

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนาไดเรกทอรีเซอร์วิสสำหรับการสนทนาผ่านเครือข่ายอินเทอร์เน็ต โดยใช้ LDAP

Development of Directory Service for Internet Chat using LDAP



รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2546
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นักศึกษา	นางสาวพนิตนาฏ รักษาทิพย์
อาจารย์ที่ปรึกษา	ผศ.ดร. จันทรบุรณ์ สถิตวิริยวงศ์
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

โครงการพัฒนาไดเรกทอรีเซอวิซสำหรับ Internet Chat โดยใช้ LDAP เป็นการพัฒนา ระบบสนทนาผ่านเครือข่ายอินเทอร์เน็ต (Chat) โดยโครงการนี้จะจัดเก็บข้อมูลผู้ใช้งาน ข้อมูลของ LDAP Server ซึ่งจะทำให้ผู้บริหารระบบสามารถที่จะทำการบริหารข้อมูลของผู้ใช้งาน เช่น การแก้ไข ลบ เพิ่มหรือค้นหาข้อมูลของผู้ใช้ได้อย่างง่ายดายและมีประสิทธิภาพมากขึ้น และระบบสามารถที่จะตรวจสอบสิทธิ์การเข้าใช้งานแอปพลิเคชันด้วยการพิสูจน์ตัวตนจริง (Authentication) ได้ทำให้มีความปลอดภัยแก่ทั้งผู้ใช้และแอปพลิเคชันเอง อีกทั้งแอปพลิเคชัน Internet Chat ได้ถูกพัฒนาขึ้นให้สามารถใช้ได้กับทุกระบบปฏิบัติการเนื่องจากแอปพลิเคชันได้ถูก ออกแบบให้ผู้ใช้งานติดต่อผ่าน Web-Based ซึ่งเมื่อผู้ใช้ติดต่ออินเทอร์เน็ตก็สามารถใช้งาน แอปพลิเคชันได้ผ่านทาง Browser

Title	Development of Directory Service for Internet Chat using LDAP
Student	Miss Panitnard Rucksatip
Advisor	Asst.Prof.Dr. Chanboon Sathitwiriyawong
Level of Study	Master of Science in information Technology
Major	Information Science
Academic Year	2003

ABSTRACT

This project is focused on developing Directory Service for Internet Chat by using Lightweight Directory Access Protocol (LDAP). The databases of users are kept in LDAP Server so that system administrators are able to manage the system such as editing, deleting, adding or searching user information easily and efficiently. The system also includes the authentication process to verify the user resulting in a high security to both users and applications in the system. The Internet Chat is developed to be cross-platform since it is designed to be a web-based application. Therefore, the user can use the chat and other applications through his/her web browser.

กิตติกรรมประกาศ

งานโครงการพัฒนาโปรแกรมสนทนาผ่านเครือข่ายอินเทอร์เน็ตโดยใช้ LDAP ที่ได้จัดทำขึ้นนี้ข้าพเจ้าได้รับการสนับสนุนจากบุคคลหลายท่าน จึงส่งผลให้โครงการนี้สามารถเสร็จสิ้น ลุล่วงตามเป้าหมายที่วางไว้ ข้าพเจ้าจึงขอขอบคุณบุคคลดังต่อไปนี้

1. บิดามารดาและครอบครัวของข้าพเจ้า ที่ดูแลเอาใจใส่ และเป็นกำลังใจให้ข้าพเจ้าเสมอมา
2. ขอบขอบคุณเพื่อน ๆ ที่คอยให้คำปรึกษาและคอยให้กำลังใจทุกคน
3. ดร. จันทรบุรณ์ สติตวิริยวงศ์ ที่ให้คำปรึกษา ชี้แนะแนวทางในการพัฒนาระบบงานให้เป็นอย่างดี
4. คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบังและอาจารย์ทุก ๆ ท่าน ที่ได้มอบความรู้ทางวิชาการต่าง ๆ เพื่อนำมาใช้ในการพัฒนา

พนิตนาฏ รักษาทิพย์
ผู้จัดทำ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VII
บทที่	
1. บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการพัฒนาระบบงาน	1
1.3 ขอบเขตของการศึกษาและพัฒนาระบบ	2
1.4 ขั้นตอนการศึกษาและพัฒนาระบบ	2
1.5 รายละเอียดของเครื่องคอมพิวเตอร์และเครื่องมือที่ใช้ในการพัฒนาระบบ	3
1.6 รายละเอียดของแต่ละบท	4
2. Directory Service	5
2.1 ระบบ Directory Service	5
2.2 โพรโตคอล LDAP และการทำงานที่เกี่ยวข้องกับ Directory	6
2.2.1 ลักษณะการทำงานของ LDAP	7
2.2.2 The LDAP Models	8
2.3 รูปแบบ Directory Schema	12
2.4 Authentication	13
3. วิเคราะห์และออกแบบระบบ	14
3.1 ความต้องการของระบบ	14
3.2 การออกแบบระบบการทำงาน	14
3.3 กระบวนการทำงานของระบบ	19
3.4 โครงสร้างต้นไม้ (Tree) ในระบบ	20
3.5 โครงสร้างของ Schema	21
3.6 รูปแบบเพิ่มข้อมูลของโปรแกรม	25

สารบัญ (ต่อ)

	หน้า
3.7 ส่วนติดต่อผู้ใช้งาน (User Interface).....	25
4. การพัฒนาระบบงาน.....	30
4.1 ขั้นตอนการพัฒนา ระบบ Directory.....	30
4.2 การพัฒนาโปรแกรม Internet Chat.....	30
4.3 ทดสอบการใช้งานโปรแกรม.....	31
5. สรุปผลการค้นคว้าและพัฒนาระบบ.....	42
5.1 ผลจากการพัฒนาระบบ.....	42
5.1.1 การศึกษาการทำงานของ Directory Service.....	42
5.1.2 การวิเคราะห์และออกแบบ.....	42
5.1.3 การพัฒนาระบบงาน.....	42
5.2 ผลจากการทดสอบโปรแกรม.....	43
5.3 อุปสรรคในการพัฒนาโปรแกรม.....	43
5.4 ข้อเสนอแนะ.....	44
บรรณานุกรม.....	45
ภาคผนวก.....	47
ประวัติผู้เขียน.....	57

สารบัญตาราง

ตารางที่

หน้า

2.1 แสดงถึงลักษณะของ Entry 8



สารบัญญภาพ

ภาพที่	หน้า
2.1	รูปแบบการแลกเปลี่ยนข้อมูลของโปรโตคอล LDAP..... 7
2.2	ส่วนหนึ่งของรูปแบบของ Directory 8
2.3	ส่วนหนึ่งของรูปแบบระบบไฟล์ในระบบ UNIX 10
2.4	ส่วนหนึ่งของรูปแบบ LDAP Directory 10
3.1	ภาพแสดง Context Diagram ของระบบ Directory Service for Internet Chat using LDAP 15
3.2	ภาพแสดง Dataflow Diagram Level 0 ของระบบ Directory Service for Internet Chat using LDAP 15
3.3	ภาพแสดง Dataflow Diagram Level 1 ของระบบ Directory Service for Internet Chat using LDAP 16
3.4	ภาพแสดง Dataflow Diagram Level 2 ของระบบ Directory Service for Internet Chat using LDAP 17
3.5	ภาพแสดง Flow Chart การทำงานของระบบ Directory Service for Internet Chat using LDAP 19
3.6	ภาพแสดงโครงสร้างของต้นไม้ในระบบ Internet Chat 20
3.7	แสดงหน้าจอการสมัครเข้าใช้งานในระบบ 26
3.8	แสดงหน้าจอตรวจสอบสิทธิ์การใช้งาน 26
3.9	แสดงหน้าจอที่ผู้ใช้ผ่านการ Authentication แล้วจะแสดงหน้าจอนี้ 27
3.10	แสดงหน้าจอ Add Contact 28
3.11	แสดงหน้าจอ Delete Contact 28
3.12	แสดงหน้าจอ Chat 29
4.1	แสดงหน้าจอของโปรแกรม Emacs 31
4.2	แสดงข้อความเตือนเมื่อผู้ใช้กรอกข้อมูลไม่ครบ 31
4.3	แสดงข้อความเตือนเมื่อผู้ใช้กรอกข้อมูล E-mail ผิดรูปแบบ 32
4.4	แสดงข้อความเตือนเมื่อผู้ใช้ได้ Password และ Confirm Password ไม่ตรงกัน 32
4.5	แสดงข้อความเมื่อผู้ใช้สามารถลงทะเบียนผ่าน 32
4.6	แสดงหน้า Index ของระบบ 33

สารบัญภาพ (ต่อ)

รูปที่		หน้า
4.7	แสดงข้อความเตือนเมื่อ Authentication ไม่ผ่าน.....	34
4.8	แสดงให้เห็นว่าผู้ใช้มีการ Sign in อยู่แล้ว.....	35
4.9	แสดงให้เห็นว่าผู้ใช้ถูก Add โดยผู้ใช้อีกคนหนึ่ง.....	36
4.10	แสดงหน้าจอหลักเมื่อผู้ใช้ Log in ผ่าน.....	37
4.11	แสดงข้อความเมื่อมีการ Add คนที่ไม่มีในระบบ.....	38
4.12	แสดงข้อความเมื่อมีการ Delete ชื่อคนที่ไม่ได้อยู่ใน List.....	38
4.13	แสดงหน้าจอการสนทนาระหว่างผู้ใช้.....	39
4.14	แสดงหน้าจอเมื่อผู้ใช้ Sign Out.....	40
จ-1	แสดงรูป LDAP Browser.....	55



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การติดต่อสื่อสารโดยการสนทนาผ่านเครือข่ายเป็นการสื่อสารกันในรูปแบบของข้อความ และเป็นที่ยอมรับกันเนื่องจากในเรื่องของค่าใช้จ่ายที่ต่ำและไม่ว่าผู้ใช้จะอยู่ที่ไหนก็สามารถจะพูดคุยติดต่อสื่อสารในรูปแบบข้อความนี้ได้ และเพื่อให้ผู้บริหารระบบสามารถที่จะควบคุมดูแลระบบบัญชีรายชื่อผู้ใช้งานได้จาก Directory Server ซึ่งเป็นศูนย์กลางเพียงที่เดียวทำให้ง่ายต่อการรักษาความปลอดภัยของแอปพลิเคชัน นอกจากนี้แอปพลิเคชัน Internet Chat ยังอาศัยความสามารถในการพิสูจน์ตัวตนจริง (Authentication) ของเทคโนโลยี Directory Service ซึ่งจะรองรับการเพิ่มขึ้นของแอปพลิเคชันและผู้ใช้ในอนาคตเป็นอย่างดี และผู้ออกแบบได้พัฒนาแอปพลิเคชัน Internet Chat ให้สามารถใช้งานได้ทุกแพลตฟอร์ม โดยอาศัย Browser เป็นส่วนติดต่อผู้ใช้

1.2 วัตถุประสงค์ของการพัฒนาระบบงาน

สำหรับแอปพลิเคชันสนทนาผ่านเครือข่ายอินเทอร์เน็ต (Internet Chat) เป็นแอปพลิเคชันที่สามารถที่จะติดต่อสื่อสารกันได้แบบ real-time มุ่งเน้นที่จะพัฒนาโดยนำข้อดีของ Directory Services มาใช้

- 1.2.1 เพื่อเก็บข้อมูลของผู้ใช้ให้เป็นรูปแบบเดียวกันคือมีโครงสร้างเป็นต้นไม้ (Tree) เพื่อในการค้นหาหรือแก้ไขข้อมูลผู้ใช้จะเป็นไปได้อย่างรวดเร็ว
- 1.2.2 เพื่อความปลอดภัยโดยระบบจะมีการพิสูจน์ตัวตนผู้ใช้ (Authentication) เพื่อให้ผู้ใช้แน่ใจว่าคู่สนทนาคือตัวจริง
- 1.2.3 เพื่อสร้างระบบบริหารบัญชีผู้ใช้งาน และมีการบริหารการใช้งานจากศูนย์กลางเพื่อเพิ่มประสิทธิภาพในการจัดการ และยังสามารถลดความซ้ำซ้อนของการเก็บข้อมูลผู้ใช้งานถ้ามีการเก็บข้อมูลไว้หลายที่
- 1.2.4 เพื่อเพิ่มความน่าเชื่อถือในด้านการทำงานของระบบเช่น การสนทนาข้อความที่ส่งจะมีความน่าเชื่อถือและถูกต้องยิ่งขึ้น และเพิ่มในเรื่องความปลอดภัย
- 1.2.5 รองรับปริมาณการใช้งานและจำนวนแอปพลิเคชันที่จะมีเพิ่มขึ้นในระบบ Directory Service ในอนาคต

1.3 ขอบเขตของการศึกษาและพัฒนาระบบ

โครงการพัฒนาไครกทอรีเซอร์วิสสำหรับการสนทนาผ่านเครือข่ายอินเทอร์เน็ต โดยใช้ LDAP นี้จะทำการศึกษาถึงการทำงานของโปรแกรม Openldap ซึ่งเป็นโปรแกรมที่ใช้เป็น Directory Server โดยที่ไม่เสียค่าลิขสิทธิ์ในการใช้งาน โดยระบบจะทำการเก็บข้อมูลผู้ใช้งาน ลงในฐานข้อมูลของ Openldap และมีการติดต่อกับผู้ใช้งาน ผ่านระบบเครือข่ายอินเทอร์เน็ต โดยมีรูปแบบการติดต่อผ่าน Browser

โปรแกรมที่ถูกพัฒนาขึ้นจะมีความสามารถ ดังต่อไปนี้

- 1.3.1 ระบบงานที่พัฒนาขึ้นสามารถทำการเชื่อมต่อเข้าใช้บริการ Directory Service เพื่อที่จะเก็บข้อมูลที่ต้องใช้ในระบบเครือข่ายได้อย่างถูกต้อง
- 1.3.2 แอปพลิเคชัน Internet Chat สามารถที่จะทำการพิสูจน์ตัวตนจริง (Authentication) ผู้ใช้งานได้ เพื่อความปลอดภัยของผู้ใช้
- 1.3.3 ระบบงานมีการติดต่อกับผู้ใช้งานผ่านระบบ Web-Based เพิ่มความสะดวกในการใช้งานในทุกแพลตฟอร์ม
- 1.3.4 แอปพลิเคชัน Internet Chat จะสามารถแสดงสถานะ Online หรือ Offline ของผู้ใช้ได้
- 1.3.5 แอปพลิเคชัน Internet Chat สามารถที่จะสืบค้นข้อมูลเพื่อการเพิ่มรายชื่อเพื่อน การลบรายชื่อเพื่อนของผู้ใช้งาน และสามารถส่งข้อความได้อย่างถูกต้อง
- 1.3.6 สามารถที่จะส่งข้อความสนทนาถึงกันได้อย่างถูกต้องและรวดเร็ว เนื่องจากระบบพัฒนาให้ส่งข้อความถึงกันแบบ real-time

1.4 ขั้นตอนการศึกษาและพัฒนาระบบ

การพัฒนาแอปพลิเคชันสนทนาผ่านเครือข่าย (Internet Chat) โดยใช้ LDAP มีขั้นตอนการทำงานดังนี้

- ขั้นตอนที่ 1 ศึกษาการทำงานของระบบ Directory Service ตลอดจนทฤษฎีต่างๆที่มีส่วนเกี่ยวข้องเพื่อนำมาใช้ในการออกแบบระบบ
- ขั้นตอนที่ 2 ศึกษาแอปพลิเคชันสนทนาผ่านเครือข่ายอินเทอร์เน็ต (Internet Chat) ที่มีอยู่ปัจจุบัน จากเอกสารและการใช้งาน โดยดูถึงรูปแบบโครงสร้างข้อมูลที่ใช้ในระบบ และความเป็นไปได้ในการพัฒนาระบบกำหนดขอบเขตในการพัฒนาระบบ

- ขั้นตอนที่ 3 ความต้องการของผู้ใช้งานในระบบ โดยค้นหาปัญหาที่เกิดจากการใช้งาน โดยข้อมูลที่ได้จากการศึกษาแอปพลิเคชันสนทนาผ่านเครือข่าย อินเทอร์เน็ต (Internet Chat) ทั่วไป
- ขั้นตอนที่ 4 วิเคราะห์และออกแบบระบบงาน รวมถึงออกแบบการทำงานของโปรแกรม ภายใต้ขอบเขตการศึกษาและพัฒนาระบบงานตามที่ได้กำหนดไว้
- ขั้นตอนที่ 5 ทำการพัฒนาโปรแกรม ทดสอบ การทำงานของโปรแกรมให้สามารถทำงานตามที่ได้ออกแบบไว้ได้
- ขั้นตอนที่ 6 ทำการทดสอบความผิดพลาดและปรับแต่งการทำงานของโปรแกรมเพื่อให้สามารถนำไปใช้งานได้โดยมีประสิทธิภาพต่อไป

1.5 รายละเอียดของเครื่องคอมพิวเตอร์และเครื่องมือที่ใช้ในการพัฒนาระบบ

เนื่องจากโปรแกรมที่ทำกำรพัฒนามีหลักการทํางานพื้นฐานของ Client/Server ดังนั้นเพื่อให้มีสภาวะการทํางานที่เสมือนมีการใช้งานจริงมากที่สุด จึงได้มีการติดตั้ง Unix Server ขึ้นโดยได้นำเอา Red Hat 8.0 มาใช้เป็นระบบปฏิบัติการ ทำหน้าที่เป็นผู้ให้บริการ Web และในขณะเดียวกันก็ทำหน้าที่เป็น LDAP Server ด้วย ส่วนทางด้าน Client นั้นได้มีการจัดหาเครื่องคอมพิวเตอร์อีก 1 เครื่องที่มีการทำงานบนระบบปฏิบัติการ Microsoft Windows ใช้ในการพัฒนา แต่เพื่อการสนทนาคุยกัน ได้จึงให้เครื่องที่ทำหน้าที่เป็น Client ด้วย

1.5.1 เครื่อง Web Server และ LDAP Server

CPU	Intel Pentium IV 1400 MHz
Memory	256 MB
Hard-Disk	20 GB
Network-Card	D-Link
Graphic-Card	nVidia TNT 2
OS	Linux Redhat 8.0
Application	Openldap 2.1.17
	Apache 1.3.27
	PHP 4.3.2

1.5.2 เครื่องที่ทำหน้าที่เป็น Client

CPU	Intel Pentium III 350 MHz
Memory	192 MB
Hard-Disk	40 GB
Network-Card	D-link
Graphic-Card	3Dfx Voodoo Banshee 16 MB
OS	Linux Redhat 8.0

1.5.3 ระบบเครือข่าย

LAN

1.5.4 เครื่องมือที่ใช้ช่วยในการพัฒนา

1.5.4.1 GNU Gcc 3.2.1

1.5.4.2 LDAP Browser 282b2

1.6 รายละเอียดของแต่ละบท

สำหรับเนื้อหาของเอกสารประกอบการพัฒนาโปรแกรมฉบับนี้ ได้มีการแบ่งเนื้อหาออกเป็นบทต่าง ๆ ดังต่อไปนี้

บทที่ 2 จะกล่าวถึง Directory Service และทฤษฎีที่เกี่ยวข้อง คำนิยามและความหมายต่าง ๆ ที่ควรทราบ รวมถึงวิธีการทำงานอย่างคร่าว ๆ ของ LDAP โครงสร้างการเก็บข้อมูล รวมถึงบริการที่ LDAP มีให้

บทที่ 3 จะเป็นการวิเคราะห์และออกแบบระบบ ซึ่งจะกล่าวถึงความต้องการปัจจุบัน รวมถึงการออกแบบโครงสร้างการทำงาน และรูปแบบการติดต่อกับผู้ใช้งาน

บทที่ 4 จะเป็นขั้นตอนการพัฒนาโปรแกรม ตามที่ได้ทำการออกแบบไว้ในบทที่ 3 รวมถึงการตรวจสอบความถูกต้องของโปรแกรมที่พัฒนาขึ้น

บทที่ 5 เป็นการสรุปผลการพัฒนา ประโยชน์ที่ได้รับจากการพัฒนาโปรแกรม ตลอดจนอุปสรรคต่าง ๆ และข้อเสนอแนะเพื่อเป็นประโยชน์แก่ผู้ที่จะนำไปใช้งาน หรือ พัฒนาโปรแกรมต่อไป

บทที่ 2

Directory Service

2.1 ระบบ Directory Service

Directory คือ ชนิดของฐานข้อมูล พิเศษ ถูกออกแบบเพื่อที่จะรวบรวมข้อมูลเกี่ยวกับคน ทรัพยากรต่างๆ และนโยบายซึ่งเป็นประโยชน์กับแอปพลิเคชันในระบบเครือข่ายในเรื่องของการบริการและอุปกรณ์ต่างๆ มาตรฐานสากลสำหรับ Directory คือ LDAP และ X.500 ซึ่งเป็นโปรโตคอลที่ Client ทั่วไปใช้ในการเข้าถึง Directory ถูกใช้อย่างรวดเร็วและเป็นจำนวนมาก Directory มีความจำเป็นอย่างมากสำหรับระบบเครือข่ายที่มีการขยายตัวและมีบทบาทอย่างต่อเนื่อง จึงต้องการในเรื่องของการแก้ปัญหา วิธีการจัดระบบเครือข่ายให้เหมาะสม

Directory Service จะเก็บข้อมูลต่างๆ ใน Directory Information Base (DIB) โดย DIB จะถูกจัดเก็บในรูปแบบของ Tree เรียกว่า Directory Information Tree (DIT)

ข้อมูลใน Directory จะถูกเก็บใน Directory Server เพียงที่เดียว โดยจะรวบรวมส่วนของ แอปพลิเคชัน ส่วนของการบริการ หรือส่วนของอุปกรณ์ไว้ ซึ่ง Directory Server ก็ยังสามารถทำงานร่วมกับ Directory Server อื่นๆ โดยการใช้โปรโตคอลชนิดเดียวกันคือ X.500 เพื่อทำหน้าที่ในการกระจาย Directory Service

ข้อแตกต่างระหว่าง Directory และ Relational Database มีดังนี้

Directory เปรียบเหมือนกับเป็น ฐานข้อมูลแบบพิเศษ เมื่อทดลองเปรียบเทียบคุณสมบัติบางอย่างกับฐานข้อมูลแล้วพบว่า

- Directory มีอัตราการอ่านข้อมูลสูงกว่าการเขียนข้อมูล (read-to-write ratio) เมื่อเทียบกับฐานข้อมูล
- การปรับเปลี่ยน หรือเพิ่มเติมโครงสร้างของ Directory ง่ายกว่าของฐานข้อมูล (extensibility)
- ในระบบเครือข่ายที่มีขนาดใหญ่ การเพิ่ม Directory Service โดยใช้วิธีการคัดลอกโครงสร้างข้อมูลเดิม (replication scale) เข้ามาช่วยจะสามารถเพิ่มประสิทธิภาพการทำงานของระบบได้ดี และง่ายกว่าฐานข้อมูล
- Directory มีการทำงานแบบเฉพาะทาง ดังนั้นจึงมีประสิทธิภาพในการทำงาน (performance) สูงกว่าการทำงานของฐานข้อมูล
- Directory มีความเป็นมาตรฐานในการพัฒนามากกว่า ฐานข้อมูล

ในปี 1980 CCITT (the International Telegraph and Telephone Consultative Committee) ภายหลังเปลี่ยนเป็น ITU (the International Telecommunications Union) ได้พัฒนา White Pages Directory Service ที่สามารถเก็บ และเรียกดูข้อมูลหมายเลขโทรศัพท์ ที่อยู่จดหมายอิเล็กทรอนิกส์ (e-mail) ได้ง่าย

ในขณะเดียวกัน ISO (the International Organization for Standardization) ก็ได้พัฒนา Directory Service ที่ทำหน้าที่เป็น Name Service สำหรับใช้ในเครือข่าย OSI (Open Systems Interconnect) ด้วยเช่นกัน (คล้ายกับ DNS ในปัจจุบัน)

ภายหลัง ในปี 1988 ทั้ง 2 มาตรฐานถูกรวมเข้าด้วยกัน โดยใช้ชื่อว่า X.500 และถูกเริ่มใช้งานอย่างจริงจังตั้งแต่ในปี 1990 เป็นต้นมา โดย CCITT

2.2 โพรโตคอล LDAP และการทำงานที่เกี่ยวข้องกับ Directory

LDAP ได้ถูกพัฒนาขึ้นโดยเลือกเอาข้อดีของ X.500 มาใช้ พร้อมกับตัดการทำงานที่ซับซ้อนที่ไม่จำเป็นออกดังนี้

- ตัดฟังก์ชันการทำงานของโปรโตคอล DAP บางอย่างออก เหลือไว้เฉพาะส่วนที่ควบคุมการทำงานระหว่าง LDAP Clients และ Sever ซึ่งถือเป็นจุดเด่นของ LDAP เนื่องจาก Client สามารถส่ง Request Search Operation และรับค่า Result Set ได้หลายค่าภายในเวลาเดียวกัน
- ข้อมูลส่วนใหญ่จะถูกแสดงในรูปของข้อความ (text) แม้ว่าจะถูกเก็บอยู่ในรูปของ Binary Encoding Rules ของ X.500 ก็ตาม ทั้งนี้เพื่อให้ง่ายต่อการพัฒนาและได้ประสิทธิภาพที่ดี
- ตัดโปรโตคอลอื่นๆที่ทำงานบน OSI network stack multilayer ออกเหลือไว้แต่ โปรโตคอล TCP/IP เท่านั้น เพื่อง่ายต่อการพัฒนา และเป็นการช่วยเพิ่มประสิทธิภาพการทำงานอีกด้วย

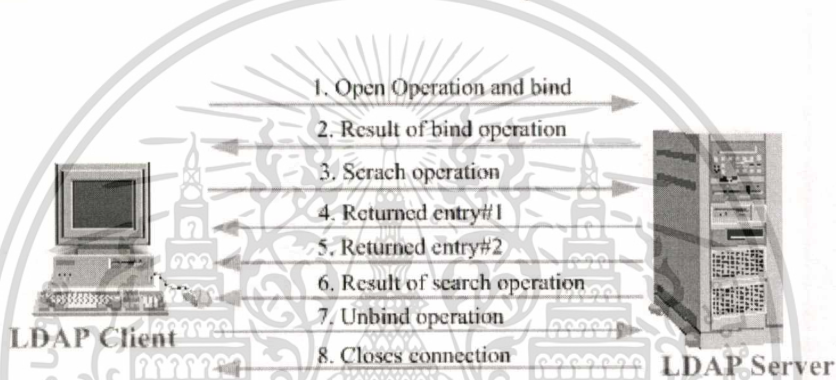
ได้มีการพัฒนาปรับปรุงแก้ไข 2 ส่วนหลัก ของ โปรโตคอล LDAP การพัฒนาแรกเป็น เวอร์ชัน ที่มีอยู่อย่างแพร่หลาย คือ LDAP เวอร์ชัน 2 ถูกกำหนดใน RFCs 1777 และ 1778 ส่วน LDAP เวอร์ชัน 3 ถูกเขียนขึ้นมาเพื่อเป็นมาตรฐานของ อินเทอร์เน็ต ถูกกำหนดใน RFCs 2251 ถึง 2256 เนื่องจากว่าเป็นเรื่องที่ใหม่มาก ดังนั้นจึงยังไม่มีผู้ขาย (vendor) รายไหนที่จะสนับสนุน LDAPv3 ได้อย่างสมบูรณ์

บทบาทของ LDAP ในการเป็นโปรโตคอลในระบบเครือข่าย มาตรฐาน LDAP ได้กำหนดให้มี 4 Model ที่จะใช้ใน Directory Model เหล่านี้สนับสนุนให้มีความสามารถที่จะทำงานร่วมกัน ระหว่าง Directory ได้ในตัว Model ได้นำแนวคิดมาจาก X.500 แต่ก็ยังขาดข้อจำกัดหลายๆข้อที่เป็นส่วนประกอบของ Model X.500

LDAP Models ประกอบด้วย 4 models ดังนี้

1. The LDAP Information Model ซึ่งจะเป็นตัวกำหนดชนิดของข้อมูล ที่สามารถจะใส่ลงไปใน Directory
2. The LDAP Naming Model ซึ่งจะกำหนดเป็นระบบว่าจะมีการอ้างถึงข้อมูลใน Directory อย่างไร
3. The LDAP Functional Model ซึ่งจะกำหนดว่าจะเข้าถึงและ update ข้อมูลใน Directory อย่างไร
4. The LDAP Security Model ซึ่งจะกำหนดว่าทำอย่างไรจึงจะสามารถป้องกันข้อมูลจากการเข้าถึงแบบ Unauthorized ได้

2.2.1 ลักษณะการทำงานของ LDAP สามารถอธิบายได้จากรูปต่อไปนี้



รูปที่ 2.1 รูปแบบการแลกเปลี่ยนข้อมูลของโปรโตคอล LDAP

เริ่มต้นจาก Client ขอติดต่อกับ LDAP Server โดยใช้ operation bind ผ่านทาง TCP connection โดยใน bind operation นี้ Client จะส่งชื่อ Directory พร้อมกับ Password เพื่อตรวจสอบสิทธิ์ ก่อนเริ่มเข้าใช้งาน Directory Service

ขั้นที่ 2 หลังจากที่ Directory ได้ตรวจสอบสิทธิ์ผู้ใช้งานแล้ว ก็จะส่งค่า success กลับไปยัง Client

ขั้นที่ 3 ผู้ใช้ส่งข้อมูลที่ต้องการค้นหาไปยัง Server

ขั้นที่ 4, 5 Server ค้นหาข้อมูลตามที่ผู้ใช้ขอมาพร้อมกับตอบกลับไปยัง Client ว่าค้นพบข้อมูล 2 หัวข้อ

ขั้นที่ 6 Server ส่งผลลัพธ์ที่ค้นหาได้ ให้กับ Client

ขั้นที่ 7 Client ส่ง unbind request ไปยัง Server เพื่อขอหยุดการเข้าใช้งาน Directory Service

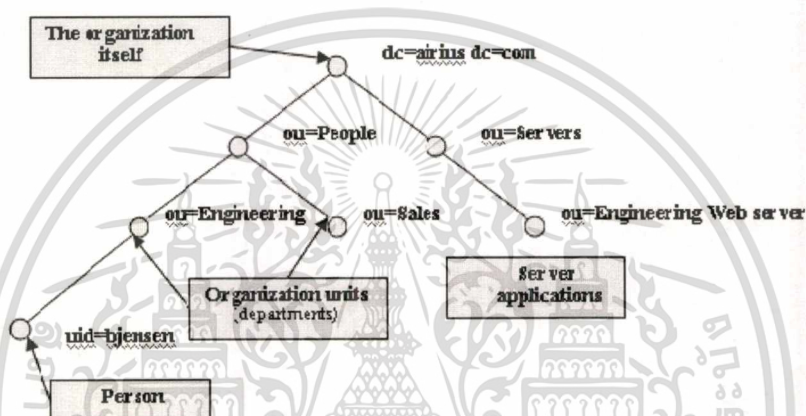
ขั้นที่ 8 Server ยกเลิกการติดต่อกับ Client

2.2.2 The LDAP Models

- **The LDAP Information Model**

กำหนดชนิดของข้อมูลที่จะเก็บใน Directory จะใช้คำว่า Entry แทนกลุ่มของข้อมูล (Information Collection) ที่อธิบายถึง Object ใน Real-World ซึ่งมักจะเปรียบเทียบกับวัตถุสิ่งของในชีวิตประจำวัน เช่น บุคคล แผนก เซิร์ฟเวอร์ (Server) เครื่องพิมพ์ เป็นต้น

ผังรูปที่ 2.2 นี้ได้แสดงตัวอย่างส่วนหนึ่งของ Directory ซึ่งแสดงถึง Object ภายในองค์กร



รูปที่ 2.2 ส่วนหนึ่งของรูปแบบของ Directory

ใน Entry จะประกอบไปด้วย Set ของ Attributes ซึ่งจะอธิบายถึงลักษณะเฉพาะของแต่ละ Object แต่ละ Attribute จะประกอบไปด้วย ส่วนระบุชนิดของข้อมูล (Type) และ ส่วนที่เก็บข้อมูล (Value) ที่อาจมี 1 ค่าหรือมากกว่าก็ได้ ดังตารางที่ 2.1

Attribute type	Attribute value
cn:	Barbara Jensen Babs Jensen
sn:	Jensen
telephonenumber:	+1 408 555 1212
mail:	babs@airius.com

ตารางที่ 2.1 แสดงถึงลักษณะของ Entry

โดยส่วนมากแล้ว เรามักจะเลือกแสดงค่า Directory entries ทั้งหมดให้อยู่ในรูปแบบของ LDIF Text Format เพราะเป็นมาตรฐานที่นิยมใช้ทั่วไป ง่ายต่อการนำเข้า (import) และ ส่งออก (export) ข้อมูลจาก Directory Server เนื่องจากข้อมูลอยู่ในรูปแบบของ ACSII Code ดังตัวอย่าง

```
dn: uid=bjensen, dc=airius, dc=com
```

```
objectClass: top
```

```
objectClass: person
```

```
objectClass: organizationalPerson
```

```
objectClass: inetOrgPerson
```

```
cn: Barbara Jensen
```

```
cn: Babs Jensen
```

```
sn: Jensen
```

```
mail: bjensen@airius.com
```

```
telephoneNumber: +1 408 555 1212
```

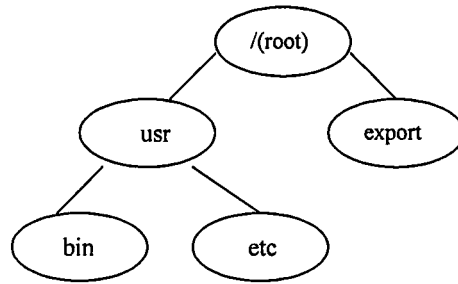
```
description: A big sailing fan.
```

ใน 1 Entry จะเริ่มต้นด้วย attribute ที่ชื่อ Dn (Distinguished Name) เสมอ หลังจากนั้น ก็จะเป็น attribute อื่นๆที่เกี่ยวข้องตามมา โดยจะแสดงในรูปแบบของ

```
<attribute type>:<attribute value>
```

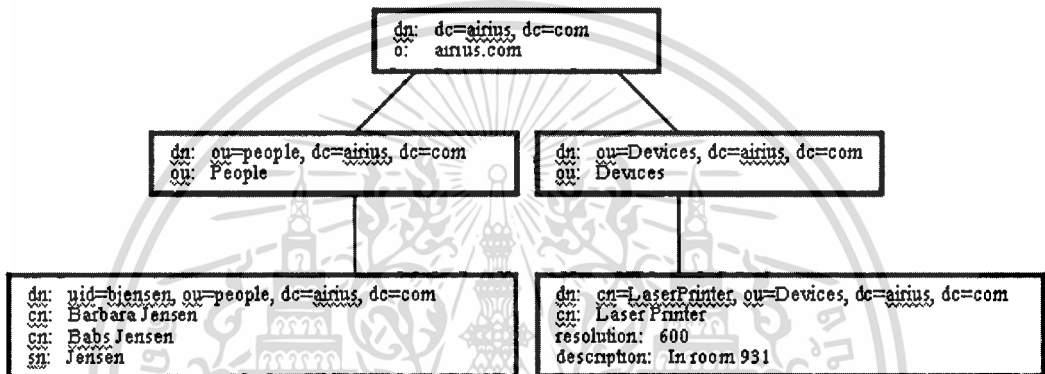
- **The LDAP naming model**

ใน Model นี้ เน้นเรื่องการจัดการ และอ้างอิงถึงข้อมูลภายใน Directory เป็นหลัก เราพบว่าการเรียงลำดับข้อมูลภายใน Entry มีลักษณะเป็นแบบ Inverted Tree ซึ่งคล้ายกับ Unix file System ต่างกันตรงที่การเรียกชื่อ Node ของ Unix file System มีลักษณะเป็นแบบ อ้างอิงจากบนลงล่าง แต่ของ LDAP เป็นแบบอ้างอิงจากล่างขึ้นบน ดังแสดงเปรียบเทียบได้จากรูปที่ 2.3,2.4



รูปที่ 2.3 ส่วนหนึ่งของรูปแบบระบบไฟล์ในระบบ UNIX

Unix file system:/root/user/bin (top - down)



รูปที่ 2.4 ส่วนหนึ่งของรูปแบบ LDAP Directory

LDAP: uid=bjensen ou=people, dc=airius, dc=com (down - top)

โดยปกติแล้ว เราจะอ้างถึงทุกๆ Entry ใน LDAP ผ่านทางส่วน Attribute Dn (Distinguished Name) ซึ่งหมายความว่าส่วนนี้จะต้องมีความเป็น Unique ในระบบ

การพิจารณาความเป็น Unique ของ RDN (Relative Distinguished Name) นั้น ให้พิจารณาที่ Node ระดับเดียวกัน ภายใต้ Parent Node ของตนก็เพียงพอ เนื่องจาก Dn จะแปรผันตามค่า Parent Node ของ Node ปัจจุบัน ดังนั้นในกรณีที่มี Parent Node แตกต่างกัน ถึงแม้ชื่อ Node จะเหมือนกัน ก็ยังคงทำให้ Dn มีความเป็น Unique อยู่เช่นเดิม ดังแสดงในตัวอย่างต่อไปนี้

dn: uid=smith, ou=sales, dc=airius, dc=com

dn: uid=smith, ou=engineering,

dc=airius, dc=com

- **The LDAP functional model**

เราแบ่งการ operation ของ LDAP function ได้เป็น 3 แบบ ดังนี้

แบบ *interrogation operations*

จัดการในเรื่องของการเรียกดูข้อมูลจาก LDAP เราไม่สามารถสั่ง read ข้อมูลโดยตรงได้ วิธีการเดียวที่เราใช้เมื่อต้องการเรียกดูข้อมูลคือคำสั่งค้นหาข้อมูล (LDAP search)

แบบ *update operations*

จัดการในเรื่องการ update ข้อมูลของ LDAP มีทั้งหมด 3 ประเภทด้วยกัน ดังนี้

- Add operation คือการสร้าง Directory
- Delete operation คือการลบ entry ออกจากระบบ
- Rename (modify dn operation) คือการแก้ไขค่า entry

แบบ *authentication and control operations*

การตรวจสอบ ยกเลิกสิทธิการเข้าใช้งาน Directory มี 3 ประเภท

- Bind operation เป็นการตรวจสอบสิทธิการเข้าใช้งานของ client
- Unbind operation เป็นการขอยกเลิกการเข้าใช้งาน Directory
- Abandon operation เป็นการขอยกเลิก operation ที่กำลังทำงานอยู่

- **The LDAP security model**

จากที่กล่าวมาทั้งหมด เราพบว่า LDAP มีลักษณะการทำงานแบบ protocol ทั่วไป ต่างกันตรงที่มีระบบในการตรวจสอบสิทธิผู้ขอใช้งานเพิ่มขึ้นมาด้วย

ก่อนการเข้าใช้งาน Directory Service ทุกครั้งเราจำเป็นต้องตรวจสอบสิทธิก่อน ดังนั้นการป้องกันความปลอดภัยของข้อมูลจึงถูกพิจารณาเฉพาะในส่วนนี้เป็นหลัก โดยมีจุดมุ่งหมายว่า ทำอย่างไรจึงจะทราบได้ว่าผู้ที่ขอเข้าใช้งานใน Directory เป็นผู้มีสิทธิใช้งานได้จริง Access control model เป็นตัวอย่างหนึ่ง ที่ใช้ในการกำหนดสิทธิการ operation ของผู้เข้าใช้งาน เช่น ผู้ใช้งานอาจเข้าใช้งาน Directory Service ได้ภายหลังจากการ bind แล้ว แต่ผู้ใช้งานสามารถปรับเปลี่ยนหรือลด attribute ได้เฉพาะบางตัวตามที่กำหนดไว้เท่านั้น ทั้งนี้ขึ้นอยู่กับสิทธิของผู้ดูแลระบบ

2.3 รูปแบบ Directory Schema

รูปแบบของ Directory Schema นั้นเกิดจากผู้ริเริ่มคิดค้น Directory และผู้ผลิตรายใหญ่ที่มีบทบาทในการกำหนดมาตรฐานของโปรโตคอล บ่อยครั้งที่มาตรฐานต่าง ๆ ในระบบอินเทอร์เน็ต นั้นเกิดจากการริเริ่มของผู้ผลิต Software รายใหญ่ เช่น IBM SUN Microsoft Oracle เป็นต้น รูปแบบของ Directory Schema ที่เป็นที่ยอมรับใช้งานมีอยู่ด้วยกัน 3 แบบ คือ

- slapd.conf
- ASN.1
- LDAPv3

โดยจะกล่าวถึง Directory Schema ที่ระบบ Internet Chat ได้ใช้ในการพัฒนาคือ Directory Schema แบบ slapd.conf ซึ่งเริ่มมีการใช้งานครั้งแรกใน Directory จากมหาวิทยาลัยมิชิแกนใช้ในซอฟต์แวร์ U-M LDAP 3.3 และต่อมาบริษัทเนสเคปอเปอเรชั่น ได้ทำการพัฒนา Directory Server และใช้ Directory Schema ใน Netscape Directory Server 3.x และ 4.x รูปแบบไฟล์ slapd.conf ที่ใช้ใน Directory จะประกอบด้วยกัน 2 ไฟล์คือ slapd.at.conf slapd.oc.conf

- slapd.at.conf (Attribute Type Definition)

เป็นไฟล์ Directory Schema ที่ใช้กำหนดเกี่ยวกับ Attribute มีรูปแบบของไฟล์ดังนี้

Attributetype	OID	NAME	[ALIASES]	SYNTAXID	[OPTION]
NAME		ชื่อของ Attribute ที่ต้องการกำหนดเช่น cn			
ALIASES		ชื่อที่เป็นออปชันให้เราสามารถกำหนดอื่น ๆ ได้ เช่น commonName			
OID		Object Identify Definition ของ Attribute			
SYNTAXID		อักษรย่อที่กำหนดชนิดของ Attribute			
OPTION		ใช้กำหนดรายการของออปชัน มีด้วยกัน 2 แบบคือ Operation และ Single			

Operation Option ใช้สำหรับ Directory Service ใช้ Attribute ในกรณีพิเศษ

- slapd.oc.conf (Object Class Type Definitions)

เป็นไฟล์ Directory Schema ที่ใช้กำหนดเกี่ยวกับ Objectclass มีรูปแบบของไฟล์ดังนี้

**objectclass [oid OID] NAME [superior SUP] [requires REQATTRS]
[allowsALLOWATTRS]**

NAME	ชื่อของ Objectclass
OID	Object Identify Definition ของคลาส (ออปชั่น)
SUP	Superior class หรือ parent class (ออปชั่น)
REQATTRS	ลิสต์ของ Attribute ที่ต้องกำหนดเมื่อใช้ Objectclass (ออปชั่น)
ALLOWATTRS	ลิสต์ของ Attribute ที่อยู่ใน Objectclass (ออปชั่น)

ทั้ง REQATTRS และ ALLOWATTRS เป็น Attribute ที่จะต้องกำหนดไว้ในไฟล์ slapd.at.conf
ข้อควรระวังในการตั้งชื่อ Objectclass ต้องไม่ตั้งชื่อซ้ำกับชื่อที่มีอยู่ในไฟล์ slapd.oc.conf

2.4 Authentication

รูปแบบการ Authentication ใน Directory Server มีรองรับไว้หลายรูปแบบเพื่อรองรับกับความต้องการใช้งานที่แตกต่างกัน การทำงานในแต่ละรูปแบบมีความเหมาะสมในการทำงานไม่เหมือนกัน รูปแบบของการ Authentication มีอยู่ด้วยกัน 5 รูปแบบคือ Anonymous Authentication, Simple Password Authentication, Simple Password Authentication Over SSL, Certificate Authentication Over SSL, Another schema via SASL สำหรับในระบบ Directory Service for Internet Chat using LDAP นี้จะพัฒนาโดยใช้การ Authentication แบบ Simple Password Authentication จะมีการใช้ User และ Password ใน Operation bind สำหรับการ Authentication เมื่อ Directory Client มีการใช้ Operation bind ด้วย User และ Password ก็จะได้รับสิทธิในการทำงาน Directory Server ตามที่กำหนดไว้เช่น read write compare delete modify เป็นต้น สิทธิในการทำงานก็จะมีความสัมพันธ์กับเรื่อง Access Control อีกที่หนึ่ง โดยพื้นฐานของ Simple Password Authentication User จะสามารถแก้ไขเปลี่ยนแปลง Attribute Value ของตัวเองได้ในบาง Attribute ที่ผู้บริหาร Directory เป็นผู้กำหนด

การใช้ Simple Password Authentication มีข้อควรระวังคือ Password ที่ส่งมาจาก Directory Client มายัง Directory Server จะไม่มีการเข้ารหัสข้อมูลในโปรโตคอล TCP/IP แสกเกอร์สามารถจะดักจับข้อมูลในสายได้ง่าย

บทที่ 3

การวิเคราะห์และออกแบบระบบ

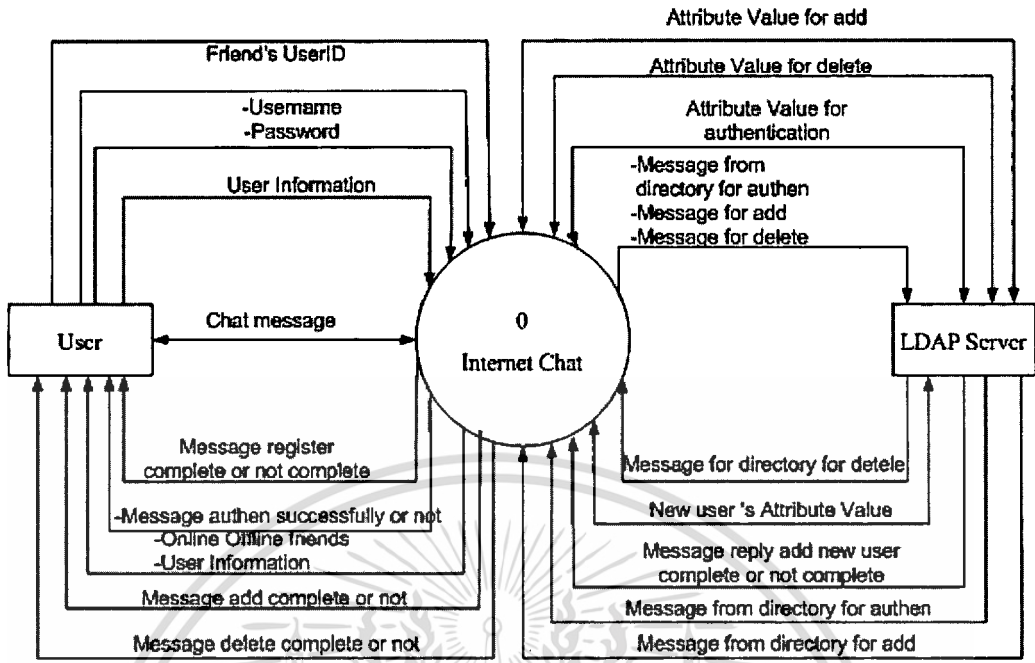
3.1 ความต้องการของระบบ

จากการศึกษาระบบที่ Internet Chat ที่ผู้ออกแบบเคยได้ใช้งานจริง รวมทั้งพิจารณาทฤษฎีที่เกี่ยวข้องในบทที่ผ่านมา จึงสามารถกำหนดความต้องการเพื่อให้ระบบที่พัฒนามีความเหมาะสมในการใช้งาน และสามารถช่วยแก้ปัญหาบางประการจากที่เคยพบได้ ดังต่อไปนี้

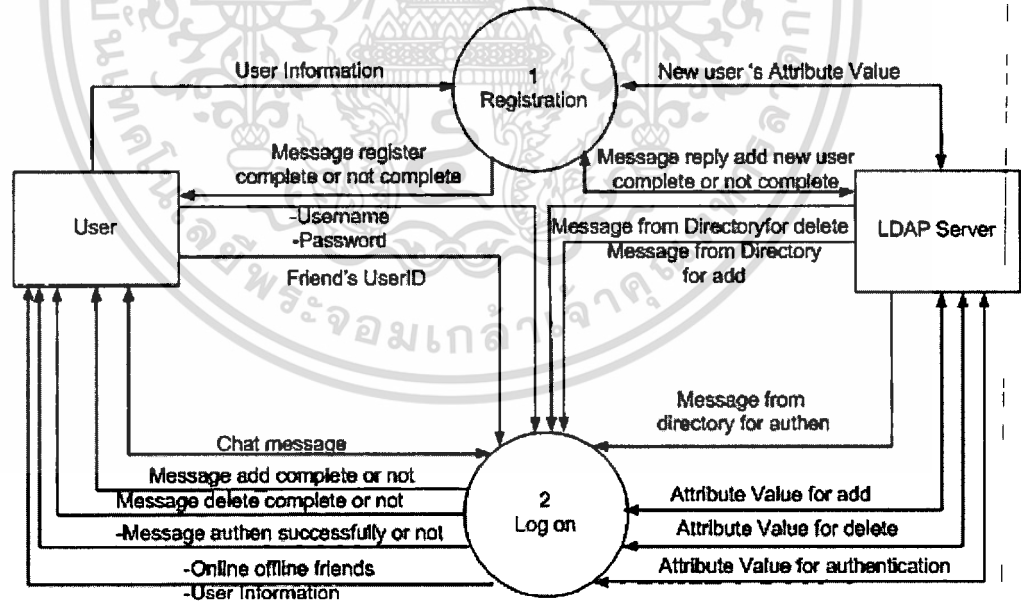
1. สามารถลดความซ้ำซ้อนได้ในการเก็บข้อมูล
2. ลดปัญหาที่เกิดจากการเก็บข้อมูล เช่น list รายชื่อของกลุ่มสนทนา ที่บางครั้งถ้าเก็บที่ Local จะทำให้เกิด List หายถ้าโปรแกรมเป็นแบบที่ต้องติดตั้งใหม่ทุกครั้งเมื่อใช้งานที่เครื่องอื่น ระบบจึงพัฒนาโดยให้มีการเก็บข้อมูลที่ Server และเป็นแบบ Web-based
3. ระบบต้องการให้สามารถคุยกันข้าม Platform ได้ จึงพัฒนาโดยให้อยู่ในรูปแบบของ Web-based และมีรูปแบบการใช้งาน โดยผ่าน Browser ของผู้ใช้
4. ระบบต้องการให้สามารถแสดงสถานะ Online หรือ Offline ของผู้ใช้ได้
5. สามารถสร้างระบบที่น่าเชื่อถือ ทั้งทางด้านการทำงาน และความปลอดภัย ในการเก็บข้อมูลบัญชีรายชื่อของผู้ใช้ รวมทั้งความปลอดภัยในการเข้าใช้ด้วย
6. สามารถสร้างระบบที่ใช้เก็บข้อมูลบัญชีผู้ใช้งานที่มีมาตรฐานที่ชัดเจน และทำให้สามารถใช้ร่วมกับระบบอื่น ทั้งที่มีอยู่แล้ว และระบบใหม่ๆที่จะนำมาใช้งานในอนาคตได้ โดยผ่านโปรโตคอล LDAP
7. สามารถสร้างระบบที่มีความสามารถในการอ่านข้อมูล ค้นหาข้อมูล เพิ่ม ลบ ได้อย่างถูกต้องและรวดเร็ว

3.2 การออกแบบระบบการทำงาน

เมื่อได้ศึกษาถึงความต้องการของระบบแล้ว จึงทำการวิเคราะห์ระบบในเชิงการไหลของข้อมูล ดังแสดงได้ตาม Data Flow Diagram ต่อไปนี้



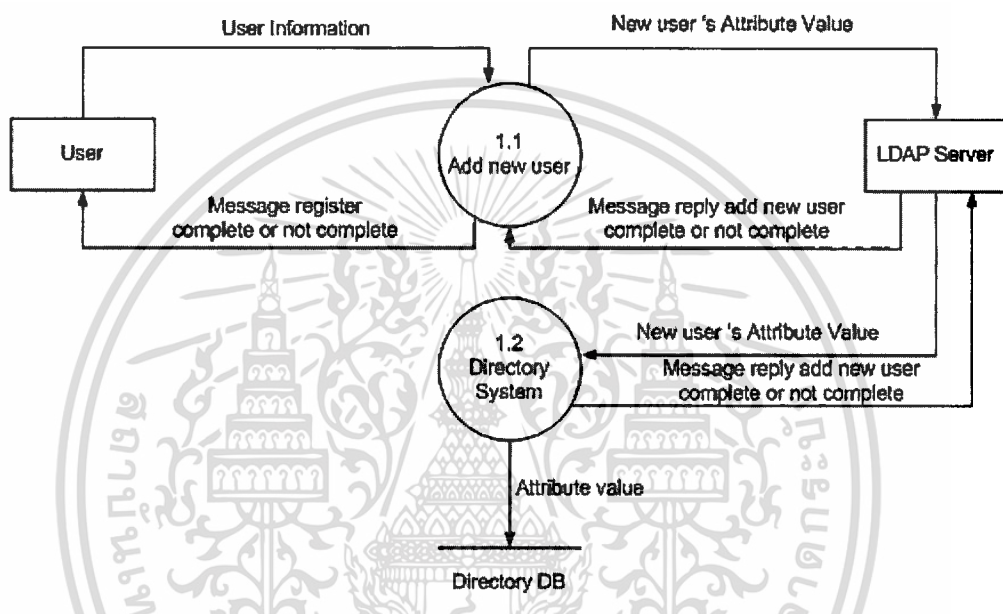
รูปที่ 3.1 ภาพแสดง Context Diagram ของระบบ Directory Service for Internet Chat using LDAP



รูปที่ 3.2 ภาพแสดง Dataflow Diagram Level 1 ของระบบ Directory Service for Internet Chat using LDAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน Dataflow Diagram ในภาพ 3.2 เป็นการแสดงการทำงานของระบบโดยรวม โดยจะมี ส่วนการทำงานของระบบแบ่งเป็น 2 ส่วน เริ่มที่ Registration จะเป็นระบบของการลงทะเบียน สมาชิกใหม่ในการใช้งาน โปรแกรม Internet Chat โดยการสมัครจะต้องทำการกรอกข้อมูล ตามที่ระบบกำหนดให้ครบ ส่วนอีกส่วนหนึ่งเป็นส่วนของ Log on S เป็นระบบที่สำหรับผู้ใช้ที่เป็น สมาชิกแล้วจะเข้ามาทำการใช้งานระบบโดยเมื่อผู้ใช้มีรหัสที่ถูกต้อง Log on ผ่านก็สามารถใช้งาน ระบบได้ตามต้องการ

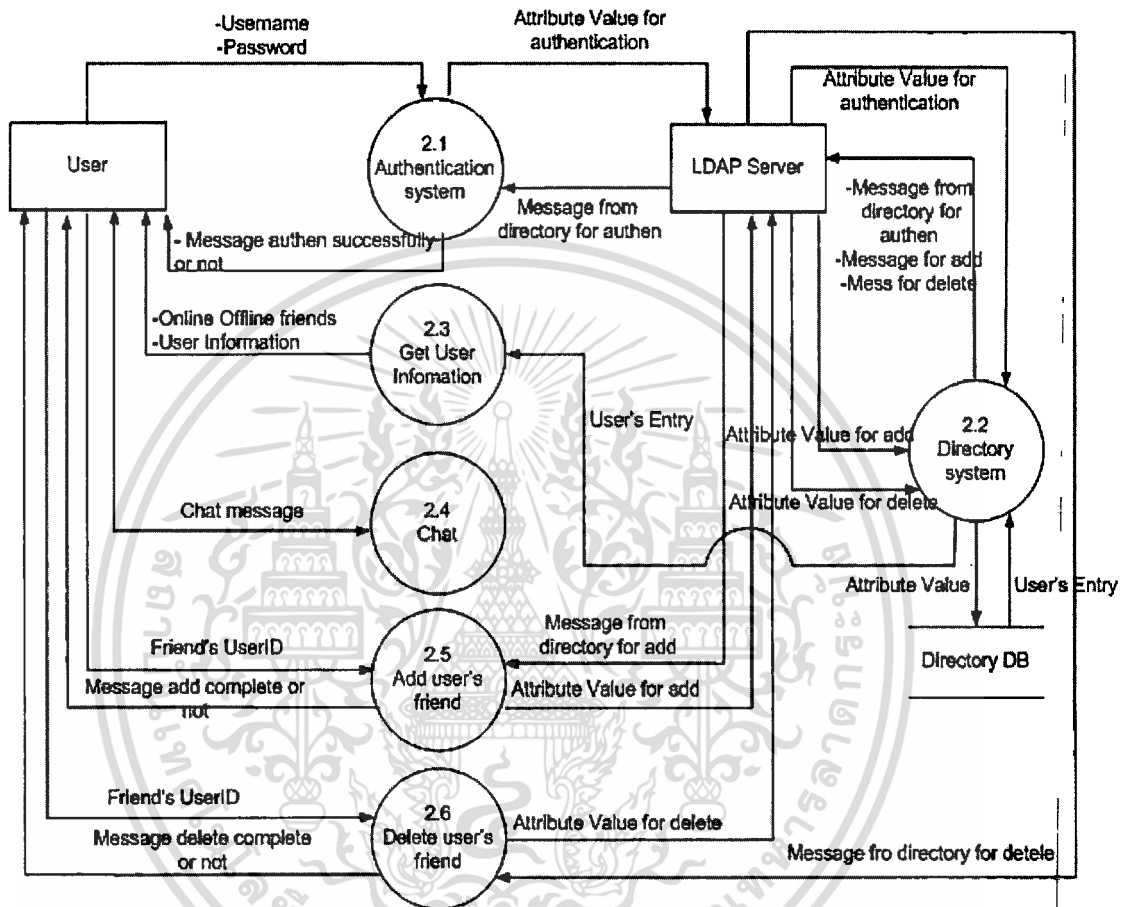


รูปที่ 3.3 ภาพแสดง Dataflow Diagram Level 2 ของ Process Registration

ใน Dataflow Diagram รูปที่ 3.3 แสดงถึงรายละเอียดขั้นตอนการทำงานของ Registration แบ่งออกเป็น 2 ส่วนย่อย ๆ โดยแต่ละส่วนมีหน้าที่ดังนี้ Add new user จะทำการรับข้อมูลจากผู้ใช้ และตรวจสอบความถูกต้องของข้อมูล format ในแต่ละ field รวมถึงตรวจสอบว่าผู้ใช้กรอกข้อมูล ให้ครบตาม field ที่จะต้องมีหรือไม่ ถ้าข้อมูลถูกต้องครบถ้วนก็สามารถที่จะทำการ Regist ได้ซึ่งในส่วนนี้เมื่อ Process ได้รับ Information ต่างๆ มากี่จะต้องทำการแปลงให้อยู่ในรูป Attribute ซึ่งตรงในส่วนของ Schema ที่ได้ออกแบบไว้ และก็จะทำการส่ง Attribute ของผู้ใช้ที่ต้องการเพิ่มลงในระบบเพิ่มผ่าน Web Server ให้ส่งผ่านโปรโตคอล LDAP อีกทีเพื่อที่จะทำการติดต่อกับระบบ Directory และทำการเพิ่มข้อมูลของผู้ใช้ลงในบัญชีผู้ใช้ใน Directory ส่วนระบบ Directory เป็นระบบที่จะรับข้อมูลค่า Attribute ต่างๆเข้ามาแล้วทำการเก็บลงใน Directory DB ซึ่งระบบจะทำการ

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

Generate UserID ให้ผู้ใช้ถ้าการลงทะเบียนเป็นสำเร็จ และยังคงตรวจสอบในเรื่องของความถูกต้องของข้อมูลที่จะเก็บลงใน Schema ด้วยเช่น Attribute บางชนิดต้องมีค่าเป็นต้น หลังจากนั้นระบบก็จะทำการส่งข้อความบอกผู้ใช้ว่าการ Register นั้นเสร็จสมบูรณ์หรือไม่ถ้าไม่เป็นเพราะสาเหตุใด



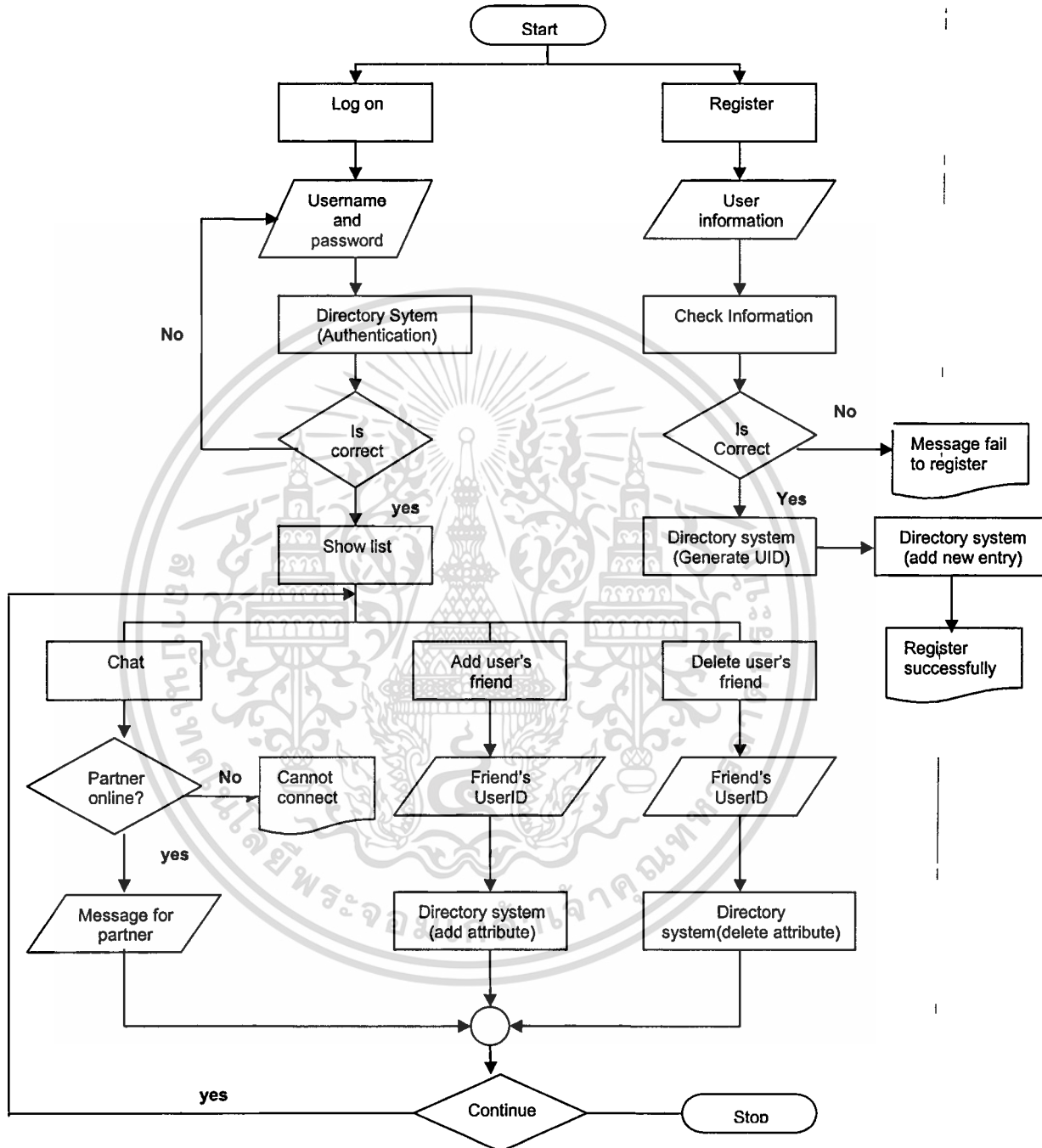
รูปที่ 3.4 ภาพแสดง Dataflow Diagram Level 2 ของ Process Log On

ใน Dataflow Diagram รูปที่ 3.4 แสดงถึงรายละเอียดของ Log on ซึ่งแบ่งเป็นส่วนย่อยๆ 6 ส่วน คือ Authentication system จะทำหน้าที่ในการรับ username และ password ของผู้ใช้ และจะทำการเปลี่ยนให้อยู่ในรูปแบบ Attribute ที่จะนำไปค้นหาในระบบ Directory และทำหน้าที่ติดต่อกับระบบ Directory ด้วยโปรโตคอล LDAP ผ่าน Operation ต่างๆ ของ LDAP ซึ่งเป็นการ Authentication และเพื่อค้นหาข้อมูลของผู้ใช้ถ้ามีเก็บอยู่ก็จะทำการดึงข้อมูลของผู้ใช้คนนั้นขึ้นมาแสดงบนหน้าจอ ถ้าผู้ใช้มีกรไ้ username และ password ผิดพลาด หรือ ไม่มีผู้ใช้นั้นในระบบ

Directory ก็จะทำการแสดงข้อความบอกให้ผู้ใช้ทราบ ส่วนระบบ Directory system เป็นส่วนการทำงานของระบบ Directory ซึ่งจะทำการค้นหาข้อมูล เพิ่มข้อมูล ลบข้อมูล ตาม Attribute Value ที่ได้รับและทำการส่งข้อมูลหรือข้อความต่างๆ ให้กับระบบที่เรียกใช้ หลังจากที่ผู้ใช้สามารถทำการ Log on เข้าระบบได้แล้วจะมี Process อีก 4 ส่วนในระบบ คือ ส่วน Get User Information Chat Add user's friend Delete user's friend ส่วนของ Get User Information จะเป็นส่วนที่ดึงข้อมูลของผู้ใช้จากระบบ Directory system ขึ้นมาแสดงให้ผู้ใช้ทราบบนหน้าจอ Browser เพื่อให้ผู้ใช้สามารถใช้งานในส่วนอื่นได้ต่อไป โพรเซส Chat จะทำหน้าที่ในการรับส่งข้อความในการสนทนาของผู้ใช้ โดยที่ข้อความนี้จะส่งกันผ่าน Web Server โดยที่ระบบจะทำการตรวจสอบสถานะของคู่สนทนาว่า Online เท่านั้นที่จะคุยได้ Add user's friend คือส่วนที่จะทำการเพิ่มรายชื่อเพื่อนของผู้ใช้ใน List หน้าจอของผู้ใช้ โดยโพรเซสจะทำการติดต่อกับ Directory ด้วย LDAP ผ่าน Web Server และจะต้องส่งค่าของ Attribute Value ให้กับ Directory system เพื่อให้ระบบ Directory ทำการเพิ่มค่า Attribute Value ลงใน Entry ของผู้ใช้ให้ตรงตาม Schema ที่ได้ออกแบบไว้ ส่วน Delete user's friend เป็นส่วนที่จะทำการลบรายชื่อเพื่อนของผู้ใช้ตามที่ผู้ใช้ต้องการ โดยโพรเซสจะรับค่า UserID ของเพื่อนที่ต้องการลบและเปลี่ยนค่านั้นเป็น Attribute Value และทำการติดต่อกับ Directory ด้วย LDAP ส่งค่า Attribute Value ที่ต้องการลบไปยังระบบ Directory system เพื่อให้ Directory ทำการลบ Attribute Value ใน Entry ของผู้ใช้

3.3 กระบวนการทำงานของระบบ

ขั้นตอนการทำงานของระบบ Internet Chat มีดังนี้

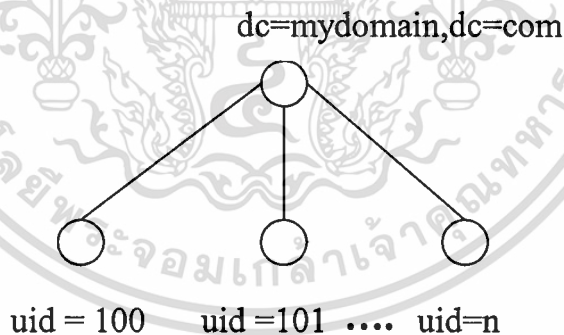


รูปที่ 3.5 ภาพแสดง Flow Chart การทำงานของระบบ Directory Service for Internet Chat using LDAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Flow Chart ในรูปที่ 3.5 อธิบายขั้นตอนการทำงานของระบบได้ดังนี้ เมื่อผู้ใช้เริ่มต้นเข้ามาใช้งานในระบบโดยผ่าน Web Browser จะมีทางเลือกคือถ้าเป็นผู้ใช้งานรายใหม่จะต้องทำการสมัครสมาชิกก่อนแต่ถ้าเป็นผู้ที่เป็นสมาชิกในระบบอยู่แล้วก็สามารถนำ username และ password ที่เคยได้ใช้ในการ Log on เข้าระบบได้เลย สำหรับการทำงานของระบบสมัครสมาชิกคือเมื่อผู้ใช้กรอกข้อมูลเข้าสู่ระบบระบบจะทำการตรวจสอบข้อมูลว่าถูกต้องครบถ้วนหรือไม่และระบบจะทำการ Generate User Id เพื่อสำหรับเป็น username ของผู้ใช้และทำการบันทึกผู้ใช้งานลงในระบบ Directory สำหรับระบบ Log on จะทำการ Authentication ผู้ใช้งานในระบบด้วย username และ password ที่ผู้ใช้ได้ใส่เข้ามาถ้าถูกต้องระบบจะทำการแสดงรายละเอียดต่างๆ ของผู้ใช้งานเช่นชื่อที่ Display หรือ List รายชื่อของเพื่อนขึ้นมาบนหน้าต่างโปรแกรมและในการใช้งานผู้ใช้งานสามารถที่จะเลือกใช้งาน โปรแกรมตามต้องการดังนี้คือสามารถสนทนากับคู่สนทนาโดยที่ข้อความจะผ่านทาง Web Server หรือผู้ใช้สามารถที่จะเพิ่มรายชื่อเพื่อนลงใน List ของตนได้โดยที่จะต้องใส่ UserID ของเพื่อนลงไปในระบบแล้วระบบ Directory จะทำการเพิ่มชื่อเพื่อนให้และสุดท้ายผู้ใช้สามารถที่จะลบชื่อเพื่อนออกจาก List โดยที่จะต้องใส่ UserID เพื่อนแล้วระบบจะทำการรับชื่อเพื่อนคนที่ต้องการจะลบเข้ามาเพื่อลงให้ระบบ Directory ทำการลบข้อมูลของเพื่อนให้

3.4 โครงสร้างต้นไม้ (Tree) ในระบบ



รูปที่ 3.6 ภาพแสดงโครงสร้างของต้นไม้ในระบบ Internet Chat

ในระบบได้ทำการออกแบบโครงสร้างต้นไม้ (Tree) ให้เป็นแบบ Flat เพื่อเก็บข้อมูลในระบบ Directory จะทำการเก็บข้อมูลผู้ใช้ไว้ใน Entry ที่ชื่อว่า uid ทั้งหมด Attribute uid ในรูปแบบที่เราใช้การสร้างเป็นแบบตัวเลขที่ต่อเนื่องกัน โดยจะใช้ Attribute uid ในการอ้าง RDN สำหรับ Directory

Attribute uid นี้ต้องกำหนดเป็น Uniquess เพื่อป้องกันปัญหาการใช้ชื่อซ้ำซ้อนและเพื่อเป็นการรวดเร็วต่อการค้นหาข้อมูลผู้ใช้และไม่ซับซ้อนมากนักในการจัดการ

3.5 โครงสร้างของ Schema

ในการเก็บข้อมูลของผู้ใช้งานระบบเพื่อให้สามารถเก็บข้อมูลของผู้ใช้งานในระบบได้อย่างครบถ้วนและถูกต้องและสามารถพัฒนาโปรแกรมได้ เราจึงต้องออกแบบลักษณะของข้อมูลและกลุ่มของข้อมูลที่มีความสัมพันธ์ โดยจะใช้รูปแบบ Schema ที่เรียกว่า Slapd.conf ซึ่ง Slapd.conf นี้จะเป็นไฟล์ซึ่งเก็บ Schema ต่าง ๆ ใน Ldap Server โดยจะแบ่งออกเป็นกลุ่มดังนี้

Attribute 'cn' เป็น Attribute ที่จำเป็นต้องมีเมื่อจะใช้ objectclass ตามที่ออกแบบ

```
attributetype ( 2.5.4.3 NAME ('cn' 'commonName' )
    DESC 'RFC2256: common name(s) for which the entity is known by'
    SUP name )
```

Attribute 'sn' ใช้ในการเก็บนามสกุลของผู้ใช้

```
attributetype ( 2.5.4.4 NAME ('sn' 'surname' )
    DESC 'RFC2256: last (family) name(s) for which the entity is known by'
    SUP name )
```

Attribute 'gn' ใช้ในการเก็บทั้งชื่อและนามสกุลของผู้ใช้

```
attributetype ( 2.5.4.42 NAME ('givenName' 'gn' )
    DESC 'RFC2256: first name(s) for which the entity is known by'
    SUP name )
```

Attribute 'uid' ใช้ในการเก็บ username ของผู้ใช้ซึ่งจะเป็นตัวเลขที่ระบบ Generate ให้

```
attributetype ( 0.9.2342.19200300.100.1.1 NAME ('uid' 'userid' )
    DESC 'RFC1274: user identifier'
    EQUALITY caseIgnoreMatch
    SUBSTR caseIgnoreSubstringsMatch
```

SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{256})

Attribute 'userPassword' ใช้ในการเก็บ password ของผู้ใช้

attributetype (2.5.4.35 NAME 'userPassword'

DESC 'RFC2256/2307: password of user'

EQUALITY octetStringMatch

SYNTAX 1.3.6.1.4.1.1466.115.121.1.40{128})

Attribute 'description' ใช้ในการเก็บรายการชื่อเพื่อนของผู้ใช้

attributetype (2.5.4.13 NAME 'description'

DESC 'RFC2256: descriptive information'

EQUALITY caseIgnoreMatch

SUBSTR caseIgnoreSubstringsMatch

SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{1024})

Attribute 'title' ใช้ในการเก็บชื่อ display บนระบบ chat ของผู้ใช้

attributetype (2.5.4.12 NAME 'title'

DESC 'RFC2256: title associated with the entity'

SUP name)

Attribute 'destinationIndicator' ใช้เพื่อให้ระบบเก็บสถานะ online / offline ของผู้ใช้

attributetype (2.5.4.27 NAME 'destinationIndicator'

DESC 'RFC2256: destination indicator'

EQUALITY caseIgnoreMatch

SUBSTR caseIgnoreSubstringsMatch

SYNTAX 1.3.6.1.4.1.1466.115.121.1.44{128})

Attribute 'mail' ใช้เพื่อเก็บ e-mail address ของผู้ใช้

```
attributetype ( 0.9.2342.19200300.100.1.3 NAME ('mail' 'rfc822Mailbox')
    DESC 'RFC1274: RFC822 Mailbox'
    EQUALITY caseIgnoreIA5Match
    SUBSTR caseIgnoreIA5SubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26{256} )
```

Attribute "initials" เพื่อเก็บสถานะในการ notify User ว่าถูก add โดย User คนอื่น

```
attributetype ( 2.5.4.43 NAME 'initials'
    DESC 'RFC2256: initials of some or all of names, but not the surname(s).'
    SUP name )
```

Attribute 'ipHostNumber' ใช้ในการเก็บ ip address ของผู้ใช้

```
attributetype ( 1.3.6.1.1.1.19 NAME 'ipHostNumber'
    DESC 'IP address'
    EQUALITY caseIgnoreIA5Match
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26{128} )
```

โดยที่แต่ละ Attribute จะถูกกำหนดไว้ใน objectclass ดังนี้

```
objectclass ( 1.3.6.1.1.1.2.6 NAME 'ipHost' SUP top AUXILIARY
    DESC 'Abstraction of a host, an IP device'
    MUST ( cn $ ipHostNumber )
    MAY ( 1 $ description $ manager ) )
```

```
objectclass ( 2.5.6.6 NAME 'person'
    DESC 'RFC2256: a person'
    SUP top STRUCTURAL
    MUST ( sn $ cn )
```

```

MAY ( userPassword $ telephoneNumber $ seeAlso $ description )
Objectclass ( 2.16.840.1.113730.3.2.2 NAME 'inetOrgPerson'
DESC 'RFC2798: Internet Organizational Person'
SUP organizationalPerson STRUCTURAL
MAY (
audio $ businessCategory $ carLicense $ departmentNumber $
displayName $ employeeNumber $ employeeType $ givenName $
homePhone $ homePostalAddress $ initials $ jpegPhoto $
labeledURI $ mail $ manager $ mobile $ o $ pager $
photo $ roomNumber $ secretary $ uid $ userCertificate $
x500uniqueIdentifier $ preferredLanguage $
userSMIMECertificate $ userPKCS12 ) )

```

```

objectclass ( 2.5.6.7 NAME 'organizationalPerson'
DESC 'RFC2256: an organizational person'
SUP person STRUCTURAL
MAY ( title $ x121Address $ registeredAddress $ destinationIndicator $
preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $
telephoneNumber $ internationaliSDNNNumber $
facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $
postalAddress $ physicalDeliveryOfficeName $ ou $ st $ l ) )

```

ในการออกแบบ Schema สามารถทำได้ 2 วิธีคือ จะทำการสร้าง Objectclass โดยสร้าง Attribute เองหรือทำการเลือกจาก Default Schema ที่จะคิดมาตอนที่ทำการติดตั้งระบบ Directory ในระบบนี้จะใช้การเลือก Attribute จาก Schema ที่ระบบ Directory มีให้เนื่องจากมีมากเพียงพอที่จะใช้งานและคุณสมบัติเข้ากับระบบได้ ผู้พัฒนาจะคำนึงถึง Attribute ที่จะสามารถใช้เก็บข้อมูลต่าง ๆ ในระบบก่อนหลังจากนั้น จะต้องทำการดูว่า Attribute ที่เลือกมาอยู่ใน Objectclass ใด ใน Schema ของ Directory และแต่ละ Objectclass ที่เลือกเป็น sup ของ Objectclass ใดบ้าง

3.6 รูปแบบเพิ่มข้อมูลของโปรแกรม

โปรแกรมนี้ได้รับการออกแบบ และกำหนด Folder สำหรับเก็บข้อมูลและโปรแกรมที่ใช้ร่วมกันกับระบบเพื่อให้ง่ายต่อการเปลี่ยนแปลงแก้ไข และง่ายในการ Update ระบบในภายหลัง ซึ่งจะมีรายละเอียดดังต่อไปนี้

- 3.6.1 /usr/local/apache เป็น Directory ที่ใช้ในการเก็บโปรแกรม Apache ซึ่งทำหน้าที่เป็น Web Server โดยระบบพัฒนาโดยใช้ Apache 1.3.27
- 3.6.2 /usr/local/BerkleyDB เป็น Directory ที่ใช้ในการเก็บโปรแกรม Berkley DB ซึ่งทำหน้าที่เป็นฐานข้อมูลที่ใช้เก็บข้อมูลของผู้ใช้งานในระบบทั้งหมด โดยระบบพัฒนาโดยใช้เวอร์ชัน 4.1.25
- 3.6.3 /usr/local/Openldap เป็น Directory ที่ใช้ในการเก็บโปรแกรม Openldap ทำหน้าที่เป็น LDAP Server ซึ่งเป็น Directory Server บริการบัญชีผู้ใช้งานทั้งหมด
- 3.6.4 /usr/local/php เป็น Directory ที่ใช้ในการเก็บโปรแกรม PHP ซึ่งทำหน้าที่เป็น API เชื่อมการติดต่อระหว่าง Web Server กับ LDAP Server ระบบพัฒนาโดยใช้ php 4.3.2
- 3.6.5 /usr/local/web เป็น Directory ที่ใช้ในการเก็บหน้าจอบ Web Page ที่จะส่งการแสดงผลไปยังผู้ใช้งาน

3.7 ส่วนติดต่อผู้ใช้งาน (User Interface)

ในส่วนการติดต่อกับผู้ใช้งานนั้น หน้าจอได้ถูกออกแบบให้ง่ายต่อการใช้งาน และได้ทำการซ่อนการแสดงผลส่วนที่ไม่จำเป็นต่อผู้ใช้งานเพื่อป้องกันความสับสนในการใช้งานและยังสามารถลดความผิดพลาดในการใช้งานได้ด้วย

Register System

First Name :*

Last Name :*

Email Address :*

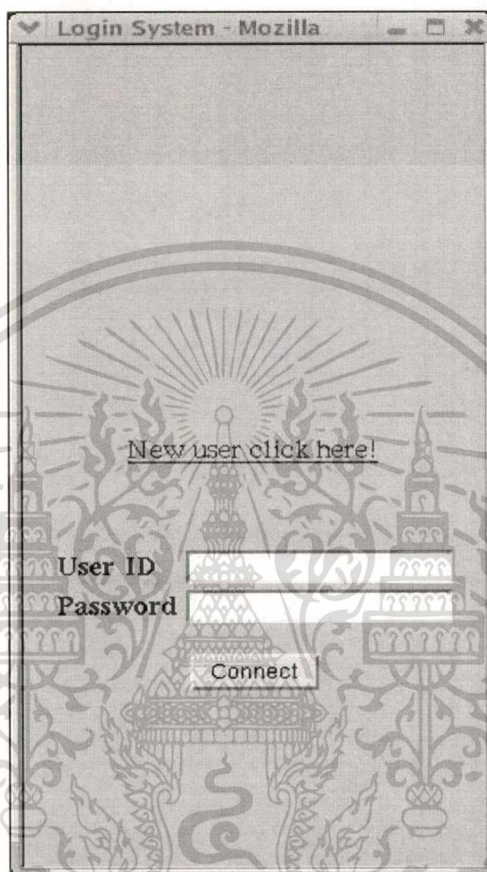
Password :*

Confirm Password:*

Register

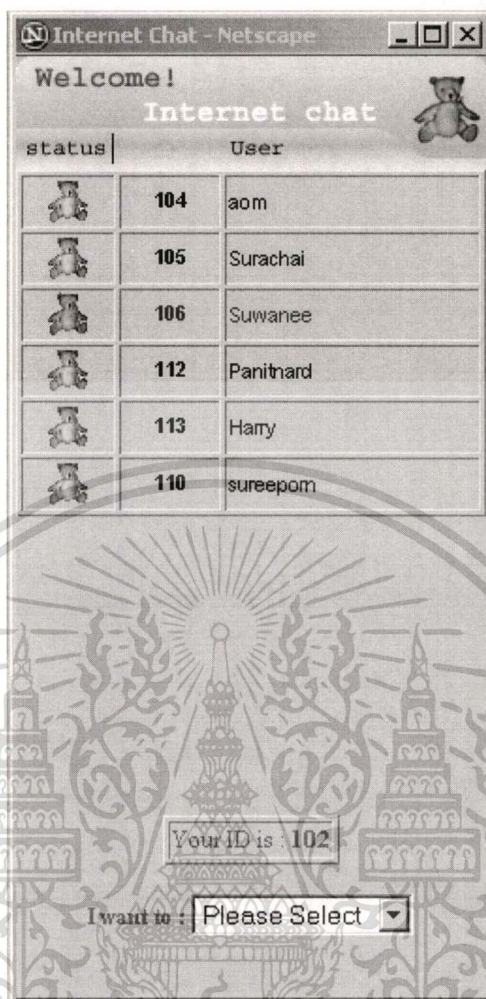
รูปที่ 3.7 แสดงหน้าจอการสมัครเข้าใช้งานในระบบ

ในการเข้าใช้งานระบบ Internet Chat จะต้องมีการสมัคร โดยผ่านทาง Register System เพื่อที่จะได้รับหมายเลข userID เพื่อใช้เป็น username ในระบบและเพื่อให้ระบบ Directory ทำการบันทึกข้อมูลผู้ใช้งานไว้ดังแสดงในรูปที่ 3.7



รูปที่ 3.8 แสดงหน้าจอตรวจสอบสิทธิ์การใช้งาน

ในการเข้าใช้งานโปรแกรมนั้นผู้ที่จะสามารถเข้าใช้งานได้ต้องผ่านการ Authentication ของระบบเพื่อความปลอดภัยแก่ผู้ใช้งานและระบบด้วยโดยต้องใส่ Username และ Password ของผู้ใช้และสำหรับผู้ที่ยังไม่ได้มีชื่อในระบบสามารถสมัครได้โดย Click ที่ New user click here! ดังแสดงในรูปที่ 3.8

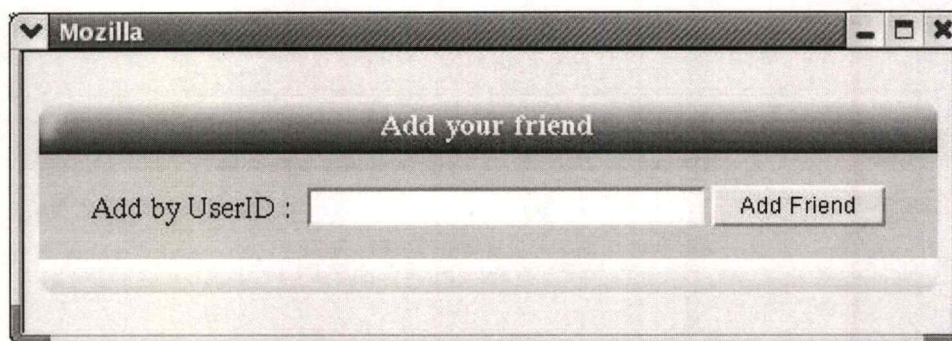


รูปที่ 3.9 แสดงหน้าจอที่ผู้ใช้ผ่านการ Authentication แล้วจะแสดงหน้าจอใน

รูปที่ 3.8 แสดงเมื่อผู้ใช้งานทำการ Login สำเร็จแล้วจะสามารถเข้าใช้งานโปรแกรมได้ โดยที่หน้าจอโปรแกรมจะแสดง userID ของผู้ใช้ให้ทราบและแสดงรายชื่อของเพื่อนทุกคนที่มีอยู่ของผู้ใช้รวมทั้งจะแสดงสถานะ Online-Offline เพื่อเป็นการบอกว่าสามารถจะสนทนากับใครได้บ้าง ในหน้าจอนี้จะมี function ให้ผู้ใช้งานได้เลือกใช้โดยที่ผู้ใช้จะเลือกได้จาก Drop Down Menu ดังนี้

- Add Contact คือการที่ผู้ใช้สามารถเลือกเพื่อที่จะ Add รายชื่อเพื่อนลงไปใน List
- Delete Contact คือการที่ผู้ใช้งานสามารถที่จะเลือกเพื่อที่จะ Delete เพื่อนคนใดคนหนึ่งได้
- Sign Out คือการที่ผู้ใช้งานเลือกเพื่อที่จะออกจากระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



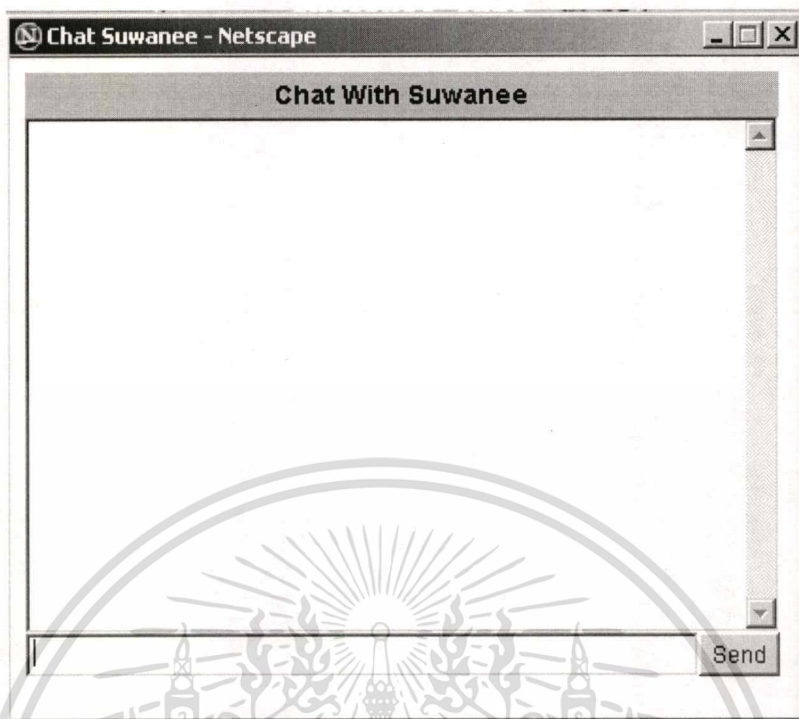
รูปที่ 3.10 แสดงหน้าจอ Add Contact

รูปที่ 3.10 แสดงหน้าจอ Add Contact ให้ผู้ใช้สามารถที่จะ Add เพื่อนที่จะสนทนาด้วยลงไปใน List รายชื่อเพื่อนโดยการ Add นั้นผู้ใช้จะต้องรู้ userID ของเพื่อนที่จะ Add เพื่อความปลอดภัยสำหรับผู้ใช้งาน



รูปที่ 3.11 แสดงหน้าจอ Delete Contact

รูปที่ 3.11 แสดงหน้าจอ Delete Contact ผู้ใช้สามารถที่จะ Delete เพื่อนคนใดคนหนึ่งออกจาก List รายชื่อที่มีอยู่ได้โดยใช้ UserID ของเพื่อนคนที่ต้องการจะ Delete และกดปุ่ม Delete



รูปที่ 3.12 แสดงหน้าจอ Chat

รูปที่ 3.12 แสดงหน้าจอ Chat จะเป็นหน้าจอเพื่อให้ผู้ใช้งานสามารถที่จะส่งข้อความสนทนาสนทนากันได้โดยที่ผู้ใช้งานจะทำการ Double Click ที่ชื่อผู้คนที่ต้องการสนทนาด้วยก็จะปรากฏหน้าต่างขึ้นมาและมีชื่อของเพื่อนที่จะสนทนาด้วยบอกที่มุมบนของหน้าต่างผู้ใช้สามารถพิมพ์ข้อความลงใน Text Box และกดปุ่ม Send หลังจากนั้นข้อความที่พิมพ์จะปรากฏที่ Text Area ข้างบนทำให้สามารถเลื่อนดูข้อความเก่าได้

บทที่ 4

การพัฒนาระบบงาน

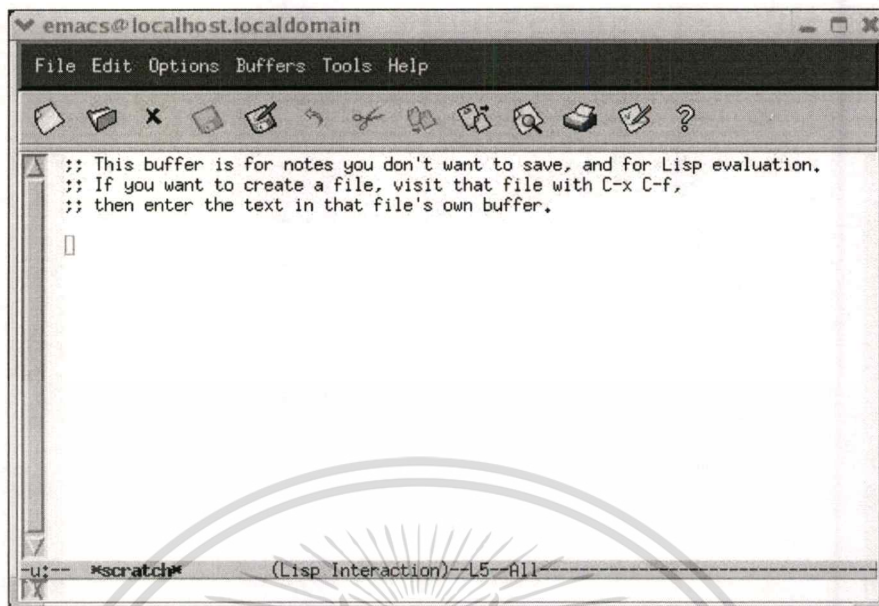
4.1 ขั้นตอนการพัฒนาาระบบ Directory

1. ศึกษาระบบ Internet Chat ที่มีอยู่ปัจจุบันลักษณะการทำงานของระบบ และความเป็นไปได้ในการพัฒนาระบบและกำหนดขอบเขตในการพัฒนาระบบ
2. ศึกษาทฤษฎีต่าง ๆ เกี่ยวกับระบบ Directory และรวมไปเรียนรู้ที่จะใช้งานเครื่องมือที่ช่วยในการพัฒนา Directory Server
3. คุณสมบัติของระบบ ทำการกำหนดความสามารถและออกแบบพัฒนา ให้สามารถรองรับกับความต้องการของผู้ใช้งานในระบบ และอยู่ในขอบเขตที่สามารถเป็นไปได้ในการพัฒนา
4. ทำการเลือกเครื่องมือที่จะใช้ในการพัฒนาโครงการโดยพิจารณาตามความเหมาะสมในการใช้งาน
5. ทำการพัฒนาโปรแกรมและทำการทดสอบ โปรแกรมเพื่อให้โปรแกรมทำงานได้อย่างถูกต้องและตรงตามความต้องการของระบบที่ได้ออกแบบไว้

4.2 การพัฒนาโปรแกรม Internet Chat

ในส่วนการพัฒนาโปรแกรม Internet Chat นั้นส่วนการทำงานได้ออกแบบให้มีการติดต่อผู้ใช้งานผ่าน Browser จึงจำเป็นต้องติดตั้ง Web Server ขึ้นมาก่อน โดยได้เลือกใช้โปรแกรม Web Server ของ Apache เนื่องจากสามารถทำงานได้รวดเร็ว และสามารถระบุเงื่อนไขการทำงานได้สะดวก และในส่วนของการติดต่อระหว่าง Web Server กับ LDAP นั้น เพื่อให้สามารถประสานการทำงานได้อย่างถูกต้องรวดเร็ว จึงได้นำเอา ภาษาสคริป PHP มาใช้ร่วมกับ Apache Web Server เพื่อใช้เป็น Application Programming Interface (API) ติดต่อกับ LDAP Server เพื่อใช้ในการบริหารจัดการข้อมูลใน LDAP Server

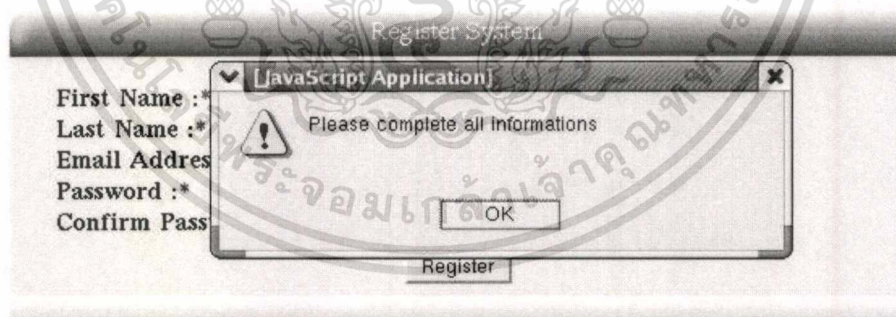
หลังจากได้ทำการติดตั้งโปรแกรมสนับสนุนต่าง ๆ แล้วจึงเลือก Editor ที่จะนำมาเขียนพัฒนาโปรแกรมผู้เขียนเลือกใช้ โปรแกรม Emacs ที่มีให้บนระบบปฏิบัติการ Linux ซึ่งเป็นโปรแกรมที่ใช้งานได้ง่ายและมีขนาดเล็ก



รูปที่ 4.1 แสดงหน้าจอของโปรแกรม Emacs

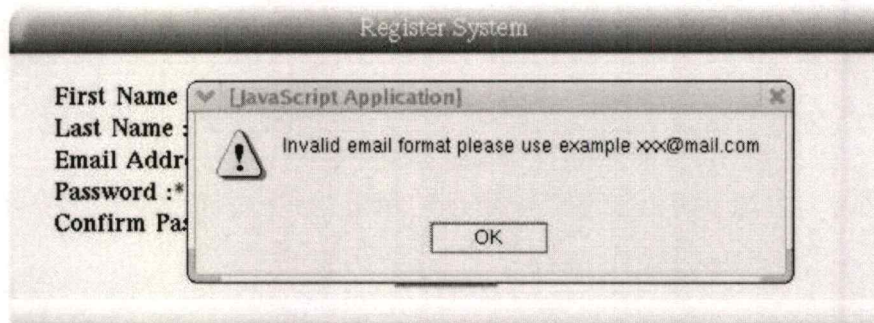
4.3 ทดสอบการใช้งานโปรแกรม

ระบบ Internet Chat ผู้ที่จะสามารถใช้งานได้ต้องเป็นผู้ที่ลงทะเบียนชื่อที่จะใช้เพื่อให้ระบบ Directory ทำการจัดเก็บข้อมูลเสียก่อนหลังจากนั้นจึงจะสามารถใช้งานระบบได้



รูปที่ 4.2 แสดงข้อความเตือนเมื่อผู้ใช้กรอกข้อมูลไม่ครบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงข้อความเตือนเมื่อผู้กรอกข้อมูล E-mail ผิดรูปแบบ



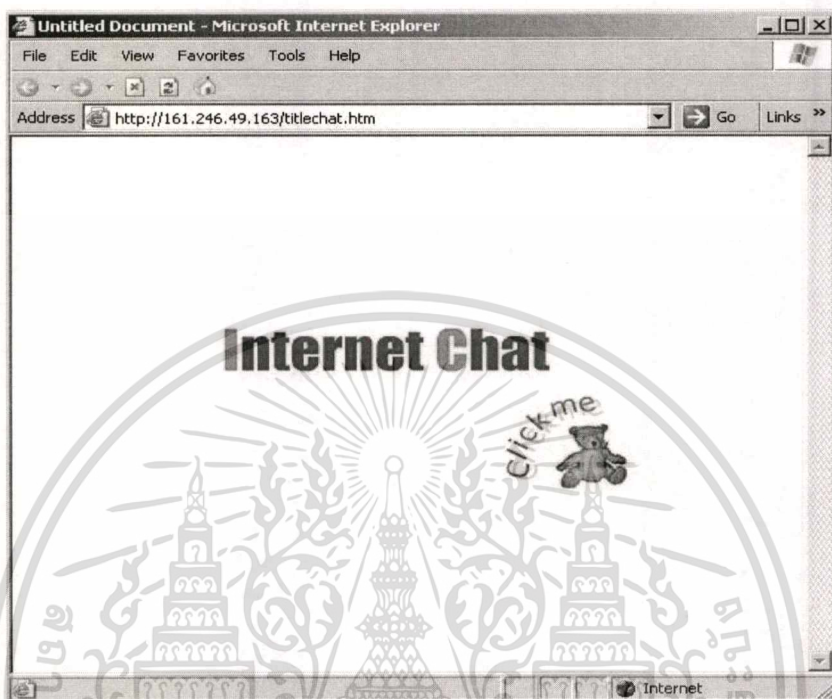
รูปที่ 4.4 แสดงข้อความเตือนเมื่อผู้ใช้ใส่ Password และ Confirm Password ไม่ตรงกัน



รูปที่ 4.5 แสดงข้อความเมื่อผู้ใช้สามารถลงทะเบียนผ่าน

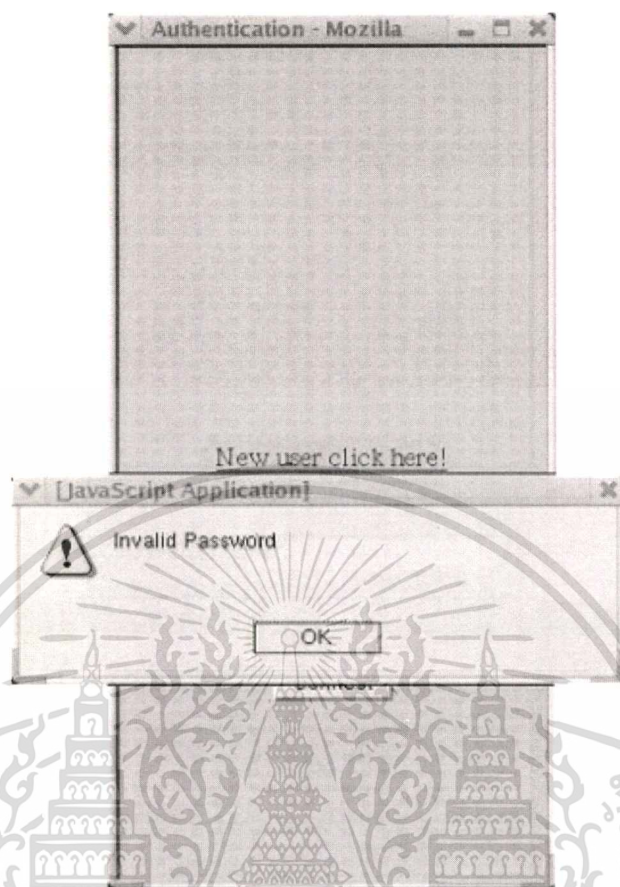
ในระบบ Registration จะมีการตรวจสอบความถูกต้องของข้อมูล que ผู้ใช้ได้ทำการกรอกถ้า ผู้ใช้กรอกข้อมูลไม่ครบตามที่ระบบต้องการหรือใส่ข้อมูลไม่ถูกต้องจะมีข้อความเตือนให้ผู้ใช้ทราบ มี 3 กรณีคือ ผู้ใช้กรอกข้อมูลไม่ครบทุกช่อง ผู้ใช้ใส่ Password และ Confirm Password ไม่ตรงกัน และกรณีที่ผู้ใช้ใส่ E-mail Address ผิดรูปแบบ เมื่อผู้ลงทะเบียนเรียบร้อยแล้วระบบจะทำการ

Generate UserID ให้เพื่อผู้ใช้งานจะต้องใช้เป็น Username ในการที่จะ Log in เข้าใช้งานระบบดังแสดง
 ในรูปที่ 4.2 ,4.3 ,4.4 ,4.5



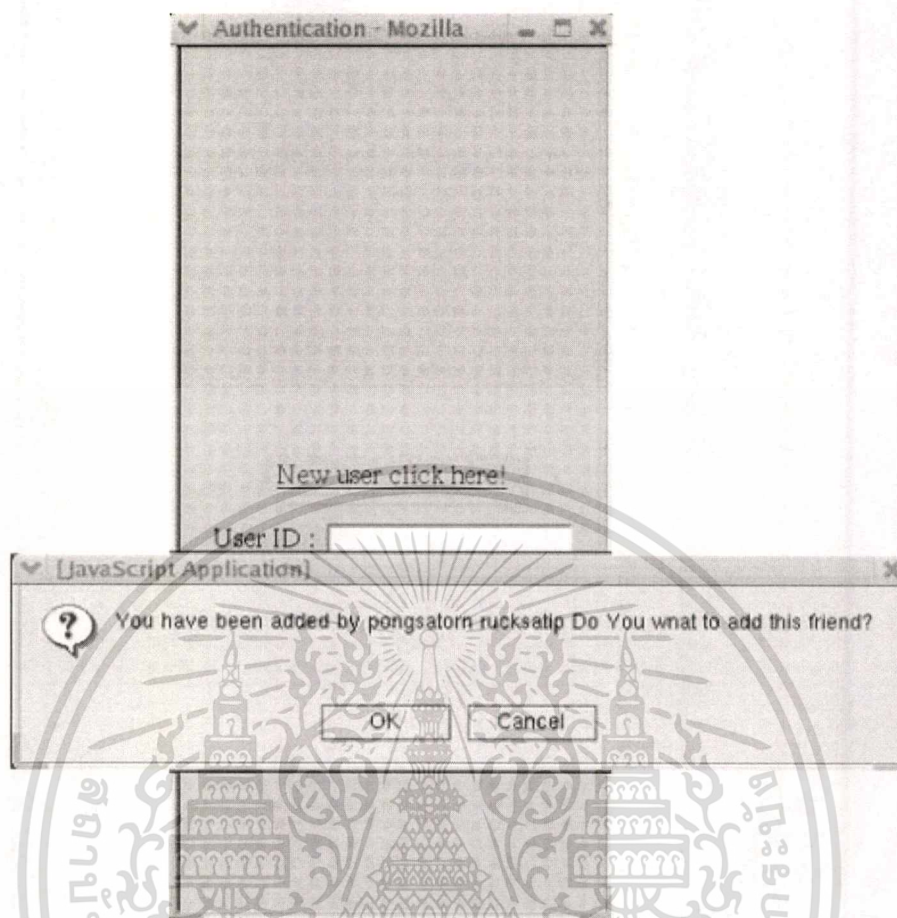
รูปที่ 4.6 แสดงหน้า Index ของระบบ

เมื่อจะเข้าระบบจะเริ่มที่หน้า Index ผู้ใช้จะต้องทำการ Click ที่สัญลักษณ์และระบบจะ Pop Up หน้าต่าง Log In ขึ้นมาเพื่อสามารถที่จะ Sign In ได้แสดงในรูปที่ 4.6



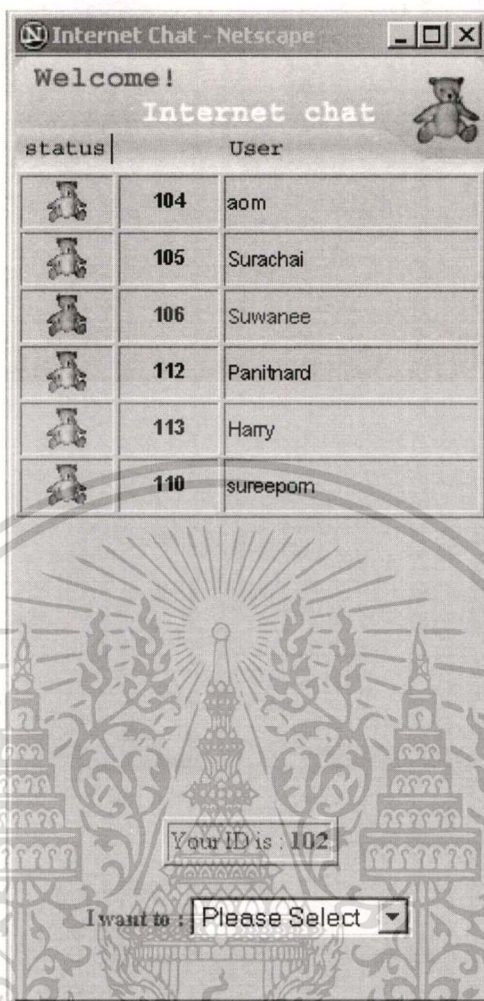
รูปที่ 4.7 แสดงข้อความเตือนเมื่อ Authentication ไม่ผ่าน

เมื่อผู้ใช้งานใส่ username และ password ระบบ Authentication จะทำการตรวจสอบว่ามีผู้ใช้อยู่ในระบบหรือไม่ถ้าไม่มีจะทำการเตือนว่าไม่มีผู้ใช้งานในระบบ และถ้ามีผู้ใช้งานในระบบแต่ผู้ใช้งานใส่ password ผิดระบบก็จะทำการแสดงข้อความเตือนดังแสดงในรูปที่ 4.7



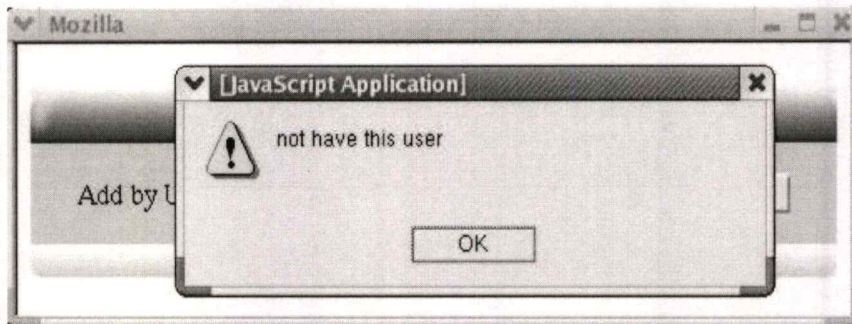
รูปที่ 4.9 แสดงให้เห็นว่าผู้ใช้ถูก Add โดยผู้ใช้คนอื่น

รูปที่ 4.9 แสดงให้เห็นถึงการที่ระบบมีการตรวจสอบเมื่อผู้ใช้ทำการ Log in จะทราบว่ามีใครบ้างที่ Add ผู้ใช้ไว้และจะแสดงชื่อของผู้ที่ Add ขึ้นเพื่อที่ผู้ใช้สามารถจะเลือกได้ว่าต้องการ Add เพื่อนคนนั้นไว้หรือไม่ต้องการก็สามารถ Add เพื่อนได้ทันทีโดยกดปุ่ม OK และเพื่อนคนนั้นก็จะมีอยู่ใน List ทันทีถ้าไม่ต้องการ ก็สามารถ Cancel ได้ โดยที่ชื่อของคนที่เราไม่ต้องการ Add จะไม่ปรากฏใน List เพื่อเป็นการปลอดภัยที่จะได้สนทนาเฉพาะผู้ที่ผู้ใช้ต้องการเท่านั้นได้ โดยถ้ามีการ Add ผู้ใช้จากคนหลายคนระบบก็จะแสดงหน้าต่างขึ้นมาเท่ากับจำนวนผู้ที่ได้ Add ผู้ใช้



รูปที่ 4.10 แสดงหน้าจอหลักเมื่อผู้ใช้ Log in ผ่าน

รูปที่ 4.10 แสดงเมื่อผู้ใช้ผ่านระบบ Authentication แล้วจะแสดงหน้าจอหลักโดยจะมีข้อมูลของผู้ใช้คือรายชื่อเพื่อนรวมไปถึงสถานะ Online-Offline ของเพื่อนแต่ละคนโดยที่ผู้ใช้จะสามารถเลือกสนทนาได้เฉพาะเพื่อนที่ Online อยู่เท่านั้น และเมื่อเพื่อนคนที่ Offline อยู่ Online ขึ้นมาระบบสามารถที่จะแสดงให้ผู้ใช้ทราบโดยทันทีโดยที่ สัญลักษณ์ของการ Online จะเป็นรูปสีเขียว และ Offline จะเป็นรูปสีแดงเมื่อเพื่อนเปลี่ยนสถานะผู้ใช้สามารถเห็นได้ทันที และผู้ใช้สามารถเลือกใช้ function ต่าง ๆ ของระบบได้จากหน้าจอนี้



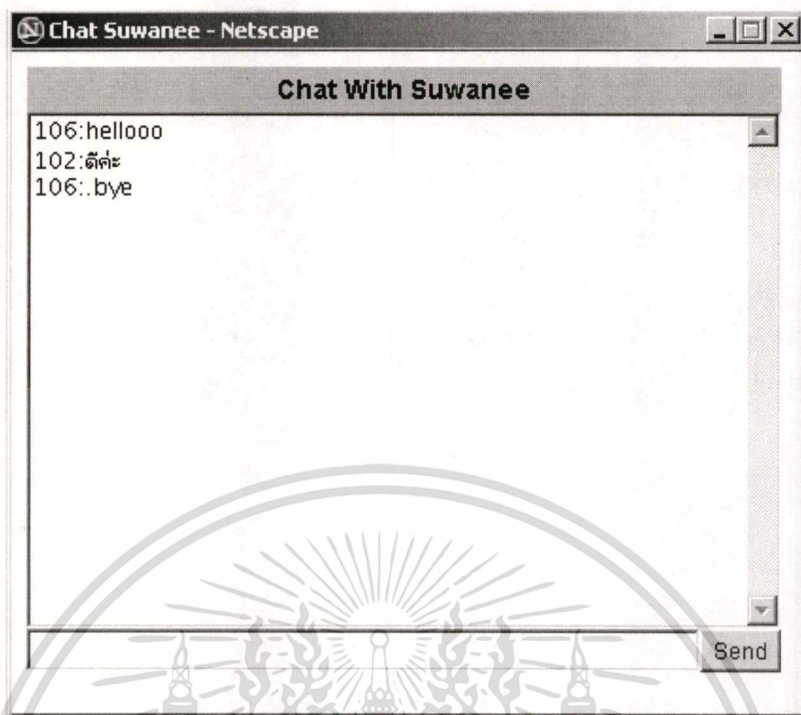
รูปที่ 4.11 แสดงข้อความเมื่อมีการ Add คนที่ไม่มีในระบบ

เมื่อผู้ใช้ทำการเลือก function Add Contact จะมีหน้าต่าง Pop Up ขึ้นมาให้ผู้ใช้ในการใส่ UserID ของเพื่อนที่ต้องการจะ Add ถ้าการ Add สำเร็จหน้าต่าง Pop Up จะปิดไปเองและมีชื่อเพื่อนคนนั้นอยู่ใน List ของผู้ใช้ทันที แต่ถ้าการ Add ไม่สำเร็จเช่น ผู้ใช้มีการใส่ UserID ที่ไม่มีอยู่ในระบบก็จะแสดงข้อความเตือนเหมือนรูปที่ 4.11 และจะกลับไปหน้า Add เหมือนเดิมเพื่อให้ใส่ใหม่



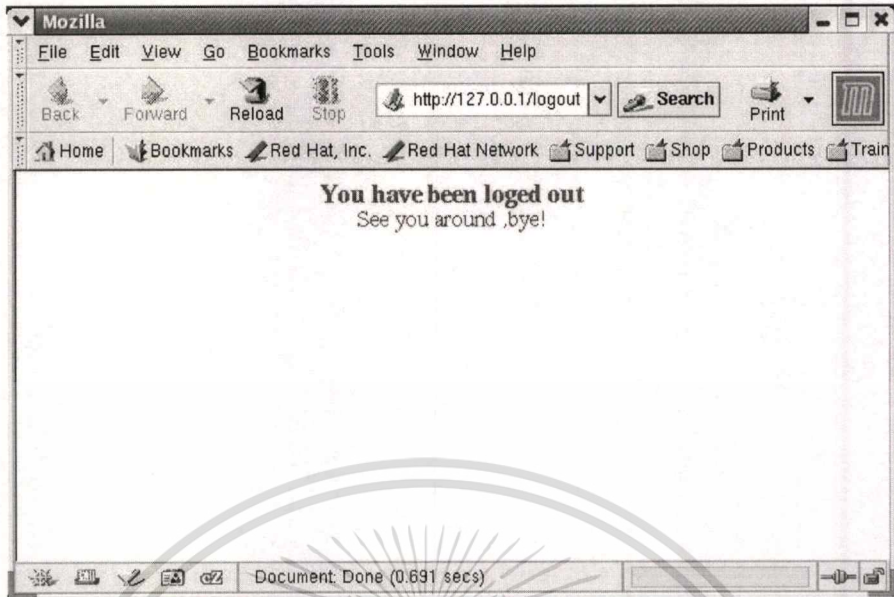
รูปที่ 4.12 แสดงข้อความเมื่อมีการ Delete ชื่อคนที่ไม่ได้อยู่ใน List

เมื่อผู้ใช้ทำการเลือก function Delete Contact ที่อยู่ในหน้าหลักก็จะมี Pop Up ขึ้นมาให้ผู้ใช้ในการใส่ UserID ของเพื่อนที่ต้องการจะ Delete ถ้าการ Delete สำเร็จหน้าต่างจะปิดไปและชื่อของเพื่อนที่ถูก Delete จะหายไปจาก List แต่ถ้าไม่สำเร็จเช่นผู้ใช้ใส่ UserID ของเพื่อนที่ไม่ได้มีอยู่ใน List ของตนก็จะแสดงข้อความเตือนเหมือนรูปที่ 4.12



รูปที่ 4.13 แสดงหน้าจอการสนทนาระหว่างผู้ใช้

เมื่อผู้ใช้ต้องการสนทนาต้องทำการ Double Click ที่ชื่อเพื่อนที่ต้องการสนทนาด้วยและจะมี Pop Up หน้าต่างการสนทนาของผู้ใช้กับเพื่อนคนนั้น โดยที่จะมีชื่อของเพื่อนแสดงอยู่มุมบนของหน้าต่างและมี UserID ของคู่สนทนาแสดงเพื่อทำให้สามารถทราบว่าประโยคข้อความที่ส่งเป็นของใครและสามารถย้อนดูข้อความเก่าได้แต่เมื่อคู่สนทนาคนใดคนหนึ่งปิดหน้าต่างไประบบจะมีการบอกให้ทราบตามตัวอย่างโดยจะมีคำว่า bye มาเมื่อต้องการคุยกับเพื่อนที่ Online คนใดคนหนึ่งต้องทำการคลิกเพื่อบอกให้คนนั้นทราบ โดยที่สัญลักษณ์ของเพื่อนคนนั้นจะมีการกระพริบว่าผู้ใช้ต้องการคุยด้วย ดังรูปที่ 4.13 จะเห็นว่าระบบสามารถแสดงได้ทั้งภาษาไทยและภาษาอังกฤษ



รูปที่ 4.14 แสดงหน้าจอเมื่อผู้ใช้ Sign Out

เมื่อผู้ใช้เลือก function Sign Out ก็จะปรากฏหน้าต่างเหมือนรูป 4.14 และสถานะของผู้ใช้จะเปลี่ยนเป็น Offline ทั้งนี้โดยที่ถ้ามีใครมีชื่อผู้ใช้อยู่ใน List ก็จะทำให้ผู้ใช้ได้ออกจากระบบแล้ว

บทที่ 5

สรุปผลการค้นคว้าและพัฒนาระบบ

5.1 ผลจากการพัฒนาระบบ

การพัฒนาระบบ Internet Chat โดยใช้ LDAP จากผลการใช้งานสามารถสรุปผลได้ดังนี้

5.1.1 การศึกษาการทำงานของ Directory Service

ผลการศึกษาพบว่า Directory Service มีประโยชน์อย่างมากในการจัดเก็บและบริหารบัญชีผู้ใช้งาน ลดความซ้ำซ้อนของข้อมูลที่เกี่ยวข้องอยู่หลาย ๆ ระบบเป็นผลให้ระบบมีความน่าเชื่อถือ และในการจัดเก็บสิ่งที่ทำให้ทราบคือ LDAP Server มีความยืดหยุ่นมากเนื่องจากมีโครงสร้างเป็น Tree ทำให้แต่ละ Entry ของผู้ใช้มีความยืดหยุ่นคือไม่จำเป็นต้องมี Attribute เหมือนกัน ทำให้สามารถประหยัดเนื้อที่ได้เป็นอย่างดีโดยจะสามารถที่จะเพิ่ม Attribute ที่ต้องการเข้าไปทีหลังได้

5.1.2 การวิเคราะห์และออกแบบ

ในการวิเคราะห์และออกแบบ ได้ทำการออกแบบเพื่อเก็บข้อมูลของผู้ใช้และเก็บข้อมูลที่จำเป็นในการพัฒนาระบบงานเพื่อระบบงานที่มีประสิทธิภาพจึงมองถึงการรองรับสิ่งที่จะเพิ่มเข้ามาในอนาคตได้ เช่น หากต้องการเพิ่มบริการอื่น ๆ ลงในระบบก็สามารถที่จะจัดการกับข้อมูลผู้ใช้ได้จากที่เดียวทำให้รวดเร็วโดยอาจจะเพิ่ม Attribute อื่น ๆ หรือเพิ่ม Schema ของบริการอื่นลงไป หรือปรับเปลี่ยน โครงสร้างของ Tree ในระบบ

5.1.3 การพัฒนาระบบงาน

ได้ทำการออกแบบการทำงานผ่าน Browser ทำให้เพิ่มความสะดวกในการใช้งานเนื่องจากสามารถใช้งานโปรแกรมได้จากหลาย ๆ ที่และทุก Plat form และเนื่องจากความยืดหยุ่นของ LDAP Server จากที่กล่าวข้างต้นทำให้การพัฒนาเป็นไปได้โดยง่าย

5.2 ผลจากการทดสอบโปรแกรม

จากการทดสอบการใช้งานโปรแกรมโดยได้ทำการใช้งานตามความสามารถหลักที่โปรแกรมสามารถทำงาน ได้ผลลัพธ์ดังนี้

- 5.2.1 สามารถที่จะเพิ่มผู้ใช้งานเข้าไปในระบบได้อย่างรวดเร็วและการทำงานมีการตรวจสอบการเพิ่มให้เป็นไปอย่างถูกต้องตามที่ออกแบบไว้
- 5.2.2 สามารถที่จะทำการ Authentication ผู้ใช้งานได้อย่างถูกต้องและรวดเร็วและรู้ถึงสถานะของผู้ใช้งานได้ตอนเริ่ม Authentication ว่ามีใคร Add ผู้ใช้หรือไม่ทำให้เป็นการเพิ่มความปลอดภัยได้เป็นอย่างดีเนื่องจากสามารถเลือกที่จะตกลงสนทนาหรือไม่ก็ได้ และสามารถป้องกันการ Sign In ซ้ำซ้อนของผู้ใช้ได้
- 5.2.3 สามารถที่จะทำการสืบค้นข้อมูลผู้ใช้งานเพื่อทำงานตาม Function Add Contact หรือ Delete Contact ได้เป็นอย่างดี คัดยที่ระบบจะตรวจสอบการมีอยู่จริงของผู้ใช้งานด้วย
- 5.2.4 สามารถที่จะแสดงสถานะของผู้ใช้งานว่า Online หรือ Offline ได้
- 5.2.5 สามารถที่จะสนทนากับผู้ที่ Online อยู่แบบ real time ได้และข้อความส่งได้ทั้งภาษาไทยและภาษาอังกฤษ

5.3 อุปสรรคในการพัฒนาโปรแกรม

- 5.3.1 การพัฒนาโปรแกรมมีการนำโปรแกรมหลายชนิดมาใช้งานร่วมกันจึงเกิดความยุ่งยากในการติดตั้งให้โปรแกรมต่าง ๆ สามารถทำงานร่วมกันได้ และใช้เวลาในการศึกษาแต่ละโปรแกรมเนื่องจากทุกโปรแกรมเป็นสิ่งที่ไม่เคยเรียนรู้มาก่อน
- 5.3.2 ในการพัฒนาระบบเนื่องจาก LDAP Sever ที่เลือกใช้ สามารถทำงานได้ดีบนระบบปฏิบัติการ Linux จึงต้องมีการจัดเตรียมเครื่องมือเพื่อเป็น LDAP Server โดยเฉพาะ อีกทั้งความไม่คุ้นเคยกับระบบปฏิบัติการ Linux มากนัก จึงทำให้ใช้เวลามากในการศึกษา
- 5.3.3 เนื่องจากระบบ Directory ยังเป็นเทคโนโลยีที่ค่อนข้างใหม่ทำให้เอกสารอ้างอิงยังมีไม่มากนักและตัวอย่างการเขียนโปรแกรมติดต่อผ่าน API เพื่อใช้งาน Directory ยังมีน้อยทำให้ต้องใช้เวลาในการศึกษามากกว่าที่ควร

5.4 ข้อเสนอแนะ

- 5.4.1 ในการติดตั้งโปรแกรมหรือทำการแก้ไขค่า Configuration ต่าง ๆ ของโปรแกรม ควรมีการสำรองข้อมูลเอาไว้ก่อนเพื่อป้องกันความผิดพลาดที่อาจจะเกิดขึ้น
- 5.4.2 กรณีต้องเปลี่ยนค่า Configuration ของ Directory ควรทำการแก้ไขในช่วงที่ไม่มีการทำงาน
- 5.4.3 ในการติดตั้งโปรแกรมที่ใช้ในการทำงานต่าง ๆ ควรทำการจดค่าที่ใช้ในการติดตั้งทุกครั้งเพื่อใช้ในการเตือนความจำ ถ้าต้องมาติดตั้งใหม่ภายหลัง



บรรณานุกรม

ชัยนันท์ กมลวดี. 2545. เพิ่มพลังเครือข่ายเต็มพิกัด ด้วย Directory Services. กรุงเทพฯ : เอส.พี.ซี บู้คส์.

Archie Reed.2000. **Implementing Directory Services**. U.S.A. McGraw-Hill.

DIRECTORY SERVICES – The role of LDAP and X.500.[Online].Available:

<http://www.dataconnection.com/press/mdwhite1.htm>

Directories Explained. [Online]. Available : <http://www.dataconnection.com/dirs/direxpl.htm>

PHP Manual. [Online]. Available : <http://php.benscom.com/manual/en/index.php>

Timothy A. Hower, Mark C. Smith, and Gordon S. Good.1998.**Understanding and Deploying**

LDAP Directory Services. U.S.A. Macmillan Network Architecture & Development.





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก. การติดตั้ง Program ที่ต้องใช้ในระบบ

ขั้นตอนการ Compile และการติดตั้งโปรแกรม Berkley DB

1. export CC=gcc
2. cd /usr/local/db-4.1.25/build_unix
3. ../dist/configure --prefix=/usr/local/BerkleyDB
4. make
5. make install

ขั้นตอนการ Compile และติดตั้งโปรแกรม Openldap

1. export CC=gcc
2. export CPPFLAGS="-I/usr/local/BerkleyDB/include"
3. export LDFLAGS="-L/usr/local/BerkleyDB/lib"
4. cd /usr/local/openldap-2.1.17
5. ./configure --prefix=/usr/local/openldap
6. make depend
7. make
8. make install

ขั้นตอนการติดตั้งโปรแกรม APACHE

1. /usr/local/apache_1.3.27
2. ./configure --prefix=/usr/local/apache --enable-module=so
3. make
4. make install

ขั้นตอนการติดตั้งโปรแกรม PHP

1. `./configure --prefix=/usr/local/php_4.3.2 --with-apxs=/usr/local/apache/bin/apxs --with-ldap=/usr/local/openldap`
2. `make`
3. `make install`

การแก้ไขค่าของ Apache ซึ่งเป็น Web Server

แก้ไขไฟล์ `httpd.conf` มีการแก้ไขดังนี้

1. `DocumentRoot "/usr/local/web"`
2. `<Directory "/usr/local/web">`
`Option Indexes FollowSymLinks Multiviews`
`AllowOverride None`
`Order allow,deny`
`Allow from all`
`</Directory>`
3. `<IfModule mod_dir.c>`
`DirectoryIndex index.html เปลี่ยนเป็น DirectoryIndex index.php index.html`
`</IfModule>`
4. `AddType application/x-httpd-php .php`

การแก้ไขค่าของ Software Openldap ซึ่งเป็น Directory Server

ทำการกำหนดค่าต่าง ๆ ในไฟล์ `/usr/local/openldap/etc/openldap/slapd.conf` โดยทำการเพิ่มค่าเหล่านี้เข้าไป

1. ทำการ `include` schema ที่มาเก็บ Objectclass และ Attribute
`include /usr/local/openldap/etc/openldap/schema/core.schema`
`include /usr/local/openldap/etc/openldap/schema/inetorgperson.schema`
`include /usr/local/openldap/etc/openldap/schema/nis.schema`
2. `pidfile` เป็นที่บอกตำแหน่งของไฟล์ `pid`
`pidfile /usr/local/openldap/var/slapd.pid`

3. argfile ที่เก็บอาร์กิวเมนต์ของ slapd
`argsfile /usr/local/openldap/var/slapd.args`
4. database ชนิดของฐานข้อมูลไคลเรททอรีถ้าเป็นฐานข้อมูลจาก `www.sleepycat.com` ก็จะเป็น `bdb` ถ้าเป็นฐานข้อมูลจาก `www.gnu.com` จะเป็น `gdbm`
`database bdb`
5. suffix BaseDN ของไคลเรททอรี
`suffix "dc=mydomain,dc=com"`
6. rootdn DN หรือชื่อของผู้มีสิทธิ์ในการบริหารไคลเรททอรีสูงสุด
`rootdn "cn=Manager,dc=mydomain,dc=com"`
7. rootpw password ของ rootdn ที่ใช้ในการบริหารไคลเรททอรี การตั้ง password เราสามารถที่จะใส่เป็นการเข้ารหัสโดยใช้ `ldappassword` ที่อยู่ใน path `/usr/local/openldap/bin`
`rootpw secret`



ภาคผนวก ข.

Configuration ของไฟล์ slapd.conf

หลังจากที่ทำการติดตั้ง Software Openldap-2.1.17 เรียบร้อยแล้ว เราจะต้อง กำหนดค่าเริ่มต้นต่าง ๆ ของ Directory Server ไฟล์ที่เก็บค่าเริ่มต้นต่าง ๆ จะอยู่ภายใต้ path /usr/local/openldap/etc/openldap ไฟล์ที่เราจะทำการแก้ไขคือ ไฟล์ slapd.conf

```
# $OpenLDAP: pkg/ldap/servers/slapd/slapd.conf,v 1.23.2.7 2003/03/24 03:54:12 kurt Exp $
#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include      /usr/local/openldap/etc/openldap/schema/core.schema
include      /usr/local/openldap/etc/openldap/schema/inetorgperson.schema
include      /usr/local/openldap/etc/openldap/schema/nis.schema
# Define global ACLs to disable default read access.
# Do not enable referrals until AFTER you have a working directory
# service AND an understanding of referrals.
#referral    ldap://root.openldap.org
pidfile      /usr/local/openldap/var/slapd.pid
argsfile     /usr/local/openldap/var/slapd.args
# Load dynamic backend modules:
# modulepath /usr/local/openldap/libexec/openldap
# moduleload back_bdb.la
# moduleload back_ldap.la
# moduleload back_ldbm.la
# moduleload back_passwd.la
# moduleload back_shell.la
# Sample security restrictions
```

```

# Disallow clear text exchange of passwords
# disallow bind_simple_unprotected
#
# Require integrity protection (prevent hijacking)
# Require 112-bit (3DES or better) encryption for updates
# Require 63-bit encryption for simple bind
# security ssf=1 update_ssf=112 simple_bind=64

# Sample access control policy:
# Root DSE: allow anyone to read it
# Subschema (sub)entry DSE: allow anyone to read it
# Other DSEs:
# Allow self write access
# Allow authenticated users read access
# Allow anonymous users to authenticate
# Directives needed to implement policy:
# access to dn.base="" by * read
# access to dn.base="cn=Subschema" by * read
# access to *
# by self write
# by users read
# by anonymous auth
#
# if no access controls are present, the default policy is:
# Allow read by all
# rootdn can always write!

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#####
# ldbm database definitions
#####

database      bdb
suffix        "dc=mydomain,dc=com"
rootdn        "cn=Manager,dc=mydomain,dc=com"
# Cleartext passwords, especially for the rootdn, should
# be avoid. See slapd.conf(5) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw        secret
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.
# Mode 700 recommended.
directory     /usr/local/openldap/var/openldap-data
# Indices to maintain
xindex        objectClass      eq

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

Source ของโปรแกรมที่ใช้ในการ Start/Stop Openldap

```
LD_LIBRARY_PATH=/usr/local/BerkeleyDB/lib
```

```
export LD_LIBRARY_PATH
```

```
case "$1" in
```

```
'start')
```

```
    echo "Starting Ldap services: " &
```

```
    /usr/local/openldap/libexec/slapd
```

```
    sleep 1s
```

```
    cat /usr/local/openldap/var/slapd.pid
```

```
    ;;
```

```
'stop')
```

```
    echo "Stop Ldap service: " &
```

```
    sleep 1s
```

```
    kill -INT `cat /usr/local/openldap/var/slapd.pid`
```

```
    ;;
```

```
*)
```

```
    echo "Usage:$0 { start|stop }"
```

```
    exit 1
```

```
    ;;
```

```
esac
```

```
exit 0
```

ภาคผนวก ง.

ตัวอย่าง Configuration File ของโปรแกรม Apache

```

ServerType standalone
ServerRoot "/usr/local/apache"
PidFile /usr/local/apache/logs/httpd.pid
ScoreBoardFile /usr/local/apache/logs/httpd.scoreboard
Timeout 300
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
MinSpareServers 5
MaxSpareServers 10
StartServers 5
MaxClients 150
MaxRequestsPerChild 0
LoadModule php4_module      libexec/libphp4.so
Port 80
User nobody
Group nobody
ServerAdmin root@localhost.localdomain
ServerName localhost
DocumentRoot "/usr/local/web"

```

```
<Directory "/usr/local/web">
```

```
Options FollowSymLinks
```

```
AllowOverride None
```

```
</Directory>
```

```
<Directory "/usr/local/web">
```

```
Options Indexes FollowSymLinks MultiViews
```

```

AllowOverride None
Order allow,deny
Allow from all
</Directory>

```

```

<IfModule mod_userdir.c>

```

```

    UserDir public_html

```

```

</IfModule>

```

```

<IfModule mod_dir.c>

```

```

    DirectoryIndex index.php index.html

```

```

</IfModule>

```

```

AddType application/x-tar .tgz

```

```

AddType image/x-icon .ico

```

```

AddType application/x-httpd-php .php .php3 .php4 .html

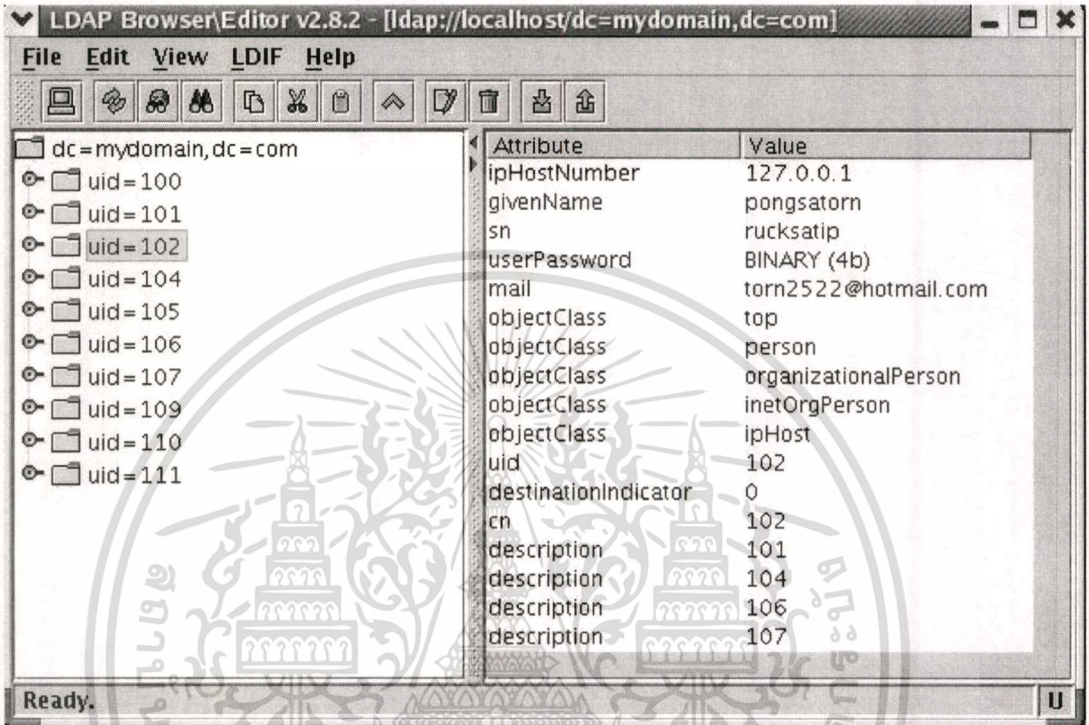
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ.

ตัวอย่าง Tree ในระบบเมื่อดูจาก LDAP Browser



รูปที่ จ-1 แสดงรูป LDAP Browser

ภาคผนวก ฉ.

ตัวอย่างไฟล์ LDIF ที่ Export จาก LDAP

dn: dc=mydomain,dc=com

dc: mydomain

objectClass: top

objectClass: domain

dn: uid=102, dc=mydomain,dc=com

ipHostNumber: 127.0.0.1

givenName: pongsatorn

sn: rucksatip

userPassword:: dG9ybg==

mail: torn2522@hotmail.com

objectClass: top

objectClass: person

objectClass: organizationalPerson

objectClass: inetOrgPerson

objectClass: ipHost

uid: 102

destinationIndicator: 0

cn: 102

description: 101

description: 104

description: 107

ประวัติผู้เขียนโครงการ

ชื่อผู้จัดทำโครงการ	นาย พนิตนาถ รักษาทิพย์
วันเดือนปีเกิด	24 มกราคม 2521
สถานที่เกิด	จ.กรุงเทพฯ
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนเซนต์โยเซฟ บางนา จ.สมุทรปราการ
มัธยมศึกษาตอนต้น	โรงเรียนเซนต์โยเซฟ บางนา จ.สมุทรปราการ
มัธยมศึกษาตอนปลาย	โรงเรียนเตรียมอุดมศึกษาพัฒนาการ จ.กรุงเทพฯ
ปริญญาตรี	มหาวิทยาลัยกรุงเทพ จ.กรุงเทพฯ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้