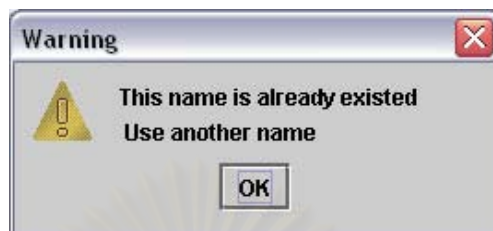
รูปที่ จ-1 การสร้างสคีมาของฐานข้อมูลด้วยตัวเอง

1.1.2 กำหนดชื่อให้สคีมาของฐานข้อมูลที่กำลังจะสร้าง แล้วจึงกดปุ่ม OK เพื่อยืนยันการสร้างสคีมาของฐานข้อมูล ดังรูปที่ จ-2



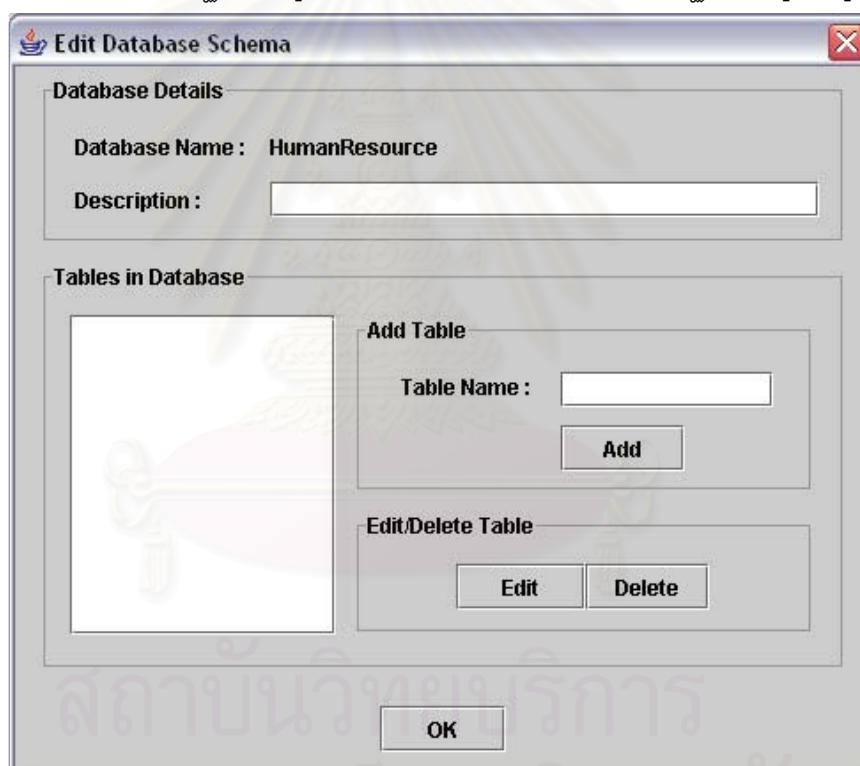
รูปที่ จ-2 การกำหนดชื่อสคีมาของฐานข้อมูล

ถ้าชื่อสคีมาของฐานข้อมูลที่กำหนดมาซ้ำกับชื่อของสคีมาของฐานข้อมูลที่มีอยู่แล้วในฐานข้อมูลของเครื่องมือ จะมีข้อความเตือนดังรูปที่ ๑-3 ซึ่งผู้ใช้ต้องกำหนดชื่อสคีมาของฐานข้อมูลใหม่ที่ไม่ซ้ำกับชื่อที่มีอยู่เดิม



รูปที่ ๑-3 ข้อความแจ้งเตือนชื่อสคีมาของฐานข้อมูลซ้ำ

แต่ถ้าชื่อสคีมาของฐานข้อมูลที่กำหนดมาไม่ซ้ำกับชื่อของสคีมาของฐานข้อมูลที่มีอยู่ เครื่องจะสร้างสคีมาของฐานข้อมูลให้แล้วแสดงหน้าต่างสคีมาของฐานข้อมูลดังรูปที่ ๑-4



รูปที่ ๑-4 หน้าแก้ไขสคีมาของฐานข้อมูล

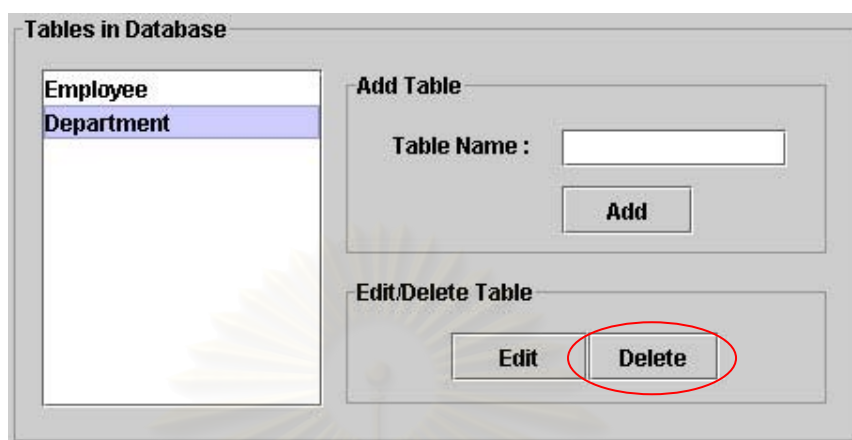
1.1.3 เพิ่มตารางข้อมูลให้กับสคีมาของฐานข้อมูล โดยการกำหนดชื่อตารางข้อมูลแล้วกดปุ่ม Add ดังรูปที่ ๑-5



รูปที่ ๑-5 การเพิ่มตารางข้อมูล

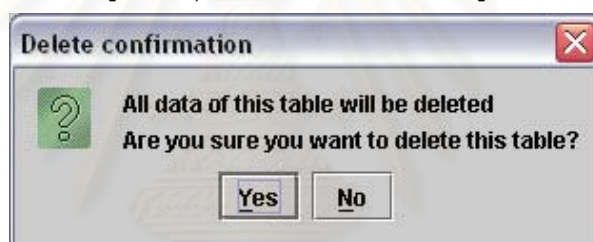
1.1.4 ลบตารางข้อมูลที่ไม่ต้องการ โดยเลือกตารางข้อมูลที่จะลบ แล้วกดปุ่ม Delete ดัง

รูปที่ ๑-6



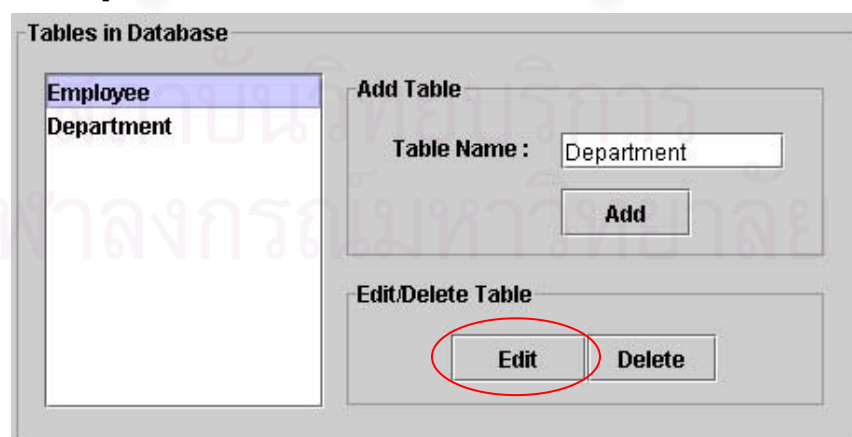
รูปที่ ๑-6 การเลือกลบตารางข้อมูล

จากนั้นเครื่องมือจะแสดงข้อความเพื่อขอการยืนยันจากผู้ใช้ในการลบตารางข้อมูล ดังรูปที่ ๑-7 ซึ่งให้ผู้ใช้กดปุ่ม Yes เพื่อลบตารางข้อมูลที่เลือก



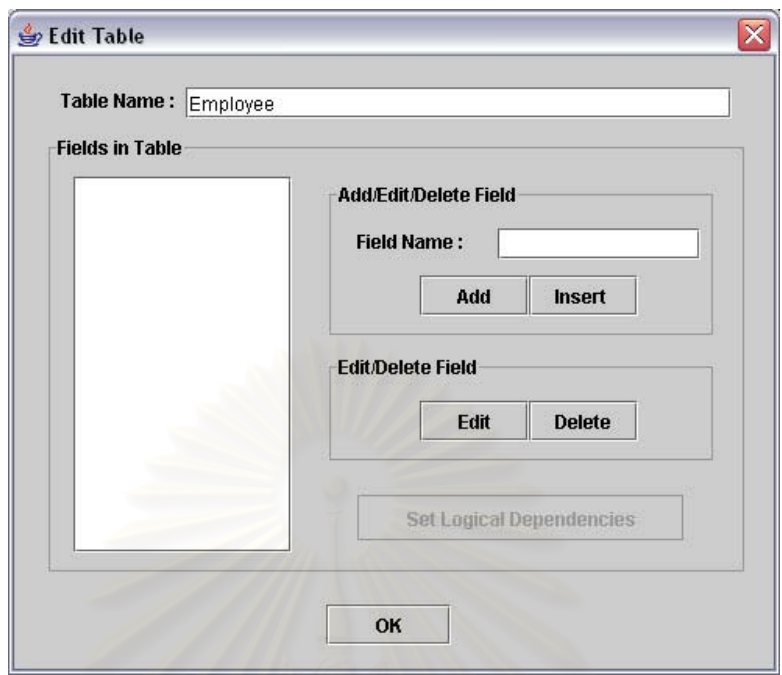
รูปที่ ๑-7 หน้าขอคำยืนยันในการลบตารางข้อมูล

1.1.5 เพิ่มเขตข้อมูลให้กับแต่ละตารางข้อมูล โดยเลือกตารางข้อมูลที่จะเพิ่มเขตข้อมูล แล้วกดปุ่ม Edit ดังรูปที่ ๑-8



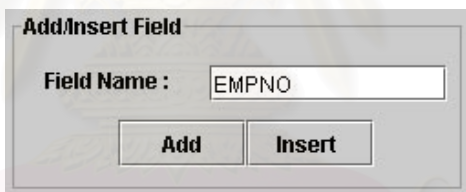
รูปที่ ๑-8 การเลือกแก้ไขตารางข้อมูล

จากนั้นเครื่องมือจะแสดงหน้าแก้ไขตารางข้อมูลขึ้นมา ดังรูปที่ ๑-9



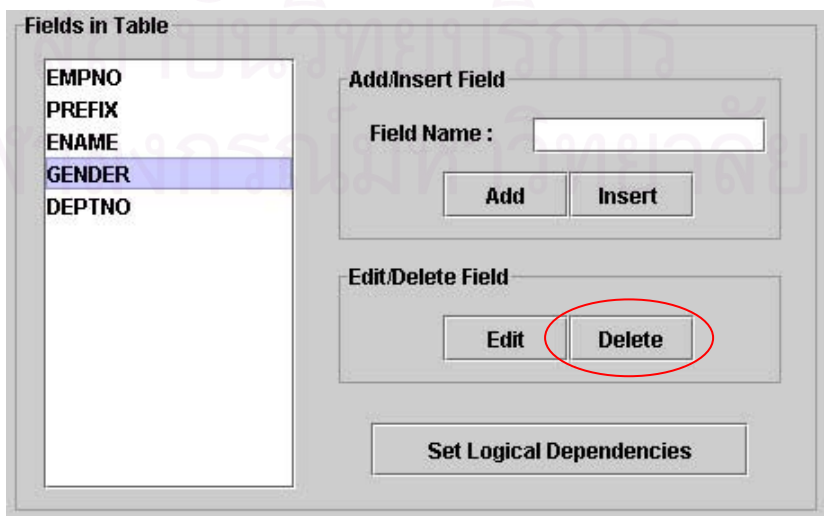
รูปที่ ๑-9 หน้าแก้ไขตารางข้อมูล

ผู้ใช้สามารถเพิ่มหรือแทรกเขตข้อมูลได้ โดยการกำหนดชื่อเขตข้อมูลแล้วกดปุ่ม Add หรือ Insert ดังรูปที่ ๑-10



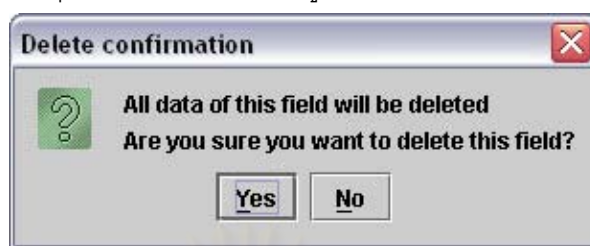
รูปที่ ๑-10 การเพิ่มเขตข้อมูล

1.1.6 ลบเขตข้อมูลที่ไม่ต้องการ โดยเลือกเขตข้อมูลที่ต้องการจะลบ แล้วกดปุ่ม Delete ดังรูปที่ ๑-11



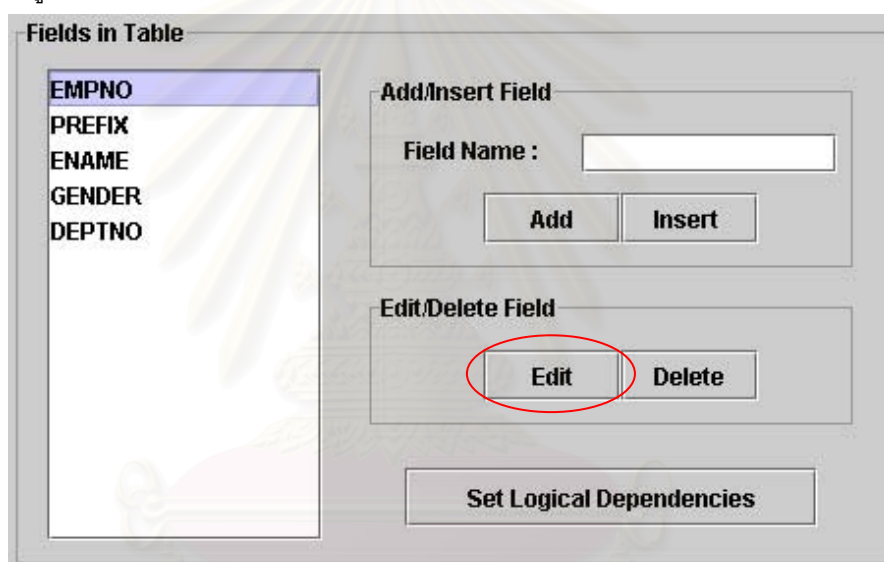
รูปที่ ๑-11 การเลือกลบเขตข้อมูล

จากนั้นเครื่องมือจะแสดงข้อความเพื่อขอการยืนยันจากผู้ใช้ในการลบเขตข้อมูล ดังรูปที่ ๑-12 ซึ่งให้ผู้ใช้งานกดปุ่ม Yes เพื่อลบเขตข้อมูลที่เลือก



รูปที่ ๑-12 หน้าขอคำยืนยันในการลบเขตข้อมูล

1.1.7 แก้ไขคุณสมบัติของแต่ละเขตข้อมูล โดยเลือกเขตข้อมูลที่ต้องการแก้ไข แล้วกดปุ่ม Edit ดังรูปที่ ๑-13



รูปที่ ๑-13 การเลือกแก้ไขเขตข้อมูล

จากนั้นเครื่องมือจะแสดงหน้าต่างแก้ไขคุณสมบัติของเขตข้อมูลขึ้นมา ดังรูปที่ ๑-14 ซึ่งผู้ใช้งานสามารถแก้ไขคุณสมบัติของเขตข้อมูลได้ โดยคุณสมบัติของเขตข้อมูลประกอบด้วย

- ชนิดข้อมูล (Field Type) ได้แก่ Character Integer Double Boolean Date Time และ Timestamp
- ขนาดของเขตข้อมูล (Size) หรือความเที่ยงตรงของจำนวน (Numeric Precision) ขึ้นอยู่กับชนิดข้อมูล ถ้าเป็นตัวอักษรจะกำหนดเป็นขนาดของเขตข้อมูล แต่ถ้าเป็นจำนวนเต็มหรือจำนวนทศนิยม จะกำหนดเป็นความเที่ยงตรงของจำนวน ซึ่งหมายถึงจำนวนหลักที่เป็นไปได้มากที่สุดของค่าตัวเลข โดยนับรวมจำนวนหลักของทศนิยมด้วยนั่นเอง

- **ขนาดของทศนิยม (Numeric Scale)** คือจำนวนหลักของทศนิยม สามารถกำหนดได้เฉพาะเขตข้อมูลที่มีชนิดเป็นจำนวนทศนิยมเท่านั้น
 - **ตัวระบุ (Identifier) หรือคีย์หลัก (Primary Key)** จะเป็นการระบุว่าเขตข้อมูลนั้นเป็นคีย์หลักหรือเป็นส่วนหนึ่งของคีย์หลักที่ใช้อ้างอิงถึงแต่ละระเบียบข้อมูลในตารางข้อมูลหรือไม่
 - **ตัวเลือก (Optional)** จะเป็นการระบุว่าเขตข้อมูลนั้นจำเป็นต้องมีค่าข้อมูลในทุกๆ ระเบียบของเขตข้อมูลหรือไม่
- เมื่อกำหนดคุณสมบัติของเขตข้อมูลแล้ว ผู้ใช้ต้องกดปุ่ม OK เพื่อบันทึกคุณสมบัติ

ที่ได้กำหนดไว้

รูปที่ ๑-14 หน้าแก้ไขคุณสมบัติของเขตข้อมูล

1.1.8 กำหนดการขึ้นต่อกันเชิงตรรกะ โดยกดปุ่ม Set Logical Dependencies ที่หน้าแก้ไขตารางข้อมูล ดังรูปที่ ๑-15

Edit Table

Table Name : Employee

Fields in Table

EMPNO
PREFIX
ENAME
GENDER
DEPTNO

Add/Insert Field

Field Name :

Add Insert

Edit/Delete Field

Edit Delete

Set Logical Dependencies

OK

รูปที่ จ-15 การเลือกกำหนดการขึ้นต่อกันเชิงตรรกะ
จากนั้นเครื่องมือจะแสดงหน้ากำหนดการขึ้นต่อกันเชิงตรรกะขึ้นมา ดังรูปที่ จ-16

Edit Logical Dependencies

Mode of Logical Dependency

Normal Logical Dependency SQL Logical Dependency

Add/Edit/Delete Normal Logical Dependencies

If the field are

then the field are

Add Edit Delete

Current Normal Logical Dependencies

Source Field	Condition value	Destination Field	Result value

Specify SQL Logical Dependency

SQL statement :

OK

รูปที่ จ-16 หน้ากำหนดการขึ้นต่อกันเชิงตรรกะ

ผู้ใช้สามารถกำหนดการขึ้นต่อกันเชิงตรรกะได้ 2 แบบคือ

1) การกำหนดการขึ้นต่อกันเชิงตรรกะแบบปกติ เริ่มจากการเลือกตัวเลือก Normal Logical Dependency จากนั้นผู้ใช้จะสามารถจัดการเงื่อนไขการขึ้นต่อกันเชิงตรรกะได้ ซึ่งแบ่งเป็น 3 ส่วนคือ

1.1) การเพิ่มเงื่อนไขการขึ้นต่อกันเชิงตรรกะ ทำได้โดยการกำหนดเขตข้อมูลต้นทาง ค่าเงื่อนไข เขตข้อมูลเป้าหมาย และค่าผลลัพธ์ที่ต้องการ แล้วกดปุ่ม Add ดังรูปที่ ๑-17

Add/Edit/Delete Normal Logical Dependencies

If the **PREFIX** field are Mr.

then the **GENDER** field are Male

Add **Edit** **Delete**

Current Normal Logical Dependencies

Source Field	Condition value	Destination Field	Result value
PREFIX	Mr.	GENDER	Male

รูปที่ ๑-17 การเพิ่มเงื่อนไขการขึ้นต่อกันเชิงตรรกะ

1.2) การแก้ไขเงื่อนไขการขึ้นต่อกันเชิงตรรกะ ทำได้โดยการกำหนดเขตข้อมูลต้นทาง ค่าเงื่อนไข เขตข้อมูลเป้าหมาย และค่าผลลัพธ์ที่ต้องการ จากนั้นเลือกเงื่อนไขที่ต้องแก้ไข แล้วกดปุ่ม Edit ดังรูปที่ ๑-18

Add/Edit/Delete Normal Logical Dependencies

If the **PREFIX** field are Mrs.||Miss

then the **GENDER** field are Female

Add **Edit** **Delete**

Current Normal Logical Dependencies

Source Field	Condition value	Destination Field	Result value
PREFIX	Mr.	GENDER	Male
PREFIX	Mrs. Miss	GENDER	Female

รูปที่ ๑-18 การแก้ไขเงื่อนไขการขึ้นต่อกันเชิงตรรกะ

1.3) การลบเงื่อนไขการขึ้นต่อกันเชิงตรรกะ ทำได้โดยเลือกเงื่อนไขที่ต้องลบ แล้วกดปุ่ม Delete ดังรูปที่ ๑-19

Source Field	Condition value	Destination Field	Result value
PREFIX	Mr.	GENDER	Male
PREFIX	Mrs. Miss	GENDER	Female
PREFIX	a	GENDER	z

รูปที่ ๑-19 การลบเงื่อนไขการขึ้นต่อกันเชิงตรรกะ
จากนั้นเครื่องมือจะแสดงข้อความเพื่อขอการยืนยันจากผู้ใช้ในการลบเงื่อนไข ดังรูปที่ ๑-20 ซึ่งให้ผู้ใช้กดปุ่ม Yes เพื่อลบเงื่อนไขที่เลือก

รูปที่ ๑-20 หน้าขอคำยืนยันในการลบเงื่อนไขการขึ้นต่อกันเชิงตรรกะ

2) การกำหนดการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล เริ่มจากการเลือกตัวเลือก SQL Logical Dependency จากนั้นจึงกำหนดคำสั่งเอสคิวแอลตามที่ต้องการ ดังรูปที่ ๑-21

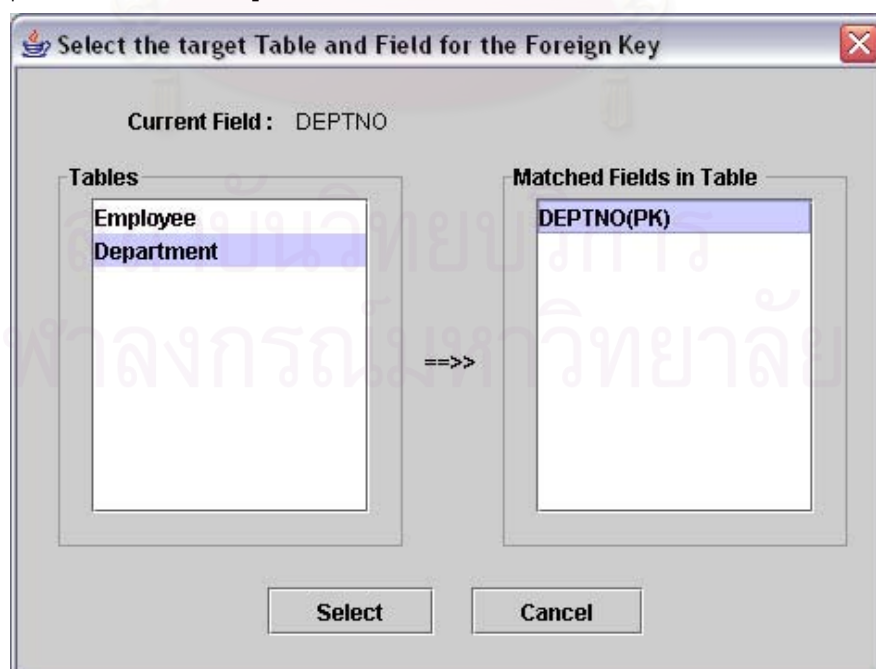
รูปที่ ๑-21 การกำหนดการขึ้นต่อกันเชิงตรรกะแบบเอสคิวแอล

1.1.9 กำหนดความคงสภาพในการอ้างอิง โดยเลือกเขตข้อมูลที่ต้องการกำหนดเป็นคีย์อ้างอิงในหน้าแก้ไขตารางข้อมูล แล้วกดปุ่ม Edit เมื่อเครื่องมือแสดงหน้าแก้ไขคุณสมบัติของเขตข้อมูลให้กดปุ่ม Set Foreign Key ดังรูปที่ ๑-22



รูปที่ ๑-22 การเลือกกำหนดคีย์อ้างอิง

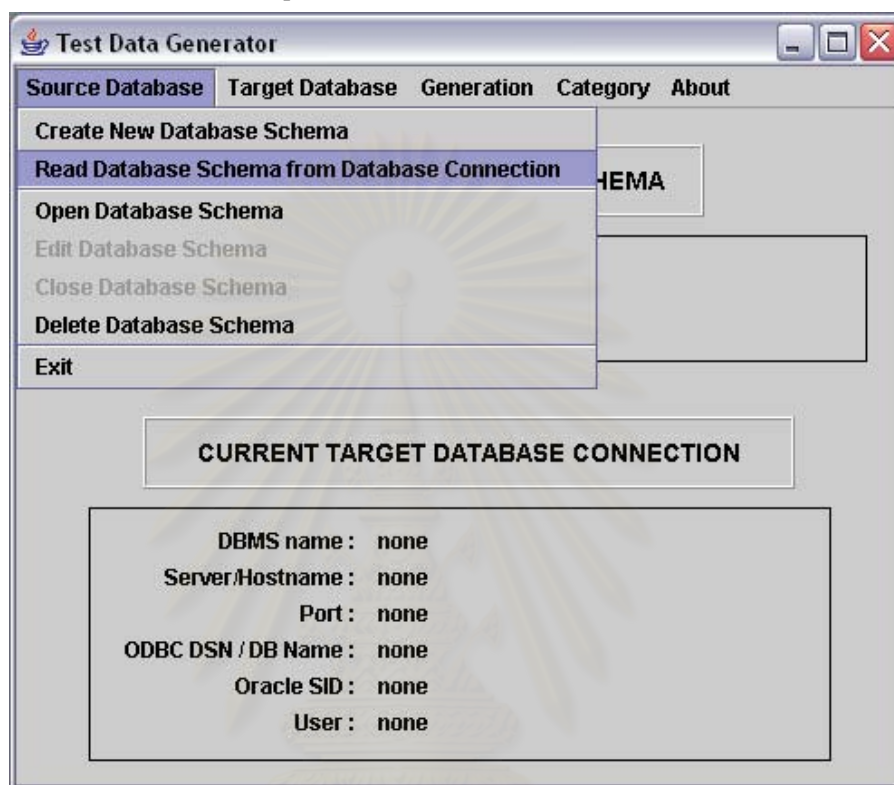
จากนั้นเครื่องมือจะแสดงหน้ากำหนดคีย์อ้างอิงขึ้นมา ซึ่งผู้ใช้สามารถเลือกตารางข้อมูลและเขตข้อมูลที่จะถูกอ้างอิงได้ เมื่อเลือกแล้วจึงกดปุ่ม Select เพื่อกำหนดให้เขตข้อมูลปัจจุบันเป็นคีย์อ้างอิง ดังรูปที่ ๑-23



รูปที่ ๑-23 หน้ากำหนดคีย์อ้างอิง

1.2 การสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลที่มีอยู่ จะมีขั้นตอนดังนี้

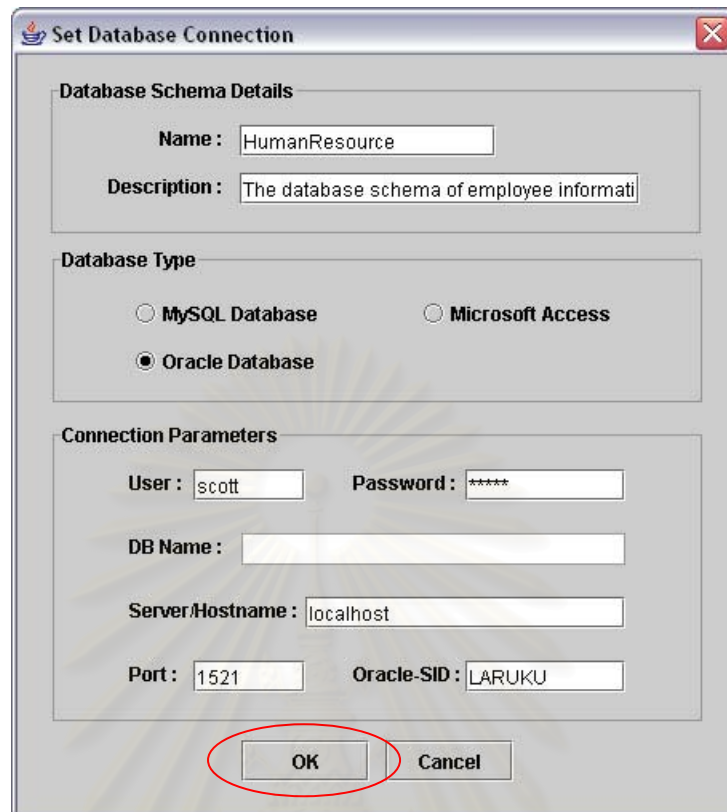
1.2.1 เลือกที่เมนู Source Database และเลือกที่เมนูย่อย Read Database Schema from Database Connection ดังรูปที่ ๑-24



รูปที่ ๑-24 การสร้างสคีมาของฐานข้อมูลจากฐานข้อมูลที่มีอยู่

1.2.2 กำหนดชื่อและคำอธิบายให้กับสคีมาของฐานข้อมูลที่ต้องการสร้าง จากนั้นเลือกชนิดของฐานข้อมูลที่จะทำการเชื่อมต่อ พร้อมทั้งกำหนดพารามิเตอร์ที่จะใช้ในการเชื่อมต่อ และกดปุ่ม OK เพื่อทำการสร้างสคีมาของฐานข้อมูล ดังรูปที่ ๑-25

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

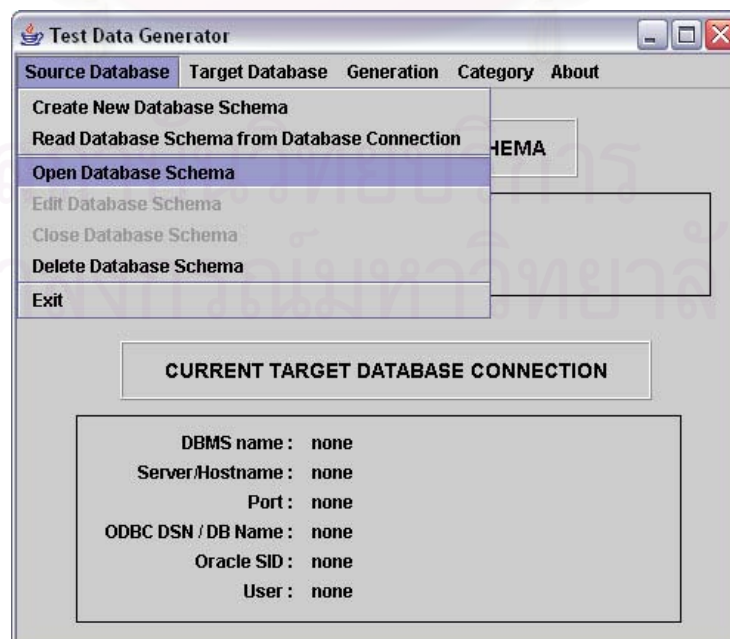


รูปที่ ๑-25 หน้ากำหนดการเชื่อมต่อกับฐานข้อมูลเพื่อสร้างสคีมาของฐานข้อมูล

1.3 การเปิดใช้สคีมาของฐานข้อมูล จะมีขั้นตอนดังนี้

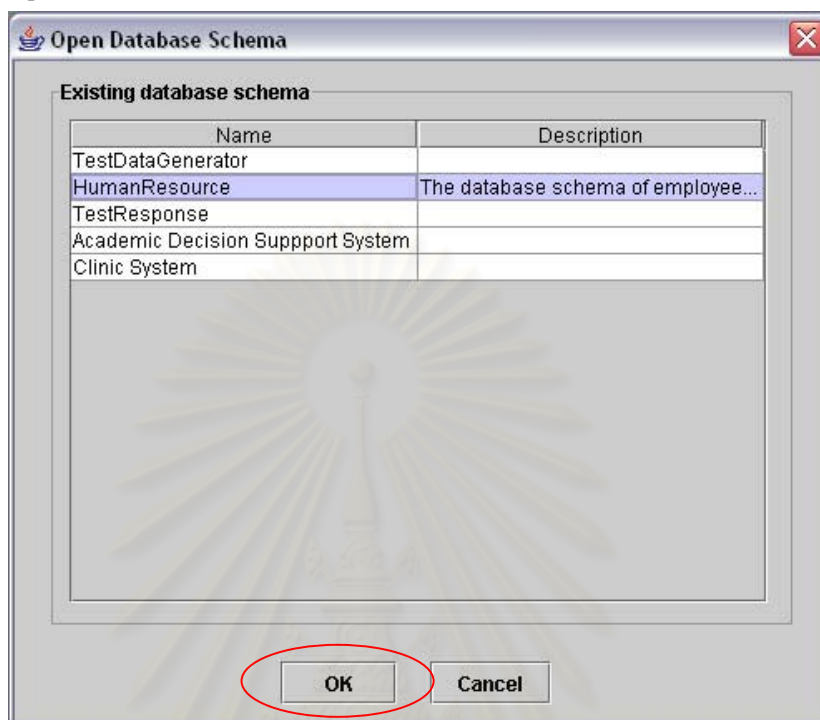
1.3.1 เลือกที่เมนู Source Database และเลือกที่เมนูย่อย Open Database Schema

ดังรูปที่ ๑-26



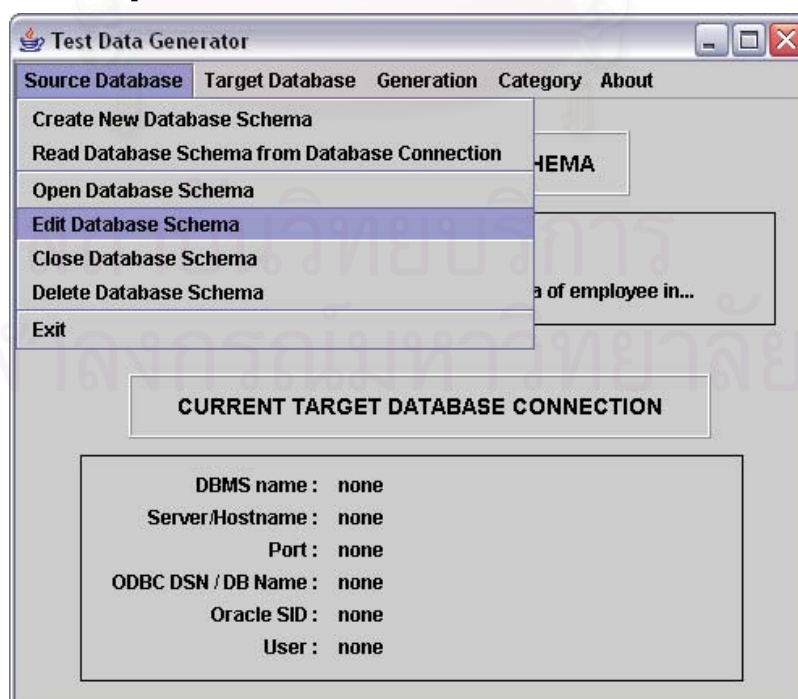
รูปที่ ๑-26 การเปิดสคีมาของฐานข้อมูล

1.3.2 เลือกสคีมาของฐานข้อมูลที่ต้องการเปิด แล้วกดปุ่ม OK เพื่อเปิดสคีมาของฐานข้อมูล ดังรูปที่ ๑-27



รูปที่ ๑-27 หน้าเปิดสคีมาของฐานข้อมูล

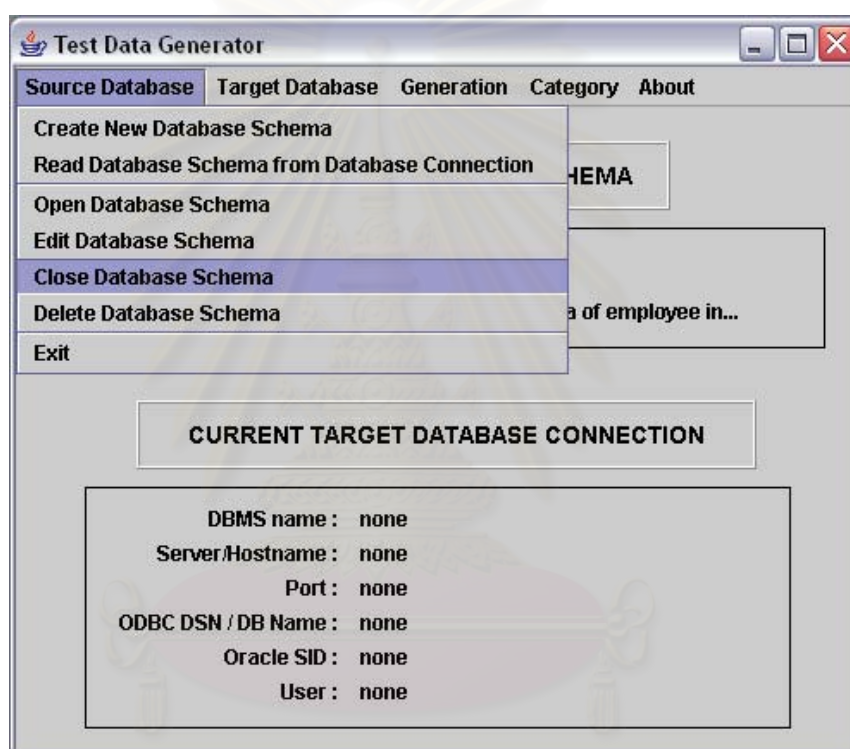
1.4 การแก้ไขสคีมาของฐานข้อมูล เลือกที่เมนู Source Database และเลือกที่เมนูย่อย Edit Database Schema ดังรูปที่ ๑-28



รูปที่ ๑-28 การแก้ไขสคีมาของฐานข้อมูล

จากนั้นเครื่องมือจะแสดงหน้าต่างแก้ไขสคีมาของฐานข้อมูล ดังรูปที่ ๑-4 ซึ่งผู้ใช้จะสามารถแก้ไขสคีมาของฐานข้อมูลได้ตามต้องการ ไม่ว่าจะเป็นการเพิ่ม แก้ไข หรือลบตารางข้อมูล การแก้ไขการขึ้นต่อกันเชิงตรรกะ และการกำหนดความคงสภาพในการอ้างอิง ดังที่กล่าวมาแล้วในส่วนการสร้างสคีมาของฐานข้อมูลด้วยตัวเอง

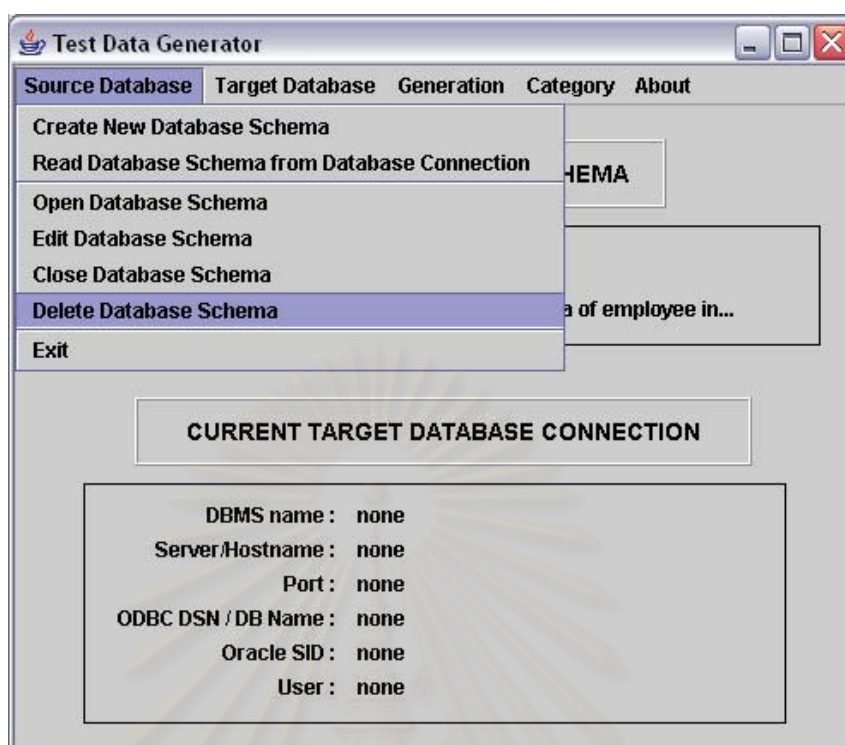
1.5 การปิดสคีมาของฐานข้อมูล ทำได้โดยเลือกที่เมนู Source Database และเลือกที่เมนูย่อย Close Database Schema ดังรูปที่ ๑-29 เพื่อทำการปิดสคีมาของฐานข้อมูล



รูปที่ ๑-29 การปิดสคีมาของฐานข้อมูล

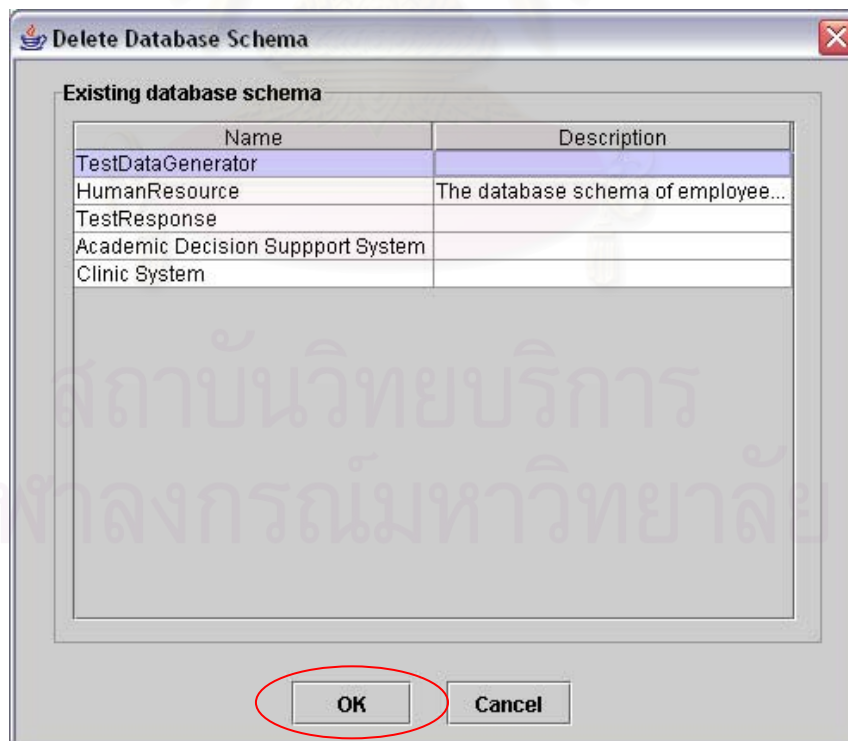
1.6 การลบสคีมาของฐานข้อมูล จะมีขั้นตอนดังนี้

1.6.1 เลือกที่เมนู Source Database และเลือกที่เมนูย่อย Delete Database Schema ดังรูปที่ ๑-30



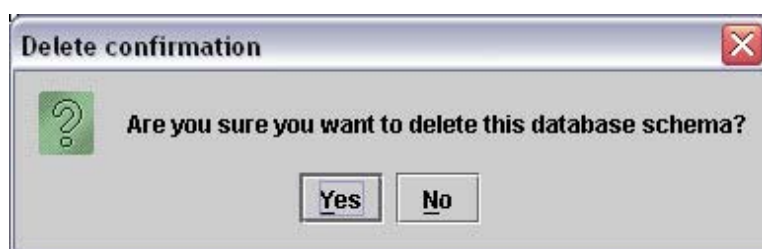
รูปที่ ๑-30 การลบสคีมาของฐานข้อมูล

1.6.2 เลือกสคีมาของฐานข้อมูลที่ต้องลบ แล้วกดปุ่ม OK ดังรูปที่ ๑-31



รูปที่ ๑-31 หน้าลบสคีมาของฐานข้อมูล

จากนั้นเครื่องมือจะแสดงข้อความเพื่อขอการยืนยันจากผู้ใช้ในการลบสคีมาของฐานข้อมูล ดังรูปที่ ๑-32 ซึ่งให้ผู้ใช้กดปุ่ม Yes เพื่อลบสคีมาของฐานข้อมูลที่เลือก

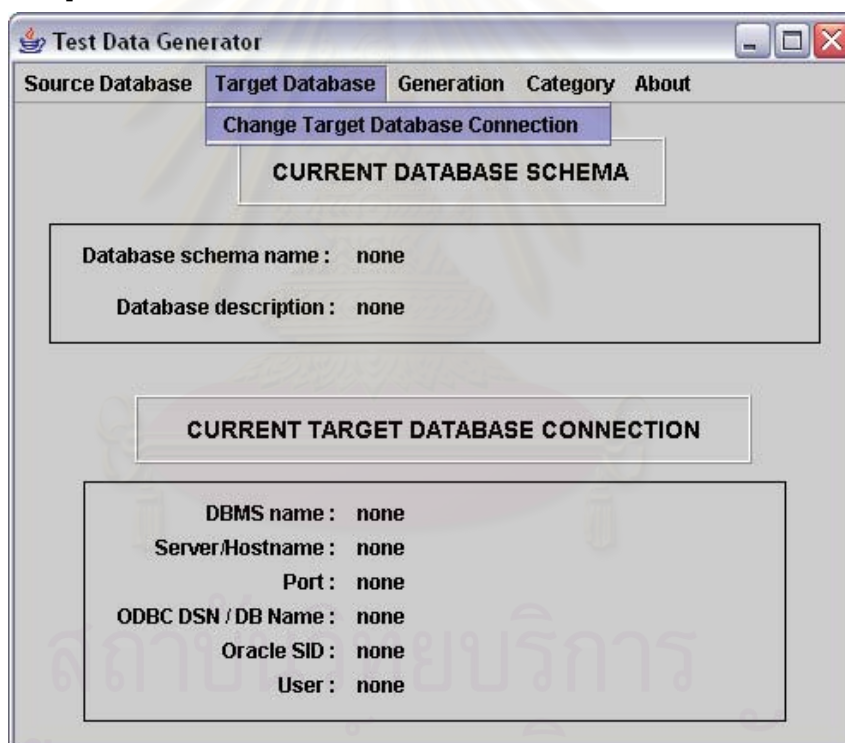


รูปที่ ๑-32 หน้าขอคำยืนยันในการลบสคีมาของฐานข้อมูล

2. การเชื่อมต่อกับฐานข้อมูลเป้าหมาย

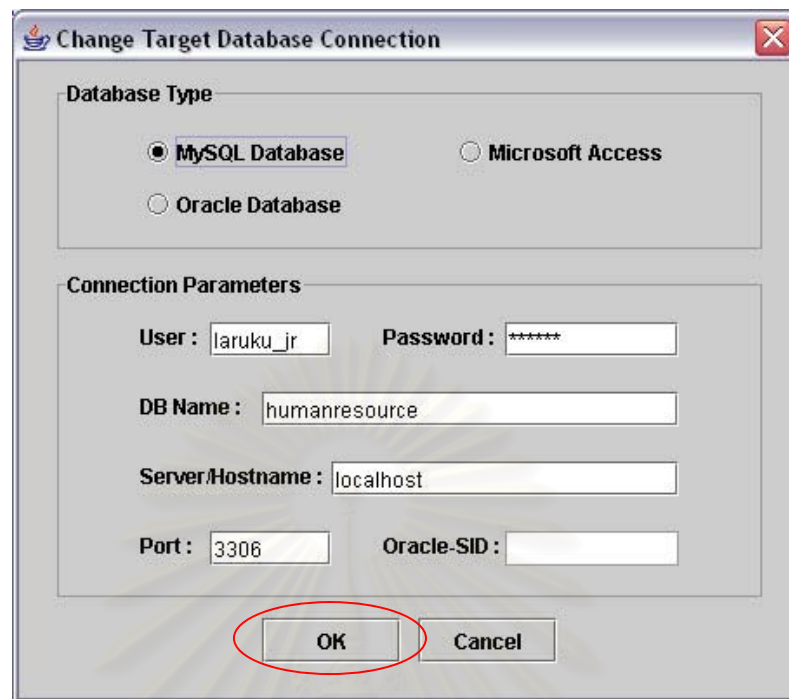
จะมีขั้นตอนการใช้งานดังนี้

2.1 เลือกที่เมนู Target Database และเลือกที่เมนูย่อย Change Target Database Connection ดังรูปที่ ๑-33



รูปที่ ๑-33 การเชื่อมต่อกับฐานข้อมูลเป้าหมาย

2.2 เลือกชนิดของฐานข้อมูลที่จะทำการเชื่อมต่อ พร้อมทั้งกำหนดพารามิเตอร์ที่จะใช้ในการเชื่อมต่อ และกดปุ่ม OK เพื่อทำการเชื่อมต่อกับฐานข้อมูลเป้าหมาย ดังรูปที่ ๑-34

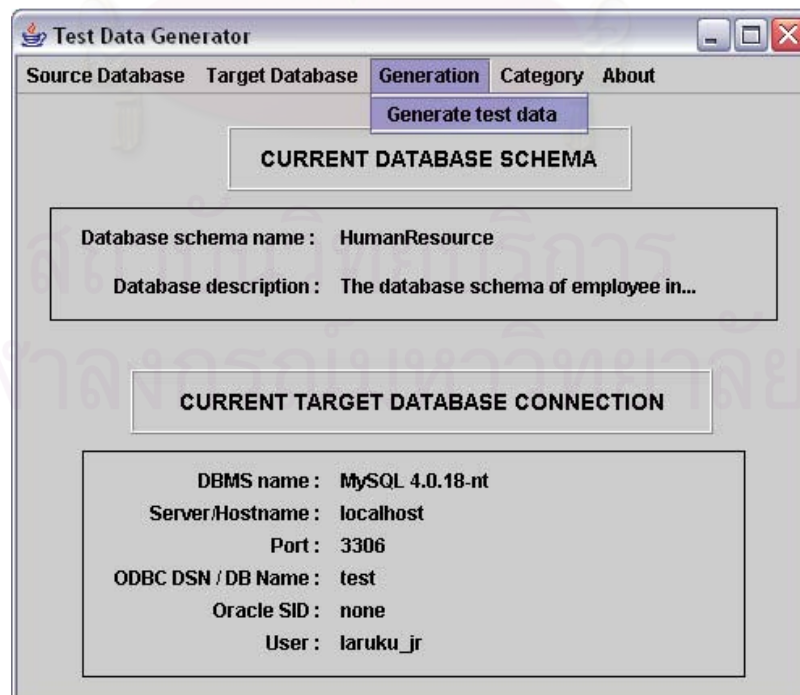


รูปที่ ๑-34 หน้ากำหนดการเชื่อมต่อกับฐานข้อมูลเป้าหมาย

3. การสร้างข้อมูลทดสอบ

จะมีขั้นตอนการใช้งานดังนี้

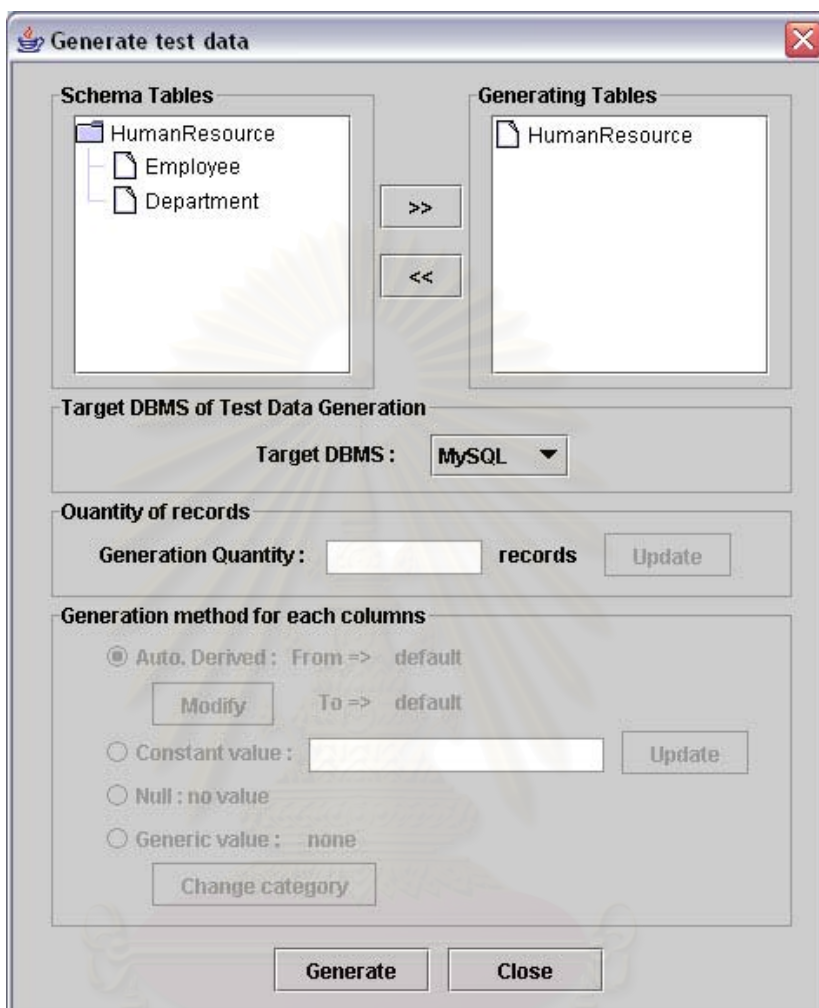
3.1 เลือกที่เมนู Generate และเลือกที่เมนูย่อย Generate test data ดังรูปที่ ๑-35



รูปที่ ๑-35 การสร้างข้อมูลทดสอบ

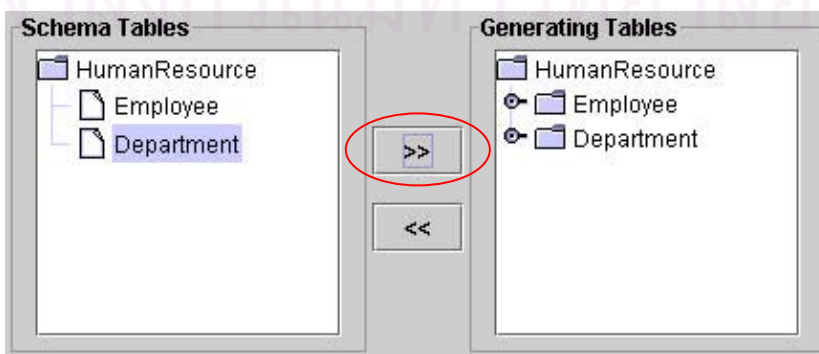
จากนั้นเครื่องมือจะแสดงหน้ากำหนดรายละเอียดการสร้างข้อมูลทดสอบขึ้นมา ดังรูปที่

จ-36



รูปที่ จ-36 หน้ากำหนดรายละเอียดการสร้างข้อมูลทดสอบ

3.2 เลือกตารางข้อมูลที่ต้องการสร้างข้อมูลทดสอบ โดยเลือกตารางที่ต้องการจากรายการด้านซ้าย แล้วกดปุ่ม >> เพื่อเพิ่มตารางข้อมูลเข้าสู่รายการด้านขวา ซึ่งเป็นรายการตารางข้อมูลที่จะสร้างข้อมูลทดสอบ ดังรูปที่ จ-37



รูปที่ จ-37 การเลือกตารางข้อมูลที่ต้องการสร้างข้อมูลทดสอบ

3.3 เลือกฐานข้อมูลเป้าหมาย ซึ่งสามารถเลือกได้ 3 ชนิดคือ MySQL Oracle และ ACCESS ดังรูปที่ ๑-38

รูปที่ ๑-38 การเลือกฐานข้อมูลเป้าหมาย

3.4 กำหนดจำนวนระเบียบข้อมูลทดสอบที่ต้องการสร้างให้กับแต่ละตารางข้อมูล โดยเลือกตารางข้อมูลที่ต้องการ จากนั้นกำหนดจำนวนระเบียบข้อมูล และกดปุ่ม Update เพื่อบันทึกจำนวนระเบียบข้อมูลที่กำหนด ดังรูปที่ ๑-39

รูปที่ ๑-39 การกำหนดจำนวนระเบียบข้อมูลทดสอบที่ต้องการสร้าง

3.5 กำหนดวิธีการสร้างข้อมูลทดสอบให้กับแต่ละเขตข้อมูล โดยเลือกเขตข้อมูลที่ต้องการ และเลือกวิธีการสร้างข้อมูลทดสอบที่ต้องการ ดังรูปที่ ๑-40

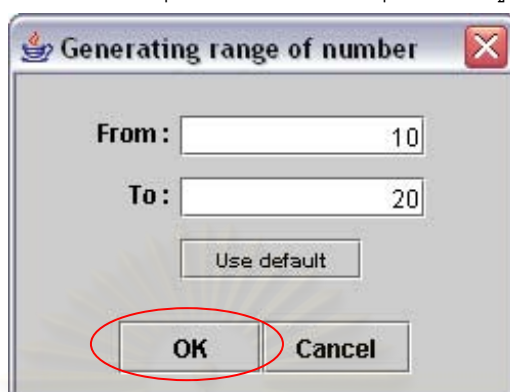
รูปที่ ๑-40 การกำหนดวิธีการสร้างข้อมูลทดสอบ

ผู้ใช้สามารถกำหนดวิธีการสร้างข้อมูลได้ 4 วิธีคือ

3.5.1 สุ่มสร้างอัตโนมัติ (Auto. Derived) ซึ่งผู้ใช้สามารถกำหนดช่วงของค่าที่ต้องการให้สุ่มได้ โดยการกดปุ่ม Modify ดังรูปที่ ๑-41

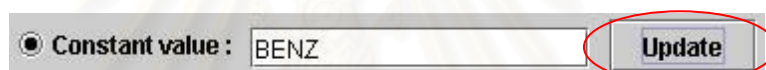
รูปที่ ๑-41 การกำหนดให้เครื่องมือสุ่มสร้างข้อมูลทดสอบอัตโนมัติ

จากนั้นเครื่องมือจะแสดงหน้ากำหนดช่วงของค่าที่ต้องการสุ่มสร้างข้อมูลทดสอบ ซึ่งผู้ใช้ต้องกำหนดค่าเริ่มต้น และค่าสิ้นสุดของช่วง แล้วกดปุ่ม OK ดังรูปที่ ๑-42



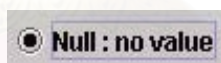
รูปที่ ๑-42 หน้ากำหนดช่วงของค่าที่ต้องการสุ่มสร้างข้อมูลทดสอบ

3.5.2 ค่าคงที่ (Constant value) คือกำหนดให้ทุกระเบียนข้อมูลของเขตข้อมูลนี้มีค่าเท่ากับค่าที่กำหนดเหมือนกันทั้งหมด โดยผู้ใช้ต้องกำหนดค่าคงที่ที่ต้องการ และกดปุ่ม Update ดังรูปที่ ๑-43



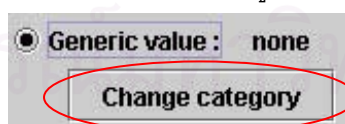
รูปที่ ๑-43 การกำหนดให้ใช้ค่าคงที่

3.5.3 ค่าว่าง (Null) คือกำหนดให้ไม่ต้องการสร้างข้อมูลให้กับเขตข้อมูลนี้ โดยจะใช้ได้กับเขตข้อมูลที่มีคุณสมบัติเป็นตัวเลือกเท่านั้น ดังรูปที่ ๑-44



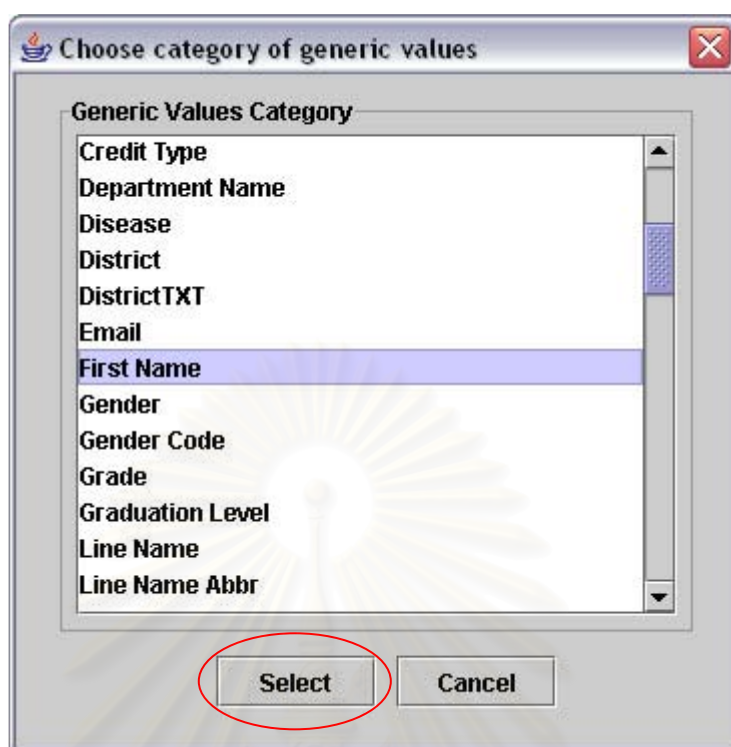
รูปที่ ๑-44 การกำหนดให้ใช้ค่าว่าง

3.5.4 ค่าทั่วไป (Generic value) คือกำหนดให้ใช้กลุ่มค่าทั่วไปซึ่งเป็นกลุ่มของค่าข้อมูลทั่วไปที่เครื่องมือได้จัดเตรียมไว้ให้ เช่น ชื่อ นามสกุล และเพศ เป็นต้น ในการสร้างข้อมูลให้กับเขตข้อมูล โดยเครื่องมือจะสุ่มค่าขึ้นมาจากกลุ่มค่าทั่วไปที่เลือกไว้ ผู้ใช้สามารถกำหนดเลือกกลุ่มค่าทั่วไปที่ต้องการได้โดยการกดปุ่ม Change category ดังรูปที่ ๑-45



รูปที่ ๑-45 การกำหนดให้ใช้กลุ่มค่าทั่วไป

จากนั้นเครื่องมือจะแสดงหน้ารายการกลุ่มค่าทั่วไปขึ้นมา ผู้ใช้สามารถเลือกกลุ่มค่าทั่วไปที่ต้องการ แล้วกดปุ่ม Select เพื่อเลือก ดังรูปที่ ๑-46

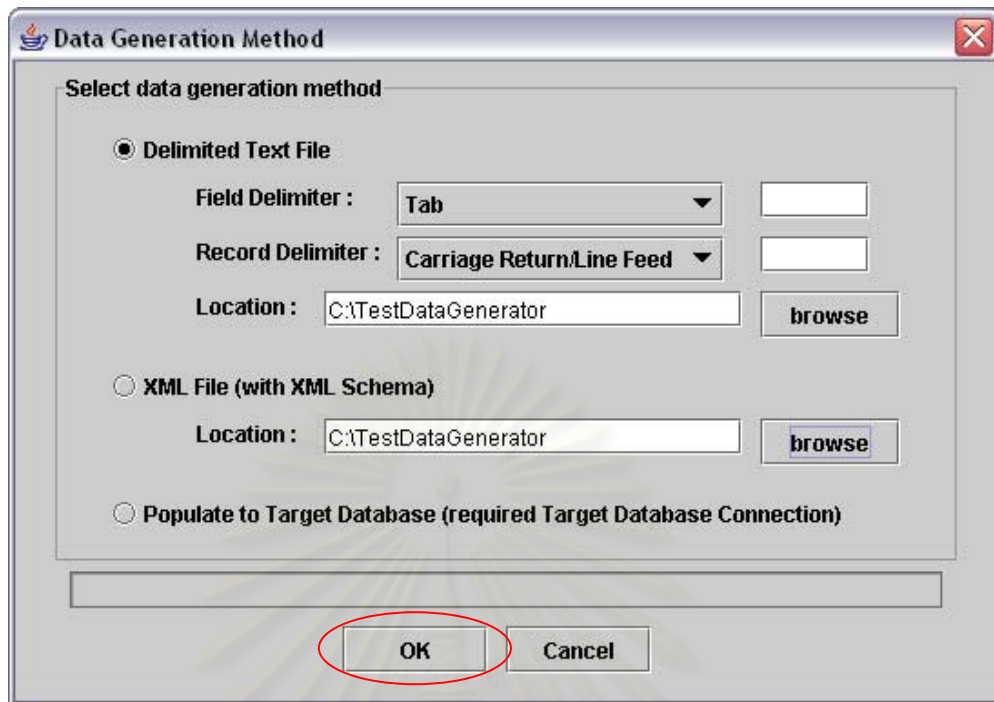


รูปที่ ๑-46 หน้ารายการกลุ่มค่าทั่วไป

3.6 เลือกรูปแบบการบันทึกข้อมูลทดสอบ ผู้ใช้สามารถเลือกได้ 3 รูปแบบคือ

- บันทึกข้อมูลทดสอบเป็นแฟ้มข้อความที่มีตัวคั่น โดยผู้ใช้สามารถกำหนดตัวคั่นระหว่างเขตข้อมูลและระเบียบข้อมูลได้ และผู้ใช้ต้องกำหนดที่อยู่ของแฟ้มข้อความที่จะบันทึกข้อมูลทดสอบด้วย
- บันทึกข้อมูลทดสอบเป็นแฟ้มเอกสารเอ็กซ์เอ็มแอล โดยผู้ใช้ต้องกำหนดที่อยู่ของแฟ้มเอกสารเอ็กซ์เอ็มแอลที่จะบันทึกข้อมูลทดสอบด้วย
- บันทึกข้อมูลทดสอบลงไปในฐานข้อมูลเป้าหมาย โดยเครื่องมือจะบันทึกข้อมูลทดสอบที่สร้างได้ลงในฐานข้อมูลเป้าหมายที่กำลังเชื่อมต่ออยู่

เมื่อผู้ใช้เลือกรูปแบบการบันทึกข้อมูลทดสอบแล้ว จึงกดปุ่ม OK เพื่อทำการสร้างข้อมูลทดสอบต่อไป

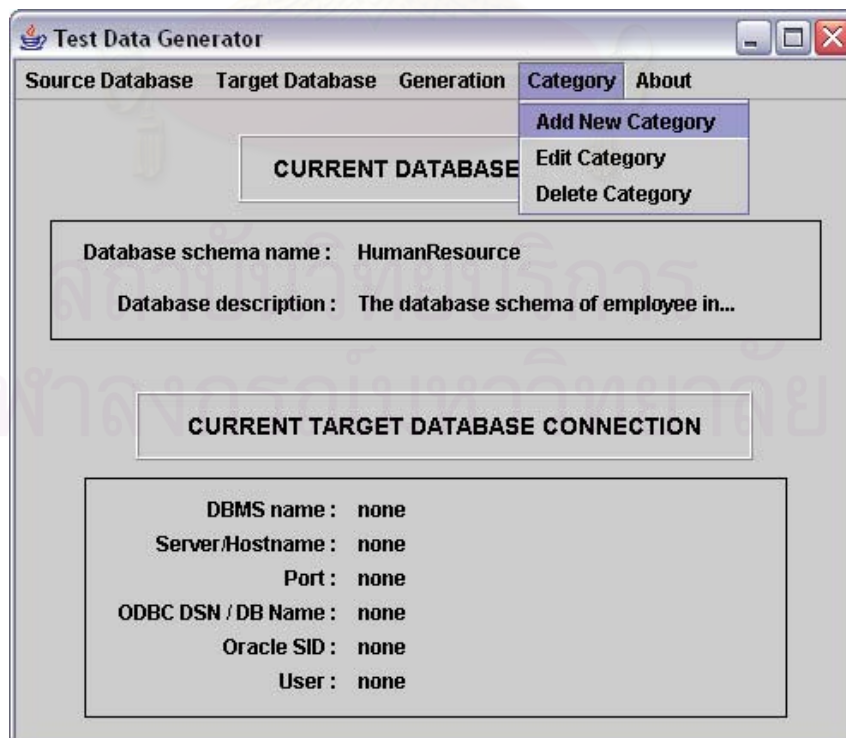


รูปที่ ๑-47 หน้าเลือกรูปแบบการบันทึกข้อมูลทดสอบ

4. การจัดการกลุ่มค่าทั่วไป

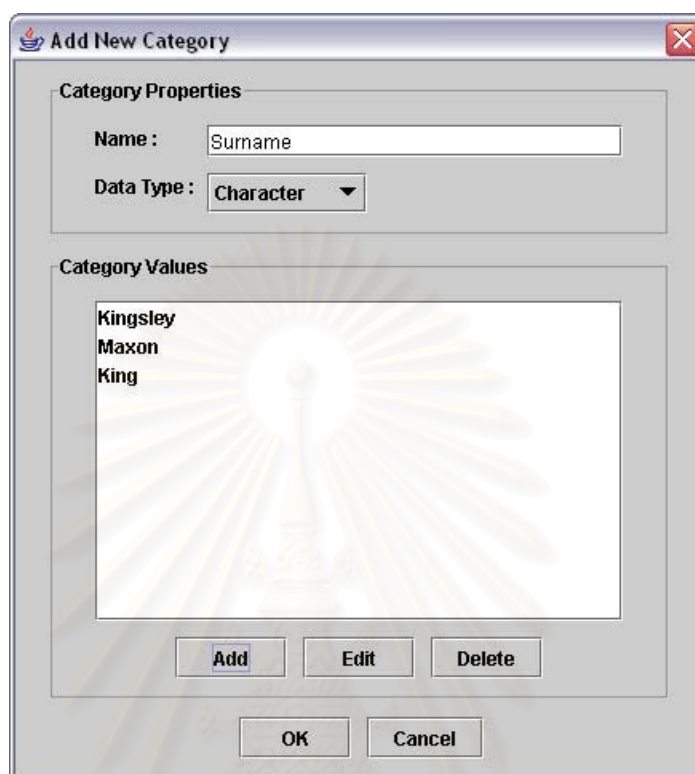
4.1 การเพิ่มกลุ่มค่าทั่วไป จะมีขั้นตอนดังนี้

4.1.1 เลือกที่เมนู Category และเลือกที่เมนูย่อย Add New Category ดังรูปที่ ๑-48



รูปที่ ๑-48 การเพิ่มกลุ่มค่าทั่วไป

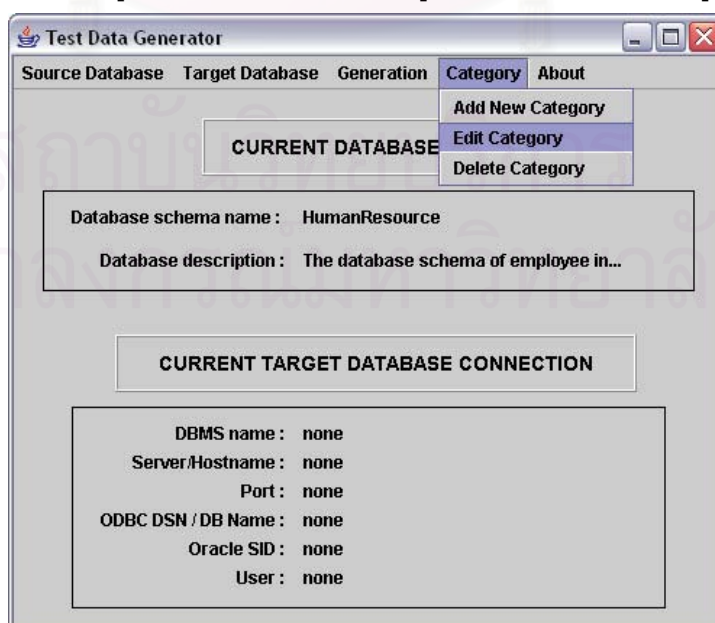
4.1.2 กำหนดข้อมูลให้กลุ่มค่าทั่วไป ได้แก่ ชื่อกลุ่มค่าทั่วไป ชนิดข้อมูล และรายการค่าทั่วไป ดังรูปที่ ๑-49



รูปที่ ๑-49 หน้าเพิ่มข้อมูลกลุ่มค่าทั่วไป

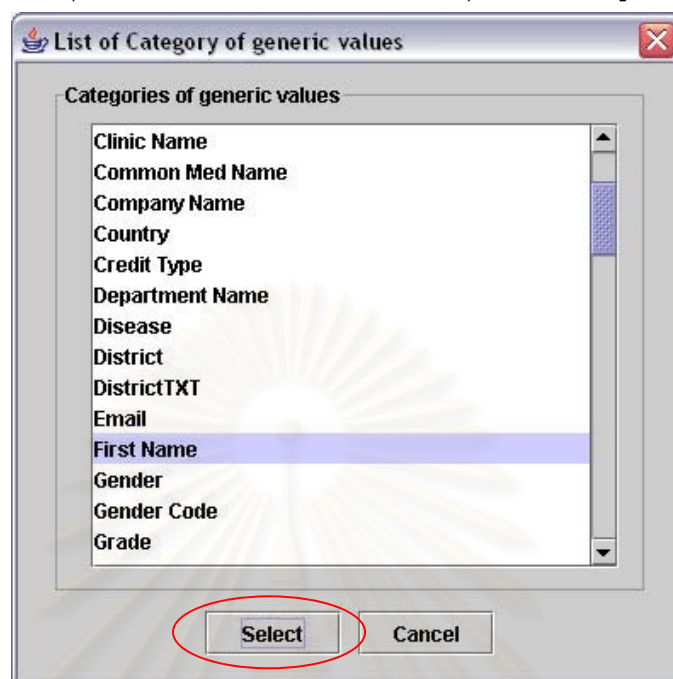
4.2 การแก้ไขกลุ่มค่าทั่วไป จะมีขั้นตอนดังนี้

4.2.1 เลือกที่เมนู Category และเลือกที่เมนูย่อย Edit Category ดังรูปที่ ๑-50



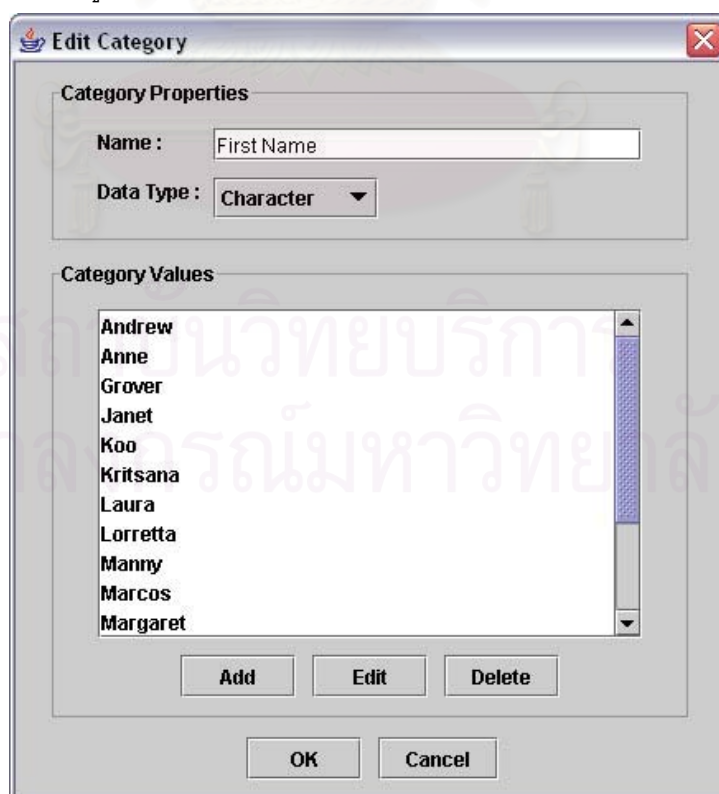
รูปที่ ๑-50 การแก้ไขกลุ่มค่าทั่วไป

4.2.2 เลือกกลุ่มค่าทั่วไปที่ต้องการแก้ไข แล้วกดปุ่ม Select ดังรูปที่ ๑-51



รูปที่ ๑-51 การเลือกกลุ่มทั่วไปที่ต้องการแก้ไข

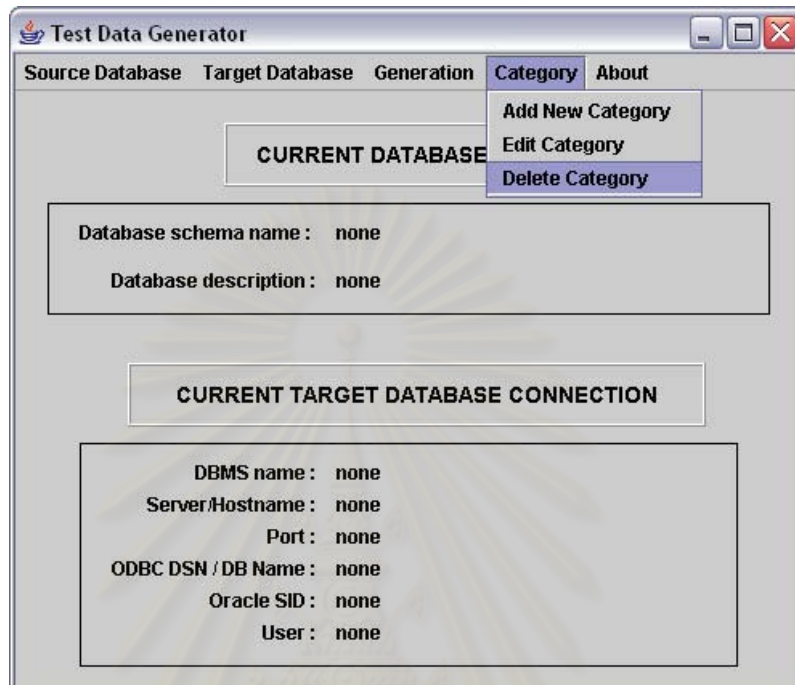
4.2.3 แก้ไขข้อมูลของกลุ่มค่าทั่วไปตามที่ต้องการ ได้แก่ ชื่อกลุ่มค่าทั่วไป ชนิดข้อมูล และรายการค่าทั่วไป ดังรูปที่ ๑-52



รูปที่ ๑-52 หน้าแก้ไขกลุ่มค่าทั่วไป

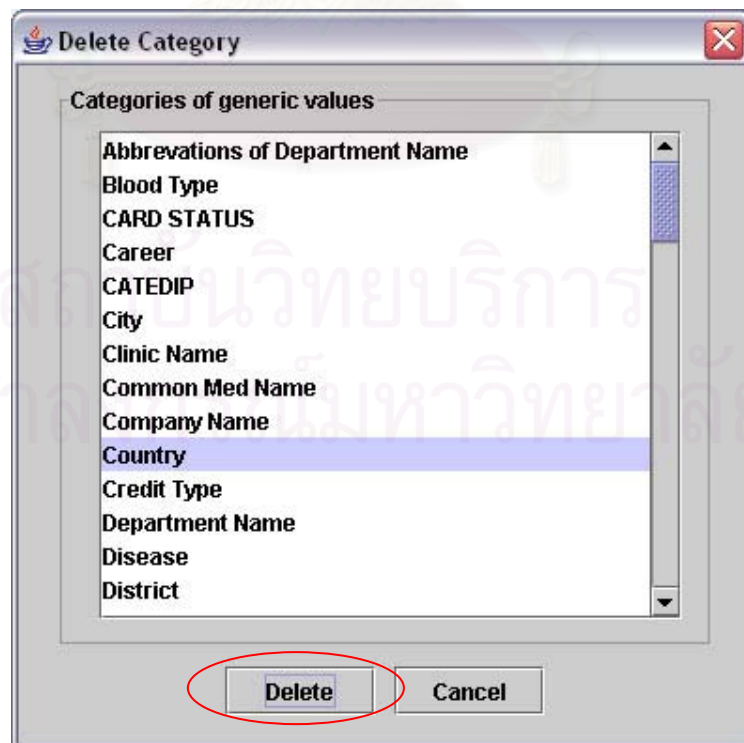
4.3 การลบกลุ่มค่าทั่วไป จะมีขั้นตอนดังนี้

4.3.1 เลือกที่เมนู Category และเลือกที่เมนูย่อย Delete Category ดังรูปที่ ๑-53



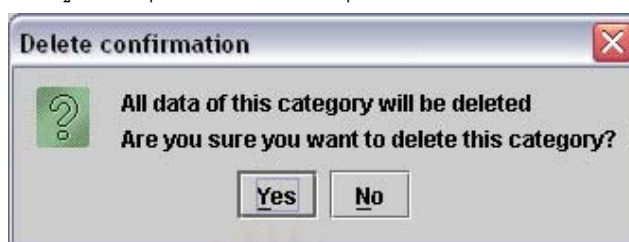
รูปที่ ๑-53 การลบกลุ่มค่าทั่วไป

4.3.2 เลือกกลุ่มค่าทั่วไปที่ต้องการลบ แล้วกดปุ่ม Delete ดังรูปที่ ๑-54



รูปที่ ๑-54 การเลือกกลุ่มค่าทั่วไปที่ต้องการลบ

จากนั้นเครื่องมือจะแสดงข้อความเพื่อขอการยืนยันจากผู้ใช้ในการลบกลุ่มค่า
ทั่วไป ดังรูปที่ ๑-55 ซึ่งให้ผู้ใช้กดปุ่ม Yes เพื่อลบกลุ่มค่าทั่วไปที่เลือก



รูปที่ ๑-55 หน้าขอคำยืนยันในการลบกลุ่มค่าทั่วไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก จ
ผลงานที่ตีพิมพ์

งานวิจัยนี้ได้รับคัดเลือกให้นำเสนอในงานประชุมวิชาการและตีพิมพ์ในเอกสาร
“Proceedings of the Second International Conference on Information and
Communication Technologies (ICT 2004)” ในระหว่างวันที่ 18-19 พฤศจิกายน พ.ศ.2547 โดย
มีหัวข้องานวิจัยชื่อ “RealGen : A Test Data Generation Tool to Support Software Testing”



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

RealGen : A Test Data Generation Tool to Support Software Testing

Kritsana Piriyaakitpaiboon and Taratip Suwannasart
 Department of Computer Engineering, Faculty of Engineering
 Chulalongkorn University, Bangkok, 10330, Thailand
 Tel: +66 2 2186956 Fax: +66 2 2186955
 Email: laruk_jr@hotmail.com, Taratip.S@chula.ac.th

Abstract

Test data generation is one of the tasks that consume a lot of cost and effort for software testers to finish because the generated test data need to be similar to the actual data. This paper presents a tool to automate test data generation for testers, named "RealGen". The tool randomly generates test data according to the criteria, supplied by testers and source database. The criteria consist of database schema, logical dependencies between fields in each table, referential integrity of database, quantity of generated records, etc. The goal of RealGen is to generate test data that is meaningful as the actual data of the software under test.

Keywords: Test Data Generation, Software Testing

1. Introduction

Software testing is one of the essential processes in the software development lifecycle and it takes a lot of cost and effort to complete because software testing is the process of executing software with the intent of finding errors. Software testing requires test cases to test the software. Beside test cases, testers also need to prepare test data in the database which will be used by the software under test. Test data preparation can be done manually by hands, copying from old version, and automatically generated by software tools. Most contemporary software relies on relational database which could support many million records of data. Since a test environment is need to be as the actual environment, test data is also required to be as large as the actual data. With this reason, the manual preparation and the copying from old version are insufficient and unsuitable methods. The automation of test data generation will reduce the cost and effort of software development and testing significantly.

Unfortunately, the test data generation tools in market today focus mainly on the data type of fields and the structure of tables in data generation process but

most of them do not concern the logical dependencies between fields in each table (for example, a record of student data with prefix as "Mr." should has gender as "Male") and the referential integrity between tables. Thus, the generated test data is not meaningful to the software under test and human sense.

This paper presents a tool to automate test data generation process, named "RealGen". The paper focuses on test data generation which is valid for software under test. Test data generation for invalid cases and test case generation techniques are not discussed here. The paper is organized as follows. Section 2 discusses the characteristics of meaningful test data. Section 3 describes the features, usage, and limitations of the tool. Empirical studies are discussed in section 4, while section 5 compares the tool with related researches/tools. Section 6 suggests the future work and the conclusion of this paper is in section 7.

2. The Meaningful Test Data

The meaningful test data is the data that is syntactically and semantically correct to the context of software under test. Test data generation cannot be just arbitrarily random. It should takes consideration on some rules to enforce the test data to be meaningful. There are 2 important characteristics to be considered during data generation: Data type and Data dependency.

2.1 Data Type

Normally, the data is enforced by its data type. The data type enforces the generated test data to be syntactically correct to the context of software under test. For example, salary data field of employee records should not accept the character string values like "ldsef". Beyond the data type, there are some specific constraints of each data type to refine the generated test data to be more meaningful to specific context such as length of string data, numeric precision, scale of numeric data,

and etc. For example, “9,999,999,999” does not make sense to be a salary of an employee.

2.2 Data Dependency

Generally, there are also some dependencies in the data. These dependencies enforce the generated test data to be semantically correct to the context of software under test. There are 2 types of dependency: logical dependency and referential integrity.

Logical dependency is the inter-field dependency which can be normally found in data. For example, a record of personal data with the prefix as “Mr.” should has gender as “Male”. Logical dependency can be a constraint or a correlation rule that enforces data over several fields. With logical dependency, the generated test data can be more meaningful.

Referential integrity is another kind of dependency which can be found generally in relational database. Referential integrity is the dependency that a field in a table refers to another field in another table which means a record in parent table must be existed before a record in child table can be existed. For example, a record of “Employee” table, which “Deptno” field refers to “Deptno” field of “Department” table, must refer to a record of “Department” table that is already existed. Otherwise, there would be an employee with unknown department exists in the database, which is not valid to the context. This invalid data record leads to non-meaningful test data.

3. RealGen

RealGen (**Realistic test data Generator**) is a tool developed to automate test data generation process in software testing. RealGen randomly generates test data according to database schema, logical dependencies between fields in each table, and referential integrity between tables, supplied by testers and a source database. The process of RealGen is shown in figure 1. The database schema can be either created manually by testers or read from source database while logical dependencies are only supplied by testers. RealGen could read database schema from 3 types of databases: MySQL, Oracle, and Microsoft Access. Testers can define 6 properties for each field in each table. The properties are field name, data type, size (or numeric precision in case of numeric data type), numeric scale, identifier flag, and optional flag. RealGen provides 7 data types for testers to select for each field including character, integer, double, Boolean, date, time, and timestamp. Identifier flag indicates whether data of the field should be unique for every record and optional flag indicates whether the field is required to have a value for every record. For logical dependencies definition, testers could define logical dependencies in 2 modes. Firstly, testers could define logical dependencies as correlation rules between fields within the table; for example, if prefix field is “Mr.” then gender field must be “Male” (prefix field is correlated to gender field).

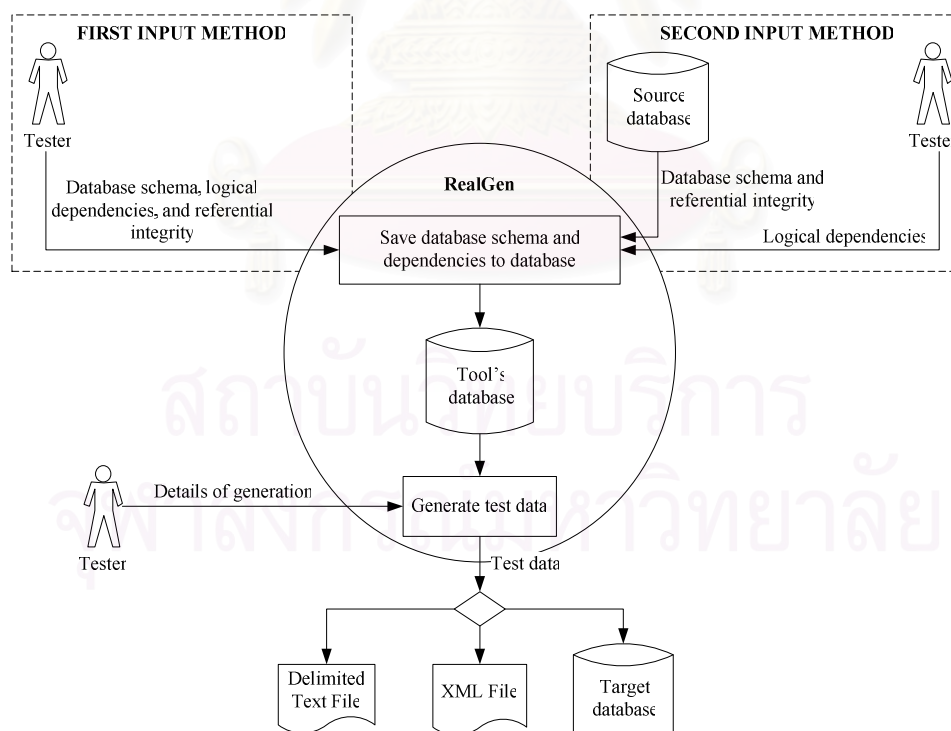


Figure 1. The process of RealGen

Another mode of logical dependencies is SQL. Testers could define logical dependencies as SQL query statement such as “select * from employee where if(prefix='Mr.', gender='Male', gender<>'Male')”. In order to define referential integrity of each field, testers must choose a target field to be referred by a current field. The target field must be an identifier of its table and has the same basic properties (data type, size, numeric scale) as the current field.

In order to start data generation process, testers need to specify some details of generation. Firstly, testers must choose tables to generate test data then testers need to choose the type of target database which generated test data will be loaded. Next, the numbers of records of each table must be specified. After that, testers need to choose a method of generation for each field in each table. There are 4 methods of generation: automatically derivation (default method), constant value, null value, and generic value. In automatically derivation, testers can specify the range of data to be used as basis during data generation. Constant value and null value mean that constant value (supplied by testers) and null value are generated for every record. Generic value means using categories of generic value (such as name, prefix, gender, province name, country name, and etc.) which provided by the tool to generate test data. Finally, testers must choose the output format of generated test data. RealGen provides 3 output formats: delimited text files, XML files, and populate data to target database directly.

In data generation process, RealGen generates test data according to the properties and the method of generation of each field. If the method of generation is automatic derivation, the data generation is as follows.

For character data type, RealGen randomly generates variable-length character string which length of string must not exceed the size of the field.

If integer data type is used, RealGen will randomly generates an integer value based on the range supplied by testers. In addition, testers must not specify the lower bound and upper bound of range that exceed minimum and maximum integer value of the target database (for example, if the type of target database is MySQL, the specified lower bound and upper bound must not exceed -2,147,483,648 and 2,147,483,647 respectively).

The generation of double data type is similar to integer data type. RealGen randomly generates a double value based on the range which is enforced by minimum and maximum double value of the target database.

If tester chooses Boolean data type for the field, RealGen will generates only 2 integer values (0 or 1) randomly for each record of table.

For date, time, and timestamp data type, the process of data generation is similar to integer and double data type. The test data are randomly generated based on the range supplied for that field. However, there are no

constraints about minimum and maximum value to be considered during the range specification.

If a field uses the constant value method, RealGen will generates, as mentioned above, a constant value supplied by testers for every record of this field and so as the null value method, every record of this field will be null.

Generic value method is the key to make the test data understandable by human sense. It is used when testers want the test data to be not just arbitrarily random. This method provides some categories of generic data, such as name of people, gender, blood type, career, and etc., for testers to use in data generation. When testers select this method for a field, data for this field in each record is randomly selected from all possible values of a selected category. In addition, RealGen provides more flexibility for testers to create more categories of generic data and update existing categories to be more useful in test data generation.

There will be an exception in the data generation process, if the field is a foreign key which refers to another field. The foreign field (referred field) must be generated data before the data generation of the current field. Then, each record of data in the current field is randomly selected from all generated records of the referred field.

After the data generation process, RealGen validates the generated test data of each table with logical dependencies. If logical dependencies are correlation rules, RealGen will apply these rules to each record of generated test data. If logical dependency is SQL query statement, RealGen will apply this query statement against the generated test data. Records that are invalid to query statement will be regenerated.

Finally, RealGen generates output of test data according to the format selected by testers. For delimited text file, RealGen generates a file per table which each field is delimited by a field delimiter and each record is delimited by a record delimiter (both delimiters are supplied by testers). If selected output format is an XML file, RealGen will convert generated test data into XML content conforming to XML schema of target database and also generates a file per table as delimited text file. If testers select to populate generated test data to the connected target database directly, RealGen will generate SQL insert statement for each record and execute this statement in target database.

3.1 Usage of RealGen

Figure 2 and figure 3 show the database schema editor screen which testers can create, edit, and delete tables in a database schema and fields of each table respectively. The properties of each field can be defined as in figure 4, and when testers want to define referential integrity of the field; they use the screen in figure 5.

Figure 6 shows the screen that defines logical dependencies of each table. In case of reading database schema from the source database, testers must specify the parameters to connect to the source database as in

figure 7. When testers want to start the test data generation process, they use the screen in figure 8 to specify the details of data generation and select the output format of generated test data in figure 9.

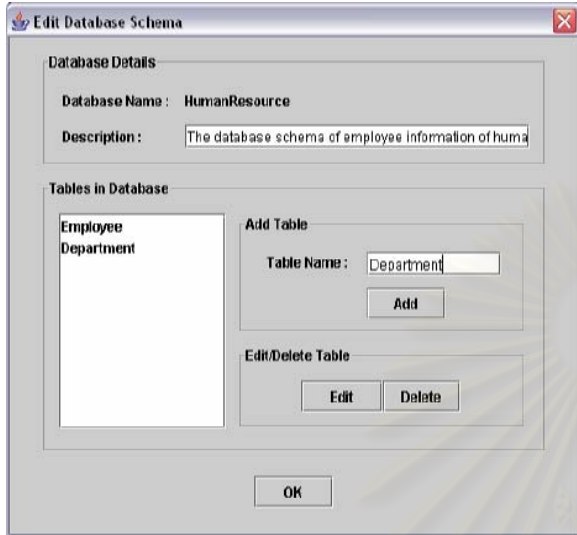


Figure 2. Specifying tables in database schema

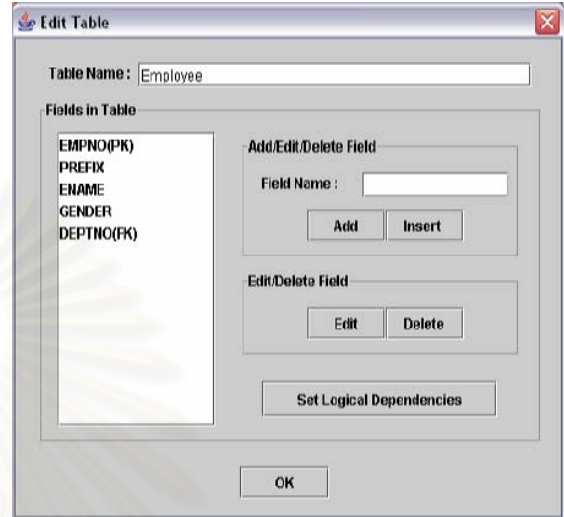


Figure 3. Specifying fields in table

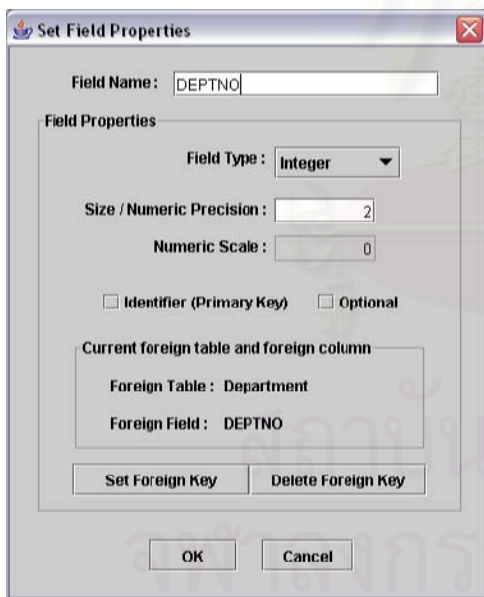


Figure 4. Specifying properties of field



Figure 5. Specifying referential integrity

Edit Logical Dependencies

Mode of Logical Dependency

Normal Logical Dependency SQL Logical Dependency

Add/Edit/Delete Normal Logical Dependencies

If the field are

then the field are

Current Normal Logical Dependencies

Source Field	Condition value	Destination Field	Result value
PREFIX	Mr.	GENDER	Male
PREFIX	Mrs. Miss	GENDER	Female

Specify SQL Logical Dependency

SQL statement :

Figure 6. Specifying logical dependencies of table

Change Target Database Connection

Target Database Connection

MySQL Database Microsoft Access

Oracle Database

Connection Parameters

User : Password :

DB Name :

Server/Hostname :

Port : Oracle-SID :

Figure 7. Specifying parameters of target database connection

Generate test data

Schema Tables

- HumanResource
 - Employee
 - Department

Generating Tables

- HumanResource
 - Employee
 - Department
 - DEPTNO(PK)
 - DNAME

Target DBMS of Test Data Generation

Target DBMS :

Quantity of records

Generation Quantity : records

Generation method for each columns

Auto. Derived : From => 0

 To => 99

Constant value :

Null : no value

Generic value : none

Figure 8. Specifying details of data generation

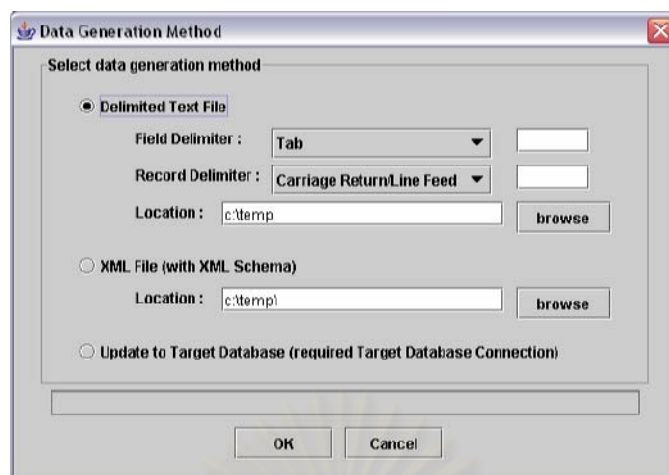


Figure 9. Specifying output format

3.2 Limitations

RealGen could support only 7 data types while the relational databases in the market today could support more than 7 data types such as CLOB, BLOB, BFILE, ROWID, and etc. Thus, when RealGen read database schema from a source database, it uses character data type instead of these non-support data types and it results in generated test data that is not conform to the original data types. Moreover, reading a database schema from the source database and populating test data to the target database are only supported for MySQL, Oracle, and Microsoft Access.

Finally, if testers choose to populate the generated test data to a target database, the target database must already have target tables because RealGen cannot create target tables by itself.

4. Empirical Studies

RealGen has been applied to generate test data for 2 real software systems: [1] and [2]. Consequently, generated test data for these software systems is examined. Both systems have both logical dependencies and referential integrity in the database schema. The result indicates that RealGen can generate test data that is meaningful to the context of both software systems.

Another empirical study of RealGen is the measurement of response time for the test data generation. Response time is a very significant factor of the test data generation thus a good test data generator must generate test data within an acceptable time. An experiment is set up to measure the response time of overall data generation process of RealGen (including the period of output generation). The experiment is conducted on a notebook which has Pentium M 1.4 GHz as processor and 512 MB DDR SDRAM. This experiment varies the test data generation on 3 factors: number of generated tables, referential integrity among tables (number of existing foreign keys and self-referenced relationship), and output of generation. Other factors are number of fields in each table, and number of generated records of each table. Table 1 shows the result of this experiment.

According to table 1, the result indicates that when there is no referential integrity among tables, the response time is relative low. When referential integrity is existed, the response time goes noticeably higher than the case with no referential integrity and it also indicates that the self-referenced relationship does not significantly affect to the response time. However, the response time is relatively acceptable when comparing to manually generation by testers. Finally, the result also indicates that when the number of tables grows, the response time is likely to grow in linear fashion.

Table 1. The result of response time experiment

No. of Tables	No. of Fields/Table	No. of Generated Records/ Table	Referential Integrity	Output Format	Response Time (hh:mm:ss)
5	10	1000	None	Delimited Text File	00:00:12
5	10	1000	None	XML File	00:00:11
5	10	1000	None	MySQL database	00:00:12
5	10	1000	None	Oracle database	00:00:30
5	10	1000	None	MS Access	00:00:17
10	10	1000	None	Delimited Text File	00:00:21
10	10	1000	None	XML File	00:00:23
10	10	1000	None	MySQL database	00:00:23
10	10	1000	None	Oracle database	00:00:55
10	10	1000	None	MS Access	00:00:33
5	10	1000	4 Foreign Keys	Delimited Text File	00:01:23
5	10	1000	4 Foreign Keys	XML File	00:01:21
5	10	1000	4 Foreign Keys	MySQL database	00:01:25
5	10	1000	4 Foreign Keys	Oracle database	00:01:36
5	10	1000	4 Foreign Keys	MS Access	00:01:27
10	10	1000	9 Foreign Keys	Delimited Text File	00:02:58
10	10	1000	9 Foreign Keys	XML File	00:02:51
10	10	1000	9 Foreign Keys	MySQL database	00:02:50
10	10	1000	9 Foreign Keys	Oracle database	00:03:25
10	10	1000	9 Foreign Keys	MS Access	00:03:02
5	10	1000	4 Foreign Keys, 1 Self	Delimited Text File	00:01:20
5	10	1000	4 Foreign Keys, 1 Self	XML File	00:01:20
5	10	1000	4 Foreign Keys, 1 Self	MySQL database	00:01:21
5	10	1000	4 Foreign Keys, 1 Self	Oracle database	00:01:28
5	10	1000	4 Foreign Keys, 1 Self	MS Access	00:01:25
10	10	1000	9 Foreign Keys, 1 Self	Delimited Text File	00:03:02
10	10	1000	9 Foreign Keys, 1 Self	XML File	00:02:53
10	10	1000	9 Foreign Keys, 1 Self	MySQL database	00:03:08
10	10	1000	9 Foreign Keys, 1 Self	Oracle database	00:03:26
10	10	1000	9 Foreign Keys, 1 Self	MS Access	00:02:57

5. Related Work

DGeST is a test data generating tool developed by [3]. To use this tool, testers must create the structure of table and determine the properties of each field manually because DGeST cannot automatically read structure of table from the existing database. This tool allows testers to provide the logical dependencies between fields in a table. The generated test data from DGeST is quite meaningful but not completely meaningful as actual data because the tool does not allow testers to provide the referential integrity among tables. DGeST can only generate a table at a time and output the generated test data into 2 formats: delimited text file and Microsoft Access file.

Datetect [4] is a test data generator developed by Banner Software, Inc. This tool allows testers to create data fields and specify the properties of them manually or read table structure from an actual database schema via ODBC. Then, testers must select the data fields to

use in output specification and also determine the number of records to generate. Datetect generates test data randomly following to the output specification. However, the generated test data is not meaningful as actual data because it considers neither logical dependencies between fields nor referential integrity among tables in database. Datetect is similar to DGeST since it generates a table at a time and provides 2 output formats including delimited text file and populate to database via ODBC.

6. Future Work

RealGen has to be further improved in several aspects such as supported data types, types of target database, and data definition and manipulation features. Currently, RealGen could support only 7 data types while relational database vendors in the market provides many various data types in their database such as CLOB, BLOB, BFILE, ROWID, SET, ENUM, and etc.

RealGen requires supporting these data types to generate a real meaningful test data to the context of software under test. Another aspect to be improved is types of target database. Since, RealGen could connect to only 3 relational databases (MySQL, Oracle, and Microsoft Access), it needs to be improved to support more famous relational databases such as Microsoft SQL Sever, DB2 Universal, and Sybase. Finally, RealGen is completely useful if it supports data definition and manipulation to target database. For example, testers can create tables, drop tables, query data, and delete data in target database instantly via the tool. Consequently, they are not required to do these operations on the database provided tools in order to use RealGen.

7. Conclusion

This paper presents a tool to automate the test data generation process, named RealGen. The main goal of RealGen is to generate test data that is meaningful as the actual data of the software under test. The meaningful test data is the data that is syntactically and semantically correct to the context of software under test. To be syntactically corrected, data type must be in consideration during the data generation process and being semantically corrected, the data generation process must consider the data dependency which includes logical dependencies between data fields and referential integrity among tables in database. RealGen generates test data by considering all of these factors so the generated test data is meaningful as the actual data in production environment. With RealGen, testers could generate test data with less effort and less errors (comparing to manually generation); hence the software

testing process is more cost-effective, reliable, and applicable.

8. Acknowledgement

This research is supported by TJTTP-OECF (Thai-Japan Technology Transfer Project – Japanese Overseas Economic Cooperation Fund) and Department of Computer Engineering – Industry Linkage Research Project, Year 2004, Chulalongkorn University.

9. References

- [1] S. Chaisang, “A Design and Development of an Out-Patient Information System for State Enterprise Medical Care Clinic”, Master Thesis, Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, 2000.
- [2] P. Thojun, “Development of a Decision Support System for Academic Selection”, Master Thesis, Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, 1996.
- [3] P. Sakulpipatsin, S. Supavita, and T. Suwannasart, “DGeST: A Data Generation Tool for Supporting Software Testing Process”, National Computer Science and Engineering Conference (NCSEC), Thailand, 2003.
- [4] Banner Software Inc., Datatect, <http://www.datatect.com/>
- [5] IEEE Computer Society, “IEEE Standard Glossary of Software Engineering Terminology”, 1983. ANSI/IEEE Std 729-1983.
- [6] Boris Beizer, “Software Testing Techniques Second Edition”, New York: Van Nostrand Reinhold, 1990.

ประวัติผู้เขียนวิทยานิพนธ์

นายกฤษณะ พิริยะกิจไพบูลย์ เกิดเมื่อวันที่ 20 พฤศจิกายน พ.ศ. 2522 ที่อำเภอวารินชำราบ จังหวัดอุบลราชธานี สำเร็จการศึกษาระดับประถมศึกษาจากโรงเรียนอนุบาลอุบลราชธานี จังหวัดอุบลราชธานี เมื่อปีการศึกษา 2534 ระดับมัธยมศึกษาจากโรงเรียนเบ็ญจะมะมหาราช จังหวัดอุบลราชธานี เมื่อปีการศึกษา 2540 สำเร็จการศึกษาระดับปริญญาตรี สาขาวิชาวิทยาการคอมพิวเตอร์ (เกียรตินิยมอันดับ 2) จากภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยขอนแก่น จังหวัดขอนแก่น เมื่อปีการศึกษา 2544 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2545 ที่อยู่ปัจจุบันที่สามารถติดต่อได้ คือ 150/1 ถนนกันทรลักษณ์ ตำบลวารินชำราบ อำเภอวารินชำราบ จังหวัดอุบลราชธานี 34190 เบอร์โทรศัพท์ +66 45 323048 อีเมลล์ laruku_jr@hotmail.com



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย