

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครอง

PARENTAL CONTROL SYSTEM

ธัญญา บุรณพันธ์พงศ์

สมเดช สิงห์โต

2พ.
ค 2112
2550

เลขหมู่.....
เลขทะเบียน..... **83017**
วัน,เดือน,ปี..... **30 ก.ค. 2551**

b. **119 5951**
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

ปริญญาานิพนธ์ปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครอง

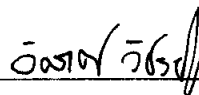
PARENTAL CONTROL SYSTEM

ผู้จัดทำ

1. นางสาวธัญญา บูรณพันธ์พงศ์ รหัสนักศึกษา 47010313
2. นายสมเดช สิงห์โต รหัสนักศึกษา 47010802



อาจารย์ที่ปรึกษา
(อาจารย์ธัญชัย ตรีภาค)



อาจารย์ที่ปรึกษา
(อาจารย์อัครเดช วัชรภูพงษ์)



อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ธนา หงษ์สุวรรณ)

โปรแกรมควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครอง

นางสาวธัญญา นูรณพันธ์พงศ์	47010313
นายสมเดช สิงห์โต	47010802
อาจารย์ธัญชัย ศรีภาค	อาจารย์ที่ปรึกษา
อาจารย์อัครเดช วัชรระภูงษ์	อาจารย์ที่ปรึกษาร่วม
ผศ. ธนา หงษ์สุวรรณ	อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2550	

บทคัดย่อ

เนื่องจากความก้าวหน้าด้านเทคโนโลยีและสื่อสารในปัจจุบันทำให้สามารถใช้คอมพิวเตอร์ในการสื่อสารในบุคคลทั่วไปได้อย่างกว้างขวางและง่ายคดียขึ้น การเข้าสู่โลกอินเทอร์เน็ตจึงไม่ใช่เรื่องยากอีกต่อไป บุคคลทั่วไปก็สามารถเข้าถึงอินเทอร์เน็ตได้ รวมไปถึงเยาวชนส่วนใหญ่ที่ได้ใช้อินเทอร์เน็ตเป็นส่วนหนึ่งของชีวิตประจำวัน ไม่ว่าจะเป็นด้านการศึกษาหาความรู้ ความบันเทิง การติดต่อสื่อสาร การซื้อขายของบนอินเทอร์เน็ต หรือรวมไปถึงการหารายได้ต่างๆทางอินเทอร์เน็ต จึงนับได้ว่า อินเทอร์เน็ตนั้นได้มีบทบาทกับเยาวชนไทยในโลกยุคปัจจุบันอย่างมาก ดังนั้นการควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครองนั้น จึงเป็นทางเลือกหนึ่งสำหรับผู้ปกครองที่ต้องการควบคุมพฤติกรรมทางอินเทอร์เน็ตของเยาวชน ซึ่งสามารถควบคุม ฝ้าคุ และ บันที่กพฤติกรรมได้ภายในโปรแกรมเดียว และยังมีกำส่งเสริมสิ่งที่เป็นประโยชน์แก่เยาวชนไทยเพื่อได้เข้าไปศึกษา รวมถึงปิดกั้นเว็บไซต์ที่มีความเสี่ยง เพื่อให้เยาวชนไทยได้ใช้อินเทอร์เน็ตให้ถูกต้อง มีประสิทธิภาพ และเกิดประโยชน์มากขึ้น

Parental Control System

Ms. Thananya Booranaphanpong	47010313
Mr. Somdech Singto	47010802
Mr. Thananchai Treepak	Advisor
Mr. Akkradach Watcharapupong	Co-Advisor
Asst.Prof. Thanna Hongsuwan	Co-Advisor
Academic Year 2007	

ABSTRACT

An advancement of communication and technology may benefit the world in many ways but it somehow cause a major problem, especially for parents. Surfing internet isn't difficult anymore, vice versa, most applicable for all purposes. The youth uses the internet every day to fulfill their education, communication, e-commerce or earning a living, and most common for entertainment. The role of internet therefore becomes more and more important for Thai youth nowadays. Parental control system, for this reason is alternatively an interest choice for parents who see it is important to control children's behavior in using of the internet. By using it to control and observe, save and report completely all-in-one. Not only parental control system can encourage Thai youth in using of internet in the most useful ways, but it also blocks vulnerable websites. Consequently Thai internet surfers can use the internet more properly and shall be in safe from various.

สารบัญ

	หน้า
บทคัดย่อไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
สารบัญ.....	III
สารบัญตาราง.....	IV
สารบัญรูป.....	VI
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตของโครงการ.....	2
1.4 วิธีการดำเนินการ.....	3
1.5 ประโยชน์ที่ได้รับ.....	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	5
2.1 ฟร็อกซ์เซิร์ฟเวอร์.....	5
2.2 อัลกอริทึมพื้นฐานที่ใช้ในโปรแกรม.....	6
2.2.1 สถิติ.....	6
2.2.2 เมตริกซ์อัลจิบรา.....	8
2.2.3 หลักการวิเคราะห์แยกแยะส่วนประกอบ.....	10
2.2.4 K-Means Clustering and K-nearest neighbor algorithm.....	11
2.2.5 K-Nearest Neighbor.....	13
2.3 การคัดกรองข้อความ.....	13
2.4 การคัดกรองรูปภาพ.....	14
2.4.1 พื้นฐานและระบบโครงสร้างสี.....	14
2.4.1.1 ระบบโครงสร้างสีอาร์จีบี.....	14
2.4.1.2 ระบบโครงสร้างสีวายซีบีซีอาร์.....	15
2.4.1.3 การแปลงรูปแบบสี.....	15
2.4.2 กระบวนการค้นหาพื้นที่สีผิว.....	16
2.4.3 กระบวนการปรับความคมชัดของภาพ.....	21
2.4.3.1 First derivative.....	21
2.4.3.2 Second derivative.....	21

สารบัญ(ต่อ)

	หน้า
2.4.3.3 การใช้ First Derivative ในการเพิ่มคุณภาพของภาพ	20
2.4.3.4 การใช้ Second Derivative ในการเพิ่มคุณภาพของภาพ	23
2.4.3.5 การตรวจหาเส้นขอบด้วย กระบวนการ โซเบล	24
2.4.4 กระบวนการทดสอบ	25
2.4.4.1 การกำจัดภาพขนาดเล็กกว่ากำหนด	26
2.4.4.2 การตรวจหาสีผิวมนุษย์	26
2.4.4.3 การตรวจสอบคุณสมบัติของภาพ	27
2.4.4.4 การจัดประเภทของภาพด้วยคุณสมบัติ	30
2.5 Packet Filtering API	32
2.5.1 คลาสใน .NET	35
2.6 IPHelper Library	36
2.7 Asynchronous Programming	41
บทที่ 3 โครงสร้างและการออกแบบ	42
3.1 โครงสร้างพื้นฐานของระบบควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครอง	44
3.1.1 โปรแกรมรวบรวมและตรวจสอบเอกสาร	44
3.1.1.1 ส่วนของการรวบรวมเอกสารหน้าเว็บไซต์	44
3.1.1.2 ระบบวิเคราะห์ความเหมาะสมของเอกสารเว็บไซต์	45
3.1.2 โปรแกรมส่วนควบคุมการใช้งานอินเทอร์เน็ต	46
3.1.2.1 ระบบคัดกรองเว็บไซต์ที่ไม่เหมาะสม	47
3.1.2.2 ระบบคัดกรองข้อมูลเชื่อมต่ออินเทอร์เน็ต	48
3.1.2.3 ระบบจัดเวลาการใช้งานอินเทอร์เน็ต	48
3.2 โครงสร้างการเชื่อมต่อของระบบและการอัปเดตข้อมูล	48
3.2.1 โครงสร้างการเชื่อมต่อ	48
3.2.2 การอัปเดตฐานข้อมูล	50
3.3.3 โครงสร้างการเชื่อมต่อระหว่างแอปพลิเคชันและเว็บไซต์ควบคุม	50
บทที่ 4 หลักการพื้นฐานบนเครือข่าย	52
4.1 โพรโตคอลเอชทีทีพี	52
4.1.1 วิธีการติดต่อของโพรโตคอลเอชทีทีพี	52
4.1.2 โครงสร้างของโพรโตคอลเอชทีทีพี	53

สารบัญ(ต่อ)

	หน้า
4.2 ฟร็อกซี่ส่วนตัว.....	55
4.2.1 ฟร็อกซี่ คืออะไร.....	55
4.2.2 ฟร็อกซี่ส่วนตัวคืออะไร.....	56
4.3 โฟโตคอลลที่ใช้ในการสื่อสารระหว่างเซิร์ฟเวอร์และไคลเอ็นต์.....	56
4.3.1 โครงสร้างของโฟโตคอลล.....	57
บทที่ 5 เทคนิคที่ใช้ในการเขียน โปรแกรม.....	58
5.1 การเขียนโปรแกรมแบบไม่ประสานเวลา.....	58
5.1.1 Asynchronous Client Application.....	60
5.1.2 Asynchronous Server Application.....	62
5.2 การสร้างและการติดตั้งวินโดวส์เซอร์วิสแอปพลิเคชัน โดย Visual Studio.NET.....	64
5.2.1 เขียนโปรแกรมวินโดวส์เซอร์วิส.....	64
5.2.2 การเพิ่มคลาส Installer ในโปรแกรม.....	64
5.2.3 การใช้เครื่องมือรันเพื่อช่วยในการติดตั้งวินโดวส์เซอร์วิส.....	64
5.2.4 การรวมส่วนประกอบของตัวจับเวลากับวินโดวส์เซอร์วิส.....	65
5.2.5 การทดลองวินโดวส์เซอร์วิสเพื่อป้องกันการปิดของแอปพลิเคชัน.....	66
5.3 นิพจน์ปรกติ.....	71
5.3.1 หลักการพื้นฐาน.....	71
5.3.2 สัญลักษณ์ที่ใช้ในนิพจน์ปรกติ.....	72
5.3.3 ตัวอย่างการใช้งาน.....	73
5.4 การปรับแต่งค่าในรีจิสตรี.....	73
5.4.1 หลักการที่เกี่ยวข้อง.....	73
5.4.2 การเลือก Subkey.....	74
5.4.3 การเปิดค่า Subkey ที่มีอยู่แล้ว.....	75
5.4.4 การเรียกค่าที่เก็บอยู่มาใช้.....	76
5.4.5 การปรับแต่งค่าของข้อมูล.....	76
5.4.6 การปิด Subkey.....	76
5.4.7 การลบ Subkey.....	77
5.4.8 การลบ Entire tree.....	77
5.4.9 การลบ Value.....	77

สารบัญ(ต่อ)

	หน้า
5.4.10 การรับค่า List of subkeys	77
5.4.11 การรับค่า List of value names	77
5.5 การเขียน โปรแกรมเพื่อใช้งานการเข้ารหัสข้อมูลแบบ RSA	78
5.5.1 การสร้างคลาส RSACryptoServiceProvider	78
5.5.2 การสร้างกุญแจสาธารณะและกุญแจส่วนตัว	78
5.5.3 การเข้ารหัสข้อมูลด้วยวิธี RSA	79
5.5.4 การถอดรหัสข้อมูลด้วย RSA	80
5.6 การทดลองสร้างระบบป้องกันการเชื่อมต่อของแอปพลิเคชัน	81
5.6.1 การเขียน โปรแกรมเพื่อดึงข้อมูลการเชื่อมต่อ	81
5.6.2 การเขียน โปรแกรมในส่วนของกำกั้นการเชื่อมต่อ	82
5.7 การสร้างระบบตรวจสอบความเหมาะสมของเว็บไซต์	84
5.7.1 การใช้งานคลาสดตรวจสอบความเหมาะสมของเว็บไซต์	84
5.7.2 การใช้งานคลาสดตรวจสอบความเหมาะสมของรูปภาพ	85
บทที่ 6 การทดลองและผลการทดลอง	86
6.1 ทดสอบประสิทธิภาพของระบบรวบรวมและวิเคราะห์เอกสารเว็บไซต์	86
6.2 ทดสอบความถูกต้องของระบบวิเคราะห์เว็บไซต์	89
6.2.1 ทดสอบกับเว็บไซต์ปกติ	89
6.2.2 ทดสอบกับเว็บไซต์อันตราย	91
บทที่ 7 บทวิจารณ์และสรุป	95
7.1 บทสรุป	95
7.2 ปัญหาอุปสรรคและแนวทางในการแก้ไข	95
7.3 แนวทางการพัฒนาต่อ	96
บรรณานุกรม	97
ภาคผนวก	98

สารบัญตาราง

ตารางที่	หน้า
2.1 คุณสมบัติ 6 ประการที่ได้จากภาพ	27
4.1 แสดงความหมายของค่า Type	57
4.2 แสดงความหมายของค่า Code	57
5.1 รายละเอียดของ subkey root	75
6.1 แสดงข้อมูลของระบบเมื่อใช้งานเฉพาะส่วนรวบรวมเอกสาร	86
6.2 แสดงข้อมูลของระบบเมื่อใช้งานเฉพาะส่วนวิเคราะห์เอกสาร	87
6.3 แสดงข้อมูลของระบบเมื่อใช้งานทั้ง 2 ส่วน	88
6.5 แสดงผลการทดสอบกับเว็บไซต์ปกติ	89
6.6 แสดงผลการทดสอบกับเว็บไซต์ปกติ	89
6.7 แสดงผลการทดสอบกับเว็บไซต์ปกติที่ผิดพลาด	91
6.8 แสดงผลการทดสอบกับเว็บไซต์อันตราย	92
6.9 แสดงผลการทดสอบกับเว็บไซต์อันตราย	92
6.10 แสดงผลการทดสอบกับเว็บไซต์อันตรายที่ผิดพลาด	94

สารบัญภาพ

รูปที่	หน้า
รูปที่ 2.1 กลไกการทำงานของ K-means	13
รูปที่ 2.2 โครงสร้างสีอาร์จีบี เป็นลูกบาศก์	15
รูปที่ 2.3 ความสัมพันธ์ระหว่างโครงสร้างสี YCbCr กับ RGB	16
รูปที่ 2.4 กลุ่มตัวอย่างสีผิวในแกนของโครงสร้างสี YCbCr	17
รูปที่ 2.5 กลุ่มตัวอย่างสีผิวในแกนของโครงสร้างสี HSV	18
รูปที่ 2.6 กรอบพื้นที่สีผิวโครงสร้างสี YCbCr ที่ระนาบ $Y = 160$	19
รูปที่ 2.7 กรอบพื้นที่สีผิวโครงสร้างสี HSV ที่ระนาบ $V = 70$	20
รูปที่ 2.8 (a) จุดของภาพที่มีขอบเขต 3×3 pixel (ค่า z เป็นค่าระดับสีเทา)	24
รูปที่ 2.8 (b)(c) Roberts cross-gradient operators	24
รูปที่ 2.8 (d)(e) Sobel operators	24
รูปที่ 2.9 (a) ตัวอย่างภาพต้นฉบับที่ต้องการหาเส้นขอบ	25
รูปที่ 2.9 (b) ภาพที่ถูกตรวจหาเส้นขอบ (Sobel gradient)	25
รูปที่ 2.10 การทำงานของการตรวจสอบภาพอนาจาร	26
รูปที่ 2.11 การทำงานของโปรแกรมการตรวจสอบภาพอนาจาร	29
รูปที่ 2.14 การทำงานของการคำนวณหาไอแกนเวกเตอร์และไอแกนแวลูส์ของชุดข้อมูลตัวอย่างและ การหาจุดเซ็นทรอยด์ 3 จุด	30
รูปที่ 2.13 การจำแนกภาพ	31
รูปที่ 2.14 Network Architecture Diagram (user mode)	32
รูปที่ 2.15 Network Architecture Diagram (kernel mode)	33
รูปที่ 3.1 โครงสร้างและสถาปัตยกรรมของโปรแกรม Parental Control System	43
รูปที่ 3.2 โครงสร้างและสถาปัตยกรรมของส่วน Crawler	44
รูปที่ 3.3 ตัวอย่างการเก็บไฟล์ตามโครงสร้างของเว็บไซต์	45
รูปที่ 3.4 โครงสร้างของระบบวิเคราะห์ความเหมาะสมของเอกสารเว็บไซต์	46
รูปที่ 3.5 โครงสร้างของระบบคัดกรองเว็บไซต์	47
รูปที่ 3.6 แสดงหน้าเว็บไซต์เมื่อเข้าเว็บที่มีอยู่ในข้อมูลเว็บไซต์ไม่เหมาะสม	47
รูปที่ 3.7 แสดงโครงสร้างของระบบคัดกรองข้อมูลเชื่อมต่ออินเทอร์เน็ต	48
รูปที่ 3.8 แสดงขั้นตอนการเชื่อมต่อสู่ระบบ	49
รูปที่ 3.9 แสดงคำรหัสประจำตัว	49
รูปที่ 3.10 แสดงขั้นตอนการอัปเดตข้อมูล	50

สารบัญญภาพ(ต่อ)

รูปที่	หน้า
รูปที่ 3.11 โครงสร้างการเชื่อมต่อระหว่างแอปพลิเคชันและเว็บไซต์ควบคุม	50
รูปที่ 4.1 โครงสร้างของข้อมูลที่ส่งผ่านโพรโทคอล HTTP	54
รูปที่ 4.2 ตัวอย่างข้อความร้องขอด้วยเมธอด GET	54
รูปที่ 4.3 ตัวอย่างข้อความร้องขอด้วยเมธอด POST	55
รูปที่ 4.4 รูปแบบของ Personal Web Proxy	56
รูปที่ 4.5 แสดงโครงสร้างของข้อมูลที่ใช้ในการสื่อสาร	57
รูปที่ 5.1 แผนภาพขั้นตอนการทำงานของไคลเอ็นต์แอปพลิเคชัน	60
รูปที่ 5.2 แผนภาพขั้นตอนการทำงานของเซิร์ฟเวอร์แอปพลิเคชัน	62
รูปที่ 5.3 ตัวอย่างส่วนของโค้ดของเมธอด เมื่อได้รับเหตุการณ์ครบรอบ interval ของตัวจับเวลา	65
รูปที่ 5.4 แสดงการสร้างตัวควบคุมเวลา	66
รูปที่ 5.5 แสดงการตรวจสอบการทำงานของโปรเซสและการสร้างโปรเซสขึ้นมาทำงาน	67
รูปที่ 5.6 Command Prompt เมื่อเข้ามาที่ Visual Studio 2005 Command Prompt	67
รูปที่ 5.7 การใช้คำสั่งในการติดตั้งโปรแกรมวินโดวเซอร์วิส	68
รูปที่ 5.8 ผลลัพธ์เมื่อทำการติดตั้งโปรแกรมวินโดวเซอร์วิสสำเร็จ	68
รูปที่ 5.9 วินโดวเซอร์วิสที่ถูกติดตั้งเรียบร้อยแล้ว	69
รูปที่ 5.10 แสดงการตั้งค่า Allow service to interact with desktop	69
รูปที่ 5.11 ปุ่ม Start ใน Properties เพื่อเรียกให้วินโดวเซอร์วิสตัวนี้รัน	70
รูปที่ 5.12 Registry Editor	74
รูปที่ 5.13 แสดงโค้ดการสร้าง RSACryptoServiceProvider	78
รูปที่ 5.14 แสดงโค้ดการสร้างกุญแจสาธารณะ(Public key) และกุญแจส่วนตัว (Private key)	78
รูปที่ 5.15 แสดงตัวอย่างของกุญแจส่วนตัว (Private key)	79
รูปที่ 5.16 แสดงตัวอย่างของกุญแจสาธารณะ(Public key)	79
รูปที่ 5.17 แสดงตัวอย่างการเข้ารหัสข้อมูลด้วยกุญแจสาธารณะ(Public key)	79
รูปที่ 5.18 แสดงตัวอย่างคำว่า “Test RSA Encryption” เมื่อถูกเข้ารหัส	80
รูปที่ 5.19 แสดงตัวอย่างการถอดรหัสข้อมูลด้วย RSA	80
รูปที่ 5.20 แสดงตัวอย่างคำว่า “Test RSA Encryption” เมื่อถูกถอดรหัส	81
รูปที่ 5.21 โค้ดการเขียนโปรแกรมโดยใช้ GetExTcpConnexions()	82
รูปที่ 5.22 ตัวอย่างโปรแกรมที่ถูกเขียนด้วย GetExTcpConnexions()	82
รูปที่ 5.23 ตัวอย่างโค้ดที่ใช้ Packet Filtering API เขียน	83

สารบัญญภาพ(ต่อ)

รูปที่	หน้า
รูปที่ 5.24 ตัวอย่างโปรแกรมที่ใช้ Packet Filtering API เขียน	84
รูปที่ 5.25 แสดงตัวอย่างการใช้งานคลาสวิเคราะห์ความเหมาะสมของเว็บไซต์	84
รูปที่ 5.26 แสดงตัวอย่างการใช้งานคลาสวิเคราะห์ความเหมาะสมของรูปภาพ	85
รูปที่ 6.1 กราฟแสดงข้อมูลค่าการใช้ CPU และ Memory เมื่อใช้งานเฉพาะส่วนรวบรวมเอกสาร	86
รูปที่ 6.2 กราฟแสดงข้อมูลค่าการใช้ CPU และ Memory เมื่อใช้งานเฉพาะส่วนวิเคราะห์เอกสาร	87
รูปที่ 6.4 กราฟแสดงข้อมูลค่าการใช้ CPU และ Memory เมื่อใช้งานทั้ง 2 ส่วน	88

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

เนื่องจากความก้าวหน้าด้านเทคโนโลยีและสื่อสารในปัจจุบันทำให้สามารถใช้คอมพิวเตอร์ในการสื่อสารในบุคคลทั่วไปได้อย่างกว้างขวางและง่ายดายขึ้น การเข้าสู่โลกอินเทอร์เน็ตจึงไม่ใช่เรื่องยากอีกต่อไป บุคคลทั่วไปก็สามารถเข้าถึงอินเทอร์เน็ตได้ รวมไปถึงเยาวชนส่วนใหญ่ที่ได้ใช้อินเทอร์เน็ตเป็นส่วนหนึ่งของชีวิตประจำวันเลยทีเดียว ไม่ว่าจะเป็นด้านการศึกษาหาความรู้ ความบันเทิง การติดต่อสื่อสาร การซื้อขายของบนอินเทอร์เน็ต หรือรวมไปถึงการหารายได้ทางอินเทอร์เน็ต จึงนับได้ว่า อินเทอร์เน็ตนั้น ได้มีบทบาทกับเยาวชนไทยในโลกยุคปัจจุบันอย่างมาก ดังนั้นการควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครองนั้น จึงเป็นทางเลือกหนึ่งสำหรับผู้ปกครองที่ต้องการควบคุมพฤติกรรมทางอินเทอร์เน็ตของเยาวชน ซึ่งสามารถควบคุม ฝ้าคุ และ บันทึกพฤติกรรมได้ภายใน โปรแกรมเดียว และยังมี การส่งเสริมสิ่งที่เป็นประโยชน์แก่เยาวชนไทยเพื่อได้เข้าไปศึกษา รวมถึงปิดกั้นเว็บไซต์ที่มีความเสี่ยง เพื่อให้เยาวชนไทยได้ใช้อินเทอร์เน็ตให้ถูกต้อง มีประสิทธิภาพ และเกิดประโยชน์มากขึ้น

โครงการ โปรแกรมควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครองนี้ เป็นโปรแกรมสำหรับ ผู้ปกครองสำหรับควบคุมและบันทึกพฤติกรรมทางอินเทอร์เน็ตของเยาวชน มีการส่งเสริมการเรียนรู้โดยแนะนำเว็บไซต์ที่มีประโยชน์เหมาะแก่การเรียนรู้ในทางตรงกันข้ามจะทำการบล็อกเว็บไซต์ที่เป็นอันตรายและไม่เหมาะสม รวมไปถึงการทำโทษเมื่อฝ่าฝืนกฎ อีกทั้งยังสามารถจัดตารางเวลาการเข้าอินเทอร์เน็ต ทำการเก็บบันทึกพฤติกรรม ต่างๆไว้ และสามารถ ควบคุมตั้งค่า หรือฝ้าคุจากที่อื่นโดยผ่านทาง Web site สามารถทำงานได้กับทั้ง IE และ Firefox

ระบบ โปรแกรมควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครองนั้น ได้แบ่งออกเป็นสามส่วน ใหญ่ๆ ได้แก่

1.1.1 ส่วนที่เป็น เครื่องแม่ข่าย ส่วนนี้จะทำหน้าที่เป็น เครื่องแม่ข่าย คอยสำรวจและเก็บ ข้อมูลเว็บไซต์ต่างๆ ที่เป็นประโยชน์และไม่เป็นประโยชน์พร้อมทั้งจัดอันดับความสำคัญและทำหน้าที่ Update ข้อมูลในแก่ส่วนของ โปรแกรม ที่ถูก ติดตั้ง ไว้ใน ตัว เครื่องลูกข่าย เพื่อใช้ในการ บล็อก พิจารณา และควบคุมพฤติกรรมต่อไป ระบบติดต่อกับ web เครื่องแม่ข่าย ซึ่งจะมีการส่ง ข้อมูลให้กับ web เครื่องแม่ข่าย หากมีการทำการร้องขอ รวมทั้งเป็น web เครื่องแม่ข่าย ของข้อมูล ในส่วนที่เป็นการควบคุมจากระยะไกล

1.1.2 ส่วนที่เป็น โปรแกรม ส่วนนี้จะถูกติดตั้งไว้ใน เครื่องลูกข่าย จากการ ติดตั้ง โดยจะ ทำหน้าที่ติดต่อกับผู้ใช้งาน โดยภายใน โปรแกรม นั้นจะมี ส่วนของการ บล็อกเว็บไซต์และ โปรแกรม

ต่างๆ ตามที่ เครื่องแม่ข่าย ได้มีการอัปเดตมาเรื่อยๆ และมีการควบคุมเวลา จัดการตารางเวลารวมทั้งบันทึก พฤติกรรมทางอินเทอร์เน็ต และมีระบบควบคุมการทำโทษ

1.1.3 ส่วนที่เป็นเว็บไซต์ควบคุม ในส่วนนี้จะถูกใช้งานในการขอข้อมูลหรือเฝ้าดูพฤติกรรมของเยาวชน ซึ่งมีตัวเลือกที่ไว้สำหรับปรับค่าได้ เป็นส่วนของหน้าเว็บเพจที่สามารถรับข้อมูลจาก เครื่องแม่ข่าย ได้เช่นเดียวกัน และสามารถปรับค่าต่างๆ เท่าที่จำเป็นได้ ซึ่งมีคุณสมบัติที่น้อยกว่าในส่วนของ โปรแกรม

1.2 วัตถุประสงค์ของโครงการ

- เพื่อให้ผู้ปกครองสามารถควบคุมการใช้งานทางอินเทอร์เน็ตของเยาวชนได้
- เพื่อให้ผู้ปกครองสามารถตรวจสอบพฤติกรรมทางอินเทอร์เน็ตของเยาวชนได้ตลอดเวลา
- เพื่อให้เยาวชนใช้อินเทอร์เน็ตให้เกิดประสิทธิภาพและประโยชน์สูงสุด โดยการจั

อันดับเว็บไซต์ที่เป็นประโยชน์แก่เยาวชนรวมถึงการส่งเสริมให้เยาวชนได้เข้าไปศึกษาเว็บไซต์นั้นๆ

- เพื่อให้เยาวชนสามารถท่องอินเทอร์เน็ตด้วยความปลอดภัยสูงสุด โดยการสำรวจและป้องกันเว็บไซต์ที่เป็นอันตราย หรือมีความเสี่ยงเช่น โทรจัน ไวรัส worm หรือ Spyware และบล็อกเพื่อไม่ให้เยาวชนได้เข้าถึงความเสี่ยงนั้นๆ

- เพื่อให้เยาวชนใช้อินเทอร์เน็ตให้เกิดความถูกต้องและเหมาะสม โดย ปิดกั้นเว็บไซต์ที่ไม่เกิดประโยชน์หรือมอมเมาเยาวชน

1.2 ขอบเขตของโครงการ

1.3.1 จัดตั้งระบบ เครื่องแม่ข่าย สำหรับอัปเดตข้อมูลและติดต่อกับ เครื่องลูกข่าย

1.3.2 จัดทำระบบเว็บไซต์ เพื่อติดต่อระหว่าง เครื่องแม่ข่าย กับในการสร้างระบบควบคุมจากระยะไกล

1.3.3 จัดทำระบบ โปรแกรม สำหรับติดตั้งเข้าไปใน เครื่องลูกข่าย ซึ่งได้แบ่งออกเป็น ส่วนย่อยๆดังนี้

- User Interface ส่วนที่ทำหน้าที่ติดต่อกับผู้ใช้งานเพื่อตั้งค่าต่างๆของระบบ
- Function เป็นฟังก์ชันสำหรับการตอบสนองสิ่งที่ผู้ใช้งานได้ทำการตั้งค่าไว้
- Extension & Plug-in เป็นส่วนที่ทำการฝังไว้กับ โปรแกรม อื่นๆเพื่อทำการควบคุมการ

เข้าถึงอินเทอร์เน็ต

1.4 วิธีการดำเนินการ

1.4.1 การเขียนโปรแกรมป้องกันเว็บไซต์

- ศึกษาและเลือกวิธีการที่จะใช้ในการ สร้าง เว็บไซต์ blocker
- ศึกษาการเขียนโปรแกรม เพื่อสร้างตัวเสริมให้แก่ web browser
- เขียนและทดสอบตัวเสริมที่สร้างขึ้น

1.4.2 การเขียนโปรแกรมป้องกันการเชื่อมต่อ

- ศึกษาการเขียนโปรแกรมและเลือกวิธีการสร้างระบบป้องกันการเชื่อมต่อ
- ศึกษาการเขียนโปรแกรม เพื่อปิดกั้นการเชื่อมต่อจากโปรแกรม
- เขียนโปรแกรมและทดสอบระบบป้องกันการเชื่อมต่อ

1.4.3 การสร้างระบบจัดสรรเวลา

- ศึกษาการเขียนโปรแกรมและเลือกวิธีการสร้าง ระบบจัดสรรเวลา
- ออกแบบหน้าตาของระบบจัดสรรเวลา
- เขียนโปรแกรมและทดสอบ ระบบจัดสรรเวลา
- ทำการเชื่อมระบบจัดสรรเวลาให้ทำงานร่วมกับส่วนอื่นๆ

1.4.4 ระบบบันทึกพฤติกรรม

- ศึกษาการสร้างระบบบันทึกพฤติกรรม
- เขียนโปรแกรมและทดสอบระบบบันทึกพฤติกรรม

1.4.5 ระบบสำรวจและวิเคราะห์ เว็บไซต์ ที่เป็นอันตราย

- ศึกษาการเขียนโปรแกรมเพื่อสำรวจ เว็บไซต์
- ศึกษาการวิเคราะห์ เว็บไซต์ ที่เป็นอันตราย
- ทำการสร้างระบบ วิเคราะห์ เว็บไซต์ ที่เป็นอันตราย

1.4.6 ระบบฐานข้อมูล

- ออกแบบฐานข้อมูล
- รวบรวมของมูลเกี่ยวกับเกมส์ออนไลน์และ โปรแกรมที่เป็น อันตราย

1.4.7 ระบบ เครื่องแม่ข่ายปรับปรุงข้อมูล

- ศึกษาการสร้างระบบ เครื่องแม่ข่ายปรับปรุงข้อมูล
- ทำการสร้างระบบปรับปรุงข้อมูลอัตโนมัติให้แก่โปรแกรม

1.4.8 ระบบควบคุมผ่านเว็บไซต์

- ศึกษาและสร้างในส่วนของการควบคุมผ่านเว็บไซต์

1.4.9 ออกแบบหน้าตาของโปรแกรม

- ออกแบบหน้าตาของโปรแกรมในส่วนของ เครื่องลูกข่าย
- ออกแบบหน้าตาของโปรแกรมในส่วนของ เครื่องแม่ข่าย
- ออกแบบหน้าตาของโปรแกรมในส่วนที่เป็นเว็บไซต์ควบคุม

1.4.10 เพื่อและปรับแก้ และ รวบรวมส่วนต่างๆ ของระบบ

1.5 ประโยชน์ที่ได้รับ

- สามารถควบคุมและติดตามพฤติกรรมทางอินเทอร์เน็ตของบุตรหลานได้ตลอดเวลา
- สามารถจัดระเบียบการเข้าใช้อินเทอร์เน็ตให้แก่เยาวชนในปกครองได้
- มีการป้องกันเว็บไซต์ที่เป็นอันตรายเพื่อความปลอดภัยในการท่องอินเทอร์เน็ตอย่าง

สูงสุด

- เยาวชนได้มีโอกาสเข้าไปยังเว็บไซต์ที่เป็นประโยชน์ ส่งเสริมการเรียนรู้มากขึ้น
- เยาวชนไม่มีโอกาสเข้าไปยังเว็บไซต์อนาจาร ไม่เหมาะสม และมอมเมาเยาวชน
- เยาวชนไทยสามารถใช้เวลาให้เกิดประโยชน์ และ ทำให้ชีวิตประจำวันมีระเบียบวินัย

มากขึ้น

- ผู้ปกครองคลายความกังวลใจเกี่ยวกับภัยทางอินเทอร์เน็ตในรูปแบบต่างๆ
- ผู้ปกครองเกิดความสะดวกสบาย เนื่องจากสามารถเฝ้าดูและควบคุมการใช้อินเทอร์เน็ต

ของเยาวชนในปกครองจากที่อื่นได้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 Proxy Server

พร็อกซี (Proxy) คือ โปรแกรมที่ทำหน้าที่เป็นได้ทั้ง เซิร์ฟเวอร์ และ ไคลเอนต์ สำหรับจุดประสงค์ในการสร้างการร้องขอเพื่อเครื่องไคลเอนต์อื่นๆ การร้องขอจะถูกให้บริการโดยการส่งเข้าไปภายในพร็อกซี ซึ่งในบางกรณีจะมีการเปลี่ยนไปยังเครื่องเซิร์ฟเวอร์อื่น พร็อกซีจะต้องสามารถจัดการได้ทั้งแบบ เซิร์ฟเวอร์ และ ไคลเอนต์ที่ต้องการรายละเอียดดังกล่าว ปัจจุบันได้แบ่งพร็อกซีออกเป็นหลายประเภทแต่สามารถแบ่งเป็นประเภทใหญ่ๆ ได้ 2 แบบ คือ

Transparent Proxy คือ พร็อกซีที่จะไม่ถูกปรับเปลี่ยนการร้องขอหรือการตอบรับเมื่อได้ถูกร้องขอเพื่อการพิสูจน์ยืนยันตัวตนของพร็อกซีและระบุตัวตน (proxy authentication และ identification) ผู้ใช้งานไม่จำเป็นต้องเช็คค่าอะไรในเครื่องคอมพิวเตอร์ของตัวเอง ตัวของ Transparent Proxy นั้นจะใช้หลักของ IP Tables ในการ redirects ผู้ที่ติดต่อผ่านมาทางพอร์ต http ให้ไปยัง พร็อกซีเซิร์ฟเวอร์ ได้เองโดยอัตโนมัติ

Non-Transparent Proxy คือ พร็อกซีที่สามารถปรับเปลี่ยนการร้องขอและตอบรับเพื่อให้บริการสำหรับผู้ใช้งาน เช่น ตัวแทนผู้ให้บริการ, การเปลี่ยนรูปของสื่อกลางประเภทต่างๆ, protocol reduction หรือ anonymity filtering ยกเว้นในที่ที่ใช้ Transparent Proxy หรือ Non-Transparent Proxy ที่อยู่ในสถานะที่แน่นอนแล้ว HTTP proxy ต้องการที่จะนำมาใช้ได้กับทั้งสองชนิดนี้

Proxy server คือ คอมพิวเตอร์ที่มีบริการต่างๆ บนระบบเน็ตเวิร์ค อนุญาตให้ไคลเอนต์สามารถทำการเชื่อมต่อไปยังบริการเน็ตเวิร์คอื่นๆ ได้ โดยไคลเอนต์จะเชื่อมต่อไปที่ proxy server ก่อน จากนั้นร้องขอการเชื่อมต่อ ไฟล์ หรือทรัพยากรอื่นๆ บนเซิร์ฟเวอร์ต่างๆ กัน ซึ่งพร็อกซีจะต้องหาทรัพยากรนั้น ส่งคืนให้กับไคลเอนต์ จากการเชื่อมต่อไปยังเซิร์ฟเวอร์ที่ระบุไว้ หรือคอยให้บริการเองบนแคช (Cache) ที่มีอยู่ แต่บางกรณี พร็อกซีอาจเปลี่ยนการร้องขอของไคลเอนต์หรือเปลี่ยนการตอบกลับจากทางฝั่งเซิร์ฟเวอร์ ขึ้นอยู่กับจุดประสงค์ที่ตั้งไว้ของการใช้งาน

พร็อกซีที่ทำการคัดกรองข้อมูลเนื้อหาบนเว็บ เรียกว่า เว็บพร็อกซี (Web Proxy) โดยเว็บพร็อกซีจำนวนมากจะเปลี่ยนรูปแบบของเว็บเพจที่จะส่งใหม่ เพื่อจุดประสงค์บางอย่างตามที่ต้องการ ผู้ควบคุมดูแลเน็ตเวิร์คจะนำพร็อกซีมาใช้ในการสกัดกั้นไวรัสคอมพิวเตอร์ หรือเนื้อหาของโปรแกรมที่มีเจตนาร้ายต่อผู้ใช้

ในหลายๆ องค์กร อาทิ เช่น บริษัท โรงเรียน จะใช้ Proxy server บังคับใช้บนเน็ตเวิร์คตามนโยบาย หรือเพื่อต้องการความปลอดภัย ปราศจากโปรแกรมประสงค์ร้าย หรือเพื่อบริการแคชในตัว

2.2 อัลกอริทึมพื้นฐานที่ใช้ในโปรแกรม

ในการทดลองจะต้องทำการแบ่งกลุ่มชุดข้อมูลตัวอย่างเพื่อใช้เป็นมาตรฐานในการพิจารณาข้อมูลใหม่ ๆ ว่าเป็นข้อมูลในกลุ่มใด โดยจะทำการแบ่งกลุ่มข้อมูลเป็นสองกลุ่ม คือ กลุ่มของภาพที่ไม่ใช่ภาพอนาจารและกลุ่มของข้อมูลที่เป็นภาพอนาจาร โดยจะใช้หลักการของการวิเคราะห์ส่วนประกอบ, K-Means และ K-Nearest Neighbor (KNN)

หลักการการวิเคราะห์ส่วนประกอบเป็นเทคนิคทางสถิติที่มีประโยชน์ในการทำงานประเภทต่างๆ อาทิเช่น การจดจำใบหน้า (Face recognition) และการบีบอัดภาพ (Image Compression) และเป็นเทคนิคพื้นฐานในการค้นหารูปแบบในข้อมูลที่มีหลายมิติ ก่อนที่จะอธิบายถึงรายละเอียดของหลักการการวิเคราะห์ส่วนประกอบ จะกล่าวถึงทฤษฎีทางคณิตศาสตร์ที่ใช้ในหลักการการวิเคราะห์ส่วนประกอบ ประกอบด้วย สถิติ (statistics) ซึ่งพิจารณาการกระจายตัวของข้อมูล และเมตริกซ์อัลจิบรา (Matrix Algebra) พิจารณาไอแกนเวกเตอร์ (Eigenvectors) และไอแกนแวลูส์ (Eigenvalues) ซึ่งเป็นคุณสมบัติสำคัญของ เมตริกซ์ ที่ซึ่งเป็นรากฐานของหลักการการวิเคราะห์ส่วนประกอบ

2.2.1 สถิติ

สถิติสามารถนำมาใช้เมื่อมีชุดข้อมูลขนาดใหญ่และต้องการวิเคราะห์ความสัมพันธ์ระหว่างแต่ละจุดของชุดข้อมูลนั้น ในที่นี้จะกล่าวถึงการวัดที่กระทำกับกลุ่มของข้อมูล เช่น ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation), วาเรียนซ์ (Variance), โควาเรียนซ์ (Covariance) และ โควาเรียนซ์เมตริกซ์ (Covariance Matrix)

- ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation)

ค่าเบี่ยงเบนมาตรฐานของชุดข้อมูลเป็นการวัดการกระจายตัวของข้อมูล คำนิยามของค่าเบี่ยงเบนมาตรฐานคือระยะทางเฉลี่ยจากค่ากลางของชุดข้อมูลถึงจุดข้อมูลหนึ่งๆ การคำนวณค่าเบี่ยงเบนมาตรฐานมีสมการดังนี้

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{(n-1)}} \quad (2.1)$$

การคำนวณหาค่าเบี่ยงเบนมาตรฐานสามารถหาได้จากกำลังสองของระยะทางจากแต่ละจุดของข้อมูลไปยังค่าเฉลี่ยของกลุ่มข้อมูล ทำการรวมค่าที่คำนวณได้ทั้งหมดหารด้วย จำนวนสมาชิก

ของกลุ่มข้อมูล (n) ลบด้วย 1 แล้วทำการหาค่ารากที่สอง

- วาเรียนซ์ (Variance)

วาเรียนซ์เป็นตัววัดการกระจายตัวของข้อมูลอีกแบบหนึ่ง มีลักษณะแบบเดียวกับ ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation) ซึ่งมีสมการดังนี้

$$s^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)} \quad (2.2)$$

ซึ่งอาจกล่าวได้ว่าวาเรียนซ์เป็นกำลังสองของค่าเบี่ยงเบนมาตรฐาน (s^2) โดยทั้งค่าเบี่ยงเบนมาตรฐานและวาเรียนซ์ต่างก็เป็นการวัดการกระจายตัวของข้อมูล แต่ค่าเบี่ยงเบนมาตรฐานจะนิยมใช้มากกว่า

- โควาเรียนซ์ (Covariance)

จากที่กล่าวมาค่าเบี่ยงเบนมาตรฐานและวาเรียนซ์เป็นการวัดการกระจายตัวของข้อมูล 1 มิติ อย่างไรก็ตามมีชุดข้อมูลจำนวนมากที่มีมากกว่า 1 มิติ และจุดประสงค์ของการวิเคราะห์ข้อมูลทางสถิติเพื่อหาว่าข้อมูลมีความสัมพันธ์ระหว่างมิติหรือไม่ โดยปกติแล้วโควาเรียนซ์จะใช้วัดข้อมูล 2 มิติ ถ้าทำการคำนวณโควาเรียนซ์ระหว่างข้อมูล 1 มิติและตัวมันเองจะได้ค่าวาเรียนซ์ แต่ถ้าข้อมูลมี 3 มิติ (x, y, z) สามารถคำนวณค่าโควาเรียนซ์ได้โดยคำนวณโควาเรียนซ์ระหว่าง x กับ y , x กับ z และ y กับ z การคำนวณค่าโควาเรียนซ์ระหว่าง x กับ x , y กับ y และ z กับ z จะได้ค่าวาเรียนซ์ของ x , y และ z

สมการของโควาเรียนซ์คล้ายกับสมการของวาเรียนซ์ ซึ่งสมการของวาเรียนซ์เป็นดังนี้

$$\text{var}(X) = \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{(n-1)} \quad (2.3)$$

และสมการของโควาเรียนซ์เป็นดังนี้

$$\text{cov}(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \quad (2.4)$$

ถ้าค่าโควาเรียนซ์ที่คำนวณออกมามีค่าเป็นบวกแสดงว่าค่าข้อมูลมีการเพิ่มขึ้นทั้งสองมิติ

แต่ถ้าค่าโควาเรียนซ์เป็นลบแสดงว่าข้อมูลค่าเพิ่มขึ้นหนึ่งมิติ ส่วนอีกมิติมีค่าลดลง แต่ถ้าค่าเป็นศูนย์แสดงว่าข้อมูลทั้งสองมิติไม่ได้ขึ้นต่อกัน โดยค่าของ $cov(X,Y)$ จะมีค่าเท่ากับ $cov(Y,X)$ เมื่อพิจารณาจากสมการ ที่ 2.4

- โควาเรียนซ์เมตริกซ์ (Covariance Matrix)

จากที่ได้กล่าวมาข้างต้น โดยโควาเรียนซ์จะคำนวณข้อมูล 2 มิติ ถ้ามีข้อมูลมากกว่า 2 มิติ แสดงว่ามีการคำนวณโควาเรียนซ์มากกว่าหนึ่งตัว เช่น ชุดข้อมูลที่มี 3 มิติ (x, y, z) สามารถคำนวณ $cov(x,y)$, $cov(x,z)$ และ $cov(y,z)$ ถ้ามีข้อมูล n มิติ จะสามารถคำนวณค่าโควาเรียนซ์ได้แตกต่างกัน

$$\frac{n!}{(n-2)! \cdot 2} \text{ ค่า}$$

ทางที่ง่ายสำหรับการคำนวณค่าโควาเรียนซ์ระหว่างมิติหลายๆมิติ ทำได้โดยการนำค่าโควาเรียนซ์ทั้งหมดใส่ลงในเมตริกซ์ ซึ่งนิยามสำหรับโควาเรียนซ์เมตริกซ์สำหรับชุดข้อมูลที่มี n มิติ เป็นดังนี้

$$C^{n \times n} = (c_{i,j}, c_{i,j} = cov(Dim_i, Dim_j)) \quad (2.5)$$

ซึ่ง $C^{n \times n}$ คือเมตริกซ์ที่มี n แถว n คอลัมน์ และ x Dim เป็นมิติที่ x ถ้ามีชุดข้อมูล n มิติ ดังนั้นเมตริกซ์จะมี n แถว n คอลัมน์ และข้อมูลในเมตริกซ์คือผลลัพธ์ของการคำนวณโควาเรียนซ์ระหว่างมิติ 2 มิติที่ต่างกัน เช่น ข้อมูลที่อยู่ในแถว 2 คอลัมน์ 3 คือการคำนวณโควาเรียนซ์ระหว่างมิติที่ 2 และมิติที่ 3

ตัวอย่างเช่น ข้อมูลมี 3 มิติ (x,y,z) ดังนั้นโควาเรียนซ์เมตริกซ์จะมี 3 แถวและ 3 คอลัมน์ดังนี้

$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix}$$

พิจารณาจากโควาเรียนซ์เมตริกซ์ จะเห็นว่าตามแนวเส้นทะแยงมุมบนซ้ายไปยังล่างขวา เป็นค่าโควาเรียนซ์ระหว่างมิตินั้นและตัวมันเอง หรือก็คือวาเรียนซ์ของมิตินั้น และจาก $cov(a,b)$ มีค่าเท่ากับ $cov(b,a)$ จะเห็นได้ว่าเมตริกซ์นี้ สมมาตรบนเส้นทะแยงมุมนี้

2.2.2 เมตริกซ์อัลจีบรา (Matrix Algebra)

ในหัวข้อนี้จะเป็นพื้นฐานของเมตริกซ์อัลจีบราที่ถูกใช้ในหลักการการวิเคราะห์ ส่วนประกอบ ซึ่งไอแกนเวกเตอร์และไอแกนแวลูส์ที่จะกล่าวต่อไปมาจากเมตริกซ์อัลจีบรา

ตัวอย่างที่ 1 เป็นเมตริกซ์อัลจิบราซึ่งมีไอแกนเวกเตอร์ (non-eigenvector)

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} x \begin{pmatrix} 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix}$$

ตัวอย่างที่ 2 เป็นเมตริกซ์อัลจิบราซึ่งมีไอแกนเวกเตอร์

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} x \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4x \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

• ไอแกนเวกเตอร์ (Eigenvectors)

จากคุณสมบัติของเมตริกซ์ซึ่งสามารถคูณกันได้ระหว่าง 2 เมตริกซ์ ไอแกนเวกเตอร์นั้นเป็นกรณีพิเศษรูปแบบหนึ่ง โดยพิจารณาจากการคูณกันระหว่างเมตริกซ์และเวกเตอร์ในตัวอย่างของหัวข้อเมตริกซ์อัลจิบรา ในตัวอย่างแรกเวกเตอร์ผลลัพธ์ที่ไม่ได้เป็นตัวเลขจำนวนเต็มคูณกับเวกเตอร์ตั้งต้น ต่างกับตัวอย่างที่สองซึ่งผลลัพธ์เป็นสเกลาร์ของเวกเตอร์ตั้งต้น จากตัวอย่างจะพบว่าเวกเตอร์ต้นฉบับประกอบด้วย 2 มิติ โดยเวกเตอร์ $\begin{pmatrix} 3 \\ 2 \end{pmatrix}$ จากตัวอย่างที่ 2 แทนเวกเตอร์ที่มีจุดกำเนิดที่ (0,0) ไปยังจุด (3,2)

ไอแกนเวกเตอร์สามารถหาได้จากเมตริกซ์ที่จำนวนคอลัมน์และจำนวนแถวเท่ากันเท่านั้น แต่ไม่ใช่ทุกเมตริกซ์เช่นนั้นจะมีไอแกนเวกเตอร์ ถ้ากำหนดให้เมตริกซ์ $n \times n$ มีไอแกนเวกเตอร์แล้ว จะมี n ไอแกนเวกเตอร์

การหาไอแกนเวกเตอร์ โดยจะทราบว่าความยาวของเวกเตอร์ไม่มีผลต่อการดูว่าเวกเตอร์นั้นเป็นไอแกนเวกเตอร์หรือไม่ แต่ทิศทางมีผล ดังนั้นการหาไอแกนเวกเตอร์มักจะมีการทำให้ความยาวของเวกเตอร์เป็นหนึ่ง ซึ่งจะทำให้ทุกไอแกนเวกเตอร์มีความยาวเท่ากัน จากตัวอย่างที่ 2 ไอแกนเวกเตอร์คือ $\begin{pmatrix} 3 \\ 2 \end{pmatrix}$ และมีความยาวของเวกเตอร์เท่ากับ $\sqrt{(3^2 + 2^2)} = \sqrt{13}$ เมื่อได้ความยาวของเวกเตอร์แล้วจะนำไปหารเวกเตอร์ตั้งต้น ซึ่งจะทำให้ได้เวกเตอร์ที่มีความยาวเท่ากับหนึ่ง

• ไอแกนแวลูส์ (Eigenvalues)

ไอแกนแวลูส์มีความสัมพันธ์ใกล้เคียงกับไอแกนเวกเตอร์จากตัวอย่างที่ 2 ค่าจำนวนเต็ม (4) เรียกว่า ไอแกนแวลูส์ซึ่งสัมพันธ์กับไอแกนเวกเตอร์ ดังนั้นสามารถพบไอแกนแวลูส์ และไอแกนเวกเตอร์มาเป็นคู่ นั้นหมายถึงเมื่อหาไอแกนเวกเตอร์ได้ก็จะได้อิแกนแวลูส์ด้วย

2.2.3 หลักการวิเคราะห์แยกแยะส่วนประกอบ (Principal Components Analysis: PCA)

หลักการวิเคราะห์แยกแยะส่วนประกอบ (PCA) คือ วิธีการบ่งชี้ถึงรูปแบบของข้อมูล และช่วยแสดงข้อมูลให้เห็นถึงจุดเด่นของความเหมือนและความต่างของข้อมูล อย่างไรก็ตาม รูปแบบของข้อมูลนั้นยากที่จะหาได้ เมื่อข้อมูลนั้นมีหลายมิติ (Dimension) และยังยากที่จะแสดงเป็นรูปภาพหรือกราฟได้ ดังนั้น PCA จึงเป็นเครื่องมือที่มีประสิทธิภาพอย่างมากในการแยกแยะหรือจำแนกข้อมูล นอกจากนี้ PCA ยังมีประโยชน์ในการช่วยกำหนดรูปแบบของข้อมูล และบีบอัดข้อมูลได้โดยการลดจำนวนมิติ ปราศจากการสูญเสียเนื้อหาของข้อมูล โดยการวิเคราะห์แยกแยะส่วนประกอบการใช้งานมีวิธีการทำดังนี้

ขั้นตอนที่ 1 รวบรวมข้อมูล

- รวบรวมข้อมูลโดยพิจารณาถึงมิติ ของปัจจัย

ขั้นตอนที่ 2 หักลบออกจากค่ากลาง (Mean)

- เราจะต้องหักลบค่ากลางออกจากแต่ละข้อมูลของมิติ ค่ากลางที่จะทำการลบนี้เป็นค่าเฉลี่ยตามขวางของแต่ละมิติ ดังนั้น ค่า x จะต้องลบด้วย \bar{x} และค่า y จะต้องลบด้วย \bar{y} ทำให้ผลลัพธ์ของชุดข้อมูลที่มีค่าเท่ากับค่ากลาง จะเท่ากับ 0 (ข้อมูลที่ถูกลบออกจากค่ากลางแล้ว จะเรียกว่า Data Adjust)

ขั้นตอนที่ 3 การคำนวณค่าเมทริกซ์ โควาเรียนซ์ (Covariance Matrix)

- สมมติว่าหากข้อมูลเป็น 2 มิติ ทำให้เมทริกซ์โควาเรียนซ์เป็น 2×2

ขั้นตอนที่ 4 การคำนวณ ไอแกนเวกเตอร์ (Eigenvectors) และ ไอแกนแวลูส์ (Eigenvalues) ของค่าเมทริกซ์โควาเรียนซ์

- เมื่อค่าเมทริกซ์โควาเรียนซ์เป็น สี่เหลี่ยมจัตุรัสแล้ว (จำนวน Row เท่ากับจำนวนของ Column) เราจะสามารถคำนวณค่าของไอแกนเวกเตอร์ และไอแกนแวลูส์ สำหรับเมทริกซ์ได้ โดยค่าไอแกนเวกเตอร์เป็นเวกเตอร์ 1 หน่วย ที่มีความยาวเท่ากับ 1 หากเราดูข้อมูลที่ลงไปในกราฟ จะเห็นว่าข้อมูลจะค่อนข้างมีรูปแบบที่เห็นได้ชัด เป็นจุดแนวเส้นทแยงมุม ซึ่งไอแกนเวกเตอร์จะแสดงให้เราเห็นว่าจำนวนข้อมูลใน 2 ชุดมีความสัมพันธ์กับเส้นตรงของกราฟอย่างไร ดังนั้นในกระบวนการทำงานของการสร้างไอแกนเวกเตอร์ของเมทริกซ์โควาเรียนซ์ เราจึงสามารถแยกเส้นที่อยู่บนกราฟซึ่งเป็นลักษณะเด่นของข้อมูลออกมาได้

ขั้นตอนที่ 5 การเลือกส่วนประกอบ และเปลี่ยนให้อยู่ในลักษณะของเวกเตอร์

- จะสังเกตได้ว่าไอแกนแวลูส์จะเป็นค่าที่ค่อนข้างแตกต่างกัน ในความเป็นจริงไอแกนเวกเตอร์กับค่าไอแกนแวลูส์ที่มีค่าสูงสุด จะเป็นหลักการส่วนประกอบของชุดข้อมูล เช่น หากไอแกนเวกเตอร์ กับ ไอแกนแวลูส์ที่มีค่าสูง จะเป็นจุดที่อยู่ตรงกลางของชุดข้อมูล ซึ่งเป็นความสัมพันธ์ที่เห็นได้อย่างชัดเจนระหว่างมิติของข้อมูล โดยไอแกนเวกเตอร์กับที่ถูกพบจากเมทริกซ์โควาเรียนซ์

ในขั้นตอนต่อไปจะเป็นลำดับโดยไอแกนแวลูส์ จากค่าสูงสุดไปต่ำสุด นี้จะช่วยให้สามารถเรียงลำดับความสำคัญของปัจจัยได้ ดังนั้นจึงสามารถตัดสินใจที่จะเพิกเฉยต่อ ปัจจัยที่มีความสำคัญน้อย หรือ ไอแกนแวลูส์มีค่าน้อย ซึ่งไม่ถือว่าเป็นการสูญเสียข้อมูลมากนัก หากคุณปล่อยบางปัจจัยออกไป ข้อมูลสุดท้ายจะมีค่าน้อยกว่าข้อมูลเริ่มต้น ถ้าข้อมูลเริ่มต้นมี n มิติ และทำการคำนวณ n ของไอแกนแวลูส์ และไอแกนแวลูส์ และดังนั้นเราจะเลือกเพียงไอแกนแวลูส์ p แรก และข้อมูลชุดสุดท้ายจะมีเพียง p มิติ จากนั้นต้องทำการแปลงให้อยู่ในรูปของเวกเตอร์ (Feature vector) โดยการนำค่าไอแกนแวลูส์ที่เลือกมาจาก list ของไอแกนแวลูส์ และเปลี่ยนเป็นเมทริกซ์ให้ไอแกนแวลูส์อยู่ในคอลัมน์

$$\text{FeatureVector} = (\text{eig1 eig2 eig3 ... eign})$$

ขั้นตอนที่ 6 กลายมาเป็นข้อมูลชุดใหม่

- นี้จะเป็นขั้นตอนสุดท้ายของ PCA และเป็นขั้นตอนที่ง่ายที่สุด โดยเราจะเลือกส่วนประกอบหรือปัจจัย (eigenvector) ซึ่งเราอยากจะทำข้อมูลทั้งหมดของเรา และเปลี่ยนเป็นรูปแบบของเวกเตอร์ โดยทำการทรานส์โพส (Transpose) ของเวกเตอร์ และคูณส่วนนี้ด้วยชุดข้อมูลเริ่มต้น

$$\text{FinalData} = \text{RowFeatureVector} \times \text{RowDataAdjust},$$

ซึ่ง RowFeatureVector เป็นเมทริกซ์ที่มีไอแกนแวลูส์อยู่ในคอลัมน์ที่ถูกรทรานส์โพส ดังนั้นไอแกนแวลูส์ในตอนนี้จะอยู่ในแนวนอน, ไอแกนแวลูส์ที่มีค่าสูงจะอยู่ด้านบนสุด และ RowDataAdjust เป็นข้อมูลที่หักออกจากค่ากลาง และทรานส์โพส ส่วน FinalData เป็นชุดข้อมูลสุดท้าย และประกอบข้อมูล (Data Items) ในคอลัมน์ และมีขนาดไปกับแถว

2.2.4 K-Means Clustering and K-nearest neighbor algorithm

K-Means เป็นอัลกอริทึมที่ใช้แยกข้อมูลออกเป็นกลุ่มๆ โดยอาศัยคุณลักษณะต่างๆ เพื่อแยกข้อมูลออกเป็น K กลุ่ม การจัดกลุ่มแบบ K-Means ทำได้โดย

1. หาจุดเซ็นทรอยด์ (Centroid)
2. หาระยะทางของแต่ละข้อมูลกับจุดเซ็นทรอยด์
3. จัดกลุ่มข้อมูลไปยังกลุ่มที่มีระยะทางจากจุดเซ็นทรอยด์ของกลุ่มนั้นสั้นที่สุด

ในตอนแรกต้องกำหนดจุดเซ็นทรอยด์ตั้งต้นไว้ก่อน โดยการเลือกจุดข้อมูลใดๆในชุดข้อมูลนั้น ในการทดลองจะทำการกำหนดจุดเซ็นทรอยด์ 3 จุด จากนั้นทำการคำนวณระยะทางจากข้อมูลแต่ละตัวถึงจุดเซ็นทรอยด์ทั้งสามจุด โดยการใช้การคำนวณระยะทางแบบยูคลิดีส (Euclidean distance) โดยมีสมการดังนี้

$$\text{Distance} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} \quad (2.6)$$

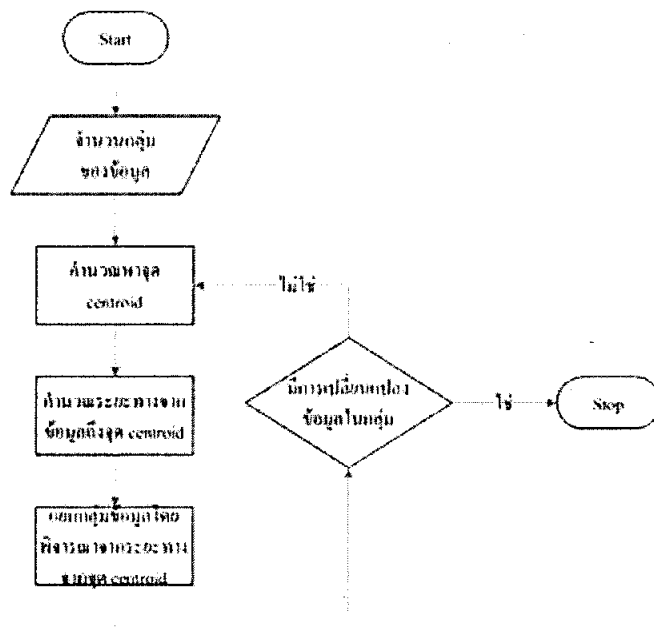
เมื่อคำนวณหาระยะทางจากข้อมูลแต่ละตัวกับจุดเซ็นทรอยด์ทั้งสามจุดแล้ว จะทำการแยกข้อมูลแต่ละตัวไปยังกลุ่มที่มีระยะทางจากจุดเซ็นทรอยด์ของกลุ่มนั้นน้อยที่สุด เมื่อได้ชุดข้อมูลของกลุ่มที่ 1, 2, 3 มาแล้ว ก็จะต้องทำการคำนวณหาจุดเซ็นทรอยด์ของกลุ่มนั้นใหม่ โดยดูจากชุดข้อมูลในกลุ่มนั้นๆ คำนวณหาจุดเซ็นทรอยด์ โดยการหาค่าเฉลี่ยของค่า x และ y ของข้อมูลแต่ละตัวในกลุ่มนั้น

$$c = \left(\frac{x_1 + x_2 + \dots + x_n}{m}, \left(\frac{y_1 + y_2 + \dots + y_n}{m} \right) \right) \quad (2.7)$$

n = จำนวนข้อมูลของกลุ่มนั้น

m = จำนวนจุด เซ็นทรอยด์

เมื่อได้จุดเซ็นทรอยด์ใหม่ของแต่ละกลุ่มแล้วก็ทำการคำนวณหาระยะทางแบบเดิมทำวนไปเรื่อยๆ จนกว่าค่าของจุดเซ็นทรอยด์จะคงที่



รูปที่ 2.1 กลไกการทำงานของ K-means

2.2.5 K-Nearest Neighbor (KNN)

วิธีการคำนวณอัลกอริทึมของ KNN มีขั้นตอนดังนี้

1. พิจารณาพารามิเตอร์ $K =$ จำนวนของ nearest neighbor
2. คำนวณระยะทางระหว่างจุดเซ็นทรอยด์ที่หามาจาก K-means กับข้อมูล
3. เรียงลำดับระยะทางและพิจารณา nearest neighbor โดยมีพื้นฐานบน K-th minimum distance
4. รวมกลุ่มของ nearest neighbor เข้าด้วยกัน
5. ใช้กลุ่มหลักของ nearest neighbor เป็นค่าที่ไว้คาดการณ์ความน่าจะเป็นของข้อมูลใหม่ ๆ ว่า เป็นกลุ่มไหน

2.3 การคัดกรองข้อความ (Text Detection)

การตรวจสอบเนื้อหาบนเว็บเพจสามารถบ่งบอกว่าเว็บไซต์นั้นเป็นเว็บธนาคารหรือไม่ โดยพิจารณาคุณสมบัติต่างๆของโครงสร้าง และรูปแบบที่ได้รับมาจากการทดลองจากหน้าเว็บเพจจำนวนมาก ผลลัพธ์ของการการตรวจสอบคุณสมบัติต่างๆ จะช่วยบอกถึงโครงสร้างของเว็บไซต์ธนาคารแยกจากเว็บไซต์ที่ไม่ธนาคาร ซึ่งมีหลายรูปแบบที่ปรากฏเฉพาะในเว็บธนาคาร

ในการค้นหาคำที่ปรากฏภายในเอกสาร HTML จะต้องพิจารณาแต่ละส่วน มีดังนี้

Title: จะอยู่ภายใน tag <TITLE> และ </TITLE> ซึ่งอยู่ในส่วนของ <HEAD>

Meta: จะอยู่ภายใน tag <META> และ </META>

Body: จะเป็นส่วนที่เราเห็นปรากฏอยู่ในเว็บเพจ

ในการทดลองนี้จะพิจารณาคูสมบัติของหน้าเว็บเพจด้วยกันทั้งหมด 10 ปัจจัย ดังนี้

- 1) จำนวน IMG Tag ทั้งหมดในหน้าเว็บเพจ
- 2) จำนวน META Tag ทั้งหมดในหน้าเว็บเพจ
- 3) จำนวนคำหยาบคายทั้งหมดใน Meta และ Title
- 4) จำนวน SCRIPT Tag ทั้งหมดในหน้าเว็บเพจ
- 5) จำนวนลิงค์ทั้งหมดในหน้าเว็บเพจ
- 6) จำนวนลิงค์ที่เป็นรูปภาพทั้งหมดในหน้าเว็บเพจ
- 7) จำนวนลิงค์ที่เป็นตัวหนังสือทั้งหมดในหน้าเว็บเพจ
- 8) จำนวนลิงค์ที่ไม่ใช่ที่อยู่ภายในเว็บตัวเองในหน้าเว็บเพจ (Non-Relative Link)
- 9) จำนวนลิงค์ที่อยู่ภายในเว็บตัวเองในหน้าเว็บเพจ (Relative Link)
- 10) จำนวน PARAM Tag ทั้งหมดในหน้าเว็บเพจ (เพื่อนับจำนวนสื่อมัลติมีเดียทั้งหลาย)

เมื่อเราได้คุณสมบัติทั้ง 10 ปัจจัยของหน้าเว็บเพจมาแล้ว นำมาทำการ Normalized ข้อมูลก่อน จากนั้นนำค่าเหล่านี้มาทำลดมิติของข้อมูล (Dimension Reduction) จาก 10 ปัจจัย (10 มิติ) ให้เหลือ 2 มิติโดยไม่มีการสูญเสียความสำคัญของข้อมูลมากนัก ซึ่งใช้เทคนิคที่เรียกว่า Eigen (อธิบายอยู่ในเนื้อหา 2.2.3 หลักการวิเคราะห์แยกแยะส่วนประกอบ Principal Components Analysis:

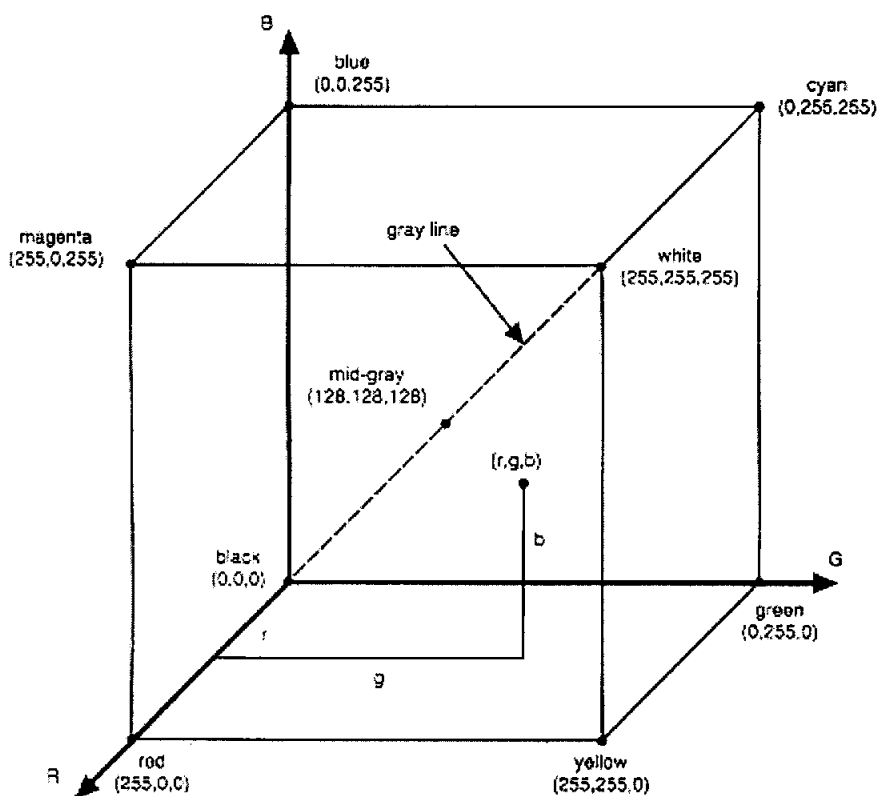
PCA) สุดท้ายนำข้อมูลที่เหลือใน 2 มิติ มาทำการพล็อตกราฟแล้วจึงพิจารณาว่าข้อมูลที่ได้ควรเป็นกลุ่มของข้อมูลเว็บไซต์ธนาคาร หรืออยู่ในกลุ่มของเว็บไซต์ปกติ ที่ไม่มีสื่อลามกอนาจาร

2.4 การคัดกรองรูปภาพ (Image Detection)

2.4.1 พื้นฐานและระบบโครงสร้างสี

2.4.1.1 ระบบโครงสร้างสีอาร์จีบี (RGB Color Model)

ในโครงสร้างนี้ สีแต่ละสีจะปรากฏในรูปแบบของแม่สี คือ สีแดง สีเขียว และ สีน้ำเงิน โครงสร้างสีนี้มีโครงสร้างเป็นลักษณะแกนคาร์ทีเซียน (Cartesian coordinate) โดยมีลักษณะเป็นทรงลูกบาศก์ มีค่าสีแดง สีเขียว และสีน้ำเงินอยู่ที่มุมทั้งสามที่เป็นแกน และมีสีฟ้าคราม สีม่วง สีเหลือง อยู่ที่มุมอีก 3 มุม สีดำจะอยู่ที่จุดกำเนิดคือถ้าทุกสีมีค่าเป็น 0 คือเป็น สีดำ สีขาวก็คือตรงข้ามกับสีดำซึ่งอยู่ที่มุม ไกลสุดจากจุดกำเนิดในลักษณะทะแยงมุม ในรูปแบบนี้ค่าระดับสีเทา (Gray scale) จะอยู่บนเส้นระหว่างสีดำและสีขาว และสีอื่นๆ ก็จะมีตำแหน่งอยู่ภายในลูกบาศก์นี้



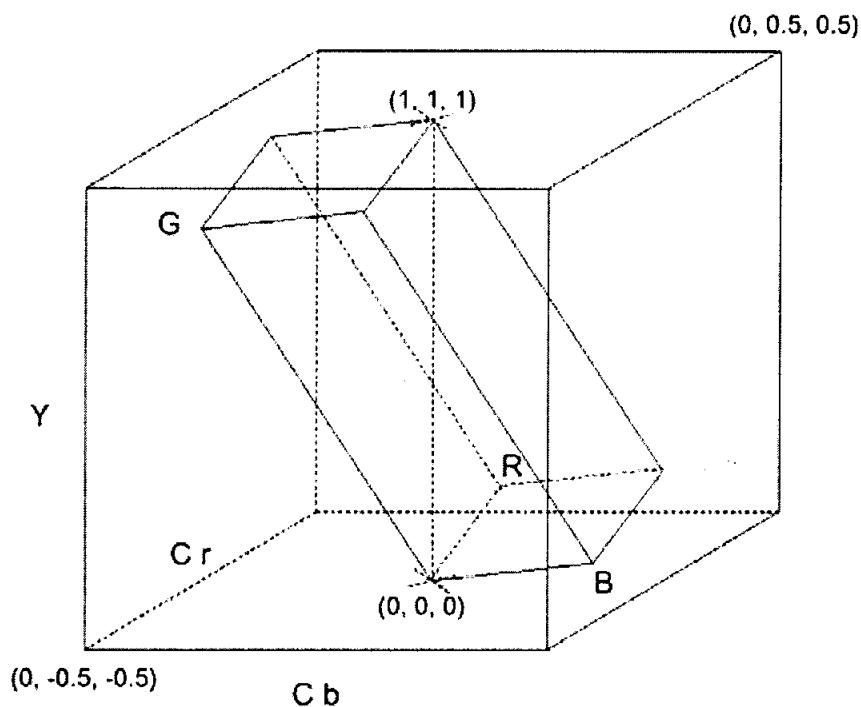
รูปที่ 2.2 โครงสร้างสีอาร์จีบี เป็นลูกบาศก์

ค่าของสีคือจุดที่อยู่บนพื้นผิวหรือในลูกบาศก์ถูกกำหนดค่าโดยเวกเตอร์ที่ชี้ออกจากจุดกำเนิด ซึ่งช่องว่างแต่ละที่ในลูกบาศก์เรียกว่า ความลึกของพิกเซล (Pixel depth) โดยแม่สีแต่ละสีมีค่า 8 บิต มีได้ 256 ค่า เมื่อรวมสีทั้งหมดของความลึกของพิกเซลในลูกบาศก์ (24 บิตอาร์จีบี) จะได้

ทั้งหมด 16,777,216 สี ภาพในโครงสร้างสีอาร์จีบี ประกอบด้วยภาพสามระนาบที่เป็นอิสระจากกัน

2.4.1.2 ระบบโครงสร้างสีวายซีบีซีอาร์ (YCbCr Color Model)

ในโครงสร้างนี้จะใช้เป็นที่แพร่หลายสำหรับดิจิทัลวิดีโอ ในรูปแบบของโครงสร้างนี้ ค่าปริมาณของแสงในการส่องสว่างจะเก็บข้อมูลนี้ไว้ในส่วนของ (Y) และในส่วนความแตกต่างของสีนั้นจะแบ่งเป็น 2 สีคือ Cb และ Cr โดย Cb จะแสดงให้เห็นถึงความแตกต่างของสีแดง และอ้างถึงค่าในหมวดสีแดงโดยโครงสร้างสีวายซีบีซีอาร์ มีความเที่ยงตรงและแม่นยำมาก ในส่วนของการส่องสว่าง และหมวดสีซึ่งเป็นโครงสร้างสีที่ใช้กันในการเข้ารหัสแบบเอ็มพีอีจี (MPEG & JPEG)



รูปที่ 2.3 ความสัมพันธ์ระหว่างโครงสร้างสี YCbCr กับ RGB

2.4.1.3 การแปลงรูปแบบสี

การแปลงรูปแบบสีจาก RGB เป็น YCbCr

$$Y = (0.299 * R) + (0.587 * G) + (0.114 * B) \quad (2.8)$$

$$Cb = (-0.168736 * R) - (0.331264 * G) + (0.5 * B) \quad (2.9)$$

$$Cr = (0.5 * R) - (0.418688 * G) - (0.081312 * B) \quad (2.10)$$

โดยกำหนดให้ค่าของ R, G และ B เป็นค่าของ สีแดง สีเขียว และ สีน้ำเงินตามลำดับ ซึ่งมีค่าตั้งแต่ 0 ถึง 255 แล้วค่าของ Y จะมีค่าในช่วงตั้งแต่ 0 ถึง 255 และ ค่า Cb, Cr อยู่ในช่วงตั้งแต่ 128- ถึง 128

การแปลงรูปแบบสีจาก YCbCr เป็น RGB

$$R = 1.164*(Y-16) + 1.596*(Cr-128) \quad (2.11)$$

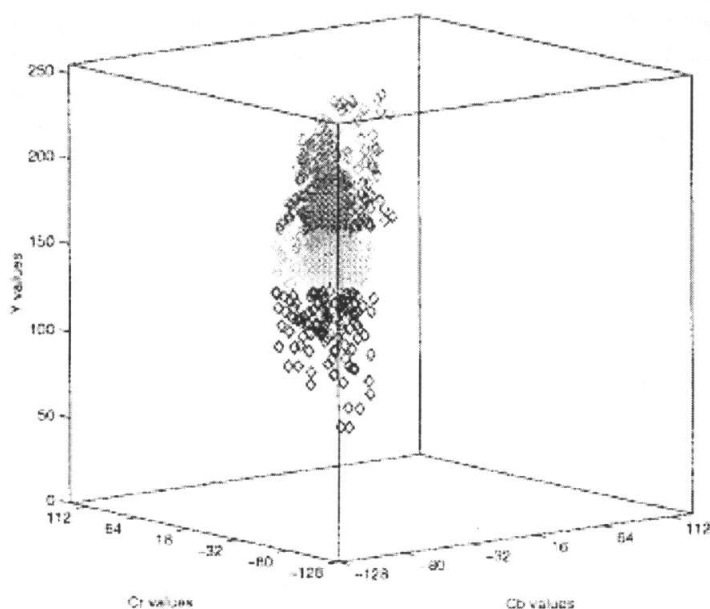
$$G = 1.164*(Y-16) - 1.391*(Cb-128) - 0.813*(Cr-128) \quad (2.12)$$

$$B = 1.164*(Y-16) + 2.018*(Cb-128) \quad (2.13)$$

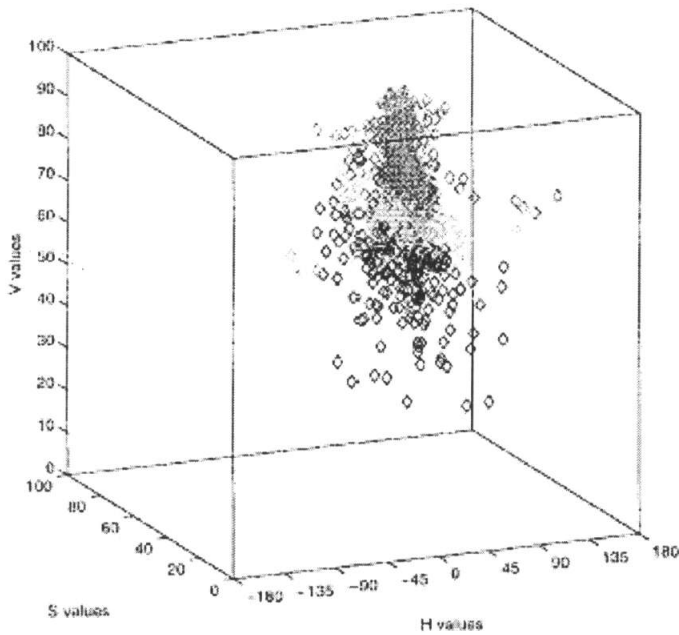
2.4.2 กระบวนการค้นหาพื้นที่สีผิว

การตรวจหาภาพอนาจารต้องอาศัยกระบวนการค้นหาพื้นที่สีผิว (Detection of Skin color regions) เพื่อแยกสีผิวมนุษย์ออกจากส่วนที่ไม่ใช่สีผิวในรูปภาพ จากงานวิจัยเรื่อง [Face Detection Using Quantized Skin Color Regions Marging and Wavelet Packet Analysis] ในหัวข้อเรื่อง “Skin Color segmentation” เสนอไว้ว่า ในงานวิจัยนั้นใช้โครงสร้างสีสองแบบ คือ YCbCr ซึ่งเป็นโครงสร้างสีที่ใช้กันในการเข้ารหัสแบบ MPEG และ JPEG และใช้โครงสร้างสีแบบ HSV(Hue ,Saturation ,Value)ที่ใช้ในงานคอมพิวเตอร์กราฟิก และใกล้เคียงกับการผสมสีของศิลปินข้อมูลที่ใช้วิเคราะห์เป็นการสุ่มตัวอย่างสีผิว 950 ตัวอย่าง โดยสุ่มจากหลากหลายเชื้อชาติและมีความสว่างแตกต่างกันไป โดยจะมีการแสดงกลุ่มตัวอย่างสีผิวในแกนของ YCbCr และ HSV ซึ่งแสดงเป็นกราฟข้อมูล 3 มิติ

จากรูปที่ 2.4 และ 2.5 เราสามารถสังเกตเห็นได้ว่า ตัวอย่างสีผิวทั้งใน YCbCr และ HSV ต่างก็มีการกระจายตัวที่มีลักษณะคล้ายกัน โดยข้อมูลค่อนข้างที่จะมีการรวมตัวกันอยู่เป็นกลุ่ม ในงานวิจัยนั้นมุ่งที่จะสร้างขอบเขต 3 มิติเพื่อให้ครอบคลุมกลุ่มของตัวอย่างสีผิวที่สุ่มได้มากที่สุดด้วยสมการเส้นตรง



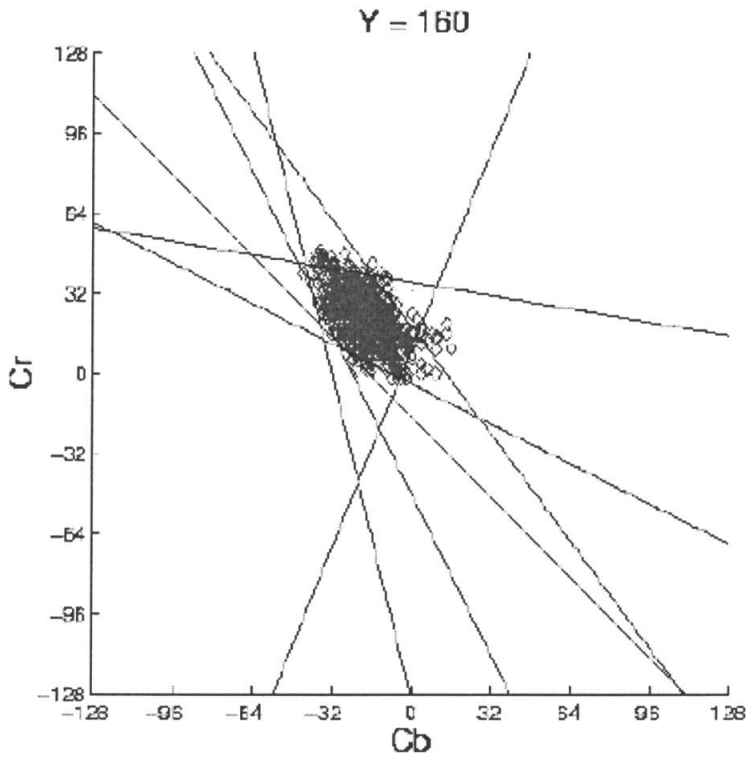
รูปที่ 2.4 กลุ่มตัวอย่างสีผิวในแกนของโครงสร้างสี YCbCr



รูปที่ 2.5 กลุ่มตัวอย่างสีผิวในแกนของโครงสร้างสี HSV

ในกราฟ YCbCr พบว่าการเปลี่ยนแปลงของค่า Y มีผลเพียงเล็กน้อยต่อการเปลี่ยนแปลงในระนาบ CbCr โดยทำการพิจารณาใน 2 กรณีคือ ภาพที่มีความมืด (ค่าของ Y มีค่าประมาณ 50) และภาพที่สว่าง (ค่าของ Y มีค่าประมาณ 240)

เมื่อกำหนดค่า $Y = 160$ เพื่อให้ได้กราฟในระนาบ CbCr และกำหนดสมการเส้นตรงให้ครอบคลุมกลุ่มตัวอย่างดังรูปที่ 2.8 การกำหนดค่า Y ค่าเดียวนี้ก่อให้เกิดความผิดพลาดในกระบวนการตรวจหาสีผิว จึงแบ่งช่วง Y ออกเป็นสองช่วงที่จุด $Y = 128$ ให้มีขอบเขตที่ต่างกัน โดยกำหนดเป็นตัวแปรที่ต่างกัน โดยที่จุด $Y = 128$ เป็นจุดแบ่งขอบเขตความมืดและสว่างและกำหนดสมการเส้นตรงแปดสมการบนพื้นที่ทั้งสองส่วน



รูปที่ 2.6 กรอบพื้นที่สีผิวโครงสร้างสี YCbCr ที่ระนาบ $Y = 160$

ทำการคำนวณหาค่า ของ θ_1 , θ_2 , θ_3 และ θ_4 ใน 2 กรณีคือ

เมื่อ $Y > 128$

$$\theta_1 = -2 + \frac{256 - Y}{16}$$

$$\theta_2 = 20 - \frac{256 - Y}{16}$$

$$\theta_3 = 6$$

$$\theta_4 = -8$$

เมื่อ $Y \leq 128$

$$\theta_1 = 6$$

$$\theta_2 = 12$$

$$\theta_3 = 2 + \frac{Y}{32}$$

$$\theta_4 = -16 + \frac{Y}{16}$$

จากช่วงของค่า Y ทั้งสองช่วง สามารถกำหนดขอบเขตสีผิวได้ด้วยสมการต่อไปนี้

$$Cr \geq -2(Cb + 24)$$

$$Cr \geq -(Cb + 17)$$

$$Cr \geq -4(Cb + 32)$$

$$Cr \geq 2.5(Cb + \theta_1)$$

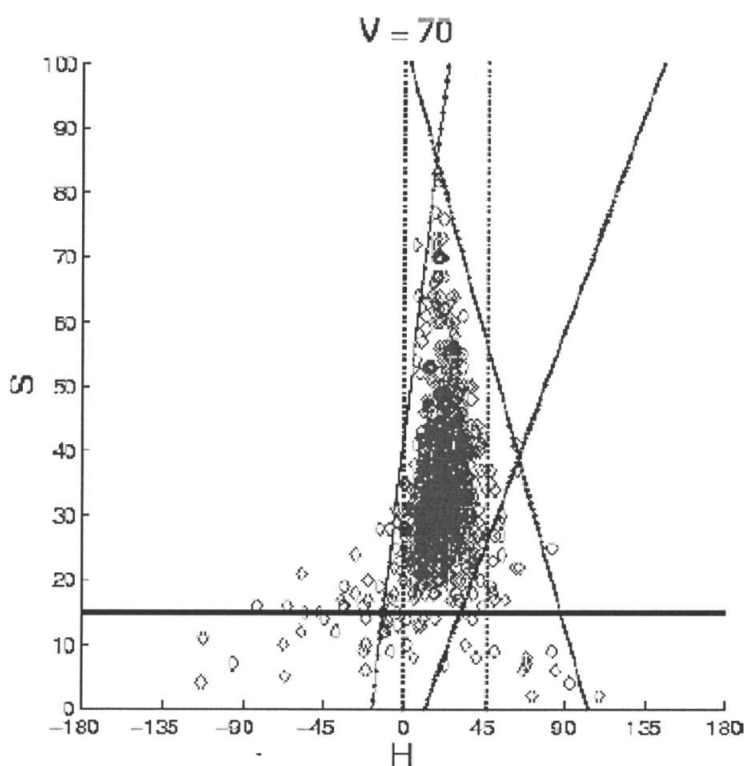
$$Cr \geq \theta_3$$

$$Cr \geq -0.5(\theta_4 - Cb)$$

$$Cr \leq \frac{220 - Cb}{6}$$

$$Cr \leq \frac{4}{3}(\theta_2 - Cb)$$

สมการเหล่านี้หมายความว่าพิกเซลเป็นสีผิวของมนุษย์ นอกเหนือจากนี้ไม่ใช่สีผิวมนุษย์ โครงสร้างสี YCbCr นั้นเป็นกรรมวิธีที่ดีในการตรวจสอบสีผิวมากกว่า โครงสร้างสี HSV อย่างไรก็ตามก็ยังมีผู้ใช้งานบางคนใช้การฉายภาพนระนาบของ HS ซึ่งทำได้โดยการกำหนดค่า thresholds ให้กับ Hue และ Saturation ผลของการใช้ค่า thresholds นี้ เราพบว่าผลของการจำแนกสีผิวจะได้รับผลกระทบจากความแตกต่างของค่าความสว่าง



รูปที่ 2.7 กรอบพื้นที่สีผิวโครงสร้างสี HSV ที่ระนาบ $V = 70$

ในการทำงานเดียวกันกับกรณีของ YCbCr ทำการกำหนดค่า $V = 70$ และได้สมการ ความสัมพันธ์ของค่า Hue และ Saturation ดังนี้

$$S \geq 10$$

$$V \geq 40$$

$$S \leq -H - 0.1V + 110$$

$$\text{ถ้า } H \geq 0 \text{ แล้ว } S \leq 0.08(100-V)H + 0.5V$$

$$\text{ถ้า } H < 0 \text{ แล้ว } S \leq 0.5H + 35$$

2.4.3 กระบวนการปรับความคมชัดของภาพ

จุดประสงค์หลักของการปรับความคมชัดของภาพคือเพื่อทำการเน้นส่วนรายละเอียดในภาพ หรือ ปรับปรุงรายละเอียดของภาพที่ถูกทำให้พร่ามัว (Blur) อาทิเช่นความผิดพลาดหรือผลกระทบที่เกิดขึ้นในกระบวนการได้มาซึ่งภาพ (Image acquisition) การใช้กระบวนการของความแตกต่างของสีภายในภาพ (image differentiation) ที่แสดงถึงอัตราความไม่ต่อเนื่องของสีเพื่อตรวจหาเส้นขอบของภาพ และส่วนที่ไม่ต่อเนื่องกันอื่นๆ เช่น สิ่งรบกวน (noise)

กระบวนการปรับความคมชัดของภาพมีพื้นฐานบน first และ second-order derivatives ซึ่งแสดงจุดสนใจบนพื้นที่ที่มีค่าระดับสีเทาคงที่ (flat segments), จุดเริ่มและจุดสิ้นสุดของความไม่ต่อเนื่อง (step and ramp discontinuities) และ พื้นที่ที่มีระดับสีไล่เลี่ยกัน (ramps) โดยมีคุณสมบัติดังนี้

2.4.3.1 First derivative

Derivative ในฟังก์ชันดิจิทัลอนั้นได้ถูกอธิบายในรูปของความแตกต่าง (Difference) โดย First derivative ต้องมีคุณสมบัติดังนี้

- ต้องมีค่าเท่ากับ ศูนย์ บนพื้นที่ที่มีค่าระดับสีเทาคงที่ (flat segments)
- ต้องมีค่าไม่เท่ากับศูนย์ที่จุดเริ่มต้นของพื้นที่ที่มีระดับสีเทาต่างกัน (gray-level step or ramp)
- ต้องมีค่าไม่เท่ากับศูนย์บนพื้นที่ที่มีระดับสีไล่เลี่ยกัน (ramps)

โดย First Order derivative ของฟังก์ชันหนึ่งมิติแนวแกน เอ็กซ์ ($f(x)$) คือ

$$\frac{\partial f}{\partial x} = f(x-1) - f(x) = f'(x) \quad (2.14)$$

2.4.3.2 Second derivative

Derivative ในฟังก์ชันดิจิทัลอนั้นได้ถูกอธิบายในรูปของความแตกต่าง (Difference) โดย Second derivative ต้องมีคุณสมบัติดังนี้

- ต้องมีค่าเท่ากับ ศูนย์ บนพื้นที่ที่มีค่าระดับสีเทาคงที่ (flat segments)
- ต้องมีค่าไม่เท่ากับศูนย์ที่จุดเริ่มต้นของพื้นที่ที่มีระดับสีเทาด่างกัน (gray-level step or ramp)
- ต้องมีค่าเท่ากับศูนย์บนพื้นที่ที่มีระดับสีไล่เฉื่อยกัน (ramps)

โดย Second Order derivative ของฟังก์ชันหนึ่งมิติแนวแกน เอ็กซ์(f(x)) คือ

$$\begin{aligned}\frac{\partial^2 f}{\partial x^2} &= f'(x) - f'(x-1) \\ &= (f(x+1) - f(x)) - (f(x) - f(x-1)) \\ &= f(x-1) + f(x+1) - 2f(x)\end{aligned}\quad (2.15)$$

2.4.3.3 การใช้ First Derivative ในการเพิ่มคุณภาพของภาพ

First Derivative ในการประมวลผลภาพนั้นเป็นเครื่องมือในการหาค่าความลาดชัน (gradient) สำหรับฟังก์ชัน $f(x,y)$ ค่าความชันของ f ที่ระยะพิกัด (x,y) ถูกแสดงในรูปของเวกเตอร์ที่มีสององค์ประกอบดังนี้

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}\quad (2.16)$$

ในทางปฏิบัติเราสามารถประมาณค่าขนาดความชันได้ด้วยการใช้ค่าสมบูรณ์ ได้ดังนี้

$$\nabla f \approx |G_x| + |G_y|\quad (2.17)$$

พิจารณารูปที่ 2.10(a) จุดของภาพที่มีขอบเขต 3 x 3 พิกเซล (ค่า z เป็นค่าระดับสีเทา) Robert [1965] ได้เสนอคำนิยามในการประมวลผลภาพยุคแรกโดยการใช้ค่าความแตกต่างในแนวทะแยงเป็น $G_x = (z_9 - z_5)$ และ $G_y = (z_8 - z_6)$ และจากสมการที่ 2.9 สามารถคำนวณหาขนาดของความชันได้ดังนี้

$$\nabla f \approx |z_9 - z_5| + |z_8 - z_6|\quad (2.18)$$

สมการที่ 2.17 นี้สามารถใช้ในลักษณะของ masks ในรูปที่ 2.10(b) และ (c) โดยเรียกว่า Roberts cross-gradient operators

อย่างไรก็ตามการใช้ขนาดของ mask เป็นเลขคู่่นั้นทำได้ไม่สะดวก โดยขนาดของ mask ที่เล็กที่สุดที่เราสนใจคือ 3 x 3 พิกเซลซึ่งเมื่อทำการคำนวณหาขนาดของความชันที่จุดที่มีค่าระดับสีเทาเท่ากับ z_5 แต่ใช้ mask ขนาด 3 x 3 ได้ดังนี้

$$\nabla f \approx |(z_7 - 2z_8 - z_9) - (z_1 - 2z_2 - z_3)| \quad (2.19) \\ + |(z_3 - 2z_6 - z_9) - (z_1 - 2z_4 - z_7)|$$

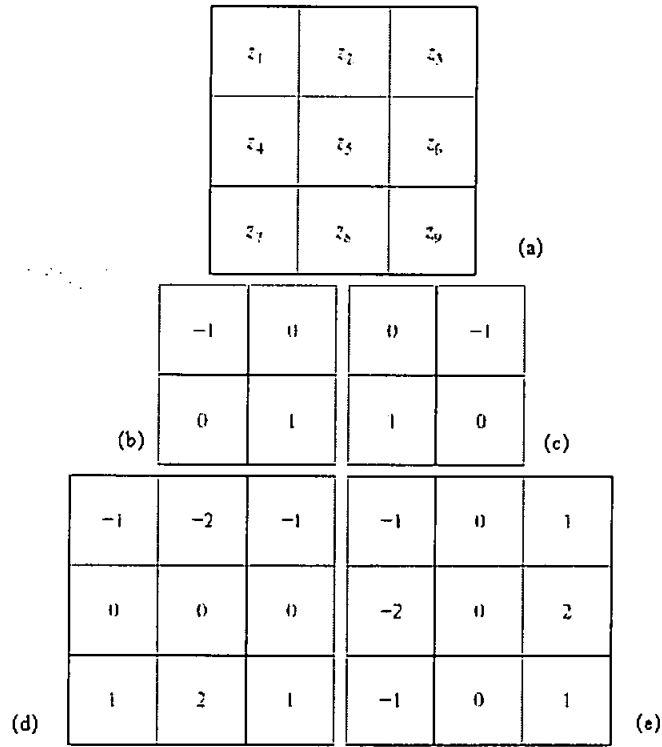
2.4.3.4 การใช้ Second Derivative ในการเพิ่มคุณภาพของภาพ

ลักษณะของการเพิ่มคุณภาพของภาพด้วย second-order derivatives เราสนใจในการทำตัวกรองแบบสมมาตร (isotropic filters) ซึ่งเป็นอิสระต่อทิศทางของความไม่ต่อเนื่องบนภาพ (rotation invariant) กระบวนการแบบสมมาตรที่ง่ายที่สุดได้แก่ Laplacian ซึ่งมีสมการดังนี้

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (2.20)$$

ซึ่งสามารถแสดงในรูปผลบวกของ 2 องค์ประกอบดังนี้

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y-1) + f(x, y+1)] \quad (2.21) \\ - 4f(x, y)$$



รูปที่ 2.8 (a) จุดของภาพที่มีขอบเขต 3 x3 pixel (ค่า z เป็นค่าระดับสีเทา)

(b)(c) Roberts cross-gradient operators

(d)(e) Sobel operators

2.4.3.5 การตรวจหาเส้นขอบด้วย กระจบวนกร โซเบล (Sobel operators)

กระจบวนกร โซเบล (Sobel Edge Operator) เป็นการตรวจหาขอบของภาพ ซึ่งข้อสันนิษฐานเบื้องต้น คือผิวของมนุษย์จะมีความราบเรียบต่อเนื่องกันไม่มีขอบเกิดขึ้นภายใน โดยการตรวจหาขอบของ Sobel จะทำโดยการนำ คอนโวลูชัน (convolution) ซึ่งจะนำพิกเซลที่อยู่ล้อมรอบ พิกเซลที่กำลังพิจารณาทำการคำนวณ โดยเราจะมีตัวเลขคงที่อยู่กลุ่มหนึ่ง เรียกว่า ค่าตัวร่วม (Mask Coefficient) เข้าไปคูณกับพิกเซลเหล่านั้น จากนั้นนำผลคูณแต่ละตัวมาทำการบวกเข้าด้วยกัน ผลลัพธ์สุดท้ายจะเก็บไว้ในตำแหน่งพิกเซลที่กำลังพิจารณาของภาพที่ผ่านการประมวลผลแล้ว

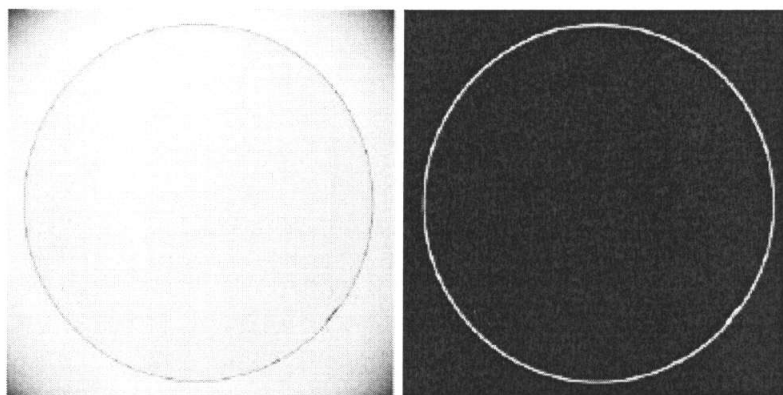
ในทิศทางแนวนอนใช้ Mask Coefficient ขนาด 3x3 ดังรูปที่.2 10 (d) คือ

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \text{ or } \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

และในทิศทางแนวตั้งใช้ mask coefficient ขนาด 3x3 ดังรูปที่.2 10 (e) คือ

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ or } \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

อย่างไรก็ตาม Sobel operators สามารถทำได้ด้วยสมการที่ 2.19 เช่นกัน โดยการให้ค่าน้ำหนักด้วยสองนั้นเพื่อให้ความสำคัญกับพิกเซลที่กึ่งกลางมากที่สุดโดยค่าน้ำหนักทั้งหมดต้องรวมกันแล้วเท่ากับ ศูนย์



(a) ตัวอย่างภาพต้นฉบับที่ต้องการหาเส้นขอบ
(b) ภาพที่ถูกตรวจหาเส้นขอบ (Sobel gradient)

2.4.4 กระบวนการทดสอบ

การตรวจหาภาพอนาจารภายในรูปภาพจะทำการระบุตำแหน่งของพิกเซลของผิวหนังและรวมกลุ่มไว้ด้วยกันเป็นกลุ่มของผิวหนัง โดยใช้รายละเอียดของสี, พื้นผิว และขอบ คุณลักษณะต่างๆถูกดึงมาจากกลุ่มของผิวหนังและใช้เป็นอินพุทเพื่อใส่เข้าไปในการจัดกลุ่ม

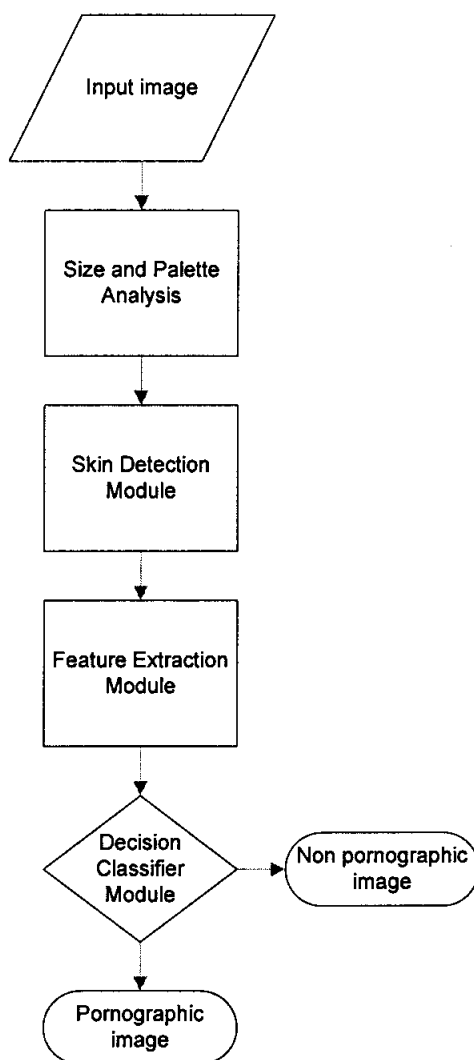
ความสัมพันธ์ระหว่างเปอร์เซ็นต์ของสีผิวกับรูปอนาจารมีค่อนข้างมาก ในขั้นตอนแรกต้องมีการแบ่งกลุ่มของสีผิว แต่การจำแนกสีของผิวหนังอาจมีความผิดพลาดได้ เนื่องจาก

- 1) ภาพมีคุณภาพต่ำ เช่น ภาพที่มี contrast ต่ำ
- 2) ภาพที่มีวัตถุที่มีสีใกล้เคียงกับสีผิวมนุษย์
- 3) ภาพผิวหนังที่มีสีผิวจางเนื่องจากการส่องสว่างและการสะท้อนของแสง

การตรวจสอบภาพอนาจารมี 4 ขั้นตอนหลักๆ คือ

1. Size and palette Analysis เป็นการกรองภาพแบบพื้นฐานที่สุด เช่น ภาพมีขนาดเล็กกว่าที่เราตั้งเอาไว้ และภาพที่มีจำนวนสีไม่มาก) น้อยกว่า 50 พิกเซล (เนื่องจากคุณสมบัติไม่เหมือนกับภาพอนาจาร

2. Skin Detection ในขั้นตอนนี้เราจะพิจารณาจากสี ,พื้นผิว ,และขอบ พิจารณาพื้นผิวที่เชื่อมต่อกัน
3. Feature Extraction การดึงคุณสมบัติ 6 อย่างจากข้อมูลภาพ
4. Decision Classifier เป็นการจัดประเภทของข้อมูลภาพ



รูปที่ 2.10 การทำงานของการตรวจสอบภาพอนาจาร

2.4.4.1 การกำจัดภาพขนาดเล็กกว่ากำหนด (Size and palette Analysis)

จากสมมติฐานภาพอนาจารควรมีขนาดใหญ่และใช้พื้นที่ส่วนใหญ่ของหน้าเอกสารเว็บ ดังนั้นเพื่อจำกัดการตรวจสอบและเพิ่มประสิทธิภาพในการตรวจสอบในขั้นต้น

ทำการตัดภาพที่มีขนาดเล็กกว่ากำหนด ออกแบบให้เท่ากับ 50x50 พิกเซล ภาพที่มีขนาดเล็กกว่าที่กำหนดจะไม่ถูกนำมาพิจารณา

2.4.4.2 การตรวจหาสีผิวมนุษย์ (skin detection module)

ในการตรวจหาสีผิวมนุษย์ที่ออกแบบมี 4 ขั้นตอนย่อยดังนี้

1. การตรวจหาพิกเซลในช่วงของสีผิว (Skin Tone Color Detection) จะพิจารณาจากค่า RGB ที่ถูกแปลงเป็นค่า YCbCr ว่าพิกเซลนั้นมีค่าอยู่ในช่วงสีผิวหรือไม่
2. การตรวจหาขอบเขตพื้นที่ของพิกเซลที่มีค่าอยู่ในช่วงสีผิว (Skin Region Expansion) จากการพิจารณาข้อที่ 1 อาจมีการรวมถึงพิกเซลข้างเคียงที่มีสีอยู่ในช่วงของสีผิว
3. การแบ่งพื้นที่ของสีผิวด้วยเส้นขอบ (Skin Region Segmentation) การตรวจหาขอบเขตจากข้อที่ 2 จะมีการพิจารณาร่วมกับกระบวนการตรวจหาขอบของ โซเบล (Sobel) พิกเซลที่อยู่บนขอบจะไม่ถูกตรวจจับว่าเป็นสีผิว การทำเช่นนี้เพื่อให้แน่ใจว่าเป็นพื้นที่ของผิวจริง
4. การกำจัดพื้นที่ของสีผิวที่มีขนาดเล็กกว่าที่กำหนดไว้ (Skin Blob Detection) กำหนดไว้ที่ 50 x50 พิกเซล

2.4.4.3 การตรวจสอบคุณสมบัติของภาพ (Feature Extraction Module)

ภาพที่ถูกรวบรวมว่าเป็นภาพอนาจาร จะถูกพิจารณาคุณสมบัติ 6 ข้อ ดังตารางที่ 2.1

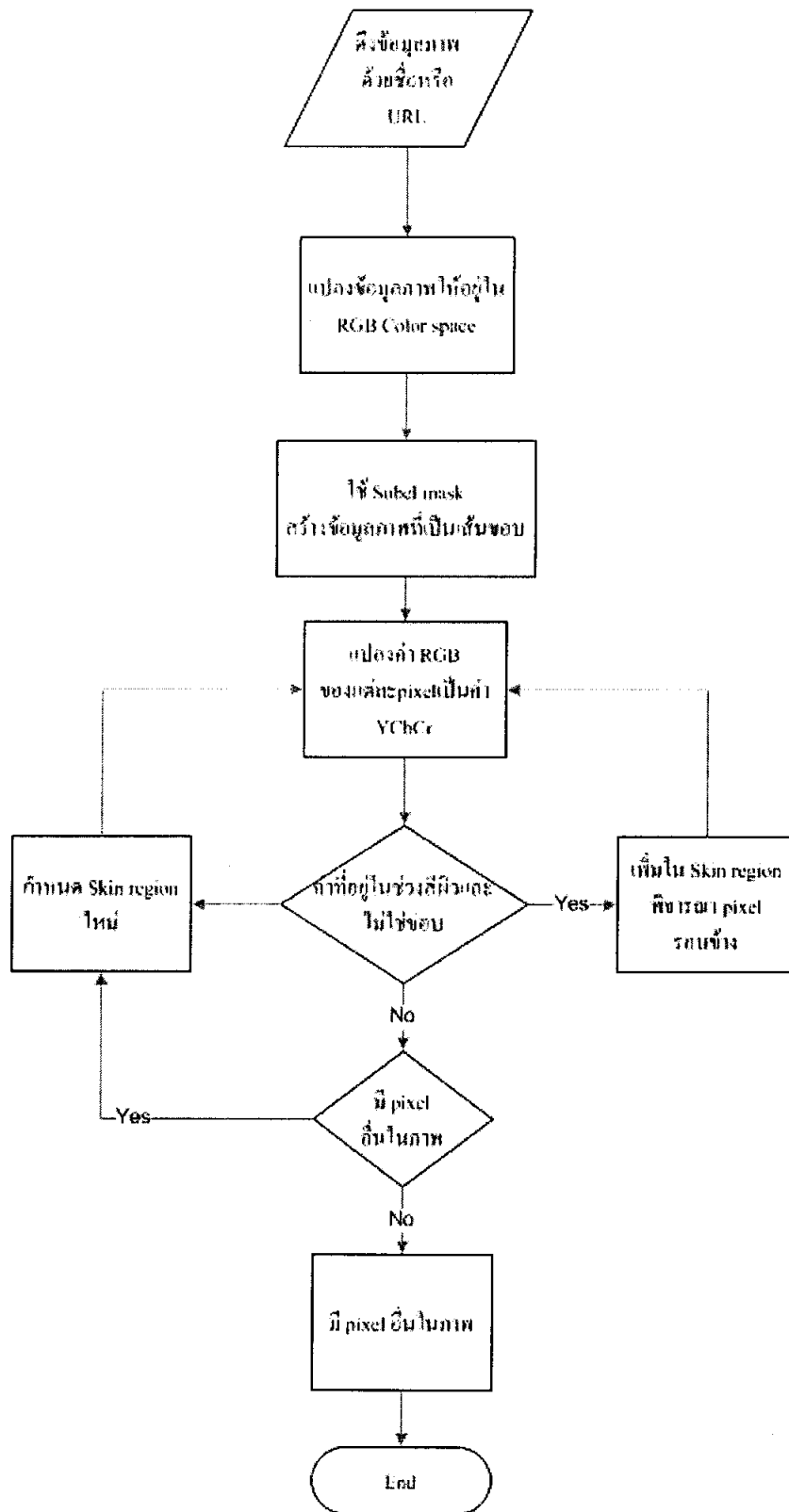
ตารางที่ 2.1 คุณสมบัติ 6 ประการที่ได้จากภาพ

ลำดับ	คุณสมบัติ	รายละเอียด
1	เปอร์เซ็นต์ของสีผิวในภาพ	อัตราของพิกเซลที่มีค่าอยู่ในช่วงของสีผิวทั้งหมดเทียบขนาดของภาพ
2	เปอร์เซ็นต์ของพื้นที่สีผิวที่มีขนาดใหญ่ที่สุด	อัตราของพิกเซลในพื้นที่ที่มีสีผิวมากที่สุดเทียบกับขนาดของภาพ
3	จำนวนชิ้นส่วนของพื้นที่ผิว	จำนวนของพื้นที่ที่มีสีผิวในภาพ
4	อัตราความสูงของพื้นที่ผิว	ขนาดความสูงของพื้นที่ที่มีสีผิวขนาดใหญ่ที่สุดต่อขนาดความสูงของภาพ
5	อัตราความกว้างของพื้นที่ผิว	ขนาดความกว้างของพื้นที่ที่มีสีผิวขนาดใหญ่ที่สุดต่อขนาดความสูงของภาพ
6	อัตราส่วนของพื้นที่ผิวขนาดใหญ่เทียบกับจำนวนพื้นที่ผิวทั้งหมด	อัตราส่วนของพื้นที่ที่มีสีผิวขนาดใหญ่ที่สุดในภาพเทียบกับขนาดพื้นที่ที่มีสีผิวทั้งหมดในภาพ

คุณสมบัติทั้ง 6 ข้อสามารถใช้จำแนกภาพอนาจารได้ โดยภาพที่ไม่มีกลุ่มของสีผิวจะถูกมองว่าไม่เป็นภาพอนาจาร ซึ่งไม่มีคุณสมบัติข้อที่ 1 จากสมมติฐานภาพอนาจารจะต้องประกอบด้วยกลุ่มของสีผิวขนาดใหญ่ต่อเนื่องกัน ซึ่งสามารถพิจารณาได้จากคุณสมบัติข้อที่ 2 และ

3 ที่แสดงถึงอัตราส่วนของพื้นที่ผิวที่มีขนาดใหญ่ที่สุดและจำนวนชิ้นส่วนของพื้นที่ผิว ภาพที่มีจำนวนของพื้นที่ผิวมากอาจจะไม่ใช่ภาพอนาจาร ความสัมพันธ์ระหว่างขนาดและรูปร่างของกลุ่มสี่เหลี่ยม ถ้าเป็นภาพอนาจารจะมีกลุ่มสี่เหลี่ยมขนาดใหญ่ คุณสมบัติข้อ 5 และ 6 เป็นความสัมพันธ์ของอัตราส่วนของความกว้างและความสูงของกลุ่มสี่เหลี่ยมต่อความกว้างและความสูงของภาพ มีแนวโน้มว่ากลุ่มสี่เหลี่ยมจะครอบคลุมพื้นที่ส่วนใหญ่ของภาพ

โครงการนี้ได้จัดทำโปรแกรมตรวจสอบภาพอนาจารที่สามารถวิเคราะห์คุณสมบัติของภาพได้ โดยมีขั้นตอนการทำงานดังรูปที่ 2.11

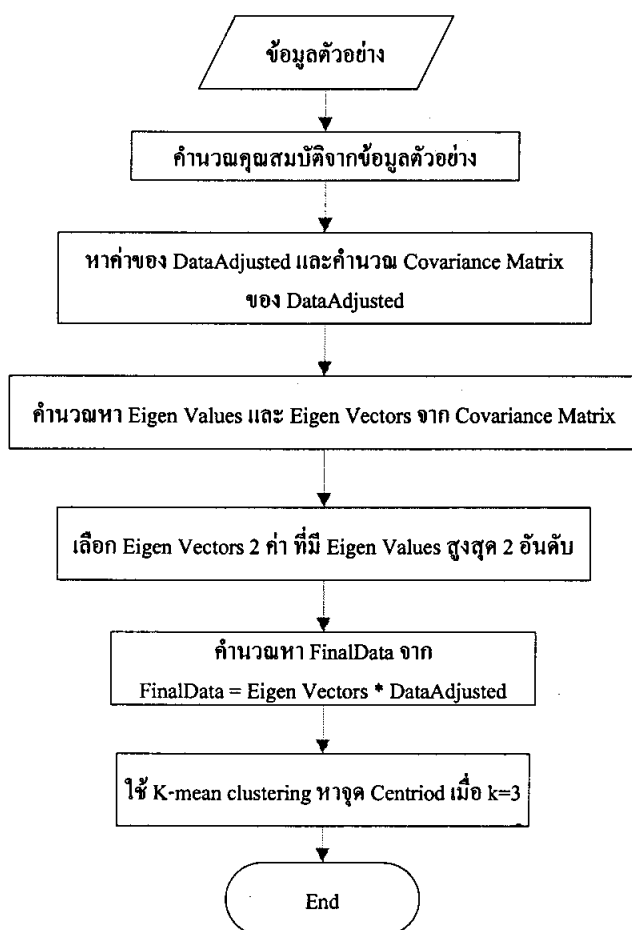


รูปที่ 2.11 การทำงานของโปรแกรมการตรวจสอบภาพอนาจาร.

2.4.4.4 การจัดประเภทของภาพด้วยคุณสมบัติ (Decision Classifier)

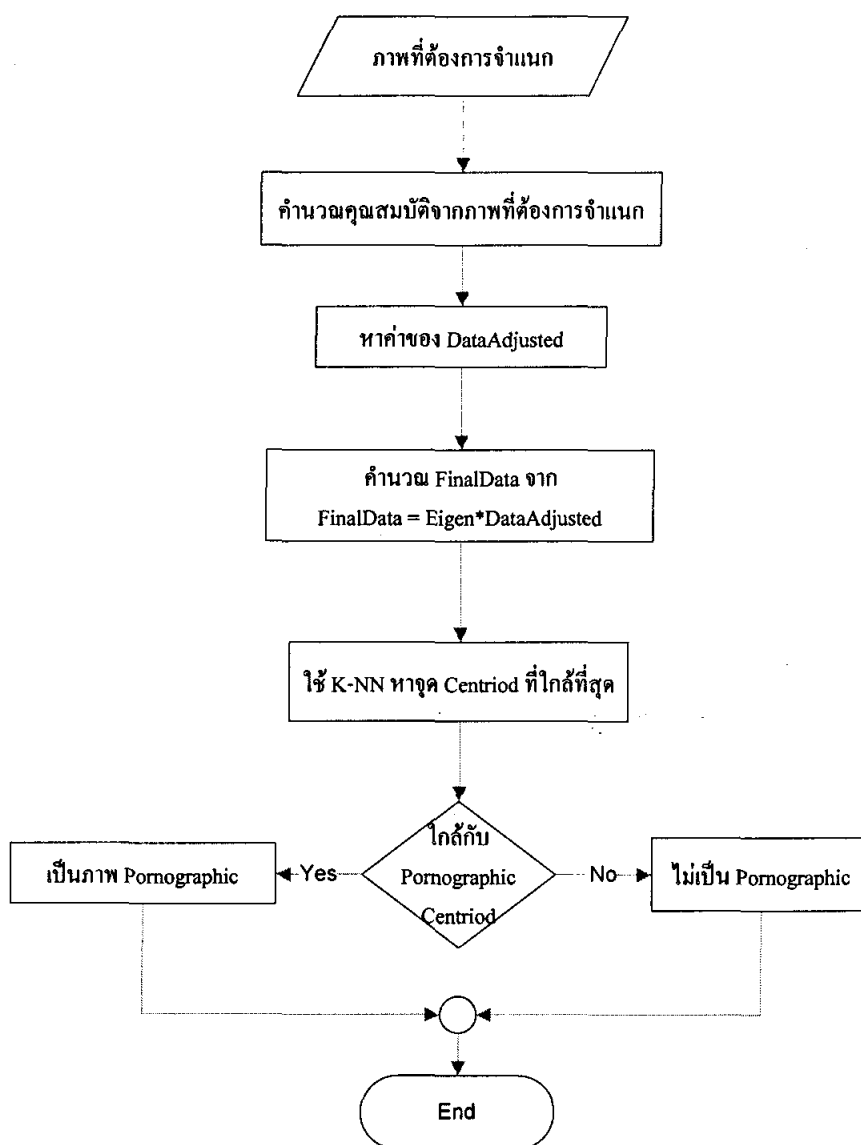
เมื่อพิจารณาคูสมบัติของภาพทั้ง 6 ข้อ เราสามารถจำแนกประเภทของภาพได้ 3 กลุ่ม ได้แก่ ภาพอนาจาร , ภาพที่ไม่ใช่ภาพอนาจารแต่มีส่วนประกอบของสีผิว และภาพที่ไม่มีส่วนประกอบของสีผิว เพื่อให้สามารถพิจารณาคูสมบัติที่มีความสำคัญมากที่สุดจำเป็นต้องทำการรวมคุณสมบัติและลดคุณสมบัติบางประการโดยไม่ทำให้ข้อมูลสูญหายมากเกินไปด้วยหลักการวิเคราะห์ส่วนประกอบที่ต้องทำการคำนวณหาไอแกนเวกเตอร์และไอแกนแวลูส์ของชุดข้อมูลตัวอย่าง แล้วทำการหาจุดเซ็นทรอยด์ 3 จุดที่แสดงถึงข้อมูลภาพ 3 ประเภทด้วยหลักการของ K-mean Clustering และเมื่อต้องการจำแนกภาพสามารถทำได้โดยการคำนวณหาระยะทางที่ใกล้ที่สุดเทียบกับจุดเซ็นทรอยด์ด้วยหลักการของ K-Nearest Neighbor (KNN)

โครงการนี้ได้จัดทำโปรแกรมจัดประเภทของภาพด้วยคุณสมบัติ ที่สามารถจำแนกประเภทของภาพด้วยคุณสมบัติของภาพได้ โดยมีขั้นตอนการคำนวณหาไอแกนเวกเตอร์และไอแกนแวลูส์ของชุดข้อมูลตัวอย่าง แล้วทำการหาจุดเซ็นทรอยด์ 3 จุดที่แสดงถึงข้อมูลภาพ 3 ประเภทด้วยหลักการของ K-mean Clustering ดังรูปที่ 2.12



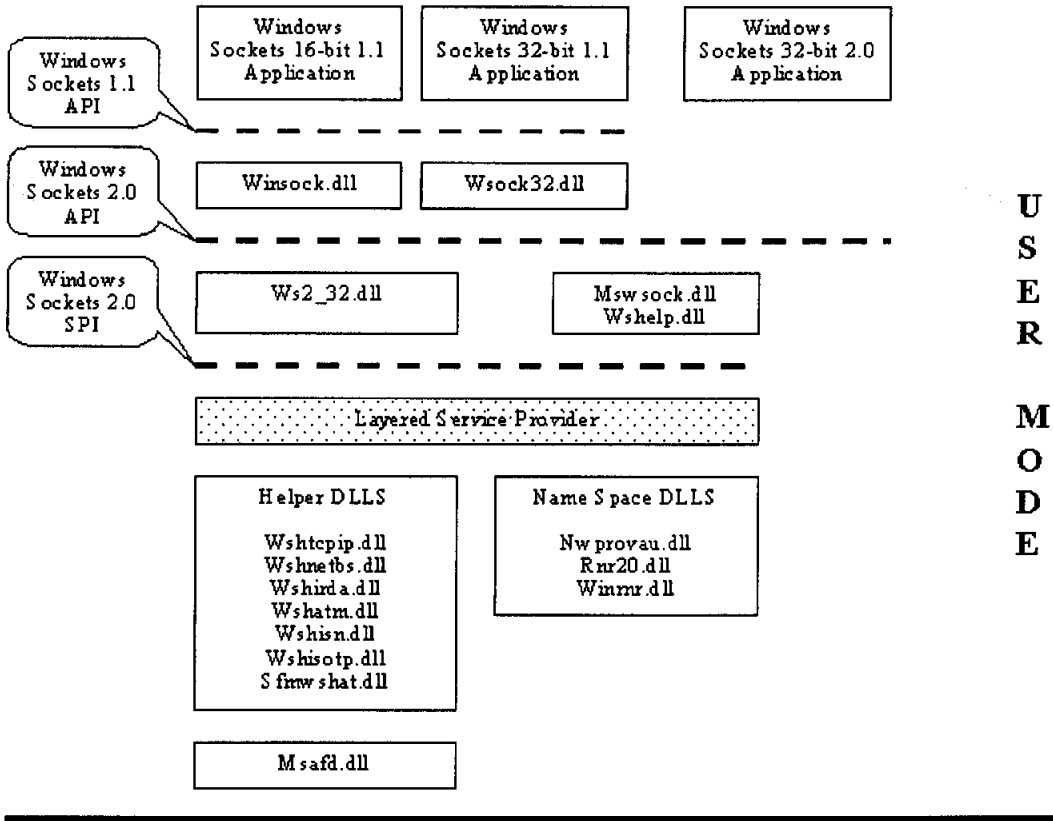
รูปที่ 2.12 การทำงานของการคำนวณหาไอแกนเวกเตอร์และไอแกนแวลูส์ของชุดข้อมูลตัวอย่างและ การหาจุดเซ็นทรอยด์ 3 จุด

เมื่อคำนวณหาไอแกนเวกเตอร์และไอแกนแวลูส์ของชุดข้อมูลตัวอย่าง แล้วทำการหาจุดเซ็นทรอยด์ 3 จุด ได้จะกระบวนการข้างต้นแล้ว เมื่อต้องการจำแนกภาพสามารถทำได้โดยการคำนวณหาระยะทางที่ใกล้ที่สุดเทียบกับจุดเซ็นทรอยด์ด้วยหลักการของ K-Nearest Neighbor (KNN) มีขั้นตอนการทำงานดังรูปที่ 2.15

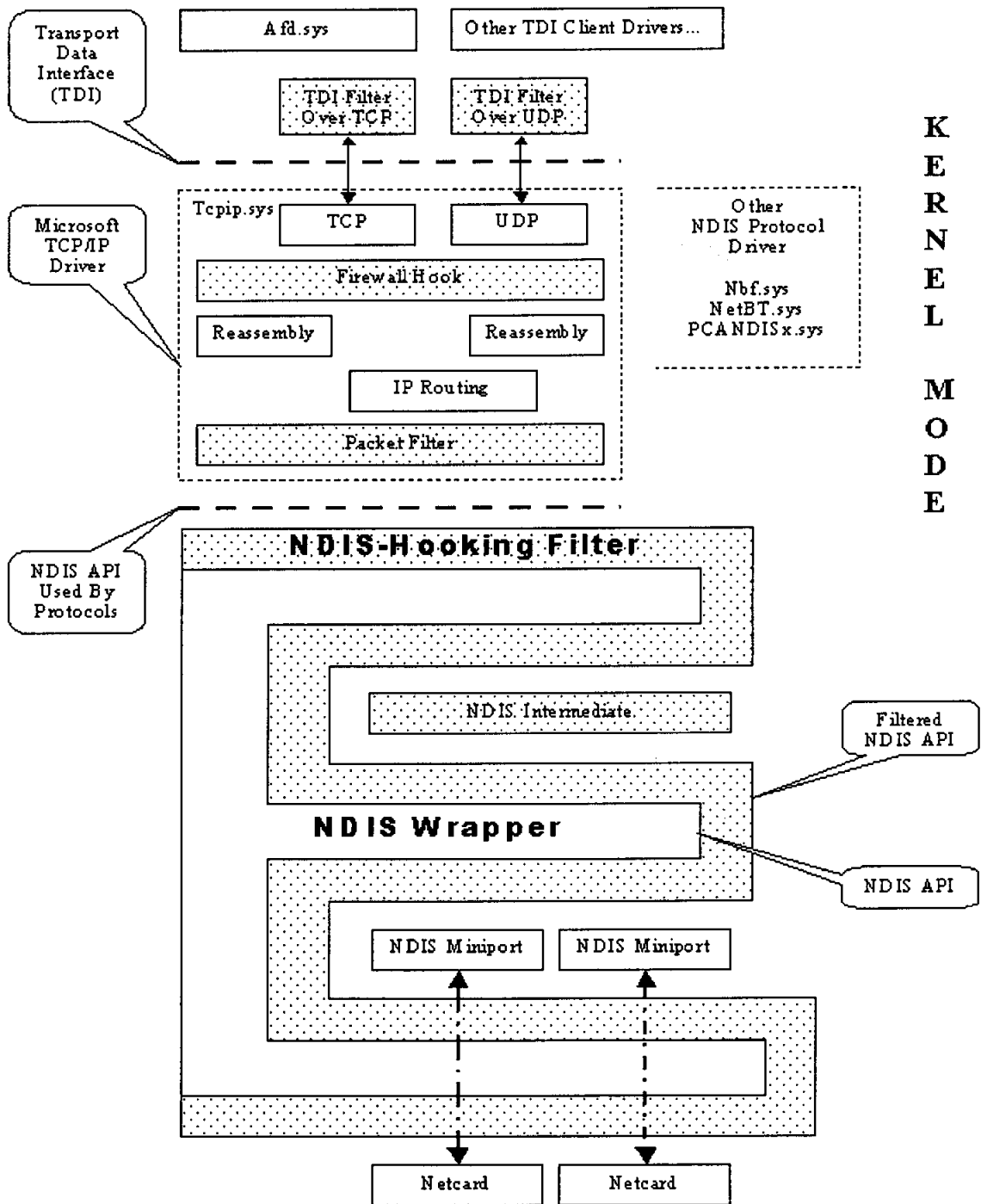


รูปที่ 2.13 การจำแนกภาพ

2.5 Packet Filtering API



รูปที่ 2.14 Network Architecture Diagram (user mode)



รูปที่ 2.15 Network Architecture Diagram (kernel mode)

ในระบบ Microsoft Windows Platform นั้นมีหลายวิธีที่จะใช้เพื่อกรองข้อมูล และแพ็คเก็ตเกิดของระบบเครือข่าย วิธีหนึ่งก็คือใช้ Packet filtering API ซึ่งเป็นเทคนิคที่ใช้ในชั้นของ User mode ในสถาปัตยกรรมระบบเครือข่าย (Network Architecture) ซึ่งมี 2 ชั้นคือ User mode และ Kernel mode

Packet filtering (ระบบกรองแพ็กเก็ต) ช่วยนักพัฒนาในการสร้างและจัดการกรองข้อมูลที่ผ่านมาเข้า และออกสำหรับ IP packet (Internet Protocol packets) แต่ละ IP adapter interface สามารถทำให้เชื่อมโยงกับตัวกรองได้มากกว่าหนึ่งตัว ตัวกรอง หรือ filter นั้นสามารถให้ pass/drop (ผ่าน/ทิ้ง) packet ได้โดยคำนึงถึงที่อยู่ของ ต้นทาง (source) และปลายทาง (destination) ได้ รวมถึง address mask, port และ โพรโตคอลที่กำหนดไว้

ใน Windows 2000 และรุ่นต่อมา มา รวมถึง Windows XP และ Windows 2003 ไมโครซอฟท์ได้รวม API นี้ไว้เพื่อเพิ่มคุณสมบัติการกรองแพ็กเก็ตเพื่อให้เรานำไปใช้ในโปรแกรมของเราได้ ตัว Packet filtering API นี้ได้ถูกนิยามไว้ใน fltdefs.h ซึ่งอยู่ในชุดพัฒนาของ ไมโครซอฟท์ที่เรียกว่า Microsoft Platform SDK แต่ฟังก์ชันในกลุ่ม Packet filtering API นี้ต้องการสิทธิของผู้ดูแลระบบในการทำงาน ยกเว้นแต่ฟังก์ชัน PfGetInterfaceStatistics

ฟังก์ชันที่สำคัญ มีดังนี้

- **PfCreateInterface** ฟังก์ชันใช้สร้าง Filter interface ใหม่ และใช้ interface ที่สร้างนี้ในการควบคุม เพิ่ม และลบตัวกรองเข้าและออกจากอะแดปเตอร์

- **PfBindInterfaceToIPAddress** ฟังก์ชันเพื่อเชื่อมโยง Interface กับ IP ที่กำหนด

- **PfAddFiltersToInterface** ฟังก์ชันเพื่อเพิ่มตัวกรองเฉพาะให้กับ Interface ที่ระบุ

- **PfAddGlobalFilterToInterface** ฟังก์ชันเพิ่มค่าการกรองให้กับตัวกรองทั้งหมดของ Interface ที่ระบุ

- **PfGetInterfaceStatistics** ฟังก์ชันเพื่อดึงค่าข้อมูลสถิติจาก Interface ที่ระบุ ฟังก์ชันนี้ยังสามารถเลือกดึงข้อมูลจากตัวกรองที่เชื่อมโยงกับ Interface ได้

- **PfRemoveGlobalFilterFromInterface** ฟังก์ชันใช้ลบค่ากำหนดการกรองทั้งหมดออกจาก Interface

- **PfRemoveFiltersFromInterface** ฟังก์ชันลบเฉพาะตัวกรองที่ระบุออกจาก Interface

- **PfUnBindInterface** ฟังก์ชันใช้ยกเลิกการเชื่อม IP กับ Interface

- **PfDeleteInterface** ฟังก์ชันลบทั้ง Interface ที่ได้สร้างไว้แล้ว

ตัวอย่างวิธีการนำไปใช้มีขั้นตอนดังนี้

- ต้องรู้หมายเลข IP ของ local interface ที่ต้องการจะเพิ่ม filter ให้ วิธีการอย่างง่ายคือ สร้าง interface ขึ้นมาและ bind เข้ากับหมายเลขที่อยู่ IP

- เพิ่มกฎของการกรองที่ต้องการ

- เมื่อเสร็จงานแล้ว ต้อง unbind ตัว interface กับ local IP address และลบ interface นั้นทิ้ง

2.5.1 คลาสใน .NET

TxFILTERController คลาสพื้นฐาน ใช้เพิ่มและลบ filter ออกจาก interface ซึ่งสามารถทำได้โดยใช้คลาสนี้เท่านั้น ซึ่งคลาสนี้จะสร้าง interface เท่าที่ต้องการ โดยไม่ต้องให้ผู้สร้างขึ้นมาเอง ภายในคลาสนี้มีเมธอด ดังนี้

- int AddFilter(IPAddress *ip, TxIpFilter *filter) เพิ่ม filter ให้กับ local IP address ถ้าไม่มีข้อผิดพลาด คืนค่า 0 เมธอดนี้ overloaded เพื่อผ่านกฎของ filter โดยปราศจากการสร้างอ็อบเจ็กต์ TxIpFilter เพื่อเกิดข้อผิดพลาดที่ตรงกับกฎนั้นก็จะถูกตัดทิ้ง

- int AddGlobalFilter(IPAddress *ip, int globalFilter) เพิ่มกฎ filter แบบ global ให้แก่ local IP address ถ้าไม่มีข้อผิดพลาด คืนค่า 0

- int RemoveFilter(IPAddress *ip, TxIpFilter *filter) ลบ filter จาก interface ถ้าไม่มีข้อผิดพลาด คืนค่า 0 เมธอดนี้ overloaded เพื่อผ่านกฎของ filter โดยปราศจากการสร้างอ็อบเจ็กต์ TxIpFilter

- int RemoveGlobalFilter(IPAddress *ip, int globalFilter) ลบกฎ filter แบบ global จาก local IP address ถ้าไม่มีข้อผิดพลาด คืนค่า 0

- CloseController() ลบ และ Unbind interface ที่สร้างไว้ทั้งหมด

TxIpFilter คล้ายกับ PF_FILTER_DESCRIPTOR ใน packet filtering API เพื่อใช้นิยามกฎของการกรอง สามารถเพิ่ม TxIpFilters ได้เท่าที่ต้องการใน TxFILTERController

2.6 IPHelper Library

ไลบรารีที่มีชื่อว่า IPHelper นี้ได้ใช้หลักการของ P/Invoke ใน .NET framework ซึ่งภายในจะมีไฟล์ IPHlpAPI32.cs ที่มี การประกาศ struct ของ function จาก IpHelperAPI.dll โดยที่ใช้ Attribute มาตราฐานจาก namespace ที่มีชื่อว่า System.Runtime.InteropServices

จุดประสงค์หลักของ Library นี้เขียนขึ้นมาเพื่อเฝ้าดูการเชื่อมต่อแบบ UCP และ TCP ที่มีการเชื่อมต่ออยู่ในขณะนั้นคล้ายๆกับ โปรแกรม Netstat ใน IpHelper Library นั้นประกอบด้วย IpHelperApi.dll โดยที่ภายในมี 4 ฟังก์ชันพื้นฐาน คือฟังก์ชันพื้นฐานสำหรับดูสถิติการเชื่อมต่อทาง UCP /TDP คือฟังก์ชัน GetTcpStats(), GetUdpStats() และฟังก์ชันสำหรับการ get the tables ของ UDP/TCP GetUdpConnexions(), GetTcpConnexions()

```
public void GetTcpStats()
{
    TcpStats = new MIB_TCPSTATS();
    IPHlpAPI32Wrapper.GetTcpStatistics(ref TcpStats);
}
```

ฟังก์ชัน GetTcpStats()

```

public void GetTcpConnections()
{
    int AF_INET = 2; // IP_v4
    int res = IPHlpAPI32Wrapper.GetExtendedTcpTable(buffer, out buffSize, true,
    AF_INET, TCP_TABLE_CLASS.TCP_TABLE_OWNER_PID_ALL, 0);
    if (res != Utils.NO_ERROR
    {
        buffer = new byte[buffSize];
        res = IPHlpAPI32Wrapper.GetExtendedTcpTable(buffer, out buffSize, true,
    AF_INET, TCP_TABLE_CLASS.TCP_TABLE_OWNER_PID_ALL, 0);
        if (res != Utils.NO_ERROR)
        {
            return;
        }
    }
    int nOffset = 0;
    int NumEntries = Convert.ToInt32(buffer[nOffset]);
    nOffset += 4;
    for (int i = 0; i < NumEntries; i++)
    {
        TCPUDPConnection row = new TCPUDPConnection();
        int st = Convert.ToInt32(buffer[nOffset]);
        row.iState = st;
        nOffset += 4;
        row.Protocol = Protocol.TCP;
        row.Local = Utils.BufferToIPEndPoint(buffer, ref nOffset, false);
        row.Remote = Utils.BufferToIPEndPoint(buffer, ref nOffset, true);
        row.PID = Utils.BufferToInt(buffer, ref nOffset);
        this.Add(row);
    }
}

```

ฟังก์ชัน GetTcpConnections()

```

public void GetUdpConnections()
{
    int AF_INET = 2; // IP_v4
    int res = IPHlpAPI32Wrapper.GetExtendedUdpTable(buffer, out buffSize,
true, AF_INET, UDP_TABLE_CLASS.UDP_TABLE_OWNER_PID, 0);
    if (res != Utils.NO_ERROR)
    {
        buffer = new byte[buffSize];
        res = IPHlpAPI32Wrapper.GetExtendedUdpTable(buffer, out
buffSize, true, AF_INET, UDP_TABLE_CLASS.UDP_TABLE_OWNER_PID, 0);
        if (res != Utils.NO_ERROR)
        {
            return;
        }
    }
    int nOffset = 0;
    int NumEntries = Convert.ToInt32(buffer[nOffset]);
    nOffset += 4;
    for (int i = 0; i < NumEntries; i++)
    {
        TCPUDPConnection row = new TCPUDPConnection();
        row.Protocol = Protocol.UDP;
        row.Local = Utils.BufferToIPEndPoint(buffer, ref nOffset, false);
        row.PID = Utils.BufferToInt(buffer, ref nOffset);
        this.Add(row);
    }
}

```

ฟังก์ชัน GetUdpConnections()

ฟังก์ชัน `GetTcpConnexion()` และ `GetUdpConnexion()` นั้นมีประโยชน์ก็จริงแต่ไม่สามารถที่จะใช้ควบคุมการติดต่อได้ ซึ่งหมายความว่าเราจะไม่สามารถปิดการเชื่อมต่อหรือรับข้อมูลที่โปรแกรมได้รับจากการเชื่อมต่อได้ และเพื่อที่จะทำให้มีความสามารถเหล่านี้ เราจึงจำเป็นต้องทราบว่า มี Process ใดที่ได้มีการเชื่อมต่อทาง Internet บ้าง

ในการแก้ปัญหานี้จึงได้มีการพัฒนาฟังก์ชัน ที่ทำหน้าที่คล้ายกับฟังก์ชัน `GetUdpConnexions()` , `GetTcpConnexions()` ทุกประการ ต่างกันเพียงที่สามารถทราบ processID ในโปรเซสที่ทำการเชื่อมต่อ ได้ด้วย ซึ่งฟังก์ชันนี้ใน Win 32 API มีชื่อว่า `AllocateAndGetTcpExTableFromStack` และ `AllocateAndGetUdpExTableFromStack` แต่มีข้อจำกัดคือ สามารถใช้ได้ ใน Window XP เท่านั้น.

```

public void GetExTcpConnections()
{
    // the size of the MIB_EXTCPROW struct = 6*DWORD
    int rowSize = 24;
    int BufferSize = 100000;
    // allocate a dumb memory space in order to retrieve nb of connexion
    IntPtr lpTable = Marshal.AllocHGlobal(BufferSize);
    //getting infos
    int res = IPHlpAPI32Wrapper.AllocateAndGetTcpExTableFromStack(
        ref lpTable, true, IPHlpAPI32Wrapper.GetProcessHeap(),0,2);
    if(res!=NO_ERROR)
    {
        Debug.WriteLine(
            "Erreur : "+IPHlpAPI32Wrapper.GetAPIErrorMessageDescription(res)
            +" "+res);
        return; // Error. You should handle it
    }
    int CurrentIndex = 0;
    int NumEntries= (int)Marshal.ReadIntPtr(lpTable);
    lpTable = IntPtr.Zero;
    Marshal.FreeHGlobal(lpTable);
}

```

ฟังก์ชัน **GetTcpExTableFromStack ()**

ที่มีการเรียกใช้งาน **AllocateAndGetTcpExTableFromStack()**

2.7 Asynchronous Programming

การเขียนโปรแกรมแบบ Asynchronous Programming ทำให้การเชื่อมต่อจะไม่ถูกปิดกั้น โดยตัวแอปพลิเคชันระหว่างที่รอคอยกระบวนการทำงานของระบบเครือข่ายเสร็จสมบูรณ์ แทนที่จะทำการปิดกั้นเซิร์ฟเวอร์ที่ทำงานอยู่ปัจจุบัน ซึ่งจะใช้รูปแบบของ .NET asynchronous ในกระบวนการติดต่อในระบบเครือข่ายบน thread อื่นๆ ระหว่างที่โปรแกรมอยู่ในระหว่างการทำงานใน thread แรก เมื่อ thread ที่ 2 ได้ทำงานเสร็จเรียบร้อยแล้ว มันจะส่ง event object ไปที่เซิร์ฟเวอร์ซึ่งเป็นการบอกสถานะของ task นั้นๆ

เหตุการณ์ (Event) สามารถที่จะส่งสัญญาณออกไปในกรณีที่เมธอดนั้นกำลังรอคอยที่จะตอบกลับ หรือไม่มีการส่งสัญญาณตอบกลับ ถ้าเหตุการณ์ไม่ได้ถูกส่งสัญญาณเมธอดที่รอคอยอยู่จะถูกปิดกั้น (block) จนกว่าสัญญาณถูกส่งมา

เราจะมีชนิดของเหตุการณ์อยู่ 2 ชนิด นั่นคือ

- Automatic event มันสามารถทำให้ตัวมันเองกลับไปสถานะไม่ส่งสัญญาณ (non-signaled state) เมื่อไรก็ตามที่มันถูกส่งสัญญาณออกไป

- Manual reset event มันจะส่งสัญญาณจนกระทั่งมันได้ถูกสั่งให้รับการ reset ไปยังสถานะไม่ส่งสัญญาณ (non-signaled state) .NET จะมี ManualResetEvent class ซึ่งเราสามารถเรียกใช้ในโปรแกรมได้ หากเราต้องการที่จะดำเนินการต่อ ให้เราทำการตั้งค่า ManualResetEvent object ไปที่สถานะส่งสัญญาณ (signaled state) จะทำให้มันหยุดปิดกั้น (Block) ทุกๆเมธอดที่ทำการรอคอยอยู่ และโปรแกรมสามารถดำเนินการต่อได้ เราจะใช้ object นี้เพื่ออำนวยความสะดวกของการเขียนโปรแกรม asynchronous programming ได้

รูปแบบการเขียนโปรแกรมแบบ Asynchronous programming เหมาะสมสำหรับโปรแกรมที่ไม่สามารถรอคอยเพื่อการทำงานบนเครือข่ายได้สมบูรณ์ก่อนที่จะทำการต่อไป

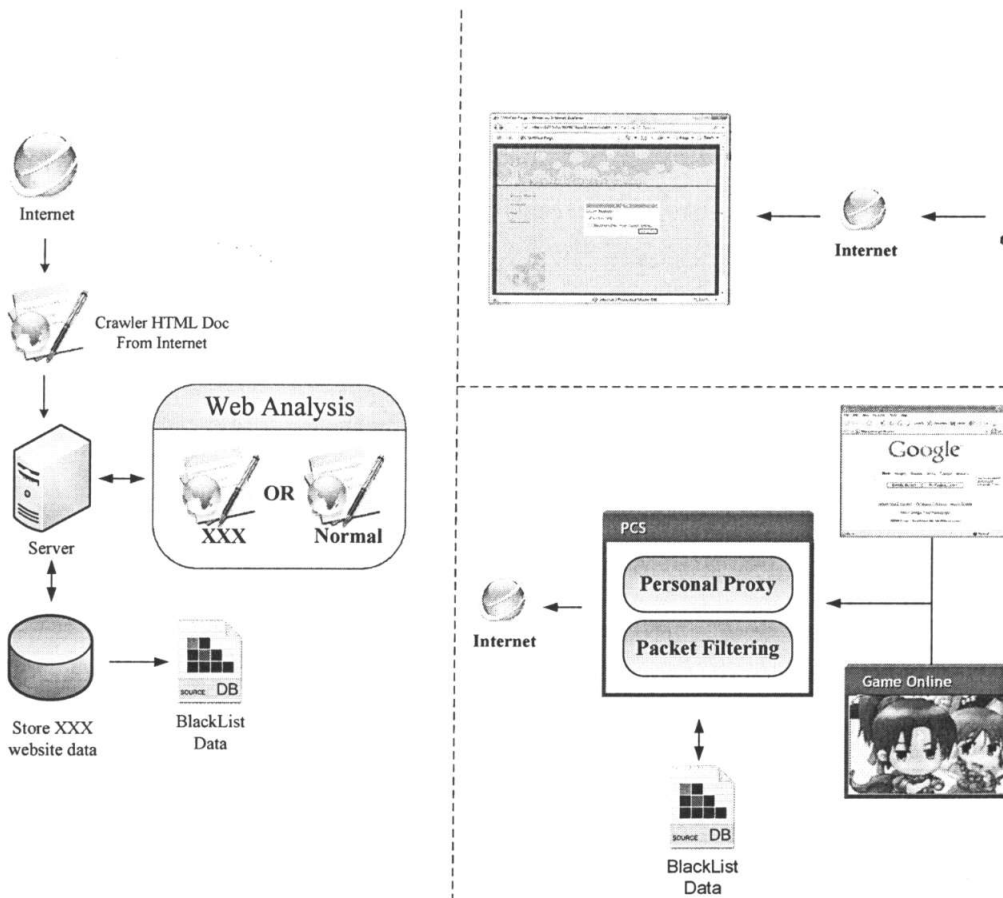
รูปแบบการเขียนโปรแกรมแบบ Asynchronous programming จะใช้เมธอดที่เรียกว่า callback ในการส่งผลของการทำงาน .NET Framework จะมี method ที่จัดการเกี่ยวกับการกำหนดค่าเริ่มต้น แต่ต้องการเมธอด callback ในการจบการทำงาน เช่น เมธอด Connect () ใช้ BeginConnect () ในการเริ่มการติดต่อไปที่อุปกรณ์เครือข่ายและในการทำงานร่วมกับเมธอด callback เพื่อให้เกิดการเชื่อมต่อที่สมบูรณ์

บทที่ 3

โครงสร้างและการออกแบบ

3.1 โครงสร้างพื้นฐานของระบบควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครอง(Parental Control System)

ระบบควบคุมการใช้งานอินเทอร์เน็ตโดยผู้ปกครอง (Parental Control System) นั้นได้แบ่งระบบออกเป็น 2 ส่วนใหญ่ๆ โดยแบ่งเป็นในส่วนของ ไคลเอนต์ (Client) ซึ่งจะถูกติดตั้งลงยังเครื่องของผู้ใช้งานเพื่อทำหน้าที่ในการป้องกันการเข้าใช้งานเว็บไซต์ (Website) และ โปรแกรมที่ไม่เหมาะสม ส่วนที่ 2 คือส่วนของเซิร์ฟเวอร์ (Server) โดยทำหน้าที่สำรวจหน้าเว็บเพจ (Webpage) ต่างๆแล้วทำการวิเคราะห์ว่าหน้านั้นมีข้อมูลที่ไม่เหมาะสมหรือไม่เพื่อที่จะได้ส่งข้อมูลให้ส่วนของไคลเอนต์ (Client) และทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ (Web Server) เพื่อให้ผู้ปกครองสามารถดูข้อมูลการใช้งานได้ทุกที่ทุกเวลา



รูปที่ 3.1 โครงสร้างและสถาปัตยกรรมของโปรแกรม Parental Control System

จากรูป การทำงาน โดยร่วมของระบบนั้นขอแบ่งออกเป็น 2 ส่วน โดยในส่วนไคลเอ็นต์ (Client) นั้นเมื่อมีการเรียกใช้งานเว็บเพจ (Webpage) หรือโปรแกรมที่มีการเชื่อมต่อออกสู่อินเทอร์เน็ต ในกรณีเรียกใช้งานเว็บเพจ (Webpage) ระบบจะทำการตรวจสอบว่าที่อยู่ หรือ URL ของหน้าเว็บเพจ (Webpage) นั้นตรงกับที่มีอยู่ในฐานข้อมูล (Blacklist) ของระบบหรือไม่ หากตรงกับที่มีอยู่ผู้ใช้ก็จะไม่สามารถเข้าถึง หน้าเว็บเพจ (Webpage) นั้นๆ ได้พร้อมทั้งแจ้งเตือนขึ้นมาว่า หน้าเว็บเพจ (Webpage) ที่กำลังจะเข้าชมนั้นเข้าข่ายไม่เหมาะสม ในส่วนของโปรแกรมก็จะมีลักษณะเช่นเดียวกัน โดยอ้างอิงจาก ชื่อของโพสเชส (Process) หากตรงกับที่มีอยู่โปรแกรมนั้นก็ จะไม่สามารถเชื่อมต่อสู่อินเทอร์เน็ตได้

ส่วนของเซิร์ฟเวอร์ (Server) นั้นการทำงานจะเริ่มจากการปล่อยคราเวลอร์ (Crawler) ออกไปเพื่อทำการเก็บหน้าเว็บเพจ (Webpage) โดยใช้ลักษณะการคลานตามลิงค์ต่างๆ ของหน้าเว็บเพจ (Webpage) ไปเรื่อยๆ เมื่อเก็บมาแล้วระบบจะทำการนำว่าวิเคราะห์ว่าเป็นหน้าเว็บเพจ (Webpage) ที่ไม่เหมาะสมหรือไม่จากข้อความรูปภาพและองค์ประกอบอื่นๆที่อยู่ในนั้นหาก

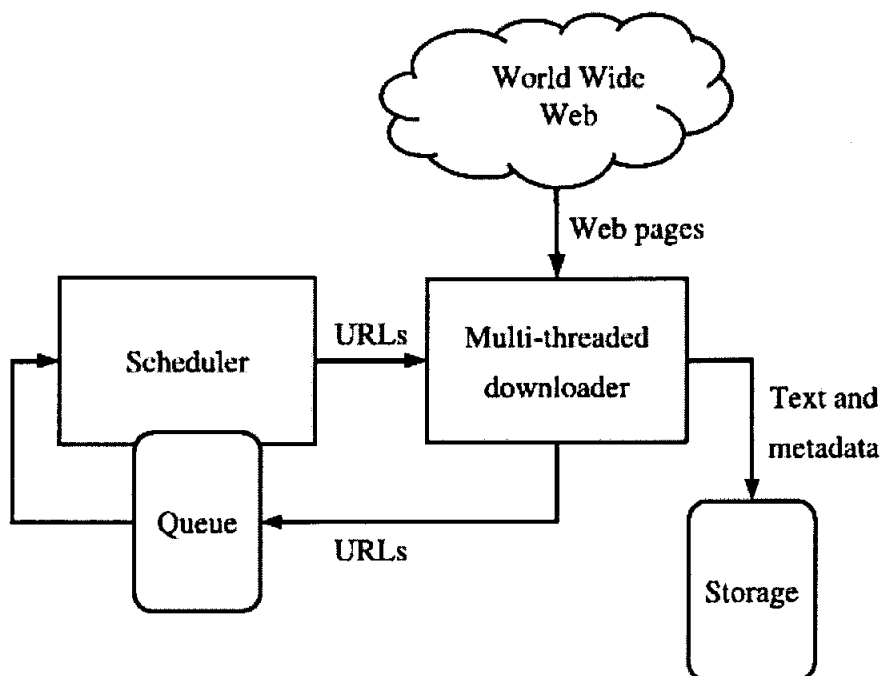
วิเคราะห์ออกมาแล้วว่าไม่เหมาะสมเซิร์ฟเวอร์ (Server) ก็จะทำการส่งรายชื่อหน้าเว็บเพจ (Webpage) นั้นไปยังไคลเอ็นต์ (Client) ทุกๆตัวเพื่อทำการปรับปรุงข้อมูล

นอกจากนี้ระบบยังมีในส่วนของเว็บเซิร์ฟเวอร์ (Web Server) เพื่อทำหน้าที่ให้ผู้ปกครองสามารถตรวจสอบการใช้งานคอมพิวเตอร์ได้ทุกที่ทุกเวลาโดยผู้ปกครองสามารถเข้าเว็บไซต์ (Website) เพื่อเชื่อมต่อไปยังเครื่องไคลเอ็นต์ (Client) โดยผู้ปกครองสามารถตรวจสอบการใช้งานต่างๆได้เช่น มีการใช้งานโปรแกรมใดบ้าง มีการเข้าใช้งานหน้าเว็บเพจ (Webpage) ใดบ้างพร้อมสามารถเรียกดูภาพหน้าจอการใช้งาน ณ ขนาดนั้นร่วมไปถึงการแก้ไขฐานข้อมูล หรือ Blacklist ได้

อย่างสะดวก

3.1.1 โปรแกรมรวบรวมและตรวจสอบเอกสาร

3.1.1.1 ส่วนของการรวบรวมเอกสารหน้าเว็บไซต์(Crawler)



รูปที่ 3.2 โครงสร้างและสถาปัตยกรรมของส่วน Crawler

จากรูปภาพจะอธิบายขั้นตอนการทำงานของระบบรวบรวมเอกสารได้ดังนี้

- เริ่มต้นจากการกำหนดที่อยู่ของเว็บไซต์เริ่มต้น(URL) เพื่อให้ระบบเริ่มทำงานดาวน์โหลดข้อมูลหน้าเริ่มต้นมา

โหนดข้อมูลหน้าเริ่มต้นมา

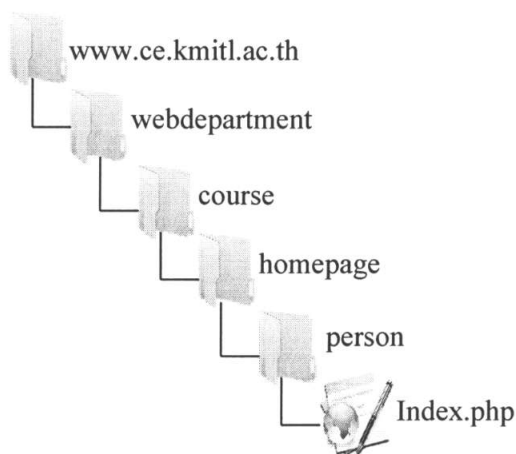
- ทำการค้นข้อมูลของหน้าเว็บไซต์ต่อไปหรือรูปภาพข้อมูลต่างๆ โดยตรวจสอบจากการค้นหาแท็กข้อมูล href และ src ตัวอย่างเช่น `` หรือ ``

- กำหนดความลึก(Depth) ของหน้าเว็บไซต์นั้นเช่น เว็บไซต์เริ่มต้นนั้นจะมีค่าความลึกเท่ากับ 0 ละทุกที่อยู่ของเว็บไซต์ที่ได้จากเว็บไซต์เริ่มต้นนั้นจะมีค่าเป็น 1 เป็นลำดับขั้นตอนไปเรื่อยๆ แล้วเก็บข้อมูลที่ได้แต่ละอันลงคิว(Queue)เพื่อรอการดาวน์โหลด

- ระบบจัดการจะทำการดึงข้อมูลจากคิวออกมาเพื่อทำการดาวน์โหลดโดยจะดึงข้อมูลที่มีลำดับความลึกน้อยสุดขึ้นมาก่อนแล้วส่งเข้าสู่ระบบ มัลติเทรดดาวน์โหลด(Multi-Thread Download) ซึ่งสามารถดาวน์โหลดได้พร้อมกันที่ละ 10 ไฟล์แล้วทำการค้นหาข้อมูลหน้าเว็บไซต์ต่อไปเก็บลงคิวเรื่อยๆ

- หากมีการพบเว็บไซต์ที่ได้ทำการดาวน์โหลดไปแล้วจะมีการตรวจสอบว่าขนาดข้อมูลนั้นมีการเปลี่ยนแปลงหรือไม่หากมีการเปลี่ยนแปลงจะทำการดาวน์โหลดมาเพื่อไปแทนที่ข้อมูลเดิม

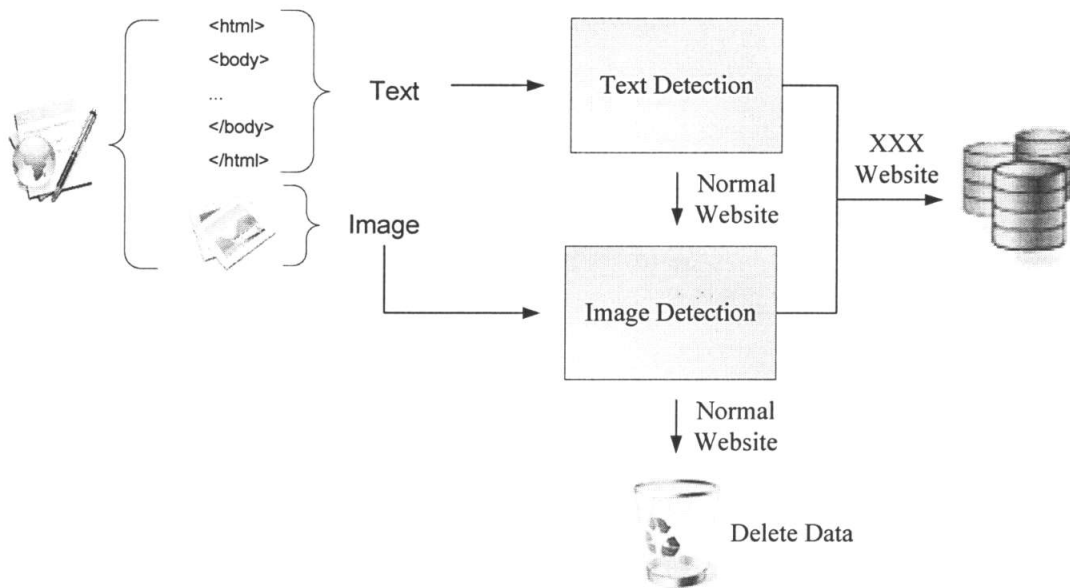
- เมื่อดาวน์โหลดข้อมูลมาแล้วจะทำการบันทึกข้อมูลนั้นลงเครื่องโดยจะสร้างรูปแบบของโครงสร้างไคเร็กทอรีให้เหมือนกับเว็บไซต์เพื่อง่ายต่อการจัดการดังรูปที่



<http://www.ce.kmitl.ac.th/webdepartment/course/homepage/person/index.php>

รูปที่ 3.3 ตัวอย่างการเก็บไฟล์ตามโครงสร้างของเว็บไซต์

3.1.1.2 ระบบวิเคราะห์ความเหมาะสมของเอกสารเว็บไซต์(Web analysis)



รูปที่ 3.4 โครงสร้างของระบบวิเคราะห์ความเหมาะสมของเอกสารเว็บไซต์

จากรูปที่ จะแสดงขั้นตอนการทำงานของระบบวิเคราะห์เว็บไซต์ได้คือขั้นแรกระบบจะอ่านข้อมูลของไฟล์ HTML ขึ้นมาจากการที่ระบบรวบรวมเอกสารเว็บไซต์ได้สร้างไว้แล้วเมื่ออ่านเอกสาร HTML ขึ้นมาแล้วนั้นจะทำการเข้าสู่กระบวนการตรวจสอบเอกสารข้อความหรือ Text detection เพื่อตรวจสอบว่าเป็นเว็บไซต์ที่เหมาะสมหรือไม่หากพบว่าไม่เหมาะสมจะทำการเก็บข้อมูลของ URL และขนาดของไฟล์ลงฐานข้อมูลของระบบไว้

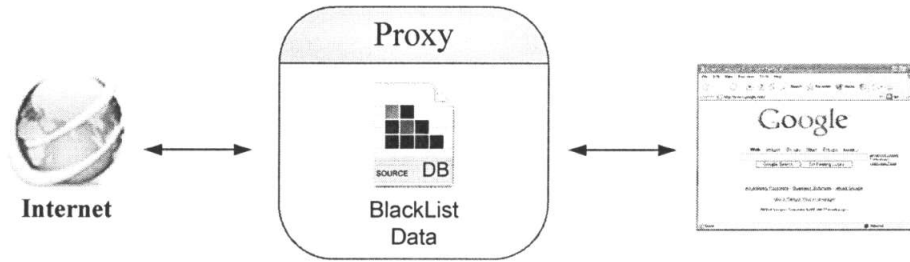
หากตรวจสอบแล้วพบว่าเป็นเว็บไซต์ธรรมดาจะนำเอกสาร HTML นั้นเข้าสู่ระบบตรวจสอบรูปภาพหรือ Image Detection โดยระบบจะทำการดึงข้อมูลรูปภาพที่มีทั้งหมดอยู่บนหน้าเอกสารนั้นออกมาแล้วนำไปตรวจสอบหากพบว่าพบว่ามีไม่เหมาะสมจะทำการเก็บข้อมูลของ URL และขนาดของไฟล์ลงฐานข้อมูลของระบบไว้เมื่อเสร็จสิ้นขั้นตอนก็จะทำการลบเอกสาร HTML และรูปภาพทิ้ง

โดยในส่วนของการตรวจสอบเอกสาร HTML และระบบการตรวจสอบรูปภาพนั้นได้ทั้ง 2 ส่วนมาจากโปรเจก โปรแกรมคัดกรองข้อมูลเว็บไซต์มาใช้ในส่วนนี้

3.1.2 โปรแกรมส่วนควบคุมการใช้งานอินเทอร์เน็ต(Client)

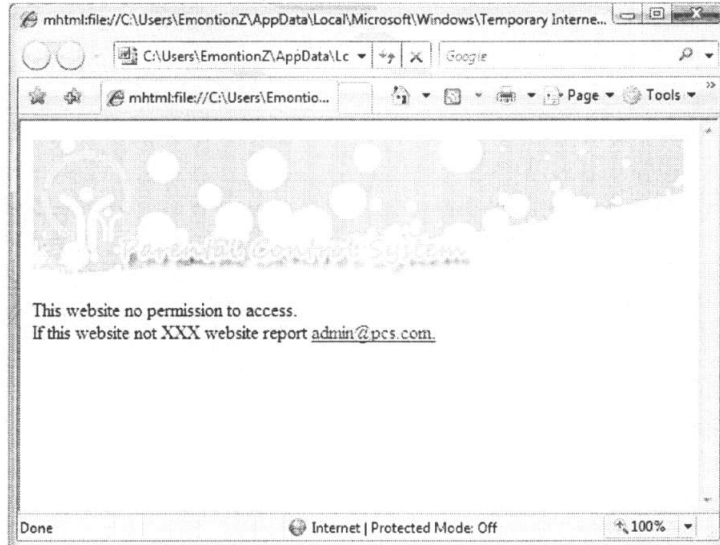
3.1.2.1 ระบบคัดกรองเว็บไซต์ที่ไม่เหมาะสม

โดยในระบบนี้มีรูปแบบการทำงานในลักษณะของฟ็อกซี่ (Personal Proxy) เพื่อใช้ในการควบคุมการใช้งานด้านเว็บไซต์



รูปที่ 3.5 โครงสร้างของระบบคัดกรองเว็บไซต์

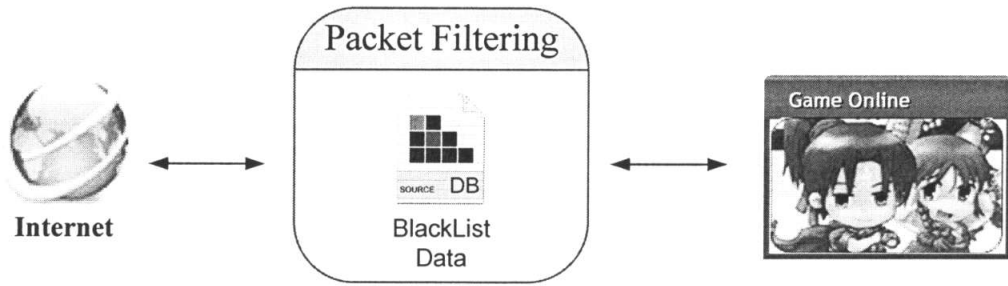
จากรูปที่ ส่วนนี้ระบบจะใช้ฟ็อกซ์ที่เป็นตัวกลางในการเชื่อมต่อเข้าสู่เว็บไซต์โดยมีเว็บเบราว์เซอร์ (Web Browser) ทำต้องการเข้าใช้เว็บไซต์จะทำการติดต่อไปยังฟ็อกซ์แทนการติดต่อยังเว็บเซิร์ฟเวอร์โดยตรงเมื่อฟ็อกซ์ได้รับการร้องขอจากเว็บเบราว์เซอร์จะทำการตรวจสอบ เว็บไซต์ (URL) ที่ร้องขอว่ามีอยู่ในข้อมูลเว็บไซต์ไม่เหมาะสมหรือไม่หากไม่มีฟ็อกซ์จะทำการร้องขอหน้าเว็บไซต์จากเว็บเซิร์ฟเวอร์แล้วส่งกลับให้เว็บเบราว์เซอร์ตามปกติ แต่หาก เว็บไซต์นั้นตรงกับข้อมูลเว็บไซต์ไม่เหมาะสมฟ็อกซ์จะไม่ทำการร้องขอหน้าเว็บไซต์จากเว็บเซิร์ฟเวอร์แต่จะส่งหน้าเว็บกลับไปที่เป็นการบอกว่าไม่อนุญาตให้เข้าใช้งาน



รูปที่ 3.6 แสดงหน้าเว็บไซต์เมื่อเข้าเว็บที่มีอยู่ในข้อมูลเว็บไซต์ไม่เหมาะสม

3.1.2.2 ระบบคัดกรองข้อมูลเชื่อมต่ออินเทอร์เน็ต

โดยระบบนี้จะทำหน้าที่ในการป้องกัน โปรแกรมที่กำหนดเชื่อมต่อสู่อินเทอร์เน็ตได้โดยในระบบนี้จะเพื่อป้องกันเกมส์ออนไลน์ต่างเชื่อมต่อสู่อินเทอร์เน็ต



รูปที่ 3.7 แสดงโครงสร้างของระบบคัดกรองข้อมูลเชื่อมต่ออินเทอร์เน็ต

เมื่อมีโปรเซสต้องการเชื่อมต่อสู่ระบบเกิดขึ้นตัวระบบจะทำการตรวจสอบข้อมูลการเชื่อมต่อต่างๆเช่น ชื่อของโปรเซส ไอพีของปลายทาง พอร์ตของปลายทาง หากตรงตามข้อมูลโปรแกรมไม่เหมาะสมระบบจะทำการตัดการเชื่อมต่อของโปรเซสนั้นและสามารถสั่งปิดโปรเซสนั้นได้ตามต้องการ

3.1.2.3 ระบบจัดเวลาการใช้งานอินเทอร์เน็ต

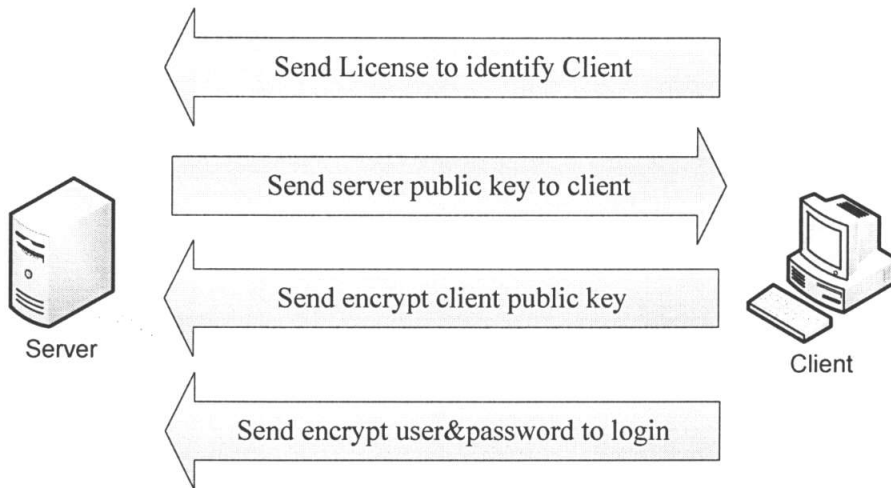
ระบบนี้จะเป็นส่วนในการจัดสรรเวลาในการใช้งานอินเทอร์เน็ตโดยจัดรูปแบบต่างๆให้เลือกใช้ในเวลาๆต่างๆกันตามความเหมาะสมเช่น อนุญาตให้เล่นเกมเฉพาะช่วง 16.00-18.00 วันจันทร์ – ศุกร์นอกนั้นม่อนุญาตระบบก็จะทำการตรวจสอบเวลาเมื่อไม่ใช่ช่วงเวลาตามที่กำหนดระบบจะทำการปิดกั้นการเชื่อมต่อของเกมจนกว่าจะถึงเวลาที่กำหนด

ระบบนี้จะมีการปรับเวลาตามเซิร์ฟเวอร์เสมอเพื่อป้องกันการปรับเปลี่ยนเวลาของเด็กๆโดยทุกๆ 5 นาทีแอปพลิเคชันจะทำการส่งเวลาปัจจุบันของเครื่องไปให้ตรวจสอบว่าคลาดเคลื่อนจากเซิร์ฟเวอร์เท่าใดหากคลาดเคลื่อนมากเซิร์ฟเวอร์จะส่งคำสั่งมายังแอปพลิเคชันเพื่อให้ตั้งเวลาของเครื่องตามที่เซิร์ฟเวอร์กำหนดมา

3.2 โครงสร้างการเชื่อมต่อของระบบและการอัปเดตข้อมูล

3.2.1 โครงสร้างการเชื่อมต่อ

การเชื่อมต่อระหว่างเซิร์ฟเวอร์และตัวแอปพลิเคชันนั้นได้มีการรักษาความปลอดภัยโดยมีการเข้ารหัสข้อมูลในการสื่อสารกัน



รูปที่ 3.8 แสดงขั้นตอนการเชื่อมต่อสู่ระบบ

ในการใช้งานนั้นต้องมีการเข้าสู่ระบบ(Login) เพื่อเป็นการยืนยันตัวตน โดยในการเชื่อมต่อสู่เซิร์ฟเวอร์นั้นจะมีขั้นตอนดังนี้

- ตัวแอฟพลิเคชัน(Client) จะทำการส่งค่า รหัสประจำตัว(License) ซึ่งจะติดมากับแอฟพลิเคชันแต่ละตัวซึ่งเก็บอยู่ใน รีจิสเตอร์ ไปให้ยังเซิร์ฟเวอร์เพื่อให้เซิร์ฟเวอร์ตรวจสอบความถูกต้องของแอฟพลิเคชัน

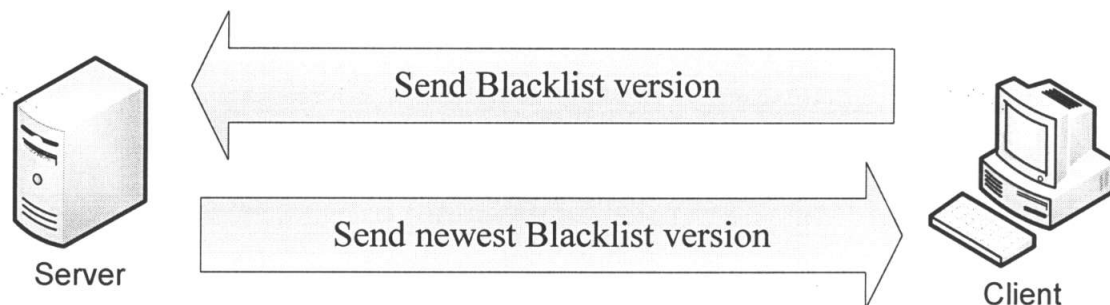
Name	Type	Data
ab(Default)	REG_SZ	(value not set)
License	REG_QWORD	0xa4c5f89e3d21f4cb (11873169351146337483)

รูปที่ 3.9 แสดงค่ารหัสประจำตัว(License)

- หากเซิร์ฟเวอร์ตรวจสอบแล้วพบว่า รหัสประจำตัวนั้นมีอยู่จริง เซิร์ฟเวอร์จะทำการสุ่มค่าขึ้นมาค่าหนึ่งแล้วนำไปสร้างการเข้ารหัสแบบ RSA เมื่อสร้างขึ้นมาแล้วเซิร์ฟเวอร์จะทำการส่ง กุญแจสาธารณะ(Public key) ไปให้ตัวแอฟพลิเคชัน
- เมื่อตัวแอฟพลิเคชันได้รับ กุญแจสาธารณะจากเซิร์ฟเวอร์แล้วจะทำการสุ่มค่าขึ้นมาเพื่อสร้างการเข้ารหัสแบบ RSA เช่นกันจากนั้นแอฟพลิเคชันจะทำการเข้ารหัส กุญแจสาธารณะของตัวเองด้วย กุญแจสาธารณะของเซิร์ฟเวอร์แล้วทำการส่งกลับไปให้ยังเซิร์ฟเวอร์
- เมื่อทำการแลกกุญแจสาธารณะกันเรียบร้อยแล้ว แอฟพลิเคชันจะทำการส่ง ชื่อเข้าใช้งาน (username) และรหัสผ่าน(password) โดยเข้ารหัสข้อมูลทั้งหมดแล้วส่งไปยังเซิร์ฟเวอร์เพื่อเป็นการยืนยันการเข้าระบบ

3.2.2 การอัปเดตฐานข้อมูล

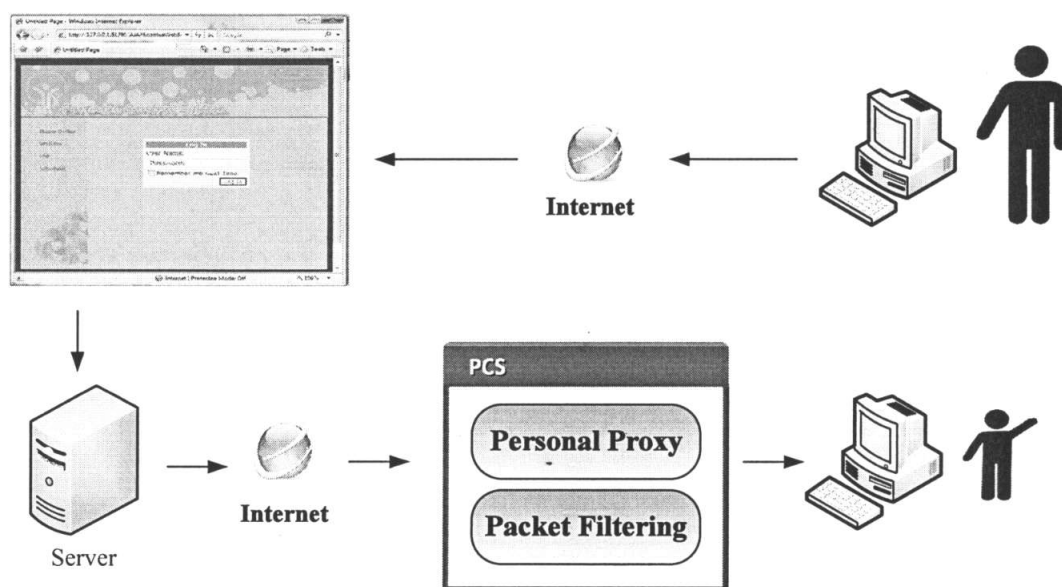
ในส่วนนี้เป็นส่วนที่สร้างขึ้นเพื่อให้แอปพลิเคชันนั้นทันสมัยต่อเว็บไซต์ที่ไม่เหมาะสมและเกมส์ออนไลน์ที่เกิดขึ้นใหม่ตลอดเวลา



รูปที่ 3.10 แสดงขั้นตอนการอัปเดตข้อมูล

- เมื่อทั้งสองฝั่งได้ทำการเชื่อมต่อและยืนยันตัวเองเรียบร้อยแล้วฝั่งแอปพลิเคชันจะส่งค่าเวอร์ชันปัจจุบันของฐานข้อมูลที่ใช้อยู่ให้กับเซิร์ฟเวอร์จากนั้นเซิร์ฟเวอร์จะทำการตรวจสอบว่าเซิร์ฟเวอร์ชั้นล่าสุดที่มีอยู่หรือไม่หากไม่ใช่เซิร์ฟเวอร์จะทำการส่งข้อมูลล่าสุดกลับไปให้แทน
- การสร้างฐานข้อมูลที่เซิร์ฟเวอร์นั้นระบบจะตรวจสอบทุกๆ 1 ชั่วโมงว่ามีเว็บไซต์ที่ไม่เหมาะสมที่ได้จากระบบวิเคราะห์เพิ่มขึ้นหรือเปล่าหากพบว่ามีเปลี่ยนแปลงเซิร์ฟเวอร์จะทำการสร้างไฟล์ฐานข้อมูลเวอร์ชันล่าสุดออกมา

3.3.3 โครงสร้างการเชื่อมต่อระหว่างแอปพลิเคชันและเว็บไซต์ควบคุม



รูปที่ 3.11 โครงสร้างการเชื่อมต่อระหว่างแอปพลิเคชันและเว็บไซต์ควบคุม

จากรูป จะแสดงให้เห็นถึงโครงสร้างการทำงานโดยผู้ปกครองสามารถติดตามและดูแลการใช้งานของเยาวชนได้จากทุกหนทุกแห่งโดยแค่เพียงผ่านเว็บไซต์ก็สามารถเชื่อมต่อสู่ตัวแอปพลิเคชันที่บ้านได้ทันทีโดยตัวเว็บไซต์นั้นมีฟังก์ชันการทำงานคร่าวๆดังนี้

- เรียกดูข้อมูลการใช้งานเว็บไซต์และโปรแกรมต่างๆที่เกิดขึ้นยังแอปพลิเคชันขึ้นมาแสดงที่หน้าเว็บไซต์
- ปรับแต่งและแก้ไขค่าต่างๆของระบบเช่น การเพิ่มหรือลบข้อมูลจากไฟล์ข้อมูล การแก้ไขแค่การจัดสรรเวลาการใช้ อินเทอร์เน็ต เป็นต้น

บทที่ 4

หลักการพื้นฐานบนเครือข่าย

4.1 โพรโทคอลเอชทีทีพี (HTTP: HyperText Transfer Protocol)

โพรโทคอลนี้สร้างขึ้นสำหรับบริการที่เรียกว่า WWW (World Wide Web) ในเครือข่ายอินเทอร์เน็ตโดยเฉพาะ โพรโทคอลนี้จะเป็นตัวกำหนดวิธีการส่งข้อมูลหรือไฟล์ระหว่างเครื่องคอมพิวเตอร์ลูกข่าย (Client) และ เครื่องแม่ข่าย (Server) รวมถึงกำหนดกฎระเบียบในการติดต่อด้วยโพรโทคอลนี้ถูกออกแบบมาให้มีความกระชับรัด สามารถทำงานได้รวดเร็ว มีการะบวนการทำงานที่ไม่ซับซ้อน และมีคำสั่งที่ใช้งานไม่มากนัก แต่สามารถรองรับข้อมูลได้ทุกแบบ ไม่ว่าจะเป็นข้อมูลทั่วไปที่เข้ารหัสแบบ MIME หรือข้อมูลที่เป็นกราฟิก เช่น ไฟล์ที่เป็น GIF หรือ JPEG เป็นต้น

4.1.1 วิธีการติดต่อของโพรโทคอล HTTP

โพรโทคอล HTTP อยู่บนพื้นฐานของระบบเครือข่ายเครื่องลูกข่าย / เครื่องแม่ข่าย (Client / Server) ที่ต้องมีการร้องขอรับบริการจากเครื่องลูกข่าย (request) และการตอบสนองหรือการให้บริการของเครื่องแม่ข่าย(response) จึงสามารถแบ่งการทำงานออกเป็น 2 ด้านคือ ด้านเว็บเครื่องแม่ข่าย และด้านเครื่องลูกข่าย โดยเครื่องลูกข่ายจะติดต่อเข้ามายังเครื่องแม่ข่ายและอ้างถึงแอดเดรสของเครื่องแม่ข่ายโดยใช้รูปแบบของURL ส่วนด้านเครื่องแม่ข่ายจะส่งข้อมูลกลับมาในรูปแบบที่เป็นภาษา HTML(HyperText Markup Language) โดยที่โพรโทคอล HTTP ใช้วิธีการเข้ารหัสในแบบ MIME เป็นมาตรฐานของการทำงาน โพรโทคอลนี้ถูกออกแบบมาให้สามารถรับส่งข้อมูลผ่าน Proxy หรือ Firewall ต่าง ๆ ได้ โดยอาศัยการเชื่อมต่อผ่านทางโพรโทคอล TCP/IP อีกทีหนึ่ง โดยใช้พอร์ตหมายเลข 80 เป็นช่องทางมาตรฐานในการติดต่อ ในทางปฏิบัติจะใช้พอร์ตหมายเลขอื่นก็ได้ ในปัจจุบันเว็บเบราว์เซอร์ทั่วไปจะกำหนดค่ามาตรฐานไว้ที่พอร์ต 80 ดังนั้นหากมีการกำหนดไว้ที่พอร์ตอื่น จะทำให้เกิดความลำบากต่อผู้ใช้ที่ต้องระบุหมายเลขพอร์ตลงในURL ด้วยด้วยเหตุที่การทำงานของโพรโทคอล HTTP เป็นแบบเครื่องลูกข่ายและเครื่องแม่ข่าย ดังนั้นการติดต่อสื่อสารใดๆ ผ่านโพรโทคอลนี้จำเป็นต้องมีเครื่องตัวแม่กับตัวลูก การสื่อสารจะสมบูรณ์ได้ การติดต่อกันระหว่างเครื่องลูกข่ายไปยังเครื่องแม่ข่ายผ่านโพรโทคอล HTTP มีขั้นตอนดังนี้

ขั้นแรก : Open Socket

เครื่องลูกข่าย จะสร้างการเชื่อมต่อ (Connection) กับเครื่องแม่ข่ายผ่านซ็อกเก็ต (Socket)

ขั้นที่สอง : Request

เครื่องลูกข่ายส่งคำร้องขอข้อมูลไปยังเครื่องแม่ข่าย

ขั้นที่สาม : Information Transfer

เครื่องแม่ข่ายจะไปหาข้อมูลที่เครื่องลูกข่ายต้องการ

ขั้นที่สี่ : Response

เครื่องแม่ข่ายส่งข้อมูลตอบสนอง (Response) กลับมายังเครื่องลูกข่ายเสมอ

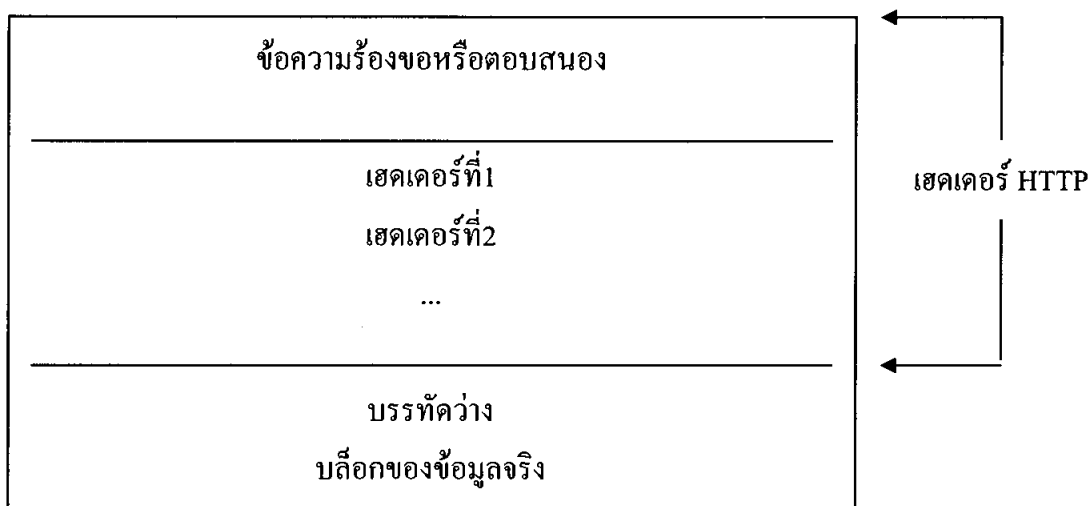
ขั้นสุดท้าย : Close Socket

ปลดการเชื่อมต่อของซ็อกเก็ตของทั้งสองฝั่งออก

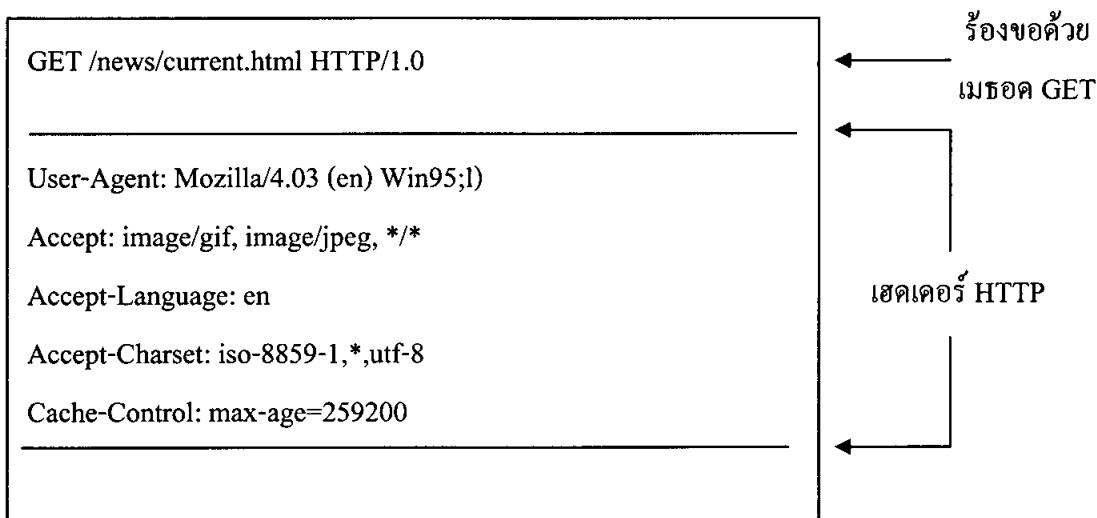
ด้วยการทำงานของโพรโตคอล HTTP ที่มีการเชื่อมต่อในระยะเวลาเพียงสั้น หรือที่เรียกว่าเป็นโพรโตคอลแบบ Connectionless ในลักษณะดังกล่าว ทำให้ในช่วงเวลาหนึ่ง ๆ เครื่องแม่ข่ายที่ให้บริการ WWW สามารถรองรับเครื่องลูกข่ายได้จำนวนมากพร้อม ๆ กัน เพราะไม่มีใครได้ทำการเชื่อมต่ออย่างถาวรในการร้องขอรับบริการจากเครื่องลูกข่าย และการตอบสนองหรือการให้บริการจากเครื่องแม่ข่ายนั้นย่อมต้องมีการรับส่งข้อมูลระหว่างกัน แต่ข้อมูลที่รับส่งกันในแต่ละครั้งไม่ได้มีเฉพาะข้อมูลเพียงอย่างเดียว แต่ละฝ่ายจะมีส่วนเฮดเดอร์ HTTP (HTTP header) เข้าไปในส่วนต้นของข้อมูลที่รับ-ส่งกันด้วย ซึ่งเฮดเดอร์ HTTP จะเป็นตัวบอกว่าข้อมูลที่ส่งมานี้เป็นข้อมูลอะไร เป็นข้อมูลการร้องขอจากเครื่องลูกข่าย หรือเป็นข้อมูลตอบสนองจากเครื่องแม่ข่าย

4.1.2 โครงสร้างของโพรโตคอล HTTP

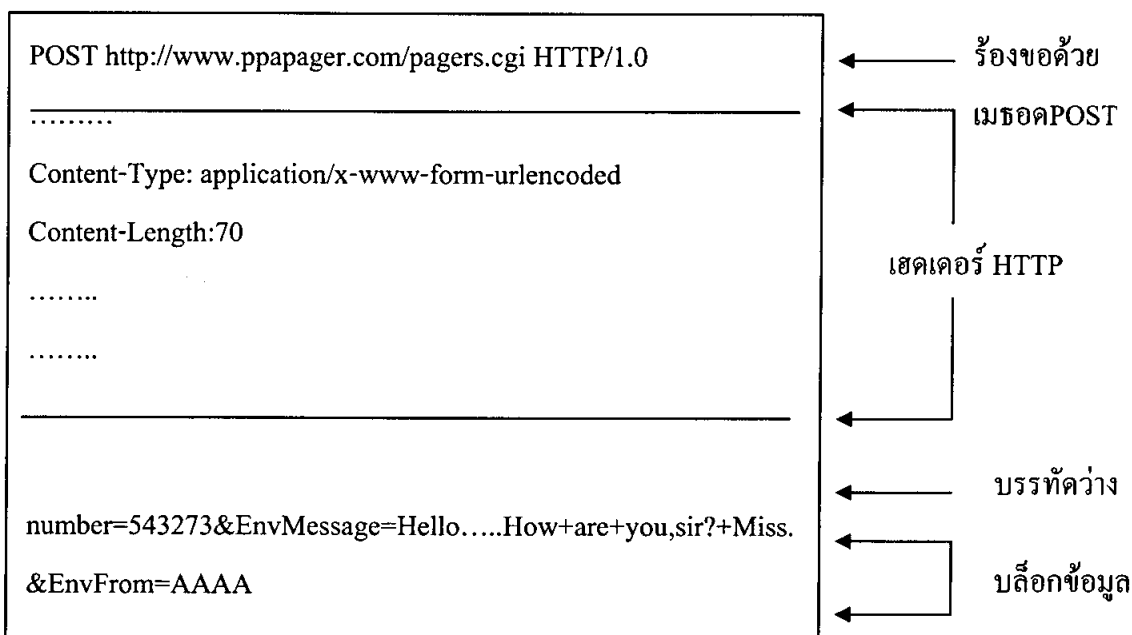
โครงสร้างของข้อมูล HTTP จะแบ่งออกเป็น 2 ส่วนใหญ่ ๆ คือ ส่วนเฮดเดอร์ หรือเรียกว่า Metadata จะเป็นส่วนเก็บข้อมูลที่จำเป็นต้องใช้ภายในโพรโตคอล ส่วนที่สองเป็นส่วนของข้อมูลจริงที่ต้องการรับส่ง ดังแสดงในรูปที่ 3.1 และมีรูปแบบการส่งข้อมูลทั้ง 2 แบบ ดังรูปที่ 3.2, 3.3



รูปที่ 4.1 โครงสร้างของข้อมูลที่ส่งผ่านโพรโทคอล HTTP



รูปที่ 4.2 ตัวอย่างข้อความร้องขอด้วยเมธอด GET



รูปที่ 4.3 ตัวอย่างข้อความร้องขอด้วยเมธอด POST

4.2 พร็อกซีส่วนตัว (Personal Web Proxy)

4.2.1 พร็อกซี (Proxy) คืออะไร

พร็อกซี (Proxy) คือโปรแกรมที่ทำหน้าที่เป็นได้ทั้ง เครื่องแม่ข่าย และ ไคลเอนต์ สำหรับจุดประสงค์ในการสร้างการร้องขอเพื่อเครื่องไคลเอนต์อื่นๆ การร้องขอจะถูกให้บริการโดยการส่งเข้าไปภายในพร็อกซี ซึ่งในบางกรณีจะมีการเปลี่ยนไปยังเครื่องเครื่องแม่ข่ายอื่น พร็อกซีจะต้องสามารถจัดการได้ทั้งแบบ เครื่องแม่ข่าย และ ไคลเอนต์ที่ต้องการรายละเอียดดังกล่าว ปัจจุบันได้แบ่งพร็อกซีออกเป็นหลายประเภทแต่สามารถแบ่งเป็นประเภทใหญ่ๆ ได้ 2 แบบ คือ

Transparent Proxy คือ พร็อกซีที่จะไม่ถูกปรับเปลี่ยนการร้องขอหรือการตอบรับเมื่อได้ถูกร้องขอเพื่อการพิสูจน์ยืนยันตัวตนของพร็อกซีและระบุตัวตน (proxy authentication และ identification) ผู้ใช้งานไม่จำเป็นต้องเซตค่าอะไรในเครื่องคอมพิวเตอร์ของตัวเอง ตัวของ Transparent Proxy นั้นจะใช้หลักของ IP Tables ในการ redirects ผู้ที่ติดต่อผ่านมาทางพอร์ต http ให้ไปยัง พร็อกซีเครื่องแม่ข่าย ได้เองโดยอัตโนมัติ

Non-Transparent Proxy คือ พร็อกซีที่สามารถปรับเปลี่ยนการร้องขอและตอบรับเพื่อให้บริการสำหรับผู้ใช้งาน เช่น ตัวแทนผู้ให้บริการ, การเปลี่ยนรูปของสื่อกลางประเภทต่างๆ, protocol reduction หรือ anonymity filtering ยกเว้นในที่ที่ใช้ Transparent Proxy หรือ Non-Transparent Proxy ที่อยู่ในสถานะที่แน่นอนแล้ว HTTP proxy ต้องการที่จะนำมาใช้ได้กับทั้งสองชนิดนี้

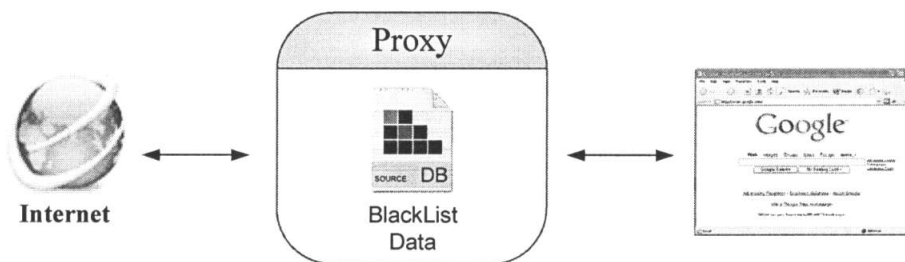
Proxy server คือ คอมพิวเตอร์ที่มีบริการต่างๆ บนระบบเน็ตเวิร์ค อนุญาตให้ไคลเอนต์สามารถทำการเชื่อมต่อไปยังบริการเน็ตเวิร์คอื่นๆ ได้ โดยไคลเอนต์จะเชื่อมต่อไปที่ proxy server ก่อน จากนั้นร้องขอการเชื่อมต่อ ไฟล์ หรือทรัพยากรอื่นๆ บนเครื่องแม่ข่ายต่างๆ กัน ซึ่งพร็อกซี่จะต้องหาทรัพยากรนั้น ส่งคืนให้กับไคลเอนต์ จากการเชื่อมต่อไปยังเครื่องแม่ข่ายที่ระบุไว้ หรือคอยให้บริการเองบนแคช (Cache) ที่มีอยู่ แต่บางกรณี พร็อกซี่อาจเปลี่ยนการร้องขอของไคลเอนต์ หรือเปลี่ยนการตอบกลับจากทางฝั่งเครื่องแม่ข่าย ขึ้นอยู่กับจุดประสงค์ที่ตั้งไว้ของการใช้งาน

พร็อกซี่ที่ทำการคัดกรองข้อมูลเนื้อหาบนเว็บ เรียกว่า เว็บพร็อกซี่ (Web Proxy) โดยเว็บพร็อกซี่จำนวนมากจะเปลี่ยนรูปแบบของเว็บเพจที่จะส่งใหม่ เพื่อจุดประสงค์บางอย่างตามที่ต้องการ ผู้ควบคุมดูแลเน็ตเวิร์คจะนำพร็อกซี่มาใช้เพื่อต้องการสกัดกั้น ไวรัสคอมพิวเตอร์ หรือเนื้อหาของโปรแกรมที่มีเจตนาร้ายต่อผู้ใช้

ในหลายๆ อಂಗกรค์ อาทิ เช่น บริษัท โรงเรียน จะใช้ Proxy server บังคับใช้บนเน็ตเวิร์คตามนโยบาย หรือเพื่อต้องการความปลอดภัย ปราศจากโปรแกรมประสงค์ร้าย หรือเพื่อบริการแคชในตัว

4.2.2 พร็อกซี่ส่วนตัว (Personal Web Proxy) คืออะไร

พร็อกซี่ส่วนตัว (Personal Web Proxy) คือพร็อกซี่ที่อยู่บนเครื่องของผู้ใช้ เมื่อผู้ใช้จะทำการร้องขอข้อมูลของหน้าเว็บเพจไปที่เว็บเครื่องแม่ข่ายต่างๆ จะต้องส่งและรับข้อมูลผ่านพร็อกซี่ที่ติดตั้งอยู่ที่เครื่องดังรูปที่ 3.4 นั่นคือ พร็อกซี่ที่อยู่ในเครื่องนี้ทำหน้าที่เป็น Port Forwarding นั่นเอง



รูปที่ 4.4 รูปแบบของ Personal Web Proxy

4.3 โทคอดลที่ใช้ในการสื่อสารระหว่างเครื่องแม่ข่ายและเครื่องลูกข่าย(PCS Potocol)

ในการสื่อสารกันระหว่างระบบนั้นได้มีการกำหนดโทคอดลขึ้นมาตัวหนึ่งเพื่อให้ง่ายและสะดวกในการติดต่อกันในระหว่างแต่ละส่วน

4.3.1 โครงสร้างของโปกคอด

1 Byte type	1 Byte code	Data
-------------	-------------	------

รูปที่ 4.5 แสดงโครงสร้างของข้อมูลที่ใช้ในการสื่อสาร

จากรูปข้อมูลที่ใช้ในการสื่อสารจะแบ่งออกเป็น 3 ส่วน โดยแบ่งเป็นแสดง ชนิดของข้อมูล คำสั่งของข้อมูล และข้อมูล โดยโดยค่าต่างๆสามารถดูได้ดังตารางดังต่อไปนี้

ตารางที่ 4.1 แสดงความหมายของค่า Type

Type	ความหมาย
0	ข้อมูลระหว่าง เซิร์ฟเวอร์กับเว็บไซต์หรือเครื่องลูกข่าย
1	ข้อมูลระหว่าง เว็บไซต์กับเครื่องลูกข่าย

ตารางที่ 4.2 แสดงความหมายของค่า Code

Code	ความหมาย	ข้อมูล
0	ยืนยันตัวตน	License Key
1	แลกเปลี่ยนแฉสาธารณะ	Public Key
2	เข้าสู่ระบบ	Username&Password
3	ตอบรับการเข้าสู่ระบบ	1:สำเร็จ 0:ไม่สำเร็จ
4	ตรวจสอบการอัปเดต	Blacklist Version
5	ข้อมูลการตั้งค่า	Config Data
6	ข้อมูลการใช้งาน	Using Data

บทที่ 5

เทคนิคที่ใช้ในการเขียนโปรแกรม

5.1 การเขียนโปรแกรมแบบไม่ประสานเวลา (Asynchronous Programming)

การเขียนโปรแกรมแบบ Asynchronous Programming ทำให้การเชื่อมต่อจะไม่ถูกปิดกั้น โดยตัวแอปพลิเคชันระหว่างที่รอคอยกระบวนการทำงานของระบบเครือข่ายเสร็จสมบูรณ์ แทนที่จะทำการปิดกั้นเซิร์ฟเวอร์ที่ทำงานอยู่ปัจจุบัน ซึ่งจะใช้รูปแบบของ .NET asynchronous ในกระบวนการติดต่อในระบบเครือข่ายบน thread อื่นๆ ระหว่างที่โปรแกรมอยู่ในระหว่างการทำงานใน thread แรก เมื่อ thread ที่ 2 ได้ทำงานเสร็จเรียบร้อยแล้ว มันจะส่ง event object ไปที่เซิร์ฟเวอร์ซึ่งเป็นการบอกสถานะของ task นั้นๆ

เหตุการณ์ (Event) สามารถที่จะส่งสัญญาณออกไปในกรณีที่เมธอดนั้นกำลังรอคอยที่จะตอบกลับ หรือไม่มีการส่งสัญญาณตอบกลับ ถ้าเหตุการณ์ไม่ได้ถูกส่งสัญญาณเมธอดที่รอคอยอยู่จะถูกปิดกั้น (block) จนกว่าสัญญาณถูกส่งมา

เราจะมีชนิดของเหตุการณ์อยู่ 2 ชนิด นั่นคือ

- Automatic event มันสามารถทำให้ตัวมันเองกลับไปสถานะไม่ส่งสัญญาณ (non-signaled state) เมื่อไรก็ตามที่มันถูกส่งสัญญาณออกไป

- Manual reset event มันจะส่งสัญญาณจนกระทั่งมันได้ถูกส่งให้รับการ reset ไปยังสถานะไม่ส่งสัญญาณ (non-signaled state)

.NET จะมี ManualResetEvent class ซึ่งเราสามารถเรียกใช้ในโปรแกรมได้ หากเราต้องการที่จะดำเนินการต่อ ให้เราทำการตั้งค่า ManualResetEvent object ไปที่สถานะส่งสัญญาณ (signaled state) จะทำให้มันหยุดปิดกั้น (Block) ทุกๆเมธอดที่ทำการรอคอยอยู่ และโปรแกรมสามารถดำเนินการต่อได้ เราจะใช้ object นี้เพื่ออำนวยความสะดวกของการเขียนโปรแกรม asynchronous programming ได้

รูปแบบการเขียนโปรแกรมแบบ Asynchronous programming เหมาะสมสำหรับโปรแกรมที่ไม่สามารถรอคอยเพื่อการทำงานบนเครือข่ายได้สมบูรณ์ก่อนที่จะทำการต่อไป

รูปแบบการเขียนโปรแกรมแบบ Asynchronous programming จะใช้เมธอดที่เรียกว่า callback ในการส่งผลของการทำงาน .NET Framework จะมี method ที่จัดการเกี่ยวกับการกำหนดค่าเริ่มต้น แต่ต้องการเมธอด callback ในการจบการทำงาน เช่น เมธอด Connect () ใช้ BeginConnect () ในการเริ่มการติดต่อไปที่อุปกรณ์เครือข่ายและในการทำงานร่วมกับเมธอด callback เพื่อให้เกิดการเชื่อมต่อที่สมบูรณ์

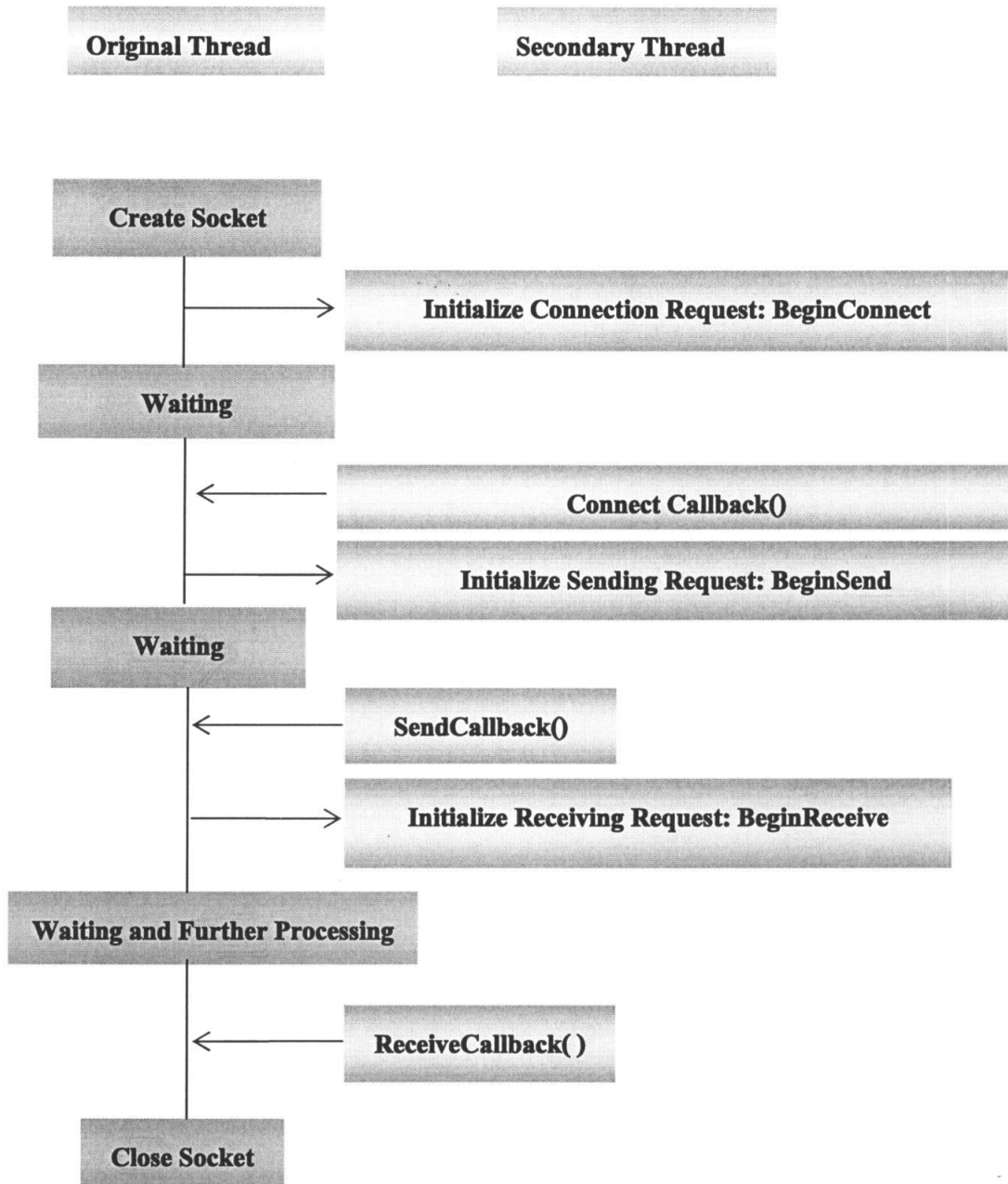
.NET Framework มีคลาสที่ใช้สำหรับการทำงาน Asynchronous นั่นคือ AsyncCallback สำหรับใช้ในโปรแกรมเพื่อที่จะทำให้กระบวนการทำงานแบบ Asynchronous สมบูรณ์

```
AsyncCallback aCallback = new AsyncCallback (AsyncCallback);
```

ในรูปแบบนี้จะใช้ Multiple threads ไปยังกระบวนการเชื่อมต่อบนเครือข่าย เพื่อความง่ายในการ synchronization ระหว่าง เทรด ต่างๆ เราจะใช้คลาส ManualResetEvent เพื่อจะระงับการทำงานของคลาสหลักและสัญญาณเมื่อการทำงานสามารถดำเนินการต่อได้

ต่อไปเราจะอธิบายการทำงานเป็นแผนผังการทำงานของทั้งฝั่งไคลเอ็นต์ และฝั่งเซิร์ฟเวอร์

5.1.1 Asynchronous Client Application



รูปที่ 5.1 แผนภาพขั้นตอนการทำงานของไคลเอ็นต์แอปพลิเคชัน

แผนภาพดังรูปที่ 5.1 แสดงขั้นตอนการทำงานของทางฝั่งไคลเอ็นต์แอปพลิเคชัน ประกอบไปด้วย 3 งานที่จะทำให้ Asynchronous สามารถทำงานได้ เป็นการเชื่อมต่อไปยังฝั่งเซิร์ฟเวอร์ ทำการส่งข้อมูล และรับข้อมูลกับทางเซิร์ฟเวอร์

เนื่องจากเราไม่สามารถทำงานกับซ็อกเก็ต หากเราไม่สามารถทำการหยุดการเชื่อมต่อระหว่างเซิร์ฟเวอร์ หรือ Deadlock สามารถที่จะเกิดขึ้นได้หากทำการรับข้อมูลแทนที่ควรจะจะต้องส่งข้อมูล เราจะใช้คลาส ManualEventReset ที่จะทำให้ Synchronize ต่าง เทรด กันเกิดขึ้นมา โดย ManualEventReset จะมีสถานะ set และ reset (เมธอด Set (), Reset ()) โดยแต่ละงานประกอบไป

ด้วยการเชื่อมต่อ การส่ง และการรับข้อมูล ซึ่งสามารถใช้เมธอด `WaitOne ()` ที่จะทำให้ เธรดหลักสามารถรองานที่เกี่ยวข้องอย่างสมบูรณ์ (เมธอดนี้จะปิดกั้นเธรดปัจจุบันจนกว่าที่จะได้ส่งสัญญาณ)

โดยงานแรกของเราคือการเชื่อมต่อไปที่เซิร์ฟเวอร์ ใช้เมธอด `BeginConnect ()` ซึ่งเป็นการเริ่มการเชื่อมต่อให้เป็นการร้องเรียกการเชื่อมต่อแบบ `Asynchronous` จากนั้นทำการสร้าง `AsyncCallback` เพื่อให้ไปที่เมธอดที่ต้องการเมื่อเกิดการ `Callback` และในเมธอดที่ `Callback` นี้ควรที่จะเรียกเมธอด `EndConnect ()` ในการหยุดการร้องเรียกการเชื่อมต่อ และ `return socket` ที่ทำการเชื่อมต่อ และสุดท้ายเราจะทำการเรียกเมธอด `Set ()` ของ `ManualResetEvent` ซึ่งจะเป็นการบอกให้กับเธรดหลักว่าการเชื่อมต่อเสร็จสิ้นแล้ว

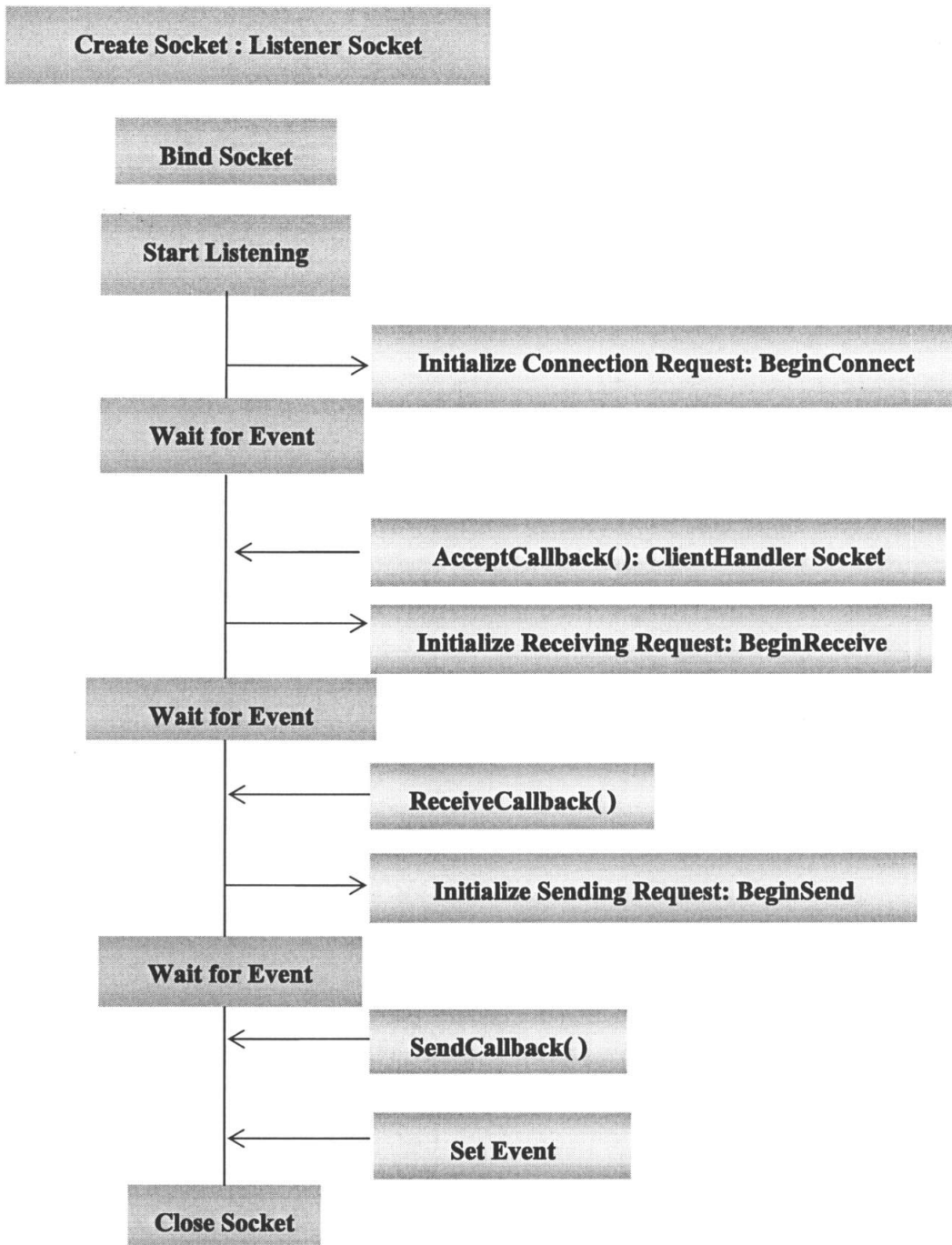
ต่อไปเราจะทำการติดต่อสื่อสารกับเซิร์ฟเวอร์ในการส่ง และรับข้อมูล ซึ่งใช้เมธอด `BeginSend ()` ที่จะใช้ในการส่งข้อมูลแบบ `Asynchronous` ไปยัง ซ็อกเก็ตที่ทำการเชื่อมต่อ ซึ่งเมธอดนี้จะเรียกฟังก์ชันที่ใช้ในการ `callback` ผ่านยัง `AsyncCallback` และทำการเรียกเมธอด `EndSend ()` เมื่อหยุดการร้องเรียกการส่งแบบ `Asynchronous` จากนั้นทำการ `Set ()` ของ `ManualResetEvent`

เมธอด `BeginReceive ()` ทำการเริ่มการรับข้อมูลแบบ `Asynchronous` จาก ซ็อกเก็ตซึ่งเมื่อเกิดการ `Callback` เมธอด `BeginReceive ()` จะทำการคืนค่า (`return`) จำนวนของไบต์ ดังนั้นเราสามารถที่จะทำการเช็คได้ว่าหากมีบางข้อมูลส่งกลับมาในคิว (`queue`)

5.1.2 Asynchronous Server Application

Original Thread

Secondary Thread



รูปที่ 5.2 แผนภาพขั้นตอนการทำงานของเซิร์ฟเวอร์แอสิงโครนัส

จากรูป 5.2 การทำงานจะเริ่มจากการสร้างซ็อกเก็ต และจากนั้นจะทำการรับฟังPort ที่จะมาเกี่ยวข้องกับโปรแกรม Asynchronous Server ต้องเรียก BeginAccept () เพื่อจะรับการติดต่อ และ

เมื่อได้รับการเชื่อมต่อเรียบร้อยแล้ว มันจะไปเรียกฟังก์ชัน `BeginReceive()` และ `BeginSend()` เพื่อส่งข้อมูลและรับข้อมูลจากซ็อกเก็ตของไคลเอ็นต์

ภายหลัง `BeginAccept()` ใ้ถูกเรียกแล้วเราจะตั้งสถานะให้อยู่ในสถานะการคอย (waiting state) ดังนั้นมันจะแตกต่างจากเซิร์ฟเวอร์ในโปรแกรมจะยังคงทำงานต่อไป เมื่อเครื่องไคลเอ็นต์พยายามที่ติดต่อฝั่งเซิร์ฟเวอร์ ซึ่งถ้าเราไม่ทำเช่นนี้โปรแกรมจะจบการทำงานก่อนเครื่องไคลเอ็นต์ ซึ่ง `ManualResetEvent` จะถูกใช้เพื่อที่จะตั้งสถานะในการคอย

ฟังก์ชัน `BeginAccept()` จะต้องใช้ 2 ตัวแปรคือตัวแรกคือตัวแปรของ `Asynchronous` (เพื่อการอ้างอิงถึงฟังก์ชัน callback) และอีกตัวแปรหนึ่งคือตัวแปรที่จะถูกใช้ส่งข้อมูลไปที่ฟังก์ชัน callback ฟังก์ชัน `BeginAccept()` จะไปเรียกฟังก์ชัน `AcceptCallback()` เมื่อการเชื่อมต่อใ้ถูกเชื่อมต่อกับซ็อกเก็ต ตัวฟังก์ชัน `Callback` นี้จะไปเรียกฟังก์ชัน `EndAccept()` ในการส่งซ็อกเก็ตใหม่กลับไปทางไคลเอ็นต์

การเขียนโปรแกรมแบบ `Asynchronous socket` จะใช้ `threads` สำหรับการดำเนินงานต่าง `asynchronous` กัน `thread` หลักจะมีไว้สำหรับกำหนดค่าเริ่มต้นของ `listener socket thread` ที่ 2 มีไว้สำหรับรับตอบรับเซิร์ฟเวอร์ที่ไ้ผ่านเข้ามาเซิร์ฟเวอร์สุดท้ายจะใช้สำหรับรับและส่งข้อมูล เราสามารถร้องขอค่า `thread ID` จากเซิร์ฟเวอร์ โดยใช้ฟังก์ชัน `GetCurrentThreadID()` หลังจากที่ได้เกิดการเชื่อมต่อโดยการเรียก `EndAccept()` ซ็อกเก็ตใหม่จะสามารถสื่อสารกับไคลเอ็นต์ โดยการเรียกใช้ฟังก์ชัน `BeginReceive()` และฟังก์ชัน `BeginSend()` ฟังก์ชัน `ReceiveCallback()` จะไปเรียกฟังก์ชัน `EndReceive()` เพื่อที่จะทำให้การทำงานแบบ `asynchronous` เสร็จสมบูรณ์ โดยจะมี `BeginSend()` ใ้ส่งข้อมูลกลับไปยัง `client` ไม่เช่นนั้นเราจะต้องเรียก `BeginReceive()` เพื่อที่จะรับข้อมูลที่ออกมา

เมื่อฟังก์ชัน `SendCallback()` ทำงานเสร็จเรียบร้อยแล้วโดยการเรียก `EndSend()` จากนั้นจะไปปิดตัวที่ควบคุมซ็อกเก็ต และไปตั้งค่า `ManualResetEvent` บนเซิร์ฟเวอร์หลัก

5.2 การสร้าง และการติดตั้งวินโดว์เซอร์วิสแอปพลิเคชัน (Windows Service Application) โดย Visual Studio.NET

วินโดว์เซอร์วิสแอปพลิเคชัน (Windows Service Application) ได้กลายเป็นส่วนหนึ่งของมาตรฐานของรูปแบบโปรเจกต์พื้นฐาน ซึ่งประโยชน์ของวินโดว์เซอร์วิส (Windows Service) คือวินโดว์เซอร์วิสสามารถเริ่มการทำงานของโปรแกรมก่อนที่ผู้ใช้ (User) คนใดคนหนึ่งจะเข้าสู่ระบบ (หากมีการปรับตั้งค่าให้เริ่มโปรแกรมเมื่อก่อนวินโดว์ทำการ Boot Up) และประโยชน์อีกอย่างหนึ่งคือสามารถรันโปรแกรมได้ตลอดเวลา ตรวจจับที่วินโดว์ยังทำการรันอยู่ โดยผู้ใช้ทั่วไปที่ไม่ใช่ admin จะไม่ทำการปิดเซอร์วิส หรือโปรเซสของโปรแกรมนี้อีก

5.2.1 เขียนโปรแกรมวินโดว์เซอร์วิส (Coding Windows Service)

ขั้นตอนแรกของการสร้างวินโดว์เซอร์วิส (ในที่นี้ยกตัวอย่างใช้ C#.NET) ให้เลือกเมนู New Project จากนั้นเลือกชนิด (Type) ของโปรเจกต์เป็น Windows Service

หลังจากสร้างโปรเจกต์เป็นวินโดว์เซอร์วิสใหม่แล้ว คลาส (Class) ที่ใช้จะถูกสืบทอด (Inheritance) มาจากคลาส System.ServiceProcess.ServiceBase ซึ่งถูกสร้างมาให้แล้ว พร้อมด้วย 4 เมธอด (Method) พื้นฐาน ได้แก่

```
protected override void OnStart(String[] args)
```

```
protected override void OnStop()
```

```
protected override void OnPause()
```

```
protected override void OnContinue()
```

5.2.2 การเพิ่มคลาส Installer ในโปรแกรม

วินโดว์เซอร์วิสแอปพลิเคชัน ต่างจากวินโดว์แอปพลิเคชัน ที่จะต้องถูกติดตั้งโดยใช้ installUtil.exe ในคลาสใหม่นี้ ให้สร้าง instance ของคลาส

```
System.ServiceProcess.ServiceProcessInstaller
```

และอีก 1 instance ของคลาส

```
System.ServiceProcess.ServiceInstaller
```

สำหรับแต่ละวินโดว์เซอร์วิสในโปรเจกต์ โดยแต่ละออบเจกต์

ของ ServiceInstaller ถูกสร้างความสัมพันธ์กับ 1 วินโดว์เซอร์วิส พิจารณาตามคุณสมบัติ

(Property) ServiceName ส่วนคุณสมบัติอื่นสามารถที่จะถูกกำหนดค่าตามลักษณะของเซอร์วิสตามต้องการ เช่น โหมด start (Automatic หรือ Manual)

5.2.3 การใช้เครื่องมือรันเพื่อช่วยในการติดตั้งวินโดว์เซอร์วิส

ตอนนี้ผู้พัฒนาสามารถคอมไพล์โปรเจกต์ไปเป็น ตัวโปรแกรมที่สามารถรันได้ และติดตั้ง เซอร์วิสบนเครื่องคอมพิวเตอร์ โดยใช้เครื่องมือของ .NET installUtil.exe ซึ่งคำสั่ง Command line ที่ต้องใช้มีดังนี้

`installUtil MyWinService.exe` สำหรับการติดตั้งวินโดวส์เซอร์วิสแอปพลิเคชัน

`installUtil /u MyWinService.exe` สำหรับการยกเลิกการติดตั้งของเซอร์วิส

5.2.4 การรวมส่วนประกอบของตัวจับเวลา (Timer Component) กับวินโดวส์เซอร์วิส

เนื่องจากตัวจับเวลาเป็นสิ่งที่ต้องการในกำหนดการของงานเรา ซึ่งมี 3 ทางในการสร้างตัวจับเวลาใน .NET Framework:

System.Timers.Timer class

System.Threading.Timer class

System.Windows.Forms.Timer class

คลาสที่ 3 เป็นตัวเลือกสำหรับแอปพลิเคชันกับ interface ดังนั้นในการใช้งานไม่เหมาะกับ เซอร์วิส ส่วนในคลาสที่ 2 ทำงานกับ TimerCallBack และ เธรด (Thread) ในการควบคุมตัวจับเวลา ดังนั้นในคลาสที่ 1 ได้ถูกนำมาใช้เกี่ยวข้องกับเมธอด และคุณสมบัติ ที่จะควบคุมการทำงาน โดยตรง และเพิ่มเหตุการณ์ (Event) โดยเรียก Timer.Elapsed เมื่อถึงช่วงเวลาในรอบซึ่งถูกตั้งไว้ เช่นตัวอย่างโค้ดดังรูปที่ 5.3

```
private void aTimer_Elapsed(object sender, ElapsedEventArgs e)
{
    if (DateTime.Now.Second == 50)
    {
        //MessageBox.Show("Now: second is 50");
        count++;
        //Create File on c:\
        StreamWriter file = new StreamWriter("c:\\Service" + count + ".txt");
        //Open File or Program: targetFile
        string targetFile = @"c:\\Service" + count + ".txt";
        System.Diagnostics.Process.Start(targetFile);
    }
}
```

รูปที่ 5.3 ตัวอย่างส่วนของโค้ดของเมธอด เมื่อได้รับเหตุการณ์ครบรอบ interval ของตัวจับเวลา

สมมติว่ามีการตั้งค่า `interval = 1000` (1000 millisecond:1 second) หมายความว่าทุกๆรอบ 1 วินาที จะเกิดเหตุการณ์ `aTimer_Elapsed` ดังตัวอย่างนี้ จะทำการเช็คว่าคุณค่าวินาทีปัจจุบันมีค่าเท่ากับ 50 หรือไม่ ถ้าเท่าจะทำการสร้างไฟล์ชื่อ `Service0.txt` ไว้ที่ `C:\` ในครั้งแรก (เนื่องจากค่าเริ่มต้นของ

count ถูกตั้งไว้เท่ากับ 0 และจะบวก 1 ทุกๆ 50 วินาที) จากนั้นจะเปิดไฟล์ Service0.txt ที่สร้างนี้ด้วยโปรแกรม notepad โปรแกรมจะทำงานนี้ไปเรื่อยๆ ในทุกๆ 50 วินาที

โดยออบเจ็กต์ของ Timer สามารถตั้งค่า และผูกกับเมธอดของวินโดว์เซอร์วิสได้ดังนี้

คุณสมบัติ Interval ถูกใช้ในการตั้งค่าเป็นช่วงเวลา ในหน่วย millisecond ที่จะเรียกเหตุการณ์ Elapsed

เมธอด Start() ตั้งค่าคุณสมบัติ Enabled ของ timer ให้เท่ากับ true คืออนุญาตให้ตัวจับเวลา timer เริ่มทำงานนับได้เพื่อส่งเหตุการณ์ event

เมธอด Stop() ตั้งค่าคุณสมบัติ Enabled ของ timer ให้เท่ากับ false มีผลให้ตัวจับเวลา timer หยุดส่งเหตุการณ์เมื่อครบช่วงเวลา

5.2.5 การทดลองวินโดว์เซอร์วิสเพื่อป้องกันการปิดของแอปพลิเคชัน

เมื่อเราได้ลองเขียนโปรแกรมให้ทำตัวเป็นวินโดว์เซอร์วิสแล้ว (จากหัวข้อย่อย 5.2.1 ถึง 5.2.4 เรื่องการสร้างและติดตั้งวินโดว์เซอร์วิส) ต่อไปจะทำการทดลองเพื่อสร้างวินโดว์เซอร์วิสขึ้นมาเพื่อใช้ในการควบคุมการเปิดของแอปพลิเคชัน โดยมีขั้นตอนการทดลองดังนี้

โดยในการทดลองนี้เราจะทำการสร้างวินโดว์เซอร์วิสเพื่อสั่งรันโปรแกรมเครื่องคิดเลขอัตโนมัติเมื่อเปิดเครื่องและเมื่อมีการปิดโปรแกรมเครื่องคิดเลขไปวินโดว์เซอร์วิสก็จะทำการสั่งรันโปรแกรมเครื่องคิดเลขขึ้นมาใหม่โดยเมื่อสร้างวินโดว์เซอร์วิสขึ้นมาแล้วให้ทำการเพิ่มตัวควบคุมเวลาเข้าไปพร้อมกับเพิ่มเหตุการณ์(Event) Timer.Elapsed ขึ้นมาด้วย โดยตั้ง interval เท่ากับ 5วินาที

```
public Windowsservice()
{
    InitializeComponent();
    System.Timers.Timer t= new System.Timers.Timer(5000);
    t.Start();
    t.Elapsed += new System.Timers.ElapsedEventHandler(timer_interval);
}
```

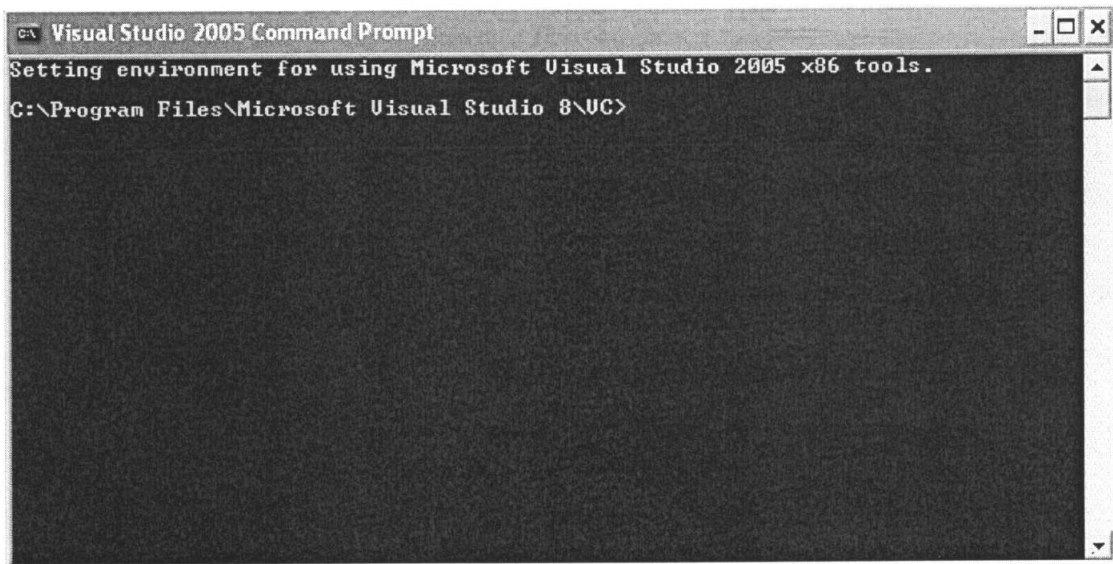
รูปที่ 5.4 แสดงการสร้างตัวควบคุมเวลา

ในเหตุการณ์ timer_interval นั้นเราจะทำการตรวจสอบว่ามีโปรเซสของโปรแกรมเครื่องคิดเลขทำงานอยู่หรือไม่ตลอดทุกครั้งหากครั้งใดตรวจสอบแล้วพบว่าโปรแกรมเครื่องคิดเลขไม่ได้ทำงานอยู่เราจะทำการสร้างโปรเซสเครื่องคิดเลขแล้วสั่งให้มันทำงานขึ้นมาอีกครั้งเรื่อยๆ

```
private void timer_interval(object sender, EventArgs e)
{
    Process[] p = Process.GetProcessesByName("calc");
    if (p.Length == 0)
    {
        Process ps = new Process();
        ps.StartInfo.FileName = @"C:\windows\system32\calc.exe";
        ps.StartInfo.WindowStyle = ProcessWindowStyle.Normal;
        ps.Start();
    }
}
```

รูปที่ 5.5 แสดงการตรวจสอบการทำงานของโปรเซสและการสร้างโปรเซสขึ้นมาทำงาน

จากนั้นทำการ build project ของวินโดวเซอร์วิสทำให้ได้ไฟล์ที่ใช้ในการรันโปรแกรมวินโดวเซอร์วิส (WindowsService1.exe) ทำการติดตั้งโปรแกรมวินโดวเซอร์วิส (WindowService1.exe) โดยใช้เครื่องมือ installUtil.exe ของ .NET เริ่มจากเข้าไปที่ Visual Studio 2005 Command Prompt ดังรูปที่ 5.6



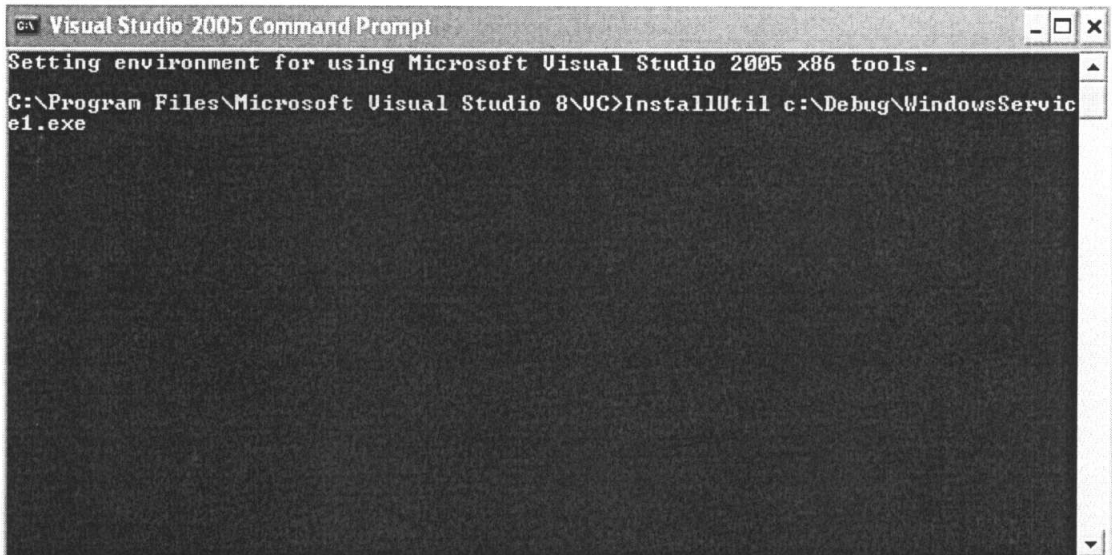
รูปที่ 5.6 Command Prompt เมื่อเข้ามาที่ Visual Studio 2005 Command Prompt

รัน InstallUtil.exe เพื่อทำการติดตั้ง โปรแกรมวินโดวเซอร์วิส โดยใช้คำสั่งดังนี้
>Installutil yourproject.exe {path ของ โปรแกรมวินโดวเซอร์วิสที่ต้องการติดตั้ง}

เช่น `InstallUtil C:\Debug\WindowService1.exe` ดังรูปที่ 5.7 และได้ผลลัพธ์ดังรูปที่ 5.8 หากทำการติดตั้งโปรแกรมวินโดวเซอร์วิสสำเร็จ

หมายเหตุ: หากต้องการยกเลิกการติดตั้งวินโดวเซอร์วิสนี้ ให้ใช้คำสั่ง

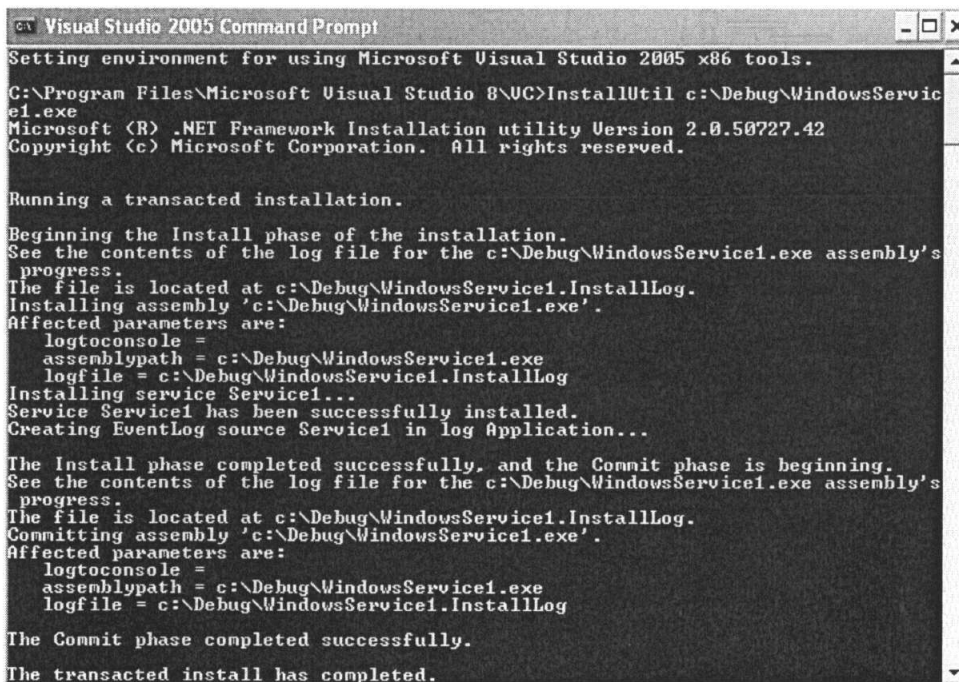
`InstallUtil /u C:\Debug\WindowService1.exe`



```

Visual Studio 2005 Command Prompt
Setting environment for using Microsoft Visual Studio 2005 x86 tools.
C:\Program Files\Microsoft Visual Studio 8\VC>InstallUtil c:\Debug\WindowsService1.exe
  
```

รูปที่ 5.7 การใช้คำสั่งในการติดตั้งโปรแกรมวินโดวเซอร์วิส



```

Visual Studio 2005 Command Prompt
Setting environment for using Microsoft Visual Studio 2005 x86 tools.
C:\Program Files\Microsoft Visual Studio 8\VC>InstallUtil c:\Debug\WindowsService1.exe
Microsoft (R) .NET Framework Installation utility Version 2.0.50727.42
Copyright (c) Microsoft Corporation. All rights reserved.

Running a transacted installation.

Beginning the Install phase of the installation.
See the contents of the log file for the c:\Debug\WindowsService1.exe assembly's progress.
The file is located at c:\Debug\WindowsService1.InstallLog.
Installing assembly 'c:\Debug\WindowsService1.exe'.
Affected parameters are:
  logtoconsole =
  assemblypath = c:\Debug\WindowsService1.exe
  logfile = c:\Debug\WindowsService1.InstallLog
Installing service Service1...
Service Service1 has been successfully installed.
Creating EventLog source Service1 in log Application...

The Install phase completed successfully, and the Commit phase is beginning.
See the contents of the log file for the c:\Debug\WindowsService1.exe assembly's progress.
The file is located at c:\Debug\WindowsService1.InstallLog.
Committing assembly 'c:\Debug\WindowsService1.exe'.
Affected parameters are:
  logtoconsole =
  assemblypath = c:\Debug\WindowsService1.exe
  logfile = c:\Debug\WindowsService1.InstallLog

The Commit phase completed successfully.
The transacted install has completed.
  
```

รูปที่ 5.8 ผลลัพธ์เมื่อทำการติดตั้งโปรแกรมวินโดวเซอร์วิสสำเร็จ

เมื่อเข้าไปที่ Service Control Manager (โดยการกด Run... จากนั้นพิมพ์ services.msc) จะเห็นได้ว่าโปรแกรมวินโดวเซอร์วิสเราได้ทำการติดตั้งเรียบร้อยแล้วดังรูปที่ 5.9 (ชื่อของวินโดวเซอร์วิสที่สร้าง คือ Service1 และตั้ง StartUp Type เริ่มต้นเป็น Automatic)

Service Name	Description	Status	Startup Type	Log On As
Shell Hardware Det...	Provides n...	Started	Automatic	Local System
Service1	Test Windo...		Automatic	Local System
Server	Supports fil...	Started	Automatic	Local System
Security Center	Monitors s...	Started	Automatic	Local System

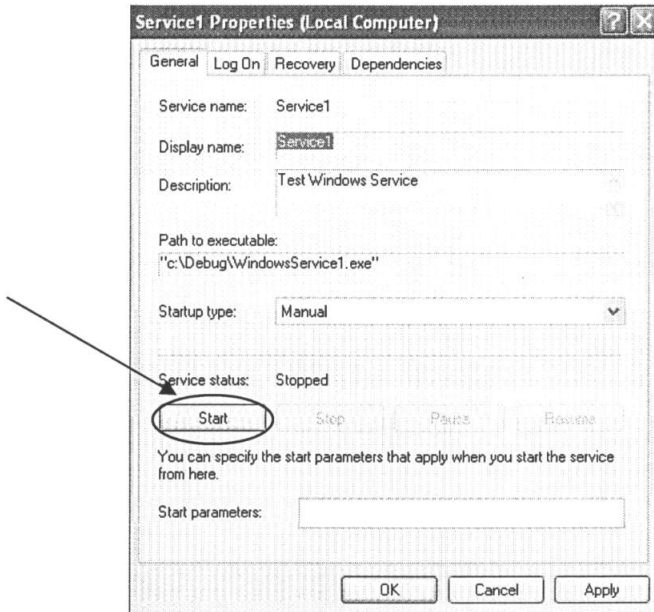
รูปที่ 5.9 วินโดวเซอร์วิสที่ถูกติดตั้งเรียบร้อยแล้ว

เข้าไปยังแท็บ Log ON แล้วเลือก Allow service to interact with desktop เพื่ออนุญาตให้วินโดวเซอร์วิสสามารถติดต่อกับวินโดวอินเทอร์เฟซได้



รูปที่ 5.10 แสดงการตั้งค่า Allow service to interact with desktop

สามารถทำการเริ่ม (Started) ให้วินโดวเซอร์วิสทำงานได้ 2 วิธี คือ 1. ทำการ Restart เครื่องคอมพิวเตอร์ หรือ 2. เข้าไปกดปุ่ม Start เองบน Properties ของวินโดวเซอร์วิสใน Services Control Manager ด้วยตัวเองดังรูปที่ 5.11



รูปที่ 5.11 ปุ่ม Start ใน Properties เพื่อเรียกให้วินโดวเซอร์วิสตัวนี้รัน

เมื่อสั่งให้เซอร์วิสทำงานแล้วจะพบว่าทุกครั้งที่เปิดเครื่องหรือมีการปิดโปรแกรมเครื่องคิดเลขลงไปจะพบวินโดวเซอร์วิสจะทำการสั่งให้รันขึ้นมาใหม่ทุกครั้งเสมอไป

5.3 นิพจน์ปกติ (Regular Expression)

นิพจน์ปกติ (Regular Expression) หมายถึง รูปแบบของลำดับ หรือ กลุ่มของสัญลักษณ์ที่ใช้แทนลำดับ หรือกลุ่มของอักขระตามที่ต้องการ โดยนิพจน์ปกติมักถูกใช้ในเครื่องมือที่ใช้เขียนโปรแกรม มีประโยชน์ช่วยในการค้นหา และจัดการเกี่ยวกับข้อความ กลุ่มอักขระตามที่ต้องการ ซึ่งขึ้นอยู่กับรูปแบบของนิพจน์ปกติที่เรากำหนดไว้

ภาษาที่ใช้ในการเขียนโปรแกรมส่วนมากจะสนับสนุนการใช้นิพจน์ปกติไว้จัดการกับข้อความ หรือที่เรียกว่าสตริง ยกตัวอย่างเช่น ภาษาเพิร์ล (Perl) ซึ่งหลักการของนิพจน์ปกติเป็นที่แพร่หลายครั้งแรกบนระบบปฏิบัติการยูนิกซ์ (Unix)

5.3.1 หลักการพื้นฐาน

นิพจน์ปกติมักจะถูกเรียกว่า รูปแบบ (Pattern) เป็นการแสดงเครื่องหมายที่อธิบาย หรือแทนถึงกลุ่มของอักขระ โดยไม่ต้องมีการเก็บแต่ละตัวอักขระทั้งหมดไว้ ยกตัวอย่างเช่น Billy, Belly และ Berlly สามารถถูกแทนได้โดยรูปแบบ “B(i|e?)lly”

การเลือกอย่างใดอย่างหนึ่ง จะใช้เครื่องหมาย “|” (Vertical bar) เช่น “gray|grey” ซึ่งสามารถจับคู่ได้ทั้ง gray และ grey

เครื่องหมายวงเล็บ “()” ใช้ในการกำหนดขอบเขต รวมกลุ่มของตัวอักขระเข้าไว้ด้วยกัน หรือลำดับความสำคัญของเครื่องหมายได้ ยกตัวอย่างเช่น “gr(a|e)y” สามารถจับคู่ได้ทั้ง gray และ grey

ตัวบ่งบอกถึงจำนวนของตัวอักขระ สามารถกำหนดได้ โดยวางไว้ยู่หลังตัวอักขระหรือกลุ่มของอักขระที่ต้องการ ซึ่งมีเครื่องหมายดังนี้

? (Question mark) หมายถึง ตัวอักขระตัวหน้าสามารถมีได้ 0 (ไม่มี) หรือ 1 ตัวเท่านั้น เช่น “colou?r” แทนได้ทั้ง colour และ color

*** (Asterisk)** หมายถึง ตัวอักขระตัวหน้าสามารถมีได้ 0 (ไม่มี), 1 หรือมากกว่า 1 ตัวก็ได้ เช่น “go*gle” แทนได้ทั้ง ggle, gogle, google, gooogle และ อื่นๆ

+ (Plus sign) หมายถึง ตัวอักขระตัวหน้าจะต้องมีอย่างน้อย 1 ตัวขึ้นไป เช่น “go+gle” แทนได้ทั้ง gogle, google, gooogle และ อื่นๆ (แต่ไม่สามารถแทน ggle ได้)

โดยโครงสร้างเหล่านี้สามารถถูกนำมาใช้ร่วมกันได้ เช่น “((great)*grand)?((fa|mo)ther)” แทนได้ทั้ง father, mother, grand father, grand mother, great grand father, great grand mother, great great grand father, great great grand mother, great great great grand father, great great great grand mother เป็นต้น

5.3.2 สัญลักษณ์ที่ใช้ในนิพจน์ปกติ

. ใช้แทนตัวอักษรอะไรก็ได้ 1 ตัว (ยกเว้นตัวที่บ่งบอกถึงการขึ้นบรรทัดใหม่ “\n”) เช่น “a.cd” แทน abcd และ “a..d” แทน abcd

\ ใช้ใส่หน้าตัวอักษรพิเศษ หรือสัญลักษณ์ต่างๆ เพื่อบอกให้รู้ว่าเครื่องหมายพิเศษที่ใช้ไม่ได้เป็นส่วนหนึ่งของไวยากรณ์ หรือเรียกอีกอย่างหนึ่งว่า escaping ยกตัวอย่างเช่น “a\b” แทนได้กับตัวอักษร a หรือ |b จะเห็นได้ว่าตัวอักษรพิเศษ “\” ที่ตามหลังสัญลักษณ์นี้จะไม่มีความหมายพิเศษ มองเป็นตัวอักษรธรรมดา ในที่นี้คือ ตัว “|” (ตัวอักษร หรือสัญลักษณ์พิเศษที่ต้อง escape มีดังนี้ ^, \$, (,), ., |, *, ?, +, \, และ {)

[] ใช้ระบุตำแหน่งในคำว่า ในตำแหน่งนี้จะประกอบไปด้วยตัวอักษรอะไรได้บ้าง เช่น [abc] แทน “a”, “b” และ “c”, [a-z] แทนตัวพยัญชนะภาษาอังกฤษตัวเล็กได้ทั้งหมด, [abcv-z] แทน “a”, “b”, “c”, “v”, “w”, “x”, “y” และ “z” เป็นต้น

[^] ใช้ระบุตำแหน่งในคำว่า ในตำแหน่งนี้จะต้องไม่ปรากฏตัวอักษรใด เช่น [^abc] แทนตัวอักษรใดก็ได้ที่ไม่ใช่ “a”, “b” และ “c”, [^a-z] แทนตัวอักษรใดก็ได้ที่ไม่ใช่ตัวพยัญชนะภาษาอังกฤษตัวเล็ก เป็นต้น

^ ใช้ระบุตำแหน่งเริ่มต้นของข้อความ เช่น “^big” แทน “big dog”, “big cat” และอื่นๆ ที่มีคำว่า big เป็นคำเริ่มต้นของข้อความ

\$ ใช้ระบุตำแหน่งสุดท้ายของข้อความ เช่น “dog\$” แทน “big dog”, “little dog” และอื่นๆ ที่มีคำว่า dog ลงท้ายสุดของข้อความ

{ } ใช้แทนจำนวนครั้งที่ซ้ำกันได้ เช่น “ab{3}” หมายถึง ให้มีตัวอักษร b จำนวน 3 ตัว นั่นคือ “abbb”, “ab{3,5}” หมายถึง ให้มีตัวอักษร b จำนวน 3-5 ตัว นั่นคือ “abbb”, “abbbb” หรือ “abbbbb”

\n ใช้แทนว่าขึ้นบรรทัดใหม่

\t ใช้แทน Tab หรือ ช่องหน้า

\r ใช้แทน return หรือ CR (Carriage return)

\w ใช้แทน word หรือ ตัวอักษรภาษาอังกฤษทั้งตัวเล็กและตัวใหญ่ ตัวเลข 0-9 รวมไปถึงเครื่องหมาย “_” ซึ่งให้ผลลัพธ์เช่นเดียวกับ [a-zA-Z_0-9]

\W ใช้อะไรก็ได้ที่ไม่ใช่ word หรือ ตัวอักษรภาษาอังกฤษทั้งตัวเล็กและตัวใหญ่ ตัวเลข 0-9 และ “_” ให้ผลลัพธ์เช่นเดียวกับ [^a-zA-Z_0-9]

\s ใช้แทนช่องว่าง หรือ Whitespace ให้ผลลัพธ์เช่นเดียวกับ [\f\n\r\t\v]

\S ใช้แทนอะไรก็ได้ที่ไม่ใช่ช่องว่าง หรือ Whitespace ให้ผลลัพธ์เช่นเดียวกับ [^\f\n\r\t\v]

\xhh ใช้แทนเลขฐานสิบหก (โดย hh ในที่นี้คือ เลขฐานสิบหก)

5.3.3 ตัวอย่างการใช้งาน

“at” จับคู่กับ ตัวอักษร 3 ตัว โดยตัวแรกจะเป็นตัวอักษรอะไรก็ได้ตามด้วย at เช่น hat, bat, cat เป็นต้น

“ $^{\wedge}[^h]at$ ” หมายถึง คำนี้ต้องขึ้นต้นด้วยตัวอักษรอะไรก็ได้ที่ไม่ใช่ h ตามด้วย at

“ $^{\wedge}[1-9][0-9]*\$$ ” หมายถึง ขึ้นต้นด้วยตัวเลข 1-9 และอาจต่อด้วยตัวเลข 0-9 ก็ได้ ในกรณีนี้หากเป็นเลข 0 โดดๆ จะไม่ได้ จะผ่านตั้งแต่ 1, 2, 3, 4 เป็นต้นไป และท้ายตัวเลขชุดนี้ต้องไม่มีตัวอะไรตาม

“ $^{\wedge}(0|[1-9][0-9]*)\$$ ” ความหมายเหมือนกับ “ $^{\wedge}[1-9][0-9]*\$$ ” แต่จะสามารถแทนเลข 0 โดดๆได้ด้วย

“ $^{\wedge}(0|?[1-9][0-9]*)\$$ ” ความหมายเหมือนกับ “ $^{\wedge}(0|[1-9][0-9]*)\$$ ” เพียงแต่ ถ้าไม่ขึ้นต้นด้วยเลข 0 สามารถมีเครื่องหมาย ลบ ได้ หรือจะไม่มีเครื่องหมายลบ ก็ได้ เครื่องหมาย ? บ่งบอกว่า จะมีหรือไม่มี ก็ได้

“ $^{\wedge}[0-9]{1,3}([0-9]{3})*(\.[0-9]{1,2})?\$$ ” หมายถึง ต้องขึ้นต้นด้วยตัวเลข 0-9 จำนวน 1-3 ตัว หรืออาจตามด้วยเครื่องหมายคอมม่า “,” พร้อมกับตัวเลข 0-9 อีก 3 ตัว(ต่อกันก็ชุดก็ได้) และสุดท้ายอาจจะต่อด้วยทศนิยม 1 หรือ 2 ตำแหน่ง หรือไม่ก็ได้ เช่น “0”, “110”, “20,325”, “3.08”, “199,789.00”, “123,456,789.99” เป็นต้น

5.4 การปรับแต่งค่าในรีจิสตรี (Registry Configuration)

รีจิสตรี คือ ที่เก็บค่าที่เกี่ยวข้องกับโปรแกรมต่างๆในวินโดวส์ เช่น การตั้งค่าต่างๆของผู้ใช้ ค่าคิฟอลต์ของระบบ รีจิสตรีจะจัดการเมื่อ โปรแกรมกำลังปิดการทำงานและทางผู้ใช้งานที่ต้องการที่จะเก็บข้อมูลเพื่อนำมาใช้อีกครั้งตอนที่เรียกโปรแกรมขึ้นมา

5.4.1 หลักการที่เกี่ยวข้อง

รีจิสตรีเปรียบเสมือนฐานข้อมูลขนาดใหญ่ ข้อมูลทั้งหมดจะถูกเก็บในรูปแบบที่คล้ายกัน คือในรูปแบบของ Key-value ซึ่งหมายถึงว่าจะมี key และมี value ที่เกี่ยวข้องกันเช่น

Key: "ForeColor"

Value: "Red"

อธิบายได้ว่าเรามี Key คือ ForeColor และมีค่าคือ Red ซึ่งจะมีข้อแตกต่างในรูปแบบที่ใช้ในการเก็บ รีจิสตรี ดังนั้นตามตัวอย่างข้างต้น ForeColor คือค่าที่มีข้อมูลคือ Red

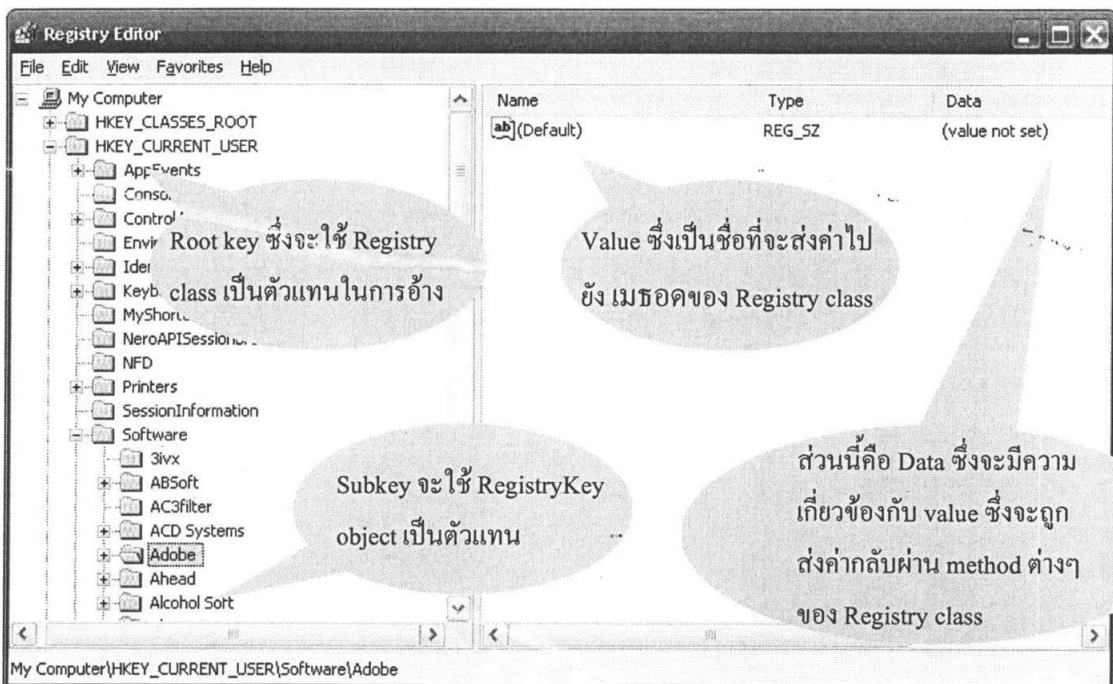
เพื่อที่จะเข้าใจโครงสร้างของรีจิสตรีให้ไปทดลองศึกษาที่เครื่องของท่านเองโดย

Start -> Run -> regedit จะปรากฏหน้าต่างใหม่ขึ้นมาซึ่งจะมีการแสดงค่าต่างๆเป็นแบบโครงสร้าง ไดรากทอรีในทางฝั่งซ้าย ซึ่งจะเก็บรายละเอียดของ values และ data ต่างๆ

วินโดวส์จะเก็บข้อมูลในรีจิสตรีค่อนข้างเยอะมาก นอกจากนี้ข้อมูลของโปรแกรมที่ได้ติดตั้งไปของผู้ใช้คนนั้นๆจะเก็บค่าปรับแต่งต่างๆในรีจิสตรีด้วยเช่นกัน ข้อมูลต่างๆที่เก็บไว้ในรีจิสตรีต้องจัดการให้ดี เพราะวารีจิสตรีจะถูกจัดเก็บไว้แยกกับไฟล์เดออร์ที่เราใช้กันอยู่

อธิบายการเขียนโปรแกรมที่เกี่ยวข้องกับรีจิสตรี

ก่อนอื่นเราต้องทำความรู้จักค่าพื้นฐานของ 4 อีลิเมนต์ที่เก็บอยู่ในรีจิสตรีคือ มี root key, subkey, value, data ซึ่งจะแสดงให้เห็นดังรูป 5.12



รูปที่ 5.12 Registry Editor

ในการเขียน รีจิสตรี ใน .NET จะใช้ 2 class ในการทำงานที่เกี่ยวกับรีจิสตรีนั่นคือ Microsoft.Win32.RegistryKey และ Microsoft.Win32.Registry ซึ่ง RegistryKey class จะถูกใช้เป็นตัวแทนของ subkey ในรีจิสตรี และ Registry class จะถูกใช้เป็นตัวแทนของ root key ในรีจิสตรีเท่านั้น (root key เปรียบเสมือน sub key ชนิดพิเศษซึ่งอยู่ระดับบนขึ้นไป)

5.4.2 การเลือก Subkey

ไม่ว่าจะเราจะต้องการค่าไหนในรีจิสตรี จะต้องเลือกบาง subkey มาก่อนซึ่งในการเลือก subkey เราจะต้องเลือก subkey root ขึ้นมาก่อน ในการเลือก root key เราจะต้องการสร้าง RegistryKey ขึ้นมาใหม่เพื่อที่จะเข้าถึง subkey ที่เราต้องการ

ในตอนที่จะเลือก root key ซึ่งทั้งหมดอยู่ใน Microsoft.Win32.Registry ในการเลือกนั้นจะต้องเลือกใน Microsoft.Win32.Registry class ซึ่งมีดังนี้

ตารางที่ 5.1 รายละเอียดของ subkey root

Field Name	Use	Description
ClassesRoot	to access HKEY_CLASSES_ROOT	ข้อมูล file-types, components และอื่นๆ
CurrentConfig	to access HKEY_CURRENT_CONFIG	ข้อมูลของฮาร์ดแวร์ปัจจุบัน
CurrentUser	to access HKEY_CURRENT_USER	ข้อมูลของผู้ใช้ปัจจุบัน
LocalMachine	to access HKEY_LOCAL_MACHINE	การปรับแต่งค่าของ current machine
Users	to access HKEY_USERS	ข้อมูลการปรับแต่งค่าของผู้ใช้

เมื่อเราเลือก root key แล้วเราต้องทำการสร้าง RegistryKey object เพื่อที่จะเรียก static RegistryKey.OpenSubKey หรือ เมธอด RegistryKey.CreateSubKey บน root key ที่เราเลือกมา การสร้าง subkey ขึ้นมาใหม่

วิธีการสร้าง subkey ด้วยเมธอด RegistryKey.CreateSubKey คือ

```
public RegistryKey CreateSubKey (string subkey);
```

เช่น

...

```
RegistryKey MyReg = Registry.CurrentUser.CreateSubKey
```

```
("SOFTWARE\\SomeCompany\\SomeApp\\SomeVer");
```

...

ในที่นี้เราจะสร้าง subkey ใหม่ใน CurrentUser root key ที่จะส่งค่า RegistryKey object กลับมา

ข้อสังเกต

- ถ้ามี subkey นี้อยู่แล้ว subkey นั้นจะส่งค่านี้กลับมาและไม่มีการสร้างค่าใหม่ ดังนั้นในทางปฏิบัติเราจึงสามารถใช้เมธอด CreateSubKey ในการสร้างและการเปิด subkey
- เมธอด CreateSubKey สามารถส่งค่า subkey กลับมาในการเขียนค่า
- ถ้าเราเรียก CreateSubKey อีกครั้ง (ภายหลังจากการสร้าง subkey) ค่า child subkeys ใหม่จะถูกสร้างขึ้น

5.4.3 การเปิดค่า Subkey ที่มีอยู่แล้ว

ในการเปิดค่า subkey ที่มีอยู่แล้ว เราจะใช้เมธอด RegistryKey.OpenSubKey เช่น

```
RegistryKey MyReg = Registry.CurrentUser.OpenSubKey (...);
```

OpenSubKey จะมีอยู่ 2 โอเวอร์โหลด คือ

```
public RegistryKey OpenSubKey (string name);
```

ซึ่งจะส่งค่า subkey ที่อ่านได้อย่างเดียว

```
public RegistryKey OpenSubKey (string name, bool writable);
```

ซึ่งจะส่งค่า subkey ในโหมดที่ถูกระบุไว้ว่าสามารถเขียนได้หรืออ่านได้อย่างเดียว โหมดที่ใช้ในการเขียนต้องตั้งให้เป็น true และถ้าต้องการให้อ่านได้อย่างเดียวให้เซตเป็น false โดยที่ 2 เมธอดนั้น เราจะต้องส่ง path ที่จะเข้าถึงลงไป ถ้าพบจะส่งค่า RegistryKey object กลับมาถ้าไม่พบจะส่งค่า null กลับมา

เช่น

...

```
RegistryKey MyReg = Registry.CurrentUser.OpenSubKey
```

```
    ("SOFTWARE\\SomeCompany\\SomeApp\\SomeVer", true);
```

...

ส่งค่า RegistryKey object กลับมาจาก CurrentUser root key กับการระบุ path ที่จะใช้ในการเขียน กับ subkey ที่มีอยู่

5.4.4 การเรียกค่าที่เก็บอยู่มาใช้

เมื่อเราได้ค่า RegistryKey object แล้วเราสามารถเข้าถึงค่า subkey และ data ได้โดยการใช้ เมธอด RegistryKey.GetValue ดังนี้

```
someRegistryKeyobject.GetValue (...);
```

GetValue มีอยู่ 2 โอเวอร์โหลดคือ

```
public object GetValue (string name);
```

เมธอด นี้จะส่งค่าที่เป็นตัวแทนของ subkey หรือ ค่า null ถ้า value ไม่มีอยู่

```
public object GetValue (string name, object defaultValue);
```

เมธอด นี้จะส่งค่าที่เป็นตัวแทนของ subkey หรือค่า default ที่ระบุไว้โดย defaultValue

จะเห็นว่า GetValue จะส่งข้อมูลกลับมาเป็นชนิด object ดังนั้นเราสามารถทำการ cast ชนิดของ ข้อมูลที่ต้องการ

5.4.5 การปรับแก้ค่าของข้อมูล

ในการเซตค่าเราจะใช้เมธอด RegistryKey.SetValue ดังนี้

```
public void SetValue (string name, object value);
```

string name จะเก็บ value ไว้ส่วนค่าของ object value จะเก็บ data ที่เราต้องการ แล้วจะแปลงค่าเป็น DWORD, binary หรือ string นี่คือการผลลัพท์ทำไม เมธอด RegistryKey.GetValue จะส่งข้อมูล ออกมาใน แบบ type object

5.4.6 การปิด Subkey

เมื่อเราได้เข้าถึง subkey แล้วเราจะต้องปิดการใช้งานด้วย โดยการให้เมธอด

RegistryKey.Close สำหรับการปิด subkey เช่น

```
RegistryKey MyReg = Registry.CurrentUser.CreateSubKey
    ("SOFTWARE\\SomeCompany\\SomeApp\\SomeVer");
int nSomeVal = (int)MyReg.GetValue("SomeVal", 0);
MyReg.SetValue ("SomeValue", nSomeVal+1);
MyReg.Close ();
```

5.4.7 การลบ Subkey

ในการลบ subkey จะมี 2 โอเวอร์โหลดของเมธอด RegistryKey.DeleteSubKey

```
public void DeleteSubKey(string subkey);
```

จะทำการลบ subkey ที่ระบุไว้

```
public void DeleteSubKey(string subkey, bool throwOnMissingSubKey);
```

จะทำการลบ subkey ที่ระบุไว้ และถ้าไม่พบ subkey จะ throwOnMissingSubKey ถ้าเซตค่าเป็น true

5.4.8 การลบ Entire tree

การลบ subkey และ subkey ย่อยๆรวมทั้ง ค่าอื่นๆทั้งหมด เราจะต้องใช้เมธอด

RegistryKey.DeleteSubKeyTree เช่น

```
public void DeleteSubKeyTree(string subkey);
```

5.4.9 การลบ Value

ในการลบ Values จะมีเมธอดอยู่ 2 โอเวอร์โหลดคือ

```
public void DeleteValue(string name);
```

โดยให้เราระบุ Value กับตัวแปร name ซึ่งจะเป็นตัวที่ถูกลบ

และ

```
public void DeleteValue(string name, bool throwOnMissingValue);
```

เมธอดนี้ก็เช่นกันแต่จะมีการ thrown exception ถ้าค่า throwOnMissingValue เป็น true

5.4.10 การรับค่า List of subkeys

การรับค่า List of subkeys จะใช้เมธอด RegistryKey.GetSubKeyNames คือ

```
public string[] GetSubKeyNames();
```

5.4.11 การรับค่า list of value names

การรับค่า list of value names จะใช้เมธอด RegistryKey.GetValueNames

```
public string[] GetValueNames();
```


5.5 การเขียนโปรแกรมเพื่อใช้งานการเข้ารหัสข้อมูลแบบ RSA (RSA Encryption)

เป็นรูปแบบการเข้ารหัสข้อมูลด้วยวิธีการเข้ารหัสกุญแจสาธารณะ ซึ่งการเข้ารหัสแบบนี้จะมีกุญแจแบ่งออกเป็น 2 ตัวด้วยกันคือ กุญแจสาธารณะ(Public key) และกุญแจส่วนตัว(Private key) โดยกุญแจสาธารณะนั้นมีไว้เพื่อเข้ารหัสข้อมูลส่วนกุญแจส่วนตัวนั้นมีไว้เพื่อถอดรหัสข้อมูลซึ่งเราเรียกวิธีการแบบนี้ว่าการเข้ารหัสแบบอสมมาตร(Asymmetric key cryptography)

ขั้นตอนการสร้างการเข้ารหัสแบบ RSA ด้วยภาษา C#.NET นั้นสามารถทำได้โดยการใช้คลาส RSACryptoServiceProvider เข้ามาช่วยโดยการใช้ คลาส RSACryptoServiceProvider นั้นต้องทำการประกาศ System.Security.Cryptography ก่อนถึงจะสามารถใช้งานได้

5.5.1 การสร้างคลาส RSACryptoServiceProvider

การสร้างคลาสนั้นสามารถทำได้โดยการ CspParameters ขึ้นมาเพื่อรับค่ารหัสเริ่มต้น โดยในตัวอย่างนี้จะให้รับค่าจากการสุ่มค่าสตริงแล้วนำไปสร้างเป็นคลาส RSACryptoServiceProvider

```
public crypton()
{
    CspParameters cp = new CspParameters();
    cp.KeyContainerName = RandomString(10, true);
    RSACryptoServiceProvider rsaEncryption = new RSACryptoServiceProvider(cp);
}
```

รูปที่ 5.13 แสดงโค้ดการสร้าง RSACryptoServiceProvider

5.5.2 การสร้างกุญแจสาธารณะ(Public key) และกุญแจส่วนตัว(Private key)

ในการสร้างกุญแจทั้งสองนั้นสามารถทำออกมาได้ให้อยู่ในรูปแบบของข้อความ XML โดยมีวิธีการสร้างได้ดังนี้

```
public void makeKey(RSACryptoServiceProvider rsaEncryption)
{
    privateKey = rsaEncryption.ToXmlString(true);
    publicKey = rsaEncryption.ToXmlString(false);
}
```

รูปที่ 5.14 แสดงโค้ดการสร้างกุญแจสาธารณะ(Public key) และกุญแจส่วนตัว (Private key)

```

<RSAKeyValue>
  <Modulus>
    tFrAZ/cEMwMrbdGCeP17sVvUXqubROCPeR0ZHPxVv8TrQVfGceUAaggjaDwDvDsVjfBo69+u6vEOuAnKGvYVxs6L1F
    F-RBcfMEnxjsLnWaP1jk1jK5vTGNXImp399KyQc-3q5+5x26BiTheMeGL5LLgmYhz3i4LH16aYiI5r51L2A5c=
  </Modulus>
  <Exponent>AQAB</Exponent>
  <P>
    6HcIF3Va0QtufikaiSY9nfdFvrmW4qMKayNngjQ39eIdzI-M5P5o04piZ5Tqnd70aQ6Q4pEvwx3bc9YN4qrc3Fw==
  </P>
  <Q>
    xpwIYjLOVd8e--2kY8HChdX6WcHaT1WvZ1ev0I3uhR9tXr7Udj-u2noIonF27dIgeF8Qm6Pu23mQX4jBMEyQFSocQ==
  </Q>
  <DP>
    ejYCxPtnUTsfct8u3GolyFrrhPqiVRO2/h2Nrvvw542G1QJ22fR9q7-m2DnL1k9eQtTripE1gOkbr5w7FI0rK5Q==
  </DP>
  <DQ>
    FumUzVdzDcYh1mL9D4y1xtLN13rQVps16AxRn1009qlxqTEoCm11P661WDjVfUkgS98Gon7PBLf/UH57qn5QgQ==
  </DQ>
  <InverseQ>
    sjkzMpsH109PP4Q5sHksv1IRAjp3MdWU-qahUUGb6kgA2F3e7mwQnAJTyEVrSswvKb988pIdlXimk+4mrxDkxg==
  </InverseQ>
  <D>
    DyhL5SD3RWJK0Bv/nz00pcVQ3SQRD7E0CUF60251zgxXfzMR9R/5+d5xa6x7j0Z23xmt1c1Uct1r7RXDR07g6Wd3
    owrk6hCjk1XRX0b/J0n+hCuxX9zcanGAE1YYL0p6rNbabZ51gP0kNwM/4BnQz39aUB14zFdNW2sEtJ+uhwE=
  </D>
</RSAKeyValue>

```

รูปที่ 5.15 แสดงตัวอย่างของกุญแจส่วนตัว (Private key)

```

<RSAKeyValue>
  <Modulus>
    m4Xj4fevWztd1nw1JFm+QhKib+XNh+DmtVCoxsV733tMiXAiQxmVmzbrDcmAvqkuIVmL1YWjSq01Zn+Qnyrh5
    TycHwcb7bLXpFov8AjEOt+eN3WkiIG7F2hdGaLkTa9U9bHCdLcmRmYwxDpuFCcbpyRkN5a3JPOEeEuqcxLR2os=
  </Modulus>
  <Exponent>AQAB</Exponent>
</RSAKeyValue>

```

รูปที่ 5.16 แสดงตัวอย่างของกุญแจสาธารณะ (Public key)

5.5.3 การเข้ารหัสข้อมูลด้วยวิธี RSA

การเข้ารหัสนั้นสามารถทำได้ด้วยการนำกุญแจสาธารณะ (Public key) นั้นมาใช้ในการเข้ารหัสข้อมูล โดยสามารถทำได้โดยทำการสร้างคลาส RSACryptoServiceProvider จากกุญแจสาธารณะที่ได้มาแล้วนำไปเข้ารหัสข้อมูล

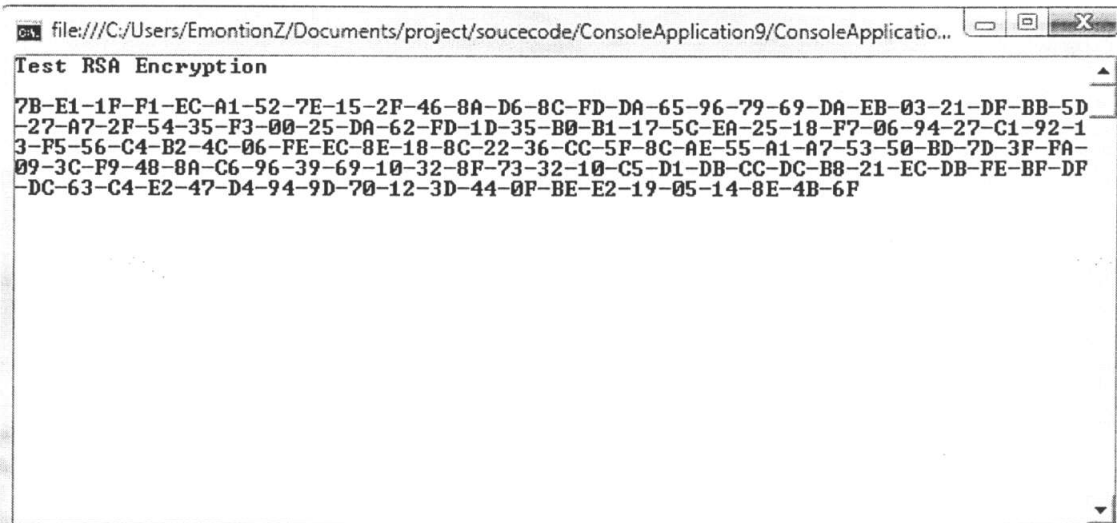
```

static void Main()
{
    StreamReader sr = new StreamReader ("C:\t.xml");
    string key = sr.ReadToEnd();
    RSACryptoServiceProvider rsaEncryption = new RSACryptoServiceProvider();
    rsaEncryption.FromXmlString (key);

    string dataString = "Test RSA Encryption";
    byte[] dataByte = ASCIIEncoding.ASCII.GetBytes (dataString);
    byte[] encrypt_byte = rsaEncryption.Encrypt (dataByte, true);
}

```

รูปที่ 5.17 แสดงตัวอย่างการเข้ารหัสข้อมูลด้วยกุญแจสาธารณะ (Public key)



รูปที่ 5.18 แสดงตัวอย่างคำว่า “Test RSA Encryption” เมื่อถูกเข้ารหัส

5.5.4 การถอดรหัสข้อมูลด้วย RSA

การถอดรหัสนั้นสามารถทำได้ด้วยการนำกุญแจส่วนตัว (Private key) นั้นมาใช้ในการถอดรหัสข้อมูลโดยสามารถทำได้โดยการสร้างคลาส RSACryptoServiceProvider จากกุญแจส่วนตัว (Private key) ที่มีอยู่แล้วนำไปถอดรหัสข้อมูล

```
static void Main()
{
    StreamReader sr = new StreamReader(@"C:\t.xml");
    string key = sr.ReadToEnd();
    sr.Close();
    StreamReader srr = new StreamReader(@"C:\e.txt");
    string dataString = srr.ReadToEnd();
    srr.Close();
    RSACryptoServiceProvider rsaEncryption = new RSACryptoServiceProvider();
    rsaEncryption.FromXmlString(key);

    byte[] dataByte = ASCIIEncoding.ASCII.GetBytes(dataString);
    byte[] data_Decrypt = rsaEncryption.Decrypt(dataByte, true);
}
```

รูปที่ 5.19 แสดงตัวอย่างการถอดรหัสข้อมูลด้วย RSA

```

file:///C:/Users/EmotionZ/Documents/project/soucecode/ConsoleApplication9/ConsoleApplicatio...
93-B2-E7-3B-F9-14-34-65-72-76-96-40-7F-2F-85-50-95-DF-CE-F3-58-BB-EA-12-C7-57-41
6B-C5-89-0E-1B-31-57-0F-A3-23-F4-3B-B8-69-D3-E8-FB-02-55-56-69-97-C3-76-7B-96-0
A-61-6C-D6-DE-5A-89-C8-93-AB-49-B9-9C-50-B8-BB-8D-4A-CC-6B-F9-48-0D-A1-BE-0D-F5-
E2-95-CA-E0-AF-30-AE-23-DF-AE-2C-93-1A-45-78-42-C5-FE-F1-DC-0C-60-38-E8-87-07-17
0F-A1-80-4A-05-8F-43-3A-35-14-ED-3A-20-32-33-22-76-B6-15-3D-8E

Test RSA Encryption

```

รูปที่ 5.20 แสดงตัวอย่างคำว่า “Test RSA Encryption” เมื่อถูกถอดรหัส

5.6 การทดลองสร้างระบบป้องกันการเชื่อมต่อของแอปพลิเคชัน

ในส่วนนี้ระบบจะแบ่งออกเป็น 2 ส่วนด้วยกันคือในส่วนของการดึงข้อมูลการเชื่อมต่อของแต่ละโพรเซส (Process) ขึ้นมา และส่วนของการปิดกั้นข้อมูลการเชื่อมต่อ

5.6.1 การเขียนโปรแกรมเพื่อดึงข้อมูลการเชื่อมต่อ Getting active TCP/UDP connections

การเขียนโปรแกรมในส่วนนี้ได้ใช้ IpHelperApi.dll ซึ่งเป็น win32 API โดย API ตัวนี้มีฟังก์ชันการทำงานหลักๆ อยู่ 6 ฟังก์ชันด้วยกันคือ

- GetUdpStats()
- GetTcpStats()
- GetUdpConnexions()
- GetTcpConnexions()
- GetExUdpConnexions()
- GetExTcpConnexions()

ในฟังก์ชันที่ 1 และ 3 จะทำงานเหมือนกันคือแสดงข้อมูลการเชื่อมต่อของ UDP โปรโตคอล ส่วนฟังก์ชันที่ 2 และ 4 จะแสดงข้อมูลการเชื่อมต่อของ TCP โปรโตคอล ส่วนในฟังก์ชันที่ 5 และ 6 นั้นจะทำงานคล้ายกับฟังก์ชันข้างต้นที่กล่าวจะต่างกันตรงที่ 2 ฟังก์ชันนี้มีความพิเศษที่สามารถแสดงค่าของ ProcessID ออกมาได้ด้วยทำให้เราสามารถรู้ได้ว่า Process ใดเชื่อมต่อไปยังที่ใดบ้าง

ในการเขียนโปรแกรมในส่วน Getting active TCP/UDP connections นั้นเราจะใช้ฟังก์ชัน GetExTcpConnexions() ของ IpHelperApi.dll จะทำให้เราได้ข้อมูลว่าในเครื่องมี Process ใดทำงานอยู่บ้างแล้วทำการเชื่อมต่อไปยังที่ใดบ้าง

```

IpHlpApidotnet.IPHelper MyAPI = new IpHlpApidotnet.IPHelper();
MyAPI.GetExTcpConnexions();
for (int i = 0; i < MyAPI.TcpExConnexions.dwNumEntries; i++)
{
    Console.WriteLine(i + ": " + MyAPI.TcpExConnexions.table[i].ProcessName + ".exe" +
        ">>>>" + MyAPI.TcpExConnexions.table[i].Remote.Address.ToString() + ":" +
        + MyAPI.TcpExConnexions.table[i].Remote.Port.ToString());
}

```

รูปที่ 5.21 โค้ดการเขียนโปรแกรมโดยใช้ GetExTcpConnexions()

```

file:///C:/Documents and Settings/Eemotionz/Desktop/test2/bin/Debug/test2.EXE
0: inetinfo.exe>>>>0.0.0.0:0
1: inetinfo.exe>>>>0.0.0.0:0
2: suchost.exe>>>>0.0.0.0:0
3: inetinfo.exe>>>>0.0.0.0:0
4: System.exe>>>>0.0.0.0:0
5: rapimg.exe>>>>0.0.0.0:0
6: inetinfo.exe>>>>0.0.0.0:0
7: utorrent.exe>>>>0.0.0.0:0
8: System.exe>>>>0.0.0.0:0
9: msmsgs.exe>>>>207.46.107.37:1863
10: Idle.exe>>>>65.54.179.203:443
11: zClientm.exe>>>>207.46.166.10:80
12: IEXPLORE.exe>>>>204.13.51.238:80
13: zClientm.exe>>>>207.46.166.107:28805
14: zClientm.exe>>>>204.160.118.124:80
15: zClientm.exe>>>>207.68.178.239:80
16: alg.exe>>>>0.0.0.0:0
17: wcescomm.exe>>>>0.0.0.0:0
18: wcescomm.exe>>>>0.0.0.0:0

```

รูปที่ 5.22 ตัวอย่างโปรแกรมที่ถูกเขียนด้วย GetExTcpConnexions()

5.6.2 การเขียนโปรแกรมในส่วนของการปิดกั้นการเชื่อมต่อ (Packet Filtering)

ในการเขียนโปรแกรมในส่วนนี้ได้ใช้ Packet Filtering API เข้ามาช่วยในการเขียน โดย API ตัวนี้มีอยู่ใน Windows XP แล้ว Windows 2000 ซึ่งใน API ตัวนี้จะประกอบไปด้วย 2 คลาส หลักๆ ก็คือ

- TxFilterController คลาสของ Interface ที่ต้องการใช้งาน
- TxIpFilter คลาสของกฎที่จะใช้กับ Interface ที่ต้องการ

ในคลาสของ TxIpFilter นั้นเราสามารถกำหนดกฎต่างๆลงไปได้ดังนี้

- flt.direction ทิศทางของข้อมูล ขาเข้า,ขาออก
- flt.ipSource IP ของต้นทางของข้อมูล
- flt.maskSource Mask ของ IP ในกรณีที่ต้องการกำหนดเป็นช่วงของ IP
- flt.ipDestination IP ของปลายทางของข้อมูล

- flt.maskDestination Mask ของ IP ในกรณีต้องการกำหนดเป็นช่วงของ IP
- flt.protocol รูปแบบของโปรโตคอล TCP,UDP,ICMP
- flt.sourcePort Port การเชื่อมต่อของต้นทาง
- flt.destinationPort Port การเชื่อมต่อของปลายทาง

ขั้นตอนการใช้งาน Packet Filtering API นั้นมี 3 ขั้นตอนดังนี้

- อันดับแรกต้องรู้ก่อนว่า IP ของ NIC ที่ต้องการใช้นั้น IP อะไรหลังจากนั้นให้เราทำการสร้าง Interface ขึ้นมาแล้วทำการ Bind IP นั้นให้แก่ interface ที่สร้างขึ้น
- ทำการใส่ กฎ (Filtering Rule) ลงไปตามที่ต้องการ
- หลังจากใช้งานแล้วให้ทำการ Unbind และลบ interface นั้นทิ้ง

```

TxFilterController fltCont = new TxFilterController();
TxIpFilter flt = new TxIpFilter();

flt.direction = TxIpFilter.OUT_DIRECTION;
flt.ipSource = IPAddress.Parse("10.0.0.32");
flt.maskSource = IPAddress.Parse("255.255.255.255");
flt.ipDestination = IPAddress.Parse("10.0.0.254");
flt.maskDestination = IPAddress.Parse("255.255.255.255");
flt.protocol = TxIpFilter.ICMP_PROTOCOL;
flt.sourcePort = TxIpFilter.ANY_ICMP_TYPES;
flt.destinationPort = TxIpFilter.ANY_ICMP_TYPES;
fltCont.AddFilter(IPAddress.Parse("10.0.0.32"), flt);
Console.WriteLine("Program Running .....");
Console.ReadLine();
fltCont.CloseController();

```

รูปที่ 5.23 ตัวอย่างโค้ดที่ใช้ Packet Filtering API เขียน

5.7.2 การใช้งานคณาตรวจสอบความเหมาะสมของรูปภาพ

การใช้งานในส่วนนี้ทำได้โดยสร้างคลาส Class_image ขึ้นมาแล้วทำการนำเข้าไฟล์รูปลงไปแล้วทำการตรวจสอบว่ามีขนาดเล็กเกินไปหรือไม่หากไม่ทำการวิเคราะห์และรีเทิร์นค่าผ่านฟังก์ชัน getResult()

```
static void Main()
{
    Bitmap image = Image.FromFile(@"C:\picture.jpg");
    Class_image identifys = new Class_image();
    identifys.importimage(image);

    if (!(identifys.is_sizeicon(img)))
    {
        identifys.checkpornographic();
        string resultreceive = identifys.getResult();
    }
}
```

รูปที่ 5.26 แสดงตัวอย่างการใช้งานคณาวิเคราะห์ความเหมาะสมของรูปภาพ

บทที่ 6

การทดลองและผลการทดลอง

6.1 ทดสอบประสิทธิภาพของระบบรวบรวมและวิเคราะห์เอกสารเว็บไซต์

การทดสอบนี้ได้ถูกทดสอบบนระบบที่ข้อกำหนดพื้นฐานดังนี้

CPU AMD X2 ความเร็ว 1.8 GHz L2 Cache 1MB

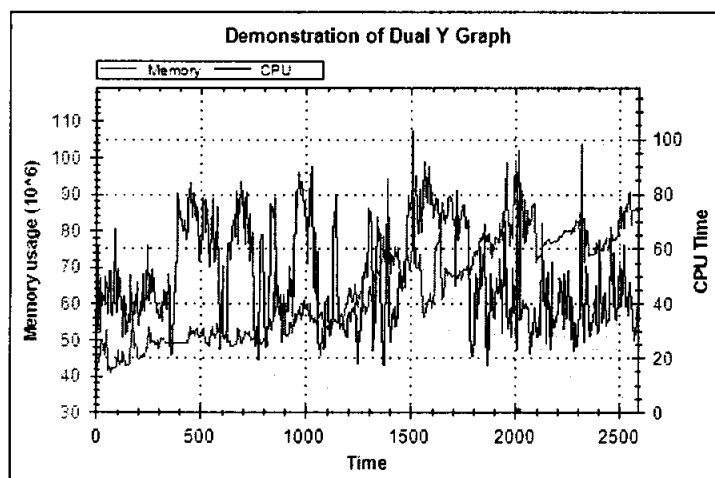
RAM DDR2 ความเร็ว 533 MHz ความจุ 2 GB

อินเทอร์เน็ต ADSL ความเร็ว 512/256

โดยในตารางที่ 6.1 นั้นจะเป็นผลการทดสอบประสิทธิภาพการทำงานของระบบในการรวบรวมเอกสารจากอินเทอร์เน็ตเพื่อดูความเร็วและทรัพยากรที่ใช้ในการทำงาน

ตารางที่ 6.1 แสดงข้อมูลของระบบเมื่อใช้งานเฉพาะส่วนรวบรวมเอกสาร

อัตราการ Crawler	จำนวน URL ที่ค้นพบ	Memory Usage	CPU Usage
7449 page/hour	119751 URL/hour	65.38 Mbytes	63.25%



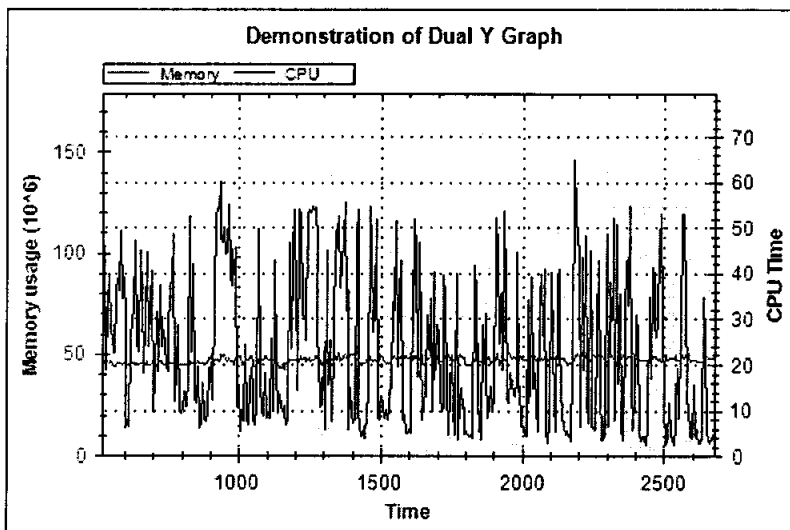
รูปที่ 6.1 กราฟแสดงข้อมูลค่าการใช้ CPU และ Memory เมื่อใช้งานเฉพาะส่วนรวบรวมเอกสาร

จากกราฟและตารางที่ 6.1 จะพบว่าระบบนั้นทำงานได้เร็วพอสมควรทั้งนี้ขึ้นอยู่กับความเร็วอินเทอร์เน็ตเป็นสำคัญด้วยและจากกราฟยังแสดงให้เห็นอีกว่าการใช้งาน CPU อยู่ในระดับที่สูงตลอดเวลาเนื่องจากการทำงานนั้นเป็นแบบระบบ มัลติเทรด (Multi-Thread) ทำให้มีการใช้งาน CPU ตลอดเวลาส่วนในด้านของเมมโมรีนั้นจะมีการใช้งานเพิ่มขึ้นเรื่อยๆจนถึงจุดๆหนึ่งจะคงที่เนื่องจากที่ถ่ายข้อมูลลงสื่อบันทึกก่อนเมื่อมีจำนวนขนาดใหญ่หลายๆเพื่อป้องกันการใช้งานเมมโมรีจนเต็ม

ตารางที่ 6.2 และกราฟที่ 6.2 นั้นจะเป็นผลการทดสอบประสิทธิภาพการทำงานของระบบในการวิเคราะห์เอกสารจากอินเทอร์เน็ตเพื่อดูความเร็วและทรัพยากรที่ใช้ในการทำงาน

ตารางที่ 6.2 แสดงข้อมูลของระบบเมื่อใช้งานเฉพาะส่วนวิเคราะห์เอกสาร

อัตราการวิเคราะห์	Memory Usage	CPU Usage
252 page/hour	48.72 MBytes	27.58%

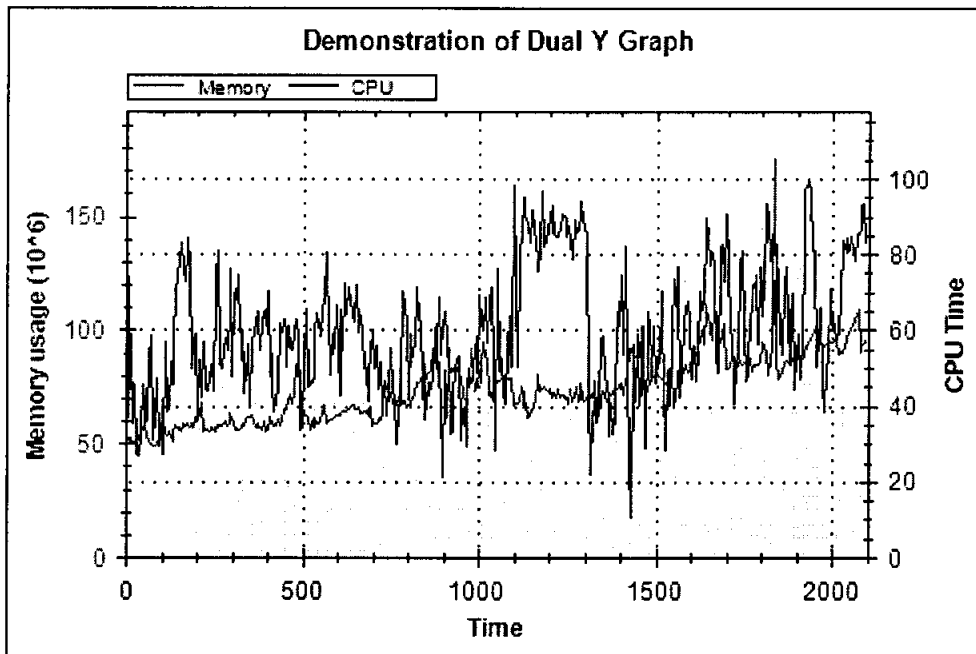


รูปที่ 6.2 กราฟแสดงข้อมูลค่าการใช้ CPU และ Memory เมื่อใช้งานเฉพาะส่วนวิเคราะห์เอกสาร

จากตารางและกราฟที่ 6.2 จะเห็นได้ว่าการทำงานของระบบนี้จะช้ามากเนื่องจากต้องมีการวิเคราะห์หลายส่วนรวมไปถึงการที่ระบบต้องเสียเวลาในการดึงเอกสารเพื่อเคมจากอินเทอร์เน็ต เช่น รูปภาพจากกราฟจะเห็นว่าทำงานของ CPU จะขึ้นลงตลอดเวลาแล้วมีการใช้งานเมมโมรี่อย่างคงที่เนื่องจากระบบนี้มีการทำงานแบบเทรคเดียวทั้งการดำเนินงานที่ได้ช้าลงไปด้วย

ตารางที่ 6.3 แสดงข้อมูลของระบบเมื่อใช้งานทั้ง 2 ส่วน

อัตราการ Crawler	จำนวน URL ที่ค้นพบ	อัตราการวิเคราะห์	Memory Usage	CPU Usage
6135 page/hour	87323URL/hour	132 page/hour	96.42 MBytes	80.58%



รูปที่ 6.4 กราฟแสดงข้อมูลค่าการใช้ CPU และ Memory เมื่อใช้งานทั้ง 2 ส่วน

6.2 ทดสอบความถูกต้องของระบบวิเคราะห์เว็บไซต์

การทดสอบนั้นจะแบ่งออกเป็น 2 ส่วนคือทดสอบกับเว็บไซต์ที่ปกติและทดสอบกับเว็บไซต์ที่เป็นเว็บไซต์อนาจารเพื่อดูความถูกต้องในการแบ่งแยกและวิเคราะห์ประเภทของเว็บไซต์

6.2.1 ทดสอบกับเว็บไซต์ปกติ

ได้มีการนำเว็บไซต์จำนวน 100 เว็บไซต์ที่มีเนื้อหาแตกต่างกันไปในแต่ละเว็บโดยทั้งหมดนั้นเป็นเว็บไซต์ปกติทั่วไปนำมาเข้าสู่ระบบวิเคราะห์เว็บไซต์ผลที่ได้ออกมาเป็นดังนี้

ตารางที่ 6.5 แสดงผลการทดสอบกับเว็บไซต์ปกติ

จำนวนเว็บไซต์ที่ใช้	ตรวจเป็นเว็บปกติ	ตรวจเป็นเว็บอนาจาร	ประสิทธิภาพ	ความผิดพลาด
100 เว็บไซต์	99 เว็บไซต์	1 เว็บไซต์	99%	1%

ตารางที่ 6.6 แสดงผลการทดสอบกับเว็บไซต์ปกติ

รายชื่อเว็บไซต์ (ไม่อนาจาร) คัดกรองได้อย่างถูกต้อง 99 เว็บ	IMG	META	KEY-WORD	SCRIPT	ALL-LINK	IMG-LINK	TXT-LINK	LINK-NON	LINK-RELA	PARAM
http://en.wikipedia.org/wiki/Wiki	15	3	0	9	318	15	303	125	193	0
http://hopeanuoli.cjb.net/	0	0	0	3	6	0	6	5	1	0
http://load-info.us/	1	2	0	0	202	1	201	3	199	0
http://offthemat.net/	2	10	0	0	6	1	5	2	4	0
http://postjung.com/	90	3	0	12	158	40	118	77	81	0
http://teenee.com/	377	1	0	8	340	17	323	326	4	8
http://top.greek-sites.eu	69	7	0	1	122	40	82	71	51	0
http://top-online.us/	2	2	0	1	202	1	201	3	199	0
http://www.212cafe.com/	65	3	0	10	104	16	88	46	58	4
http://www.365jukebox.com/	106	9	0	6	445	41	404	177	268	14
http://www.abitur-werne.de/blog	2	2	0	16	152	2	150	94	58	0
http://www.aios.org/	119	1	0	1	53	11	42	13	40	0
http://www.amazon.com/	62	2	0	5	283	50	233	3	280	0
http://www.appjob.com/	392	5	0	5	402	121	281	129	273	0
http://www.apple.com/	29	6	0	11	34	23	11	3	31	0
http://www.asiasoft.co.th/	34	1	0	5	12	3	9	12	0	0
http://www.bbc.co.uk/thai/	62	8	0	4	58	17	41	33	25	0
http://www.bbznnet.com/	36	3	0	2	18	5	13	12	6	0
http://www.bloggang.com/	17	2	0	14	195	5	190	70	125	0
http://www.cnn.com/	259	5	0	33	259	70	189	32	227	0
http://www.cool50.eu/	113	6	0	1	238	62	176	138	100	0
http://www.craftercity.com/	70	7	0	0	150	58	92	2	148	0
http://www.dailynews.co.th/	232	8	0	16	167	62	105	4	163	0

ตารางที่ 6.6 แสดงผลการทดสอบกับเว็บไซต์ปกติ(ต่อ)

รายชื่อเว็บไซต์(ไม่อนุจาร) คัดกรองได้อย่างถูกต้อง	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://www.defendingarnhem.com/	21	3	0	1	38	16	22	7	31	0
http://www.dek-d.com/	421	1	0	13	453	72	381	146	307	0
http://www.doctorsan.com/	256	7	0	13	241	43	198	83	158	4
http://www.downloadget.com/	18	6	0	9	47	7	40	24	23	0
http://www.elovedates.com/	104	8	0	20	168	101	67	128	40	0
http://www.exteen.com/	31	1	0	0	63	1	62	55	8	2
http://www.filevalley.com/	34	4	0	15	136	13	123	22	114	0
http://www.freeflashonlinegames.com/	50	4	0	24	74	12	62	4	70	0
http://www.freehitwebcounters.com/	70	2	0	28	90	17	73	15	75	0
http://www.freehitwebsitecounter.com/	60	2	0	27	88	10	78	13	75	0
http://www.free-palm-soft.com/	31	4	0	45	126	19	107	2	124	0
http://www.free-pocket-pc-games.com/	25	4	0	42	58	12	46	2	56	0
http://www.freeproxyserverlist.com/	13	4	0	42	15	0	15	1	14	0
http://www.gmember.com/	211	5	0	12	100	26	74	59	41	45
http://www.google.com/	0	1	0	0	1	0	1	1	0	0
http://www.gr50.com/	7	6	0	0	17	6	11	17	0	0
http://www.greekmeeting.com/	17	6	0	1	25	16	9	25	0	0
http://www.hartl.info/	9	20	0	5	40	9	31	25	15	0
http://www.hunsa.com/2005/	287	3	0	8	376	45	331	335	41	0
http://www.ini3.co.th/	212	3	0	9	62	18	44	39	23	0
http://www.jabchai.com/	196	4	0	9	234	94	140	180	54	11
http://www.kapook.com/	294	3	0	13	31	111	200	292	19	4
http://www.kirstiealleyfan.com/	0	4	0	0	0	0	0	0	0	0
http://www.komchadluek.net/	399	4	0	8	186	86	100	71	115	55
http://www.lyricsmusicongs.com/	11	4	0	44	63	0	63	1	62	0
http://www.manager.co.th/	871	1	0	15	600	172	428	142	458	9
http://www.marketathome.com/	228	5	0	8	287	50	237	107	180	46
http://www.matichon.co.th/	160	4	0	5	132	51	81	10	122	23
http://www.meemodel.com/home/	80	3	0	2	263	9	254	261	2	4
http://www.meesook.com/	119	2	0	19	386	29	357	271	115	2
http://www.moohin.com/	364	0	0	7	301	45	256	235	66	2
http://www.mthai.com/	200	6	0	14	442	88	354	393	49	4
http://www.musicatm.com/	133	2	0	15	81	20	61	31	50	6
http://www.narak.com/	228	6	0	5	243	65	178	102	141	4
http://www.pantip.com/	99	1	0	19	296	81	215	146	150	2
http://www.pantipmarket.com/	124	1	0	6	341	40	301	331	10	2
http://www.passaparolach.com/	56	2	0	9	172	31	141	169	3	0
http://www.play-bit.com/	43	4	0	47	109	28	81	1	108	0
http://www.playpark.com/	14	1	0	4	8	3	5	4	4	0
http://www.pramool.com/	199	3	0	4	136	29	107	67	69	0
http://www.rheuma.de/	20	7	0	3	19	12	7	3	16	0
http://www.sangabrielmission.org/	24	0	0	16	38	20	18	1	37	0
http://www.sanook.com/	250	5	0	24	375	40	335	343	32	0
http://www.scandicon.net/	21	1	0	3	5	1	4	0	5	0
http://www.secretsofredgate.com/	6	3	0	3	20	2	18	1	19	0
http://www.sf.in.th/intro/	14	1	0	4	8	3	5	4	4	0
http://www.short-love-poem.com/	25	4	0	21	75	15	60	60	15	0

ตารางที่ 6.6 แสดงผลการทดสอบกับเว็บไซต์ปกคิ(ต่อ)

รายชื่อเว็บไซต์(ไม่อนาจาร) กักรองได้อย่างถูกต้อง	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://www.siamphone.com/	7	8	0	7	31	4	27	26	5	0
http://www.siamza.com/home/	591	3	0	33	279	50	229	212	67	0
http://www.siamzone.com/	180	3	0	14	284	81	203	5	279	3
http://www.softwarewin.com/	89	13	0	26	161	23	138	22	139	0
http://www.storythai.com/	86	6	0	16	111	11	100	96	15	0
http://www.swisswatchstation.com/	34	24	0	31	51	11	40	20	31	0
http://www.tarad.com/	353	9	0	14	367	108	259	205	162	0
http://www.thaibg.com/	62	5	0	6	98	17	81	7	91	3
http://www.thaiload.com/	21	2	0	2	138	0	138	14	124	8
http://www.ThaiSecondhand.com/	112	3	0	4	331	53	278	286	45	3
http://www.thaiware.com/	246	11	0	12	134	38	96	53	81	6
http://www.limber-merchants-importers.com/	52	3	0	1	75	8	67	38	37	0
http://www.toolshed-ps.com/	22	6	0	9	31	14	17	3	28	0
http://www.truehits.net/	96	2	0	3	148	18	130	141	7	6
http://www.ttonline.net/	349	18	0	6	243	27	216	116	127	10
http://www.uploadtoday.com/	33	3	0	9	12	4	8	5	7	0
http://www.vadino.com/	61	5	0	27	110	57	53	40	70	0
http://www.vividroom.com.au/	21	6	0	3	41	11	30	2	39	0
http://www.void.be/	2	3	0	0	13	0	13	8	5	0
http://www.w4lgh.com/	1	4	0	0	3	1	2	0	3	0
http://www.yenta4.com/	274	7	0	17	395	91	304	207	188	3
http://www.dellawear.de/	14	8	1	1	9	7	2	2	7	0
http://www.t-scripts.com/	2	2	1	0	21	2	19	21	0	0

ตารางที่ 6.7 แสดงผลการทดสอบกับเว็บไซต์ปกคิที่ผิดพลาด

รายชื่อเว็บไซต์ (ไม่อนาจาร) กักรองผิดพลาด 1 เว็บ	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://www.boxsat.com/	58	7	4	11	189	45	144	140	49	0

จากการทดลองจะพบมีข้อผิดพลาดที่มีเว็บไซต์ปกคิถูกวิเคราะห์ออกมาว่าเป็นเว็บไซต์อนาจารจากการตรวจสอบเว็บที่ผิดพลาดจึงพบว่าเว็บไซต์นั้นเป็นเว็บไซต์ขายอุปกรณ์เสริม PCMCIA ซึ่งพบว่าในเว็บไซค์นั้นมีคำว่า Cam เป็นจำนวนมากทำให้ระบบนั้นเข้าใจผิดวิเคราะห์ออกมาเป็นเว็บไซต์อนาจาร

6.2.2 ทดสอบกับเว็บไซต์อนาจาร

ได้มีการนำเว็บไซต์จำนวน 100 เว็บที่มีเนื้อหาเป็นเว็บไซต์อนาจารไปนำมาเข้าสู่ระบบวิเคราะห์เว็บไซต์ผลที่ได้ออกมาเป็นดังนี้

ตารางที่ 6.8 แสดงผลการทดสอบกับเว็บไซต์อนจาร

จำนวนเว็บไซต์ที่ใช้	ตรวจเป็นเว็บปกติ	ตรวจเป็นเว็บอนจาร	ประสิทธิภาพ	ความผิดพลาด
100 เว็บไซต์	4 เว็บไซต์	96 เว็บไซต์	96%	4%

ตารางที่ 6.9 แสดงผลการทดสอบกับเว็บไซต์อนจาร

รายชื่อเว็บไซต์ (อนจาร) คัดกรองได้อย่างถูกต้อง 96 เว็บ	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://clothing-mania.com/	175	0	1	1	503	100	403	32	471	0
http://ohgood.com/	3	0	0	6	1206	1	1205	16	1190	0
http://oh-teen.com/	878	2	5	5	917	0	917	44	873	0
http://spunkysheets.com/	164	4	11	1	161	96	65	109	52	0
http://straight.theater.moviesmonster.com/	53	6	0	8	211	52	159	8	203	4
http://www.247pantyhose.com/	17	2	1	1	362	16	346	58	304	0
http://www.adult-series.com/	169	9	5	3	180	39	141	151	29	0
http://www.ah-me.com/	244	5	3	5	424	49	365	34	390	0
http://www.apieceofass.com/	4	2	4	28	49	4	45	46	3	0
http://www.aroundtheworldsex.com/	10	6	9	6	248	7	241	32	216	0
http://www.asstraffic.com/	35	0	3	2	34	14	20	18	16	0
http://www.babesglamour.com/	136	3	4	3	202	107	95	197	5	0
http://www.bestxxxseries.com/	1	3	3	7	1574	0	1574	90	1484	0
http://www.bizarre-rituals.com/	10	4	10	11	0	394	4	390	112	282
http://www.blogchalking.com/life/psm/forced-big-cock-sex/	20	8	6	0	187	0	187	184	3	0
http://www.bootae.com/	127	13	9	5	261	122	139	9	252	0
http://www.candylist.com/	14	6	3	2	350	12	338	334	16	0
http://www.chubbyland.com/	10	3	3	3	273	4	269	194	79	0
http://www.cleancafe.com/	8	4	18	4	728	6	722	30	698	0
http://www.crazynakedchick.com/	251	4	2	1	394	18	376	166	228	0
http://www.crazy-series.com/	96	8	3	1	237	95	142	237	0	0
http://www.dariusporn.com/	99	11	9	1	637	98	539	17	620	0
http://www.el-ladies.com/	77	3	4	3	85	66	19	81	4	0
http://www.eurothumb.com/	194	22	5	7	1201	177	1024	29	1172	0
http://www.fmgalleries.com/	55	3	9	4	209	10	199	88	121	0
http://www.footster.net/	6	5	3	8	10	4	6	8	2	0
http://www.freebigmovies.com/	417	12	14	4	557	32	525	357	200	0
http://www.freepicseries.com/	11	4	3	6	545	5	540	60	485	0
http://www.freepornpictures.org/	81	8	12	2	531	37	494	100	431	0
http://www.frogsex.com/	1	1	0	5	928	0	928	81	847	0
http://www.fuckcat.com/	10	3	3	0	852	0	852	9	843	0
http://www.fullvidz.com/	45	3	5	2	670	6	664	628	42	0
http://www.greenthumbstgp.com/	8	15	23	5	56	4	52	56	0	0
http://www.hankshoneys.com/	15	10	9	5	6	5	1	6	0	0
http://www.hardsex-movies.com/	164	8	3	10	315	159	156	128	187	0
http://www.hornygrannysex.com/	587	4	5	4	660	13	647	86	574	0

ตารางที่ 6.9 แสดงผลการทดสอบกับเว็บไซต์อนาจาร(ต่อ)

รายชื่อเว็บไซต์(อนาจาร) คัดกรองได้อย่างถูกต้อง (ต่อ)	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://www.hotfreesex4all.com/	268	8	5	5	396	218	178	66	330	0
http://www.hotfunhouse.com/	10	2	6	2	411	5	406	9	402	0
http://www.huggernet.com/	99	3	3	6	230	60	170	167	63	0
http://www.imagefest.com/	4	2	0	28	48	4	44	46	2	0
http://www.infospiral.com/category/mature/	257	2	3	1	101	0	101	90	11	0
http://www.ixixi.com/	21	11	6	4	2029	21	2008	8	2021	0
http://www.jennysbookmarks.com/	5	3	5	2	740	0	740	13	727	0
http://www.justseries.com/	1	9	9	0	2	0	2	1	1	0
http://www.legs-fetish.com/	5	3	4	4	26	2	24	26	0	0
http://www.legs-fetish.com/thumbnailpost/	23	4	5	1	753	15	738	620	133	0
http://www.leg-zone.com/	7	15	13	4	4	1	3	2	2	0
http://www.mature.dk/	4	2	7	15	18	1	17	1	17	0
http://www.meoland.com/	30	8	12	7	1530	27	1503	1389	141	0
http://www.nslit.com/	22	3	7	8	660	19	641	642	18	0
http://www.old69.com/	0	3	4	4	1212	0	1212	1123	89	0
http://www.onefullmovie.com/	21	2	2	5	64	9	55	57	7	0
http://www.pandalist.com/	27	2	2	2	89	24	65	89	0	0
http://www.pinkvisualtgp.com/	65	3	10	2	118	55	63	62	56	0
http://www.pornfoundation.com/	16	13	24	3	941	0	941	1	940	0
http://www.pornpixxxs.com/	59	4	5	10	669	57	612	25	644	0
http://www.premiumxxxcore.com/	9	8	14	9	393	6	387	37	356	0
http://www.pued.com/	22	5	13	3	971	2	969	123	848	0
http://www.pushinpussy.com/	5	7	19	3	346	5	341	222	124	0
http://www.pussydivine.com/	4	6	8	7	7	2	5	7	0	0
http://www.realxworld.com/	1	3	4	1	766	0	766	43	723	0
http://www.series-xxx.com/	13	3	7	2	6	0	6	5	1	0
http://www.sicklist.com/	8	10	3	3	1861	4	1857	5	1856	0
http://www.smutcastle.com/	0	3	5	2	2	0	2	1	1	0
http://www.spankstation.com/	1	12	12	2	941	0	941	7	934	0
http://www.stocking-mania.com/	8	2	4	1	5	4	1	4	1	0
http://www.stocking-movies.com/	60	3	6	4	241	57	184	240	1	0
http://www.stockings-world.com/	68	13	4	7	139	47	92	127	12	0
http://www.stocking-tease.com/	234	4	5	7	526	126	400	524	2	0
http://www.teeniesxxx.com/	232	3	6	4	245	145	100	27	218	0
http://www.tgplibrary.com/	4	1	5	0	7	0	7	6	1	0
http://www.tiava.com/	2	3	4	1	1298	2	1296	17	1281	0
http://www.tommys-bookmarks.com/	4	5	7	2	58	2	56	30	28	0
http://www.transmgp.com/	215	8	5	7	292	166	126	82	210	0
http://www.twilightsex.com/	296	3	6	5	475	62	413	284	191	0
http://www.uniformslutzone.com/	0	7	2	1	49	0	49	49	0	0
http://www.vibe.com/	35	3	1	14	188	23	165	43	145	0
http://www.videoshome.com/	128	4	3	3	884	8	876	353	531	0
http://www.voyeurzine.com/	3	10	3	3	602	1	601	599	3	0
http://www.worldsex.com/	67	4	8	15	605	64	541	488	117	0
http://www.x-3.net/	6	4	4	2	734	4	730	15	719	0
http://www.xxxvoque.net/	256	2	1	5	1668	102	1566	14	1654	0
http://www.yusex.com/	8	2	2	2	29	7	22	28	1	0
http://www.between-legs.com/	15	3	0	4	158	5	153	156	2	0

ตารางที่ 6.9 แสดงผลการทดสอบกับเว็บไซต์อนาจาร(ต่อ)

รายชื่อเว็บไซต์(อนาจาร) คัดกรองได้อย่างถูกต้อง (ต่อ)	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://www.puppykibble.com/	170	1	0	2	192	141	51	141	51	0
http://www.redhothoneys.com/	47	1	0	3	73	27	46	6	67	0
http://www.thumbclub.com/	1	0	0	1	359	1	358	0	359	0
http://www.uniformpost.com/	97	0	0	6	133	92	41	14	119	0
http://www.vixena.it/	52	6	0	3	135	37	98	95	40	0
http://www.wolflist.com/	2773	1	0	1	2819	0	2819	1	2818	0
http://www.x-mint.com/	121	7	0	4	327	120	207	0	327	0
http://www.xxbeauties.com/	119	1	0	1	345	111	234	136	209	0

ตารางที่ 6.10 ตารางแสดงการทดสอบกับเว็บไซต์อนาจารที่ผิดพลาด

รายชื่อเว็บไซต์(อนาจาร) คัดกรองผิดพลาด 4 เว็บ	IMG	META	KEY- WORD	SCRIPT	ALL- LINK	IMG- LINK	TXT- LINK	LINK- NON	LINK- RELA	PARAM
http://stockingfever.com/	5	0	0	3	242	1	241	88	154	0
http://www.kongolist.com/	9	1	0	33	34	2	32	5	29	0
http://www.libraryofthumbs.com/	1	3	0	0	2	0	2	2	0	0
http://www.stockingpost.com/	4	0	0	1	2	1	1	1	1	0

จากผลการทดลองจะเห็นว่าเว็บไซต์จำนวนหนึ่งที่ผิดพลาดจากการตรวจสอบมาเมื่อตรวจสอบจึงพบว่าเว็บที่ตรวจสอบผิดพลาดนั้นเป็นเว็บที่มีลักษณะเป็นเว็บรวมลิงค์ซึ่งในหน้านั้นไม่มีคำหยาบคายอยู่ใน <Title> และ <META> เลขร่วมไปถึงยังไม่มีรูปภาพอนาจารใดๆด้วยจึงทำให้ผ่านการตรวจสอบไปได้

บทที่ 7

บทวิจารณ์และสรุป

7.1 บทสรุป

โครงการนี้เป็นโครงการใหม่ที่ได้จัดทำขึ้นมา โดยประกอบด้วย 3 ส่วนหลักๆ ดังนี้

1. ระบบ Crawler & Web analysis ซึ่งทำหน้าที่รวบรวมเอกสาร HTML จากทั่วโลกมาวิเคราะห์ว่าเป็นเว็บอนาจารหรือไม่แล้วทำการส่งให้ตัวแอปพลิเคชันเพื่อใช้ในการอ้างอิงในการใช้งาน
2. แอปพลิเคชันทำหน้าที่ป้องกันและจัดการการใช้งานอินเทอร์เน็ตโดยสามารถป้องกันการเข้าใช้งานเว็บอนาจารหรือเกมออนไลน์ที่ไม่อนุญาตได้
3. เว็บไซต์ควบคุมไว้สำหรับผู้ปกครองที่ไม่ได้อยู่ที่บ้านต้องการดูผลการใช้งานอินเทอร์เน็ตของเด็กๆหรือต้องการเปลี่ยนการตั้งค่าของแอปพลิเคชัน

7.2 ปัญหาอุปสรรคและแนวทางในการแก้ไข

1. ประสิทธิภาพของการทำงานของระบบ crawler และ web analysis นั้นขึ้นอยู่กับความเร็วของเครื่องคอมพิวเตอร์และอินเทอร์เน็ต
 2. การป้องกันการปิดการใช้งานของแอปพลิเคชันจากผู้ที่ไม่ได้รับอนุญาตยังไม่สามารถทำได้อย่างสมบูรณ์ หาก แอปพลิเคชันนั้นถูกใช้งานอยู่ในผู้ใช้งานระดับ Administrator
 3. การทำงานของ crawler และ web analysis ยังช้าอยู่มากเนื่องจากต้องใช้การประมวลผลที่สูงและอินเทอร์เน็ตที่มีความเร็วสูงหากเว็บไซต์ใดมีรูปจำนวนมากๆจะทำให้การทำงานช้าลงมาก
 4. การวิเคราะห์หน้าเว็บไซต์ยังมีข้อผิดพลาดหรือคลาดเคลื่อนไป เพราะอันเนื่องมาจาก
ในเว็บเพจแต่ละหน้ามีโครงสร้างที่หลากหลายทำให้จำแนกได้ยากกว่าเป็นเว็บไซต์ลามกอนาจารหรือไม่ ทำให้โปรแกรมอาจเกิดความผิดพลาดได้ เนื่องจากเนื้อหาในเว็บไซต์ที่ไม่ใช่เว็บไซต์ลามกอนาจารบางเว็บไซต์ อาจมีคุณสมบัติใกล้เคียงกับเว็บไซต์ลามกอนาจาร
- 4.2 กรณีการพิจารณาคำศัพท์ว่าเป็นคำที่ไม่เหมาะสมหรือไม่ จะเกิดข้อผิดพลาดก็ต่อเมื่อ
- คำๆ นั้นตรงกับชื่อเฉพาะต่างๆ เช่น ชื่อเมือง ชื่อคน
 - คำๆ นั้นตรงกับเนื้อหาที่อยู่ในเว็บ Wikipedia หรือประเภท Dictionary

- คำๆ นั้นตรงกับเนื้อหาที่อยู่ในเว็บเกี่ยวกับเพศศึกษา ซึ่งเราสามารถแก้ปัญหาในส่วนนี้ได้โดยมีการเพิ่มชื่อชื่อ URL ที่น่าเชื่อถือ (Trusted URL) ลงในโปรแกรม และไม่กรองเว็บไซต์ที่ประกอบไปด้วย Domain Name ที่ได้รับความน่าเชื่อถือ อาทิ เช่น .edu, .ac.th, .co.th, .go.th, .gov เป็นต้น

7.3 แนวทางการพัฒนาต่อ

1. พัฒนาในส่วนของ web analysis ให้ทำงานแบบ multi-threading อย่างสมบูรณ์แบบเพื่อเพิ่มความเร็วในการทำงานของระบบ
2. ปรับปรุงการทำงานของส่วน crawler และ web analysis ในใช้ทรัพยากรน้อยลงกว่านี้เพื่อเพิ่มประสิทธิภาพในการทำงานของระบบ
3. พัฒนาให้สามารถรองรับการทำงานได้ในหลายๆ โพรโทคอล เช่น https และ ftp
4. พัฒนาให้ระบบสามารถแยกแยะอันตรายอื่นๆ จากเว็บไซต์ได้เพิ่มขึ้นอีก เช่น Spyware
5. พัฒนาให้ระบบสามารถเก็บข้อมูลด้านอื่นๆ ของการใช้งานคอมพิวเตอร์ได้เพิ่มขึ้นเช่นการใช้ Instant Messenger

บรรณานุกรม

RFC Documentation [online]. Available:

<http://www.freesoft.org/CIE/RFC/2068/index.htm>

MSDN Documentation [online]. Available:

<http://msdn2.microsoft.com/en-us/default.aspx>

The Code Project - Free Source Code and Tutorials [online]. Available:

<http://www.codeproject.com>

The number one the number one developer site [online]. Available:

<http://www.codeguru.com/>

Rafael C. Gonzalez / Richard E. Woods, Digital Image Processing, Prentice Hall PTR

Andrew Krowczk / Vinod Kumar / Nauman Laghari / Ajit Mungse / Christian Nagel /
Tim Parker / Srinivasa Sivakumar , Professional .NET Network Programming, WROX
Press

W.H Ho, P.A. Watters “Identifying and Blocking Pornographic Content”

ASP.Net examples (example source code) Organized by topic [online]. Available:

<http://www.java2s.com>

C# Corner- C-Sharp C#.NET CSharp VB.NET ASP.NET Visual Studio .NET [online].

Available:<http://www.c-sharpcorner.com/>

Christophe Garcia, Georgios Tziritas “Face Detection Using Quantized Skin Color Regions
Merging and Wavelet Packet Analysis”, IEEE Trans. Multimedia, vol.1, no.3, PP. 264-
277, September 1999.

Rein-Lien Hsu, Mohamed Abdel-Mottaleb, Anil K. Jain “Face Detection in Color Images”,
IEEE Transaction on pattern analysis and machine intelligence, Vol. 24 No.5 May 2002

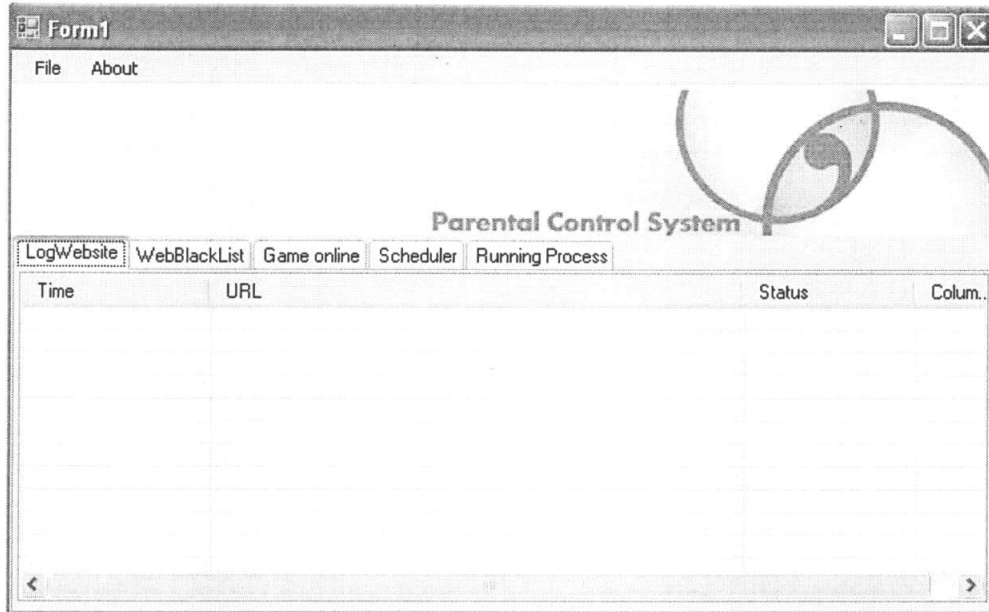
Qing-Fang Zheng , Wei Zeng , Gao Wen , Wei-Qiang Wang “Shape-based Adult Image

Detection”, Proceedings of the Third International Conference on Image and Graphics
(ICIG’04)

พฤกษ์ ษานานุสนธิ์, วรวิทย์ ลีลาประเสริฐวงศ์. 2549 “โปรแกรมคัดกรองข้อมูลเว็บ.” ปรินิพนธ์
วิศวกรรมศาสตร์ สาขาวิชาวิศวกรรม คอมพิวเตอร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้า
คุณทหารลาดกระบัง.

ภาคผนวก
คู่มือการใช้ซอฟต์แวร์

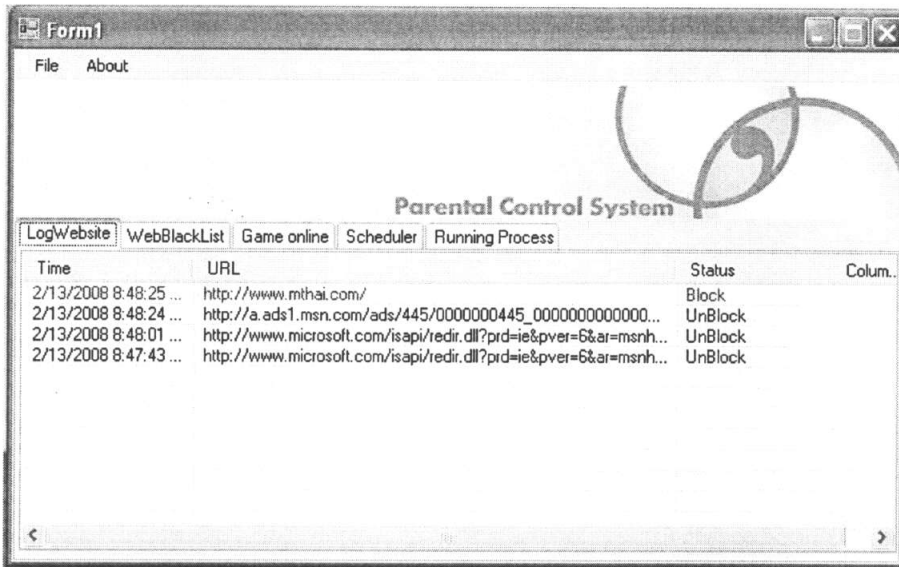
คู่มือการใช้ซอฟต์แวร์ (User Manual)



ซอฟต์แวร์ตัวนี้นั้น ได้มีเมนูต่างๆอยู่ด้วยกัน 5 เมนู โดยแบ่งเป็น

- Log Website แสดงข้อมูลบันทึกการใช้งานเว็บไซต์ต่างๆ
- WebBlackList แสดงข้อมูล BlackList ของเว็บไซต์
- Game Online แสดงข้อมูลของ List ของเกมออนไลน์
- Scheduler ตั้งค่าข้อมูลการจัดสรรเวลาใช้งานอินเทอร์เน็ต
- Running Process แสดงข้อมูลโปรแกรมที่มีการเชื่อมต่ออินเทอร์เน็ต

Log Website

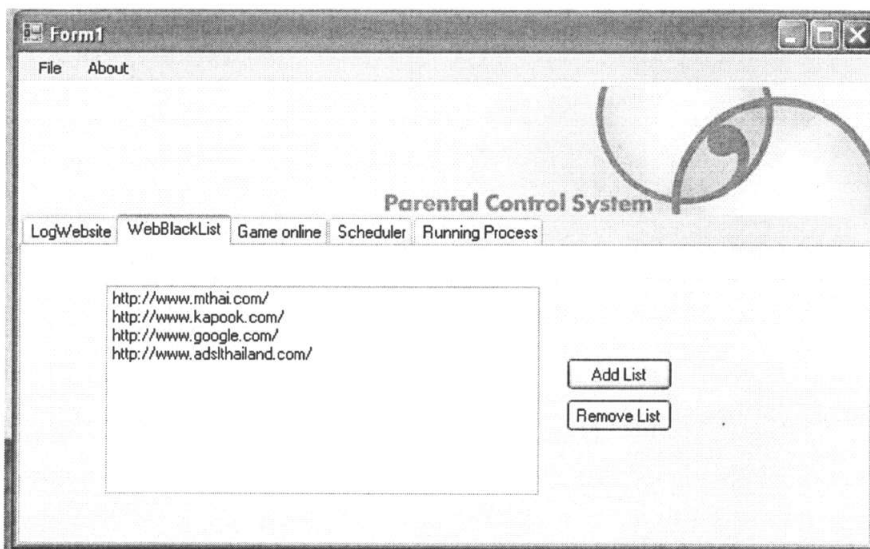


The screenshot shows the 'LogWebsite' tab in the Parental Control System. It displays a table with the following data:

Time	URL	Status	Column
2/13/2008 8:48:25 ...	http://www.mthai.com/	Block	
2/13/2008 8:48:24 ...	http://a.ads1.msn.com/ads/445/0000000445_000000000000...	UnBlock	
2/13/2008 8:48:01 ...	http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnh...	UnBlock	
2/13/2008 8:47:43 ...	http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnh...	UnBlock	

Log Website นั้นจะแสดงข้อมูลการเข้าใช้งานเว็บไซต์ต่างๆ โดยจะแสดงของมูลวันที่เข้าใช้ URL และสถานะว่า สามารถเข้าใช้งานได้หรือไม่ หากติดสถานะ Block จะปรากฏเป็นตัวสีแดงขึ้นมา

WebBlackList



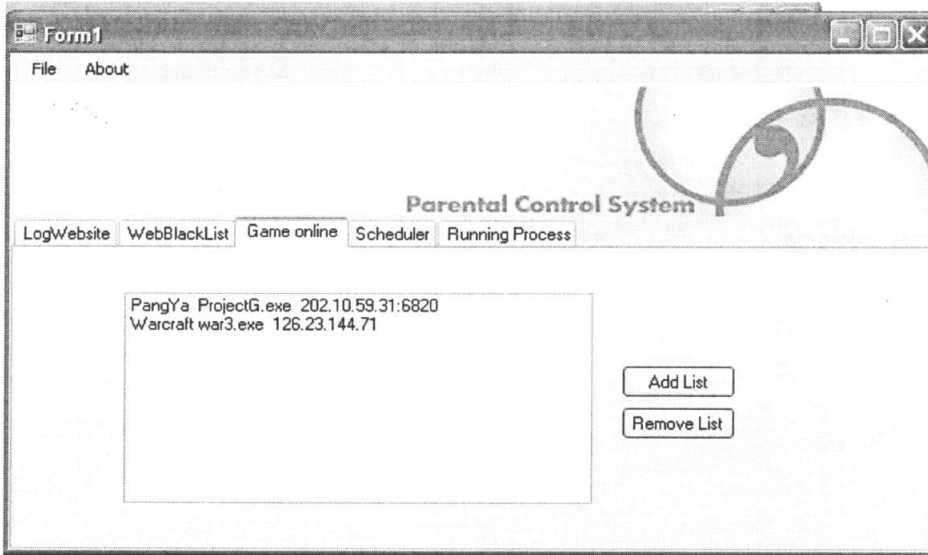
The screenshot shows the 'WebBlackList' tab in the Parental Control System. It displays a list of blocked URLs in a text box:

- http://www.mthai.com/
- http://www.kapook.com/
- http://www.google.com/
- http://www.ads1thailand.com/

Below the list are two buttons: 'Add List' and 'Remove List'.

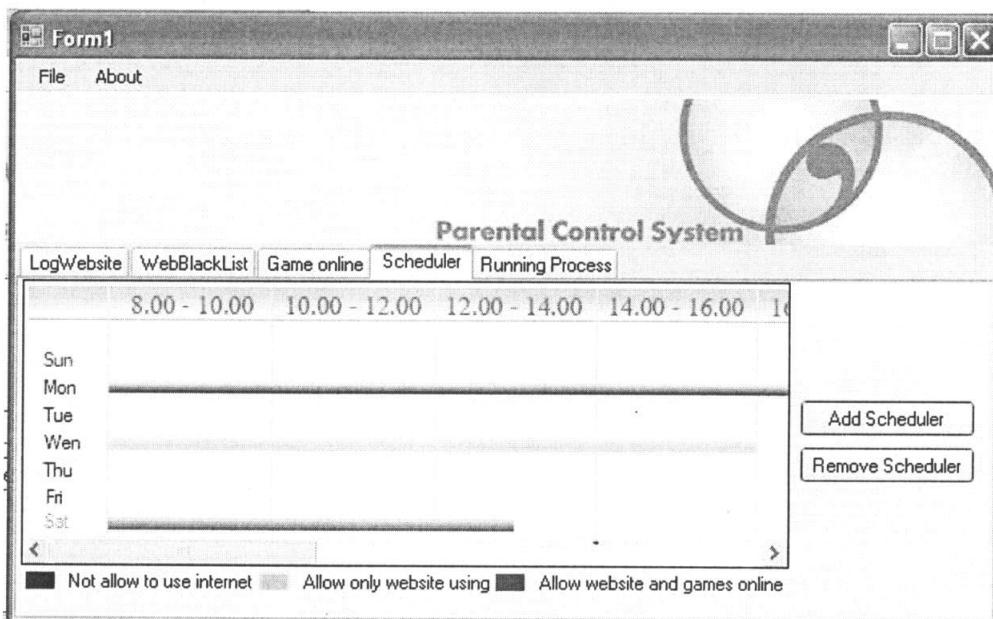
WebBlackList จะเป็นส่วนที่แสดงข้อมูลเว็บไซต์ต้องห้ามที่มีอยู่โดยผู้ใช้สามารถเพิ่มเติมหรือเอาออกได้ที่ ปุ่ม Addlist และ Remove List

Game Online



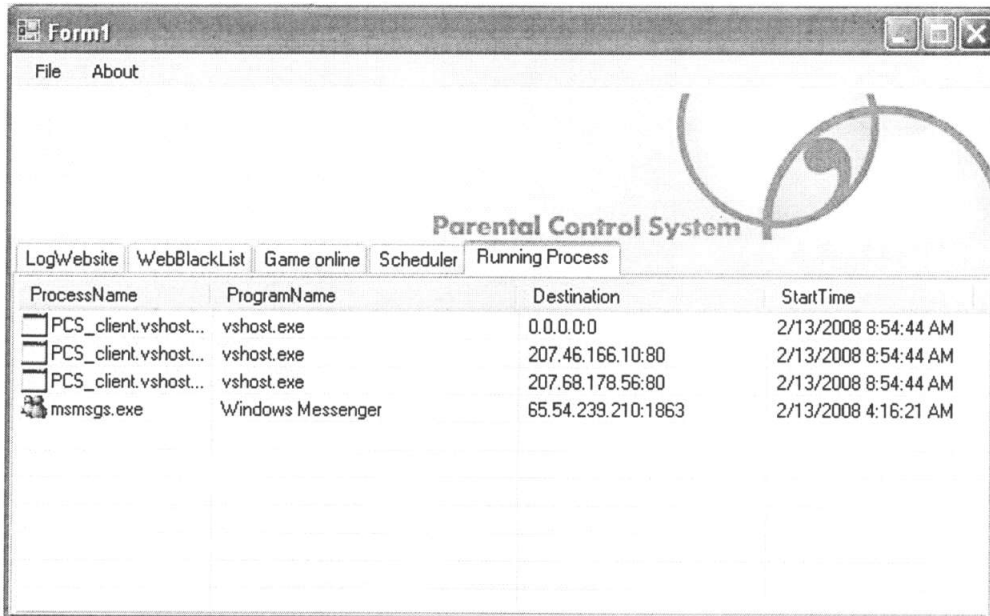
Game Online เป็นส่วนที่แสดงข้อมูลของโปรแกรมที่ห้ามเชื่อมต่ออินเทอร์เน็ตโดยผู้ใช้งาน สามารถเพิ่มเติมหรือเอาออกได้ที่ ปุ่ม Add List และ Remove List

Scheduler

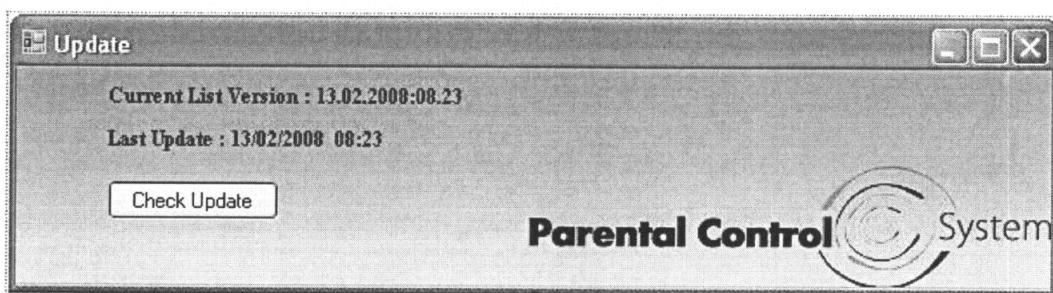


Scheduler เป็นส่วนที่ใช้ในการตั้งค่าเพื่อกำหนดการใช้งานเล่นอินเทอร์เน็ตโดยจะมีให้เลือก 3 แบบ คือ ห้ามใช้งานอินเทอร์เน็ต ใช้งานได้เฉพาะเว็บไซต์ และใช้งานได้ทุกอย่างโดยสามารถเลือกรูปแบบการใช้งานลงไปในเวลาที่ต้องการได้โดยเลือกที่ Add Scheduler

Running Process



Running Process แสดงข้อมูลของโปรแกรมที่มีการเชื่อมต่อสู่อินเทอร์เน็ตโดยผู้ใช้งานสามารถคลิกขวาแล้วเลือกที่จะ Block การเชื่อมต่อของ โปรแกรมหรือสามารถที่จะปิดเลยก็ได้



Update สามารถเข้าได้ที่ Menu File->Update เมื่อเข้ามาจะแสดงรายละเอียดเวอร์ชันของ List ที่ใช้อยู่ เวลา update ครั้งล่าสุดและสามารถทำการ update ได้โดยการกด ที่ปุ่ม Check update

หากผู้ใช้งานกด Minimize หรือ Close โปรแกรมไปโปรแกรมจะถูกยุบลงมาอยู่ที่ Task List โดยหากผู้ต้องการใช้งานต้องทำการใช้ Password เพื่อทำงานเรียกโปรแกรมกลับมาใช้งานอีกครั้ง

