



ระบบสารสนเทศระบบทะเบียนและประมวลผล
 ส่วนงานลงทะเบียนและค่าธรรมเนียม ผ่านระบบเครือข่ายอินเทอร์เน็ต

The office Of The Register's Information System
 Registration and Fee Section via Internetwork System

โดย

นายวิทยา แก้วรัศมีกุล	รหัส	38013289
นายอนุชิต จันสศิริพานิช	รหัส	38013301
นายอนุสรณ์ เหลืองรุ่งทรัพย์	รหัส	38013302

อาจารย์ที่ปรึกษา

อาจารย์วิบูลย์ พร้อมพานิชย์

วัน เดือน ปี 15.ค.ค.2541
 เลขทะเบียน.....038993
 เลขเรียกหนังสือ...1. 20234 0.582 7

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
 ภาควิชา วิศวกรรมคอมพิวเตอร์
 คณะวิศวกรรมศาสตร์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่นใด
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038993

ปริญญาโทปีการศึกษา 2540

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบสารสนเทศของสำนักทะเบียนและประมวลผล ส่วนงานลงทะเบียนและค่าธรรมเนียม ผ่านเครือข่ายอินเทอร์เน็ต

ผู้จัดทำ

1. นายวิทยา แก้วรัศมีกุล
2. นายอนุชิต จันสธิรพานิช
3. นายอนุสรณ์ เหลืองรุ่งทรัพย์



..... อาจารย์ที่ปรึกษา

(วิบูลย์ พร้อมพานิชย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบสารสนเทศของสำนักทะเบียนและประมวลผล
ส่วนงานลงทะเบียนและค่าธรรมเนียม ผ่านระบบเครือข่ายอินเทอร์เน็ต

วิทยา แก้วรัมย์กุล

อนุชิต จันสถิรพานิช

อนุสรณ์ เหลืองรุ่งทรัพย์

วิบูลย์ พร้อมพานิชย์ อาจารย์ที่ปรึกษา

ปีการศึกษา 2540

บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอ การพัฒนาระบบการลงทะเบียนเรียนของนักศึกษา โดยใช้ระบบเครือข่ายอินเทอร์เน็ตเป็นตัวกลางเชื่อมต่อระหว่างคอมพิวเตอร์ที่ให้บริการ (Server) กับคอมพิวเตอร์ของนักศึกษาที่ใช้งาน (Client) เพื่อการจัดการการลงทะเบียนเรียนแบบครบวงจร

ในการพัฒนาระบบมีการนำเอาระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System : RDBMS) ที่มีชื่อว่า Microsoft SQL Server 6.5 และระบบปฏิบัติการ (Operating System) ที่มีชื่อว่า Microsoft Window NT 4.0 Thai Enable เข้ามาใช้ในการจัดการระบบการลงทะเบียนทั้งหมด โดยมีระบบการบริการข้อมูลสารสนเทศทางอินเทอร์เน็ต (Internet Information Server : IIS) ที่ทำงานภายใต้ระบบปฏิบัติการ Windos NT เป็นตัวจัดการการข้อมูลระหว่างคอมพิวเตอร์ที่ให้บริการ กับคอมพิวเตอร์ที่รับบริการที่ใช้โปรแกรมสืบค้นข้อมูล (Web Browser) ในการทำงาน

**The Office Of The Register's Information System
Registration and Fee Section via Internetwork System**

Withaya Gaewrasameekul

Anichit Chansatirapanich

Anusorn Luangrungsub

Wiboon Prompanich Advisor

1997

ABSTRACT

This thesis presents the development of the study registration of student. Using internetwork system to connect between a computer server and the computer client at student uses. Which used for management the study registration system in closed circuit.

During the implementation , this uses Microsoft SQL Server 6.5 to be a RDBMS (Relational Database Management System) and uses the Microsoft Window NT 4.0 Thai Enable for operating system come to manage all of the registration system. For the Internet Information System (IIS), It is run under the Microsoft Window NT. The Internet Information System is used to manage the information between the computer server and the computer client which runing the web browser in the process of work.

สารบัญ

เรื่อง	หน้าที่
บทที่ 1 บทนำ	1
บทที่ 2 การออกแบบฐานข้อมูล	3
2.1 โมเดลเชิงสัมพันธ์	3
2.2 ระบบจัดการฐานข้อมูลเชิงสัมพันธ์	3
2.3 การออกแบบฐานข้อมูลเชิงสัมพันธ์	4
2.4 การออกแบบฐานข้อมูลโดยใช้การ Normalization	4
2.5 การออกแบบฐานข้อมูลโดยใช้ NIAM	5
2.6 โอเพ่นดาต้าเบสคอนเน็คติวิตี (ODBC : Open Database Connectivity)	7
2.7 สตอร์โพชีเยอร์ (Stored Procedure)	9
บทที่ 3 Internet Information Server	13
3.1 Internet Information Server	13
3.2 การนำ Internet Information Server มาใช้งาน	13
3.3 Internet Information Server ทำงานอย่างไร	13
3.4 นำ Internet Information Server มาใช้งานได้อย่างไร	16
3.5 คุณลักษณะของ Internet Database Connector	19
3.6 การทำงานของระบบ	29
บทที่ 4 Java Script	31
4.1 Java Script	31
4.2 Java Script ไม่ใช่ Java	31
4.3 ข้อจำกัดของ Java Script	33
4.4 ชนิดของข้อมูลและการตั้งชื่อตัวแปร	35
4.5 การกำหนด Literals	35
4.6 ฟังก์ชัน (Function)	36
4.7 ระบบสมการ (Expression)	38
4.8 โอเปอเรเตอร์ (Operator)	39
4.9 อ็อบเจ็กต์ (Object)	42
4.10 อ็อบเจ็กต์โมเดลของ Java Script (Java Script Objects)	42
4.11 Event ใน Java Script	44
4.12 การใช้งานฟอร์ม	45

สารบัญ (ต่อ)

เรื่อง	หน้าที่
บทที่ 5 ระบบรักษาความปลอดภัยบน Internet	51
5.1 การป้องกันผู้บุกรุก	51
5.2 การเข้ารหัสบน Web	54
บทที่ 6 ISAPI (Internet Server Application Programming Interface)	59
6.1 ISAPI	59
6.2 การทำงานของ ISAPI	59
6.3 Common Gateway Interface (CGI)	61
6.4 ข้อเปรียบเทียบระหว่าง ISAPI กับ CGI	62
6.5 การควบคุม Web Server อย่างละเอียด	64
6.6 การถ่ายทอดตำแหน่ง URL	64
6.7 การรักษาความปลอดภัย	64
บทที่ 7 Hyper Text Markup Language (HTML)	65
7.1 HTML	65
7.2 การเขียน HTML เบื้องต้น	66
7.3 การเชื่อมโยงหรือ Linking	70
7.4 mailto	71
บทที่ 8 การดำเนินงานและการพัฒนาระบบ	74
8.1 วัตถุประสงค์ของโครงการ	74
8.2 ขอบเขตของโครงการ	74
8.3 ขั้นตอนในการพัฒนาโครงการ	75
8.4 โครงสร้างของระบบ	76
8.5 การวิเคราะห์และออกแบบระบบ	76
8.6 ลำดับขั้นการทำงานของระบบ	88
8.7 ลักษณะของหน้าจอส่วนที่ติดต่อกับผู้ใช้ (User Interface)	90
บทที่ 9 การทดลองและผลการทดลอง	95
9.1 ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการทดลอง	95
9.2 ความต้องการทางด้านซอฟต์แวร์	96
9.3 การทดลองและผลการทดลอง	97
9.4 สรุปผลการทดลอง	104

สารบัญ (ต่อ)

เรื่อง	หน้าที่
บทที่ 10 สรุป วิจัย และแนวทางในการพัฒนา	105
10.1 ความสามารถของระบบ	105
10.2 ข้อจำกัดของระบบ	105
10.3 แนวทางการพัฒนาต่อ	105
10.4 วิจัยโครงการ	106



สารบัญรูปภาพ

รูป	หน้าที่
รูปที่ 2.1 แสดงสัญลักษณ์ของชนิดเอนดีทีภาคีภาควิชา	5
รูปที่ 2.2 แสดงสัญลักษณ์ของชนิดเลเบลภาควิชา	6
รูปที่ 2.3 แสดงความสัมพันธ์อ้างอิงแบบ one to one	6
รูปที่ 2.4 แสดงการเขียนความสัมพันธ์อ้างอิงแบบ one to one อย่างย่อ	6
รูปที่ 2.5 แสดงความจริงแบบ many to one	6
รูปที่ 2.6 แสดงความจริงแบบ many to many	6
รูปที่ 2.7 โครงสร้างแบบลำดับชั้นของ ODBC	7
รูปที่ 3.1 การเชื่อมต่อของไคลเอนต์กับเซิร์ฟเวอร์ผ่านเครือข่ายอินเทอร์เน็ต	13
รูปที่ 3.2 การร้องขอของไคลเอนต์และการตอบสนองของเซิร์ฟเวอร์	14
รูปที่ 3.3 การที่ผู้ใช้งานจะติดต่อกับฐานข้อมูลโดยใช้ Inter net Database Connector	15
รูปที่ 3.4 ลักษณะของ Directory Listing	16
รูปที่ 3.5 ลักษณะของการเชื่อมต่อ Intranet	16
รูปที่ 3.6 ลักษณะของการประยุกต์ใช้ Intranet ในบริษัท	17
รูปที่ 3.7 การตั้งเว็บเซิร์ฟเวอร์ที่รัน IIS	17
รูปที่ 3.8 ลักษณะของเครือข่ายในบริษัทซึ่งใช้เว็บเซิร์ฟเวอร์ที่รัน IIS	18
รูปที่ 3.9 การที่เครือข่ายของหลายๆประเทศเชื่อมต่อกันผ่านอินเทอร์เน็ต	18
รูปที่ 3.10 แสดงขั้นตอนการทำงานของระบบ	29
รูปที่ 4.1 อีปเจ็คต์โมเดลของ Java Script	43
รูปที่ 5.1 แสดงขั้นตอนกระบวนการรักษาความปลอดภัยของ IIS	51
รูปที่ 5.2 ลำดับชั้นของการเชื่อมต่อ	54
รูปที่ 6.1 ISAPI Extension (Application)	59
รูปที่ 6.2 ISAPI Filter	60
รูปที่ 6.3 ลักษณะของเว็บไซต์ที่ใช้ ISAPI Filter และ ISAPI Application	61
รูปที่ 6.4 Common Gateay Interface (CGI)	61
รูปที่ 8.1 โครงสร้างหลักของระบบลงทะเบียนเรียนผ่านอินเทอร์เน็ต	76
รูปที่ 8.2 แผนผังการไหลของข้อมูลในระดับที่ 1	77
รูปที่ 8.3 แผนผังการไหลของข้อมูลในระดับที่ 2	78
รูปที่ 8.4 แผนผังการไหลของข้อมูลในระดับที่ 3(1)	78
รูปที่ 8.5 แผนผังการไหลของข้อมูลในระดับที่ 3(2)	79

สารบัญรูปรูป (ต่อ)

รูป	หน้าที่
รูปที่ 8.6 การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM ส่วนของการลงทะเบียน	80
รูปที่ 8.7 การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM ส่วนของการเก็บค่าธรรมเนียม (1)	81
รูปที่ 8.8 การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM ส่วนของการเก็บค่าธรรมเนียม (2)	82
รูปที่ 8.9 การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM ส่วนของวิชาต่อเนื่อง	83
รูปที่ 8.10 การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM ส่วนของตารางประเภทวิชา	83
รูปที่ 8.11 เพจแรกของระบบลงทะเบียน	91
รูปที่ 8.12 เพจเมื่อ Login ผ่าน	91
รูปที่ 8.13 เพจแรกของระบบการลงทะเบียนเรียน	92
รูปที่ 8.14 เพจของการเปลี่ยนแปลงรายวิชา	92
รูปที่ 8.15 เพจของการเพิ่มรายวิชา	93
รูปที่ 8.16 เพจของการถอนรายวิชา	93
รูปที่ 8.17 เพจของการตรวจสอบ	94
รูปที่ 9.1 แสดงหน้าจอรอข้อมูลนำเข้าจากผู้ใช้งาน	97
รูปที่ 9.2 แสดงหน้าจอที่ผ่านการตรวจสอบการเข้าใช้งานระบบ	97
รูปที่ 9.3 ภาพแสดงหน้าจอการยืนยันการเลือกรายวิชา	98
รูปที่ 9.4 ภาพแสดงหน้าจอหลังการยืนยันการลงทะเบียน	98
รูปที่ 9.5 ภาพแสดงหน้าจอการยืนยันการเปลี่ยนแปลงรายวิชา	99
รูปที่ 9.6 ภาพแสดงหน้าจอหลังการเปลี่ยนแปลงวิชาเรียน	99
รูปที่ 9.7 ภาพแสดงหน้าจอการยืนยันการเพิ่มวิชาเรียน	100
รูปที่ 9.8 ภาพแสดงหน้าจอหลังการเพิ่มรายวิชา	100
รูปที่ 9.9 ภาพแสดงหน้าจอยืนยันการถอนวิชาเรียน	101
รูปที่ 9.10 ภาพแสดงหน้าจอหลังการถอนรายวิชาเรียน	101
รูปที่ 9.11 ภาพแสดงหน้าจอการตรวจสอบรายการต่างๆ	102
รูปที่ 9.12 ภาพแสดงหน้าจอหลังการตรวจสอบรายการวิชาที่ลงทะเบียน	103
รูปที่ 9.13 ภาพแสดงหน้าจอหลังการตรวจสอบรายการชำระค่าธรรมเนียม	103
รูปที่ 9.14 ภาพแสดงหน้าจอหลังการตรวจสอบค่าธรรมเนียม	104

สารบัญตาราง

ตาราง	หน้าที่
ตารางแสดงฟิลด์ที่ต้องการในไฟล์ Internet Database Connector (.idc)	20
ตารางแสดงฟิลด์ที่เป็น Option ในไฟล์ Internet Database Connector (.idc)	20
ตารางแสดง ODBC Advanced Option Fields	22
ตารางแสดงโอเปอเรเตอร์ของ Statement <%if%>	25
ตารางแสดงตัวแปรของ Internet Information Seever	27
ตารางแสดงการเปรียบเทียบระหว่าง Java Script และ Java	33
ตารางแสดงตัวจับเหตุการณ์ที่เหมาะสมกับเหตุการณ์ที่เกิดขึ้น	44
ตารางของ Properties, Methods และ Event Handlers ของอิลิเมนต์ต่างๆ	49
ตารางการเก็บค่าธรรมเนียม (Fee)	84
ตารางค่าธรรมเนียม (FeeCost)	84
ตารางรายการค่าธรรมเนียมย่อย (FeeSubType)	85
ตารางรายการยกเว้นค่าธรรมเนียม (FeeExcept)	85
ตารางชื่อรายการยกเว้นค่าธรรมเนียม (FeeExceptName)	85
ตารางชนิดค่าธรรมเนียม (FeeType)	85
ตารางวิชาที่ลงทะเบียน (Register)	86
ตารางใบลงทะเบียน (RegisterCard)	86
ตารางวิชาต่อเนื่อง (Prerequisite)	87
ตารางประเภทวิชา (SubjectType)	87
ตารางTemp (TempTable)	87

บทที่ 1

บทนำ

จากสภาพการณ์ปัจจุบันการลงทะเบียนของนักเรียนนั้นนักเรียนจะต้องเดินทางไปลงทะเบียนที่สถาบัน
 เองที่ฝ่ายทะเบียนสถาบันที่นักเรียนนั้นศึกษาอยู่และในบางที่ขั้นตอนการลงทะเบียนนั้นค่อนข้างยุ่งยากเช่นนัก
 ศึกษาคงกรอกใบลงทะเบียนเอง ต้องไปยื่นใบลงทะเบียนเองและบางครั้งสถานที่ ๆ ชำระเงินก็อยู่คนละที่กับฝ่าย
 ทะเบียนทำให้ต้องชำระค่าธรรมเนียมอีกที่หนึ่ง และการลงทะเบียนแบบนี้จะต้องให้นักศึกษาให้มาลงทะเบียน
 ณ. วันเวลาที่กำหนด เพราะฉะนั้นจะเห็นได้ว่าการลงทะเบียนในลักษณะนี้ค่อนข้างยุ่งยาก และเสียเวลา ความต้อง
 การที่จะแก้ไขปัญหาเหล่านี้ด้วยวิธีการทางคอมพิวเตอร์ จึงได้มีแนวความคิดที่จะนำระบบเครือข่ายคอมพิวเตอร์มา
 ใช้พัฒนาระบบการลงทะเบียนให้มีประสิทธิภาพและอำนวยความสะดวกให้กับทุกฝ่ายที่เกี่ยวข้อง

ระบบเครือข่ายอินเทอร์เน็ต เป็นความสำเร็จของการพัฒนาเทคโนโลยีการสื่อสารด้วยคอมพิวเตอร์ซึ่งกำลัง
 ได้รับความนิยมอย่างมากในปัจจุบัน เหตุผลสำคัญได้แก่ความสามารถของคอมพิวเตอร์ซึ่งเป็นอุปกรณ์หรือเครื่อง
 มือของเครือข่าย อินเทอร์เน็ต โดยสามารถรับและส่งข้อมูลได้ทุกรูปแบบและเป็นเครือข่ายที่เชื่อมโยงถึงกันทั่วโลก
 สามารถส่งข้อมูลไปยังฐานข้อมูลของเครื่องปลายทางที่อยู่ไกลๆ และสามารถดึงข้อมูลจากเครื่องปลายทางที่อยู่
 ไกลๆ ได้

จากข้อดีของระบบเครือข่ายอินเทอร์เน็ตนี้ จึงได้ถูกนำมาประยุกต์ใช้ร่วมกับระบบฐานข้อมูลของการลง
 ทะเบียนนักศึกษา เพื่อพัฒนาเป็น "ระบบการทะเบียนนักศึกษาผ่านทางเครือข่ายอินเทอร์เน็ต" ซึ่งนักศึกษา
 สามารถที่จะลงทะเบียนผ่านทางเว็บไซต์ของการลงทะเบียนนี้ โดยที่นักศึกษาไม่ต้องเดินทางมาถึงสถาบันเพื่อมาลง
 ทะเบียนเรียนด้วยตัวเอง ซึ่งจะช่วยให้มีความคล่องตัว ลดขั้นตอนที่ยุ่งยากซับซ้อน ประหยัดเวลา และประหยัดค่า
 เดินทาง เป็นอย่างมาก

โครงการนี้เป็นการพัฒนาในส่วนของเว็บเซิร์ฟเวอร์ (Web Server) เพื่อรองรับการขอใช้บริการจากเครื่อง
 ลูกข่าย (Client) ที่ใช้บราวเซอร์ (Browser) เข้ามาทำการกับระบบ โดยในเว็บเซิร์ฟเวอร์แบ่งการพัฒนาออกเป็น 2
 ส่วนหลัก ๆ คือ

- ส่วนที่เป็นระบบฐานข้อมูลที่จัดการเกี่ยวกับข้อมูลนักศึกษา ข้อมูลลงทะเบียน ข้อมูลค่าธรรมเนียม
 ตลอดจนข้อมูลด้านผลการเรียนของนักศึกษา
- ส่วนที่เป็นการระบบการจัดการข้อมูลส่วนกลาง จะเกี่ยวกับการเป็นตัวกลางการเชื่อมต่อการส่งข้อมูล
 ระหว่างเครื่องลูกข่าย (Client) และเครื่องแม่ข่าย (Web Server)

สำหรับการพัฒนาระบบจะเริ่มจากการศึกษาระบบฐานข้อมูลเดิม เพื่อนำมาประยุกต์ใช้ให้เข้ากับระบบใหม่ที่ทำ
 การคิดค้นขึ้น ทำการวิเคราะห์ระบบทั้งหมดเพื่อหาเครื่องมือ (Tool) ที่เหมาะสมในการพัฒนา ทำการออกแบบระบบให้ม
 ความสัมพันธ์กันทั้งหมด และพัฒนาตามจุดประสงค์ที่ได้วางไว้ เพื่อให้ได้ระบบงานลงทะเบียนที่มีประสิทธิภาพมากที่สุด

รายละเอียดของปัญญานิพนธ์ฉบับนี้สามารถแบ่งได้ตามหัวข้อต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1 : บทนำกล่าวถึงที่มา รายละเอียดเกี่ยวกับปริญญาบัตรฉบับนี้

บทที่ 2 : ทฤษฎีและหลักการเกี่ยวกับระบบฐานข้อมูล เช่นแนวความคิดเกี่ยวกับการจัดการฐานข้อมูลแบบรีเลชันการออกแบบฐานข้อมูล เป็นต้น

บทที่ 3 : ทฤษฎีและหลักการเกี่ยวกับระบบบริการข่าวสารข้อมูลทางอินเทอร์เน็ต (Internet Information Server : IIS)

บทที่ 4 : ทฤษฎีและหลักการเกี่ยวกับการใช้งาน Java Script บนเว็บเพจ เช่น เรื่องเหตุการณ์(Event) ระบบสมการ อ็อบเจกต์ภายใน และการใช้งานฟอร์ม เป็นต้น

บทที่ 5 : ทฤษฎีและหลักการเกี่ยวกับการรักษาความปลอดภัยบนบราวเซอร์ จะบรรยายเกี่ยวกับระบบรักษาความปลอดภัยแบบใช้กุญแจสองคอก และระบบโปรโตคอลแบบ Secure HTTP (S-HTTP)

บทที่ 6 : ทฤษฎีและหลักการเกี่ยวกับกับระบบของ Internet Server Application Programming Interface (ISAPI) รวมถึงระบบของ CGI พร้อมทั้งการเปรียบเทียบข้อดีและข้อเสียต่าง ๆ

บทที่ 7 : ทฤษฎีและหลักการเกี่ยวกับการใช้งานภาษา Hyper Text Markup Language (HTML) สำหรับการสร้างเว็บเพจเพื่อการติดต่อกับบราวเซอร์บนฝั่ง Client ด้วยแท็กภาษาของ HTML

บทที่ 8 : การบรรยายเกี่ยวกับการออกแบบและพัฒนาในระบบในส่วน โครงสร้าง ตารางฐานข้อมูล โมเดลฐานข้อมูล ลักษณะของโครงการ และลำดับการทำงานต่าง ๆ

บทที่ 9 : การทดลองและผลการทดลองใช้งานระบบลงทะเบียน เป็นการทดลองใช้งานระบบขั้นต้น เพื่อดูผลลัพธ์การทำงานที่เกิดขึ้น

บทที่ 10 : การบรรยายเกี่ยวกับความสามารถของระบบ ข้อจำกัดของระบบ และการวิจารณ์โครงการ

บทที่ 2

การออกแบบฐานข้อมูล

2.1 โมเดลเชิงสัมพันธ์

โมเดลเชิงสัมพันธ์คือ การอธิบายความสัมพันธ์ของข้อมูลที่จัดเก็บด้วยระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System : RDBMS) ซึ่งคิดค้นโดย Codd's เป็นโครงสร้างของข้อมูลในรูปแบบตาราง คำว่า Relational นั้นไม่ใช่ Relation ship (ความสัมพันธ์) คำว่า Relation คือ Subset ของ Cartesian Product ของ Domain

ข้อดี ของโมเดลเชิงสัมพันธ์มีดังนี้

- โครงสร้างข้อมูลนั้นง่ายต่อการใช้งาน
- ภาษาที่ใช้มันมีความสามารถสูงเช่น SQL (Structure Query Language)
- มีเครื่องมือที่ช่วยให้ผู้ใช้จัดการกับฐานข้อมูล
- ง่ายต่อการแก้ไขการออกแบบที่ผิดพลาด
- มีความสอดคล้องกับหลักการของฐานข้อมูล

2.2 ระบบจัดการฐานข้อมูลเชิงสัมพันธ์

ถ้ามีการนำเสนอระบบงานฐานข้อมูลสิ่งที่เราควรพิจารณา มีดังนี้

1. **Data Structure** มี 1 เดียวคือ ต้องเป็น Relation เท่านั้น
2. **Integrity Rules** เป็นกฎข้อบังคับความถูกต้องของการจัด โครงสร้างข้อมูลมีดังนี้
 - Entity Integrity : Primary Key Value จะเป็น null ไม่ได้
 - Referential Integrity : Foreign Key Value จำเป็นที่จะต้อง match กับ Primary Key Value หรือ ไม่ก็ต้องเป็น null
3. **Data Manipulation** : ภาษาต้องเป็น Relational Complete Language

ในระบบฐานข้อมูลเชิงสัมพันธ์นั้นจะมีความสัมพันธ์ของข้อมูลอยู่ 3 แบบภายในระหว่าง Table

one-to-one การที่มีเรคคอร์ด 1 เรคคอร์ดใน Table มีความสัมพันธ์กับอีก 1 เรคคอร์ดใน primary table

one-to-many การที่มีเรคคอร์ด 1 เรคคอร์ดหรือมากกว่าในการเชื่อมความสัมพันธ์กับ 1 เรคคอร์ดใน primary table

many-to-many การที่มี เรคคอร์ด 1 เรคคอร์ดหรือมากกว่าในการเชื่อมความสัมพันธ์กับ 1 เรคคอร์ดหรือมากกว่าใน primary table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การออกแบบฐานข้อมูลเชิงสัมพันธ์

ในปัจจุบันวิธีการออกแบบฐานข้อมูลที่รู้จักกันอย่างแพร่หลายได้แก่ การออกแบบฐานข้อมูลโดยใช้การ Normalization ซึ่งเป็นวิธีที่มีแนวคิดในการปรับปรุงคุณสมบัติของรีเลชันอย่างเป็นขั้นตอน อย่างมีระบบเพื่อลดโอกาสที่จะเกิดความผิดพลาดเนื่องจากการเปลี่ยนแปลงข้อมูล ซึ่งจะมีขั้นตอนหลักๆ 5 ขั้นตอน ผลลัพธ์สุดท้ายเป็น Fifth Normal Form (5NF) จะได้ฐานข้อมูลที่ถูกต้องที่สุดและไม่มีความซ้ำซ้อน

อย่างไรก็ตามถึงแม้การ Normalization จะวิธีที่สามารถออกแบบฐานข้อมูลได้ดีแต่วิธีนี้จะไม่เหมาะสมเมื่อต้องการที่จะออกแบบระบบฐานข้อมูลขนาดใหญ่ที่มี Attribute จำนวนมากซึ่งทำให้การออกแบบมีความยากลำบาก

วิธีการออกแบบฐานข้อมูลโดยใช้ NIAM (Nijssen's Information System Analysis Method) เป็นเครื่องมือในการออกแบบ เนื่องจากแนวคิดที่จะให้ Conceptual Schema มีพื้นฐานมาจากโครงสร้างภาษารธรรมชาติ ใช้รูปประโยคที่มี ประธาน ,กริยา ,กรรม วิธีแสดงรูปแบบความสัมพันธ์ของข้อมูลเป็นแบบจำลองที่มีความหมายและมีเครื่องหมายแสดงความสัมพันธ์ของข้อมูลและข้อจำกัดของข้อมูลได้อย่างชัดเจน นอกจากนี้ยังสามารถแปลง Conceptual Schema เป็น Relational Database Schema ซึ่งอยู่ในรูปของ Fifth Normal Form เนื่องจากวิธีนี้ใช้รูปสัญลักษณ์ที่แสดงความสัมพันธ์ของข้อมูลและง่ายต่อการทำความเข้าใจ ดังนั้นจึงสะดวกในการออกแบบฐานข้อมูลของระบบงานใหญ่ๆ และโครงการนี้ก็ได้ออกแบบฐานข้อมูลโดยใช้ NIAM

2.4 การออกแบบฐานข้อมูล Normalization

วิธีการ Normalization นั้นจะเป็นวิธีที่จะนำมาใช้ในการขจัดความซ้ำซ้อนของข้อมูลในฐานข้อมูล ซึ่งมีจุดมุ่งหมายเพื่อให้การ Insert ,Delete ,Update เร็กคอร์ดภายใน Table โดยปราศจากข้อผิดพลาดซึ่งจะมีขั้นตอนต่างๆ ดังนี้

First Normal Form (1NF) คือ Relation ใดๆจะอยู่ใน 1NF ก็ต่อเมื่อ ไม่มี repeating group คือ attribute จะไม่สามารถแบ่งย่อยต่อไปได้ (Atomic Value)

Second Normal Form (2NF) คือ จะต้องเป็น 1NF และทุกๆ non key จะต้องขึ้นกับ Primary key เท่านั้น

Third Normal Form (3NF) คือ จะต้องเป็น 2NF ก่อนและ non key attribute จะต้องไม่ขึ้นอยู่กับตัวเอง 3NF นั้นส่วนใหญ่คืออยู่แล้วยกเว้น 3NF ที่มีลักษณะ 3 ประการดังนี้

- มี multiple candidate keys
- candidate key นั้นเป็น composite key คือ key ที่ประกอบมาจากมากกว่า 1 attribute
- candidate key นั้น overlap กัน

ถ้ามีครบ 3 ประการกล่าวได้ว่าเป็น 3NF ที่ไม่ใช่ BCNF คือยังไม่ดีพอ

Boyce-Codd Normal Form (BCNF) คือ ทุกๆ determinant เป็น candidate key

Fourth Normal Form (4NF) คือ Multi-Valued dependencies จะถูกขจัดทิ้ง

Fifth Normal Form (5NF) คือ ไม่สามารถ Split ตารางต่อไปได้แล้ว

2.5 การออกแบบฐานข้อมูลโดยใช้ NIAM

Nijssen เป็นผู้ที่ยึดมั่นโมเดลแบบนี้ขึ้นมาเพราะต้องการโมเดลใหม่ที่คนส่วนใหญ่ใช้ได้ โดยที่ไม่จำเป็นต้องทราบ Normal form เพราะฉะนั้นคนเขียน NIAM เสร็จแล้วแม่ปทีเดียวได้ 5NF (Fifth Normal form) เลย

2.5.1 NIAM มีขั้นตอนการออกแบบอยู่ 9 ขั้นตอน (CSDP 9 steps : Conceptual Schema Design Procedure)

- step 1 : กำหนดขอบเขตของงาน (Universe of Discourse : UoD) และความจริงที่เกิดขึ้นภายในขอบเขตของงานที่กำหนดไว้
- step 2 : วาด Conceptual Schema Diagram โดยคร่าวๆ จากความจริงในขอบเขตงาน
- step 3 : จัดรูปของ schema ให้เป็นระเบียบและหาชนิดความจริงที่ได้รับข้อมูลมาจากความจริงชนิดอื่น
- step 4 : เพิ่มสัญลักษณ์แสดง Uniqueness constraints
- step 5 : ตรวจสอบความถูกต้องของชนิดความจริง
- step 6 : เพิ่มสัญลักษณ์แสดง Lexical ,Mandatory Role, Subtype constraint
- step 7 : ตรวจสอบ Unique Identifier ของแต่ละชนิดเอนตีตี้
- step 8 : เพิ่มสัญลักษณ์แสดง Equality ,Exclusion ,Subset constraints
- step 9 : ตรวจสอบความสมบูรณ์ของ Conceptual Schema ว่าต้องสอดคล้องกับตัวอย่างข้อมูล และไม่มี ความซ้ำซ้อนของข้อมูล

2.5.2 ส่วนประกอบพื้นฐานของ NIAM

- ชนิดเอนตีตี้ (Entity Type)
- ชนิดเลเบล (Label Type)
- ชนิดความจริง (Fact Type)
- ชนิดอ้างอิง (Reference Type)
- ข้อจำกัดเพื่อความถูกต้องของข้อมูล (Integrity Constraints)

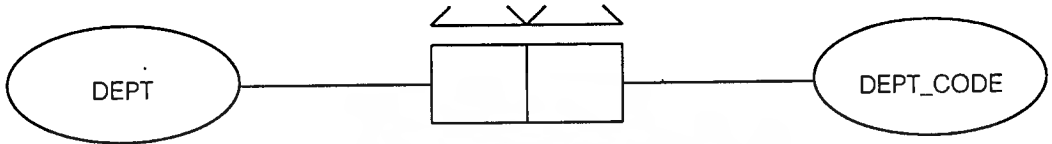
2.5.3 สัญลักษณ์และตัวอย่างการใช้ส่วนประกอบพื้นฐานของแบบจำลอง NIAM แสดงไว้ดังรูป

DEPT

แสดงสัญลักษณ์ของชนิดเอนตีตี้ภาควิชา



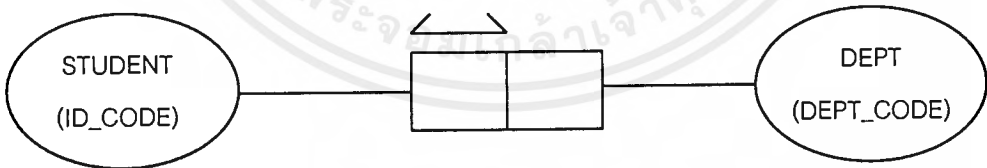
แสดงสัญลักษณ์ของชนิดเลเบลภาควิชา



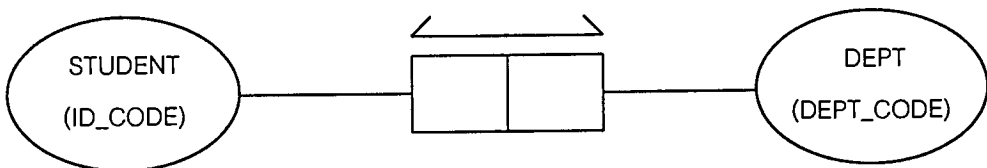
แสดงความสัมพันธ์อ้างอิงแบบ one to one หมายความว่า ภาควิชาใดจะมีรหัสภาควิชาเพียง
รหัสเดียวเท่านั้นและไม่ซ้ำกับภาควิชาอื่นๆ



แสดงการเขียนความสัมพันธ์อ้างอิงแบบ one to one อย่างย่อ



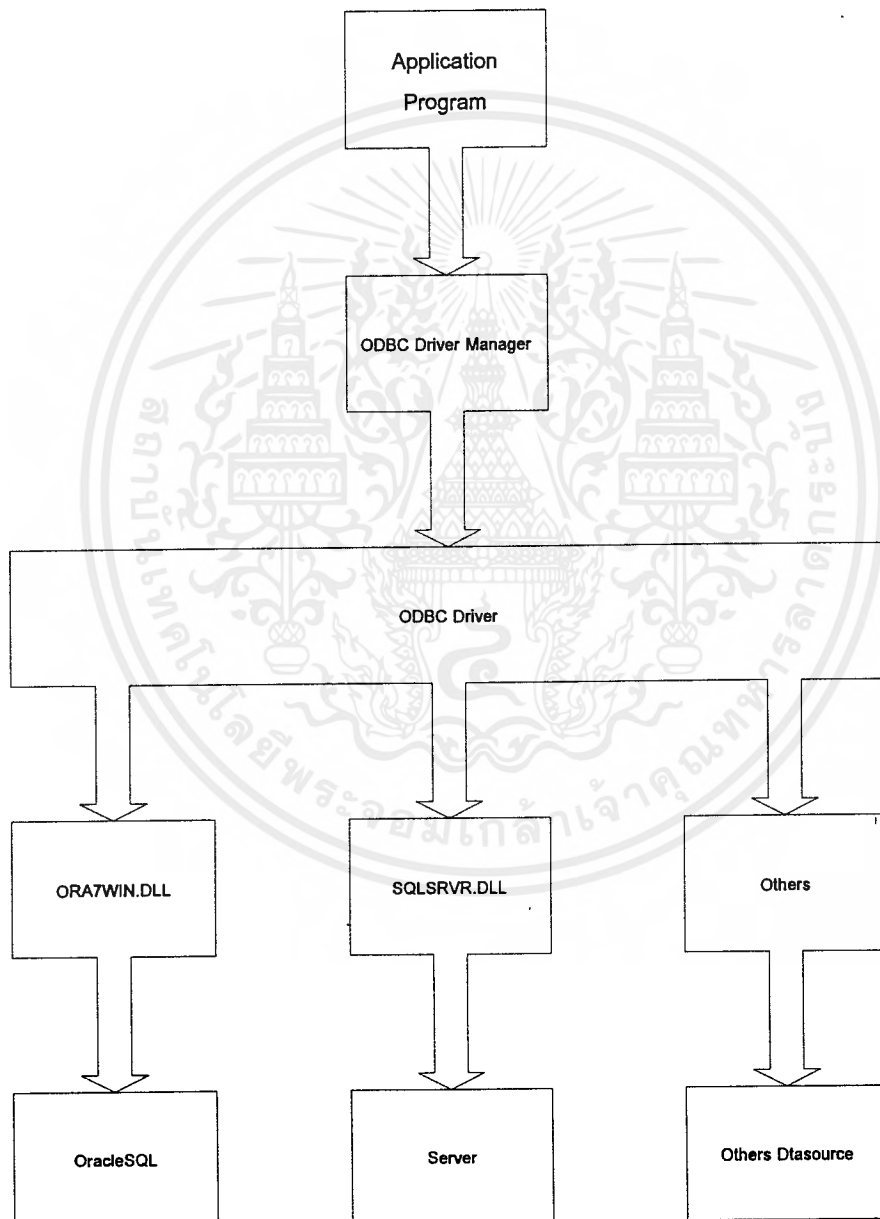
แสดงความจริงแบบ many to one หมายความว่า นักศึกษาหนึ่งคนจะสังกัดภาควิชาได้เพียงภาคเดียว แต่ภาควิชา
ใดๆ สามารถมีนักศึกษาในสังกัดได้มากกว่า 1 คน



แสดงความจริงแบบ many to many หมายความว่านักศึกษาหนึ่งคนสามารถลงทะเบียนเรียนได้หลายวิชา และแต่ละวิชาที่เปิดสอน สามารถรับจำนวนนักศึกษาได้มากกว่าหนึ่งคน แต่นักศึกษาที่ลงทะเบียนวิชาใด ๆ แล้ว จะลงทะเบียนซ้ำวิชาเดิมไม่ได้

สรุปได้ว่าวิธีการออกแบบฐานข้อมูลโดยใช้ NIAM นั้นสามารถที่จะแปลงรูปเป็น Relational Database System ที่อยู่ในรูปของ Fifth Normal Form ได้โดยง่าย เหมาะสมกับการออกแบบฐานข้อมูลของระบบงานขนาดใหญ่

2.6 โอเพินดาต้าเบสคอนเน็คติวิตี (ODBC : Open Database Conectivity)



โครงสร้างแบบลำดับชั้นของ ODBC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การที่จะทำการอินเตอร์เฟสเพื่อที่จะทำให้โปรแกรมประยุกต์สามารถเข้าถึงข้อมูลที่เกี่ยวข้องอยู่ในระบบฐานข้อมูล ซึ่งถูกดูแลโดยระบบจัดการฐานข้อมูล DBMS (Database Management System) ได้นั้นเราจะทำการติดต่อโดยผ่านทาง ODBC ซึ่งก็คือมาตรฐานในการติดต่อนั่นเอง และใช้คำสั่งภาษา SQL เป็นมาตรฐานในการติดต่อ โดยที่ก่อนที่จะมีการพัฒนา ODBC ขึ้นมานั้น การพัฒนาแอปพลิเคชันจะขึ้นกับ DBMS ซึ่งจะมีลักษณะเป็น เอมเบด เอสคิวเอล (Embedded SQL) ประสิทธิภาพของเอ็มเบดเอสคิวเอลนี้ จะขึ้นอยู่กับฮาร์ดแวร์และสภาพแวดล้อมของระบบปฏิบัติการ ซึ่งจะทำให้ซอสโค้ดจะต้องถูกคอมไพล์ (Compile) ใหม่สำหรับแต่ละสภาพแวดล้อม

การที่แอปพลิเคชันใช้ ODBC เป็นตัวติดต่อเพื่อการเข้าถึงข้อมูลนั้นจะทำให้แอปพลิเคชันสามารถเข้าถึงข้อมูลที่ถูกจัดเก็บโดย DBMS ที่แตกต่างกันได้ ซึ่งทำให้ผู้พัฒนาแอปพลิเคชันสามารถนำแอปพลิเคชันที่พัฒนาแล้วไปใช้กับ DBMS ตัวใดก็ได้ โดยที่ผู้ใช้สามารถเพิ่มเติม โมดูล (Database Driver) ที่จะลิงก์แอปพลิเคชันให้ใช้กับ DBMS ที่ต้องการได้

DLL (Dynamically Linked Libraries) เป็น ไดรฟ์เวอร์ที่ทำหน้าที่แปลง ODBC ไปเป็นฟังก์ชันที่คาล์วซอร์สสามารถเรียกใช้งานได้ ซึ่งก็คือการแปลงจาก ODBC SQL Syntax ไปเป็น Syntax ที่ คาล์วซอร์สเข้าใจ

ODBC เป็นมาตรฐานสำหรับการสร้างไดรฟ์เวอร์ของฐานข้อมูล ตามรูปนั้นแอปพลิเคชันโปรแกรมใช้งาน ODBC ผ่านคำสั่งมาตรฐานคือ ODBC API (Open Database Connectivity Application Program Interface) ODBC ไดรฟ์เวอร์จะ ใช้คาล์วซอร์สตามแอปพลิเคชันโปรแกรม โดยใช้คำสั่งของ ODBC และสามารถใช้งานคาล์วซอร์สอื่นๆที่มี ODBC ไดรฟ์เวอร์ ได้โดยไม่ต้องปรับเปลี่ยนแปลงโค้ด

2.6.1 ส่วนประกอบของ ODBC (ODBC Components)

2.6.1.1 แอปพลิเคชัน (Application) คือโปรแกรมที่ทำการใช้งานข้อมูลจากแหล่งของข้อมูล ซึ่งเขียนขึ้นโดยภาษาโปรแกรมมิ่งต่างๆไป มีหน้าที่หลักในการเข้าถึงข้อมูลภายในฐานข้อมูล โดยเรียกใช้ฟังก์ชันของ ODBC ทำการส่งคำสั่ง SQL และการรับข้อมูลผลลัพธ์

2.6.1.2 ไดรฟ์เวอร์ แมเนเจอร์ (Driver Manager) เป็นส่วนที่จัดการเกี่ยวกับการเรียกใช้ .DLL ต่างๆ ตามความต้องการของแอปพลิเคชันโปรแกรม โดยจะทำการโหลดไดรฟ์เวอร์ในนามของแอปพลิเคชัน

2.6.1.3 ไดรฟ์เวอร์ (Driver) ในที่นี้หมายถึง .DLL ต่างๆซึ่งทำหน้าที่รับคำสั่ง SQL จากแอปพลิเคชัน แล้วส่งคำสั่ง SQL ต่อไปยังคาล์วซอร์ส (Data source) ด้วยชุดคำสั่งที่คาล์วซอร์สสามารถเข้าใจได้ แล้วรอรับผลตอบสนอง (Result) เพื่อส่งต่อไปยังแอปพลิเคชัน นอกจากนี้ยังสามารถแก้ไขปรับปรุงรูปแบบการร้องขอจากแอปพลิเคชันเพื่อให้สอดคล้องกับ DBMS

2.6.1.4 คาล์วซอร์ส (Data source) ประกอบด้วยข้อมูลที่ผู้ใช้ต้องการจะเข้าถึง รวมไปถึงระบบปฏิบัติการ (Operating System : OS) ที่เกี่ยวข้อง กับ DBMS และเน็ตเวิร์กแพลตฟอร์ม (Network Platform) (ถ้ามี)

2.6.2 ชนิดของไครฟ์เวอร์

ODBC กำหนดชนิดของไครฟ์เวอร์ไว้ 2 รูปแบบคือ

2.6.2.1 ซิงเกิลเทียร์ (Single-tier) ไครฟ์เวอร์ชนิดนี้จะมีหน้าที่เอ็กซีคิวท์ฟังก์ชันที่ส่งมาจาก ODBC รวมทั้งคำสั่ง SQL ด้วย ดังนั้นไครฟ์เวอร์ชนิดนี้จึงมีหน้าที่บางส่วนของดาต้าซอร์สอยู่ด้วย

2.6.2.2 มัลติเทียร์ (Multiple-tier) ไครฟ์เวอร์ชนิดนี้จะมีหน้าที่เอ็กซีคิวท์เฉพาะฟังก์ชันที่ส่งมาจากโอดีบีซีเท่านั้น ส่วนคำสั่ง SQL จะส่งไปให้กับตัวดาต้าซอร์ส ซึ่งระบบ ๆ หนึ่งอาจจะประกอบด้วยไครฟ์เวอร์ทั้งสองชนิดเลยก็ได้

2.6.3 ข้อดีในการอินเตอร์เฟสกับโอดีบีซี

2.6.3.1 สามารถกำหนดคำสั่ง SQL ได้ทั้งในซอร์สโค้ด (Source Code) ของแอปพลิเคชันนั้น และขณะกำลังทำงาน (Run time)

2.6.3.2 ซอร์สโค้ดเดียวกันสามารถเข้าถึงข้อมูลที่เกี่ยวข้องบน DBMS คนละตัวกันได้

2.6.3.3 โพรโตคอลในการสื่อสารข้อมูล (Data Communication Protocol) ระหว่างแอปพลิเคชันแต่ละตัวไม่มีความสำคัญ

2.6.3.4 สามารถปรับแต่งรูปแบบของข้อมูลที่ได้รับและส่งระหว่างแอปพลิเคชันกับตัวโอดีบีซีได้ตามต้องการ

จากที่ได้กล่าวมาทั้งหมดพอจะสรุป ความสามารถของ ODBC ได้ว่า ODBC นี้จะใช้เพื่อติดต่อไปยังฐานข้อมูลในระบบอื่นๆ จะช่วยให้โปรแกรมของเราเป็นอิสระจากฐานข้อมูลมากขึ้น โดยที่เราสามารถสร้างโปรแกรมขึ้นมาทดสอบกับฐานข้อมูลจำลองในเครื่องของเราเองก่อนได้ จากนั้นเมื่อนำไปใช้จริงกับฐานข้อมูลที่อยู่บนเครื่องเซิร์ฟเวอร์ก็เพียงแต่เปลี่ยนที่คำสั่งโอเพนดาต้าเบส (Opendatabase) เพียงคำสั่งเดียวก็เสร็จ

2.7 สตอร์โปรซีเยอร์ (Stored Procedure)

สตอร์โปรซีเยอร์ คือ กลุ่มของคำสั่งภาษา SQL ที่ถูกแปลไว้ก่อนแล้ว (Pre-compiled) โดยผู้ให้บริการข้อมูล และจะมีแผนการทำงานที่ถูกกำหนดไว้ก่อนล่วงหน้าแล้ว (pre-determine execution plan) สตอร์โปรซีเยอร์จะให้ประโยชน์มากกว่าโค้ดเอสคิวแอลที่ถูกฝังไว้ในบิสซิเนส เซิร์ฟเวอร์ เพราะจะช่วยให้ประสิทธิภาพและความสามารถในการควบคุมฐานข้อมูล มีเพิ่มมากขึ้น

ประโยชน์ของการใช้ สตอร์โปรซีเยอร์

1) เพิ่มประสิทธิภาพของระบบ (performance)

สตอร์โปรซีเยอร์เป็นคำสั่งของ transact SQL ที่ถูกคอมไพล์ในตอนที่ยังเอ็กซีคิวท์ครั้งแรกแล้วเก็บไว้ในฐานข้อมูล มีลักษณะเป็นยูนิท (unit) และสตอร์โปรซีเยอร์จะถูกเอ็กซีคิวท์เมื่อมีการเรียกใช้ (กล่าวคือมี

การเอ็ชคิวครั้งต่อไปจะไม่ต้องถูกคอมไพล์ซ้ำอีก ทำให้รวดเร็วขึ้น ซึ่งต่างจาก SQL Statement ที่ถูกเขียนไว้ในโปรแกรมซึ่งโค้ดเหล่านี้ต้องถูกคอมไพล์ทุกครั้งที่มีการเรียกใช้ โปรแกรม)

2) เพิ่มความถูกต้องของข้อมูล

การใช้สตอร์โพรซีเจอร์ ทำให้การเปลี่ยนแปลงข้อมูลไม่กระทบกระเทือนถึง primary key และการเปลี่ยนแปลงข้อมูลใดๆจะต้องผ่านการตรวจสอบ (Check) จากเงื่อนไขใน สตอร์ โพรซีเจอร์ก่อนว่าถูกต้องตามที่กำหนดไว้หรือไม่

2.7.1 การสร้างสตอร์โพรซีเจอร์ (Create Stored Procedure)

การสร้างสตอร์โพรซีเจอร์สามารถทำได้โดยใช้คำสั่ง CREATE PROCEDURE ซึ่งมีรูปแบบดังนี้

```
CREATE PROCEDURE [owner,] procedure_name [;number]
[@parameter_name datatype [=default] [OUTput]
...
[@parameter_name datatype [=default] [OUTput]
[FOR REPLICATION] | [WITH RECOMPILE] , ENCRYPTION
AS sql_statements
```

2.7.2 การลบสตอร์โพรซีเจอร์ (Dropping Stored Procedure)

การลบสตอร์โพรซีเจอร์สามารถทำได้โดยใช้คำสั่ง DROP PROCEDURE ซึ่งถ้ามี procedure จำนวนหลายๆ procedure ก็สามารที่จะ DROP ได้โดยใช้คำสั่งเดียว ซึ่งมีรูปแบบดังนี้

```
DROP PROCEDURE procedure_name_1,.....,procedure_name_n
```

2.7.3 ตัวอย่างของการเขียนสตอร์โพรซีเจอร์ (ทั้งการสร้างและการลบ)

ตัวอย่างที่ 1 : การสร้างสตอร์โพรซีเจอร์

```
CREATE PROCEDURE Test_Procedure
AS
select * from student
```

เมื่อนำไปทำการสร้างเสร็จแล้วก็สามารถลองเอ็ชคิว ได้โดยใช้ EXEC[UTE] procedure_name ซึ่งจากตัวอย่างที่1 ถ้าจะเอ็ชคิวก็ใช้คำสั่งว่า EXEC Test_Procedure ผลลัพธ์ที่ได้จะเหมือนกับการ select ตาราง student ทุกประการ

ตัวอย่างที่ 2 : การสร้างสตอร์โพรซีเจอร์แบบมีพารามิเตอร์

```
CREATE PROCEDURE Get_Student ( @std_id )
AS
select std_id ,std_name ,std_dept ,std_faculty ,yearstudy
from student
where std_id = @std_id
```

สังเกตเห็นได้ว่าพารามิเตอร์นั้นหน้าตัวแปรจะต้องมีเครื่องหมาย @ ด้วย และสามารถที่จะมีพารามิเตอร์ได้มากกว่า 1 ตัวได้

ตัวอย่างที่ 3 : การสร้างสตอร์โพรซีเจอร์โดยมีการใช้คำสั่งเงื่อนไขต่างๆ อยู่ใน

```
Create Procedure sp_prerequisite(@stdid char(8),@subj char(8))
as
begin
    declare @presubj_id char(8)
    declare @grade char(1)

    if exists( select presubj_id from prerequisite where subj_id = @subj)
    begin
        select @presubj_id = presubj_id from prerequisite where subj_id = @subj

        select @grade = t1.regis_subj_grade
        from register t1, registercard t2
        where t1.regis_no = t2.regis_no and
            t1.regis_subj_id = @presubj_id and
            t2.regis_std_id = @stdid
    end
end
```

การมีคำสั่งเงื่อนไขต่างๆร่วมกับภาษา SQL นั้นจะช่วยเพิ่มประสิทธิภาพของภาษา SQL ได้อย่างมาก ซึ่งสตอร์โพรซีเจอร์ได้เตรียมความสามารถเชิงภาษาโปรแกรมมิ่งที่สามารถทำงานในลักษณะ Flow Control อย่างเช่น ภาษา C ,Pascal ทำให้การ retrieval ,update ,delete ,addition และ manipulation ตารางในฐานข้อมูลเป็นไปได้ง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 4 : การลบสตอร์โพซีเซอร์

DROP PROCEDURE Get_Student

เมื่อทำการ execute คำสั่งตามนี้โพซีเซอร์ Get_Student จะถูกลบออกจากรานข้อมูล



บทที่ 3

Internet Information Server

3.1 Internet Information Server

Microsoft Internet Information Server เป็น Web server ที่ทำให้บริการข้อมูลบนอินเทอร์เน็ต หรือบน อินเทอร์เน็ตได้ Internet Information Server จะส่ง Information ต่าง ๆ โดยใช้โปรโตคอล Hypertext Transfer Protocol (HTTP) นอกจากนี้ Internet Information Server ยังสามารถที่จะกำหนดให้มีบริการเป็น File Transfer Protocol (FTP) และ gopher services ได้

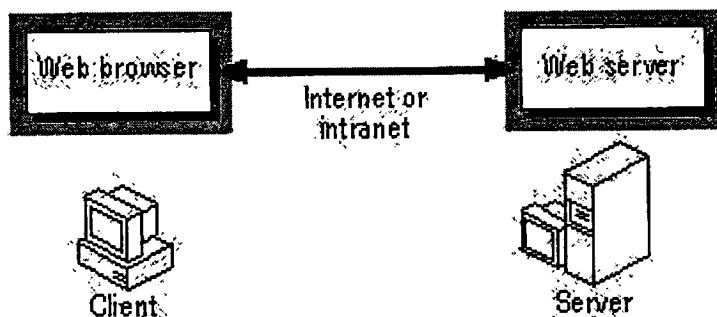
3.2 การนำ Internet Information Server มาใช้งาน

Internet Information Server สามารถนำมาใช้ในงานต่าง ๆ ได้ตามความต้องการ ซึ่งอาจจะเป็นดังงานดังต่อไปนี้

- ให้บริการ Home Page บนอินเทอร์เน็ตสำหรับบริษัท โดยมีลักษณะเป็นจดหมายข่าว, ข้อมูลการขาย หรือ การรับสมัครพนักงาน
- ให้บริการข้อมูลเกี่ยวกับสินค้า และรับการสั่งซื้อสินค้าจากลูกค้า
- ให้บริการที่เป็นโปรแกรมโต้ตอบ (Interactive Program)
- ให้บริการแก่หน่วยงานที่อยู่ห่างไกลสามารถเข้าถึงฐานข้อมูลการขายของบริษัทได้อย่างง่ายดาย
- ใช้สำหรับการดำเนินการและการติดตามฐานข้อมูล

3.3 Internet Information Server ทำงานอย่างไร

Web เป็นระบบพื้นฐานหนึ่งของการร้องขอ และการตอบสนอง Web browsers จะร้องขอ information โดยการส่ง URL ไปยัง Web server เมื่อ Web server ได้รับการร้องขอแล้วก็จะตอบสนองโดยการส่ง Hypertext Markup Language (HTML) page กลับไปยัง Web browsers



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัว HTML Page นั้นสามารถเป็นได้ทั้ง Static Page และ Dynamic Page

- Static Page คือรูปแบบคงที่ของไฟล์ HTML ที่ทำการเก็บไว้ที่ Web stie ที่พร้อมใช้งาน
- Dynamic Page คือรูปแบบไม่คงที่ของไฟล์ HTML ที่ server ต้องสร้างขึ้นด้วย information ที่ จะทำการตอบสนองแก่ผู้ใช้งาน หรืออาจจะเป็นรายการไฟล์ที่มีอยู่หลาย ๆ ไฟล์สำหรับการ เลือกลงใช้ และ folders บน Web site

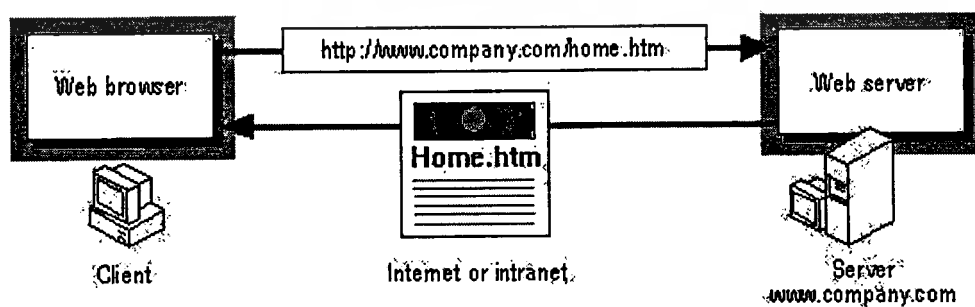
3.3.1 การร้องขอด้วย URL ของ Web Browsers

ทุก page บนอินเทอร์เน็ต หรือบนอินทอร์เน็ตจะมีรูปแบบการระบุของ URL แบบเดียวกัน เมื่อ Web Browsers จะร้องขอ page จาก Web server จะทำการส่ง URL ไปยัง Web Server หลังจากนั้น Server จะใช้ information ใน URL นั้นมากำหนดและแสดงผล Page นั้น

3.3.2 การตอบสนองของ Web server

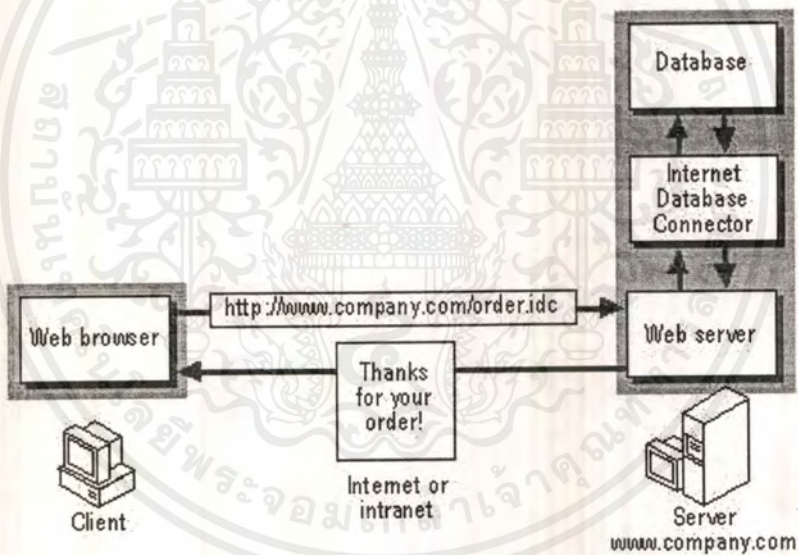
Web server จะตอบสนองสู่ Web browser ที่ร้องขอด้วยการส่ง HTML page กลับไป ส่วน page ที่ จะทำการส่งกลับไปที่ จะประกอบด้วย 3 ชนิดคือ static HTML page, dynamic HTML page หรือ directory-listing page

- Static Pages เป็น page ที่มีลักษณะของ HTML คงที่ที่เตรียมไว้สำหรับการร้องขอ ซึ่ง Web server จะส่ง HTML page นี้กลับไปยังผู้ใช้งาน แต่ page ประเภทนี้จะไม่มีการทำงานแบบ พิเศษ กล่าวคือผู้ใช้งานแค่ทำการร้องขอ Static Page ได้โดยการพิมพ์ URL (เช่น <http://www.ce.kmitl.ac.th/std3p.html>) หรือโดยคลิกที่ลิงค์พอยต์ ตัว URL ที่ทำการร้องขอ ไปจะถูกส่งไปยัง Server และ Server ก็ตอบสนองโดยการส่ง Static HTML pages กลับมา



- **Dynamic Pages** เป็น page ที่ถูกเขียนขึ้นให้มีการตอบสนองกับการร้องขอของผู้ใช้งาน กล่าวคือ Web browsers จะรวบรวม information ที่แสดงเป็นฟอร์มบน page ไม่ว่าจะเป็น text boxes, menus, และ check boxes ซึ่ง information เหล่านี้ผู้ใช้งานจะเป็นผู้ทำการเลือก หรือใส่ลงใน page นั้น เมื่อผู้ใช้งานคลิกที่ปุ่มบนฟอร์มเพื่อทำการส่ง จากนั้นข้อมูลจากฟอร์มทั้งหมดจะถูกส่งไปยัง Web server ซึ่ง server ก็จะนำข้อมูลเหล่านั้นส่งเข้าสู่ script หรือ application เพื่อทำการประมวลผล หรือทำการ queries หรือส่งข้อมูลเข้าสู่ฐานข้อมูล เมื่อกระบวนการเสร็จเรียบร้อย server จะส่งผลการทำงานกลับสู่ผู้ใช้งานในรูปแบบ HTML page

รูปต่อไปนี้แสดงถึงการที่ผู้ใช้งานส่งใบสั่งซื้อไปยังฐานข้อมูลโดยใช้ Internet Database connector คือเมื่อผู้ใช้งานมีแบบฟอร์มที่ใส่กรอกข้อมูลแล้วและคลิกที่ปุ่มส่ง ข้อมูลในฟอร์มทั้งหมดจะถูกส่งไปยัง server จากนั้น server จะนำข้อมูลลงฐานข้อมูล แล้วทำการขึ้นชั้นรับรับสั่งซื้อโดยส่ง HTML page กลับไปยังผู้ใช้งานที่ส่งใบสั่งซื้อมา



- **Directory Listing** ถ้าผู้ใช้งานจะส่งคำถาม (query) โดยไม่มีการเจาะจงไฟล์ใด ๆ ก็สามารถทำได้โดยการเขียน default document สำหรับ Web site หรือ สำหรับ directory ที่เจาะจง หรือสามารถที่จะกำหนดให้มีการ browsing เฉพาะ directory ใด ๆ บน server นั้นได้ ถ้าไม่มีการสร้าง default document สำหรับ directory และ directory browsing แล้ว directory listing (hypertext ของ Windows Explorer หรือ File manager listing) จะถูกส่งกลับไปยังผู้ใช้งานในรูปแบบของ HTML page ซึ่งผู้ใช้งานสามารถเข้าไปยัง directory listing นั้นได้โดยการคลิกเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



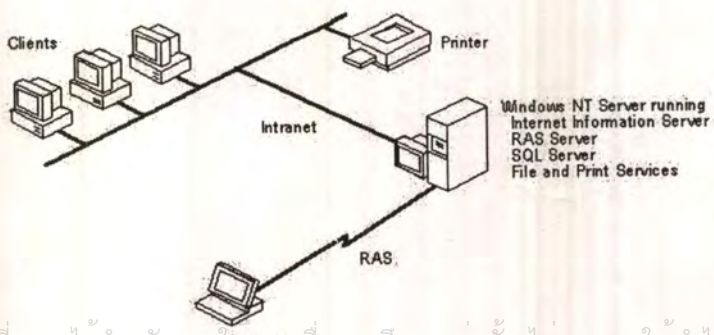
3.4 นำ Internet Information Server มาใช้งานได้อย่างไร

Internet Information Server มีความอ่อนตัวมากเหมาะกับหน้าที่การทำงานต่าง ๆ ขององค์กร มันสามารถที่ติดตั้งเพื่อขยาย (scalable) จากระบบ server สนับสนุน site เดียว ไปเป็น server สนับสนุนหลาย site ได้ ตัวอย่างเช่น www.microsoft.com และ www.msn.com ซึ่งเป็นอีกกลุ่มหนึ่งในหลาย ๆ Web site บนระบบอินเทอร์เน็ตทุกวันนี้ และทั้งคู่ใช้ เป็น multiple servers โดยใช้ Microsoft Internet Information Server และการเชื่อมต่อคิงที่กล่าวต่อไปนี้มีจุดมุ่งหมายให้เพื่อความเข้าใจในขอบเขตของการใช้ Internet Information Server ที่สามารถเป็นไปได้

3.4.1 Intranet Scenarios

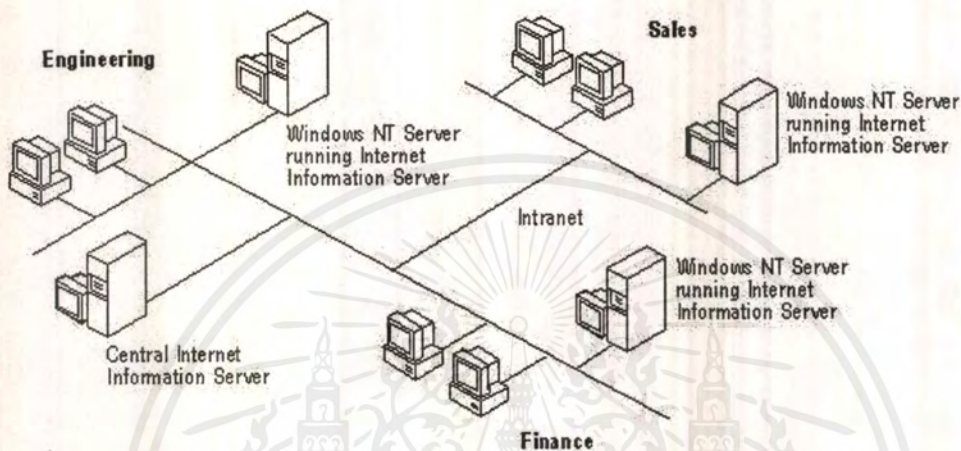
Internet Information Server จะทำการรวมสถานะแวดล้อมที่มีเข้าเป็นระบบเดียว เนื่องจากว่า Internet Information Server มีการรวมระบบความปลอดภัย (security) ของ Windows NT และระบบเครือข่าย (Networking) ไว้ด้วยกัน จึงสามารถที่จะเพิ่ม Software เข้าสู่คอมพิวเตอร์ที่มีอยู่ และหมายเลขบัญชีของผู้ใช้งาน (user accounts) ได้ คือไม่จำเป็นต้องกำหนดว่าต้องนำคอมพิวเตอร์เครื่องใดมาใช้เป็น Internet Information Server

ตัวอย่างการใช้งานในระดับกลุ่มงานเล็ก ๆ สามารถที่จะเพิ่ม Internet Information Server เข้าสู่ File server และ Print server ที่มีอยู่ได้ กลุ่มงานของ Web server สามารถที่จะทำ Web-style pages ส่วนบุคคลได้ สามารถกำหนดกลุ่มงานที่มีงานเหมือนกัน สามารถที่จะให้การเชื่อมต่อเข้าสู่กลุ่มงานของระบบฐานข้อมูล Structured Query Language (SQL) หรือนำมาใช้เป็น Remote Access Service (RAS) เพื่อทำการ dial-up เข้ามาใช้งานทรัพยากรต่าง ๆ ในกลุ่มงานจาก site ที่ห่างไกลได้



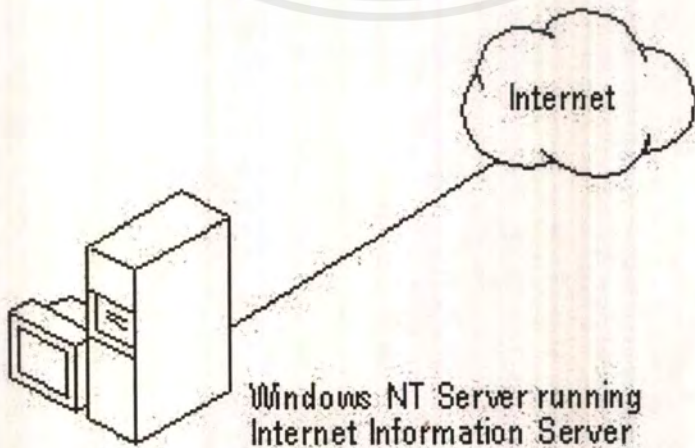
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระบบธุรกิจขนาดใหญ่ที่มีหลาย ๆ แผนกหรือหลาย ๆ กลุ่มงาน ซึ่งแต่ละแผนกต้องมีการใช้งาน Internet Information Server บนแฟ้มข้อมูลที่มีอยู่บนไฟล์ server ของกลุ่มงานเฉพาะนั้น ดังนั้น server ที่เป็นศูนย์กลางข้อมูลอาจจะต้องใช้เป็นข้อมูลเครือข่ายของบริษัท เช่น ใช้เป็นคู่มือการทำงานของพนักงาน หรือเป็น directory ของบริษัท



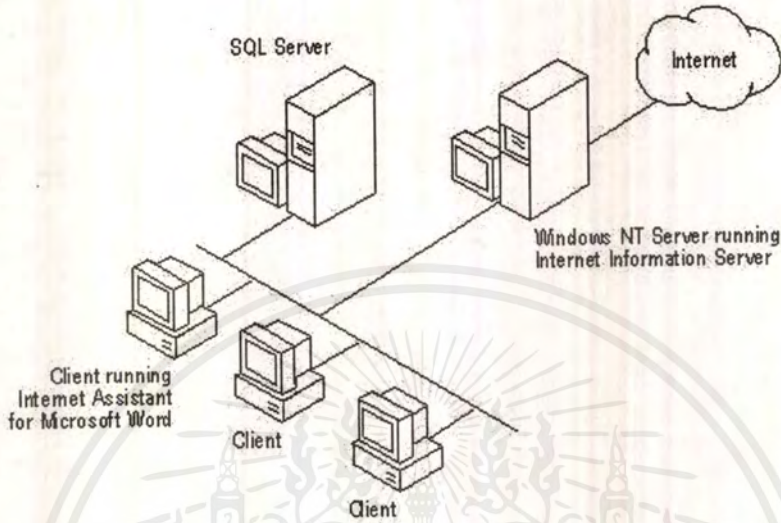
3.4.2 Internet Scenarios

Internet Information Server สามารถทำเป็น Web server ง่ายๆ บนอินเทอร์เน็ตได้ ดังรูป

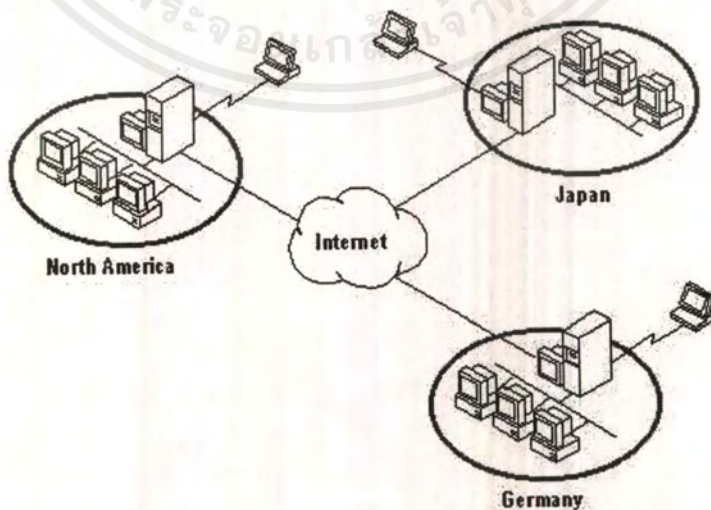


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน site ขนาดใหญ่สามารถที่จะเข้าใช้งาน Internet Information Server โดยผ่านเครือข่ายภายใน ให้พนักงานสามารถเข้าค้นหาข้อมูลใน server หรือเพื่อใช้โปรแกรมเครื่องมือต่าง ๆ ได้ เช่น ใช้งาน Microsoft FrontPage เพื่อเขียนข้อมูลภายใน server เอง



Internet Information Server เป็นส่วนหนึ่งของบริการ Windows NT ที่สามารถที่จะสร้าง server ให้มีหลาย ๆ หน้าที่การทำงาน ตัวอย่างเช่น บริษัทหนึ่ง ๆ ในโลกต่างก็มี site แยกต่างหาก ซึ่งสามารถใช้ Internet Information Server เพื่อทำการสื่อสารระหว่าง site ด้วยกันได้ และด้วยความอ่อนตัวนี้เองของการเข้าใช้ในระบบอินเทอร์เน็ต จึงสามารถเพิ่ม RAS เข้าสู่ Internet Information Server เพื่อทำการ dial-up access เข้าสู่แต่ละระบบบริหารเน็ต หรือระบบอินเทอร์เน็ต ได้อย่างไม่ยากนัก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 คุณลักษณะของ Internet Database Connector

Internet Database Connector มีอยู่หลายคุณลักษณะที่ช่วยสำหรับการเขียนเว็บเพจ ที่ประกอบด้วยข้อมูลที่มาจากระบบฐานข้อมูล

3.5.1 ไฟล์ Internet Database Connector

ไฟล์ Internet Database Connector ประกอบไปด้วยข้อมูลต่าง ๆ ที่ใช้สำหรับเข้าถึงฐานข้อมูล ซึ่งมีคุณลักษณะต่อไปนี้

3.5.1.1 Parameters

พารามิเตอร์ คือชื่อและค่า ของฟอร์มใน HTML เช่น <INPUT ... > ซึ่งชื่อก็สามารถกำหนดได้โดยตรงใน URLs โดยชื่อและค่าเหล่านี้จะถูกส่งไปโดย Web Browser ที่จะนำไปใช้ใน SQL Statement บน Server เช่น

```
<FORM METHOD="post" ACTION="/script/Register/Register.Internet">
<INPUT NAME="id" VALUE="38013289">
</FORM>
```

ในไฟล์ Internet Database Connector จะนำพารามิเตอร์ที่ชื่อ "id" นี้มาใช้งานดังนี้

SQLStatement:

```
+ Select fname, lname
+ From student
+ where std_id = %id%
```

สิ่งสำคัญคือชื่อของพารามิเตอร์ใน Internet Database Connector ต้องตรงกันกับในฟอร์มของ HTML และบนเว็บเพจ พารามิเตอร์ ต้องปิดด้วยสัญลักษณ์เปอร์เซ็นต์ (%) เพื่อแยกความแตกต่างจากตัวแปรปกติใน SQL หลังจากที่ Internet Database Connector นำพารามิเตอร์เข้ามาในไฟล์ .idc แล้ว มันก็จะนำค่าที่ถูกส่งมาจากเว็บเบราว์เซอร์เข้ามาแทนใน SQL Statement และส่งให้กับ ODBC Driver ต่อไป

3.5.1.2 ฟิลด์ในไฟล์ Internet Database Connector (.idc)

ตารางต่อไปนี้เป็นฟิลด์ที่สามารถกำหนดได้ในไฟล์ Internet Database Connector

หมายเหตุ : พารามิเตอร์หรือตัวแปรของ Server อาจมีการอ้างอิงได้ในทุก ๆ ที่ในไฟล์

.idc

Field	Description
Datasource	เป็นชื่อที่ตรงกันกับ Datasource ใน ODBC ของระบบ (DSN)
Template	เป็นชื่อของไฟล์ HTML Extention ที่มีรูปแบบเพื่อการรับข้อมูลที่ได้จากการ Query ซึ่งไฟล์นี้มีสกุลว่า .htx
SQL Statement	เป็นคำสั่ง SQL ที่ใช้ทำการประมวลผล ที่ประกอบไปด้วยพารามิเตอร์ และค่าต่างๆ SQL Statement นี้สามารถมีได้หลาย ๆ ชุดในไฟล์ Internet Database Connector เดียวกัน สำหรับบรรทัดที่อยู่ถัดไปจากบรรทัดของ SQL Statement จะต้องขึ้นต้นด้วยเครื่องหมายบวก (+)

ตารางแสดงฟิลด์ที่ต้องการในไฟล์ Internet Database Connector (.idc)

Field	Description
DefaultParameters = <i>param</i> = <i>value</i> [, <i>param</i> = <i>value</i>] [..]	พารามิเตอร์ต้องมีค่าเท่านั้นจึงจะใช้ได้ในไฟล์ Internet Database Connector
Expires	จำนวนวินาทีที่จะรอก่อนที่จะมีการรีเฟรชแคช Output Page นั้นใหม่ ฟิลด์นี้มีประโยชน์มากเมื่อต้องการให้แสดงผลการ Query ฐานข้อมูลใหม่ หลังจากหมดเวลาที่ตั้งไว้ โดยปกติ IDC จะไม่มีแคช Output Page มันจะมีก็ต่อเมื่อมีการฟิลด์ Expires เท่านั้น
MaxFieldSize	เป็นจำนวนสูงสุดของบัพเฟอร์ที่จองไว้โดย Internet Database Connector สำหรับแต่ละฟิลด์ ถ้ามีอักขรตัวใดหลุดออกไปจากบัพเฟอร์นี้ก็จะถือว่าปล่อยให้หลุดไป พารามิเตอร์นี้จะใช้กับเฉพาะฟิลด์ที่มีการ Return ค่าจากฐานข้อมูลมากกว่า 8192 Byte เพราะค่า Default ของบัพเฟอร์นั้นอยู่ที่ 8192
ODBCConnection	เพิ่มฟิลด์นี้ด้วยค่า <i>pool</i> ก็จะเป็นการเชื่อมต่อกับ <i>pool</i> ซึ่งเก็บการเชื่อมต่อเข้ากับฐานข้อมูลที่เปิดไว้เพื่อในครั้งต่อ ๆ ไป โดย IDC จะส่งข้อมูลผ่านทาง การเชื่อมต่อ <i>pool</i> เพื่อการลำดับการประมวลผลครั้งต่อ

	ไป ของไฟล์ .Internet Database Connector ที่มีค่าของ Datasource, Username และ Password เหมือนกัน การใช้งานอ็อปชันนี้เป็นการเพิ่มประสิทธิภาพของ Internet Database Connector ถ้าเป็นอ็อปชัน <i>nonpool</i> ก็จะไม่มีการเชื่อมต่อเข้ากับ pool หากให้ค่าเป็น <i>nopool</i> ก็จะเป็นการจัดการแเคชของการเชื่อมต่อแบบโดยตรง ดังนั้นถ้าจะมีการจำกัดจำนวนการเชื่อมต่อ ก็ไม่ต้องทำการเชื่อมต่อทั้งหมดเข้ากับ pool มิฉะนั้นจะไม่มีผู้ใดสามารถทำการเชื่อมต่อเข้ากับ SQL Server ได้
Password	เป็น Password ที่ใช้สำหรับ User Name นั้น ถ้า Password มีค่าเป็น Null ก็ไม่จำเป็นที่ต้องใส่ไว้
RequiredParameters	เป็นชื่อของพารามิเตอร์ ซึ่งในส่วนของ Httpodbc.dll จะรับรู้ว่ามีมาจาก Client ถ้าไม่มันจะทำการ Return ค่า Error ออกมา และ ชื่อพารามิเตอร์นี้สามารถแยกได้โดยเครื่องหมายลูกน้ำ (,)
Translationfile	เป็นที่ที่ไฟล์ที่จะทำการ Map เป็นอักษรที่ไม่ใช่ภาษาอังกฤษ (เช่น , & หรือ ก) ดังนั้นบราวเซอร์สามารถแสดงในรูปแบบของ HTML ได้ อย่างเหมาะสม ถ้าไฟล์ที่จะทำการแปลงนั้นไม่ได้อยู่ใน Directory เดียวกันกับไฟล์ .idc ก็ต้องมีการระบุถึงที่อยู่ (Path) ลงไปด้วย และด้วยการแปลงไฟล์นี้เองก็จะสามารถแสดงให้อยู่ในรูปแบบใด ๆ ได้
Username	ชื่อที่ถูกต้องสำหรับการการใช้ Datasource Name ในฟิลด์ Datasource
Content-Type	ที่ถูกต้องของ MIME นั้นจะบอกว่าจะได้รับการ Return อะไรจาก Client ถ้าไฟล์ .htx ประกอบไปด้วย HTML ส่วนใหญ่ จะเป็น “text/html”

ตารางแสดงฟิลด์ที่เป็น Optionn ใน ไฟล์ Internet Database Connector (.idc)

3.5.2 ODBC Advanced Optional Fields

Option ของ ODBC ระดับสูงยอมให้มีการทำ Debug และการปรับแต่งแบบละเอียด ของ ODBC driver ที่ใช้โดย Internet Database Connector ซึ่งมีรูปแบบดังนี้

ODBCOption: Option Name=Value [, option Name=Value...]

ตัวอย่างเช่น

ODBCOption: SQL_QUERY_TIMEOUT=10, SQL_OPT_TRACE=1,
SQL_OPT_TRACEFILE=C:\Sql.log

ตัวอย่างนี้คือการหยุดการทำงานของ SQLStatement เมื่อมีการทำงานมากกว่า 10 วินาทีขึ้นไป และให้มีการติดตาม (Trace) การเรียกใช้ฟังก์ชันของ ODBC แล้วเก็บลงใน Sql.log ด้วย และ Option อื่น ดังตารางต่อไปนี้

Option Name	Value	Purpose
SQL_ACCESS_MODE	0 = Read / Write 1 = Read Only	ตัวบ่งชี้สถานะสำหรับ ODBC Driver หรือ Data Source ที่ทำการเชื่อมต่อด้วยนั้นไม่จำเป็นต้องมีใน SQL Statement ซึ่งก็คือสาเหตุที่ทำให้มีการ Update เกิดขึ้น และ ใน Mode นี้สามารถใช้เป็นกลยุทธในการ Locking, การจัดการ Transaction หรืออื่น ๆ โดย Driver ไม่จำเป็นต้องป้องกันแต่ละ Statement จากการ Submitted ของ Data Source เมื่อถามถึงกระบวนการของ SQL Statement การกระทำของ Driver และ Data Source จะไม่ทำการ Read-only ระหว่างการเชื่อมต่อแบบ Read-only โดยใน SQL_ACCESS_MODE จะให้โหมด 0 เป็น Default ซึ่งสามารถอ่าน และเขียนได้
SQL_LOGIN_TIMEOUT	Integer	เป็นจำนวนของวินาทีที่จะรอดังแต่การร้องขอ Logon จนเสร็จสมบูรณ์ ก่อนที่จะทำการ Disconnect ไป ค่า Default จะขึ้นอยู่กับ Driver และต้องไม่เป็นศูนย์ ถ้าค่าเป็นศูนย์ Timeout จะเป็นการ Disable และการเชื่อมต่อจะรอไปแบบไม่มีกำหนด ถ้าการกำหนด Timeout เกินค่าสูงสุดของค่า Timeout Logon ใน Data Source ค่านั้นก็จะถูกเปลี่ยนให้เป็นค่าสูงสุดแทน
SQL_OPT_TRACE	0 = Trace off 1 = Trace on	เมื่อมีการทำ Tracing แต่ละครั้งของการเรียกใช้ฟังก์ชัน ODBC โดย Httpodbc.dll จะทำการเขียน Trace File ถ้าไฟล์นี้มีอยู่แล้ว ODBC จะทำการ

		เขียนต่อท้ายเข้าไป แต่ถ้าไม่มีจะสร้างขึ้นใหม่ด้วยชื่อที่กำหนดหรือใช้ชื่อว่า Sql.log
SQL_OPT_TRACEFILE	File name	เป็นชื่อของ Trace File ที่ใช้เมื่อมีการให้ค่าของ SQL_OPT_TRACE = 1 ซึ่งชื่อ Default คือ Sql.log
SQL_PACKET_SIZE	Integer	เป็นขนาดของแพ็คเกจในเน็ตเวิร์กหน่วยเป็น Byte ซึ่งใช้สำหรับแลกเปลี่ยนข้อมูลข่าวสารระหว่าง DBMS และ Web Server
SQL_TRANSLATE_DLL	File name	เป็นชื่อของ DLL ที่ประกอบด้วย ฟังก์ชัน SQLDriverToDatasource และ SQLDatasourceToDriver ที่ทำการ load และจัดการตามกระบวนการ เช่นการแปลงชุดอักษรเป็นต้น
SQL_TRANSLATE_OPTION	Integer	เป็นค่าควบคุมฟังก์ชันการแปลง ซึ่งเป็นการกำหนดการทำงานของ Translation DLL
SQL_TXN_ISOLATION	Integer 1 = Read Uncommitted 2 = Read Committed 4 = Repeatable Read 8 = Serializable 16 = Versioning	กำหนดระดับของ Transaction Isolation โดยที่ Internet Database Connector ไม่สนับสนุนการขยายตัวของ Transaction ในไฟล์ .idc ใดๆก็ตาม สำหรับ DBMSs การกำหนดค่า SQL_TXN_ISOLATION ให้เป็น 1 (Read Uncommitted) จะมีผลในการทำงานพร้อมกันกันมาก ๆ ได้ดี อย่งไรก็ดีด้วยการกำหนดค่าแบบนี้ ข้อมูลจะไม่มี Commit ในฐานข้อมูลด้วย โดย Transaction อื่น ๆ อาจทำการดึงออกไปได้
SQL_MAX_LENGTH	Integer	จำนวนสูงสุดของข้อมูลที่ Driver ส่งคืนกลับมาจาก Character หรือ Binary Column โดยอ็อบชั่นนี้จึงใจที่จะลดการจราจรบนเน็ตเวิร์ก และควรใช้ก็ต่อเมื่อ Data Source (ตรงกันข้ามกับ Driver) ใน Multiple-tier Driver สามารถทำได้
SQL_MAX_ROWS	Integer	จำนวนสูงสุดของแถวที่จะส่งคืนมาจากการ SELECT Statement ถ้ามีค่าเท่ากับศูนย์ (Default) แล้ว Driver จะส่งมาทุกแถว ซึ่ง Option นี้จึงใจที่จะลดการจราจรบนเน็ตเวิร์ก เมื่อ Datasource สามารถที่จะกำหนดค่าการส่งกลับของแถวได้ด้วยตัวมันเอง ในทางตรงกันข้าม MaxRecords เป็นตัวแปรภายใน Internet Database Connector ที่ใช้สำหรับกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถเผยแพร่หรือใช้เพื่อวัตถุประสงค์อื่นใดได้โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		ค่าของจำนวนแถวที่ต้องการรับ
SQL_NOSCAN	0 = Scan for and convert escape clauses 1 = Do not scan for and convert escape clauses	กำหนดให้ Driver ไม่ต้อง Scan SQL String สำหรับกรณีที่มีการหลุดลอดออกไป ถ้าค่าเป็น 0 (Default) Driver ก็จะ Scan String สำหรับกรณีหลุดลอด ถ้าค่าเป็น 1 Driver จะไม่มีการ Scan SQL String สำหรับกรณีการหลุดลอด เช่นตัว Driver ส่ง Statement โดยตรงมายัง Data Source ถ้า Statement นี้ไม่ได้ประกอบด้วยกรณีหลุดลอดของ ODBC รูปแบบเฉพาะที่ถูกปิดไว้ด้วยเครื่องหมายปีกกา ({}) แล้ว ก็จะทำให้ค่า Option นี้เป็น 1 จะทำให้ได้ประสิทธิภาพเพิ่มขึ้น โดยเข้าไปโดยตรงไม่ต้องทำการ Scan SQL String ให้เสียเวลา
SQL_QUERY_TIMEOUT	Integer 0 = No timeout	จำนวนของวินาทีที่จะรอสำหรับ SQL Statement ที่ทำการประมวลผลก่อนที่จะทำการยกเลิกการ Query เมื่อ Set เป็น 0 (Default) ก็จะไม่มีการ Timeout ถ้ากำหนดค่า Timeout มากเกินค่าสูงสุดของค่า Timeout ใน Datasource หรือ น้อยกว่าค่าต่ำสุด Driver จะแทนด้วยค่าสูงสุด ต่ำสุดนั้น
Integer	Driver Specific	ค่าอ็อบชั่นใน Driver-Specific สามารถที่จะกำหนดในรูปแบบของ <i>number = value</i> ตัวอย่างเช่น 2499 = 1, 123 = String

ตารางแสดง ODBC Advanced Option Fields

3.5.3 ไฟล์ HTML Extension (.htx)

ไฟล์ HTML Extension ประกอบด้วยจำนวนของ Keyword ที่ควบคุมว่าจะแสดงผลในรูปแบบของโครงสร้างเอกสาร HTML ใดอย่างไร Keyword เหล่านี้แบ่งประเภทออกได้ดังนี้

3.5.3.1 <%begindetail%>, <%enddetail%>

Keyword ที่อยู่ใน <%begindetail%> ... <%enddetail%> ที่อยู่ในส่วนของไฟล์ HTML Extension จะเป็นพื้นที่ที่ซึ่งข้อมูลที่ออกมาจากฐานข้อมูลจะมาทำการซ้อนทับ (Merge) ซึ่งภายใน Section จะไม่มีการจำกัดชื่อคอลัมน์ ด้วย <% and %> หรือ <!--%--> เป็นการกำหนดตำแหน่งของการ Return ของข้อมูล จากการ Query ตัวอย่างเช่น

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<%begindetail%>
<%std_id%> : <%std_name%>
<%enddetail%>
```

หลังจากการซึ่อนแล้วข้อมูลคอลัมน์ `std_id` และ `std_name` จะแสดงออกมา ซึ่งทุก ๆ คอลัมน์สามารถอ้างอิงได้โดยวิธีนี้ และชื่อคอลัมน์นี้สามารถอ้างอิงได้ในทุก ๆ ที่ของไฟล์ HTML Extension

หมายเหตุ : ถ้าไม่มีเรคคอร์ด Return มา จากการ Query ในส่วนของ `<%begindetail%>` จะถูกข้ามไป สำหรับแต่ละ SQL Statement นั้น จะให้ Result Set (เช่น SELECT) ควรมีการกำหนดให้มีความสอดคล้องตรงกันกับในส่วนของ `<%begindetail%>` `<%enddetail%>` ในไฟล์ .htx ด้วย

3.5.3.2 `<%if%>`, `<%else%>`, `<%endif%>`

ไฟล์ HTML Extension สามารถประกอบด้วยเงื่อนไขที่เป็น Statement If - Then - Else เพื่อควบคุมให้อยู่ในโครงสร้างของเว็บเพจ โดยมี Syntax ทั่วไปดังนี้

```
<%if condition %>
HTML text
[ <%else%> HTML text ]
<%endif%>
```

โดยที่ *condition* จะอยู่ในรูปแบบดังนี้

value1 operator value2

และ *operator* สามารถเป็นหนึ่งในรูปแบบดังต่อไปนี้

EQ	if <i>value1</i> equals <i>value2</i>
LT	if <i>value1</i> is less than <i>value2</i>
GT	if <i>value1</i> is greater than <i>value2</i>
CONTAINS	if any part of <i>value1</i> contains the string <i>value2</i>

โอเปอเรนด์ *value1* และ *value2* สามารถเป็นชื่อ Column, หนึ่งในตัวแปรภายใน (Built-in Variable เช่น CurrentRecord or maxRedcords), ชื่อตัวแปรของ HTTP หรือ ค่าคงที่ เมื่อใช้ใน Statement `<%if%>` ค่าที่ไม่จำกัดก็จะเป็น `<% and %>`

3.5.4 CurrentRecord, MaxRecords

CurrentRecord เป็นตัวแปรภายใน (Built-in Variable) ที่ประกอบด้วยที่ประกอบด้วยจำนวนครั้งของการทำงานในส่วนของ `<%begindetail%>` โดยครั้งแรกที่ผ่านเข้ามายัง `<%begindetail%>` ค่าจะเป็นศูนย์ ถ้าดับต่อมา ค่าของ CurrentRecord จะเปลี่ยนไปทุก ๆ ครั้ง ที่เรคคอร์ดถูกดึงมาจากฐานข้อมูล

MaxRecord เป็นตัวแปรภายใน เช่นเดียวกันกับ CurrentRecord จะประกอบด้วยค่าของ MaxRecord พิลด์ ในไฟล์ Internet Database Connector ทั้งใน MaxRecord และ CurrentRecord สามารถใช้งานได้ใน Statement `<%if%>` เท่านั้น

3.5.5 พารามิเตอร์จากไฟล์ Internet Database Connector

พารามิเตอร์จากไฟล์ Internet Database Connector สามารถที่จะใช้อ้างอิงได้ในไฟล์ HTML Extension โดยต้องมีการกำหนดไว้หน้าชื่อของพารามิเตอร์ด้วย "idc" และตามด้วยจุด (.) เช่น

Student ID is : `<%idc.std_id%>`

3.5.6 ตัวแปรของ HTTP

หลาย ๆ ตัวแปรภายในไฟล์ HTML Extension สามารถให้ข้อมูลข่าวสารจำนวนมากเกี่ยวกับสภาพแวดล้อม และการเชื่อมต่อของ Web Client เข้าสู่ Server ข้อมูลเหล่านี้จะมาจากส่วนหัว (Header) ที่ถูกส่งมาโดย Client ในการที่จะเข้าถึงมันเหล่านั้น โดยใช้ Internet Database Connector ต้องทำการปรับเปลี่ยนสิ่งต่าง ๆ ดังต่อไปนี้

1. เพิ่ม HTTP_ เข้าไปเพื่อการเริ่มต้น
2. ปรับเปลี่ยนเส้นขีดยาว (Dash) ไปเป็นขีดเส้นใต้ (Underscores)
3. ปรับเปลี่ยนอักษรทั้งหมดให้เป็นตัวอักษรตัวใหญ่

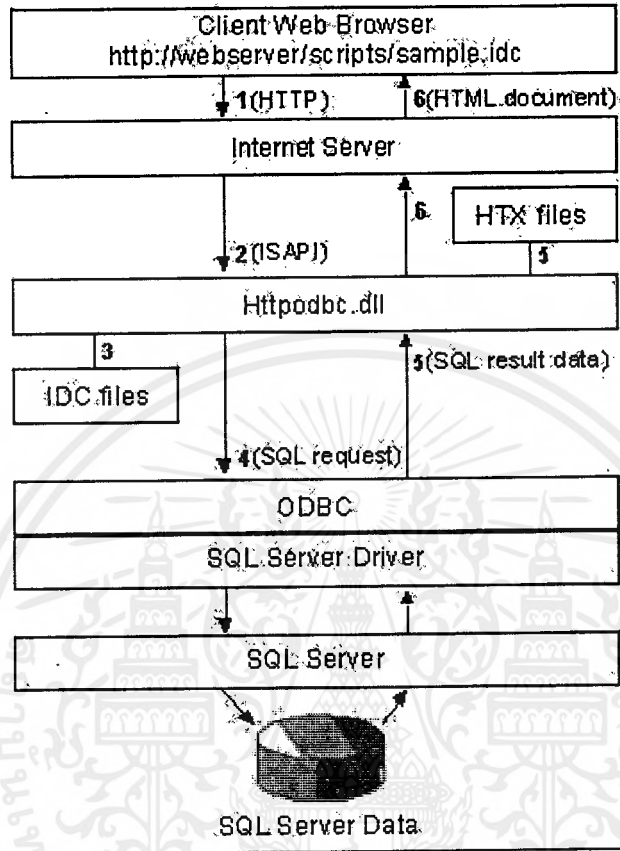
ตารางต่อไปนี้เป็นชุดของตัวแปรปริยาย (Default) ซึ่งตัวแปร Environment เหล่านี้เป็นตัวแปรสำหรับ CGI Application และตัวแปร HTTP สำหรับ IDC Application

Variable	Meaning
ALL_HTTP	<p>ส่วนหัวของ HTTP ไม่ได้เป็นตัวแปรตามตารางนี้ และตัวแปรนี้จะอยู่ในรูปของ HTTP_<the fieldname > ตัวอย่างเช่น</p> <p>HTTP_ACCEPT: */*, q=0.300, audio/x-aiff, audio/basic, image/jpeg, image/gif, text/plain, text/html HTTP_USER_AGENT: Microsoft Internet Explorer/0.1 (Win32) HTTP_REFERER: http://webserver/samples/dbsamp/dbsamp3.htm HTTP_CONTENT_TYPE: application/x-www-form-urlencoded HTTP_CONTENT_LENGTH: 10</p>
AUTH_TYPE	ชนิดของ authorization ที่ใช้ ถ้า User name เป็นเจ้าหน้าที่ผู้เขียนและจัดการโดย Server ถ้าไม่มีก็จะเป็น Basic
CONTENT_LENGTH	จำนวนไบต์ที่สคริปต์ที่คาดว่าจะได้รับจาก Client
CONTENT_TYPE	Content Type ของข้อมูลข่าวสารจะส่งให้ภายใน Body ของ Post Request
GATEWAY_INTERFACE	การเข้าใช้ของ CGI (Common Gateway Interface) กำหนดได้จากการคอมไพล์ (Compile) ของ Server
HTTP_ACCEPT	<p>เป็นส่วนหัวของ HTTP ที่เป็นกรณีพิเศษ คือมีค่าของการยอมรับ จะเป็นฟิลต์ที่นำมาต่อไว้ โดยที่จะแยกจากกันโดยเครื่องหมายลูกน้ำ (,) ตัวอย่างเช่น ถ้ามีส่วนหัวดังนี้</p> <p>accept: */*; q=0.1 accept: text/html accept: image/jpeg</p> <p>จากนั้นตัวแปร HTTP_ACCEPT จะมีค่าเป็น :</p> <p>*/*; q=0.1, text/html, image/jpeg</p>
LOGON_USER	เป็นรายชื่อบัญชีผู้ใช้งาน (Account) ของ Window NT
PATH_INFO	เป็นเส้นทางของข้อมูลข่าวสารที่ได้รับมาจาก Client ที่ประกอบไปด้วย ส่วนของเส้นทางของ URL หลังจาก Script name แต่จะก่อนการ Query สตริง (ถ้ามี)
PATH_TRANSLATED	ค่าของ PATH_INFO แต่พร้อมด้วยทุก ๆ ชื่อ Virtual Path ที่ขยายเข้ามาตามข้อกำหนดของ Directory
QUERY_STRING	ข้อมูลข่าวสารนั้นจะตามด้วยเครื่องหมายคำถาม (?) ใน URL ที่เข้าไปอ้างอิงกับสคริปต์
REMOTE_ADDR	เป็น IP address ของ Client

REMOTE_HOST	ชื่อ Host สำหรับ Client
REMOTE_USER	User name ที่ให้โดย Client และได้รับการรองรับว่าเข้าใช้งานได้โดย Server
REQUEST_METHOD	วิธีการที่ HTTP ร้องขอ
SCRIPT_NAME	ชื่อของโปรแกรมสคริปต์ที่ทำการประมวลผล
SERVER_NAME	ชื่อ Host ของ Server (หรือ IP address) ที่ทำการอ้างอิงใน URL
SERVER_PORT	TCP/IP Port ที่ซึ่งต้องการร้องขอการรองรับ
SERVER_PORT_SECURE	มีค่าเป็น 0 หรือ 1 โดยค่า 1 จะระบุถึงการร้องขอใน Port เข้ารหัส
SERVER_PROTOCOL	เป็นชื่อและรุ่นของโปรโตคอลที่ทำการดึงข้อมูลข่าวสาร ที่เกี่ยวข้องกับการร้องขอ ซึ่งปกติจะเป็น HTTP/1.0
SERVER_SOFTWARE	ชื่อและรุ่นของ Web Server ภายใต้การทำงานของ Internet Sever Extension
URL	เป็น URL ที่ทำการร้องขอ

ตารางแสดงตัวแปรของ Internet Information Server

3.6 การทำงานของระบบ



รูปแสดงขั้นตอนการทำงานของระบบ

การทำงานมีขั้นตอนต่าง ๆ ดังนี้

1. Internet Information Server รับ URL

- URL จะถูกส่งมาจาก Web browsers

2. Internet Information Server จะ load ไฟล์ Httpodbc.dll ขึ้นมาและให้ค่า Information ที่มาพร้อมกับ URL

- ไฟล์ .idc จะถูกเชื่อม (Map) เข้ากับ Httpodbc.dll และ Httpodbc.dll ก็จะทำการ load ไฟล์ Internet Database Connector และ Item อื่นด้วย Information ที่มากับ URL ผ่านทาง Internet Information Server

3. Httpodbc.dll จะทำการอ่าน ไฟล์ Internet Database Connector

- ไฟล์ Internet Database connector จะประกอบด้วยหลาย ๆ Entries ที่อยู่ในรูปแบบ

field : value

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง Entries ที่บรรจุในไฟล์ Sample.idc ที่อ้างถึงใน URL ข้างต้น

Datasource : Web SQL

Username : sa

Template : sample.htx

SQLStatement :

```
+SELECT au_lname, ytd_sales
+      form pubs.dbo.titleview
+      where ytd_sales > 5000
```

ในตัวอย่างไฟล์ .idc มี

- **DataSource** คือ Web SQL ซึ่งจะเป็นการบอก ODBC ว่าให้เรียกใช้ DataSource ที่ชื่อ Web SQL
- **Username** จะต้องเป็นชื่อที่สามารถ logon มาที่ ODBC DataSource ได้ ซึ่งในตัวอย่างนี้มี 'sa' เป็น Username ที่เป็น account อยู่บน Microsoft SQL Server
- **Template** เป็นการกำหนดไฟล์ที่จะนำเอา result ที่ได้จาก Database มาทำการซ้อน (Merge)
- **SQLStatement** จะประกอบด้วยคำสั่ง SQL ต่าง ๆ ที่ทำการ Execute

4. IDC จะ connect เข้ากับ ODBC DataSource และทำการ Execute คำสั่ง SQL ที่อยู่ในไฟล์ .idc นั้น

- การเชื่อมต่อเข้าสู่ ODBC DataSource จะถูกทำโดย IDC ซึ่งในตัวอย่างนี้ เมื่อทำการ load ODBC Driver สำหรับ SQL Server ก็จะทำให้การ Connect เข้าสู่ Server ด้วยชื่อ DataSource ที่กำหนดไว้
- เมื่อการ connection สำเร็จ SQL Statement ใน Internet Database Connector จะถูกส่งเข้าสู่ SQL Server ODBC Driver และเข้าสู่ SQL Server อีกทีหนึ่ง

5. IDC จะ fetch data จาก Database และซ้อน (Merge) result เข้าสู่ไฟล์ HTML extension

- หลังจาก SQL Statement ถูก Execute แล้ว IDC จะอ่าน HTML Extension file ที่ถูกกำหนดไว้ใน Sample.idc (Sample.htx) ไฟล์ HTML Extension (.htx) ประกอบด้วย Tag ต่าง ๆ ของ HTML ซึ่ง IDC จะใช้ควบคุมว่าข้อมูลที่ return จาก SQL Statement จะซ้อน (Merge) เข้าไปอย่างไร และแสดงผลอย่างไร

6. IDC จะส่ง Merge Document นี้กลับไปยัง Internet Information Server และreturn กลับไปยัง client

- หลังจากข้อมูลทั้งหมดได้ถูกทำการ Merge เข้าสู่ Sample.htx แล้ว HTML Document ที่สมบูรณ์ก็จะถูกส่งกลับไปยัง client

บทที่ 4

JAVA SCRIPT

4.1 Java Script

Java Script คือ ภาษาคอมพิวเตอร์ชนิด Object-based สำหรับใช้บนโฮมเพจเพื่อเพิ่มประสิทธิภาพให้กับ ภาษา HTML (Hyper Text Markup Language) เช่น การตอบสนองต่อเหตุการณ์ (Event) หรืออินพุตต่าง ๆ ที่ โฮมเพจได้รับ ไม่ว่าจะเป็นเหตุการณ์จากเมาส์ ควบคุมรูปแบบการแสดงผล เล่นไฟล์เสียง สั่ง Java Applet ให้ทำงาน ติดต่อกับระบบ Plug-in ของบราวเซอร์ (Web Browser) ควบคุมการทำงานของแบบฟอร์ม (Form) และกรอบเฟรม (Frame) ของโฮมเพจ เป็นต้น Java Script คล้ายกับเป็นตัวกลางคอยจัดการและเชื่อมโยงข้อมูลระหว่างหน้าโฮมเพจกับบราวเซอร์ รูปแบบของภาษาจะเป็นแบบ “สคริปต์” และถูกเขียนรวมอยู่กับเนื้อสคริปต์ของภาษา HTML ในความหมายของคำว่า “สคริปต์” คือเป็นไฟล์ข้อมูลตัวอักษรธรรมดาที่เก็บรวบรวมคำสั่งต่าง ๆ ของภาษาคอมพิวเตอร์นั้น ๆ เรียงต่อกันแล้วนำไปรันหรือปฏิบัติงานได้เลย โดยไม่ต้องผ่านการถูกคอมไพล์ก่อน แต่อย่างไรก็ตามการรันหรือการปฏิบัติงานนั้น ต้องใช้อินเตอร์พรีเตอร์เป็นตัวรันหรือเป็นตัวแปลสคริปต์ อินเตอร์พรีเตอร์สำหรับ Java Script ในที่นี้คือการทำงานส่วนหนึ่งของบราวเซอร์นั่นเอง

4.2 Java Script ไม่ใช่ Java

ความเข้าใจเกี่ยวกับ Java Script และ Java ในปัจจุบันนั้น ยังมีบุคคลที่ยังสับสนกันอยู่มาก ในที่นี้จะอธิบายถึงความแตกต่างของทั้ง 2 ภาษา

4.2.1 Java

- Java เป็นภาษาโปรแกรมเชิงวัตถุ (Object-Oriented Programming Language)

Java ถูกพัฒนาขึ้นโดยบริษัท Sun Microsystems ซึ่งไม่ได้ออกแบบเพื่อใช้สร้าง Application บนเว็บเท่านั้นแต่ Java เป็นเหมือนภาษาโปรแกรมมิ่งทั่วไป เช่น C, C++, Pascal และ FORTRAN ในส่วนของสไตล์การเขียนโปรแกรมและ Syntax ของ Java นั้นเหมือนกับ C++ ซึ่งผู้เขียนโปรแกรมสามารถสร้างคลาส (Class) ใหม่ของตัวเองขึ้นมาได้ โดยการจัดกลุ่มของ Object และ Data Structure (ชนิดข้อมูล) ซึ่งคลาสนี้จะหมายถึงกลุ่มของ Object ที่เกี่ยวข้อง หรือสัมพันธ์กัน

- Java สามารถนำไปพัฒนาเป็นได้ทั้ง Application และ Applet

Java ช่วยให้ผู้เขียนโปรแกรมสามารถสร้างโปรแกรมประยุกต์บนเว็บแบบโต้ตอบกันได้และสามารถรันได้โดยไม่ขึ้นกับแพลตฟอร์ม (Platform) หรือระบบปฏิบัติการใด ๆ Java ช่วยให้เขียน Application เผยแพร่บนเครือข่ายหรือ Server แล้วรันบนระบบต่างกันได้อย่างปลอดภัย คลาสไลบรารีมีฟังก์ชันมากมายให้เลือก สามารถ Execute ได้หลาย Path พร้อม ๆ กัน ซึ่งสอดคล้องกับระบบปฏิบัติการ

Windows NT/95 โปรแกรมประยุกต์ที่เขียนขึ้นด้วย Java นี้เรียกว่า จาวาแอปเพล็ต (Java Applet) โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Applet นี้ เป็นโปรแกรมขนาดเล็ก ๆ ที่สามารถทำให้เกิดความเคลื่อนไหว นำเสนอแบบมัลติมีเดีย เล่นเกมส์แบบ Real Time หรือเล่นเกมสับนเครือข่ายกับผู้เล่นหลายคนก็ได้

- Java ต้องอาศัยซอฟต์แวร์ในการคอมไพล์

การเขียนหรือรันโปรแกรม Java จะต้องเตรียมพร้อมด้านฮาร์ดแวร์ และซอฟต์แวร์ โดยฮาร์ดแวร์อาจเป็น Sun SPARCStation ที่รัน Solaris 2.3 หรือสูงกว่า หรือ Intel x86 ที่รัน Windows NT / 95 พร้อมบราวเซอร์ Netscape 2.0 ขึ้นไป, ตัวแปลภาษาจาวา (Compiler Javac) และตัววิวแอปเพล็ต (Applet Viewer) ที่มีอยู่ในรีลีสของซัน (Sun Microsystems)

4.2.2 Java Script

- Java Script เป็นภาษาสคริปต์ที่มีพื้นฐานอยู่บนวัตถุ (Object-Based Structure)

Java Script มีโครงสร้างพื้นฐาน (Basic Structure) และ Syntax เช่นเดียวกับ Java แต่อาจมี ฟังก์ชันและสไตล์การเขียนโปรแกรมแตกต่างไปจาก Java บ้างแม้ว่า Java Script จะมีออบเจกต์พื้นฐาน, พร็อพเพอร์ตี้ (-Properties) , เมธอด (Method) และชนิดข้อมูล (Data Type) จำกัด แต่ก็เพียงพอในการพัฒนา Application บนฝั่งไคลเอนต์ (Client-Side Application) ซึ่งผู้ใช้สร้างออบเจกต์และฟังก์ชันขึ้นใหม่ได้ แต่ไม่เหมือนกับการสร้างคลาสขึ้นมาใหม่ และ Inheritance ที่มีใน Java หรือภาษาโปรแกรมเชิงวัตถุอื่น ๆ ดังนั้น Java Script จึงเป็นเพียงภาษาสคริปต์ที่อยู่บนพื้นฐานของวัตถุ (Object-Based Scripting Language) ไม่ใช่ภาษาโปรแกรมเชิงวัตถุอย่างสมบูรณ์ และไม่ต้องมีการศึกษาการเขียนโปรแกรมมากเหมือน Java ซึ่งเป็นภาษาโปรแกรมมิ่งเต็มตัว

- Java Script เป็น Interpreted Language

นั่นคือ การรันโปรแกรมจะเกิดขึ้นพร้อม ๆ กับการใช้งานหรือ Source Code จะถูกประมวลผลในขณะที่รันโปรแกรม ซึ่งแตกต่างจาก Java เนื่องจากโปรแกรมที่เขียนด้วย Java จะต้องนำไปคอมไพล์ก่อนการรันโปรแกรม ซึ่งข้อดีของ Interpreted Language ก็คือง่ายในการพัฒนา และแก้ไขโปรแกรม

- โค้ดของ Java Script จะถูกฝังอยู่ในเอกสาร HTML

ในการเขียนโค้ด Java Script นั้น จะเขียนแทรกอยู่ภายในเอกสาร HTML และถูกดาวน์โหลดพร้อมกัน

- Java Script ไม่จำเป็นต้องใช้ซอฟต์แวร์ใด ๆ

ในการเขียนโปรแกรมด้วย Java Script ไม่ต้องใช้ซอฟต์แวร์ใด ๆ ทั้งสิ้น เพียงแค่มี Text Editor เช่น Notepad และบราวเซอร์ที่สนับสนุน Java Script ก็สามารถรันโปรแกรมได้แล้วโดยบราวเซอร์ที่สนับสนุน Java Script ก็ได้แก่ Netscape 2.0 และ Internet Explorer 3.0 ขึ้นไป

ตารางแสดงการเปรียบเทียบระหว่าง Java Script และ Java	
Java Script	Java
- ไม่มีการถูกคอมไพล์ก่อนรัน แต่จะรันได้เลยโดยใช้อินเทอร์เน็ตเทอร์บน Client	- ถูกคอมไพล์ที่ Server ก่อนรันบน Client
- Object-Based, รหัสใช้โอปเจ็คท์ภายในซึ่งมีอยู่แล้วและไม่มีการประกาศคลาส หรือการสืบทอดคลาส	- Object-Oriented, แอปเพลทมีการสร้างโอปเจ็คส์และสืบทอดคลาส
- รหัสรวมอยู่ในเนื้อสคริปท์ HTML	- แอปเพลทจะแยกต่างหากจากสคริปท์ HTML
- ไม่มีการประกาศชนิดของตัวแปร	- ต้องมีการประกาศชนิดของตัวแปร
- ตรวจสอบความถูกต้องในขณะที่รัน ถ้ามีข้อผิดพลาดเกิดขึ้นอินเทอร์เน็ตเทอร์บจะแจ้งให้ทราบในขณะที่รัน	- คอมไพล์ ถ้าคอมไพล์ไม่ผ่านไม่สามารถรันได้
- ไม่สามารถเขียนข้อมูลลงฮาร์ดดิสก์ได้	- ไม่สามารถเขียนข้อมูลลงฮาร์ดดิสก์ได้

4.3 ข้อจำกัดของ Java Script

ถึงแม้ Java Script จะมีความยืดหยุ่นใช้งานได้ สามารถพัฒนาได้อย่างอิสระโดยไม่ขึ้นกับสภาพแวดล้อมใด ๆ สำหรับเว็บเพจแต่ละเพจ แต่ก็ยังมีข้อจำกัดสำหรับ Java Script อยู่ดังต่อไปนี้

- ข้อจำกัดด้านบราวเซอร์ (Browser)
- ข้อจำกัดด้านแพลตฟอร์ม (Platform)
- ข้อจำกัดด้านความปลอดภัย (Security)

4.3.1 ข้อจำกัดด้านบราวเซอร์ (Browser-Related Limitations)

เนื่องจากโค้ดของ Java Script ทำการรันบนไคลเอ็นต์ (The Browser) และการอินเทอร์เน็ตเทอร์บ จะต้องกระทำภายในบราวเซอร์ตามลำดับงาน ซึ่งถ้าบราวเซอร์ตัวใดไม่สามารถแปลคำสั่งแล้วทำตามได้ การแสดงผลและการทำงานของเพจนั้น ๆ อาจไม่มีความสมบูรณ์เป็นได้ บราวเซอร์ที่สนับสนุนการทำงานของ Java Script มีหลัก ๆ ที่นิยมใช้กันคือ

- Netscape Navigator 2.0 ขึ้นไป
- Microsoft's Internet Explorer 3.0 ขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เบราว์เซอร์รุ่นก่อนหน้านี้ ไม่ได้มีการสนับสนุนการแปลภาษาของ Java Script ไม่เพียงแต่เบราว์เซอร์ Navigator และ Explorer เท่านั้นที่ใช้งานกันในปัจจุบัน ยังมีเบราว์เซอร์จากค่ายอื่น ๆ อีกมากมายเช่น

- Internet Explorer 2.0
- Mosaic, เรืองฟอร์มบน NCSA, Spry, Quarterdeck
- Lynx เบราว์เซอร์แบบข้อความล้วน
- MacWeb เบราว์เซอร์บนระบบ Macintosh
- HotJava เบราว์เซอร์ที่ทำงานเกี่ยวกับ Java โดยตรง ของ Sun Microsystems และอื่น ๆ

4.3.2 ข้อจำกัดทางด้านแพลตฟอร์ม (Platform-Related Limitations)

Java Script สามารถรันได้ทุกแพลตฟอร์มที่ Navigator ของ Netscape ทำงานได้ (Windows, Macintosh, and Unix) แต่ทุกแบบก็ไม่ได้รันแล้วได้ผลเหมือนกันหมด เช่นตัวอย่างดังต่อไปนี้

- การใช้คุกกี้ (Cookies) ในการเก็บข้อมูลชั่วคราวต่าง ๆ เกี่ยวกับผู้ใช้งาน ซึ่งทั้งใน Windows และใน Macintosh มีการทำงานที่ไม่เหมือนกัน
- การสร้างตัวเลขแบบสุ่ม (Random Number Generation) โดยเมครอดนี้ทำงานได้เฉพาะบนยูนิกซ์แพลตฟอร์มเท่านั้น

4.3.3 ข้อจำกัดด้านความปลอดภัย (Security-Related Limitations)

ในความจริงแล้ว ถ้าจะพูดให้ถูกต้องอาจเรียก Java Script ได้ว่า ถูกจำกัด มากกว่าการเป็นแค่ข้อจำกัด เนื่องจากว่าเป้าหมายหลักคือป้องกัน และขัดขวางบุคคลที่จะเข้าไปทำในสิ่งที่ไม่ใช่ผู้ใช้งานเว็บเพจแบบธรรมดา ซึ่งเราอาจเคยได้ยินเกี่ยวกับการลักลอบเข้าใช้ข้อมูลในระบบ Network อยู่เสมอ ๆ ด้วยเหตุนี้ทางผู้สร้างเบราว์เซอร์จากค่ายต่าง ๆ ก็พยายามที่จะทำให้ Java Script มีความปลอดภัยสำหรับการใช้งานได้

Java Script นี้ถูกออกแบบให้เป็น “Web-Safe“ ให้มีความปลอดภัยมากกว่าวิธีการใช้งานเว็บแบบเดิม ๆ (เช่น Perl scripting และ CGI) ดังต่อไปนี้

- Java Script ไม่สามารถเปิด, อ่าน, เขียน หรือ บันทึกแฟ้ม บนคอมพิวเตอร์ของผู้ใช้งานได้ ทำได้เฉพาะการเข้าใช้งานข้อมูล (Information) ต่าง ๆ ที่มีอยู่ในหน้านั้นเท่านั้น
- Java Script ไม่สามารถเปิด, อ่าน, เขียน หรือ บันทึกแฟ้ม บนคอมพิวเตอร์เซิร์ฟเวอร์ (Web Server) ได้ โดยที่สามารถใช้ได้เพียง นำ Java Script มาสร้างคำสั่ง HTML ซึ่งคำสั่ง HTML ก็ไม่สามารถใช้เปิดไฟล์ได้อยู่แล้ว
- Java Script ไม่สามารถที่จะใช้สร้างไวรัส (Virus) เพื่อทำความเสียหายบนเครื่องคอมพิวเตอร์ได้ ยกเว้นแต่กรณีเลวร้ายที่สุดคือใช้ Java Script เพื่อทำสิ่งใดสิ่งหนึ่งแล้วไม่เป็นไปตามลำดับขั้นตอน เกิดการ Error ขึ้นจนไม่สามารถควบคุมได้ (บางครั้งต้องทำการปิดเบราว์เซอร์)

4.4 ชนิดของข้อมูลและการตั้งชื่อตัวแปร

4.4.1 ชนิดของข้อมูลใน Java Script แบ่งออกได้ดังนี้

1. ข้อมูลชนิดตัวเลข (Numbers)

ข้อมูลชนิดนี้ง่ายต่อการเข้าใจที่สุด เพราะใน Java Script มีข้อมูลชนิดตัวเลขให้ใช้แต่ไม่ได้แยกกันชัดเจนว่าเป็น Byte, Integer, Float หรือ Double แต่สามารถใช้รวมกันได้ ผู้ใช้สามารถกำหนดตัวแปรชนิดตัวเลขให้มีค่าเป็นได้ทั้งจำนวนเต็มและจำนวนจริงไปพร้อม ๆ กัน เช่น 53, 4.3875 หรือ -23 เป็นต้น

2. ข้อมูลชนิดตรรกะหรือบูลีน (Boolean)

ใช้ในการคำนวณร่วมกับโอเปอเรเตอร์ทางตรรกศาสตร์ เราสามารถกำหนดค่าตัวแปรชนิดตรรกะ หรือ บูลีนให้มีค่าเป็น True หรือ False เช่นเดียวกันกับในภาษาอื่น ๆ

3. ข้อมูลชนิดสตริง (String)

สตริงเป็นข้อมูลที่ต้องการเครื่องหมายฟันทวน (Quotation marks) เป็นตัวกำหนดขนาด (เช่น "Hello" หรือ "First_last")

4. ข้อมูลชนิดว่างเปล่า (Null)

เป็นค่าพิเศษที่บ่งบอกถึงความว่างเปล่า หรือ ไม่มีค่าอะไรเลย

4.4.2 การตั้งชื่อตัวแปร

การตั้งชื่อตัวแปร (Variable Name) Java Script ยอมรับตัวอักษรภาษาอังกฤษ (a-z , A_Z) , ตัวเลข (0-9) และเครื่องหมายขีดเส้นใต้ (Underscore '_') อักษรตัวแรกของชื่อตัวแปรต้องเป็นตัวอักษรหรือเครื่องหมายขีดเส้นใต้เท่านั้น (เช่น myValue, height9_8, _iName, ฯลฯ) สำหรับการตั้งชื่อนั้นอักษรตัวเดียวกันแต่เป็นอักษรตัวใหญ่ตัวเล็ก (เช่น a กับ A, p กับ P) จะไม่เหมือนกัน เช่น Mr_Withaya และ mr_Withaya จะไม่เหมือนกัน

หมายเหตุ : ภาษา Java Script ไม่จำเป็นต้องมีการประกาศชนิดของตัวแปรเหมือนกันกับภาษาอื่น เพราะการกำหนดชนิดและการแปลงชนิดของตัวแปรของภาษา Java Script จะกระทำโดยอัตโนมัติ

4.5 การกำหนด Literals

คำว่า Literal ในเชิงโปรแกรมมิ่ง จะหมายถึงข้อมูลที่ไม่มีการเปลี่ยนแปลงค่า ไม่ว่าจะเกิดอะไรขึ้น หรืออีกนัยหนึ่งอาจกล่าวได้ว่า Literal คือ ค่า Constant นั้นเอง ค่า Literals ถูกจำแนกได้ 4 ประเภทคือ

4.5.1 ค่าจำนวนเต็ม (Integers)

สามารถกำหนดเป็นเลขฐานสิบ เลขฐานสิบหก และเลขฐานแปด ซึ่งหากเป็นเลขฐานสิบต้องประกอบด้วยตัวเลขที่ไม่ขึ้นต้นด้วย 0 ถ้าเป็นเลขฐานแปดจะต้องนำหน้าด้วย 0 และตัวเลขถัดมาต้องเป็นเลข 0-7 ส่วนของเลขฐานสิบหกจะต้องนำหน้าด้วย 0x หรือ 0X และถัดมาเป็นเลข 0-9 หรือตัวอักษร a-f หรือ A-F

4.5.2 ค่าจำนวนจริง (Floating-Point Integers)

การกำหนดจำนวนคงตัวที่เลขจำนวนจริง โดยเลขจำนวนจริงดังกล่าวต้องประกอบด้วย ตัวเลข 0-9 ซึ่งเขียนอยู่หน้าและหลังจุดทศนิยม กรณีที่เป็นเลขยกกำลังก็ให้ใช้ตัว "e" หรือ "E" นำหน้า ส่วนที่ยกกำลัง โดยตัวเลขยกกำลังอาจนำหน้าด้วยเครื่องหมาย + หรือ - ก็ได้ เช่น 3E-12 จะหมายถึง 3×10^{-12} เป็นต้น

4.5.3 ค่าคงตัวตรรกะ (Boolean Literals)

การกำหนดค่าคงตัวของตัวแปรชนิดตรรกะหรือบูลีน เป็นได้ 2 ชนิดคือ True หรือ False เท่านั้น

4.5.4 ค่าคงตัวชนิดสตริง (String Literals)

การใช้จำนวนค่าคงตัวชนิดสตริงจะต้องกำหนดอยู่ในเครื่องหมายอัญประกาศ (") หรือ เดี่ยว (') สำหรับส่วนที่เป็นสตริงอาจเป็นได้ทั้งตัวเลขและตัวอักษร เช่น "2499" หรือ "Titanic" เป็นต้น นอกจากนี้สตริงยังอาจประกอบด้วยตัวอักษรพิเศษบางกลุ่ม ได้แก่

- \b ใช้แทน Backspace
- \f ใช้แทน Form Feed
- \n ใช้แทนการขึ้นบรรทัดใหม่ หรือ New Line
- \r ใช้แทน Carriage Return
- \t ใช้แทน TAB

4.6 ฟังก์ชัน (Function)

ฟังก์ชัน คือ โปรแกรมย่อยที่สามารถเรียกให้ทำงานหลาย ๆ ครั้งโดยไม่ต้องเขียนโปรแกรมซ้ำ ๆ กันหลายแห่ง ซึ่งจะช่วยประหยัดเวลาในการพัฒนาโปรแกรม อีกทั้งการปรับปรุงแก้ไขโปรแกรมก็จะง่ายขึ้นคือแก้ไขเพียงที่เดียวเท่านั้น

Java Script ช่วยให้ผู้เขียนโปรแกรมสามารถกำหนดฟังก์ชันขึ้นมาใหม่ได้ เพื่อนำไปใช้ให้เหมาะกับงานที่ต้องการ และเนื่องจาก Java Script ทำงานอยู่บนพื้นฐานของอ็อบเจกต์ ดังนั้นฟังก์ชันของ Java Script จึงสามารถนำไปใช้เป็นเมธอดในอ็อบเจกต์ที่เราสร้างขึ้นใหม่ได้

4.6.1 การทำงานของฟังก์ชัน

การใช้งานฟังก์ชันนั้น ต้องเริ่มจากการกำหนดฟังก์ชันขึ้นมาก่อน และมีขั้นตอนการทำงานของโปรแกรมอยู่ภายใน หลังจากนั้นจะมีการเรียกใช้ โดยเรียกใช้ชื่อของฟังก์ชัน เมื่อ Java Script เจอการเรียกใช้ฟังก์ชัน มันจะกระโดดไปทำงานในฟังก์ชัน เมื่อการทำงานในฟังก์ชันนั้นสิ้นสุดลง Java Script ก็จะกระโดดกลับมายังคำสั่งถัดไปหลังจากคำสั่งที่เรียกใช้งาน

4.6.2 การกำหนดฟังก์ชัน

การกำหนดฟังก์ชันจะเริ่มต้นการเขียนฟังก์ชันด้วยคำว่า Function ตามด้วยชื่อของฟังก์ชัน และพารามิเตอร์ที่จะส่งให้ฟังก์ชัน ดังรูปแบบต่อไปนี้

```
function FunctionName( parameters,...,..... ) {
    Java Script Code
}
```

การตั้งชื่อของฟังก์ชันจะใช้กฎเกณฑ์เดียวกับการตั้งชื่อตัวแปรของภาษาโปรแกรมมิ่งทั่ว ๆ ไป คือ ตัวแรกต้องเป็นตัวอักษรแล้วจะตามด้วยตัวอักษรหรือตัวเลขก็ได้ และใช้เครื่องหมายขีดล่าง (Underscore) ได้ แต่ชื่อของฟังก์ชันจะต้องคำนึงถึงตัวพิมพ์ใหญ่ด้วย (Case Sensitive) เช่น MyFunction, Myfunction และ myFunction จะเป็นฟังก์ชันที่ต่าง ๆ กัน และพารามิเตอร์ที่จะส่งให้ฟังก์ชัน ถ้ามีมากกว่าหนึ่งตัว ให้คั่นแต่ละพารามิเตอร์ด้วยเครื่องหมายจุลภาค (,) หากฟังก์ชันไม่มีการรับพารามิเตอร์ ภายในเครื่องหมายวงเล็บ () ก็ว่างเปล่าดังรูปแบบต่อไปนี้

```
function FunctionName( ) {
    Java Script Code
}
```

4.6.3 ตำแหน่งที่จะเขียนฟังก์ชัน

การเขียนฟังก์ชันนั้นต้องเขียนอยู่ในระหว่างแท็ก <SCRIPT> และ </SCRIPT> เสมอ ในระหว่างการโหลดไฟล์ HTML โค้ดสคริปต์ที่อยู่ในรูปฟังก์ชันจะถูกอ่านเข้ามาเก็บไว้ในหน่วยความจำ แต่ยังไม่รันในทันที เนื่องจากฟังก์ชันใด ๆ จะรันได้ต้องถูกเรียกใช้จากส่วนอื่นของไฟล์เท่านั้น ดังนั้นตำแหน่งที่เหมาะสมสำหรับการเขียนฟังก์ชันจึงควรอยู่ในระหว่างแท็ก <HEAD> และ </HEAD> เพราะใจได้โหลดเข้ามาในหน่วยความจำก่อนมีการเรียกใช้

4.6.4 การส่งผ่านพารามิเตอร์

การส่งผ่านพารามิเตอร์ให้ฟังก์ชัน มีข้อควรทราบดังนี้

1. พารามิเตอร์ที่ถูกส่งให้ฟังก์ชันจะใช้เป็นชื่อตัวแปรหรือค่าคงที่ (Literals) ก็ได้
2. ถ้าตัวแปรถูกส่งให้ฟังก์ชัน การแก้ไขค่าตัวแปรในฟังก์ชันจะไม่มีผลต่อค่าตัวแปรที่ส่งไปให้ เนื่องจากตัวแปรที่เกิดในฟังก์ชันจะเป็นตัวแปรแบบท้องถิ่น (Local Variable) คือใช้ได้เฉพาะภายในฟังก์ชันเท่านั้น
3. ค่าตัวแปรภายในฟังก์ชัน จะถูกกำหนดขึ้นมาใหม่จากการเรียกใช้ฟังก์ชันแต่ละครั้ง
4. ตัวแปรที่ถูกประกาศภายในฟังก์ชันจะเรียกว่า Local Variable ส่วนตัวแปรที่ถูกประกาศภายในโค้ดสคริปต์แต่อยู่นอกฟังก์ชันจะเรียกว่า Global Variable
5. ถ้าตัวแปรที่ถูกประกาศภายในฟังก์ชัน (Local Variable) มีชื่อเดียวกับ Global Variable การเปลี่ยนแปลงค่าตัวแปรภายในฟังก์ชันจะไม่มีผลต่อ Global Variable นั้น

4.6.5 การส่งค่ากลับ

หลังจากที่ฟังก์ชันทำงานเสร็จแล้ว อาจต้องมีการส่งค่ากลับไปให้ส่วนที่เรียกฟังก์ชันนั้นทำได้ การส่งค่ากลับทำได้โดยใช้คำสั่ง Return แล้วตามด้วยค่าที่ต้องการส่งกลับ โดยค่าที่ส่งกลับนี้อาจเป็นค่าคงที่, ตัวแปร หรือนิพจน์ ที่สามารถหาค่าตอบได้เพียงค่าเดียวเท่านั้น เช่นตัวอย่างการเขียนฟังก์ชันการหาค่ากำลังสองดังต่อไปนี้

```
function power( number ) {
    var power = number * number;
    return power;
}
```

หรือ

```
function power( number ) {
    return number * number;
}
```

4.7 ระบบสมการ (Expression)

ปกติใน Expression จะต้องประกอบไปด้วยกลุ่มตัวแปร จำนวนค่าคงตัว และโอเปอเรเตอร์ โดยสามารถประเมินค่าของ Expression ออกมาเป็นข้อมูลชนิดตัวเลข สตริง หรือบูลีนให้ได้ ค่าที่เกิดขึ้นอาจใช้กำหนดค่ากับตัวแปรหรือใช้เป็นค่าอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากผู้เขียนโปรแกรมเคยเขียนโปรแกรมด้วยภาษาซี ก็คงจำได้ว่าในภาษาซีมีการใช้ Condition Expression ซึ่งก็เป็นประโยคเงื่อนไขที่ใช้สำหรับกำหนดค่า การใช้ Condition Expression แทนประโยค If...Then จะช่วยให้เขียนโปรแกรมได้กระชับมากขึ้น รูปแบบของประโยคมีดังนี้

$(\text{condition}) ? \text{val1}, \text{val2}$

ความหมายของประโยคนี้คือ ถ้า condition เป็นจริง Expression ดังกล่าวจะประเมินค่าเป็น val1 มิฉะนั้นจะเท่ากับค่าของ val2

4.8 โอเปอเรเตอร์ (Operator)

การคำนวณจะเกี่ยวข้องกับการหาค่าของโอเปอแรนด์ (Operands) ด้วย โอเปอเรเตอร์ (Operators) โดยที่ โอเปอเรเตอร์ คืออักษรพิเศษที่เป็นตัวควบคุมว่าสมการ (Expression) นั้นจะหาค่าด้วยวิธีใด (เช่น +, -, / หรือ *) ส่วน โอเปอแรนด์ คือสิ่งที่อยู่ในสมการ (เช่น ตัวแปร, ค่าคงที่ หรือ สมการอื่น ๆ)

Java Script แบ่งโอเปอเรเตอร์ออกเป็น 2 ชนิดคือ Binary และ Unary โดยที่ Binary โอเปอเรเตอร์ ต้องการโอเปอแรนด์ สองตัว ที่อยู่ในลักษณะหนึ่งตัวอยู่หน้า โอเปอเรเตอร์ และ อีกหนึ่งตัวอยู่หลัง โอเปอเรเตอร์ ดังรูปแบบต่อไปนี้

Operand1 operator Operand2 เช่น 2499 + 2541

สำหรับ Unary โอเปอเรเตอร์ ต้องการเพียงแค่ โอเปอแรนด์ เพียงตัวเดียว ที่อยู่ด้านหน้าหรือด้านหลังของโอเปอเรเตอร์ดังรูปแบบต่อไปนี้

Operator Operand เช่น -2499, -x

หรือ

Operand Operator เช่น x++

4.8.1 โอเปอเรเตอร์คำนวณ (Aithmetic Operators)

มีทั้งหมด 8 โอเปอเรเตอร์ คือ +, -, *, /, %, --, ++ และ - โดย 5 ตัวแรกจะเป็นแบบ Binary และ 3 ตัวหลังจะเป็น Unary ซึ่งสามารถแบ่งเป็นประเภทต่าง ๆ ดังต่อไปนี้

4.8.1.1 Standard Arithmetic Operators

- + โอเปอเรเตอร์ทำการบวก
- - โอเปอเรเตอร์ทำการลบ
- * โอเปอเรเตอร์ทำการคูณ
- / โอเปอเรเตอร์ทำการหาร

4.8.1.2 Modulus (%)

ใช้หาค่าผลลัพธ์ที่เป็นเศษเหลือจากการหารของตัวเลขจำนวนเต็ม เช่น $22 \% 5$ ผลลัพธ์ที่ได้จะเป็น 2

4.8.1.3 Increment (++)

ใช้สำหรับเพิ่มค่าของโอเปอเรเตอร์ขึ้นหนึ่งค่า เช่น `Operand ++` ถ้าหากในขณะนั้น `Operand` มีค่าเท่ากับ 2499 เมื่อใช้ โอเปอเรเตอร์ นี้แล้วก็จะได้ค่าเป็น 2500 ในอีกความหมายหนึ่งของ Operator ตัวนี้คือ `Operand = Operand + 1`

4.8.1.4 Decrement (--)

ใช้สำหรับลดค่าของโอเปอเรเตอร์ลงหนึ่งค่า และมีวิธีการคล้ายกันกับ Increment

4.8.1.5 Unary Negation (-)

เป็นโอเปอเรเตอร์แสดงว่า `Operand` นั้น มีค่าเป็นลบ

4.8.2 โอเปอเรเตอร์เปรียบเทียบ (Comparison Operators)

เป็นโอเปอเรเตอร์ที่ทำการเปรียบเทียบและส่งค่ากลับเป็นแบบตรรกะ (True หรือ False) ประกอบด้วย 6 ชนิดด้วยกันได้แก่

- Equal (==) ถ้าโอเปอเรนต์เท่ากันจะ Return True
- Not Equal (!=) ถ้า โอเปอเรนต์ ไม่เท่ากันจะ Return True
- Greater than (>) ถ้า โอเปอเรนต์ ซ้ายมากกว่า โอเปอเรนต์ ขวาจะ Return True
- Greater than or equal to (>=) ถ้า โอเปอเรนต์ ซ้ายมากกว่า หรือเท่ากับ โอเปอเรนต์ ขวาจะ Return True
- Less than (<) ถ้า โอเปอเรนต์ ซ้ายน้อยกว่า โอเปอเรนต์ ขวาจะ Return True
- Less than or equal to (<=) ถ้า โอเปอเรนต์ ซ้ายน้อยกว่าหรือเท่ากับ โอเปอเรนต์ ขวาจะ Return True

4.8.3 โอเปอเรเตอร์สตริง (String Operator)

เป็นโอเปอเรเตอร์ที่ใช้จัดการกับสตริง ซึ่งนอกจากจะโอเปอเรเตอร์ที่ใช้เปรียบเทียบแล้วยังอาจใช้เครื่องหมาย '+' เพื่อทำการเชื่อม (Concatenated) สตริงสองตัวเข้าไว้ด้วยกัน หรืออาจใช้โอเปอเรเตอร์ '+=' ก็ได้

4.8.4 โอเปอเรเตอร์ด้านลอจิก (Logical Operators)

โอเปอเรเตอร์ชุดนี้ใช้กระทำกับโอเปอเรนด์ชนิดตรรกะ (Boolean) เพื่อประเมินค่าว่าเป็น True หรือ False ประกอบด้วย 3 กรณีคือ

- && (and) จะ Return True ถ้า โอเปอเรนด์ เป็นจริงทั้งคู่
- || (or) จะ Return True ถ้า โอเปอเรนด์ ตัวใดตัวหนึ่งเป็นจริง
- ! (not) ใช้สำหรับสลับค่าจาก True เป็น False

4.8.5 โอเปอเรเตอร์เลื่อนบิต (Bitwise Operators)

โอเปอเรเตอร์กลุ่มนี้ใช้สำหรับจัดการกับ โอเปอเรนด์ชนิดตัวเลขในฐานะเลขฐานสองแทนที่จะเป็นเลขฐานสิบแบ่งเป็น 2 กลุ่ม

4.8.5.1 กลุ่มที่ 1 โอเปอเรเตอร์ที่ใช้จัดการแบบลอจิกได้แก่ & (and), | (or) และ ^ (xor) การกระทำกับ โอเปอเรนด์จะจับคู่ทีละบิตจนครบ 32 บิต

4.8.5.2 กลุ่มที่ 2 โอเปอเรเตอร์ที่ใช้เลื่อนบิตในโอเปอเรนด์ ได้แก่

- << สำหรับเลื่อนบิต ไปทางซ้าย
- >> สำหรับเลื่อนบิตไปทางขวาโดยจะเลื่อนเครื่องหมายไปด้วย
- >>> สำหรับเลื่อนบิตไปทางขวาโดยใช้เลขศูนย์เลื่อนเดิมเข้าไปแทน

4.8.6 โอเปอเรเตอร์กำหนดค่า (Assignment Operators)

Assignment Operator ใช้สำหรับกำหนดค่าโอเปอเรนด์ด้านซ้ายมือจากค่าโอเปอเรนด์ ด้านขวามือ โดยใช้หลักการนำโอเปอเรเตอร์ที่กล่าวมาแล้วมาใช้ร่วมกับเครื่องหมายเท่ากับเหมือนในภาษาซี โอเปอเรเตอร์แบบนี้แบ่งได้เป็น 2 กลุ่ม

4.8.6.1 กลุ่มที่ 1 ใช้เครื่องหมาย โอเปอเรเตอร์ทางคณิตศาสตร์คือ +=, -=, *=, /= และ %=

4.8.6.2 กลุ่มที่ 2 ใช้ โอเปอเรเตอร์จัดการบิตคือ <<=, >>=, >>>=, &=, |= และ ^=

ความหมายของ โอเปอเรเตอร์ Assignment จะเป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- $x += y$ จะเท่ากับ $x = x + y$
- $x -= y$ จะเท่ากับ $x = x - y$
- $x >>= y$ จะเท่ากับ $x = x >> y$

4.9 อ็อบเจ็กต์ (Object)

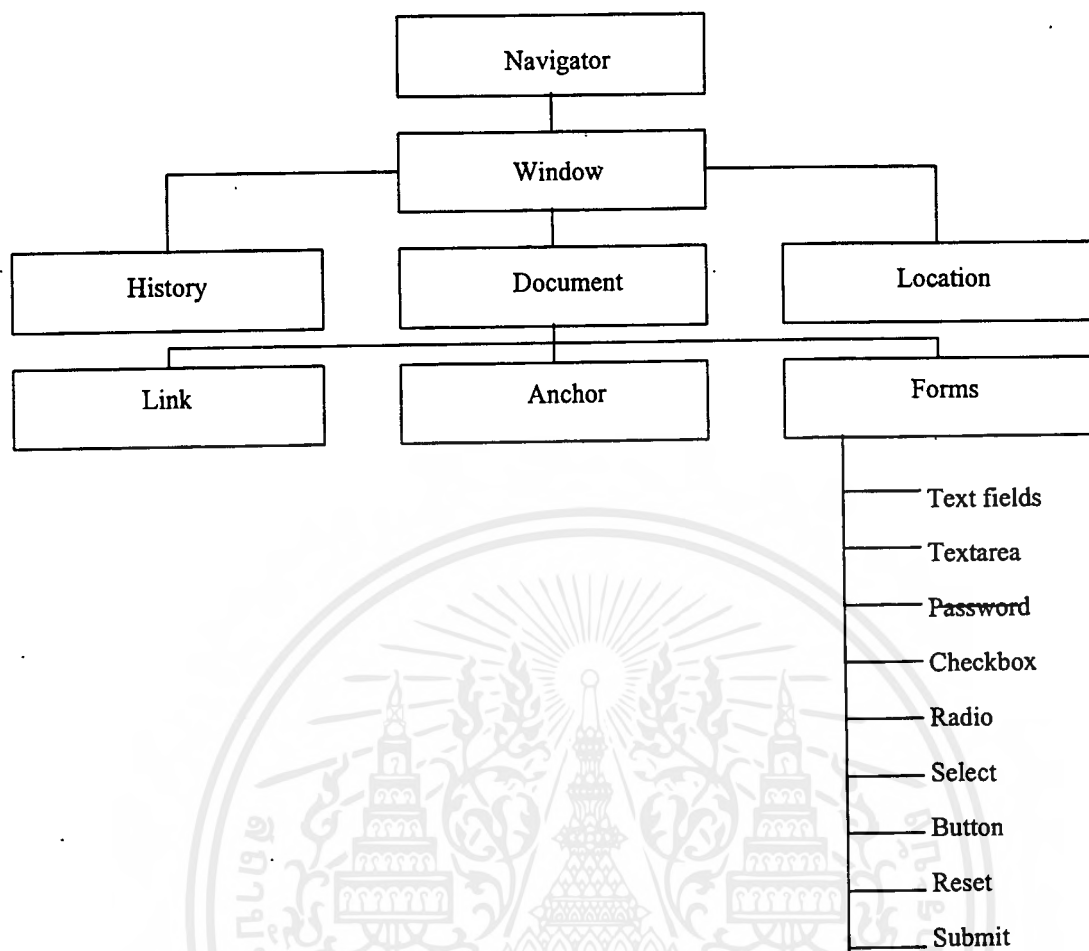
“ อ็อบเจ็กต์ ” คือข้อมูลในรูปแบบหนึ่งของภาษา Java Script ซึ่งเป็นข้อมูลที่ประกอบขึ้นด้วยคุณสมบัติหรือพร็อพเพอร์ตี้ (Property) ต่าง ๆ พร็อพเพอร์ตี้ของอ็อบเจ็กต์มีอยู่ 2 ชนิดคือ ตัวแปร และ ฟังก์ชัน (หรือ เมธอด) ดังเช่นในโลกแห่งความเป็นจริงอ็อบเจ็กต์เปรียบเสมือน “ วัตถุ ” เช่น รถยนต์ย่อมมีคุณสมบัติที่บ่งบอกถึงเอกลักษณ์ของตัวเองเช่นมีล้อมีเครื่องยนต์ (ตัวแปรของอ็อบเจ็กต์ จะเก็บข้อมูลของล้อและเครื่องยนต์) และรถยนต์มีหน้าที่เป็นพาหนะสำหรับใช้สัญจรไปมา โดยใช้ล้อเพื่อเคลื่อนที่ และมีเครื่องยนต์ที่เผาผลาญเชื้อเพลิง เพื่อส่งแรงขับเคลื่อนไปสู่ล้อให้สามารถพุ่งทะยานไปข้างหน้าได้ (ฟังก์ชันของอ็อบเจ็กต์ทำหน้าที่นำเอาข้อมูลจากตัวแปร ที่มีอยู่ไปใช้คำนวณ เพื่อทำให้เกิดกิจกรรมในที่นี้คือ ทำให้รถยนต์เคลื่อนที่ไปได้) และนี่คือหลักการของ OOP ซึ่งพยายามจำลองโลกแห่งความเป็นจริง มาสู่การเขียนโปรแกรมคอมพิวเตอร์ ซึ่งเป็นคุณสมบัติหนึ่งของภาษา Java Script การอ้างอิงอินสแตนซ์ (Instance) หรือตัวตนหนึ่ง ๆ ของอ็อบเจ็กต์ในหน่วยความจำทำได้โดยกำหนดตามรูปแบบนี้

ObjectName.PropertyName

เช่น document.write(“ My heart will go on “) หลังจากทำคำสั่งนี้แล้ว Java Script ก็แสดงข้อความว่า My heart will go on ออกมา ซึ่งก็เป็นการเรียกใช้ฟังก์ชัน ที่เป็นพร็อพเพอร์ตี้หนึ่งของอ็อบเจ็กต์ document

4.10 อ็อบเจ็กต์โมเดลของ Java Script (Java Script Objects)

ใน Java Script มีลำดับของ Hierarchy ที่อยู่ในลักษณะของ Tree ซึ่งในแต่ละ Level จะประกอบด้วยอ็อบเจ็กต์ต่าง ๆ ดังต่อไปนี้



- อ็อบเจกต์ Window

เป็นอ็อบเจกต์ที่อยู่ด้านบนสุด อ็อบเจกต์อื่น ๆ ทั้งหมดอยู่ภายใต้อ็อบเจกต์ Window จะมีความสัมพันธ์แบบหนึ่งต่อหนึ่งกับอ็อบเจกต์ Window เช่นภายใต้ Window หนึ่งจะประกอบไปด้วย หนึ่ง Location หรือ URL เดียวเท่านั้น อ็อบเจกต์ Window จะมีเมธอดและพร็อบเพอร์ตี้ ที่ใช้ในการทำงานกับ Window โดย Window อาจประกอบด้วยอ็อบเจกต์ Frame หลาย ๆ เฟรมก็ได้ โดย อ็อบเจกต์ Frame ก็เป็น Window นั้นเอง

- อ็อบเจกต์ Location

มีพร็อบเพอร์ตี้ และ เมธอดที่ใช้ทำงานกับ URL ในขณะนั้น

- อ็อบเจกต์ History

มีพร็อบเพอร์ตี้ สำหรับใช้เก็บรายละเอียดของ URL ที่เคยใช้

- อ็อบเจกต์ Document

เป็นอ็อบเจกต์หนึ่งที่ถูกใช้งานมาก โดยจะมีอ็อบเจกต์, พร็อบเปอร์ตี้ และ เมททอดที่ใช้ทำงานกับ Document หรือเอกสารที่เราเปิดใช้งานอยู่ในขณะนั้น ใน Document หนึ่ง ๆ อาจประกอบไปด้วย Form หลายฟอร์มก็ได้ และ ในแต่ละฟอร์มก็จะประกอบไปด้วย Element ต่าง ๆ

4.11 Events ใน Java Script

Java Script เป็นภาษาแบบ Event-Driven Language คือสามารถใช้เหตุการณ์หรือ Events (เช่นจากเมาส์คลิก หรือ ปุ่มกด) ควบคุมการโต้ตอบกันระหว่างผู้ใช้งานเพจนั้นกับแอปพลิเคชันที่สร้างขึ้น เพื่อให้ผลลัพธ์ที่เกิดขึ้นนั้นมีความเข้าใจกันได้ดีระหว่างผู้สร้างและผู้ใช้งาน และเมื่อมีเหตุการณ์แล้วก็จะต้องมีตัวจับเหตุการณ์หรือ Event Handlers เพื่อจับเหตุการณ์ที่เกิดขึ้นแล้วมาทำงานที่ได้วางไว้ ซึ่งตัวจับเหตุการณ์นี้จะต้องเลือกใช้ให้เหมาะกับเหตุการณ์ เพราะในขณะที่ใช้งานเว็บเพจอยู่นั้น ได้เกิดเหตุการณ์ต่าง ๆ ขึ้นมากมาย อาทิเช่น การกดปุ่มเมาส์ การเปลี่ยนแปลงตำแหน่งเคอร์เซอร์ของเมาส์ให้ไปอยู่ในหรือนอกพื้นที่ของแบบฟอร์มนั้น ๆ เป็นต้น เราสามารถเขียนตัวจับเหตุการณ์หรือ Event Handlers ได้ตามรูปแบบดังนี้

```
< TAG EventHandler = " Java Script Code " >
```

ตัวจับเหตุการณ์ใน Java Script แบ่งออกเป็น 2 ประเภทได้แก่

1. System Events เช่น onLoad และ onUnload
 2. Mouse Events เช่น onCicck, onFocus, onBlur, onChange, onSelect, onSubmit และ onMouseOver
- จากเหตุการณ์ดังกล่าวมาข้างต้นสามารถสรุปได้ดังต่อไปนี้

ตารางแสดงตัวจับเหตุการณ์ที่เหมาะสมกับเหตุการณ์ที่เกิดขึ้น		
เหตุการณ์	เกิดขึ้นเมื่อ	ตัวจับเหตุการณ์
blur	ออกจากพื้นที่ที่กำลังทำงานอยู่	onBlur
click	กดปุ่มเมาส์ลงไปบนแบบฟอร์มหรือจุดเชื่อมต่อ (Links, Anchors)	onClick
change	มีการเปลี่ยนแปลงข้อความใน Textfields, Textareas หรือมีการเปลี่ยนแปลงของแบบฟอร์มการเลือก Radio buttons, Check buttons	onChange
focus	เข้าไปทำงานในพื้นที่ของแบบฟอร์ม	onFocus
load	โหลดข้อมูลในหน้าโฮมเพจ	onLoad
mouseover	เลื่อนเคอร์เซอร์ของเมาส์ไปอยู่บนพื้นที่จุดเชื่อมต่อ(Links, Anchors)	onMouseOver
select	มีการเลือกข้อมูลภายในแบบฟอร์ม	onSelect
submit	การยอมรับข้อมูลทั้งหมดในแบบฟอร์ม	onSubmit
unload	ออกจากหน้าโฮมเพจ	onUnload

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานตัวจับเหตุการณ์ ต้องใช้ให้เหมาะกับเหตุการณ์ เนื่องจากอ็อบเจกต์แต่ละตัวมีการรับรู้เหตุการณ์ที่ไม่เหมือนกัน ดังรายการต่อไปนี้

อ็อบเจกต์	ตัวจับกับเหตุการณ์ที่ใช้ได้
Selection List	onBlur, onChange, onFocus
Text Element	onBlur, onChange, onFocus, onSelect
Textarea Element	onBlur, onChange, onFocus, onSelect
Button Element	onClick
Checkbox Element	onClick
Radio Button	onClick
Hypertext Link	onClick, onMouseOver, onMouseOut
Reset Button	onClick
Submit Button	onClick
Document	onLoad, onUnload
Window	onLoad, onUnload
Form	onSubmit

4.12 การใช้งานฟอร์ม

ฟอร์มเป็นรูปแบบหนึ่งของการรับข้อมูลจากผู้ใช้ ซึ่งสามารถใช้ Java Script เข้ามาช่วยเสริมการทำงานของอิลิเมนต์ (element) หรืออินพุตต่าง ๆ ได้

4.12.1 อ็อบเจกต์ฟอร์ม

อ็อบเจกต์ฟอร์มเป็นหนึ่งในหลาย ๆ อ็อบเจกต์ที่มีการใช้งานมากในการเขียน Java Script อ็อบเจกต์ฟอร์มที่กำหนดไว้บนเพจอาจประกอบไปด้วยอ็อบเจกต์ต่าง ๆ เช่น เช็กบ็อกซ์ (Checkbox), เรดิโอบัทตอน (Radio button), ซีเล็คชันลิสต์ (Selection list) และ ซับมิท (Submit) เป็นต้น

การเขียนฟอร์มนั้นจะเขียนอยู่ในระหว่างแท็ก <FORM> และ </FORM> โดยภายในแท็ก <FORM> นั้นจะเขียนแท็ก <INPUT> เพื่อสร้างอินพุตหรืออิลิเมนต์ (Element) ขึ้นมาในฟอร์มซึ่งต้องบอกชนิดของอิลิเมนต์ด้วย เนื่องจากอิลิเมนต์ส่วนใหญ่ใช้แท็กเดียวกันคือ <INPUT> โครงสร้างการเขียนแท็ก <FORM> และ <INPUT> เป็นดังนี้

```
<FORM>
```

```
<INPUT>
```

```
<INPUT>
```

```
</FORM>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.12.2 ฟอร์มเปอร์ตี้ของอ็อบเจกต์ฟอร์ม

- **action** ฟอร์มเปอร์ตี้นี้จะใช้อ่านค่าที่กำหนดในแอตทริบิวต์ **Action** ของอ็อบเจกต์ฟอร์ม ซึ่งเป็นค่า URL ของเซิร์ฟเวอร์ที่จะส่งข้อมูลให้ เช่น

```
<FORM METHOD=POST ACTION="/scripts/Register.idc">
```

ดังนั้นฟอร์มเปอร์ตี้ **action** มีค่า `"/scripts/Register.idc"`

- **element** ผู้ใช้อาจใช้ฟอร์มเปอร์ตี้ **elements** นี้ร่วมกับ **index** ซึ่งชี้ไปยังอะเรย์ของอิลิเมนต์ที่อยู่ในฟอร์มเช่น **Checkbox**, **Text** และ **Button** เป็นต้น โดยมีรูปแบบการใช้งานดังนี้

```
formName.elements[ index ] และ
```

```
formName.elements.value
```

ในแบบแรกใช้เพื่อเข้าถึงอิลิเมนต์ที่อยู่ในฟอร์ม ซึ่งเราจะสามารถเรียกใช้ฟอร์มเปอร์ตี้ของอิลิเมนต์และเมทอดได้ เช่น ฟอร์ม `frmInfo` มีอ็อบเจกต์ที่เก็บข้อมูลอื่นหนึ่ง เราสามารถใช้ค่าของเท็กซ์ได้โดยเรียก `frmInfo.elements[0].value` ส่วนการใช้ฟอร์มเปอร์ตี้ของอิลิเมนต์ในแบบที่สองจะมีค่าเท่ากับจำนวนอิลิเมนต์ที่อยู่ในฟอร์ม เช่น ฟอร์ม `frmOrder` มี 4 อิลิเมนต์ จะมีค่า `frmOrder.elements.length` เท่ากับ 4

- **encoding** ฟอร์มเปอร์ตี้นี้จะใช้อ่านค่าที่กำหนดในแอตทริบิวต์ **Encoding** ของอ็อบเจกต์ฟอร์ม
- **length** ฟอร์มเปอร์ตี้นี้จะเก็บค่าจำนวนอิลิเมนต์ที่สร้างอยู่บนฟอร์มอ็อบเจกต์
- **name** ฟอร์มเปอร์ตี้นี้จะใช้อ่านค่าที่กำหนดในแอตทริบิวต์ **Name** ของอ็อบเจกต์ฟอร์ม
- **Method** ฟอร์มเปอร์ตี้จะใช้อ่านค่าที่กำหนดในแอตทริบิวต์ **Method** ของอ็อบเจกต์ฟอร์ม
- **target** ฟอร์มเปอร์ตี้จะใช้อ่านค่าที่กำหนดในแอตทริบิวต์ **Target** ของอ็อบเจกต์ฟอร์ม

4.12.3 เมทอดของอ็อบเจกต์ฟอร์ม

Submit เมทอดนี้ใช้สำหรับขั้บมิตฟอร์ม ซึ่งทำหน้าที่เดียวกับบัทตอน **Submit** คือส่งข้อมูลกลับไปให้เซิร์ฟเวอร์โดยที่ไม่ต้องกำหนดบัทตอน **Submit** ไว้บนฟอร์ม

4.12.4 Event Handlers ของอ็อบเจกต์ฟอร์ม

onSubmit ใช้ดักจับเหตุการณ์ **Submit** ว่าเมื่อมีการคลิกที่บัทตอน **Submit** แล้วจะต้องเรียกโค้ด **JavaScript** ไคให้มาทำงานเช่น ตรวจสอบข้อมูลบนฟอร์มก่อนที่จะมีการส่งข้อมูลไปให้เซิร์ฟเวอร์ เป็นต้น

4.12.5 อิลิเมนต์ต่าง ๆ บนฟอร์ม

- button เป็นปุ่มคำสั่งอื่น ๆ นอกเหนือจาก ปุ่ม Reset และ Submit
- Submit เป็นปุ่มคำสั่งเพื่อส่งข้อมูลออกจากฟอร์มไปยังเซิร์ฟเวอร์
- reset เป็นปุ่มคำสั่งเคลียร์ข้อมูลในฟอร์มให้เป็นเหมือนฟอร์มที่ยังไม่ถูกใช้งาน
- text เป็นกรอบรับข้อความหรือเท็กซ์บ็อกซ์ (Text Box)
- password เป็นกรอบรับข้อความแบบรหัสผ่าน
- hidden เป็นเท็กซ์บ็อกซ์ที่ถูกซ่อนจากบราวเซอร์ จะใช้เก็บข้อมูลต่าง ๆ ที่ไม่ต้องการให้ผู้ที่เปิดเว็บเพจเห็น เช่น จำนวนครั้งที่ผู้ใช้เข้าชมิตฟอร์ม แต่อย่างไรก็ตามข้อมูลในฟิลด์ hidden นี้จะหายไปเมื่อมีการ โหลดเพจใหม่
- radio button เป็นปุ่มให้ผู้ใช้ได้เลือกซึ่งสามารถเลือกได้เพียงปุ่มเดียวเท่านั้นในแต่ละหัวข้อใด ๆ
- checkbox เป็นปุ่มให้ผู้ใช้ได้เลือกซึ่งสามารถเลือกได้มากกว่าหนึ่งในแต่ละหัวข้อ ๆ
- select list เป็นกรือปลาดาวน์ลิสต์ (Dropdown List) หรือลิสต์บ็อกซ์ (List Box)
- textarea เป็นกรอบรับข้อความหลาย ๆ บรรทัด

4.12.6 การสร้างอิลิเมนต์ต่าง ๆ บนฟอร์ม

การสร้างอิลิเมนต์บนฟอร์ม ส่วนใหญ่แล้วจะเขียนภายในแท็ก <INPUT> ซึ่งมีแอตทริบิวต์บังคับที่ต้องเขียน 2 ตัว คือ TYPE เพื่อบอกชนิดของอิลิเมนต์ และ NAME เพื่อบอกชื่ออิลิเมนต์นั้น ๆ เหมือนกับการตั้งชื่อตัวแปร

4.12.6.1 Button, Submit และ Reset

นอกจากแอตทริบิวต์บังคับแล้วสามารถเขียนแอตทริบิวต์ VALUE เพื่อกำหนดข้อความที่จะปรากฏในปุ่มคำสั่ง เช่น

```
<INPUT TYPE="BUTTON" NAME ="BUTTON1" VALUE="OK">
```

4.12.6.2 Text, Password และ Hidden

นอกจากแอตทริบิวต์บังคับแล้ว สามารถเขียนแอตทริบิวต์เพิ่มดังนี้

- VALUE เพื่อกำหนดข้อความที่จะให้ปรากฏบนกรอบรับข้อความ
- SIZE เพื่อกำหนดความกว้างของกรอบรับข้อความ
- MAXLENGTH เพื่อกำหนดความยาวสูงสุดของข้อความ

เช่น

```
<INPUT TYPE="TEXT" NAME="TxtBox1" VALUE="Gaewrasameekul">
```

```
<INPUT TYPE="TEXT" NAME="TxtBox2" SIZE=30 MAXLENGTH=30>
```

```
<INPUT TYPE="PASSWORD" NAME="ID" MAXLENGTH=8>
```

4.12.6.3 Radio

นอกจากแอตทริบิวต์บังคับแล้วมีแอตทริบิวต์เพิ่มดังนี้

- VALUE เพื่อกำหนดค่าที่จะใช้อ้างอิงว่าเรดิโอบัททอนตัวใดถูกเลือก
- CHECKED เพื่อกำหนดให้เรดิโอบัททอนถูกเลือกเป็นค่าโดยปริยาย

เช่น

```
<INPUT TYPE="CHECKBOX" NAME="Aoe" VALUE="OK" CHECKED>
```

```
<INPUT TYPE="CHECKBOX" NAME="C&C" VALUE="OK" CHECKED>
```

```
<INPUT TYPE="CHECKBOX" NAME="Warcraft" VALUE="OK" CHECKED>
```

4.12.6.4 Select

การสร้างอิลิเมนต์ Select นี้จะเขียนภายในระหว่างแท็ก <SELECT> และ </SELECT> ซึ่งมีแอตทริบิวต์ดังนี้

- NAME เพื่อบอกชื่อของอิลิเมนต์ Select
- SIZE เพื่อกำหนดจำนวนลิสต์ที่จะแสดงในกรอบลิสต์บ็อกซ์ และภายในแท็ก <SELECT> นี้จะเขียนแท็ก <OPTION> เพื่อสร้างหัวข้อของลิสต์ทั้งหมดซึ่งแอตทริบิวต์ของแท็ก <OPTION> มีดังนี้

- VALUE เพื่อกำหนดหัวข้อที่จะใช้อ้างอิงว่าหัวข้อลิสต์ใดถูกเลือก
- SELECTED เพื่อกำหนดให้หัวข้อลิสต์นั้นถูกเลือกเป็นค่าปริยาย (Default)

เช่น

```
<SELECT NAME="ListSite" SIZE=4>
```

```
<OPTION VALUE="www.Microsoft.com">Microsoft
```

```
<OPTION VALUE="www.cnn.com">CNN
```

```
<OPTION VALUE="www.netscape.com" SELECTED> Netscape
```

```
<OPTION VALUE="www.yahoo.com">Yahoo
```

```
<OPTION VALUE="www.javasoft.com">JavaSoft
```

```
</SELECT>
```

4.12.6.5 Textarea

การสร้างอิลิเมนต์ Textarea นี้จะเขียนภายในแท็ก <TEXTAREA> และ </TEXTAREA> ซึ่งมีแอตทริบิวต์ดังนี้

- NAME เพื่อบอกชื่อของอิลิเมนต์ Textarea
- ROWS เพื่อกำหนดจำนวนแถวของเท็กซ์บ็อกซ์
- COLS เพื่อกำหนดจำนวนคอลัมน์ของเท็กซ์บ็อกซ์

- WRAP เพื่อกำหนดวิธีการปิดบรรทัด (Virtual หรือ Physical)

ส่วนการกำหนดข้อความที่จะให้ปรากฏบน Textarea ก็สามารถทำได้โดยเขียนข้อความแทรกในระหว่างแท็ก <TEXTAREA> และ </TEXTAREA>

เช่น <TEXTAREA NAME="Comment" ROWS=4 COLS=40>

Comment Text Hear

</TEXTAREA>

4.12.7 Properties, Methods และ Event Handlers ของอิลิเมนต์ต่าง ๆ

Elements	Properties	Methods	Event Handlers
button submit reset	- name (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ NAME) - value (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ VALUE)	click ()	onClick
text textarea	- default Value (ใช้อ่านค่าปริยายที่กำหนดในแอตทริบิวต์ VALUE) - name (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ NAME) - value (ใช้อ่านค่าปัจจุบันที่ปรากฏในอิลิเมนต์)	focus () blur () select ()	onBlur onFocus onChange onSelect
hidden	- name (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ NAME) - value (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ VALUE)	ไม่มี	ไม่มี
radio	- checked (ใช้อ่านค่า Status ปัจจุบันของ radio) - default Checked (ใช้อ่านค่าปริยายของ radio) - Index (ใช้อ่านค่า index ของเรดิโอบัททอนที่ถูกเลือก) - length (ใช้อ่านจำนวนของเรดิโอบัททอนในกลุ่ม) - name (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ NAME) - value (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ VALUE)	click ()	onClick
checkbox	- checked (ใช้อ่านค่า Status ปัจจุบันของ Checkbox) - default Checked (ใช้อ่านค่าปริยายของ Checkbox) - name (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ NAME) - value (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ VALUE)	click ()	onClick
Select	- selected Index (ใช้อ่านค่า index ปัจจุบันที่ถูกเลือก) - default Selected (ใช้อ่านค่าที่ถูกเลือกเป็น default ในแท็ก <OPTION>)	ไม่มี	onBlur onFocus onChange

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<ul style="list-style-type: none"> - index (ใช้อ่านค่า index ที่ถูกเลือกใน OPTION) - Selected (ใช้อ่านค่า Status ปัจจุบันของ OPTION) - text (ใช้อ่านค่าของข้อความที่แสดงในเมนูของ OPTION ที่ถูกเลือก) - value (ใช้อ่านค่าที่กำหนดในแอตทริบิวต์ VALUE ของแท็ก <OPTION>) 		
--	--	--	--



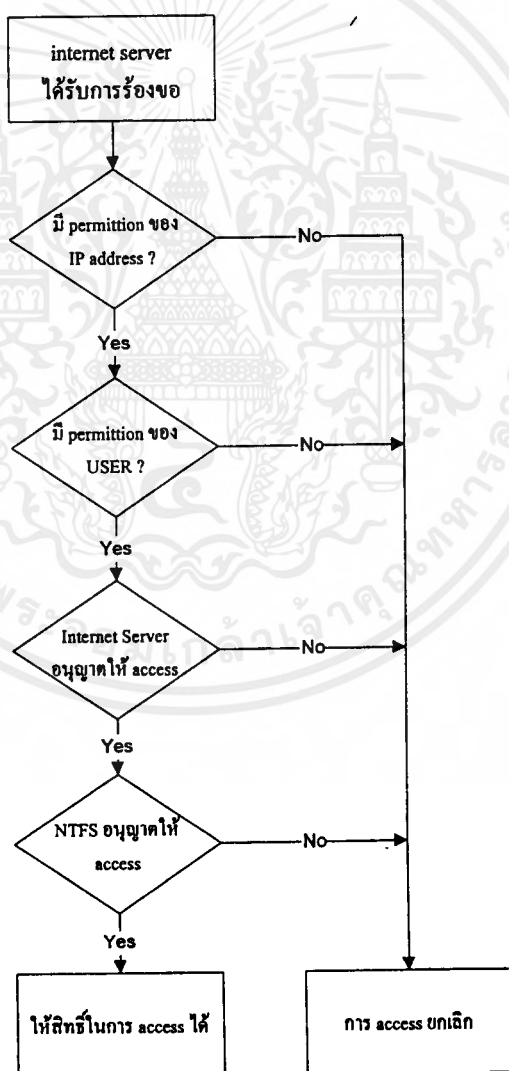
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ระบบรักษาความปลอดภัยบน Internet

5.1 การป้องกันผู้บุกรุก

ในการจัดการกับ web server ซึ่งมีการเข้า access เป็นจำนวนมากโดยผ่านทาง internet หรือการที่จะบำรุงรักษาเอกสารข้อมูลต่างๆ บนเครือข่าย internet นั้น เรื่องระบบรักษาความปลอดภัย (security) เป็นเรื่องสำคัญที่จะต้องนำมาพิจารณา ระบบปฏิบัติการ windows NT ถูกออกแบบมาเพื่อช่วยป้องกันระบบจากผู้บุกรุกได้ Internet Information Server ได้ถูกสร้างขึ้นตาม windows NT security model และได้เตรียมเรื่องของการตรวจสอบ (monitoring) และคุณสมบัติด้านความปลอดภัยลงไปด้วย



แสดงขั้นตอนกระบวนการรักษาความปลอดภัยของ IIS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.1 Internet Information Sever Security ทำงานอย่างไร

IIS ได้ถูกสร้างขึ้นตาม windows NT security model ซึ่ง windows NT security นี้จะช่วยป้องกันคอมพิวเตอร์และทรัพยากร (resource) ต่างๆของมัน โดยการให้ User account และ Passwords เราสามารถที่จะควบคุมการ access resource ของคอมพิวเตอร์ โดยการจำกัด user ที่มี account ที่ถูกต้องเท่านั้น นอกจากนี้ยังสามารถที่จะใช้ Windows NT file System (NTFS) เพื่อจัดการเรื่องของการอนุญาตให้ใช้ file และ folders บนคอมพิวเตอร์ได้ และยังสามารถควบคุมการ access folders และ files โดยสามารถที่จะป้องกันการ copy files ของ user จาก folders หรือป้องกันการเข้ามา execute files ที่อยู่ใน folders ได้

IIS สนับสนุนโปรโตคอล SSL (Secure Socket Layer) ซึ่งเป็นการรักษาความปลอดภัยโดยเข้ารหัสข้อมูลเมื่อมีการส่งข้อมูลระหว่าง client กับ server เมื่อ IIS web server ได้รับความร้องขอ (request) information จาก browser มันจะตัดสินใจว่าการร้องขอนั้นถูกต้องหรือไม่ ซึ่งตามรูปคือรูปแบบง่าย ๆ ของกระบวนการรักษาความปลอดภัย ที่ถูกใช้เมื่อมีการร้องขอเข้ามา

5.1.2 การควบคุมการ access แบบ Anonymous

web server จำนวนมากที่มีบริการ www,ftp,gopher เกือบทั้งหมดจะมีการ access แบบ Anonymous ซึ่งก็คือ client จะร้องขอโดยไม่ต้องมี username และ password มาเลยดังเช่น กรณีของ

- FTP logon โดยใช้ username เป็น Anonymous
- การร้องขอของบริการ gopher ทั้งหมด
- web browser ที่ไม่ต้องมีการใส่ username และ password ลงใน HTTP header

ถึงแม้ว่า user ไม่ได้ logon ด้วย username และ password ก็จริงแต่เราก็สามารถที่จะยังคงควบคุมและ monitor การ access แบบ Anonymous ได้ เมื่อการร้องขอของ Anonymous ถูกรับ บริการ "impersorate" จะจัดให้ user นั้นเป็นแบบ Anonymous logon การร้องขอจะประสบความสำเร็จได้ ถ้า user ที่ logon แบบ Anonymous นั้นได้มีการกำหนด permission ให้ access resource ที่ร้องขอมานั้นได้ ซึ่งจะตัดสินใจโดย resource's Access Control (ACL) ถ้าไม่ได้มีการกำหนด permission ให้กับ user ที่ logon แบบ Anonymous การร้องขอนั้นจะ fail ซึ่งเราสามารถที่จะจัดให้บริการ www ตอบสนองการ fail ของ Anonymous โดยต้องให้ user ส่ง username และ password มาด้วยเพื่อมาตรวจสอบก่อน ซึ่งกระบวนการนี้เรียกว่า Authentication

การ logon แบบ Anonymous นั้น user account จำเป็นที่จะต้องถูกต้องตาม user account ของ windows NT บน server ที่ได้เตรียมบริการ webservice ด้วย และ password จำเป็นที่จะต้อง match กับ password ของ user ในฐานข้อมูลของคอมพิวเตอร์ของ user ซึ่ง user account และ password สามารถที่จะจัดการได้ใน windows NT user manager โดยตั้งค่า user rights ในเมนู Pollicies ซึ่ง user account ที่ logon แบบ Anonymous จำเป็นที่จะต้องเป็น logon locally

account ชื่อ IUSR_computername จะถูกสร้างโดยอัตโนมัติ (โดยการ random password) ระหว่างที่ IIS setup ตัวอย่างเช่น ถ้า computer name คือ marketing1 หลังจาก access แบบ Anonymous แล้ว account name จะเป็น IUSR_marketing1 โดย default แล้ว webclient ทั้งหมดจะใช้ account นี้

ในบริการแบบ www และ ftp เราสามารถอนุญาตหรือ ป้องกันการ access แบบ Anonymous (ใน gopher นั้นการร้องขอเป็นแบบ Anonymous) และเปลี่ยน password สำหรับ account นั้นได้ด้วย

5.1.3 การควบคุมการ access แบบ Authentication

website นั้นสามารถที่จะจำกัดให้ access ได้เพียงแบบ Authentication ก็ได้ นั่นคือ webclient ต้องส่ง username และ password ที่ถูกต้องมาด้วย password Authentication มีประโยชน์ในกรณีที่ต้องการให้เพียงบุคคลบางบุคคลเข้ามา access website ของเรา การ access นั้นสามารถที่จะมีทั้งการ access แบบ Authenticate และ Anonymous logon ได้ในเวลาเดียวกัน

บริการ www ได้เตรียมรูปแบบของ Authentication ได้ 2 รูปแบบ ซึ่งมีแบบ basic และแบบ windows NT Challenge/Response (บางครั้งจะถูกอ้างถึง NTLM)

แบบ Basic Authentication นั้นจะไม่มี การเข้ารหัสในการส่งผ่านข้อมูลระหว่าง client และ server เพราะ Basic Authentication ส่ง windows NT username และ password ของ client จะไม่มี การเข้ารหัสที่จำเป็นบน networks ฉะนั้นก็จะง่ายต่อการที่ผู้บุกรุกจะรู้ username และ password ได้

แบบ windows NT Challenge/Response Authentication จะมีการป้องกัน password และได้เตรียมเรื่องของความปลอดภัยในการ logon บน network ใน windows NT Challenge/Response Authentication นั้น user account จะถูกรับเข้ามาจาก client ซึ่งก็คือ user ใ้ดำเนินการ logon ที่คอมพิวเตอร์ client เพราะว่า account นี้ได้รวม windows NT domain ของมันเข้าไป ดังนั้นต้องเป็น account ที่ถูกต้องบน windows NT server ที่ทำการรัน IIS อยู่

โดย default แล้วจะมีทั้ง basic และ windows NT Challenge/Response Authentication ถ้า browser สนับสนุน windows NT Challenge/Response มันก็จะใช้ Authentication method แบบนั้น นอกเหนือจากนี้แล้วจะใช้แบบ Basic Authentication

5.1.4 Anonymous logon และ Client Authentication ทำงานอย่างไร

ในบริการ www นั้นเมื่อได้รับการร้องขอจาก client ที่มีการส่ง username และ password มาแล้วนั้น "Anonymous logon" user account จะไม่ถูกใช้ในกระบวนการร้องขอ ถ้าการบริการไม่ได้ถูก grant permission ในการ access resource ที่ทำการร้องขอในขณะที่ใช้ username และ password มาด้วยนั้น การร้องขอนั้นจะ fails และจะส่งข้อความแจ้งการผิดพลาดกลับไปฝั่ง client

เมื่อการร้องขอแบบ Anonymous เกิด fail เพราะว่า "Anonymous logon" user account ไม่ได้มี permission ที่จะมา access resource ที่ต้องการและบริการ www สนับสนุนระบบ Authentication ด้วยแล้ว ถ้าการร้องขอนั้นมีการบ่งบอกว่า client ที่บริการอยู่นั้นมีการสนับสนุน HTTP Basic Authentication ด้วยแล้ว web browser ส่วนมากจะแสดง dialog box ที่ใช้กรอก username และ password และทำการส่งการร้องขอแบบ Anonymous ใหม่ซึ่งจะมีการส่ง username และ password ที่ user ได้ใส่เข้ามาให้ด้วย

ถ้า web browser สนับสนุนโปรโตคอล windows NT Challenge/Response Authentication และ บริการ www ได้ถูกจัดโครงสร้างให้สนับสนุนโปรโตคอลนี้ เมื่อการร้องขอแบบ Anonymous www นั้น

fail เนื่องจากไม่ได้มี permission ก็จะส่งผลให้มีการใช้โปรโตคอล windows NT Challenge/Response Authentication โดยอัตโนมัติ browser จะส่ง username และเข้ารหัส password จาก client ที่บริการอยู่ และ client ที่ร้องขอจะถูกประมวลผลอีกครั้งโดยใช้ user Information ของ client

ถ้าบริการ www ถูกจัดโครงสร้างมาให้สนับสนุนทั้ง Basic และ windows NT Challenge/Response แล้ว web server จะส่ง Authentication methods ทั้งคู่ลงใน header มายัง web browser และถ้า browser สนับสนุนโปรโตคอล windows NT Challenge/Response ก็จะใช้โปรโตคอลนี้ แต่ถ้าไม่สนับสนุนก็จะไปใช้โปรโตคอล Basic Authentication แทน

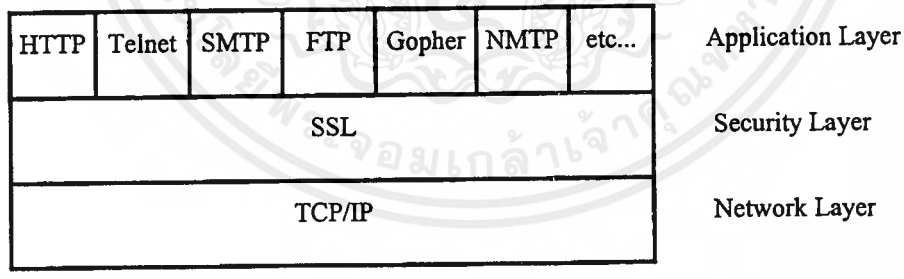
5.2 การเข้ารหัสบน Web

การเข้ารหัสบน web นั้นมีวิธีการมากมาย แต่ก็มีโปรโตคอล 2 แบบที่นิยมใช้กันใน web browser ซึ่งก็คือ Secure Socket Layer (SSL) และ Secure Hypertext Transfer Protocol (S-HTTP)

5.2.1 Secure Socket Layer (SSL)

IIS ได้เสนอโปรโตคอลสำหรับการรักษาความปลอดภัยของข้อมูลใน layer ที่อยู่ระหว่าง service protocol (HTTP) และ (TCP/IP) โปรโตคอลรักษาความปลอดภัยนี้เรียกว่า Secure Socket Layer (SSL) ซึ่งได้มีการเตรียมเรื่องของ การเข้ารหัสข้อมูล (data encryption), การยืนยันผู้ส่งผู้รับ (Server Authentication) และความถูกต้องของข้อมูล (Message integrity) สำหรับการเชื่อมต่อแบบ TCP/IP

SSL นั้นถูกออกแบบให้อยู่ในส่วนที่ติดจาก Application Layer ซึ่งมีบริการ HTTP ,telnet ,SMTP ,FTP ,gopher และ NMTP แต่อยู่ก่อนขั้นของการเชื่อมต่อแบบ TCP/IP ดังรูป



ลำดับชั้นของการเชื่อมต่อ

SSL นั้นจะเข้ารหัสที่ชั้น network layer และลักษณะการเข้ารหัสนั้นจะใช้เทคโนโลยีการเข้ารหัสแบบกุญแจ (key) 2 ดอกเป็นหลักซึ่งก็คือจะมีกุญแจดอกหนึ่งไว้เข้ารหัสและอีกดอกหนึ่งไว้ถอดรหัส ซึ่งกุญแจในการเข้ารหัสนั้นจะเป็นแบบ กุญแจสาธารณะ (Public key) โดยเป็นกุญแจที่จะประกาศให้คนรู้ได้ และถึงแม้จะตกถึงใครก็ไม่สามารถไขดูเอกสารได้ ส่วนกุญแจที่ใช้ในการถอดรหัสนั้นเรียกว่า กุญแจส่วนตัว (Private key) ซึ่งจะปิดเป็นความลับและรู้เฉพาะผู้รับเพียงผู้เดียวเท่านั้น ตัวอย่างของวิธีเข้ารหัสแบบนี้เช่น การเข้ารหัสแบบ RSA เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการทำงานของการเข้ารหัสแบบกุญแจ 2 ดอกคือ ในการส่งเอกสารนั้นถ้าผู้ใช้ต้องการให้ใครส่งเอกสารมาให้แบบเป็นความลับ ผู้ใช้จะสร้างกุญแจ 2 ดอกตามที่กล่าวมาแล้วส่งกุญแจสาธารณะไปให้ผู้ที่ส่งข้อมูลมาไว้เพื่อใช้กุญแจนี้ในการเข้ารหัส เพิ่มที่ถูกเข้ารหัสแล้ว เรียกว่า ไซเฟอร์ (Cypher) เมื่อเอกสารไซเฟอร์มาถึงผู้รับ ก็จะใช้กุญแจเฉพาะของเขาไขดูเอกสารนี้ เพราะฉะนั้นจะเห็นได้ว่ากุญแจเฉพาะนี้จะรู้เพียงคนเดียวจึงปลอดภัยกับข้อมูล

SSL จะส่งข้อมูลที่ไต่ทำการเข้ารหัสแล้วซึ่งมันจะช้ากว่าการส่งข้อมูลที่ไม่มีการเข้ารหัส ซึ่งอาจทำให้ performance ของ site เราแย่ลงได้ ฉะนั้นก็ควรจะพิจารณาใช้ SSL กับข้อมูลที่สำคัญมากๆ เช่น ข้อมูลของ Credit Card เป็นต้น

5.2.1.1 ขั้นตอนการทำ SSL Security บน Web Server

1. generate ไฟล์ key pair และไฟล์การร้องขอ (request file)
2. ขอ Certificate จาก Certificate Authority
3. ติดตั้ง Certificate บน server
4. ติดตั้ง SSL Security บน folder ที่บริการ www

สิ่งสำคัญ จำไว้ว่าข้อความที่กล่าวต่อไปนี้จะทำได้เมื่อทำการ enable SSL Security

- สามารถที่จะ enable SSL Security บน root ของ web site ได้ (โดย default แล้วก็คือ \InetPub\WWWroot)
- หรือบน virtual folder อื่นๆ
- เมื่อทำการ enable ครั้งหนึ่งและจัดการสิทธิ์ต่างๆ แล้วเพียงแต่ทำการ enable SSL บน client ก็จะสามารถติดต่อสื่อสารกับ www folder ที่ได้ enable SSL ได้
- URL ที่ชี้ไปยังเอกสารของ www folder ที่ enable SSL ไว้จะต้องใช้ "https://" แทนการใช้ "http://" ใน URL ซึ่งการ link โดยใช้ "http://" ใน URL นั้นจะไม่สามารถทำงานได้

5.2.1.2 การ Generate key pair

ในขั้นตอนของการ enable SSL security บน web server นั้นจะต้อง generate key pair และหลังจากนั้นจะได้ SSL Certificate มา มีขั้นตอนดังนี้

1. ในเมนูย่อยของ Microsoft Internet Server ให้คลิกที่ key manager หรือคลิกที่ ไอคอนบนทูลบาร์ของ Internet Service Manager
2. จากเมนู key คลิก Create New Key
3. ใน dialog box ชื่อ Create New Key and Certificate Request ให้เติมข้อมูลต่างๆเข้าไป มีดังนี้
 - keyname : ตั้งชื่อ key ที่จะทำการสร้าง
 - password : กำหนด password ที่จะเข้ารหัสกุญแจส่วนตัว (Private key)

- Bits : โดย default แล้ว key manager จะ generate key pair ความยาว 1024 bits ในการกำหนด key ให้มีความยาว 512 หรือ 768 bits ทำโดยเลือกใน box
 - organization : ให้ตีเทียบกับ International Organization for Standardization (ISO) ลงทะเบียนในระดับบน หรือองค์กร หรือชื่อบริษัท
 - Organizational Unit : แผนกในบริษัทเช่น Marketing
 - Common Name : domain name ของ server ตัวอย่างเช่น www.doraemon.com
 - Country : เป็นอักษรย่อของชื่อประเทศ ใช้ตัวย่อ 2 ตัวเช่น US,FR,AU,UK และอื่นๆ
 - State/Province : เช่น washington ,Alborta ,California และอื่นๆ
 - Locality : เมืองที่บริษัทของผู้ใช้ตั้งอยู่เช่น Redmond หรือ Toronto
 - Request file : พิมพ์ชื่อของ request file ที่จะทำการสร้าง
4. หลังจากเติมเสร็จแล้วออกจาก from คลิก ok
 5. พิมพ์ password ใหม่ตามที่ได้พิมพ์ใน from ไปแล้วและคลิก ok ไอคอนจะปรากฏขึ้นเหมือน key ได้ถูกสร้างขึ้นเมื่อ key ถูกสร้างแล้ว บนหน้าจอจะปรากฏ information เกี่ยวกับ key ใหม่และบอก ว่าทำอะไรจึงจะรับ Certificate
 6. หลังจากอ่าน New Key information แล้วคลิก ok
 7. ทำการ save New key จากเมนู server เลือก Commit Changes New
 8. ถ้าต้องการยืนยันการเปลี่ยนแปลงให้คลิก ok

ตอนนี้ key ก็จะปรากฏใน key Manager Windows ได้ชื่อของคอมพิวเตอร์สำหรับที่ซึ่งได้ทำการสร้าง key โดย default แล้ว key จะถูก generate บน local computer เมื่อได้ทำการ generate key pair แล้วก็จำเป็นที่จะต้องรับ Certificate และหลังจากนั้นก็ติดตั้ง Certificate ด้วย key pair สำหรับ information เกี่ยวกับการรับ Certificate จะกล่าวถึงในหัวข้อถัดไป

5.2.1.3 การได้รับ Certificate

key ที่ถูก generate โดย key manager จะไม่ถูกต้องสำหรับการใช้งานจนกว่าจะได้รับ Certificate ที่ถูกต้องสำหรับมันจาก Certificate Authority เช่น Verisign ส่งไฟล์การร้องขอ Certificate ไปยัง Certificate Authority เพื่อ รับ Certificate ที่ถูกต้อง เมื่อได้ทำการร้องขอแล้ว key จะมียู่บน host computer แต่ไม่สามารถที่จะใช้ได้ สำหรับคำสั่งที่ได้รับมานั้น Verisign Certificate จะอ้างอิงได้ที่ Verisign's Web site "<http://www.verisign.com/microsoft/>"

5.2.1.4 การติดตั้ง Certificate ด้วย key pair

หลังจากที่การร้องขอ Certificate เสร็จสมบูรณ์แล้ว ก็จะได้รับ Signed Certificate จาก Certificate Authority โปรแกรม key manager จะสร้าง file เหมือนกับตัวอย่างข้างล่าง

-----BEGIN CERTIFICATE-----

IEBDSCEXoCHQEWLQMJSOZILvoNVQECSQAwcSETMRkOAMUTBhMuVrM
mIoAnBdNVBAoTF1JTQSBEYXRhIFNlY3VyaXR5LCBjb250aWZlbnRzZmljYXRIMSQwIgwYDlVQDEExtPcGVuIEIhc
QLExNQZXJzb25hIENlcnRpZmljYXRIMSQwIgwYDlVQDEExtPcGVuIEIhc
mtldCBUZXR0IFNlcnZlciAxMTAwHhcNOTUwNzE5MjAyNzZmWWhcNOTYw
NTE0MjAyOTEwWjBzMQswCQYDlVQDEExtPcGVuIEIhc

-----END CERTIFICATE-----

การติดตั้ง Certificate

1. ในกลุ่มโปรแกรม Internet Server คลิกที่ key manager
2. ในวินโดว์ key manager ให้เลือก key pair ที่ match กับ Signed Certificate
3. จากเมนู key ให้เลือก Install Key Certificate
4. เลือก Certificate file จาก list และคลิก open
5. ให้พิมพ์ password ที่ใช้ในการสร้าง key pair ซึ่ง key และ Certificate จะถูกนำมารวมกันและเก็บลงใน registry ของ server
6. จากเมนูของ server เลือก Commit Change New
7. หลังจากนั้นถ้าต้องการยืนยันการเปลี่ยนแปลงทั้งหมด คลิก ok

ถ้าไม่ได้ระบุ IP Address ในขณะที่กำลังติดตั้ง Certificate แล้ว Certificate ที่เหมือนกันจะถูกใช้กับ virtual server ทั้งหมดที่สร้างขึ้น ถ้ามีหลายๆ site บน server เดียวก็สามารถที่ระบุ ให้ Certificate นั้นใช้เพียง Address พิเศษโดยเฉพาะได้โดยการเพิ่ม (adding) IP Address นั้นเข้าไป

5.2.1.5 การจัดการกับ Directory ให้มี SSL

เมื่อได้จัดการเกี่ยวกับ Certificate แล้วก็จำเป็นที่จะต้อง enable คุณลักษณะ (feature) ต่างๆ ของ SSL จาก Internet Service Manager ลงไป SSL นั้นสามารถนำไปใช้กับ virtual folder ที่อยู่ใน web site ได้

การกำหนด SSL

1. ใน Internet Service Manager ให้ดับเบิลคลิกที่ www service ที่แสดง property sheets ของมัน หลังจากนั้นคลิกที่แท็บ Directories
2. เลือก folder ที่ต้องการให้มี SSL หลังจากนั้นคลิก Edit Properties
3. เลือก option ชื่อ Require Secure SSL Channel และคลิก ok

5.2.2 Secure Hypertext Transfer Protocol (S-HTTP)

เป็นมาตรฐานการรักษาความปลอดภัยของข้อมูลอีกแบบหนึ่งของบริการ www ซึ่งเสนอขึ้นโดย EIT (Enterprise Integration Technologies) โดยจะมีการเข้ารหัสข้อมูล (encrypted) S-HTTP นั้นจะเข้ารหัสที่ application layer และจะใช้การเข้ารหัสได้สองแบบคือ DES (Data Encryption Standard) และ RSA (Rivest Shamir Alderman) ซึ่ง DES นั้นเป็นการเข้ารหัสโดยใช้กุญแจดอกเดียวกันคือ ผู้ส่งไขกุญแจเปิด ผู้รับนำกุญแจดอกเดียวกันมาไขเปิดอ่าน ส่วน RSA นั้นใช้วิธีการเข้ารหัสแบบกุญแจสองดอกตามที่ได้อธิบายไปแล้ว

ในการใช้งานให้ระบุเป็น shttp ในส่วนของ anchor ชุดของข้อมูล http ที่ต้องการรักษาความลับนั้นจะถูก encapsulate ลงในข้อความของโปรโตคอล S-HTTP โดยจะมีหัวเรื่องหรือไบบะหน้าที่ข้อความต้นฉบับโดย

- ระบุลักษณะของการตอบรับ
- ระบุประเภทของการเข้ารหัสว่าใช้วิธีการใดและพารามิเตอร์
- ระบุกุญแจ (key)
- ระบุเกณฑ์การรับรอง (Authentication) สำหรับการแลกเปลี่ยนข้อมูล
- ระบุรายละเอียดใน shttp anchor ว่าเป็น DN ,NONCE,CRYPTOPTS

ตัวอย่างชุดของ Heading ของ S-HTTP สำหรับการขอแบบ GET

Secure * Secure-HTTP/1.1

Content-Privacy-Domain: PKCS-7

Content-Transfer-Encode: 7BIT

FSAJ;JFGJGAiuyJGiJFSDAOPJTRDui;FJOPSiTJGMFVACMV/VMG;AJGG/
FDGJ;/GHJD;SIRU[JI466IGfdf;BMH7J;MG;'GKJS;'J0SJG;H-SHKL;SJG-fgty
DS;LGJESP[JD;GHJI-E0Y9J435=-0I04EJY;SJGIUPETY=QGJFTUTYJGV'/Af

เมื่อมีการร้องขอจาก ผู้ขอใช้บริการ (client) ไปยัง ผู้ให้บริการ (server) โดยผ่านทาง URL จากนั้นทาง server ก็จะส่งข้อมูลที่ถูกรับมานั้นไปให้ผู้รับ ซึ่งผู้รับจะทราบลักษณะของข้อมูลได้จากส่วนที่เป็น header ซึ่งเป็นส่วนที่ระบุ Content type นั่นเอง ซึ่งเพิ่มข้อมูลในลักษณะ http นั้นส่วนของ Content type จะเป็น application/http และ Content type ของข้อความที่มีการเข้ารหัสโดย S-HTTP จะเป็น application/shttp

บทที่ 6

ISAPI (Internet Server Application Programming Interface)

6.1 ISAPI

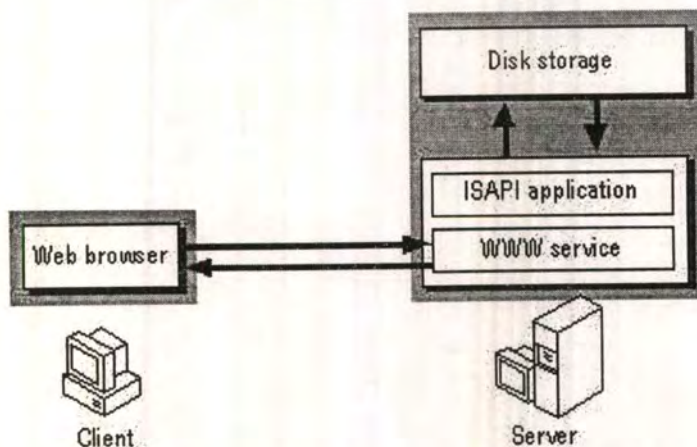
ISAPI คือ การให้บริการของ HTTP Server หรือผู้ให้บริการ Web แก่การร้องขอของ Client เช่น โปรแกรม Web Browser ในรูปแบบของการสั่งงานให้ Server ทำงาน (Run) หรือปฏิบัติงาน Application หรือ โปรแกรมเพื่อจุดประสงค์ต่างๆ ที่ไม่คงที่ (Dynamic) เช่น การเรียกค้นรายการจากฐานข้อมูล การสั่งซื้อสินค้า การลงทะเบียนนักศึกษา เป็นต้น ซึ่งจะแตกต่างจากการให้บริการแบบปรกติ (Static) เช่นการให้บริการเฉพาะเอกสาร Home Page (HTML) แบบธรรมดาทั่วไป

ISAPI ทำงานในลักษณะเดียวกับ CGI (Common Gateway Interface) แต่ได้ปรับปรุงข้อด้อยต่างๆ ที่พบใน CGI และเพิ่มความสามารถใหม่ๆ ที่ CGI ไม่เคยมีมาก่อน

6.2 การทำงานของ ISAPI

โดยพื้นฐานแล้ว ISAPI ทำงานเหมือนกับ CGI เกือบทุกประการ เช่น การกรอกแบบฟอร์ม ซึ่งปรากฏอยู่บนหน้า Home Page ของ Web Browser ข้อมูลที่ผู้ใช้กรอกเข้าไปในแบบฟอร์มจะถูกส่งจากฝั่ง Client ผ่านเครือข่าย Internet ไปสู่ Web Server ที่มีโปรแกรม CGI หรือ ISAPI ที่คอยประมวลผลข้อมูลเหล่านั้นอยู่ เช่น การนำข้อมูลที่ได้จากแบบฟอร์มมาเป็น KEY ในการค้นหาข้อมูลบางอย่างบนฐานข้อมูล (Database Server) เมื่อได้ผลลัพธ์จากฐานข้อมูลเรียบร้อยแล้ว โปรแกรม CGI หรือ ISAPI ก็จะแปลงข้อมูลที่ได้รับกลับมาอยู่ในรูปแบบของเอกสาร HTML (Hyper Text Makeup Language) เพื่อส่งกลับไปยัง Web Browser ของผู้ให้บริการ

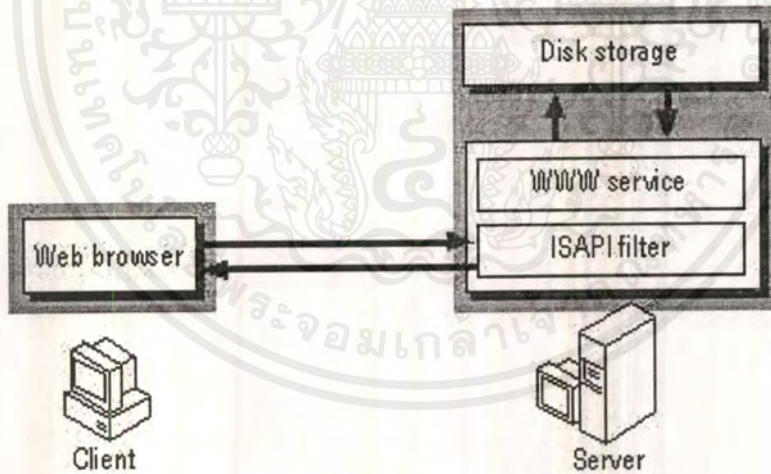
โปรแกรมที่ได้ตอบกับ Web Client ซึ่งปฏิบัติงานอยู่บน Web Server นี้ในระบบของ ISAPI เรียกว่า "Extension" (Application) มีการทำงานดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของ ISAPI Extension จะทำงานอยู่บน Web Server ในรูปแบบของ DLL (Dynamic Link Library) การทำงานของ DLL มีวิธีการดังนี้คือ ในอดีตนั้น Function การทำงานต่างๆภายในโปรแกรมจะต้องถูก Compile และ Link กับ Object ที่สำคัญเป็นภาษาเครื่อง (Machine Code) ให้เสร็จก่อนการทำงานจริง ซึ่ง Function ทั้งหลายจะอยู่ตายตัว (Static) ภายในโปรแกรม ฉะนั้นเราจึงสามารถใช้ Function ได้จำกัดเฉพาะภายในโปรแกรมซึ่งเป็นการทำงานที่ไม่อ่อนตัว DLL ได้เข้ามาแก้ปัญหานี้โดยการเก็บ Function มาตรฐานต่างๆ ไว้ในรูปแบบของ Library เมื่อโปรแกรมหลักทำการโหลด DLL ที่ต้องการมาเก็บไว้ในหน่วยความจำ ทำให้สามารถ Link หรือเชื่อมต่อกับ Function ที่ต้องการได้ในขณะปฏิบัติงานจริง ในเมื่อ ISAPI ทำงานอยู่ในเนื้อที่หน่วยความจำของ Web Server ตลอดเวลาทำให้การทำงานเป็นไปด้วยความรวดเร็ว เมื่อมีการร้องขอจาก Client

ISAPI Filter ทำหน้าที่เป็นตัวกรองที่คอยควบคุมการทำงานของ Web Server กับการติดต่อในรูปแบบของ โพรโทคอล HTTP ทั้งก่อนและหลังการให้บริการ Web Server การกรองข้อมูลการติดต่อของ Filter นี้ ส่งผลโดยตรงต่อการรักษาความปลอดภัยของระบบ ISAPI ได้เข้ามายกระดับความสามารถให้กับโพรโทคอล HTTP ทำให้ผู้พัฒนาระบบสามารถเพิ่มคุณสมบัติใหม่ๆ ให้กับ HTTP ได้ เมื่อ Web Server เริ่มทำงาน ISAPI Filter ก็จะถูกโหลดให้ปฏิบัติงาน โดยทันทีเพื่อคอยตรวจสอบการติดต่อจาก Web Client ที่จะเกิดกับ Web Server

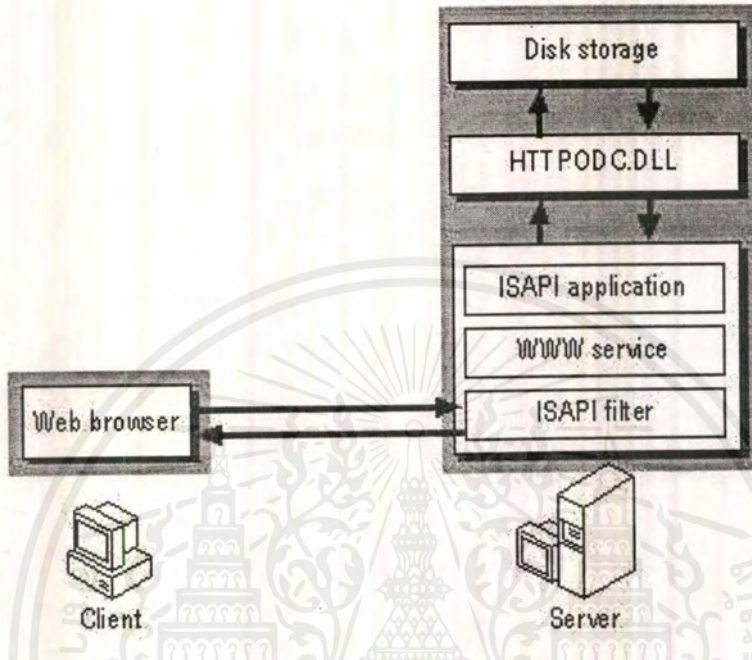


สรุประดับการใช้ ISAPI ก็จะมี Web Server สามารถใช้งาน ISAPI ได้ 2 แบบคือ

1. แบบ Extension ซึ่งมีหลักการทำงานเหมือน CGI ทุกประการ
2. แบบ Filter ซึ่งเป็นความสามารถใหม่ที่เพิ่มเข้ามา สำหรับควบคุมกลไกการทำงานของโพรโทคอล HTTP อย่างละเอียด (Fine - grained Control)

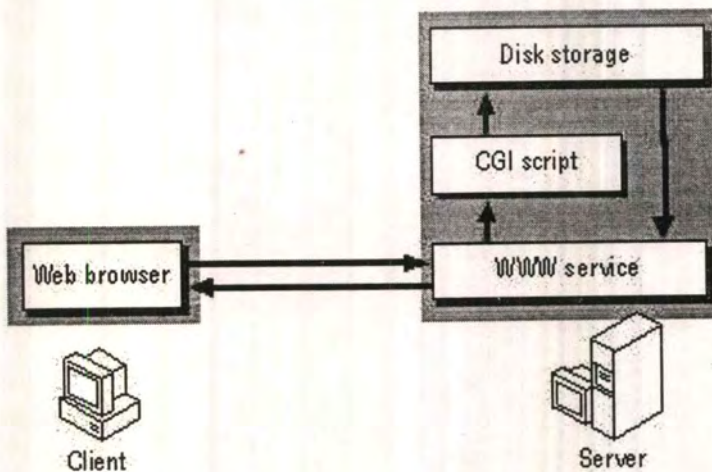
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถที่จะสร้าง Site ที่มีความซับซ้อนสูงๆ โดยใช้ ISAPI Filter และ ISAPI Application ,ISAPI Extension สามารถที่จะรวมเข้ากับ Internet Database Connector (IDC) เพื่อสร้าง Site ที่มี Interactive สูง (Response กับ Request) ได้ด้วย ดังรูป



6.3 Common Gateway Interface (CGI)

CGI เป็นชุดของ ข้อกำหนดสำหรับการผ่านของ Information ระหว่าง client Web browser และ Web server และ CGI application ในส่วนของ client Web browser สามารถ start ตัว CGI application โดยการใส่ลงใน HTML form หรือ คลิ๊กลิงค์ใน HTML page บน Web server ซึ่งการทำงานก็จะคล้ายกับของ ISAPI



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 ข้อเปรียบเทียบระหว่าง ISAPI กับ CGI

ทั้ง CGI และ ISAPI นั้นต่างก็มีข้อได้เปรียบเสียเปรียบตามการใช้งานที่แตกต่างกันไปดังนี้

6.4.1 ข้อได้เปรียบของ CGI

CGI ได้รับการยอมรับให้เป็นวิธีการมาตรฐาน สำหรับการสร้าง Web site ให้เป็นระบบตอบโต้ได้ (Interactive Web site) มานาน ในอดีต CGI ถือกำเนิดมาจากระบบปฏิบัติการ “UNIX” วัตถุประสงค์เบื้องต้นของ CGI ก็คือเพื่อตอบสนองต่อการร้องขอข้อมูลจาก Client ซึ่งข้อมูลนั้นไม่ได้ถูกเก็บในรูปแบบ HTML แต่อยู่ในรูปแบบ Application ซึ่งสามารถปฏิบัติงานต่างๆได้ ซึ่ง CGI Application ถูกเรียกให้ปฏิบัติงานในรูปแบบของ Process บนระบบปฏิบัติการโดยรับข้อมูลจาก Standard Input ในรูปแบบของ Environment Variable จาก Web Sever หลังจากการประมวลผลของ CGI Application แล้วข้อมูลผลลัพธ์จะถูกส่งไปยัง Web Server ผ่านทาง Standard Output เพื่อส่งกับไปยัง Client ต่อไป ซึ่งจะเป็นหนึ่ง Process ต่อการร้องขอ เมื่อมีการร้องขอหลายๆครั้งก็ต้องมีการสร้าง Process ใหม่ขึ้นมา

จะเห็นได้ว่าหลักการทำงานเหล่านี้เป็นพื้นฐานของระบบ UNIX ซึ่งเป็นระบบปฏิบัติการแบบหลายงาน (Preemptive Multitasking) และหลายผู้ใช้ (Multi - user) ได้ในเวลาเดียวกันอย่างไม่มีปัญหา ข้อมูลที่ได้รับทาง Standard Input จะเก็บอยู่ในรูปของ Environment Variable ฉะนั้นการพัฒนาและการแก้ไข (Debugging) CGI Application จึงทำได้ง่ายและยังมีลักษณะเด่นที่สำคัญอีก ดังต่อไปนี้

- การแยก Process การทำงาน

ในขณะที่ CGI Application หลายๆ ตัวกำลังปฏิบัติงานอยู่นั้น การใช้ Resource ของแต่ละ Process เช่นพื้นที่ของหน่วยความจำจะถูกแยกจากกันโดยเด็ดขาดแม้ว่า Process ใดๆ เกิด Hang-up ขึ้นมาก็ไม่ส่งผลอย่างไรต่อการทำงานของ Web Server หรือ Process อื่นๆ แต่อย่างไรส่งผลให้การทำงานของ Web Server ค่อนข้างมั่นคง

เมื่อเปรียบเทียบระหว่าง CGI กับ ISAPI Extension ที่ทำงานในรูปแบบของ DLL ซึ่งถูกโหลดให้ทำงานอยู่ตลอดเวลาในพื้นที่หน่วยความจำของ Web Server แม้ว่า ISAPI Extension จะมีข้อดีในแง่ของความเร็วในการปฏิบัติงานสูงแต่ถ้า Extension นั้นเกิดเสียบและค้างการทำงานขึ้นมา ก็จะส่งผลให้ Web Server ทั้งระบบหยุดการทำงานทันที

- การแก้ไขง่าย

การแก้ไข CGI Application ทำได้ค่อนข้างง่าย เพราะว่า CGI Application ทำงานแบบแยก Process จาก Web Server เราจึงสามารถแก้ไข Application โดยไม่มีการไปรบกวน การทำงานของ Web Server หรือ Process อื่นๆ เลย ซึ่ง ISAPI จะไม่สามารถทำได้ เพราะในขณะที่ DLL ถูกโหลดให้ทำงานในหน่วยความจำ ไฟล์ DLL นั้นจะถูกปิด LOCK ไม่สามารถเขียนทับหรือแก้ไขได้ ต้องหยุดการทำงานของ Server เพื่อให้ปล่อย Lock ไฟล์ DLL ก่อนจึงทำการแก้ไขได้

6.4.2 ข้อได้เปรียบของ ISAPI

เนื่องจาก ISAPI นั้นเป็นฟังก์ชัน API ที่เอาไว้ติดต่อกับ Internet Server ที่ชื่อ IIS ซึ่งแถมมากับระบบปฏิบัติการ Window NT 4.0 ซึ่งเป็นข้อได้เปรียบที่ทั้งระบบปฏิบัติการ Window NT 4.0 , IIS Web Server และการเชื่อมต่อ ISAPI เป็นมาตรฐานเดียวกัน เพราะมาจากบริษัท MicroSoft เหมือนกัน ทำให้การพัฒนาจะง่ายขึ้น ISAPI สามารถใช้ประโยชน์จากคุณสมบัติของ Web Server และระบบปฏิบัติการได้อย่างเต็มที่ และยังมีลักษณะเด่นที่สำคัญอีกดังต่อไปนี้

- ความเร็วในการตอบสนอง

ในแง่ของการตอบสนอง นั้น Home Page (HTML) ที่สร้างโดย ISAPI นั้นสามารถทำได้เร็วกว่า CGI เพราะว่า ISAPI Extension ถูกเรียกใช้ในรูปแบบของ DLL ซึ่งนั่นหมายถึง Extension ได้ใช้เนื้อที่ในหน่วยความจำที่เป็นส่วนเดียวกับ Web Server และรอให้บริการอยู่เสมอ คือเมื่อมีการร้องขอจาก Client ไปยัง Extension Web Server ก็เพียงหา Extension นั้นในหน่วยความจำ ถ้าพบว่าได้ถูกโหลดเอาไว้ในหน่วยความจำแล้ว Extension นั้นก็สามารถทำงานและสร้าง HomePage ได้เลย แต่ถ้าไม่พบก็จะ โหลดจาก HardDisk สู่หน่วยความจำแล้วปฏิบัติงานต่อไป ซึ่งจะเห็นได้ว่า โหลดครั้งเดียวแต่ใช้งานได้เป็นร้อยๆ พันๆ ครั้ง

- ลดภาระของ Server

สำหรับ CGI นั้นจุดเด่นที่ Application ที่ปฏิบัติงานแยกจาก Process ของ Server นี้ กลายเป็นข้อบกพร่องอย่างยิ่ง เมื่อมีการร้องขอจาก Client ในเวลาเดียวกันเป็นจำนวนมากๆ ทำให้ Web Server ต้องรับภาระในการสร้างและดูแล Process จำนวนมาก ทำให้ Web Server ต้องรับภาระในการดูแล Process มากยิ่งขึ้นซึ่งส่งผลกระทบต่อ

- ทรัพยากรของระบบถูกใช้มากขึ้น
- เวลาที่ใช้ในการตอบสนองมากขึ้นเพราะต้องเสียไปกับการสร้าง การดูแล และการทำลายแต่ละ Process

ดังนั้น Server จะมีภาระที่วุ่นวายมากในการจัดการกับทุก Process ซึ่ง ISAPI สามารถเอาชนะปัญหาเหล่านี้เป็นที่เรียบร้อยเพราะแต่ละการเชื่อมต่อของ Client สู่ Web Server เพื่อการใช้งาน ISAPI Extension Web Server ไม่มีความจำเป็นที่ต้องโหลด Extension ซ้ำซาก จาก Hard Disk

การตอบสนองต่อการร้องขอแต่ละอันทำได้โดยการแยก Thread หรือส่วนงานเล็กๆ ภายในหนึ่ง Process ของ ISAPI Application จุดนี้เองจึงเป็นประโยชน์ในการลดภาระของ Server เพราะสิ้นเปลืองทรัพยากรสำหรับหนึ่ง Process สำหรับหนึ่ง Extension เท่านั้น

6.5 การควบคุม Web Server อย่างละเอียด

ISAPI ทำให้เราสามารถควบคุมการทำงานของ Web Server ได้มากขึ้นกว่าการใช้ CGI โดยการ
ใช้ ISAPI Filter “ดัก” การทำงานทั้งก่อนและหลัง (Proprocessing and Postprocessing) ของการร้องขอ
ผ่านโปรโตคอล HTTP ได้ ซึ่งนั่นหมายความว่าผู้ดูแลระบบสามารถจัดการเหตุการณ์ (Event) ก่อนหน้า
และหลังจากการส่งงานให้ ISAPI Extension ปฏิบัติงานได้ เราสามารถใช้ ISAPI Filter ตรวจสอบสิทธิ์
ของการเข้าใช้ระบบเพื่อรับประกันความปลอดภัย ซึ่ง ISAPI Filter เป็นเสมือนปราการสำหรับตรวจสอบ
เหตุการณ์ต่างๆ ที่เกิดขึ้นและสามารถที่จะยอมรับหรือปฏิเสธการร้องขอนั้นได้

6.6 การถ่ายทอดตำแหน่ง URL

ISAPI Exetension สามารถถ่ายทอดการร้องขอจาก URL ที่ Exetension ทำงานอยู่ไปสู่ URL ของ Web
server อื่นได้ ความสามารถในข้อนี้ถือเป็นประโยชน์อย่างยิ่งกรณีที่ข้อมูลถูกจาก Web site หนึ่งไปสู่อีก Web site
โดยตำแหน่ง URL ยังคงเดิมสามารถอ้างอิงถึงได้ ส่วนการเข้าถึงข้อมูลที่แท้จริง client จะถูกกำหนดเส้นทางไปสู่
Web site นั้นได้โดยอัตโนมัติ

6.7 การรักษาความปลอดภัย

จะใช้เทคนิคของ ISAPI Filter เพื่อจัดการกับสิทธิ์การเข้าใช้ระบบบน Web Server ได้ โดยที่นโยบายของ
ความปลอดภัยเราเป็นผู้กำหนดเอง และ อีกเทคนิคหนึ่งก็คือ ISAPI ทำงานในรูปแบบของ DLL ซึ่งเป็นการยากที่ผู้
ไม่ประสงค์ดี ที่จะมาแทรกแซงระบบเราได้

บทที่ 7

Hyper Text Markup Language (HTML)

7.1 HTML

HTML หรือ Hyper Text Markup Language เป็นภาษาลำดับงานแบบหนึ่ง ใช้สำหรับจัดทำหน้าตาการแสดงผลของข้อมูลที่ถูกจัดเก็บใน web site หรือ web server ตามปรกติแล้วข้อมูลที่นำมาแสดงจะมีหน้าตาเป็น hypertext (ถ้ายังไม่เข้าใจว่า hypertext คืออะไร ก็ขอให้ลองนึกดูหน้าจอ Help ของโปรแกรม Windows ดู ซึ่งจะเป็นข้อความยาวๆ พร้อมกับคำบางคำที่สามารถคลิกเพื่อให้เชื่อมโยงไปยังหน้าจออื่นๆ ได้)

ภาษาลำดับงาน HTML จะมีส่วนที่เรียกว่า Markup tag (ส่วนขยายลักษณะหรือเรียกย่อๆว่า tag) ที่จะคอยบอกให้โปรแกรม web browser ทั้งหลายได้ทราบว่า หน้าจอหรือข้อความนั้นๆ จะถูกแสดงอยู่ในลักษณะอย่างไร ส่วนขยายลักษณะหรือ Markup tag นี้ จะเป็นข้อความหรือคำสั่งที่มีเครื่องหมาย < > ครอบเอาไว้ และใช้เครื่องหมาย “/” เป็นตัวจบ เช่น ขึ้นต้นด้วย <H1> จบด้วย </H1> ภาษาลำดับงาน HTML นี้ มีการพัฒนาปรับปรุงให้มีความสามารถในการแสดงผลในรูปแบบแปลกๆ ใหม่ๆ ขึ้นเรื่อยๆ จาก HTML เวอร์ชัน 1.0 มาจนถึง HTML เวอร์ชัน 3.2 ในปัจจุบันที่มีลูกเล่นมากมาย แต่ผู้ใช้บริการต้องระวังในกรณีที่ใช้โปรแกรม web browser เก่าๆ ที่ไม่สามารถรับคำสั่ง HTML 3.0 หรือ 3.2 ได้ เมื่อเรียกมาใช้บริการ World Wide Web ที่ web site ที่เขียนด้วย HTML 3.0 หรือ 3.2 ก็อาจจะแสดงผลไม่ถูกต้องได้

ภายใน web page ที่ถูกสร้างจากภาษา HTML นั้น จะมีส่วน Markup tag ในแบบต่างๆ กันเพื่อบอกโปรแกรม browser ได้ทราบและแสดงผลได้ถูกต้อง ขอยกตัวอย่างของ tag เพื่อให้เห็นภาพง่ายๆ ดังนี้

<HTML> เป็น tag ที่จะปรากฏขึ้นต้นในแต่ละ page เสมอ เพื่อบอกให้ทราบว่าข้อมูลหรือคอนเทนต์ภายในนี้ถูกเขียนขึ้นในรูปแบบของ HTML และจบที่ tag </HTML> แสดงว่าจบข้อมูลหรือจบเอกสารนี้แล้ว

<HEAD> เป็น tag ที่บอกว่าข้อความต่อไปนี้เป็นหัวเรื่องและจบด้วย tag </HEAD>

<TITLE> เป็น tag ที่จะบอกว่าข้อความต่อไปนี้เป็นชื่อไตเติลของ page นี้ซึ่งจะแสดงอยู่ที่ส่วนแสดงไตเติลของโปรแกรม web browser เมื่อใช้งานมาที่หน้าจอนี้ จบด้วย </TITLE>

<BODY> เป็น tag ที่จะบอกว่า ส่วนของข้อความต่อไปนี้ทั้งหมดและรวมถึงรูปกราฟิกจะเป็นส่วนของเนื้อหาข้อมูลใน page นี้

<P> เป็น tag ที่จะแบ่งแต่ละย่อหน้า (paragraph) ออกจากกัน

 เป็น tag ที่บอกให้ขึ้นบรรทัดใหม่

 และ <I> เป็น tag ที่บอกให้แสดงอักษรเป็นตัวหนา หรือตัวเอียง (Bold หรือ Italic) ตามลำดับ

ตามที่กล่าวไปแล้วว่าภาษา HTML ได้มีการพัฒนามาเรื่อยๆ จนเป็นเวอร์ชัน 3.2 สิ่งใหม่ๆ และมีเพิ่มเข้ามาก็คือ tag ที่แสดงลูกเล่นต่างๆ ซึ่งโปรแกรม browser บางโปรแกรมก็จะเลือกใช้ไม่ได้ เช่น <BLINK> และ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<CENTER> เป็น tag นี้ได้โปรแกรม browser บางตัวจะไม่รู้จัก tag ทั้งสองนี้เลย ซึ่งก็จะไม่สามารถแสดงผลตามที่ต้องการได้

7.2 การเขียน HTML เบื้องต้น

ภาษา HTML นิยาม Element หลักร่างๆ เพื่อใช้ในการเขียนเอกสาร Element หนึ่งๆ อาจประกอบด้วย tag (เริ่มต้นด้วยอักษร <และลงท้ายด้วยตัว อักษร> Attribute และ Text ซึ่งอยู่ในรูปแบบดังนี้

<tag-name>text</tag-name> <tag-name attribute-name=arg>text</tag-name> และ <tag-name>

7.2.1 Element ที่มีในทุก Home page

ทุก Home page จะต้องมี Element ที่กำหนดขอบเขตของเอกสาร ส่วนหัวเอกสาร ส่วนเนื้อเรื่อง ของเอกสารซึ่งจะอยู่ในรูปแบบดังนี้

```
<HTML>
<HEAD>
<TITLE>Title Sentence</TITLE>
</HEAD>
<BODY Attribute ... >
Document Body
</BODY>
</HTML>
```

- <HTML> ... </HTML> เป็น element ที่ใช้กำหนดขอบเขตของเอกสาร
- <HEAD> ... </HEAD> เป็น element ที่ใช้กำหนดขอบเขตของหัวเรื่องซึ่ง โดยส่วนใหญ่แล้วหัวเรื่องจะประกอบด้วย element <TITLE> ... </TITLE> ซึ่งจะแสดงหัวข้อเรื่องของเอกสาร Title Sentence จะถูกแสดงที่ Windows Bar ของ Browser
- <BODY> ... </BODY> ใช้กำหนดขอบเขตของเนื้อหาในเอกสาร โดยอาจ กำหนด Attribute ของ Body เช่น <body background="picture/bg.jpg" text="#ff00ff" link="#ff0000"> background="picture/bg.jpg" มีความหมายว่าให้สร้างฉากหลังของ homepage โดยใช้ภาพจาก ไฟล์ bg.jpg เป็น relative pathname ซึ่งเทียบกับ directory ที่ เอกสารฉบับนี้เก็บอยู่ text="#ff00ff" เป็นการกำหนดสีของตัวอักษรที่แสดงบน homepage ที่ ดังกล่าวเกิดจากการผสมของสีแดง เขียว และน้ำเงิน(RGB) โดยแสดงเป็นเลขฐาน 16 ในที่นี้ ff00ff สีแดงมีค่า 255 สีเขียวมีค่า 0 และสีน้ำเงินมีค่า 255 ทำให้ได้สีม่วง link ="#ff0000"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาดูเท่านั้น เมื่อนุญาตให้นำไปเผยแพร่บนเว็บไซต์ใดๆ ไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการกำหนดสีของตัวอักษรที่ใช้เชื่อมโยงไปยังเอกสารอื่น(ซึ่งจะอธิบายต่อไป) ในที่นี้จะได้สีแดงหรืออีกตัวอย่าง เช่น `<body> bgcolor="#feefff" text="#0000ff" link="#ff0000"> bgcolor="#feefff"` เป็นการสร้างสีพื้นของ homepage จาก code นี้ จะทำให้ได้ชมพูอ่อน ตัวอักษรได้สีน้ำเงินและ link จะเป็นสีแดง

7.2.2 Header Element

Element ที่มีรูปแบบ `<h> ... </h>` โดยที่ y แทนตัวเลข ตั้งแต่ 1 ถึง 6 ใช้ในการแสดงส่วนของประโยคที่ให้เป็นหัวข้อ โดยที่ `<h1> ... </h1>` เป็นอักษรขนาดใหญ่ที่สุดและ `<h6> ... </h6>` เป็นตัวอักษรขนาดเล็กที่สุด

7.2.3 List Element

- List จำกัดความ

การใช้ List จำกัดความในเอกสาร HTML นี้ค่อนข้างจะซับซ้อน เพราะเราแทรก element อื่นๆ เข้าไปด้วย โดยสังเกต element ที่อยู่ในรูปแบบ

```
<dl>
<dt> <!--topic-->
<dd> <!--defination-->
<dt>
<dd>
</dl>
```

ในที่นี้ `<!--Comment-->` แทน comment element ใน HTML

ในเอกสาร HTML ทุก element ที่อยู่หลัง tag `<dt>` จะประกอบกับเป็น topic และทุก element ที่อยู่หลัง tag `<dd>` จะประกอบกับเป็นคำบรรยาย

- List แบบไม่มี ลำดับ

List นี้ เหมาะกับการใช้แสดงรายการที่ไม่มีลำดับ มีรูปแบบดังนี้

```
<ul>
<li> <!--First Item-->
<li> <!--Second Item-->
:
</ul>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทุก element ที่อยู่หลัง tag จะถูกแสดงโดยมีการย่อหน้าและนำหน้าด้วยเครื่องหมายสามเหลี่ยม หรือ สี่เหลี่ยม หรือวงกลม ขึ้นอยู่กับ browser

- **List แบบมี ลำดับ**

List นี้ เหมาะกับการใช้งานแสดงรายการที่เป็นลำดับ มีตัวเลขกำกับ มีรูปแบบดังนี้

```
<ol>
<li> <!--First Item-->
<li> <!--Second Item-->
:
</ol>
```

ทุก element ที่อยู่หลัง tag จะถูกแสดงโดยมีการย่อหน้าและมีตัวเลขกำกับในกรณีที่เป็น ordered ซ้อนกัน จะมีการแสดงลำดับของหัวข้อย่อยด้วย เช่น 1.1}1.1.1 เป็นต้น

7.2.4 Paragraph

paragraph เป็นกลุ่มประโยค และ element ที่อยู่ใน element <p align="center" > ... </p> attribute align="center" เป็น option ใช้ในกรณีที่เราต้องการให้ paragraph อยู่ตรงกลางของ homepage ในไฟล์ Link.htm เราใช้ Element นี้โดยจัดให้ paragraph ข้างในอยู่ตรงกลางของ Homepage มี tag ที่มีความคล้ายคลึงกับ <p> คือ
 (break) ซึ่งจะใช้เมื่อเราต้องการขึ้นบรรทัดใหม่ ใน paragraph เดียวกันถ้าเราใช้ <p> แทน
 paragraph จะถูกแตกเป็น 2 paragraph

7.2.5 Logical Styles และ Physical Styles

มี element บาง element ใช้เปลี่ยนตัวอักษรเป็นตัวเอียง หรือตัวเข้ม เราสามารถแบ่ง element นี้เป็น 2 กลุ่ม คือ ทางความหมาย (Logical) และทางกายภาพ (Physical) Logical จะเปลี่ยนตัวอักษรโดยใช้ความหมายในการใช้งานเป็นหลัก เช่น ต้องการเน้น ต้องการแสดงว่าเป็นตัวแปร ตัวอักษรที่ปรากฏจริงเป็นอย่างไรขึ้นอยู่กับ browser ส่วน Physical Styles จะเปลี่ยนตัวอักษรตามที่จะปรากฏจริง การเลือกใช้กลุ่มใดขึ้นอยู่กับดุลยพินิจและควรใช้กลุ่มเดียวกันทั้งเอกสาร แต่อย่างไรก็ตามควรตระหนักว่า Physical Styles อาจจะไม่ Support โดย HTML Version ใหม่

7.2.6 Logical Styles

element ต่อไปนี้จะใช้เป็นหัวเรื่องของหนังสือหรือภาพยนตร์

` ... `

(emphasize) เน้น

` ... `

(strong emphasize) เน้นหนัก

`<cite> ... </cite>`

element ต่อไปนี้จะใช้ในงานคอมพิวเตอร์

`<code> ... </code>`

แสดงว่าเป็น source code (code)

`<samp> ... </samp>`

แสดงว่าเป็น output (sample output)

`<kbd> ... </kbd>`

แสดงว่าเป็น input จาก keyboard (keyboard)

`<var> ... </var>`

แสดงว่าเป็นตัวแปร (variable)

7.2.7 Physical Styles

` ... `

ตัวเข้ม (bold)

`<i> ... </i>`

ตัวเอียง (italic)

`<tt> ... </tt>`

ตัวพิมพ์ (Typewriter font)

7.2.8 Address

เป็น element ที่มักใช้ใช้การแสดง Email Address เช่น

`<address> This Web is maintained by s8013302@kmitl.ac.th</address>`

7.2.9 Horizontal Rules

`<hr>` tag ใช้ในการสร้างเส้นแนวนอน ส่วนมากใช้ในการแยกข้อความที่ไม่ขึ้นต่อกันเป็นส่วนๆ และเราสามารถเปลี่ยนขนาดของเส้นได้ โดยใช้ Attribute Size และ Width เช่น

`<hr size=4 width=50%>` นั้นหมายความว่าขนาดความหนาของเส้น 4 มิลลิเมตร ขนาดความยาว 50% ของ Homepage

7.2.10 Inline Images

Web browser สามารถแสดงภาพ (ซึ่งจะแสดงติดกับอักษร) ที่มีรูปแบบมาตรฐานของภาพ เช่น XBM, TIF, JPG, GIF และ PNG เราสามารถใส่ภาพใน Homepage โดยใช้ element

``

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนขนาดภาพทำได้โดยใช้ Attribute Width และ Height ในหน่วยขนาด มิลลิเมตร
เช่น

```

```

โดย Default แล้ว ตัวอักษรที่ตามรูป จะอยู่ตำแหน่งด้านล่างติดกับรูป การจัดตำแหน่งของตัวอักษรทำได้โดยใช้ Attribute คือ align=top หรือ center จะทำให้ตัวอักษรอยู่บริเวณข้างบนติดกับรูป และตรงกลางติดกับรูปตามลำดับ

ในกรณีที่เรต้องการ Support Browser ที่ไม่สามารถแสดงภาพได้ เช่น โปรแกรม lynx อาจแทนรูปดังกล่าวด้วยประโยคที่เหมาะสมกับรูป ทำได้โดยใช้ Attribute alt=text

7.3 การเชื่อมโยงหรือ Linking

ดังที่ได้กล่าวมาแล้วว่าคุณสมบัติหนึ่งของภาษา HTML คือความสามารถในการเชื่อมโยงระหว่างเอกสารซึ่งไม่เกี่ยวข้องกัน Browser จะทำให้โดดเด่นขึ้น (highlight) โดย highlight รูปหรือข้อความที่บอกถึง link element ที่ใช้ในการทำ link เรียกว่า anchor มีรูปแบบดังนี้

```
<a href="filename">Describe Text</a>
```

Describe Text เป็นข้อความที่จะถูก highlight และ link ไปยัง file ชื่อ filename

7.3.1 การเชื่อมโยงระหว่างเอกสาร

เราสามารถ link ไปยังเอกสารอื่นโดยระบุ Relative path โดยเทียบจาก path ของเอกสารปัจจุบัน
เช่น

```
<a href="docs/link.htm"> Link</a>
```

ไฟล์ link.htm จะอยู่ใน subdirectory ชื่อ docs รูปแบบดังกล่าวนี้เรียกว่า relative link เราอาจกำหนดเป็น absolute pathname (URL) แต่ถ้าเป็นกรณีอยู่บน server เดียวกัน การใช้ relative pathname จะมีประสิทธิภาพกว่า

URL (Uniform Resource Locator) ใช้ในการระบุตำแหน่งของไฟล์บน server URL ประกอบด้วยประเภทของ service, address ของ server และตำแหน่งของไฟล์ ดังนี้

scheme://host.domain[:port]/path/filename เช่น HERE

นอกจากต้นแบบ http ยังมีรูปแบบอื่น เช่น file, ftp, gopher, news, telnet และ WAIS เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3.2 Link ภายใน section ของ เอกสาร

ทำได้เหมือนกับการ link ของ เอกสารเพียงแต่เราต้องเพิ่มตำแหน่งของเอกสารที่เราต้องการ link ไปในเพจของเรา เช่น ถ้าเราต้องการ link จากไฟล์ main.htm ไปยังตำแหน่งต่าง ๆ ของไฟล์ link.htm เราสามารถทำ link ได้ดังนี้

```
<a href="link.htm#internet"> ... </a>
```

```
<a href="link.htm#netscape"> ... </a>
```

```
<a href="link.htm#html"> ... </a>
```

ซึ่งหมายความว่าให้ link ไปยังไฟล์ link.htm ในตำแหน่งที่ mark ว่า internet, netscape และ html ตามลำดับ แต่ถ้าเราต้องการ link ไปในไฟล์เดียวกัน เช่น link ภายในไฟล์ link.htm เราก็สามารถทำ mark ดังนี้

```
<a name="internet"> ... </a>
```

```
<a name="netscape"> ... </a>
```

```
<a name="html"> ... </a>
```

7.4 mailto

เราสามารถทำให้ผู้ที่เข้าเยี่ยมชม home page ของเราสะดวกในการติดต่อกับบุคคลที่อ้างถึงใน home page โดยใส่ mailto ใน link เช่น `Me` เพื่อให้ผู้อ่านสามารถส่ง email ถึง 's8013302@kmitl.ac.th' ได้โดยคลิกที่ ME

จากความรู้ในการสร้าง home page ข้างต้นจะเห็นว่า HTML เป็นภาษาที่ประกอบด้วยแท็กต่างๆ มากมาย ซึ่งแท็กที่เราจะได้พบเห็นอยู่บ่อยๆ บนเอกสาร HTML มีต่างๆ ดังนี้

<code><!-></code>	สร้างคำอธิบายยัง html ของคุณที่จะไม่สามารถมองเห็นบนเพจ
<code><a href></code>	สร้าง link ไปยังเพจอื่นๆ หรือ link ไปยังบริเวณอื่นบนเพจเดียวกัน
<code><a name></code>	สร้างจุด anchor กระโดดไปยังส่วนบางส่วนของเพจ
<code><address></code>	ระบุชื่อหรือที่อยู่บนเว็บเพจ
<code><applet></code>	เริ่มต้นคำสั่งสำหรับ java application
<code><area></code>	กำหนดพื้นที่การคลิกของ image map
<code><audioscope></code>	กำหนดการปรากฏของการแสดงกราฟิกสำหรับเสียงเมื่อมีการ view ด้วย Web TV
<code></code>	การทำภายในเท็กซ์ของ tags เพื่ออักษร bold
<code><base></code>	การระบุ URL ของภาในเพจของโค้ด HTML
<code><basefont></code>	การกำหนดค่าเริ่มต้น ขนาด สี และฟอนต์ สำหรับทำเพจทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการศึกษาค้นคว้า เปรียบเทียบเนื้อหาไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<bgsound>	การเพิ่มเสียงให้กับแบคกราวด์บนเว็บเพจ โดยการใช้ไฟล์ midi หรือ wav
<big>	การทำเท็กซ์ให้มีขนาดใหญ่ระดับหนึ่งมากกว่า base Font
<blink>	การทำเท็กซ์ที่มีอยู่ใน tag กระพริบในระหว่างเวลาปกติ
<blackface>	การทำฟอนต์ขนาดสองเท่า &bold สำหรับผู้ใช้ Web TV
<body>	กำหนดรูปภาพแบคกราวด์ เท็กซ์ link & สีของ link ที่ไปเยี่ยมชม
<bg>	ชุดของเท็กซ์ที่ย่อหน้า สำหรับผู้ใช้ Web TV
 	ทำ line break หลังจากชุดของเท็กซ์หรือกราฟิก
<caption>	สร้างชื่อของตาราง
<center>	จัดกึ่งกลางชุดมูลเท็กซ์และกราฟิก
<cite>	ตัวพิมพ์เข้ม (Emphasized) ในตอนหนึ่งของข้อเขียนของเท็กซ์ในรูปแบบตัวเอียง (italics)
<code>	รูปแบบเท็กซ์ในฟอนต์ monospace
<comment>	วางคำอธิบายใน HTML และซ่อนคำอธิบายจากเพจจริง
<dd>	ย่อหน้าเท็กซ์หนึ่งระดับเมื่อใช้ภายใน <dl> tag
<dfn>	กำหนดเงื่อนไข
<dir>	ทำรายชื่อสั้นๆ ของ items ในหนึ่งแนวตั้ง
<dl>	แสดงรายชื่อของเงื่อนไข & คำจำกัดความ
<dt>	การกำหนด term ในรายชื่อ <dl>
	การทำตัวเข้ม (Emphasized) ตอนหนึ่งของข้อเขียนของเท็กซ์ในตัวเอียง (italics)
<embed>	การฝังไฟล์เสียงหรือ avi movie ลงยัง html
<fn>	การสร้าง anchor point ที่คล้ายกับ <a name> tags
	การเปลี่ยนขนาดสี & ฟอนต์ของเท็กซ์
<form>	กำหนดเพจหรือพื้นที่ของเท็กซ์ไปยัง input form
<frame>	กำหนด frame ภายใน <frames> tag
<framset>	กำหนด attributes ของเพจที่จะมีการใช้ frames
<h1> ... <h6>	แสดงเท็กซ์ในรูปแบบ header ให้มีขนาดใหญ่ขึ้น
<head>	กำหนดส่วนบนของเอกสาร
<hr>	สร้างเส้นในแนวนอน
<html>	การระบุที่เอกสารว่าเขียนใน html
<I>	แสดงเท็กซ์ในลักษณะตัวเอียง (italics)
	แทรกรูปภาพ imagemap หรือแอนิเมชันไปยังเพจ
<input>	กำหนดค่าอินพุตฟิลด์ใน form
<isindex>	สร้าง prompt ไปยังการจัดเก็บความต้องการสำหรับ index
<kbd>	แสดงเท็กซ์ในฟอนต์ monospaced

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	สร้าง bullet point หรือบรรทัดใหม่ ในรายชื่อเมื่อใช้ภายใน <div>, <menu>, <lo> และ tag
<limittext>	กำหนดค่าความกว้างสูงสุดสำหรับเท็กซ์เพื่อทำการแสดงสำหรับการใช้ด้วย Web TV
<link>	สร้างการจัดระเบียบสำหรับชุด navigating ของเอกสาร
<listimg>	ชุด perfromtted (รูปแบบเดิมๆ)ของเท็กซ์ใน list (รายการ)ที่แสดงในฟอนต์ monospaced
<map>	จำกัดความ client-side map
<marquee>	สร้างเท็กซ์ scroll เมื่อ view ด้วย MSIE
<menu>	การสร้างรายชื่อ
<nobr>	การทำ line break เพื่อหยุดในเท็กซ์หรือชุดของรูปภาพ
<noframes>	การจำกัดการเห็น frame ถ้าการ view นั้นเกิดจากบราวเซอร์ที่ไม่มีความสามารถ
	การกำหนดการเริ่มต้นและจบของ list
<option>	การกำหนด list ของ items ที่ view จาก choose (ตัวเลือก)
<p>	การกำหนดการเริ่มต้นและสิ้นสุดของวรรค ตอน
<plaintext>	แสดงเท็กซ์ในฟอนต์ monospace
<pre>	ชุด Preformats ของเท็กซ์ในฟอนต์ monospace
<s>	การวาด line ผ่านชุดหรือเท็กซ์ถ้ามีการ cross กันภายนอก
<samp>	แสดงเท็กซ์ในฟอนต์ monospace
<script>	การกำหนด ค่าเริ่มต้น และสิ้นสุดของ java script
<select>	แสดง list ของ items ในแต่ละ view จากฟอร์มตัวเลือก (choose)
<server>	เส้นทางต่างๆ ของการทำแอนิเมชันจากเซิร์ฟเวอร์
<small>	แสดง เท็กซ์ขนาดเล็กหนึ่งขนาดนอกจากคำดีพอลท์เท็กซ์
<strike>	การวาดเส้นผ่านเท็กซ์ถ้ามีการลากตัดกัน ไปบึงภายนอก
	แสดงเท็กซ์ในฟอนต์ boldface (ตัวพิมพ์หนา)
<sub>	แสดงเท็กซ์เหมือน subscript (เล็กและต่ำลง)
<sup>	แสดงเท็กซ์เหมือน subscript (เล็กและโผล่ขึ้น)
<table>	สร้างตาราง
<td>	การกำหนดเซลล์ภายในของตาราง
<textarea>	การระบุขนาดของพื้นที่เท็กซ์

บทที่ 8

การดำเนินงานและการพัฒนาระบบ

8.1 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาระบบการลงทะเบียนเรียนแบบใหม่ (โครงการเดิมคือ โครงการระบบสารสนเทศของสำนักทะเบียนและประมวลผล ส่วนงานลงทะเบียนและชำระค่าธรรมเนียม ซึ่งเป็นโปรแกรมประยุกต์บนระบบเครือข่ายแบบไคลเอนท์/เซิร์ฟเวอร์) โดยใช้เทคโนโลยีการสื่อสารคอมพิวเตอร์ที่เป็นเครือข่ายอินเทอร์เน็ต
2. เพื่อนำความรู้ในด้านต่างๆที่ได้เรียนมาทึเช่นความรู้ทางด้าน วิชาการระบบฐานข้อมูล, วิชาด้านคอมพิวเตอร์เน็ตเวิร์ก, วิชาด้านโปรแกรมมิ่ง และอื่นๆ นำมาประยุกต์ใช้ในโครงการนี้
3. เพื่อศึกษาการประยุกต์ใช้งานระบบจัดการฐานข้อมูล (DBMS) โครงการนี้ใช้ระบบจัดการฐานข้อมูล Microsoft SQL Server และ IIS (Internet Information Server) เป็นระบบที่ใช้สำหรับติดตั้ง Web Server ซึ่งทั้ง 2 อย่างนี้ทำงานบนระบบปฏิบัติการ Window NT 4.0 Thai Enable
4. เพื่อศึกษาวิธีการเขียน Web Page โดยนำเอา Java Script มาประยุกต์ใช้ร่วมกับภาษา HTML ซึ่งภาษา HTML นี้เป็นภาษาที่ใช้สำหรับเขียนโฮมเพจ
5. เพื่อศึกษาวิธีการเขียน Script สำหรับไฟล์ .IDC และ .HTX สำหรับการติดต่อระบบฐานข้อมูลผ่านทางเว็บเบราว์เซอร์
6. เพื่อศึกษาการเขียนภาษา SQL และการสร้างสตอร์โพซีเจอร์ภายใน SQL Server
7. โครงการนี้สามารถนำมาทำเป็นโครงการต้นแบบของระบบงานที่ต้องมีการติดต่อกันระหว่างเว็บเบราว์เซอร์ กับระบบฐานข้อมูล

8.2 ขอบเขตของโครงการ

เนื่องจากงานด้านการลงทะเบียนนักเรียนนั้น มีขอบเขตการทำงานที่กว้าง จึงมีการแบ่งส่วนการทำงานหลักออกเป็นส่วนย่อยๆ ซึ่งแต่ละส่วนต่างมีความสำคัญคือ

- ระบบงานทะเบียนประวัตินิสิต
- ระบบงานหลักสูตรและการศึกษา
- ระบบการจัดชั้นเรียนและการจัดห้องเรียน
- ระบบงานการลงทะเบียนเรียน
- ระบบการคิดค่าธรรมเนียมการศึกษา

โครงการนี้ได้รับผิดชอบในส่วนของระบบงานการลงทะเบียนเรียน และการคิดค่าธรรมเนียมการศึกษาของสำนักทะเบียนกลาง หน้าที่ของส่วนนี้คือทำการรับข้อมูลการลงทะเบียนเรียนของนักศึกษา นำไปจัดเก็บและเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมวลผลพร้อมทั้งคำนวณหาค่าธรรมเนียมของนักศึกษาในแต่ละภาคการศึกษา ในส่วนของระบบการลงทะเบียน มีฟังก์ชันการทำงานหลักๆ คือ

- การลงทะเบียนเรียน
- การเพิ่มและการถอนวิชาเรียน
- การเปลี่ยนวิชาเรียน
- การตรวจสอบการลงทะเบียน ,การชำระค่าธรรมเนียม และตรวจสอบผลการเรียน

8.3 ขั้นตอนในการพัฒนาโครงการงาน

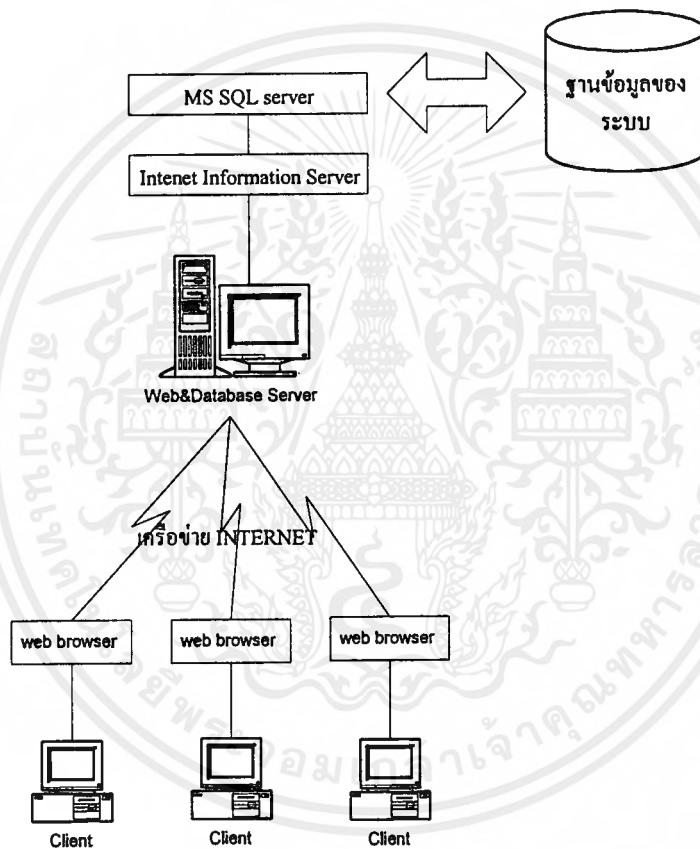
1. สรุปขอบเขตของงานที่จะทำ โดยพิจารณาถึงลักษณะการนำไปใช้ในงานจริง
2. ศึกษาขั้นตอนต่างๆที่ควรจะเป็นของระบบแล้วนำมาเขียนเป็น Data Flow Diagram และ Diagram แสดงขั้นตอนการทำงานของระบบ
3. ศึกษาฐานข้อมูลของระบบการลงทะเบียนของโครงการเก่า (โครงการระบบสารสนเทศของสำนักทะเบียนและประมวลผล ส่วนงานลงทะเบียนและชำระค่าธรรมเนียม ซึ่งเป็น โปรแกรมประยุกต์บนระบบเครือข่ายแบบไคลเอนท์/เซิร์ฟเวอร์) และนำ Data Flow Diagram มาใช้ประกอบในการพิจารณาการออกแบบฐานข้อมูลเพิ่มเติมจากฐานข้อมูลเดิม โดยใช้ NIAM แสดงความสัมพันธ์ของข้อมูลแล้วทำการแปลงออกมาเป็นตาราง จากนั้นจึงทำการระบุนิคมของข้อมูลและลักษณะของข้อมูลที่จะเก็บลงตาราง
4. ทำการติดตั้งระบบปฏิบัติการ Window NT 4.0 Thai Enable, ติดตั้ง IIS Web Server และติดตั้ง Microsoft SQL Sever
5. ทำการนำเข้าข้อมูลที่จำเป็นต่อการทดลอง ของโครงการนี้
6. ทำการออกแบบลักษณะของหน้าจอ (User Interface) และแบบฟอร์มต่างๆ ที่จะนำมาแสดงบนเบราว์เซอร์ โดยยึดหลักการใช้งานง่ายและถูกต้อง
7. สำหรับการสร้างแบบฟอร์มนั้นจะสร้างโดยใช้ภาษา HTML และ Java Script ในการพัฒนา จะเขียนแล้วเก็บลงในรูปแบบของไฟล์ HTX
8. สำหรับส่วน Server จะต้องมีการเขียน Script IDC ขึ้นมาเพื่อทำหน้าที่เป็นตัวกลางในการที่ เว็บเบราว์เซอร์ จะทำงานกับฐานข้อมูลที่อยู่บน Server นี้ นอกจากนั้นยังต้องสร้างสตอร์โพซิเจอร์ต่างๆที่จำเป็นต่อระบบเพื่ออำนวยความสะดวกในการพัฒนาระบบให้มีประสิทธิภาพยิ่งขึ้น
9. ในการพัฒนาต้องมีการตรวจสอบความถูกต้องของระบบในทุกๆขั้นตอน ถ้าพบข้อผิดพลาดก็มีความจำเป็นที่จะต้องแก้ไขข้อผิดพลาดนั้นเพราะถ้าไม่ทำการแก้ไขแล้วก็จะไม่สามารถทำการพัฒนาระบบต่อไปได้ทุกครั้งที่มีการแก้ไขจะมีการบันทึกปัญหาต่างๆและวิธีการแก้ไขลง Log Book ไว้ด้วย
10. จัดทำเอกสารอธิบายถึงรายละเอียดทั้งหมดของระบบทั้งในด้านทฤษฎีต่างๆที่นำมาใช้และบอกถึงผลงานที่ได้ทำไปพร้อมแนวทางการพัฒนาต่อ
11. จัดทำคู่มือการใช้งาน (User Manual)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.4 โครงสร้างของระบบ

โครงสร้างหลักของระบบการลงทะเบียนเรียนผ่านอินเทอร์เน็ต นั้นประกอบไปด้วยส่วนสำคัญๆ ดังนี้

1. ผู้ขอใช้บริการ (Client) จะขอใช้บริการได้โดยใช้เว็บเบราว์เซอร์ ซึ่งโครงงานนี้ต้องการเว็บเบราว์เซอร์ของ Internet Explorer เวอร์ชัน 3.0 ขึ้นไปและ Netscape Navigator เวอร์ชัน 3.0 ขึ้นไป
2. ระบบเครือข่ายซึ่งเชื่อมต่อกับระบบเครือข่าย Internet คือต้องสนับสนุนการสื่อสารด้วยโปรโตคอล TCP/IP
3. ผู้ให้บริการ (Web&Database Server) เป็นส่วนที่ให้บริการ WWW และบริการฐานข้อมูล



รูปโครงสร้างหลักของระบบลงทะเบียนเรียนผ่าน Internet

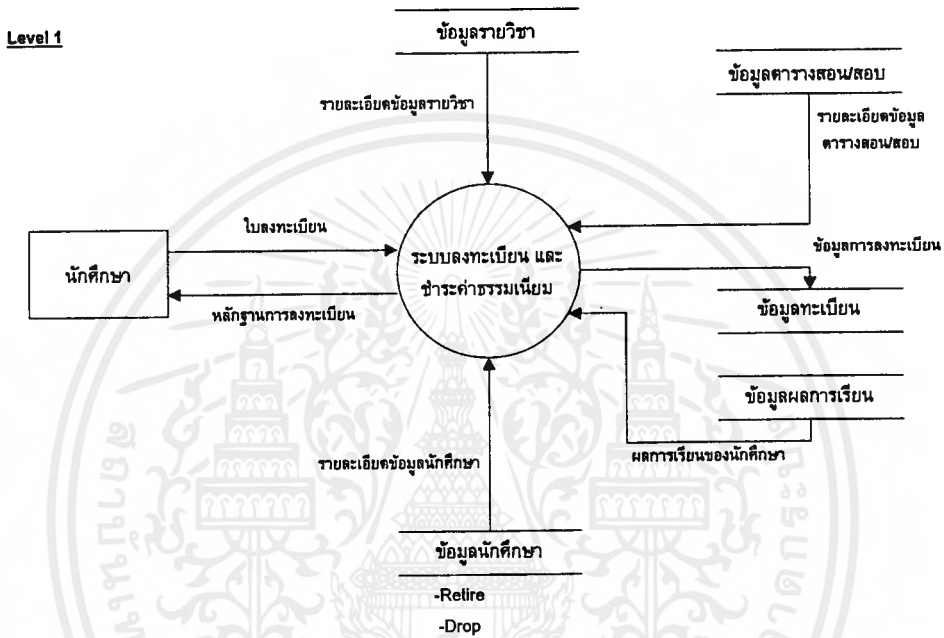
8.5 การวิเคราะห์และออกแบบระบบ

หลังจากที่ได้ศึกษาขั้นตอนต่างๆ ที่จำเป็นของการลงทะเบียนของนักศึกษาในแต่ละภาคเรียนก็จะต้องนำขั้นตอนเหล่านั้นมาออกแบบ โดยเขียนเป็นโมเดลที่ทำการอธิบายระบบ โดยจะใช้ Data Flow Diagram มาอธิบายถึงลักษณะการไหลของข้อมูลตั้งแต่เริ่มกระบวนการจนเสร็จสิ้นกระบวนการ ซึ่งจะมีประโยชน์อย่างมากเมื่อเวลาทำการพัฒนาโครงการให้เป็นไปตามลำดับขั้นที่ถูกต้องและเหมาะสม เพราะการอธิบายด้วยรูปจะสามารถทำให้เห็นภาพโดยรวมได้มากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนฐานข้อมูลนั้น ครงงานนี้ ออกแบบฐานข้อมูลโดยใช้ NIAM model มาแสดงความสัมพันธ์ของข้อมูล ซึ่งเมื่อทำการ map ออกมาแล้วจะได้ตารางฐานข้อมูลที่เป็น Fifth Normal Form (5NF) เลย โดย ไม่ต้องทำการ Normalized อีก

8.5.1 แผนผังการไหลของข้อมูล (Data Flow Diagram)

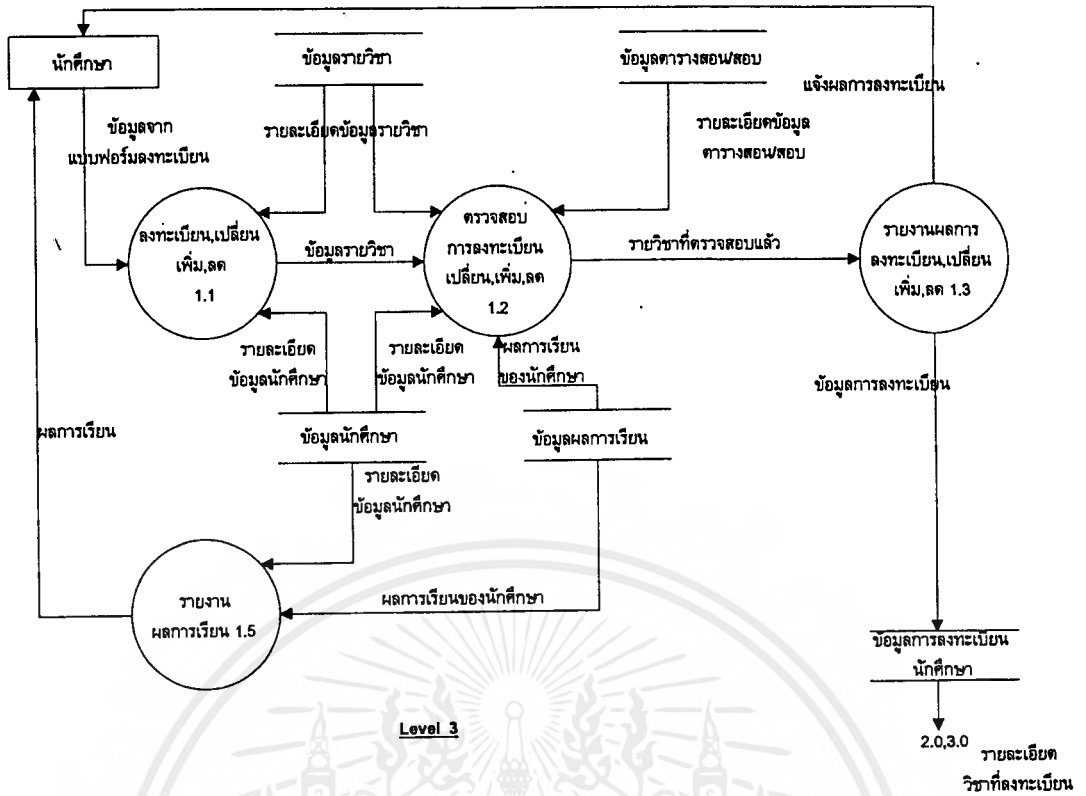


แผนผังการไหลของข้อมูลในระดับที่ 1

ในระดับที่ 1 นี้จะอธิบายถึงภาพโดยรวมกว้างๆของระบบซึ่งจะประกอบไปด้วย

- ส่วนที่นำเข้าข้อมูลซึ่งก็คือ นักศึกษา
- ส่วนประมวลผล หมายถึง ระบบลงทะเบียนและชำระค่าธรรมเนียม
- ส่วนเก็บข้อมูล หมายถึง ฐานข้อมูล

และจากระดับที่ 1 นี้จะทำการแตกเป็นระดับย่อยๆเพื่ออธิบายรายละเอียดของระบบงานต่อไป



แผนผังการไหลของข้อมูลในระดับที่ 3 (2)

8.5.2 แผนภาพ NIAM (NIAM Diagram)

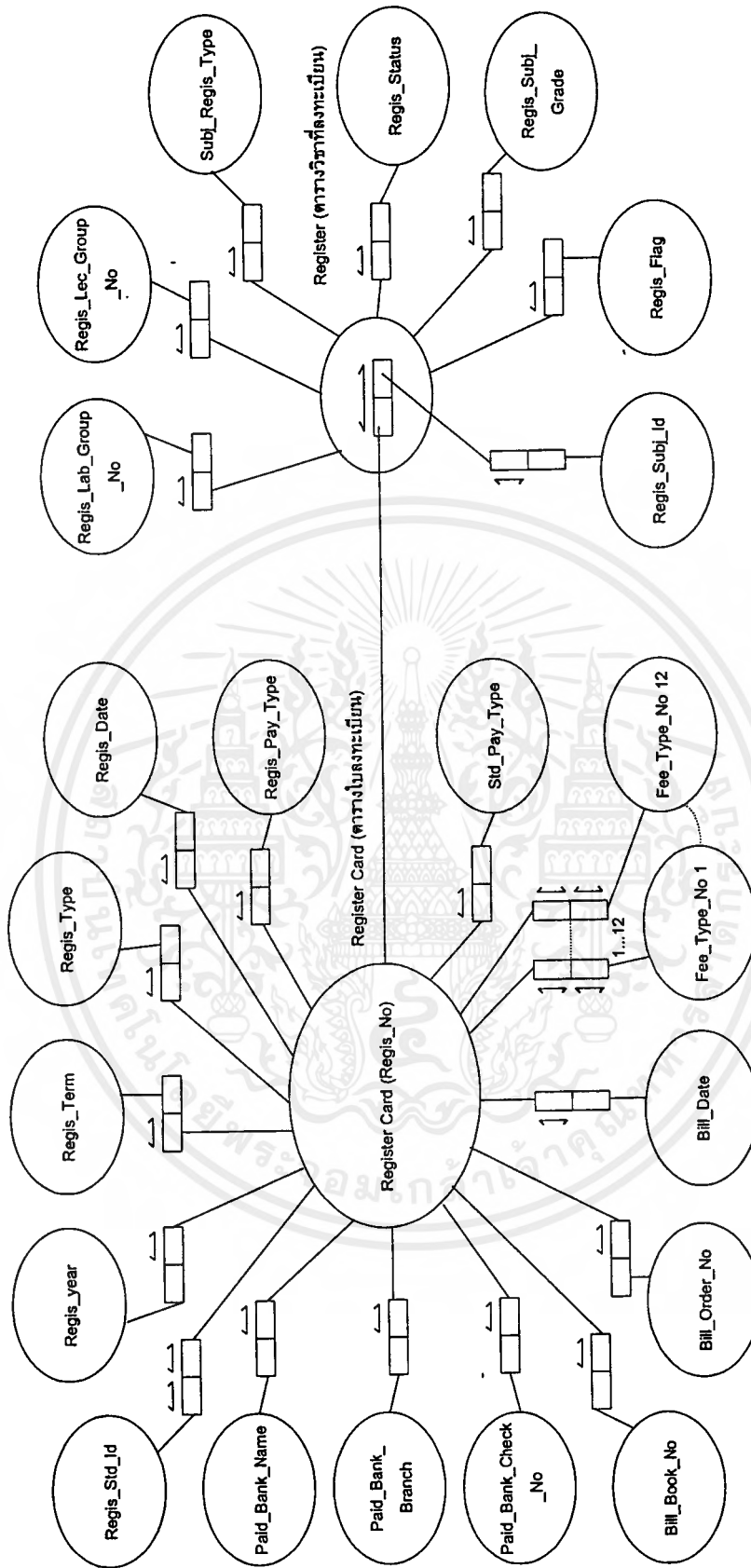
ในส่วนของการออกแบบฐานข้อมูลนั้นจะพิจารณาจากฐานข้อมูลของโครงการแล้วว่ายังมีสิ่งใดที่จะต้องเพิ่มเติมบ้าง เพื่อให้เหมาะสมกับระบบใหม่ ซึ่งฐานข้อมูลของโครงการเก่าจะประกอบไปด้วยตารางที่เกี่ยวกับการลงทะเบียนและชำระค่าธรรมเนียม

ส่วนที่เพิ่มเติมเข้ามาในโครงการนี้ก็คือ

ตารางของวิชาต่อเนื่อง (Prerequisite Subject) ที่เก็บข้อมูลของวิชาที่จำเป็นต้องเรียนอย่างค่อนเนื่อง ตารางนี้สร้างขึ้นเพื่อรักษาความถูกต้องของกฎกติกาในงานด้านการลงทะเบียนเรียน ตัวอย่างเช่น ถ้า นักศึกษาไม่ได้เรียนหรือไม่ผ่านวิชา MATH1 มากี่ไม่สามารถที่จะลงเรียน วิชา MATH2 ได้ เป็นต้น

ตาราง Temptable อันนี้ทำมาเพื่ออำนวยความสะดวกให้กับระบบ ในการที่จะใช้เก็บค่าปัจจุบันของ Serial Number ของใบลงทะเบียนและเมื่อมีนักศึกษามาลงทะเบียน ระบบนี้ก็จะมีการ Increase ค่า Serial Number ขึ้นอีก 1 โดยอัตโนมัติ

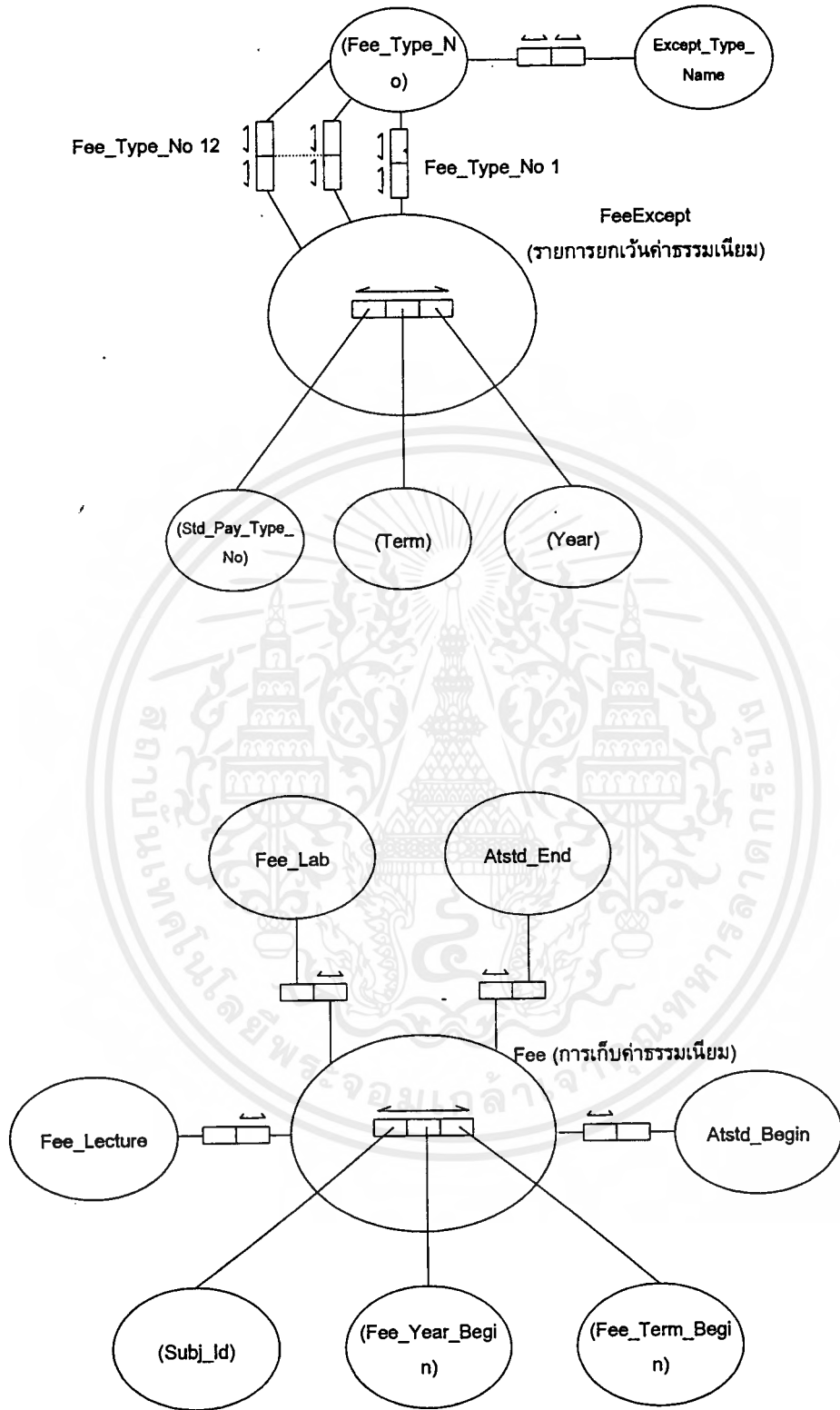
ตารางที่บ่งบอกประเภทของรายวิชาซึ่งแบ่งเป็นกลุ่มวิชาพื้นฐานและกลุ่มวิชาเฉพาะ



การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM

ส่วนของการลงทะเบียน

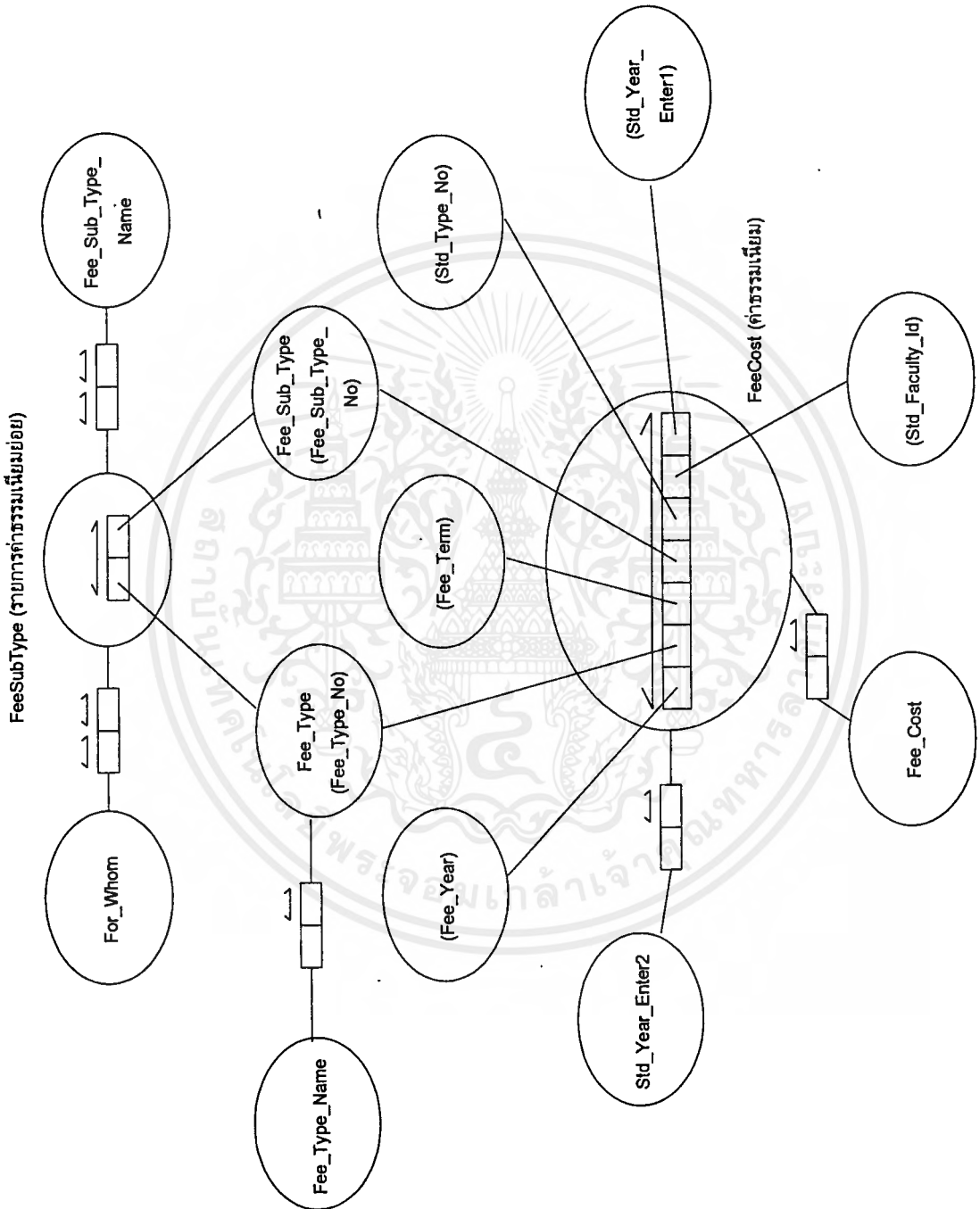
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM

ส่วนของการเก็บค่าธรรมเนียม (1)

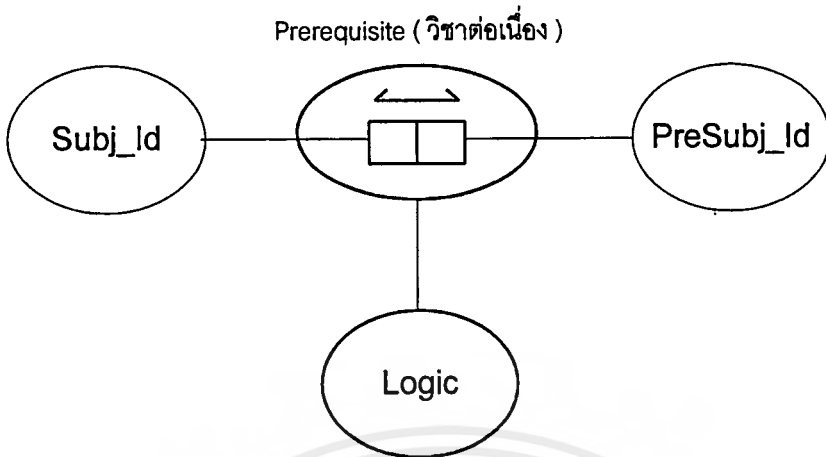
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



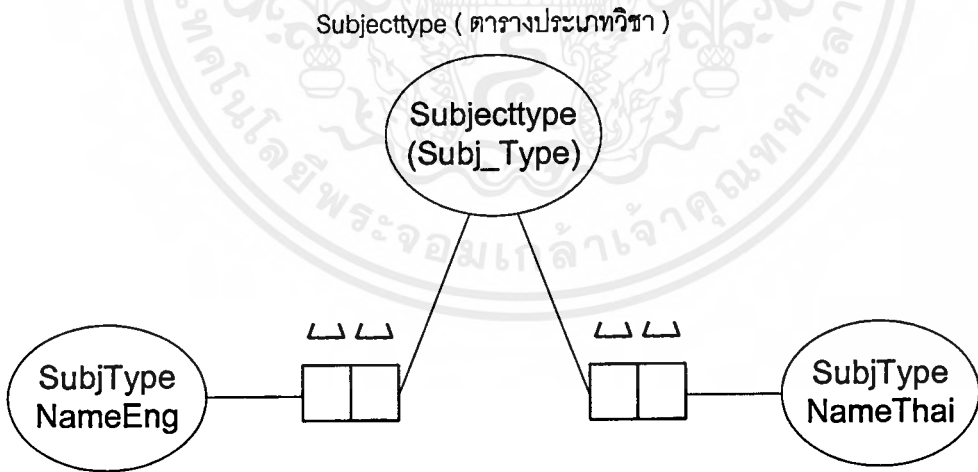
การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM

ส่วนของการเก็บค่าธรรมเนียม (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM
ส่วนของวิชาต่อเนื่อง



การออกแบบฐานข้อมูลเชิงสัมพันธ์โดยใช้ NIAM
ส่วนของตารางประเภทวิชา

ตารางระบบชำระค่าธรรมเนียม

Fee (การเก็บค่าธรรมเนียม)

Column Name	Datatype	Size	Key
Subj_Id	char	8	pk
Fee_Year_Begin	char	4	pk
Fee_Term_Begin	char	1	pk
Fee_Lecture	int	4	
Fee_Lab	int	4	
Atstd_Begin	char	4	
Atstd_End	char	4	

FeeCost (ค่าธรรมเนียม)

Column Name	Datatype	Size	Key
Fee_Type_No	int	4	pk
Fee_Year	char	4	pk
Fee_Term	char	4	pk
Fee_Sub_Type_No	int	4	pk
Std_Type_No	char	2	pk
Std_Faculty_Id	char	2	pk
Std_Year_Enter1	char	4	pk
Std_Year_Enter2	char	4	
Fee_Cost	int	4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FeeSubType (รายการค่าธรรมเนียมย่อย)

Column Name	Datatype	Size	Key
Fee_Type_No	int	4	pk
Fee_Sub_Type_No	int	4	pk
Fee_Sub_Type_Name	char	50	
For_Whom	char	50	

FeeExcept (รายการยกเว้นค่าธรรมเนียม)

Column Name	Datatype	Size	Key
Std_Pay_Type_No	char	4	pk
Term	char	1	pk
Year	char	4	pk
Fee_Type_No1	int	4	
::	::	:	
Fee_Type_No12	int	4	

FeeExceptName (ชื่อรายการยกเว้นค่าธรรมเนียม)

Column Name	Datatype	Size	Key
Std_Except_Type_No	char	4	pk
Except_Type_Name	char	50	

FeeType (ชนิดค่าธรรมเนียม)

Column Name	Datatype	Size	Key
Fee_Type_No	int	4	pk
Fee_Type_Name	char	50	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางระบบการลงทะเบียน

Register (ตารางวิชาที่ลงทะเบียน)

Column Name	Datatype	Size	Key
Regis_No	int	4	pk
Regis_Subj_Id	char	8	pk
Regis_Lab_Group_No_char	char	2	
Regis_Lec_Group_No_char	char	2	
Subj_Regis_Type	char	1	
Regis_Status	char	1	
Regis_subj_Grade	char	2	
Regis_Flag	char	1	

RegisterCard (ตารางใบลงทะเบียน)

Column Name	Datatype	Size	Key
Regis_No	int	4	pk
Regis_Std_Id	char	8	
Regis_Year	char	4	
Regis_Term	char	1	
Regis_Type	char	1	
Regis_Date	datetime	8	
Regis_Pay_Type	char	50	
Std_Pay_Type	char	50	
Bill_Book_No	char	4	
Bill_Date	datetime	8	
Fee_Type1	int	4	
::	::	:	
Fee_Type12	int	4	
Paid_Bank_Name	char	20	
Paid_Bank_Branch	char	30	
Paid_Bank_Check_No	int	4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Prerequisite (วิชาต่อเนื่อง)

Column Name	Datatype	Size	Key
Subj_Id	char	8	pk
PreSubj_Id	char	8	pk
Logic	char	1	

SubjectType (ประเภทวิชา)

Column Name	Datatype	Size	Key
Subj_Type	char	8	pk
SubjTypeNameEng	char	50	
SubjTypeNameThai	char	50	

TempTable (ตาราง Temp)

Column Name	Datatype	Size	Key
Term	char	1	
Year	char	4	
Serial_No	int	4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.6 ลำดับขั้นการทำงานของระบบ

8.6.1 ขั้นตอนการ Login เพื่อเข้าใช้ระบบ

1. ผู้ขอใช้บริการทำการเปิดเบราว์เซอร์ และเรียกใช้ระบบโดยพิมพ์ที่อยู่ลงใน URL
2. เมื่อสามารถทำการติดต่อกับ Server ได้สำเร็จทาง Server จะส่งเพจแรกซึ่งจะเป็นแบบฟอร์มสำหรับการ Login เพื่อเข้าใช้ระบบนี้มาให้
3. เมื่อผู้ใช้ทำการกรอก Login ID และ รหัสผ่าน (Password) ลงในฟอร์มและกดปุ่มตกลง (Submit) แล้วข้อมูลจะถูกตรวจสอบเบื้องต้นโดย Java Script ที่รวมอยู่ในโค้ด HTML การตรวจสอบก็เช่นว่า ผู้ใช้ไม่สามารถกรอกข้อมูลที่มีชนิดของข้อมูลผิดไปจากที่ระบบต้องการได้ เป็นต้น ก่อนที่จะส่งไปยัง Server ในลักษณะของ Post Method ซึ่งข้อมูลที่ส่งไปจะอยู่ในลักษณะ Hyper Text Transfer Protocol (HTTP)
4. เมื่อด้าน Server ได้รับข้อมูลเข้ามาจะไปเรียก Script IDC ที่ระบุมาที่ฟอร์ม HTML ที่ส่งมาจากผู้ใช้ หรือ Client ซึ่ง Script IDC นั้นจะทำหน้าที่คล้ายกับ CGI คือจะเอ็กรหัสคำสั่ง SQL ที่อยู่ใน IDC นี้ และจะมีการเรียกใช้งานสตอร์โพซิเจอร์เพื่อตรวจสอบการ Login ของผู้ใช้ ซึ่งผลลัพธ์จะเป็นการรีเทิร์นค่ารหัสตัวเลข Integer โดยจะนำค่านี้มาเปรียบเทียบกับผู้ใช้สามารถที่จะเข้าใช้ระบบได้หรือไม่
5. ถ้าผู้ใช้มีสิทธิเข้าใช้ระบบได้ ก็จะมีเมนูให้เลือกทำการลงทะเบียนเรียน, เพิ่มวิชาเรียน, ถอนวิชาเรียน, เปลี่ยนวิชาเรียน และการตรวจสอบ ตามรายการหลักของระบบ

8.6.2 ขั้นตอนการลงทะเบียนเรียน

1. หลังจากกดปุ่มลงทะเบียนเรียนแล้วข้อมูลของผู้ใช้ซึ่งก็คือ Std_Id จะถูกส่งไปยังฝั่ง Server
2. Script IDC ก็จะนำข้อมูลนี้ไปใช้สำหรับคำสั่ง SQL เพื่อที่จะ query ข้อมูลที่เกี่ยวกับวิชาเรียนตามรหัสนักศึกษา และข้อมูลที่ได้นั้นเป็นรายวิชาที่นักศึกษานั้นมีสิทธิ์ลงทะเบียนได้จะถูกนำมารวม (merge) กับไฟล์ . HTX ซึ่งเป็นแบบฟอร์มในรูป HTML แล้วส่งกลับไปยังฝั่ง Client
3. ผู้ใช้จะเลือกรายวิชาที่ต้องการได้โดยคลิกที่ Check Box หน้ารหัสวิชานั้น
4. เมื่อกดปุ่มตกลง (Submit) จะหมายถึง การยืนยันการลงทะเบียน ตามข้อมูลรายวิชา ที่ถูกเลือกเพื่อทำการส่งไปยังฝั่ง Server อีกครั้งหนึ่ง
5. เมื่อด้าน Server ได้รับข้อมูลรหัสวิชาแล้วจะมีการเรียกใช้สตอร์โพซิเจอร์ที่เกี่ยวกับการลงทะเบียน ซึ่งทำหน้าที่ Insert รายวิชาลงตารางการลงทะเบียน และทำการคิดคำนวณค่าธรรมเนียมทั้งหมดที่จะต้องชำระ นอกจากนี้ยังมีการตรวจสอบกฎข้อบังคับต่าง ๆ ด้วย อาทิเช่น ตรวจสอบวิชาต่อเนื่อง (Prerequisite) เป็นต้น
6. หลังจากที่มีการนำข้อมูลเกี่ยวกับการลงทะเบียนไปลงในตารางแล้ว ก็จะมีการ query ข้อมูลเหล่านั้นขึ้นมาเพื่อส่งกลับไปยังฝั่ง Client ทำให้ผู้ใช้ได้ทราบว่าได้ลงทะเบียนเรียนวิชาอะไรไป

แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.6.3 ขั้นตอนการเปลี่ยนวิชาเรียน

1. หลังจากกดปุ่มเปลี่ยนแปลงแล้วข้อมูลของผู้ใช้ซึ่งก็คือ Std_Id จะถูกส่งไปยังฝั่ง Server
2. Script IDC จะนำข้อมูลนี้ไปใช้สำหรับคำสั่ง SQL เพื่อที่จะ query ข้อมูลที่เกี่ยวกับวิชาเรียนตามรหัส นักศึกษานั้น ที่ได้ทำการลงทะเบียนไปแล้ว ข้อมูลที่ได้จะถูกนำมารวม (merge) กับไฟล์ HTX ซึ่งเป็นแบบฟอร์มในรูปแบบ HTML แล้วส่งกลับไปยังฝั่ง Client
3. ผู้ใช้จะต้องป้อนรหัสวิชาที่ต้องการเปลี่ยนและรหัสวิชาใหม่ที่ต้องการจะลงเรียนลงใน Text Box
4. เมื่อกดปุ่มตกลง (Submit) จะหมายถึง การยืนยันการเปลี่ยนวิชา ข้อมูลใน Text Box จะถูกตรวจสอบเบื้องต้นโดย Java Script ที่รวมอยู่ในโค้ด HTML ข้อมูลรายวิชา (รหัสวิชา) เหล่าจะถูกส่งไปยังฝั่ง Server อีกครั้งหนึ่ง
5. เมื่อด้าน Server ได้รับข้อมูลรหัสวิชาแล้วจะมีการเรียกใช้สคริปต์ไพอีเซอร์ที่เกี่ยวข้องกับการเปลี่ยนแปลงวิชาเรียน ซึ่งทำหน้าที่ Update รายวิชาลงตารางการลงทะเบียน นอกจากนี้ยังมีการตรวจสอบกฎข้อบังคับต่างๆด้วยอาทิเช่น ตรวจสอบว่าวิชาใหม่ที่จะเปลี่ยนนั้นนักศึกษาสามารถเรียนได้หรือไม่, ตรวจสอบวิชาต่อเนื่อง (Prerequisite) เป็นต้น
6. หลังจาก Update ตารางแล้ว ก็จะมีการ query ข้อมูลเหล่านั้นขึ้นมาเพื่อส่งกลับไปยังฝั่ง Client ทำให้ผู้ใช้ได้ทราบว่าได้เปลี่ยนวิชาและลงทะเบียนเรียนวิชาอะไรไปแล้ว

8.6.4 ขั้นตอนการเพิ่มวิชาเรียน

1. หลังจากกดปุ่มเพิ่มวิชาเรียนแล้วข้อมูลของผู้ใช้ซึ่งก็คือ Std_Id จะถูกส่ง ไปยังฝั่ง Server
2. Script IDC จะนำข้อมูลนี้ไปใช้สำหรับคำสั่ง SQL เพื่อที่จะ query ข้อมูลที่เกี่ยวกับวิชาเรียนที่นักศึกษา ที่ได้ทำการลงทะเบียน ไปแล้ว ข้อมูลที่ได้จะถูกนำมารวม (merge) กับไฟล์ HTX ซึ่งเป็นแบบฟอร์ม ในรูปแบบ HTML แล้วส่งกลับไปยังฝั่ง Client
3. ผู้ใช้จะต้องป้อนรายวิชาที่ต้องการเพิ่มลงใน Text Box
4. เมื่อกดปุ่มตกลง (Submit) จะหมายถึง การยืนยันการเพิ่มวิชา ข้อมูลรายวิชา (รหัสวิชา) ที่ต้องการเพิ่มจะถูกส่งไปยังฝั่ง Server อีกครั้งหนึ่ง
5. เมื่อด้าน Server ได้รับข้อมูลรหัสวิชาแล้วจะมีการเรียกใช้สคริปต์ไพอีเซอร์ที่เกี่ยวข้องกับการเพิ่มวิชาเรียน ซึ่งทำหน้าที่ Insert รายวิชาลงตารางการลงทะเบียน และทำการกีดคำนวณค่าธรรมเนียมที่จะต้องชำระ นอกจากนี้ยังมีการตรวจสอบกฎข้อบังคับต่างๆด้วยอาทิเช่น ตรวจสอบว่าวิชานั้นเป็นวิชาของคณะเดียวกันหรือไม่, ตรวจสอบวิชาต่อเนื่อง (Prerequisite) เป็นต้น
6. หลังจากที่มีการนำข้อมูลที่เกี่ยวข้องกับเพิ่มวิชาไปลงในตารางแล้ว ก็จะมีการ query ข้อมูลเหล่านั้นขึ้นมา เพื่อส่งกลับไปยังฝั่ง Client ทำให้ผู้ใช้ได้ทราบว่าได้เพิ่มวิชาหรือลงทะเบียนเรียนวิชาอะไรไปแล้ว

8.6.5 ขั้นตอนการถอนวิชาเรียน

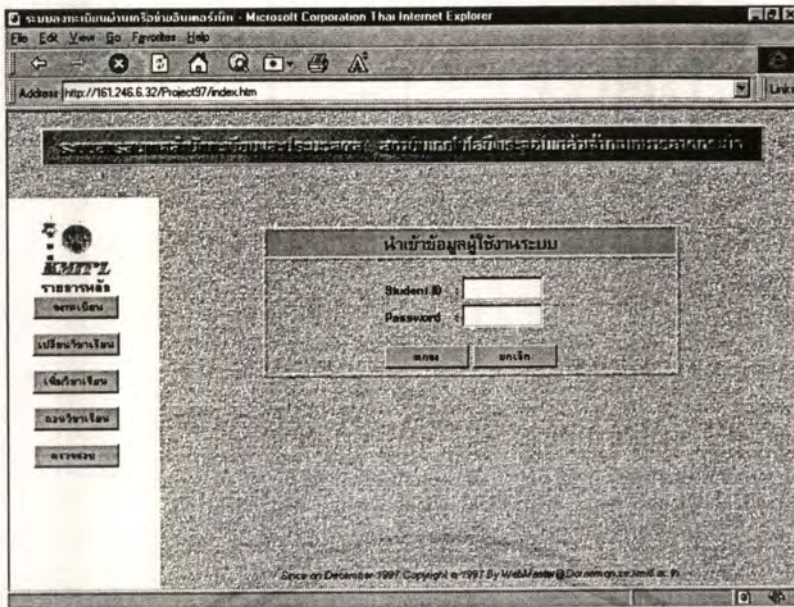
1. หลังจากกดปุ่มถอนวิชาเรียนแล้วข้อมูลของผู้ใช้ซึ่งก็คือ Std_Id จะถูกส่งไปยังฝั่ง Server
2. Script IDC จะนำข้อมูลนี้ไปใช้สำหรับคำสั่ง SQL เพื่อที่จะ query ข้อมูลที่เกี่ยวกับวิชาเรียนที่นักศึกษาได้ทำการลงทะเบียนไปแล้ว ข้อมูลที่ได้จะถูกนำมารวม (merge) กับไฟล์ HTX ซึ่งเป็นแบบฟอร์มในรูปแบบ HTML แล้วส่งกลับไปฝั่ง Client
3. ผู้ใช้จะเลือกรายวิชาที่ต้องการถอนได้โดยคลิกที่ Check Box หน้ารหัสวิชานั้น ไม่ให้มีเครื่องหมายถูก
4. เมื่อกดปุ่มตกลง (Submit) จะหมายถึง การยืนยันการถอนวิชา ข้อมูลรายวิชา (รหัสวิชา) ที่ต้องการจะถอนถูกส่งไปยังฝั่ง Server อีกครั้งหนึ่ง.
5. เมื่อด้าน Server ได้รับข้อมูลรหัสวิชาแล้วจะมีการเรียกใช้สคริปต์ไพอซีเอร์ที่เกี่ยวกับการถอนวิชาเรียนซึ่งทำหน้าที่ Delete รายวิชาที่ต้องการถอน
6. หลังจากทำการลบรายวิชาในตารางแล้ว ก็จะมีการ query ข้อมูลเหล่านั้นขึ้นมาเพื่อส่งกลับไปฝั่ง Client ทำให้ผู้ใช้ได้ทราบว่าได้ถอนวิชาหรือลงทะเบียนเรียนวิชาอะไรไปแล้ว

8.6.6 การตรวจสอบ

1. หลังจากกดปุ่มตรวจสอบแล้วจะมีการแสดงรายการของสิ่งที่จะทำการตรวจสอบมีดังนี้
2. ตรวจสอบการลงทะเบียน
3. ตรวจสอบการชำระค่าธรรมเนียม
4. ตรวจสอบผลการเรียน
5. การตรวจสอบผลการเรียนและการชำระค่าธรรมเนียมนั้น ผู้ใช้ทำได้โดยปุ่มของการทำงานเหล่านั้น แต่สำหรับการตรวจสอบผลการเรียนนั้นผู้ใช้งานเป็นที่จะต้องกรอก ปีการศึกษาและภาคการศึกษาที่ต้องการทราบด้วย
6. เมื่อกดปุ่มเหล่านี้แล้วข้อมูลของผู้ใช้หรือรวมถึงข้อมูลที่ใช้ได้กรอกลงไปนั้นจะถูกส่งไปยัง Server Script IDC จะนำข้อมูลนี้ไปใช้สำหรับคำสั่ง SQL เพื่อที่จะ query ข้อมูลที่เกี่ยวกับการลงทะเบียนหรือ การชำระค่าธรรมเนียมหรือผลการเรียน ข้อมูลที่ได้จะถูกนำมารวม (merge) กับไฟล์ HTX ซึ่งเป็นแบบฟอร์มในรูปแบบ HTML แล้วส่งกลับไปฝั่ง Client

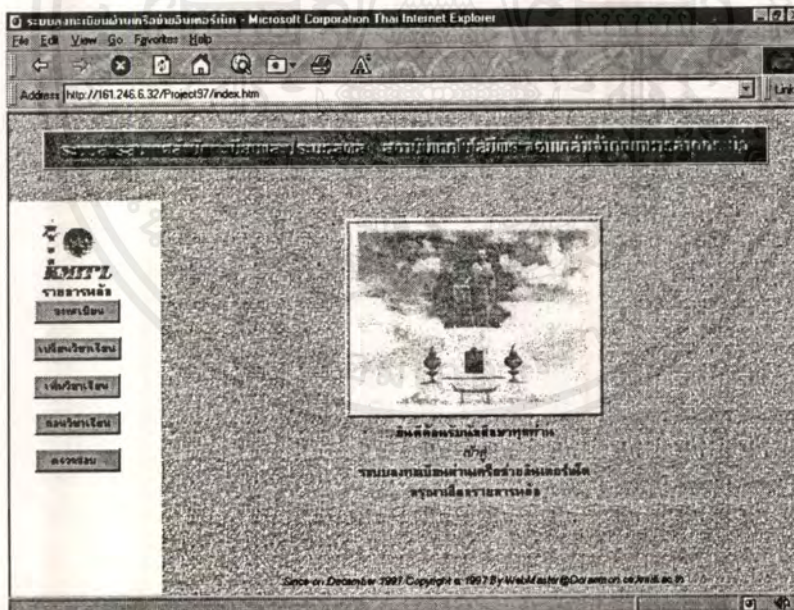
8.7 ลักษณะของหน้าจอส่วนที่ติดต่อกับผู้ใช้ (User Interface)

หลังจากที่ผู้ใช้ทำการเปิดบราวเซอร์และอ้างถึง URL ของระบบนี้แล้ว ก็จะได้พบกับเพจแรก (Home page) ซึ่งเป็นหน้าจอของการ Login เข้าสู่ระบบ ซึ่งจะมีลักษณะดังนี้



เพจแรกของระบบลงทะเบียน

จากนั้นเมื่อนักศึกษาป้อน Username และ Password แล้วกดปุ่มตกลง ข้อมูลจะถูกส่งไปตรวจสอบที่ฝั่ง Server ถ้าการ Login ผ่าน Server จะส่งเพจถัดไปกลับมาให้ดังรูป

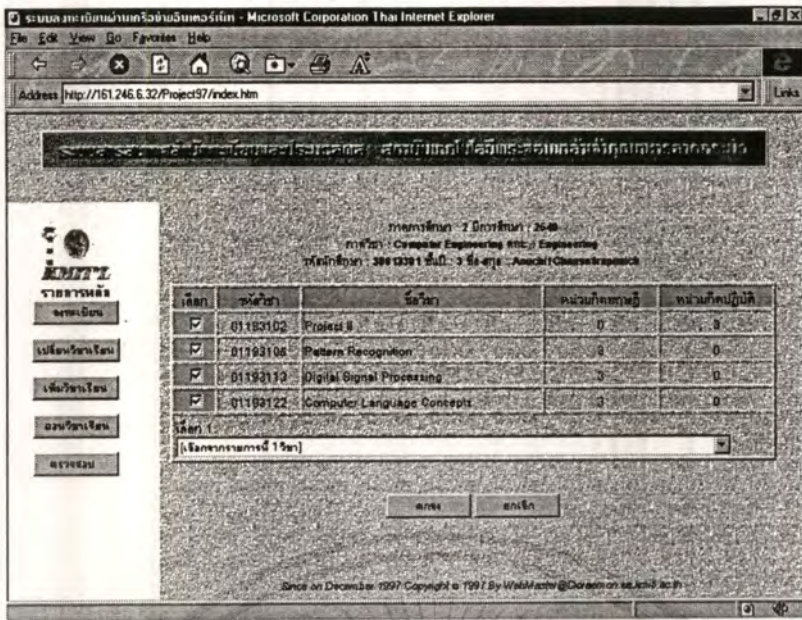


เมื่อ Login ผ่านแล้วระบบจะแสดงเพจนี้

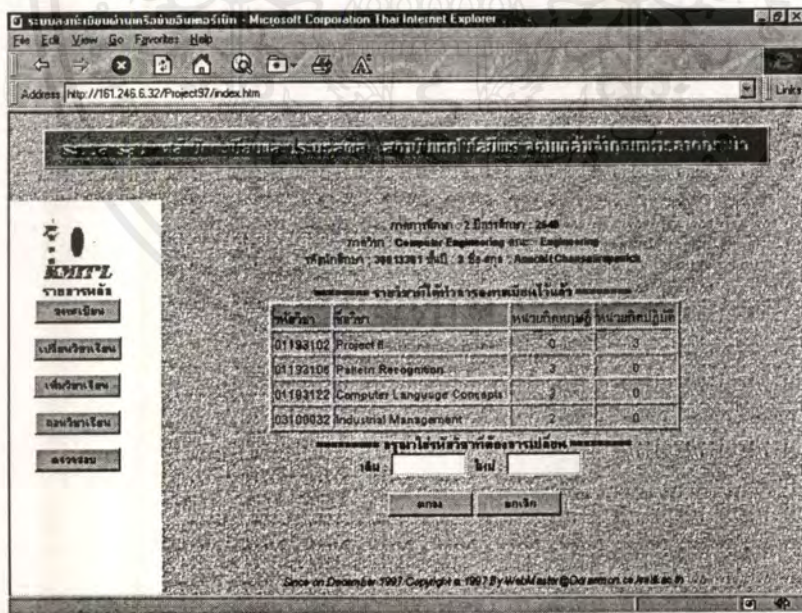
จากนั้นก็ให้กดปุ่มเลือกรายการที่ต้องการจากเมนูทางด้านซ้ายมือ ซึ่งผลที่ได้จากการกดปุ่มในแต่ละเมนูมีดังนี้

- เมื่อกดปุ่มลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



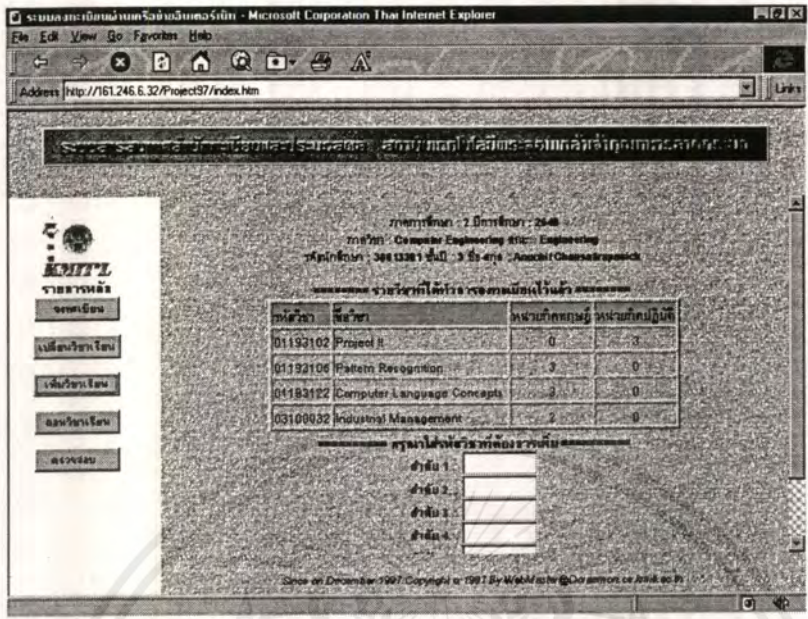
- เมื่อกดปุ่มเปลี่ยนวิชาเรียน



เพจของการเปลี่ยนแปลงรายวิชา

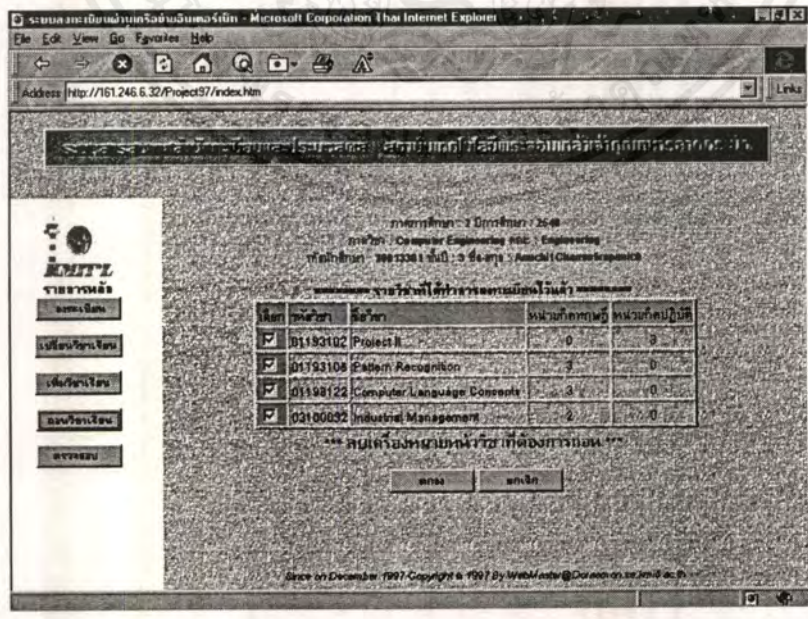
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• เมื่อกดปุ่มเพิ่มวิชาเรียน



เพจของการเพิ่มรายวิชา

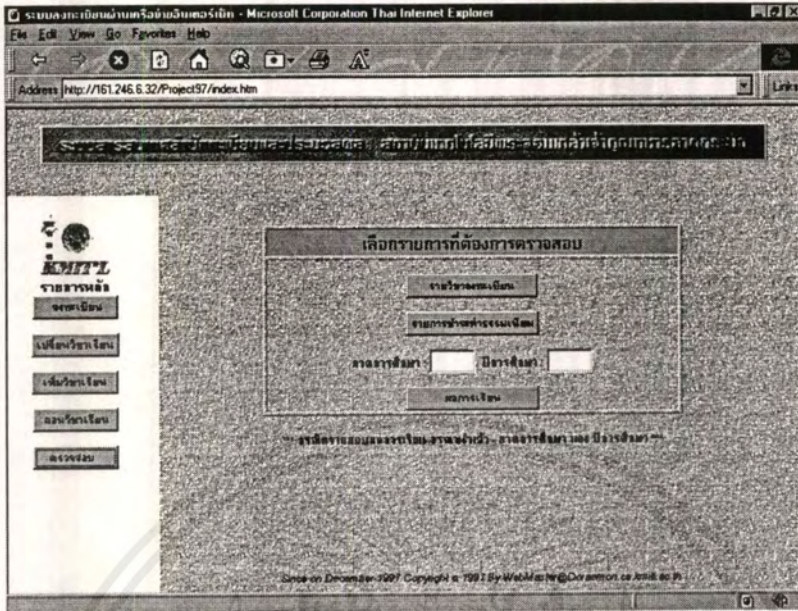
• เมื่อกดปุ่มถอนวิชาเรียน



เพจของการถอนรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อคลิกปุ่มตรวจสอบ



เพจของการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9

การทดลองและผลการทดลอง

9.1 ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการทดลอง

9.1.1 คอมพิวเตอร์ที่ทำหน้าที่ควบคุมระบบเครือข่าย

1. หน่วยประมวลผลกลาง (CPU) เพนเทียม 133 MHz
2. หน่วยความจำหลัก (RAM) 32 MB
3. หน่วยความจำชนิดความเร็วสูงภายใน (Internal Cache) ขนาด 32 KB
4. หน่วยความจำชนิดความเร็วสูงภายนอก (External Cache) ขนาด 512 KB
5. เครื่องอ่าน CD-ROM
6. Floppy Disk Drive ขนาด 3.5 นิ้ว สำหรับอ่านดิสก์ขนาด 1.44 MB
7. จอภาพสีขนาด 14 นิ้ว
8. Hard Disk Drive ขนาด 5 นิ้ว มีขนาดความจุ 1.2 MB
9. แป้นพิมพ์ (Keyboard) ที่มีอักษรภาษาไทย / อังกฤษ ตัวเลข และเครื่องหมายสัญลักษณ์พิเศษ 101 ปุ่ม
10. เม้าส์ชนิด 2 ปุ่มคำสั่ง
11. เครื่องสำรองไฟฟ้าฉุกเฉิน (UPS)

9.1.2 เครื่องไมโครคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องลูกข่าย

1. หน่วยประมวลผลกลาง (CPU) 486DX2 66 MHz
2. หน่วยความจำหลัก (RAM) 16 MB
3. หน่วยความจำชนิดความเร็วสูงภายใน (Internal Cache) ขนาด 16 KB
4. หน่วยความจำชนิดความเร็วสูงภายนอก (External Cache) ขนาด 256 KB
5. จอภาพสีขนาด 14 นิ้ว
6. Floppy Disk Drive ขนาด 3.5 นิ้ว สำหรับอ่านดิสก์ขนาด 1.44 MB
7. แป้นพิมพ์ (Keyboard) ที่มีอักษรภาษาไทย / อังกฤษ ตัวเลขและเครื่องหมายสัญลักษณ์พิเศษ จำนวน 101 ปุ่ม
8. เม้าส์ชนิด 2 ปุ่มคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.2 ความต้องการทางด้านซอฟต์แวร์

9.2.1 คอมพิวเตอร์ที่ทำหน้าที่ควบคุมระบบเครือข่าย

1. ระบบปฏิบัติการไมโครซอฟท์ วินโดวส์เอ็นที เวอร์ชัน 4 ภาษาไทย (Microsoft Windows NT 4.0 Thai Enable)
2. ระบบจัดการการบริการข้อมูลข่าวสารทางอินเทอร์เน็ต เวอร์ชัน 3 (Internet Information Server 3.0)
3. ระบบจัดการฐานข้อมูลไมโครซอฟท์เอสคิวแอลเซิร์ฟเวอร์ เวอร์ชัน 6.5 (Microsoft SQL Server 6.5)

9.2.2 คอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องลูกข่าย

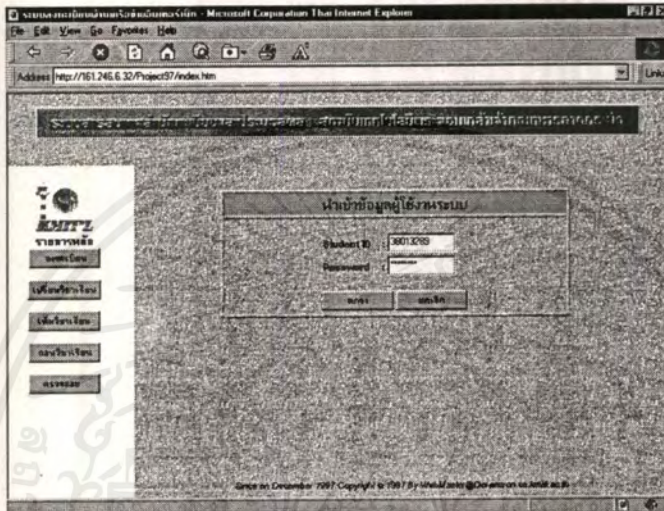
1. ระบบปฏิบัติการไมโครซอฟท์วินโดวส์ 95 ภาษาไทย (Microsoft Windows 95 Thai Edition)
2. โปรแกรมค้นหาข้อมูลในอินเทอร์เน็ต (Web Browser) ไมโครซอฟท์ อินเทอร์เน็ต เอ็กพลอเรอร์ เวอร์ชัน 3.02 (Microsoft Internet Explorer 3.02) และ เน็ตสเคปเนวิเกเตอร์ เวอร์ชัน 3.0 (Netscape Navigator 3.0)

9.3 การทดลองและผลการทดลอง

9.3.1 การเข้าใช้งานระบบ

ขั้นตอนทดลอง

1. เปิดเว็บเบราว์เซอร์ และทำการใส่ข้อมูลผู้ใช้งาน ที่เป็น Username และ Password
2. กดปุ่มตกลงเมื่อข้อมูลที่นำเข้าถูกต้อง



รูปแสดงหน้าจอรอข้อมูลนำเข้ามาจากผู้ใช้งาน

ผลการทดลอง

เมื่อข้อมูลถูกต้องจะได้รับเพจแสดงดังนี้



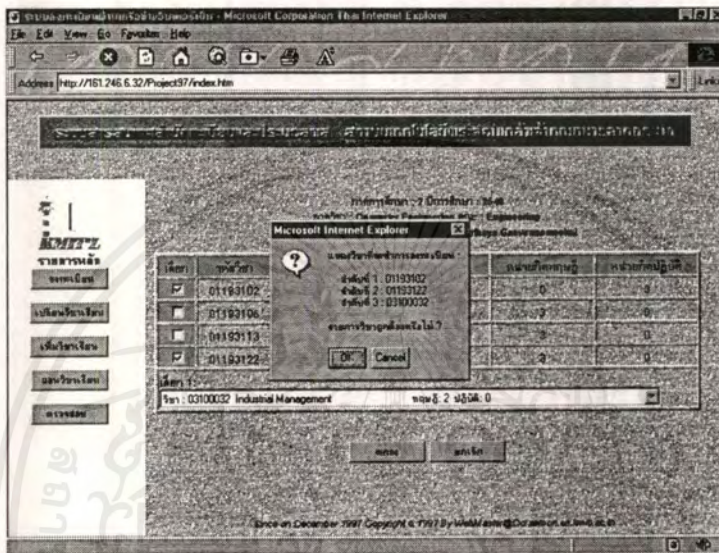
รูปแสดงหน้าจอที่ผ่านการตรวจสอบการเข้าใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.3.2 การลงทะเบียนวิชาเรียน

ขั้นตอนการทดลอง

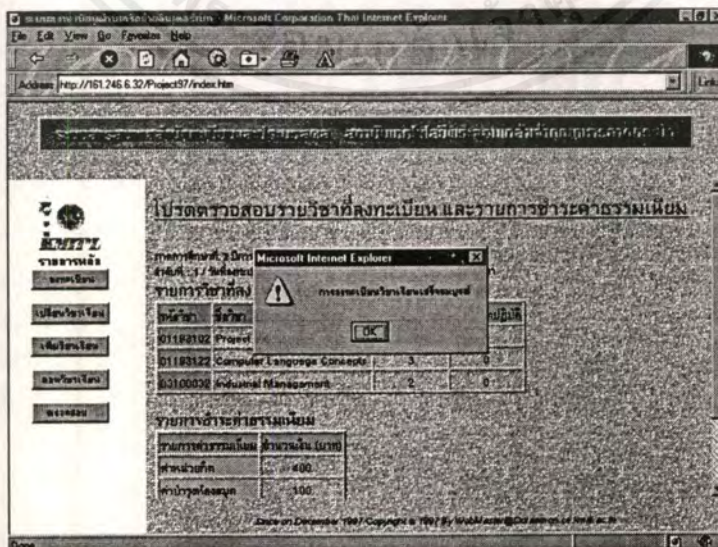
1. กดปุ่มลงทะเบียน
2. ทำการเลือกรายวิชาที่ต้องการลงทะเบียน
3. กดปุ่มตกลงเมื่อการเลือกเสร็จสมบูรณ์
4. ตรวจสอบรายการ และกดปุ่มตกลงเพื่อยืนยันรายวิชาที่เลือกไว้



ภาพแสดงหน้าจอยืนยันการเลือกรายวิชา

ผลการทดลอง

หลังจากการลงทะเบียนแล้วจะแสดงเพจของรายวิชาที่เลือกไว้ และรายการค่าธรรมเนียมดังนี้



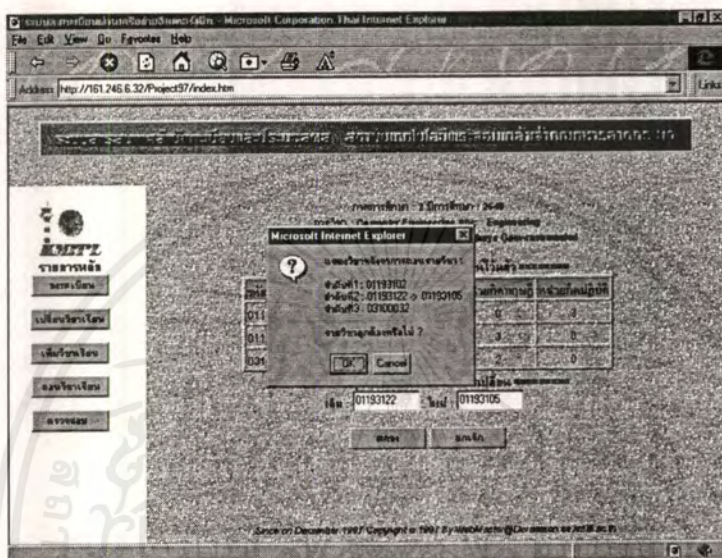
ภาพแสดงหน้าจอหลังการยืนยันการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.3.3 การเปลี่ยนรายวิชา

ขั้นตอนการทดลอง

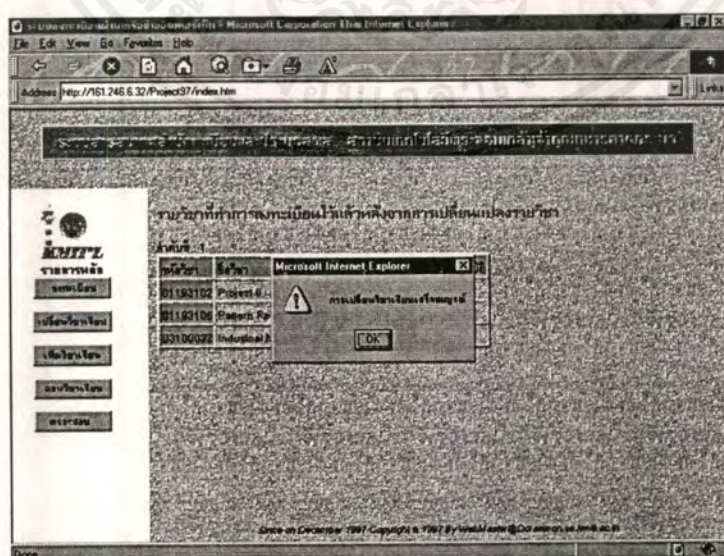
1. กดปุ่มเปลี่ยนวิชาเรียน
2. ใส่รหัสวิชาเดิมและรหัสวิชาใหม่ที่ต้องทำการเปลี่ยน
3. กดปุ่มตกลงเมื่อการใส่ข้อมูลรหัสวิชาเสร็จสมบูรณ์
4. ตรวจสอบรายการ และกดปุ่มตกลงเพื่อยืนยันรายวิชาที่ต้องการเปลี่ยน



ภาพแสดงหน้าจอการยืนยันการเปลี่ยนรายวิชา

ผลการทดลอง

เมื่อการเปลี่ยนวิชาแล้วเสร็จก็จะได้รับหน้าจอแสดงรายการวิชาที่เปลี่ยนไป



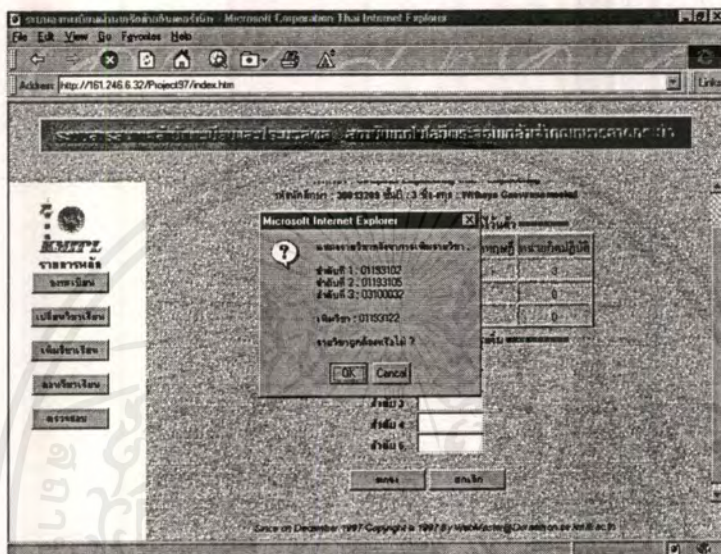
ภาพแสดงหน้าจอหลังการเปลี่ยนแปลงวิชาเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.3.4 การเพิ่มวิชาเรียน

ขั้นตอนการทดลอง

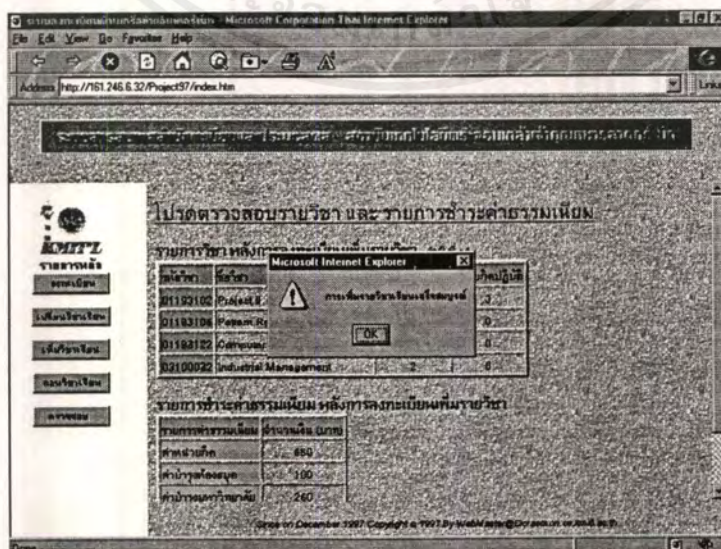
1. กดปุ่มเพิ่มวิชาเรียน
2. ใส่รหัสวิชาใหม่ที่จะทำการเพิ่ม
3. กดปุ่มตกลงเมื่อการใส่ข้อมูลรหัสวิชาเสร็จสมบูรณ์
4. ตรวจสอบรายการ และกดปุ่มตกลงเพื่อยืนยันรายวิชาที่ต้องการเพิ่ม



ภาพแสดงหน้าจอการยืนยันการเพิ่มวิชาเรียน

ผลการทดลอง

เมื่อทำการ ใส่รหัสวิชาที่จะทำการเพิ่มและกดปุ่มตกลงเพื่อยืนยันแล้ว จะ ได้รับเพจดังนี้



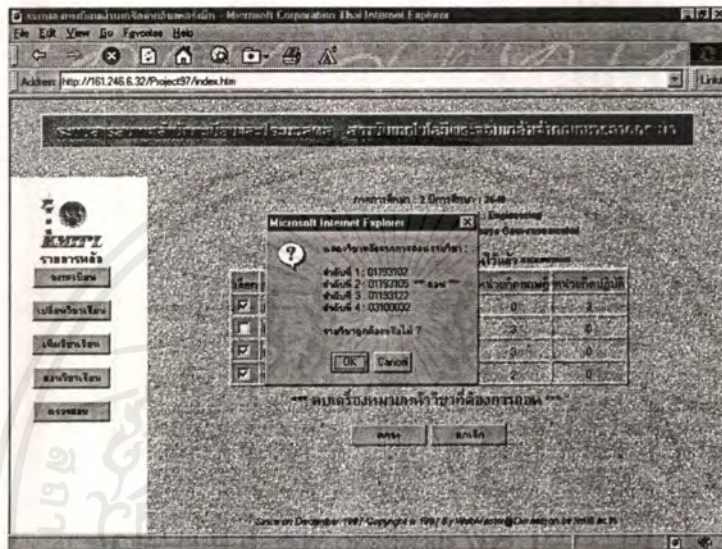
ภาพแสดงหน้าจอหลังการเพิ่มรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.3.5 การถอนวิชาเรียน

ขั้นตอนการทดลอง

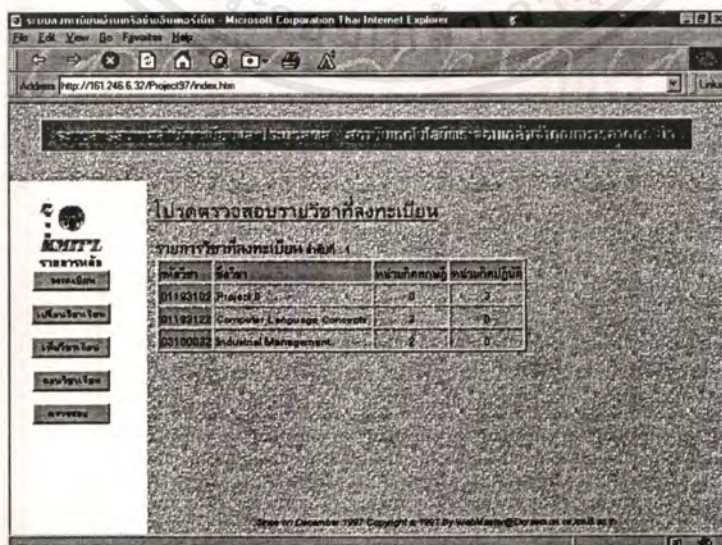
1. กดปุ่มถอนวิชาเรียน
2. นำเครื่องหมายหน้าวิชาที่จะทำการถอนออก
3. กดปุ่มตกลงเมื่อการถอนวิชาเสร็จสมบูรณ์
4. ตรวจสอบรายการ และกดปุ่มตกลงเพื่อยืนยันรายวิชาที่ต้องการถอน



ภาพแสดงหน้าจอยืนยันการถอนวิชาเรียน

ผลการทดลอง

เมื่อการถอนวิชาเสร็จสมบูรณ์จะได้รับเพจแสดงรายวิชาที่ลงทะเบียนที่ยังเหลืออยู่



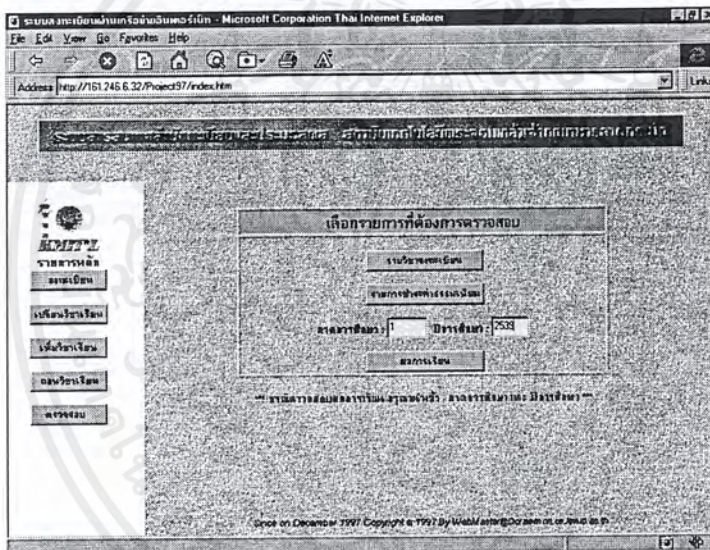
ภาพแสดงหน้าจอหลังการถอนรายวิชาเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.3.6 การตรวจสอบ

ขั้นตอนการทดลอง

1. กดปุ่มตรวจสอบ
2. กดปุ่มตรวจสอบ รายวิชาเรียนที่ลงทะเบียน เมื่อต้องการตรวจสอบรายวิชาที่ทำการลงทะเบียนไว้
3. กดปุ่มตรวจสอบ รายการชำระค่าธรรมเนียม เมื่อต้องการตรวจสอบรายการค่าธรรมเนียมที่ต้องชำระ
4. กดปุ่มตรวจสอบ ผลการเรียน เมื่อต้องการตรวจสอบผลการเรียนของแต่ละภาค / ปีการศึกษา
5. หากเป็นการตรวจสอบผลการเรียนต้องนำเข้าสู่ข้อมูลภาคการศึกษา และปีการศึกษาที่ต้องการให้แสดงด้วย

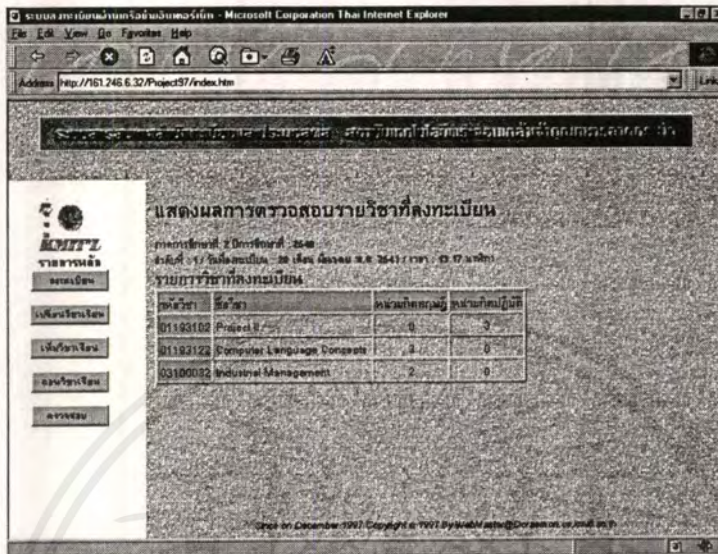


ภาพแสดงหน้าจอการตรวจสอบรายการต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

เมื่อทำการคปรุ่มตรวจสอบต่าง ๆ จะได้รับเพจแสดงดังนี้



แสดงผลการตรวจสอบรายวิชาที่ลงทะเบียน

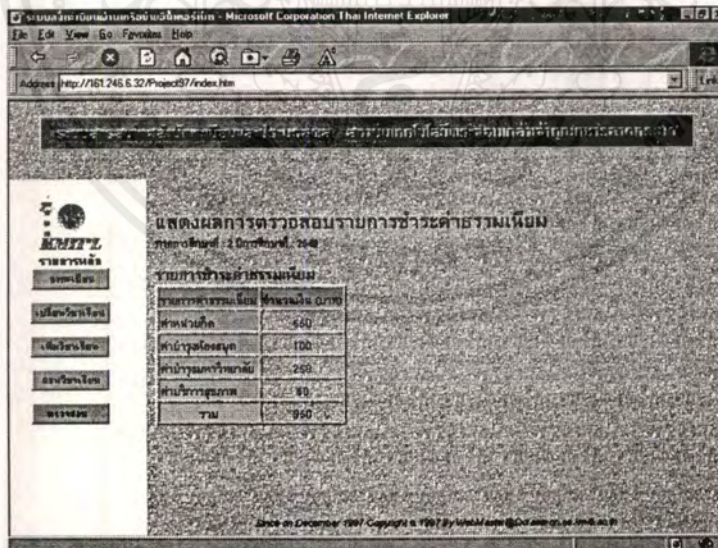
ภาคเรียนที่ : 2 ปีการศึกษา : 2548
วันที่ : 17 สิงหาคม 25 18:41 เวลา : 17:17 น.

รายการวิชาที่ลงทะเบียน

รหัสวิชา	ชื่อวิชา	คะแนนก่อนสอบ	คะแนนก่อนสอบ
01183102	Project II	0	5
01183123	Foreigner Language Concepts	4	0
03100032	Industrial Management	2	0

Print on December 1997 Copyright © 1997 By htmMam@Dorland.com.hk

ภาพแสดงหน้าจอหลังการตรวจสอบรายการวิชาที่ลงทะเบียน



แสดงผลการตรวจสอบรายการวิชาค่าธรรมเนียม

ภาคเรียนที่ : 2 ปีการศึกษา : 2548

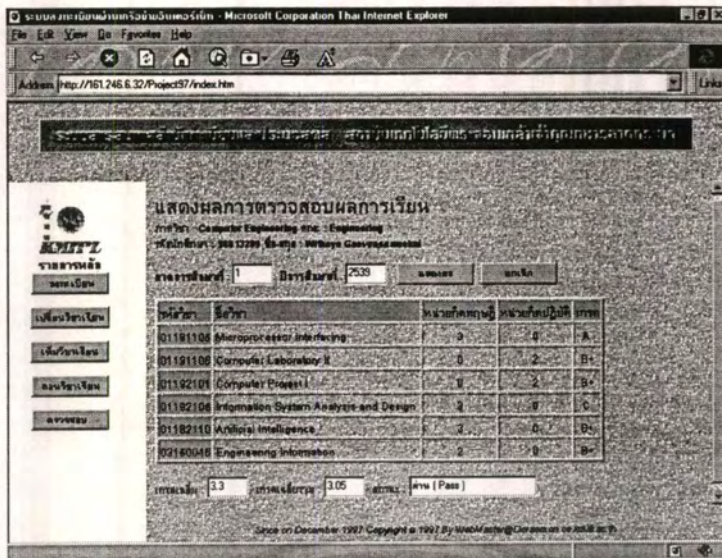
รายการวิชาค่าธรรมเนียม

รายการค่าธรรมเนียม	จำนวนเงิน บาท
ค่าหน่วยกิต	540
ค่าบำรุงสอนยุค	100
ค่าบำรุงมหาวิทยาลัย	250
ค่าบริหารสุขภาพ	10
รวม	960

Print on December 1997 Copyright © 1997 By htmMam@Dorland.com.hk

ภาพแสดงหน้าจอหลังการตรวจสอบรายการชำระค่าธรรมเนียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพแสดงหน้าจอหลังการตรวจสอบค่าธรรมเนียม

9.4 สรุปผลการทดลอง

ผลการทดลองสามารถรองรับการทำงานได้ดี สามารถตรวจสอบข้อมูลนำเข้าที่ผิดประเภทได้ สามารถตรวจสอบวิชาที่ลงได้ หรือลงไม่ได้ มีการตรวจสอบการลงทะเบียนซ้ำซ้อนได้ มีไดอะล็อกบ็อก (Dialog box) แจ้งเตือนถึงสถานะการทำงานขณะนั้นอยู่ตลอดเวลา และสามารถที่จะแสดงผลการเรียนในแต่ละภาค / ปี การศึกษาใด ๆ ได้ อย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 10

สรุป วิจารณ์ และแนวทางในการพัฒนา

10.1 ความสามารถของระบบ

1. สามารถทำการลงทะเบียนแบบ Online ได้ตลอดเวลาที่ต้องการ
2. มีความสะดวกในการปรับเปลี่ยน, เพิ่ม, ลด วิชาเรียน
3. สามารถตรวจสอบผลการเรียนในทุก ๆ ภาคการศึกษา แบบ Online ได้
4. รองรับการขอใช้บริการการลงทะเบียนของนักศึกษาที่เข้ามาทำการลงทะเบียนได้เป็นจำนวนมาก
5. สามารถแสดงรายวิชาที่ทำการลงทะเบียนในแต่ละภาคการศึกษานั้น ๆ ได้
6. สามารถที่จะทำการตรวจสอบ การลงทะเบียน, เปลี่ยน, เพิ่ม, ลด วิชาเรียน ให้เป็นไปตาม กฎข้อบังคับของสถาบันได้
7. มีการคำนวณค่าธรรมเนียมที่จะต้องชำระได้

10.2 ข้อจำกัดของระบบ

1. การจัดรูปแบบการแสดงผลบนเว็บเพจของบราวเซอร์แต่ละตัวที่ต่างค่ายกัน ยังไม่มีมาตรฐานที่แน่นอนทำให้การแสดงผลมีความแตกต่างกันออกไป
2. ในเรื่องของการชำระค่าธรรมเนียมระบบยังไม่สามารถที่จะแยก รายการชำระเงินได้ เช่น การเพิ่มวิชาเรียนจำเป็นที่จะต้องใช้การ Update ค่าหน่วยกิตลงในตาราง ทำให้เวลาแสดงผลไม่สามารถแสดงแบบแยกใบเสร็จได้
3. SQL Statement ใน Script IDC ยังมีความซ้ำซ้อนกันอยู่มาก ถ้าหากมีการแก้ไข SQL Statemnet เหล่านี้ก็ต้องตามไปแก้ไขในทุก ๆ ไฟล์ ที่ใช้ Statement แบบเดียวกัน
4. ใน Script IDC ไม่สามารถใช้ SQL Statement ในระดับ Advance ได้
5. การลงทะเบียนวิชายังไม่สามารถจัดแสดงการแบ่งกลุ่มการเรียนของวิชาเลือกได้
6. ยังไม่มีการจัดการเชื่อมต่อกับส่วนอื่น ๆ ที่มีอยู่ในระบบสารสนเทศของสำนักทะเบียนและประมวลผล

10.3 แนวทางการพัฒนาต่อ

1. เนื่องจากไฟล์ Script IDC-HTX ไม่เหมาะสมกับงานขนาดใหญ่ ที่มีความซับซ้อนมาก ทำให้ต้องมีการปรับเปลี่ยนเทคโนโลยีส่วนกลาง การจัดการข้อมูลใหม่
2. การตรวจสอบข้อบังคับ ระเบียบการต่าง ๆ ยังไม่รัดกุมพอ และขาดต่อการแก้ไข จะต้องทำการปรับปรุงให้มีความรัดกุมและถูกต้องมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.4 วิจารณ์โครงการ

โครงการระบบสารสนเทศของสำนักทะเบียนและประมวลผล ส่วนงานลงทะเบียนและค่าธรรมเนียมผ่านเครือข่ายอินเทอร์เน็ตนี้ ก่อนข้างจะมีความสมบูรณ์ในการใช้งานมากในระดับที่น่าพอใจ โดยที่สามารถที่จะรองรับกิจกรรมการลงทะเบียนของนักศึกษาได้เกือบทุกกรณีตามกฎหมายที่วางไว้ ไม่ที่จะเป็นการลงทะเบียน การเปลี่ยนแปลงวิชาเรียน การเพิ่มวิชาเรียน การถอนวิชาเรียน หรือแม้กระทั่ง การตรวจสอบต่าง ๆ และข้อมูลที่ได้จากการลงทะเบียนของนักศึกษา ก็สามารถนำไปใช้เพื่อการประมวลผล หรือการสืบค้นจากหน่วยงานอื่นได้

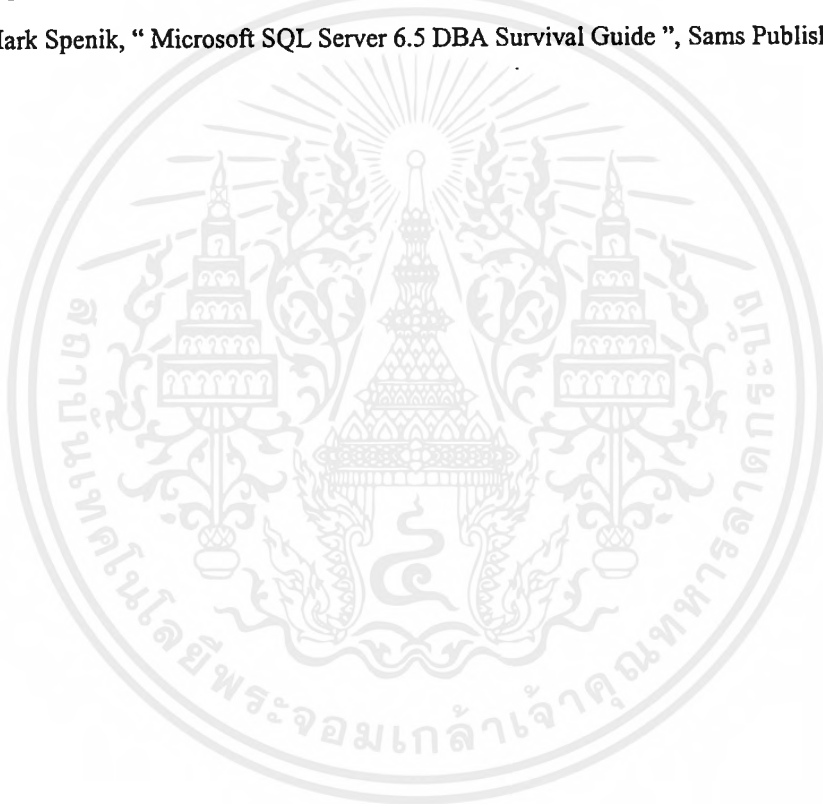
ในส่วนของ IIS นั้น เป็นระบบจัดการข้อมูลส่วนกลางที่มีประสิทธิภาพการทำงานยังไม่เป็นที่น่าพอใจนัก เนื่องจากการใช้งานต้องมีกระบวนการหลายขั้นตอน และในบางขั้นตอนก็เป็นทางตันของระบบ ซึ่งแก้ไขก็สามารถทำได้ แต่ก็เกิดความซ้ำซ้อนขึ้นดังที่ได้กล่าวมาแล้ว โดยมากปัญหาการทำงานเกี่ยวกับ IIS จะอยู่ในส่วนของ SQL Statement ที่ในบางคำสั่ง SQL นั้นไม่สามารถทำงานบางสิ่งบางอย่างได้ ทั้งที่คำสั่งเหล่านี้เมื่อนำไปทำงานในระบบอื่น ๆ สามารถทำงานได้ และในส่วนที่ต้องนำผลการร้องขอข้อมูลออกแสดงผล ก็มีปัญหบ้างแต่ก็เป็นเพียงปัญหาทั่วไปเท่านั้น หากดูในภาพรวมแล้วการนำ IIS มาใช้งานกับระบบทะเบียนก็ยังคงสามารถใช้งานได้โดยไม่มีปัญหา

สำหรับทางด้านเว็บเบราว์เซอร์ที่ต้องนำโค้ด Java Script มาใช้สำหรับเว็บเพจแต่ละหน้า การเปิดเผยโค้ดของเบราว์เซอร์ก็ไม่มีผลกระทบต่อระบบความปลอดภัย เนื่องจากโค้ดที่สร้างขึ้น มีหน้าที่เพียงจัดการเรื่องการติดต่อกับผู้ใช้งาน ให้มีความสะดวกและเข้าใจการทำงานของระบบได้ง่ายขึ้น ซึ่งส่วนนี้เองจะเป็นส่วนช่วยลดงานของ Server ที่ต้องคอยมาดักจับข้อมูลผิดปกติ ที่ได้รับมา

ในส่วนของฐานข้อมูลนั้นตารางต่างๆถูกออกแบบมาให้สามารถใช้งานได้ก่อนข้างครอบคลุมในระดับหนึ่ง ยังมีส่วนน้อยที่ยังไม่ค่อยเหมาะสมนักเช่น ตารางที่เกี่ยวกับการชำระค่าธรรมเนียมนั้น จะการกำหนดให้มีชนิดของค่าธรรมเนียมทั้งหมด 12 ชนิดซึ่งไม่ค่อยถูกต้องนักเพราะถ้ามีการแก้ไขหรือเพิ่มเติมจะค่อนข้างทำได้ลำบาก และการใช้งาน Microsoft SQL Server นั้นยังไม่เต็มความสามารถอย่างเช่น ไม่มีการกำหนด Rules, Constraint, Trigger ต่างๆ ให้กับตาราง เป็นต้น ทั้งที่สมควรจะต้องนำมาใช้ในด้านการบริหารฐานข้อมูล โครงการนี้ใช้ประโยชน์จากสตอร์โพซิเจอร์ได้มาก ในสตอร์โพซิเจอร์นี้จะใช้การเขียนโปรแกรมแบบ Flow Control ประยุกต์ใช้งานร่วมกับภาษา SQL ซึ่งทำให้การทำงานมีประสิทธิภาพอย่างดีเยี่ยม

บรรณานุกรม

1. Microsoft Corporation, “ Product Document of Microsoft Internet Information Server 3.0 “, Microsoft Corporation, 1997
2. Stephen Feather, “ JavaScript By Example”, Que Corporation, 1996
3. จิตเกษม พัฒนาศิริ, “ สร้าง Homepage ด้วย HTML “, บริษัท ธนาเพรส แอนด์ กราฟฟิค จำกัด, พ.ศ. 2539
4. ดร.ศุภมิตร จิตตะยโสธร, “ Relational Database Design “ , สถาบันไทยสารสนเทศเทคโนโลยี (ITIT), 1994
5. Goren, “ Visual Basic 4 Enterprise “, Que Corporation, 1996
6. Stephen Wynkoop, “ Using Microsoft SQL Server 6.5 Second Edition “, Que Corporation, 1997
7. Orryn Sledge, Mark Spenik, “ Microsoft SQL Server 6.5 DBA Survival Guide “, Sams Publishing, 1996



กิตติกรรมบัตร



โครงการระบบสารสนเทศของสำนักทะเบียนและประมวลผล ส่วนงานลงทะเบียนและค่าธรรมเนียม ผ่านระบบเครือข่ายอินเทอร์เน็ต นี้จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดีถ้าปราศจากบุคคลดังต่อไปนี้

1. สิ่งศักดิ์สิทธิ์ทั้งหลายที่ลบล้างกาลให้โครงการสำเร็จลงได้ โดยปราศจากการรบกวนจากมารผจญต่างๆ
2. บิดา มารดา ผู้ให้กำเนิด เลี้ยงดูให้ความรัก ความเป็นห่วงเป็นใย พร้อมทั้งให้ความสนับสนุนทางการเงินอย่างไม่ขาดแคลน และ บรรดา ญาติ พี่น้อง คอยถามทุกข์สุข ขามที่เราเหน็ดเหนื่อย อ่อนล้า
3. อ.วิบูลย์ พร้อมพานิชย์ ที่ประสิทธิ์ประสาทวิชาและให้คำปรึกษารวมถึงความสะดวกด้านสถานที่รวมทั้งอุปกรณ์ในการดำเนินงาน
4. พี่อ้วน, พี่อ้อด TC#03 และ Computer Engineering (ต่อเนื่อง) #03 ที่จุดประกายความคิดและแนวทางในการพัฒนาโครงการนี้
5. ทีมงานพัฒนาระบบ (เอ, โอ, จอห์น) ที่ได้ร่วมแรงร่วมใจปฏิบัติงานจนสำเร็จลุล่วง
6. พี่หนึ่ง, พี่ชัย, พี่หวาน ที่ให้ความเอื้อเฟื้อในเรื่องสถานที่ซุกหัวนอน ให้ยืมหนังสือ ให้คำปรึกษา และมุขตลกต่างๆ ขามเครียด
7. สมาชิกห้องมัลติมีเดียที่ให้ยืมคอมพิวเตอร์ใช้เพื่อพัฒนาทักษะในยามที่ขาดแคลน
8. สำนักงานเลขาธิการสภาผู้แทนราษฎร กองกรรมาธิการ ๒ อาคารรัฐสภา ๒ ที่เอื้อเฟื้อเครื่องคอมพิวเตอร์และเลเซอร์พรีนเตอร์ในการทำปฏิญญาพันธนี้
9. ศิลปินเพลงจากค่ายต่างๆที่ช่วยขับกล่อมบรรเลงเพลงในขณะทำงานหลังเวลาราชการ
10. มิสเตอร์บิลล์เกตต์ แห่ง Microsoft เจ้าพ่อวงการซอฟต์แวร์ที่สร้างซอฟต์แวร์ดีๆ มาให้ใช้
11. บุคคลที่มีส่วนช่วยให้โครงการสำเร็จขึ้นมาได้ทั้งที่นึกได้ก็ดีและนึกไม่ได้ก็ตาม
12. สุดท้ายขอกราบขอบพระคุณอาจารย์ทุกท่านที่ประสิทธิ์ประสาทวิชาความรู้ให้แก่ศิษย์ตั้งแต่ครั้งยังเยาว์จวบจนถึงปัจจุบัน

วิทยา แก้วรัศมีกุล

อนุชิต จันสทิรพานิช

อนุสรณ์ เหลืองรุ่งทรัพย์

23 มีนาคม 2541