

โปรแกรมป้องกันผู้บุกรุกคอมพิวเตอร์ใช้งานคอมพิวเตอร์ส่วนบุคคล
INTRUSION PREVENTION SOFTWARE FOR PERSONAL COMPUTER



วิทยานิพนธ์นี้เขียนขึ้นโดยเป็นส่วนหนึ่งของโครงการศึกษาวิจัยของสถาบันวิจัยและพัฒนาบูรพาภิวัตน์ มหาวิทยาลัยราชภัฏบุรีรัมย์

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

โปรแกรมป้องกันผู้ละเมิดการใช้งานเครื่องคอมพิวเตอร์
INTRUSION PREVENTION SOFTWARE FOR PERSONAL COMPUTER



ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2557

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมป้องกันผู้ละเมิดการใช้งานเครื่องคอมพิวเตอร์

INTRUSION PREVENTION SOFTWARE FOR PERSONAL COMPUTER

ผู้จัดทำ

1. นายธนดล ชวโนวานิช รหัสนักศึกษา 53010618



..... อาจารย์ที่ปรึกษา
(อาจารย์ อัครเดช วัชรภพพงษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INTRUSION PREVENTION SOFTWARE FOR PERSONAL COMPUTER

Mr. Thanadon Chavanovanich 53010618

Mr. Akkradach Watcharapupong Advisor

Academic Year 2014

ABSTRACT

In the present, the security of personal computer has become extremely important because many users always store critical data in their personal computers that is the target for a hacker. For example, the hacker uses your personal computer for stealing someone else's data which take you become the hacker unconsciously. Although an operating system has a secure accessible system but it is inadequate. As the hacker has a lot of procedures for avoiding or the user does not set the login password that is easy for hacker to attack your computer.

So I have an inspiration to developing intrusion prevention software for personal computer which I take the advantage of web camera in personal computer to detect a hacker who wants to violate personal computer.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้จะไม่สำเร็จล่วงไปได้ หากมิได้รับความกรุณาจากอาจารย์ อัครเดช วัชรระภูพงษ์ อาจารย์ที่ปรึกษาโครงการ ซึ่งเป็นผู้ให้คำแนะนำแนวคิด ตลอดจนแก้ไขข้อบกพร่องต่างๆมาโดยตลอด จนกระทั่งโครงการฉบับนี้เสร็จสมบูรณ์ ผู้ศึกษาขอกราบขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคุณพ่อ คุณแม่และครอบครัวที่ให้คำปรึกษา และคอยสนับสนุนในเรื่องต่างๆ รวมทั้งเป็นกำลังใจที่ดีมาโดยตลอด

ขอบคุณนางสาว กนกพร ตันประเสริฐ ที่ช่วยให้คำแนะนำและช่วยสละเวลามาให้คำปรึกษาในการทำโครงการ รวมถึงการช่วยออกแบบโครงการในบางส่วน

ขอบคุณเพื่อนๆ ที่ช่วยให้คำแนะนำดีๆ ทางด้านความคิดสร้างสรรค์ และคอยเป็นกำลังใจในการทำงานเสมอมา

และสุดท้ายนี้ ขอขอบพระคุณทุกท่านที่มีส่วนช่วยให้โครงการนี้สำเร็จล่วงไปได้ด้วยดี ซึ่งอาจจะมีได้กล่าวถึงมา ณ ที่นี้ด้วย

นาย ธนดล

ชวโนวานิช

สารบัญ

	หน้า
โปรแกรมป้องกันผู้ละเมิดการใช้งานเครื่องคอมพิวเตอร์.....	I
INTRUSION PREVENTION SOFTWARE FOR	II
PERSONAL COMPUTER.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	2
1.4 วิธีการดำเนินการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ส่วนประกอบของรายงาน.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	4
2.1 รู้จักกับภาษา Visual Basic และการใช้งาน Visual Studio 2013 เบื้องต้น.....	4
2.1.1 Visual Basic คืออะไร.....	4
2.1.2 ข้อดีของภาษา Visual Basic.....	4
2.1.3 รุ่นของภาษา Visual Basic	5
2.1.4 รู้จักกับ Visual Basic.NET.....	5
2.1.5 รู้จักกับ Microsoft .NET	6
2.1.6 สถาปัตยกรรม .NET Framework	6
2.2 พื้นฐานการเขียนโปรแกรมด้วยภาษา Visual Basic 2010	7
2.2.1 การกำหนดชื่อตัวแปร (Variable)	7
2.2.2 คำสงวน (Reserved Words)	8

สารบัญ (ต่อ)

	หน้า
2.2.3 ชนิดของข้อมูลใน Visual Basic (Variable Type).....	10
2.2.4 การประกาศตัวแปร และการกำหนดค่าข้อมูลให้กับตัวแปร.....	11
2.2.5 ค่าคงที่ (Constant)	12
2.2.6 โอเปอเรเตอร์ (Operator).....	13
2.2.7 การเขียนคำอธิบายโปรแกรม (Comment).....	16
2.2.8 คำสั่ง (Statement)	16
2.2.9 การรับข้อมูลและแสดงข้อความ	17
2.2.10 การแสดงข้อความ	18
2.2.11 การขึ้นบรรทัดใหม่	20
2.2.12 โครงสร้างคำสั่งควบคุมการทำงานของโปรแกรม	21
2.3 การใช้งานโปรแกรม Visual Studio 2013.....	24
2.3.1 ส่วนประกอบที่สำคัญของหน้าต่างโปรแกรม.....	24
2.3.2 การสร้าง Project (Create New Project)	27
2.3.3 รายละเอียดของหน้าต่าง New Project มีดังนี้	28
2.3.4 การทำงานกับหน้าต่างโปรแกรม	29
2.3.5 การทดสอบการทำงาน (RUN).....	31
2.3.6 การหยุดการรัน (Stop Running)	31
บทที่ 3 การออกแบบและพัฒนา	32
3.1 รายละเอียดของโปรแกรมที่พัฒนา.....	32
3.1.1 อินพุทของโปรแกรม.....	32
3.1.2 เอาท์พุทของแบบจำลอง.....	32
3.1.3 ฟังก์ชันการทำงาน.....	32
3.1.4 ข้อจำกัดของโปรแกรมที่พัฒนา.....	33
3.1.5 ภาษาและเครื่องมือที่ใช้ในการพัฒนา	33
3.2 ขั้นตอนการทำงานของโปรแกรม	34

สารบัญ (ต่อ)

	หน้า
3.3 แผนภาพ UML (Unified Modeling Language Diagram)	35
3.3.1 Use Case Diagram	35
3.3.2 Class Diagram	35
3.4 การออกแบบส่วนติดต่อกับผู้ใช้งาน.....	37
บทที่ 4 การทดลองและผลการทำงาน.....	41
4.1 การทดลองใช้โปรแกรมเบื้องต้น.....	41
4.1.1 การใช้งานปุ่ม Lock	42
4.1.2 การใช้งานปุ่ม Options.....	44
4.1.3 การใช้งานปุ่ม Exit	50
บทที่ 5 บทสรุปและข้อเสนอแนะ	51
5.1 บทสรุปของโครงการ.....	51
5.2 ปัญหา อุปสรรค และแนวทางการแก้ไข.....	51
5.3 แนวทางในการพัฒนาต่อ	51
บรรณานุกรม.....	52

สารบัญตาราง

ตารางที่	หน้า
2.1 คำสงวน (Reserved Word).....	8
2.2 ชนิดของข้อมูลใน Visual Basic.....	10
2.3 โอเปอเรเตอร์ทางคณิตศาสตร์.....	13
2.4 โอเปอเรเตอร์สำหรับการเปรียบเทียบ.....	13
2.5 โอเปอเรเตอร์สำหรับการกำหนดค่า.....	14
2.6 โอเปอเรเตอร์ทางตรรกะ.....	15
2.7 ปุ่ม MessageBoxButtons.....	19
2.8 ปุ่ม MessageBoxIcon.....	19
2.9 ปุ่ม MessageBoxDefaultButton.....	20



สารบัญรูป

รูปที่	หน้า
2.1 แสดงสถาปัตยกรรมของโปรแกรม Microsoft Visual Studio 2008	6
2.2 เปรียบเทียบการทำงานของ .NET กับ JAVA	7
2.3 ตัวอย่างการสร้าง Form 1.....	18
2.4 ตัวอย่างการสร้าง Form 2.....	18
2.5 หน้าต่างโปรแกรม Microsoft Visual Studio 2013.....	24
2.6 แสดงแถบหัวเรื่อง title bar	24
2.7 แสดงแถบเมนู menu bar.....	25
2.8 แสดงหน้าต่างเริ่ม Start Page.....	25
2.9 แสดงแถบเครื่องมือ.....	26
2.10 แสดงหน้าต่าง Solution Explorer.....	26
2.11 แสดงแถบสถานะ Status Bar	27
2.12 แสดงการสร้าง Project ใหม่จากหน้าต่างเริ่มต้น (Start Page).....	27
2.13 แสดงการสร้าง Project ใหม่จากเมนู File.....	28
2.14 แสดงหน้าต่างการสร้าง Project ใหม่.....	28
2.15 แสดงหน้าต่างหลังจากสร้าง Project ใหม่.....	29
2.16 แสดงหน้าต่าง Solution	30
2.17 แสดงกล่องเครื่องมือ Tool Box.....	30
2.18 แสดงการหยุดการรันโดยกดที่ปุ่ม Stop.....	31
3.1 ขั้นตอนการทำงานของโปรแกรม	34
3.2 แผนภาพ Use Case สำหรับการใช้งานโปรแกรม.....	35
3.3 แผนภาพ Class Diagram ของโปรแกรม.....	36
3.4 หน้าแรกของโปรแกรม.....	37
3.5 หน้าหลักของโปรแกรม	37
3.6 หน้าหลักของโปรแกรมเมื่อนำเมาส์ไปวางที่ปุ่ม Lock.....	38
3.7 หน้าหลักของโปรแกรมเมื่อนำเมาส์ไปวางที่ปุ่ม Options.....	38
3.8 หน้าหลักของโปรแกรมเมื่อนำเมาส์ไปวางที่ปุ่ม Exit.....	39

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.9 หน้า Options ของโปรแกรม (TabPage1).....	39
3.10 หน้า Options ของโปรแกรม (TabPage2)	40
3.11 หน้า Lock Screen ของโปรแกรม.....	40
4.1 หน้าต่างแรกของโปรแกรม.....	41
4.2 หน้าต่างหลักของโปรแกรม	42
4.3 หน้า Lock Screen ของโปรแกรม.....	42
4.4 หน้า Lock Screen แสดงช่องใส่รหัสผ่านที่แท้จริง.....	43
4.5 Email แจ้งเตือนว่ามีผู้บุกรุก.....	43
4.6 หน้าต่าง Options ของโปรแกรม 1.....	44
4.7 หน้าต่าง Options ของโปรแกรม 2.....	44
4.8 Enable WebCam.....	45
4.9 ไม่ได้ติดตั้ง WebCam.....	46
4.10 Lock computer if system is idle.....	46
4.11 Send Notification to your Email.....	46
4.12 หน้าต่างใส่รหัสผ่าน Email.....	47
4.13 Confirm your Email.....	47
4.14 หน้าต่างแสดงเมื่อใส่ Email แล้ว.....	47
4.15 Global Hotkey.....	48
4.16 แสดง Key จากปุ่ม View	49
4.17 Use Schedule.....	49
4.18 Program still running.....	50

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

ปัจจุบันนี้ การรักษาความลับของข้อมูลเป็นเรื่องสำคัญมาก เพราะในปัจจุบันผู้คนหันมาใช้อุปกรณ์ทางด้านไอทีในการเก็บข้อมูลต่างๆมากขึ้น โดยข้อมูลที่เก็บลงในอุปกรณ์ไอทีนี้เรียกว่า ข้อมูลดิจิทัล ทำให้การโจรกรรมข้อมูลดิจิทัลในปัจจุบันนั้นแพร่หลายอย่างมาก ทำให้ผู้ใช้งานคอมพิวเตอร์ในปัจจุบันต้องหาวิธีป้องกันป้องกันด้วยวิธีการต่างๆเพื่อไม่ให้ผู้ประสงค์ร้ายเข้ามาโจรกรรมข้อมูลได้

การป้องกันการเข้าใช้งานเครื่องคอมพิวเตอร์นั้นมีหลากหลายวิธีด้วยกัน ยกตัวอย่างเช่น การป้องกันของระบบปฏิบัติการโดยการใส่รหัสผ่าน หรือลงโปรแกรมประยุกต์เพื่อช่วยป้องกัน ซึ่งวิธีดังกล่าวสามารถลดการละเมิดการเข้าใช้งานเครื่องคอมพิวเตอร์ได้ แต่ก็ยังมีจุดบกพร่องอยู่ คือสามารถป้องกันการเข้าใช้งานเครื่องคอมพิวเตอร์ได้เพียงอย่างเดียว แต่ไม่สามารถระบุตัวคนร้ายได้ ซึ่งอาจส่งผลให้เกิดเหตุการณ์ละเมิดเช่นนี้อีกในอนาคต อีกทั้งการป้องกันของระบบปฏิบัติการหรือโปรแกรมประยุกต์ยังมีช่องโหว่ให้ผู้ประสงค์ร้ายสามารถลักลอบเข้าใช้งานได้

ด้วยสาเหตุต่างๆดังที่กล่าวมานั้น เป็นเหตุผลที่เกิดเป็นที่มาของโครงการนี้ เนื่องจากข้าพเจ้ามีความต้องการที่จะแก้ไขจุดบกพร่องดังกล่าว จึงได้ทำการพัฒนาโปรแกรมประยุกต์ เพื่อนำมาใช้ในการป้องกันการละเมิดใช้งานเครื่องคอมพิวเตอร์โดยการใส่รหัสผ่าน โดยจุดเด่นของโปรแกรมนี้อคือการใช้ประโยชน์จากกล้อง (Webcam) ของคอมพิวเตอร์ เพื่อใช้ในการถ่ายภาพผู้ประสงค์ร้ายในขณะที่ละเมิดการใช้งาน

1.2 วัตถุประสงค์ของโครงการ

- 1) เพื่อระบุตัวตนของผู้ประสงค์ร้าย ในขณะที่กำลังทำการละเมิดการใช้งานเครื่องคอมพิวเตอร์
- 2) เพื่อป้องกันผู้ประสงค์ร้ายที่ต้องการละเมิดการใช้งานเครื่องคอมพิวเตอร์ในอีกระดับหนึ่ง

1.3 ขอบเขตของโครงการ

- 1) ออกแบบโปรแกรมเพื่อใช้ป้องกันเครื่องคอมพิวเตอร์ด้วยรหัสผ่าน
- 2) โปรแกรมสามารถเรียกใช้งานกล้อง (Webcam) ของเครื่องคอมพิวเตอร์ได้
- 3) โปรแกรมสามารถส่งอีเมล (E-mail) ถึงเจ้าของเครื่องคอมพิวเตอร์ได้
- 4) โปรแกรมสามารถส่งรูปภาพที่ถ่ายจากกล้อง (Webcam) ผ่านทางอีเมล (E-mail) ได้
- 5) โปรแกรมสามารถบันทึกรูปภาพลงในคอมพิวเตอร์ได้

1.4 วิธีการดำเนินการ

- 1) ศึกษาการเขียนโปรแกรมด้วยภาษา Visual Basic
- 2) ศึกษาการเรียกใช้กล้อง (Webcam) ของเครื่องคอมพิวเตอร์
- 3) ศึกษาการป้องกันการป้อนอินพุตขณะทำการล็อคเครื่องคอมพิวเตอร์
- 4) ออกแบบโปรแกรมป้องกันตามที่ได้ศึกษาไว้
- 5) พัฒนาโปรแกรมตามที่ได้ออกแบบไว้เพื่อให้สามารถใช้งานได้จริง
- 6) ทดลองโปรแกรมป้องกัน ได้แก่ การใช้งาน การเรียกใช้กล้อง (Webcam) และการส่งอีเมล (E-mail)
- 7) วิเคราะห์ผล และสรุปผลการทดลอง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้รับความรู้ ความเข้าใจเกี่ยวกับการเขียนโปรแกรมด้วยภาษา Visual Basic
- 2) ได้เรียนรู้การทำงานของระบบปฏิบัติการ ในเรื่องการป้อนอินพุตเบื้องต้น
- 3) นำโปรแกรมที่พัฒนาไปใช้งานได้จริง และเกิดประโยชน์สูงสุด

1.6 ส่วนประกอบของรายงาน

รายงานนี้ได้แบ่งเนื้อหาออกเป็น 5 บท คือ

บทที่ 1 บทนำ กล่าวถึงความสำคัญและที่มาของโครงการ วัตถุประสงค์ของโครงการ ขอบเขตของโครงการ วิธีการดำเนินการ ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของรายงาน

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในโครงการ ประกอบด้วย พื้นฐานภาษา Visual Basic และการใช้งานโปรแกรม Visual Studio 2013

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การออกแบบ กล่าวถึงขั้นตอนการออกแบบ ขั้นตอนการทำงาน โครงสร้างของระบบ Use Case Diagram ของระบบ Class Diagram และ User Interface ของระบบ

บทที่ 4 การทดลองและผลการทำงาน จะเป็นการแนะนำการใช้งานของโปรแกรม รวมถึงฟังก์ชันต่างๆเบื้องต้น

บทที่ 5 สรุปผลการทดลอง กล่าวถึงการสรุปผลการทดลอง อุปสรรคและการแก้ปัญหาของการทดลอง แนวทางการพัฒนาของโครงการต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 รู้จักกับภาษา Visual Basic และการใช้งาน Visual Studio 2013 เบื้องต้น

ในบทนี้จะเป็นการแนะนำให้รู้จักกับภาษา Visual Basic ว่าคืออะไรและมีความสำคัญอย่างไร อีกทั้งยังเปรียบเทียบกับภาษาอื่นๆ ว่ามีข้อแตกต่างจากภาษาอื่นๆ อย่างไร

2.1.1 Visual Basic คืออะไร

ในหลายปีที่ผ่านมาวงการคอมพิวเตอร์ได้มีการพัฒนาในด้านต่างๆ เป็นอย่างมากโดยเฉพาะทางด้านซอฟต์แวร์ ภาษา Visual Basic ถือเป็นโปรแกรมภาษาคอมพิวเตอร์ตัวหนึ่ง มีความสามารถในการทำงานที่คล้ายกับภาษาคอมพิวเตอร์อื่นๆ เช่น Java , PasCal , C++ , C# แต่ออกแบบมาให้ใช้งานได้ง่ายยิ่งขึ้น

Visual Basic หรือ VB เป็นภาษาโปรแกรมแบบ GUI สร้างโดยบริษัทไมโครซอฟท์ ภาษานี้เป็นหนึ่งในภาษาโปรแกรมยอดนิยมสำหรับโปรแกรมที่ใช้ในด้านธุรกิจ ภาษานี้พัฒนามาจากภาษาเบสิก และยังได้พัฒนาต่อเป็นภาษา VB.NET อีกด้วย Visual Basic สนับสนุน Rapid Application Development (RAD) ทั้งด้านการพัฒนาโปรแกรมประยุกต์แบบ graphical user interface (GUI) , การเข้าถึงฐานข้อมูลโดยใช้การเชื่อมต่อแบบ DAO, RDO, หรือ ADO, และการสร้าง ActiveXcontrol จุดเด่นอีกอย่างหนึ่งของ Visual Basic คือนักเขียนโปรแกรมสามารถนำโปรแกรมประยุกต์หลายๆ โปรแกรมมารวมกันในโปรแกรมเดียว และยังสามารถประยุกต์ใช้คอมโพเนนต์ของ Visual Basic ที่มีเตรียมไว้ให้แล้วได้อีกด้วย

2.1.2 ข้อดีของภาษา Visual Basic

- 1) Simple คืองานต่อการเขียนโปรแกรม เนื่องจากไม่มีตัวแปรพอยน์เตอร์ และมีกลไกในการจัดการกับหน่วยความจำโดยอัตโนมัติ
- 2) Platform independent สามารถนำไปทำงานบนระบบปฏิบัติการอื่นได้ โดยคุณสมบัตินี้เรียกว่า “Write once run anywhere”
- 3) Object Oriented Programming เป็นการเขียนโปรแกรมเชิงวัตถุ
- 4) Dynamic สามารถปรับเปลี่ยนเพิ่มเติมไลบรารีต่างๆได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 รุ่นของภาษา Visual Basic

ภาษา Visual Basic นั้นได้มีการพัฒนามาตั้งแต่รุ่นแรกๆ ที่ทำงานบนระบบปฏิบัติการ DOS มาจนถึงปัจจุบันที่ทำงานอยู่บนระบบปฏิบัติการวินโดวส์ก่อนจะมาเป็น ภาษา Visual Basic.NET ให้เราใช้กันได้นั้น เราอาจจะคิดว่า ภาษา BASIC เป็นของ Microsoft คิดค้นขึ้น

แท้จริงแล้ว ภาษา BASIC เป็นคำที่เกิดจากอักษรย่อของคำว่า Beginner's All-purpose Symbolic Instruction Code ถูกพัฒนาขึ้นตั้งแต่ในช่วงต้นปี ค.ศ. 1963 ที่วิทยาลัย Dartmouth College ในสหรัฐอเมริกา โดย จอห์น เคเมเนย์ John G. Kemeny และ ธอมัส เคิร์ตส์ Thomas E. Kurtz ออกแบบมาให้เป็นภาษาคอมพิวเตอร์ที่ใช้งานได้ง่ายในการเขียนโปรแกรม โดยสมัยก่อนมีการใช้งานบนเครื่องไมโครคอมพิวเตอร์ โดยมีการนำมาทำเป็น ชุดคำสั่งถาวร หรือ Firmware เพื่อเก็บไว้ใน ROM บนไมโครคอมพิวเตอร์รุ่นแรกๆ รวมถึงการเขียนโปรแกรมทั่วไปด้วย ด้วยความง่ายจึงเป็นที่แพร่หลายและได้รับความนิยม และต่อมาก็ได้เกิดรุ่นต่างๆ ของ BASIC มาอีก เช่น

- 1) ในปี 1975 BASIC for Altair by Bill Gates
- 2) ในปี 1980 GWBasic by Microsoft
- 3) ในปี 1980 QuickBasic by Microsoft
- 4) ในปี 1991 Visual Basic by Microsoft

และยังมี Turbo BASIC อีกจากนั้นได้ผ่านการพัฒนาต่อเนื่องมาอีกหลายรุ่นนับตั้งแต่ Visual Basic รุ่นแรก จนมาเป็น Visual Basic 6 ในปี 1998 และเมื่อมีการพัฒนา .NET Framework ขึ้น การเปลี่ยนแปลงทางด้านโครงสร้างของภาษา BASIC ครั้งใหญ่จนกลายมาเป็น Visual Basic.NET ที่มีใช้กันอยู่ในปัจจุบัน

2.1.4 รู้จักกับ Visual Basic.NET

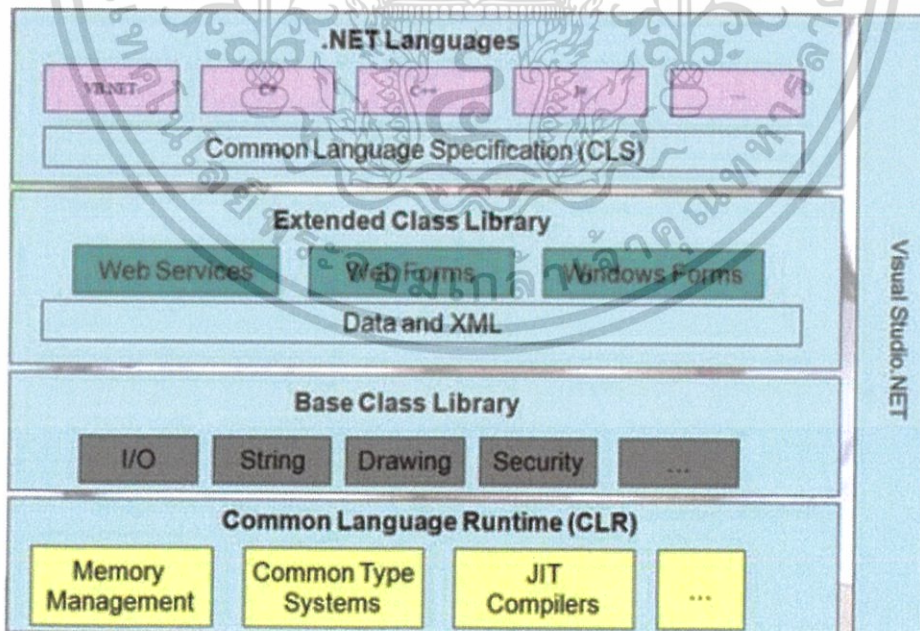
Visual Basic.NET เป็นภาษาที่พัฒนาต่อจาก Visual Basic 6.0 หรือพูดง่ายๆ ก็คือเป็น Visual Basic Version 7 ซึ่งขยายขีดความสามารถที่ Visual Basic เดิมไม่สามารถทำได้ โดยเฉพาะในเรื่องของการเป็นภาษาเชิงวัตถุอย่างแท้จริง (สนับสนุนโครงสร้างของภาษาที่เป็น OOP 100%) ทำให้โครงสร้างภาษาของ Visual Basic.NET นั้นมีความสมบูรณ์มากขึ้น แต่ก็ยังคงสนับสนุนรูปแบบการเขียนแบบเดิมไว้ในบางส่วนเพื่อความสะดวกสำหรับผู้ที่ย้ายจาก Visual Basic Version ก่อนหน้านี้มาเป็น Visual Basic.NET โดย Visual Basic.NET เป็นภาษาหนึ่งที่อยู่ในชุดเครื่องมือ Microsoft Studio.NET โดยจะใช้ IDE (Integrated Development Environment) ร่วมกับภาษาอื่นอีก 3 ภาษา ที่อยู่ในชุดเครื่องมือนี้ ซึ่งได้แก่ Visual C#, Visual C++ และ Visual J#

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 รู้จักกับ Microsoft .NET

- 1) Microsoft .NET หรือเรียกสั้น ๆ ว่า .NET เป็นเทคโนโลยีของบริษัทไมโครซอฟต์
- 2) .NET คือแพลตฟอร์มที่ใช้สำหรับพัฒนาซอฟต์แวร์สำหรับระบบปฏิบัติการ Windows
- 3) นำเสนอหลักการที่ว่า สามารถพัฒนาซอฟต์แวร์ด้วยภาษาอะไรก็ได้ที่เราถนัดและสามารถใช้งานร่วมกับโปรแกรมภาษาอื่น ๆ ได้อย่างกลมกลืน
- 4) .NET มีภาษาใหม่ๆ เกิดขึ้นภายใต้กฎเกณฑ์มาตรฐานเดียวกัน เช่น VB.NET, C#, C++.NET, J#.NET หรือแม้แต่ COBAL.NET
- 5) ทุกภาษาที่สนับสนุน .NET จะอยู่ภายใต้มาตรฐานเดียวกันที่เรียกว่า Common Language Specifications (CLS) และโครงสร้างพื้นฐานตั้งแต่ชนิดข้อมูล, ชุดคำสั่งพื้นฐานเช่นการจัดการ I/O, ฐานข้อมูล ที่อยู่ภายใต้ CLS ทำให้สามารถพัฒนาซอฟต์แวร์โดยใช้ภาษาโปรแกรมได้หลายภาษา
- 6) โปรแกรมที่เราเขียนขึ้นมา เมื่อคอมไพล์ (compile) แล้วจะอยู่ในรูป intermediate language ที่เรียกว่า MSIL (Microsoft Intermediate Language)

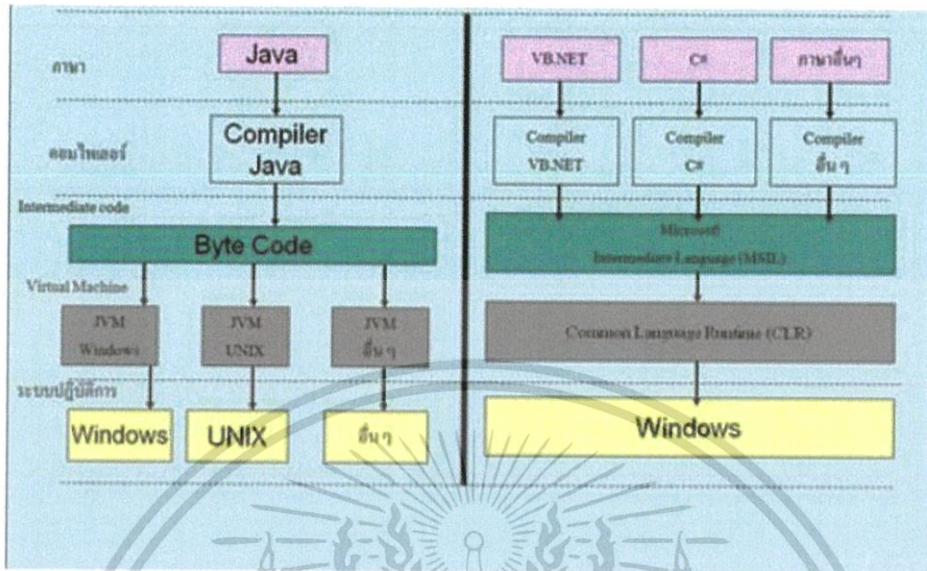
2.1.6 สถาปัตยกรรม .NET Framework



รูปที่ 2.1 แสดงสถาปัตยกรรมของโปรแกรม Microsoft Visual Studio 2008

เปรียบเทียบการทำงานของ .NET กับ JAVA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 เปรียบเทียบการทำงานของ .NET กับ JAVA

2.2 พื้นฐานการเขียนโปรแกรมด้วยภาษา Visual Basic 2010

ภาษาคอมพิวเตอร์แต่ละภาษาจะมีรูปแบบหรือโครงสร้างทางภาษาเป็นของตนเอง ซึ่งอาจจะแตกต่างกันบ้างในแต่ละภาษา โครงสร้างทางภาษาจะเป็นตัวควบคุมให้โปรแกรมทำงานตามข้อกำหนดที่เราต้องการ ในการพัฒนาแอปพลิเคชันโดยใช้ภาษาแต่ละภาษา ผู้เขียนโปรแกรมจะต้องทำความเข้าใจโครงสร้างของภาษาที่จะใช้ก่อน เพื่อให้การเขียนโปรแกรมไม่มีข้อผิดพลาด การศึกษาภาษา Visual Basic จะต้องทำความเข้าใจเกี่ยวกับรูปแบบและข้อกำหนดพื้นฐานต่างๆ ของภาษา ไม่ว่าจะเป็นตัวแปร โอเปอเรเตอร์ เงื่อนไขและรูปแบบต่างๆ เป็นต้น

2.2.1 การกำหนดชื่อตัวแปร (Variable)

ตัวแปร หมายถึง ชื่อที่ใช้แทนข้อมูลชนิดต่างๆ โดยข้อมูลที่จะนำมาเก็บในตัวแปรจะต้องตรงกับชนิดของข้อมูลที่กำหนดไว้ การนำข้อมูลไปใช้งานจะกระทำผ่านตัวแปรที่กำหนดนี้

กฎการตั้งชื่อตัวแปรที่สำคัญ

- 1) ชื่อของตัวแปรต้องเริ่มต้นด้วยตัวอักษร a-z เท่านั้นจะเป็นตัวพิมพ์ใหญ่หรือพิมพ์เล็กก็ได้ ส่วนตัวถัดไปจะเป็นตัวอักษรหรือตัวเลขก็ได้
- 2) ห้ามใช้อักษรพิเศษอื่น ๆ ที่ไม่ใช่ตัวอักษรหรือตัวเลข เช่น ?, +, -, *, /, ... ยกเว้นเครื่องหมาย _ (underscore) เท่านั้น
- 3) ต้องไม่ซ้ำกับ Reserved Words ใน Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 คำสงวน (Reserved Words)

คำสงวน หรือ “Reserved Words” เป็นคำที่ Visual Basic สงวนไว้ใช้เป็นคำสั่ง ไม่สามารถนำคำเหล่านี้ไปเป็นชื่ออย่างอื่นได้ คำสงวนใน Visual Basic มีดังนี้

ตารางที่ 2.1 คำสงวน (Reserved Word)

AddHandler	AddressOf	Alias	And
AndAlso	As	Boolean	ByRef
Byte	ByVal	Call	Case
Catch	CBool	CByte	CChar
CDate	CDbl	CDec	Char
CInt	Class	Class	CLng
	Constraint	Statement	
CObj	Const	Continue	CSByte
CShort	CSng	CStr	CType
CUInt	CULng	CUShort	Date
Decimal	Declare	Default	Delegate
Dim	DirectCast	Do	Double
Each	Else	Elseif	End Statement
End<keyword>	EndIf	Enum	Erase
Error	Event	Exit	False
Finally	For(inFor...Next)	For	Friend
		Each...Next	
Function	Get	GetType	GetXMLNamespace
Global	GoSub	GoTo	Handles
If	If()	Implements	Implements Statement
Imports (.NET Namespace and Type)	Imports (XML Namespace)	In	In (Generic Modifier)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Inherits	Integer	Interface	Is
IsNot	Let	Lib	Like
Long	Loop	Me	Mod
Module	Module Statement	MustInherit	MustOverride
MyBase	MyClass	Namespace	Narrowing
New Constraint	New Operator	Next	Next (in Resume)
Not	Nothing	NotInheritable	NotOverridable
Object	Of	On	Operator
Option	Optional	Or	OrElse
Out (Generic Modifier)	Overloads	Overridable	Overrides
ParamArray	Partial	Private	Property
Protected	Public	RaiseEvent	ReadOnly
ReDim	REM	Remove Handler	Resume
Return	SByte	Select	Set
Shadows	Shared	Short	Single
Static	Step	Stop	String
Structure Constraint	Structure Statement	Sub	SyncLock
Then	Throw	To	True
Try	TryCast	TypeOf...Is	UInteger
ULong	UShort	Using	Variant
Wend	When	While	Widening
With	WithEvents	WriteOnly	Xor
#Const	#Else	#Elseif	#End
#If	=	&	&=

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

*	*=	/	/=
\	\=	^	^=
+	+=	-	-=
>> Operator (Visual Basic)	>>= Operator (Visual Basic)	<<	<<=

2.2.3 ชนิดของข้อมูลใน Visual Basic (Variable Type)

ในการประกาศตัวแปรสิ่งที่จะต้องกระทำคู่กันก็คือการระบุชนิดของข้อมูลที่จะเก็บในตัวแปรนั้นด้วย ชนิดของข้อมูลใน Visual Basic มีดังนี้

ตารางที่ 2.2 ชนิดของข้อมูลใน Visual Basic

ชนิดของข้อมูล	ขอบเขต	จำนวนไบต์
Boolean	True หรือ False	2
Byte	0 ถึง 255 (unsigned)	1
Char	0 ถึง 65535 (unsigned)	2
Decimal	แบบไม่มีจุดทศนิยม : 0 ถึง $\pm 79,228,162,514,264,337,593,543,950,335$ แบบมีจุดทศนิยม : 0 ถึง $\pm 7.9228162514264337593543950335$	16
Double	ช่วงลบ : $-1.79769313486231570E+308$ ถึง $-4.94065645841246544E-324$ ช่วงบวก : $4.94065645841246544E-324$ ถึง $1.79769313486231570E+308$	8
Integer	-2,147,483,648 ถึง 2,147,483,647	4
Long (Long Integer)	-9,223,372,036,854,775,808 ถึง 9,223,372,036,854,775,807	8
Object	เก็บข้อมูลชนิดใดก็ได้	4
Short	-32,768 ถึง 32,767	2
Single	ช่วงลบ : $-3.4028235E+38$ ถึง $-1.401298E-45$	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ภายนอก
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	ช่วงบวก : 1.401298E-45 ถึง 3.4028235E+38	
String (Variable Length)	ขึ้นกับ Platform	0 ถึง ประมาณ 2 billion (2^{31})

2.2.4 การประกาศตัวแปร และการกำหนดค่าข้อมูลให้กับตัวแปร

รูปแบบการประกาศตัวแปรในภาษา Visual Basic จะใช้คำสั่ง Dim ซึ่งมีรูปแบบ ดังนี้

Dim	ชื่อตัวแปร	As	ชนิดของข้อมูล
-----	------------	----	---------------

2.2.4.1 การกำหนดค่าข้อมูลให้กับตัวแปร

ในการกำหนดค่า (Assignment) ให้กับตัวแปร สำหรับชนิดของข้อมูลแต่ละชนิดจะมีรูปแบบแตกต่างกัน โดยใช้เครื่องหมาย “=” ในการให้ค่ากับตัวแปร

1) ข้อมูลชนิดตัวเลข

ข้อมูลชนิดตัวเลข เช่น Byte, Integer, Double จะสามารถใส่ตัวเลขลงไปได้เลย (ไม่ต้องมีเครื่องหมาย ,) เช่น

โปรแกรมที่ 2.1 ข้อมูลชนิดตัวเลข

```
Dim x As Integer = 4851
Dim y As Double = 1422.576
```

2) ข้อมูลชนิดบูลีน

ข้อมูลชนิดบูลีน (Boolean) จะมีค่าเป็นไปได้เพียง 2 ค่า คือ True หรือ False อย่างใดอย่างหนึ่ง สามารถนำไปกำหนดค่าให้กับตัวแปรได้ เช่น

โปรแกรมที่ 2.2 ข้อมูลชนิดบูลีน

```
Dim a As Boolean = True
Dim b As Boolean = False
```

3) ข้อมูลชนิดสตริง

ข้อมูลชนิดสตริง (string) คือ ข้อมูลที่เป็นสายอักขระหรือเป็นการนำเอาอักขระแต่ละตัวมาวางเรียงต่อกัน ซึ่งข้อมูลชนิดสตริงจะมีความยาวเท่าไรก็ได้ และไม่จำเป็นต้องเขียนอักขระทุกตัว เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นใบปะกันเอกสารนี้ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ติดกัน การกำหนดข้อมูลที่เป็นสตริง จะต้องกำหนดจุดเริ่มต้นและสิ้นสุดของสตริงเสมอ โดยใช้เครื่องหมาย Double quote (“ ”) เช่น

โปรแกรมที่ 2.3 ข้อมูลชนิดสตริง

```
Dim strName As String = "Mr. Thanadon Chavanovanich"
```

ในกรณีที่สตริงจำเป็นจะต้องมีเครื่องหมาย Double quote (“ ”) อยู่ด้วยให้เขียนเครื่องหมายนี้ 2 อันซ้อนกัน เช่น กำหนดตัวแปร str มีค่าเป็น I Love “VB” for .NET programming จะต้องเขียน ดังนี้

โปรแกรมที่ 2.4 ใส่ Double quote ซ้อนกัน

```
Dim str As String = "I Love ""VB"" for .NET Programming"
```

2.2.5 ค่าคงที่ (Constant)

ค่าคงที่ คือ ค่าที่กำหนดให้กับตัวแปรค่าใดค่าหนึ่ง เพื่อจะนำไปใช้งานตลอดทั้งโปรแกรม โดยไม่สามารถเปลี่ยนค่าตัวแปรนั้นได้ การกำหนดค่าคงที่มีรูปแบบ ดังนี้

Const	ชื่อค่าคงที่	=	ค่าที่กำหนด
-------	--------------	---	-------------

เช่น Const PI = 3.141

หรือ กำหนดชนิดของข้อมูลด้วย เช่น

Const	ชื่อค่าคงที่	As	ชนิดข้อมูล	=	ค่าที่กำหนด
-------	--------------	----	------------	---	-------------

เช่น Const VAT As Integer = 7

โดยปกติ การกำหนดชื่อค่าคงที่ นิยมใช้ภาษาอังกฤษตัวใหญ่ เพื่อให้แตกต่างจากตัวแปร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.6 โอเปอร์เรเตอร์ (Operator)

โอเปอร์เรเตอร์ คือ สัญลักษณ์ที่ใช้ในการประมวลผลทางคณิตศาสตร์หรือทางตรรกะ ภาษา Visual Basic แบ่ง Operator ได้ดังนี้

2.2.6.1 โอเปอร์เรเตอร์ทางคณิตศาสตร์

ใช้ในการคำนวณทางคณิตศาสตร์ เช่น

ตารางที่ 2.3 โอเปอร์เรเตอร์ทางคณิตศาสตร์

Operator	ความหมาย
+	บวก
-	ลบ
*	คูณ
Operator	ความหมาย
/	หาร
^	ยกกำลัง เช่น $2^3 = 8$
Mod	หารเอาเฉพาะเศษ เช่น $8 \text{ Mod } 3 = 2$
\	หารโดยเอาเฉพาะจำนวนเต็ม ตัดเศษทิ้ง เช่น $10 \setminus 3 = 3$

2.2.6.2 โอเปอร์เรเตอร์สำหรับการเปรียบเทียบ

เครื่องหมายสำหรับการเปรียบเทียบ เพื่อหาค่าความจริงระหว่าง 2 นิพจน์ ซึ่งหากผลการเปรียบเทียบเป็นจริง จะได้ผลลัพธ์เป็น True แต่หากผลการเปรียบเทียบเป็นเท็จจะได้ผลลัพธ์เป็น False

ตารางที่ 2.4 โอเปอร์เรเตอร์สำหรับการเปรียบเทียบ

Operator	ความหมาย
=	ใช้ในการเปรียบเทียบว่าเท่ากันหรือไม่ $y = 10, z = 10, c = (y = z)$
<>	ไม่เท่ากับ $y = 10, z = 10, c = (y <> z)$
>	มากกว่า $c = (10 > 5)$
<	น้อยกว่า $c = (10 < 5)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

\geq	มากกว่าหรือเท่ากับ $c = (10 \geq 5)$
\leq	น้อยกว่าหรือเท่ากับ $c = (10 \leq 5)$

2.2.6.3 โอเปอเรเตอร์สำหรับการกำหนดค่า (Assignment)

โอเปอเรเตอร์สำหรับการกำหนดค่า เป็นเครื่องหมายสำหรับการกำหนดค่าของตัวแปรทางด้านซ้ายของเครื่องหมายด้วยค่าที่อยู่ทางด้านขวาของเครื่องหมาย

ตารางที่ 2.5 โอเปอเรเตอร์สำหรับการกำหนดค่า

Operator	ความหมาย
=	เท่ากับ เช่น $X = 10$
Operator	ความหมาย
\wedge	ยกกำลังด้วย เช่น $2 \wedge 3$ หมายถึง 2 ยกกำลังด้วย 3 เท่ากับ 8
$*$	คูณด้วย เช่น $X * = 10$ หมายถึง $X = X * 10$
$/$	หารด้วย เช่น $X /= 10$ หมายถึง $X = X / 10$
\backslash	หารด้วย เช่น $X \backslash = 10$ หมายถึง $X = X \backslash 10$
$+$	บวกด้วย เช่น $X += 10$ หมายถึง $X = X + 10$
$-$	ลบด้วย เช่น $X -= 10$ หมายถึง $X = X - 10$
$\&$	ใช้สำหรับข้อมูลชนิดสตริงเพื่อนำข้อมูลชนิดสตริงมาต่อท้าย
	$x = \text{"Hello"}$ $y = \text{"World"}$ $z = x \& y$
	$x \&= y$ จะได้ $x = \text{"HelloWorld"}$

2.2.6.4 โอเปอเรเตอร์ทางตรรกะ (Logical)

โอเปอเรเตอร์ทางตรรกะเป็นการเปรียบเทียบระหว่าง 2 นิพจน์ ซึ่งแต่ละนิพจน์ที่นำมาเปรียบเทียบจะต้องมีค่าเป็น True หรือ False ใดๆอย่างหนึ่ง และผลลัพธ์ที่ได้ก็จะเป็น True หรือ False ใดๆอย่างหนึ่งด้วยเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 โอเปอร์เรเตอร์ทางตรรกะ

Operator	ความหมาย
And	ถ้าทั้งสองนิพจน์เป็นจริงทั้งคู่ ผลที่ได้จะเป็นจริง นอกนั้นเป็นเท็จหมด
Or	ถ้าทั้งสองนิพจน์เป็นเท็จทั้งคู่ ผลที่ได้จะเป็นเท็จ นอกนั้นเป็นจริงหมด
Not	ผลที่ได้จะเป็นค่าตรงกันข้ามกับค่าของนิพจน์เดิม
Xor	ถ้านิพจน์แรกกับนิพจน์หลังเหมือนกัน (จริง - จริง / เท็จ - เท็จ) ผลที่ได้จะเป็นเท็จ นอกนั้นเป็นจริง

2.2.6.5 โอเปอร์เรเตอร์สำหรับการเชื่อมต่อสตริง (String Concatenation)

การเชื่อมต่อข้อความหรือสตริงเข้าด้วยกัน จะใช้เครื่องหมายอยู่ 2 ตัว คือเครื่องหมาย “+” และเครื่องหมาย “&”

ถ้าเป็นการเชื่อมต่อสตริงกับสตริง เข้าด้วยกัน สามารถใช้เครื่องหมาย + หรือ เครื่องหมาย & ก็ได้ เช่น

“Hello” + “World”

หรือ “Hello” & “World”

แต่ถ้าเป็นการเชื่อมต่อระหว่างสตริงกับตัวเลข จะต้องใช้เครื่องหมาย & เช่น “Visual Basic” & 2010

ถ้าต้องการเครื่องหมาย + เพื่อเชื่อมต่อสตริงกับตัวเลขจะต้องแปลงตัวเลขให้อยู่ในรูปแบบสตริงก่อน ซึ่งมีวิธีการ ดังนี้

1) ใช้ฟังก์ชัน CStr()

เช่น “Visual Basic” + CStr(2010)

2) ใช้ฟังก์ชัน ToString()

เช่น “Visual Basic” + 2010.ToString()

2.2.7 การเขียนคำอธิบายโปรแกรม (Comment)

คำอธิบายโปรแกรม คือ ข้อความที่ผู้พัฒนาโปรแกรมต้องการที่จะบรรยายคำสั่ง หรือเป็นสิ่งที่เตือนความจำ เมื่อทำการแก้ไขเปลี่ยนแปลงโค้ดโปรแกรม ก็สามารถศึกษาได้จากคำอธิบายโปรแกรม ซึ่งคำอธิบายโปรแกรมหากล่าว โปรแกรมจะไม่นำมาประมวลผลด้วย

ในการเขียนคำอธิบายใน Visual Basic จะใช้เครื่องหมาย Quote คือ ‘ วางไว้หน้าคำอธิบายโปรแกรม ดังนั้นตั้งแต่เครื่องหมาย ‘ ไปจนจบบรรทัดโปรแกรมจะถือว่าเป็นคำอธิบายโปรแกรม (Comment) เช่น Dim x As Integer ‘x เป็นตัวแปรชนิดจำนวนเต็ม

ข้อสังเกต : คำอธิบายโปรแกรม (Comment) จะแสดงข้อความเป็นสีเขียว

2.2.8 คำสั่ง (Statement)

Statement (คำสั่ง) คือ คำสั่งที่สั่งให้คอมพิวเตอร์ทำงานโดย Visual Basic จะถือว่าเป็น 1 บรรทัด คือ 1 Statement หรือ เมื่อสิ้นสุดบรรทัด ก็ถือว่าเป็นการสิ้นสุด Statement เช่น

โปรแกรมที่ 2.5 แสดง Statement

```
Dim a As Integer
a += 20
```

ในกรณีที่ต้องการเขียน Statement ไว้ในบรรทัดเดียวกันหลาย ๆ Statement ให้ใช้เครื่องหมาย “ : ” คั่นระหว่าง Statement เช่น

โปรแกรมที่ 2.6 เขียน Statement ในบรรทัดเดียวกัน

```
Dim a, b, c As Integer
a = 10 : b = 20 : c = 30
```

ในกรณีที่ Statement มีความยาวมากจำเป็นจะต้องพิมพ์หลายบรรทัดให้ใช้เครื่องหมาย “ _ ” วางไว้ท้ายบรรทัดที่ต้องการให้ขึ้นบรรทัดใหม่ เช่น

โปรแกรมที่ 2.7 การใช้เครื่องหมาย _

```
Dim a As _
Integer = 10
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่เป็นสตริง จะต้องแยกสตริงออกจากกันก่อนแล้วใช้การเชื่อมตัวสตริงด้วย & หรือ + ตามด้วยเครื่องหมาย _ เช่น

โปรแกรมที่ 2.8 ใช้เครื่องหมาย _ กับการเชื่อม String แบบที่ 1

```
Dim conStr As String = _
"Provider = Microsoft.ACE.OLEDB.12.0" & _
"Data Source = C:\Database\stock.accdb"
```

หรือ

โปรแกรมที่ 2.9 ใช้เครื่องหมาย _ กับการเชื่อม String แบบที่ 2

```
Dim nameStr As String = _
"Thanadon Chavanovanich"
```

2.2.9 การรับข้อมูลและแสดงข้อความ

2.2.9.1 การรับข้อมูล

การรับข้อมูลเป็นการให้ผู้ใช้สามารถกำหนดข้อมูลลงไปเองตามต้องการ โดย Visual Basic จะใช้ฟังก์ชัน InputBox() ในการรับข้อมูลจากผู้ใช้ เพื่อนำไปประมวลผล
รูปแบบ

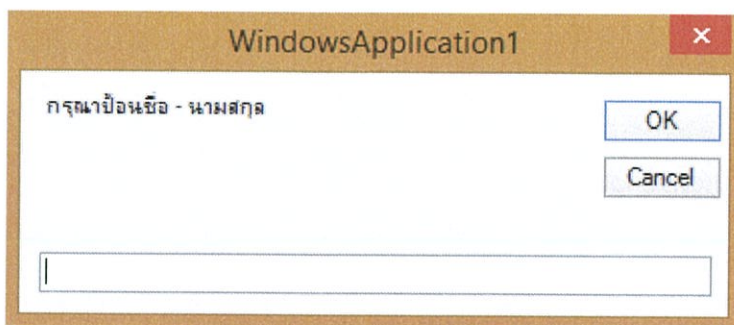
InputBox("ข้อความ")

เนื่องจากฟังก์ชันนี้ใช้รับข้อมูล ดังนั้นจะต้องสร้างตัวแปรไปรับค่าจาก InputBox() เช่น

โปรแกรมที่ 2.10 การสร้าง Form แบบที่ 1

```
Dim name As String
name = InputBox("กรุณาป้อนชื่อ - นามสกุล")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ตัวอย่างการสร้าง Form 1

โปรแกรมที่ 2.11 การสร้าง Form แบบที่ 2

```
name = InputBox("กรุณาป้อนชื่อ - นามสกุล", "ป้อนข้อมูล")
```



รูปที่ 2.4 ตัวอย่างการสร้าง Form 2

2.2.10 การแสดงข้อความ

การแสดงข้อความเป็นการแจ้งข้อมูลให้แก่ผู้ใช้ เช่น ผลลัพธ์ หรือ คำเตือน ซึ่งมีรูปแบบอย่างง่าย ดังนี้

รูปแบบ

MsgBox("ข้อความ")

หรือ

MessageBox.Show("ข้อความที่ต้องการแสดง", "ข้อความบน TitleBar",
MessageBoxButtons, MessageBoxIcon, MessageBoxDefaultButton)






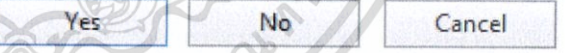
เช่น

โปรแกรมที่ 2.12 การใช้ MessageBox.Show

```
MessageBox.Show ("ต้องการออกจากโปรแกรมใช่หรือไม่ ?", _
"ออกจากโปรแกรม", MessageBoxButtons.YesNo, MessageBoxIcon.Question, _
MessageBoxDefaultButton.Button1)
```






MessageBoxButtons เป็นการแสดงปุ่มกด ได้แก่

ตารางที่ 2.7 ปุ่ม MessageBoxButtons


คำสั่ง	ความหมาย
MessageBoxButtons.AbortRetryIgnore	
MessageBoxButtons.OK	
MessageBoxButtons.OKCancel	
MessageBoxButtons.RetryCancel	
คำสั่ง	ความหมาย
MessageBoxButtons.YesNo	
MessageBoxButtons.YesNoCancel	

MessageBoxIcon เป็นการแสดงรูปแบบ icon บน MessageBox ได้แก่

ตารางที่ 2.8 ปุ่ม MessageBoxIcon

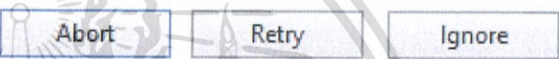

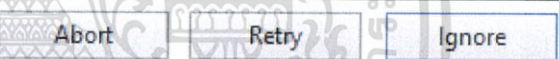
คำสั่ง	ความหมาย
MessageBoxIcon.Asterisk	
MessageBoxIcon.Information	
MessageBoxIcon.Question	
MessageBoxIcon.Error	
MessageBoxIcon.Hand	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MessageBoxIcon.Stop	
MessageBoxIcon.Exclamation MessageBoxIcon.Warning	
MessageBoxIcon.None	ไม่แสดงรูปภาพ

MessageBoxDefaultButton เป็นการกำหนดให้ปุ่มใด Focus ได้แก่

ตารางที่ 2.9 ปุ่ม MessageBoxDefaultButton

คำสั่ง	ความหมาย
MessageBoxDefaultButton.Button1	
MessageBoxDefaultButton.Button2	
คำสั่ง	ความหมาย
MessageBoxDefaultButton.Button3	

2.2.11 การขึ้นบรรทัดใหม่

ปกติแล้วสตริงทั้งหมดจะถูกเขียนที่บรรทัดเดียวกัน ซึ่งบางครั้งอาจมีความยาวมากเกินไป ดังนั้น Visual Basic จะมีคำสั่งที่ใช้กำหนดให้ขึ้นบรรทัดใหม่ คือ vbNewLine โดยการนำมาเชื่อมต่อกับสตริงด้วยโอเปอร์เรเตอร์ "&" ณ จุดที่ต้องการขึ้นบรรทัดใหม่ เช่น

โปรแกรมที่ 2.13 การขึ้นบรรทัดใหม่

```
Dim str As String
str = "Visual Studio 2010 ประกอบด้วย" & vbNewLine
str &= "Visual Basic 2010" & vbNewLine
str &= "Visual C# 2010 " & vbNewLine
str &= "Visual C++ 2010"
MessageBox.Show (str, "แสดงข้อความขึ้นบรรทัดใหม่")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.12 โครงสร้างคำสั่งควบคุมการทำงานของโปรแกรม

2.2.12.1 การตรวจสอบเงื่อนไขด้วย if (โครงสร้างแบบเลือกทำ)

คำสั่ง If จะเป็นการตรวจสอบเงื่อนไขที่อยู่ต่อจาก If ว่าเป็น จริง หรือ เท็จ ถ้าเงื่อนไขเป็นจริง (True) จะทำตามคำสั่งที่อยู่ต่อจาก Then... ถึง End If แต่ถ้าเงื่อนไขเป็นเท็จ (False) จะไม่ทำตามคำสั่งที่อยู่ระหว่าง Then ถึง End If

รูปแบบคำสั่ง

โปรแกรมที่ 2.14 รูปแบบคำสั่ง if

```
If < เงื่อนไข > Then
    คำสั่งที่ปฏิบัติเมื่อเงื่อนไขเป็นจริง
End If
```

เช่น

โปรแกรมที่ 2.15 การใช้คำสั่ง if

```
Dim x As Integer = 120
If x < 30 Then
    MessageBox.Show ("x = " + x.ToString())
'MessageBox.Show("x = " &
x)
End If
```

1) คำสั่ง If...Then...Else

เป็นคำสั่งในการตรวจสอบเงื่อนไขที่อยู่ต่อจาก If ว่าเป็นจริงหรือเท็จ ถ้าเงื่อนไขเป็นจริงจะทำตามคำสั่งที่อยู่หลังคำสั่ง Then แต่ถ้าเงื่อนไขเป็นเท็จจะทำตามคำสั่งที่อยู่หลังคำสั่ง Else

รูปแบบคำสั่ง

โปรแกรมที่ 2.16 รูปแบบคำสั่ง if else

```
If < เงื่อนไข > Then
    คำสั่งที่ปฏิบัติเมื่อเงื่อนไขเป็นจริง
Else
    คำสั่งที่ปฏิบัติเมื่อเงื่อนไขเป็นเท็จ
End If
```

เช่น

โปรแกรมที่ 2.17 การใช้คำสั่ง if else

```
Dim x As Integer = 20
If x < 30 Then
    MessageBox.Show (" x มีค่าน้อยกว่า 30 x = " + x.ToString())
Else
    MessageBox.Show (" x มีค่ามากกว่า 30 x = " + x.ToString())
End If
```

1) การใช้คำสั่ง If ซ้อนกัน (Nest if)

ในกรณีที่ต้องการตรวจสอบเงื่อนไขหลาย ๆ เงื่อนไข จำเป็นต้องใช้คำสั่ง If ซ้อนกันหลายชั้น
รูปแบบคำสั่ง

โปรแกรมที่ 2.18 รูปแบบคำสั่ง if ซ้อนกัน (Nest if)

```
If < เงื่อนไข 1 > Then
    คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ 1 เป็นจริง
    If < เงื่อนไข 2 > Then
        คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ 2 เป็นจริง
        If < เงื่อนไข n > Then
            คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ n เป็นจริง
        End If
    End If
Else
    คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ 1 เป็นเท็จ
End If
```

1) การใช้คำสั่ง If...Elseif

ใช้สำหรับการตรวจสอบเงื่อนไขในกรณีที่มีหลาย ๆ เงื่อนไข ทำให้สะดวกต่อการใช้งาน
รูปแบบคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่ 2.19 รูปแบบคำสั่ง if...elseif

```

If < เงื่อนไข 1 > Then
    คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ 1 เป็นจริง
    ElseIf < เงื่อนไข 2 > Then
        คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ 2 เป็นจริง
        ElseIf < เงื่อนไข 3 > Then
            คำสั่งที่ปฏิบัติ ถ้าเงื่อนไขที่ 3 เป็นจริง
            :
            :
            :
        Else
            คำสั่งที่ปฏิบัติ ในกรณีที่ไม่มีเงื่อนไขใดเป็นจริง
End If

```

1) การใช้คำสั่ง If

คำสั่ง If เป็นฟังก์ชันในการตรวจสอบเงื่อนไข ถ้าเงื่อนไขเป็นจริงให้ทำอะไร ถ้าเงื่อนไขเป็นเท็จ ให้ทำอะไร

รูปแบบ

IIf (เงื่อนไข, ผลลัพธ์กรณีเงื่อนไขเป็นจริง, ผลลัพธ์ในกรณี
เงื่อนไขเป็นเท็จ)

โปรแกรมที่ 2.20 การใช้คำสั่ง iif

```

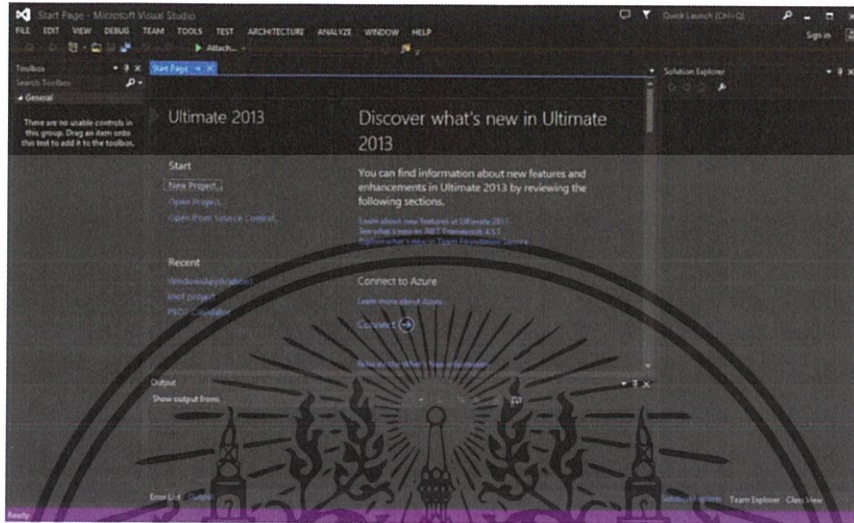
Dim number As Integer
Dim str As String
number = InputBox ("กรณำป้อนตัวเลขจำนวนเต็ม")
str = IIf ( (number Mod 2 = 0) , "เลขคู่" , "เลขคี่ ")
MessageBox.Show (str, "ผลลัพธ์")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การใช้งานโปรแกรม Visual Studio 2013

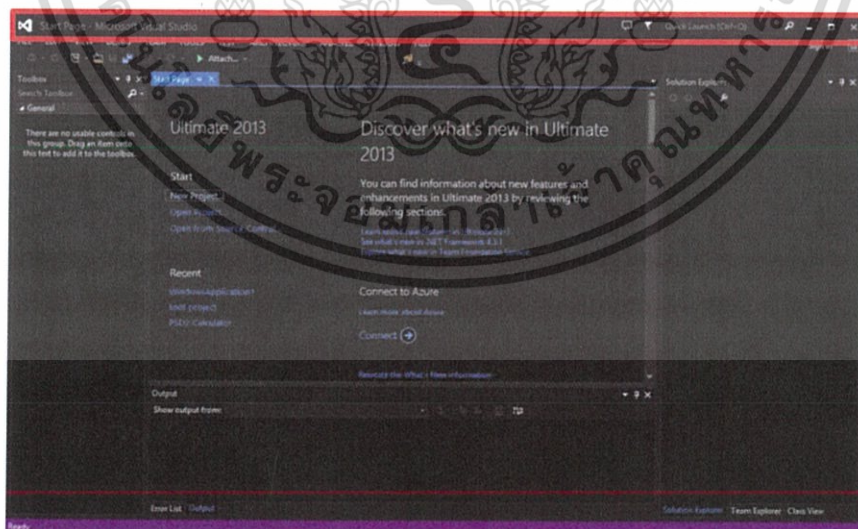
หน้าต่างโปรแกรม Microsoft Visual Studio 2013



รูปที่ 2.5 หน้าต่างโปรแกรม Microsoft Visual Studio 2013

2.3.1 ส่วนประกอบที่สำคัญของหน้าต่างโปรแกรม

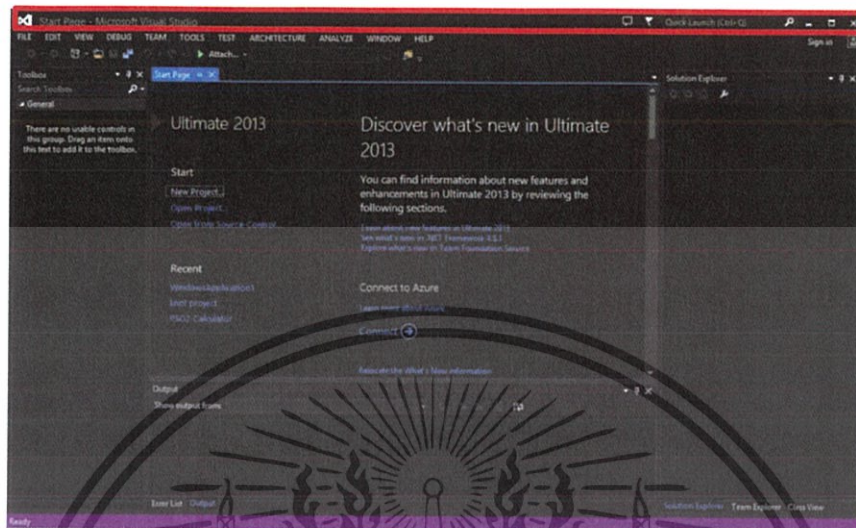
- 1) แถบหัวเรื่อง title bar : เป็นแถบแสดงหัวเรื่องโดยปกติจะบอกชื่อของโปรแกรม



รูปที่ 2.6 แสดงแถบหัวเรื่อง title bar

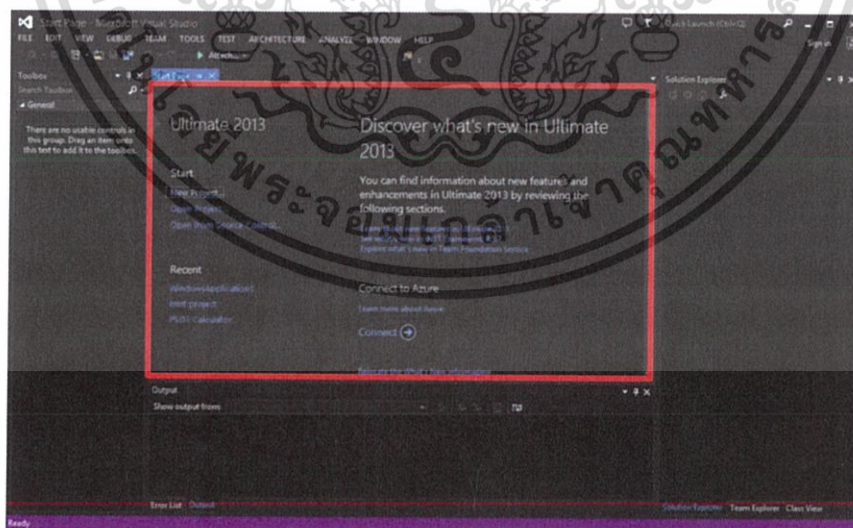
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) แถบรายการคำสั่ง menu bar : เป็นแถบแสดงรายการคำสั่งต่างๆ ของโปรแกรม



รูปที่ 2.7 แสดงแถบเมนู menu bar

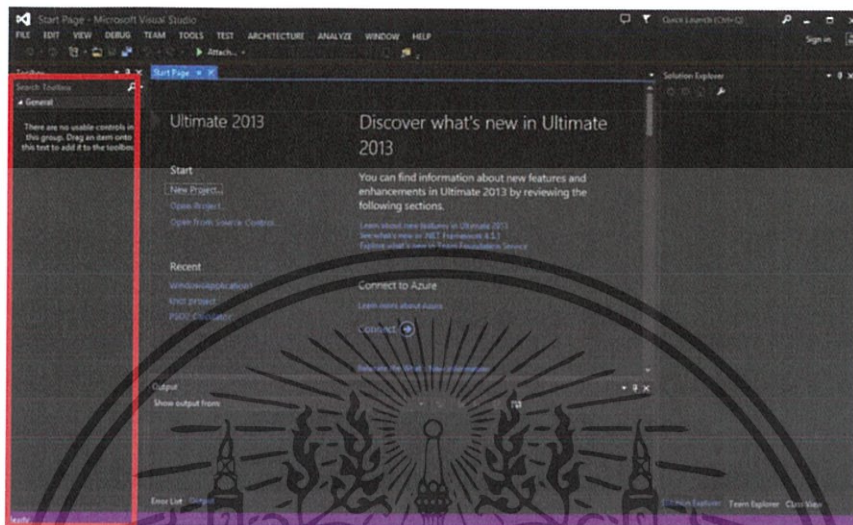
- 3) หน้าต่างเริ่ม (Start Page) : เป็นหน้าต่างเริ่มการทำงานของโปรแกรม ซึ่งจะมีส่วนที่แสดงรายชื่อโปรเจกต์ที่เราได้เคยสร้างไว้แล้วในกรณีทีเพิ่งเปิดใช้งานครั้งแรกจะไม่มีรายชื่อปรากฏอยู่



รูปที่ 2.8 แสดงหน้าต่างเริ่ม Start Page

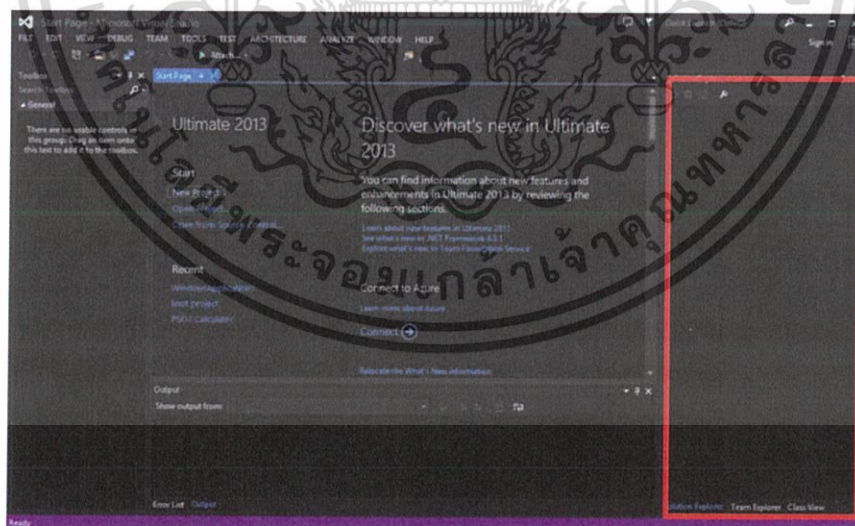
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 4) กล่องเครื่องมือ (ToolBox) : เป็นส่วนที่เก็บเครื่องมือต่างๆ ที่เราจะนำมาใช้งานในโปรเจกต์ของเรา แต่ในกรณียังไม่มีเครื่องมือเนื่องจากเรายังไม่ได้สร้าง Project ใดๆ



รูปที่ 2.9 แสดงแถบเครื่องมือ

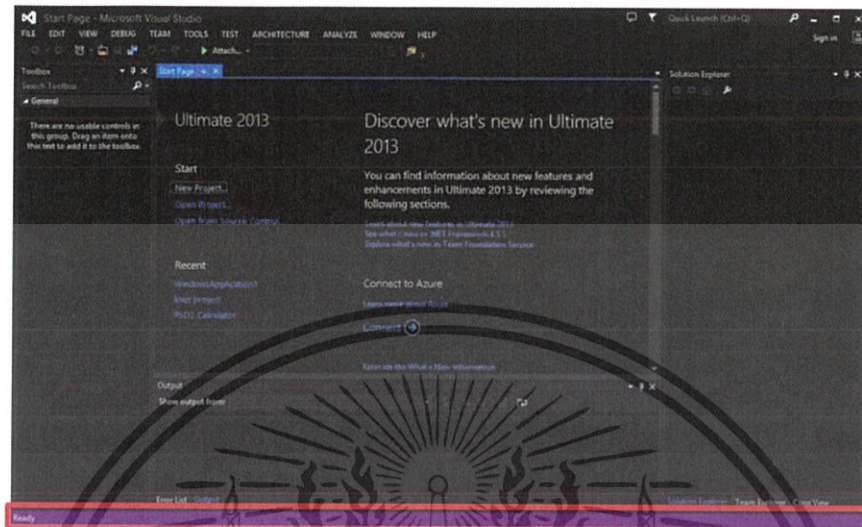
- 5) หน้าต่าง Solution Explorer : หน้าต่างที่แสดงรายชื่อและจัดการกับโปรเจกต์ของเรา



รูปที่ 2.10 แสดงหน้าต่าง Solution Explorer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) แถบสถานะ Status Bar : เป็นแถบที่มีไว้สำหรับบอกสถานะของการทำงานของโปรแกรม

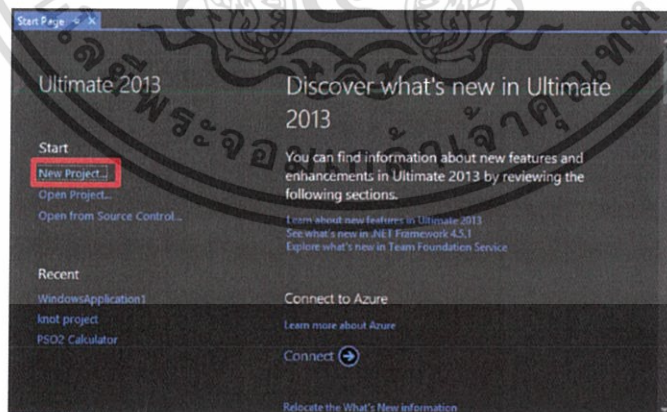


รูปที่ 2.11 แสดงแถบสถานะ Status Bar

2.3.2 การสร้าง Project (Create New Project)

ในการเขียนโปรแกรมของภาษา Visual Basic การสร้างโปรแกรมเริ่มด้วยการสร้างโปรเจกต์ใหม่ซึ่งสามารถทำได้ 2 วิธีดังนี้

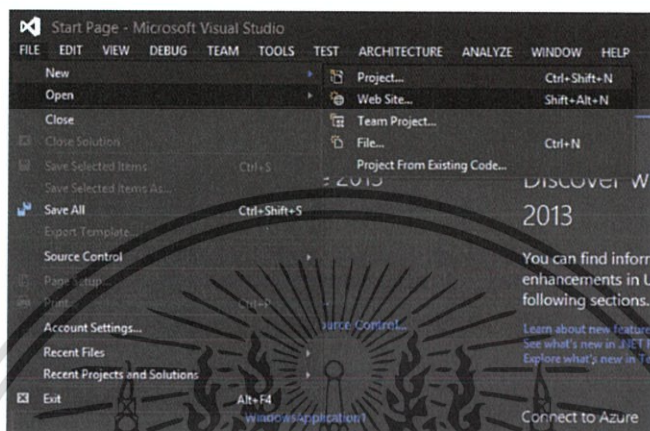
- 1) คลิกที่ ปุ่ม Create New Project ในหน้าต่างเริ่มต้น (Start Page) ดังรูป



รูปที่ 2.12 แสดงการสร้าง Project ใหม่จากหน้าเริ่มต้น (Start Page)

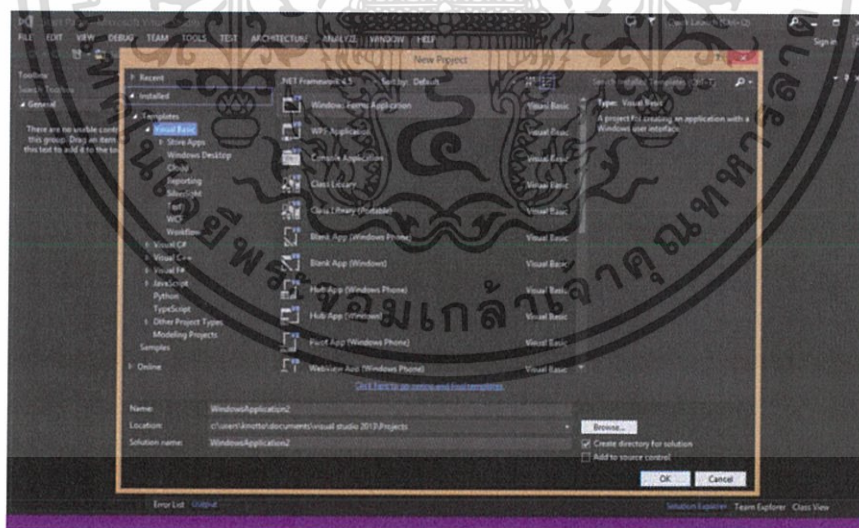
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) สามารถทำได้โดย
 - 2.1) ไปที่เมนู File เลือก New
 - 2.2) เลือก Project



รูปที่ 2.13 แสดงการสร้าง Project ใหม่จากเมนู File

จากนั้นจะมีหน้าต่างปรากฏขึ้นดังรูปที่ 2.14



รูปที่ 2.14 แสดงหน้าต่างการสร้าง Project ใหม่

2.3.3 รายละเอียดของหน้าต่าง New Project มีดังนี้

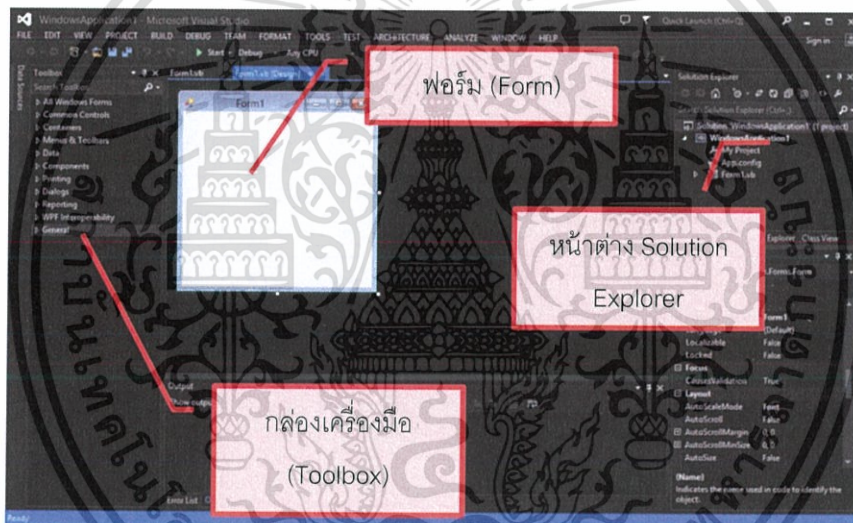
- 1) Project Types : ส่วนนี้ไว้สำหรับเราให้เลือกภาษาที่ต้องการพัฒนา
- 2) Templates : เลือกลักษณะของ Project ที่เราต้องการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) Name : เป็นส่วนที่ต้องกำหนดชื่อของโปรเจกต์
- 4) Location : เก็บที่อยู่ของโปรเจกต์
- 5) Solution Name : เป็นการกำหนดชื่อของ Solution ซึ่งปกติจะเป็นชื่อเดียวกับโปรเจกต์แต่อาจตั้งเป็นชื่ออื่นได้

ในที่นี่ให้เราเลือก ดังนี้

- 1) ตรง Installed Templates >> ให้เลือกเป็น Visual Basic แล้วเลือก window
- 2) ตรง Templates >> ก็ให้เลือกเป็น Window application
- 3) ตั้งชื่อโปรเจกต์ให้เรียบร้อยแล้วกดปุ่ม OK



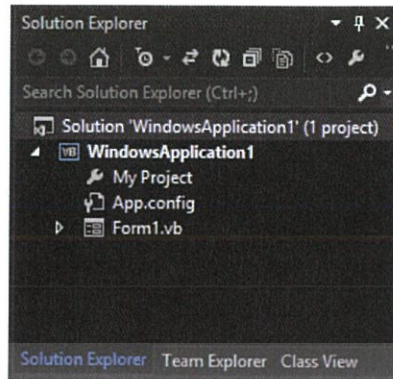
รูปที่ 2.15 แสดงหน้าต่างหลังจากสร้าง Project ใหม่

2.3.4 การทำงานกับหน้าต่างโปรแกรม

2.3.4.1 หน้าต่าง Solution

เป็นหน้าต่างที่รายละเอียดของโซลูชัน (หลายๆ โปรเจกต์รวมกันเรียกว่า Solution) โปรเจกต์รวมทั้งเรายังสามารถจัดการกับโปรเจกต์ได้โดยการเพิ่ม ลบ และเปลี่ยนชื่อของ Solution, Project และ Form

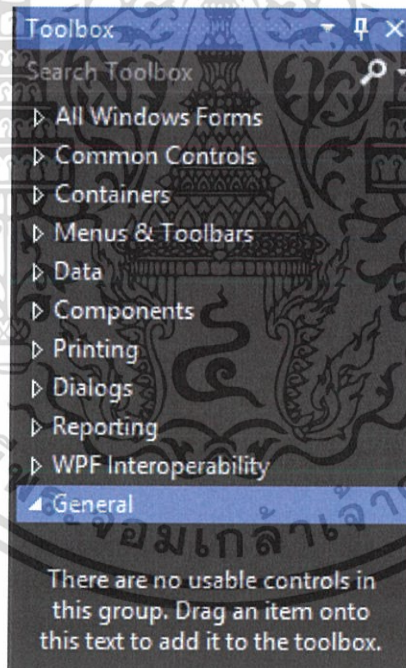
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 แสดงหน้าต่าง Solution

2.3.4.2 แถบเครื่องมือ (Tool Box)

เป็นหน้าต่างที่เก็บออบเจกต์ต่างๆ ที่เรานำมาใช้ในงานในโปรแกรม



รูปที่ 2.17 แสดงกล่องเครื่องมือ Tool Box

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

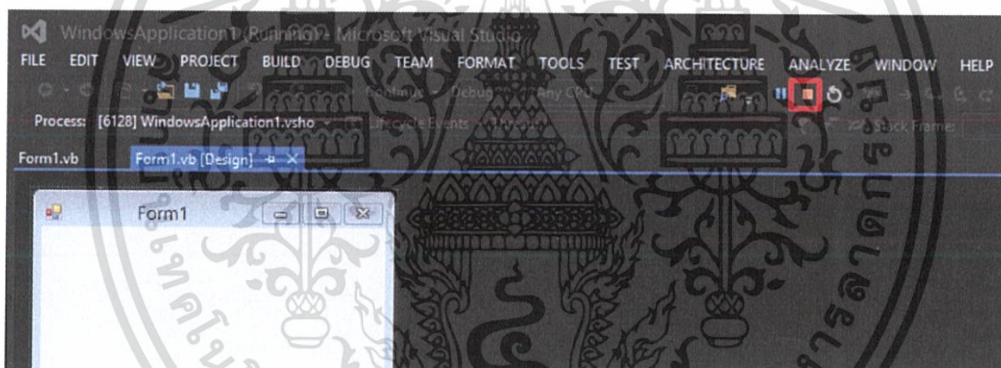
2.3.5 การทดสอบการทำงาน (RUN)

การทดสอบการทำงานหรือการรันโปรแกรมเพื่อดูผลลัพธ์ใน Visual Studio 2013 สามารถทำได้ 3 วิธีดังนี้

- 1) กดคลิกที่ปุ่ม  ในแถบเครื่องมือ Standard หรือ
- 2) ไปที่เมนู Debug เลือก Start Debugging หรือ
- 3) กดปุ่ม F5 บนคีย์บอร์ด

2.3.6 การหยุดการรัน (Stop Running)

หลังจากที่เราทำการรันโปรแกรมเพื่อทดสอบการทำงานแล้ว หากเราต้องการจะหยุดการรันหรือกลับมาแก้ไขโปรแกรมต้องทำการหยุดการรันทุกครั้ง โดยกดที่ปุ่ม Stop ในแถบเครื่องมือ Standard



รูปที่ 2.18 แสดงการหยุดการรันโดยกดที่ปุ่ม Stop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและพัฒนา

3.1 รายละเอียดของโปรแกรมที่พัฒนา

3.1.1 อินพุทของโปรแกรม

- 1) การตั้งค่าให้โปรแกรมทำงานทันทีเมื่อเปิดเครื่องคอมพิวเตอร์
- 2) กล้อง (Webcam) ที่ต้องการใช้งาน
- 3) อีเมลล์ของผู้ใช้งาน
- 4) พาสเวิร์ดของผู้ใช้งาน
- 5) Hotkey ในการกดเพื่อแสดง Unlock Box
- 6) เวลาเพื่อนับถอยหลังให้ล็อคเครื่องคอมพิวเตอร์เมื่อไม่ได้ใช้งาน
- 7) Schedule ของโปรแกรมว่าให้ล็อคเครื่องคอมพิวเตอร์เมื่อใด

3.1.2 เอาท์พุทของแบบจำลอง

- 1) แสดงหน้าต่างโปรแกรมเพื่อล็อคไม่ให้คนอื่นเข้ามาใช้งานเครื่องคอมพิวเตอร์
- 2) รูปถ่ายเมื่อมีผู้บุกรุกเครื่องคอมพิวเตอร์
- 3) อีเมลล์ส่งไปบอกผู้ใช้งานว่ามีผู้บุกรุก

3.1.3 ฟังก์ชันการทำงาน

- 1) โปรแกรมสามารถป้องกันไม่ให้ผู้บุกรุกกดปุ่มใดๆหรือกระทำการใดๆต่อคอมพิวเตอร์ได้เลยจนกว่าจะพิมพ์รหัสผ่าน
- 2) โปรแกรมสามารถถ่ายรูปผู้บุกรุกได้ทันที เมื่อผู้บุกรุกกดปุ่มตามที่กำหนดไว้
- 3) โปรแกรมจะทำการส่งอีเมลล์ไปบอกกับเจ้าของเครื่องคอมพิวเตอร์ว่ามีผู้บุกรุก
- 4) รูปที่ถ่ายผู้บุกรุกจะถูกแนบมากับอีเมลล์ส่งให้กับเจ้าของเครื่อง
- 5) รูปที่ถ่ายผู้บุกรุกจะถูกบันทึกไว้ในเครื่องคอมพิวเตอร์
- 6) ผู้ใช้สามารถตั้งเวลาให้ล็อคเครื่องคอมพิวเตอร์ได้ ในขณะที่ไม่ได้ใช้งาน
- 7) ผู้ใช้สามารถตั้ง Schedule ของโปรแกรมได้ ว่าให้ล็อคเครื่องเมื่อใด
- 8) ผู้ใช้สามารถตั้ง Hotkey ได้ตามใจชอบ เพื่อให้แสดง Unlock Box ขึ้นมา

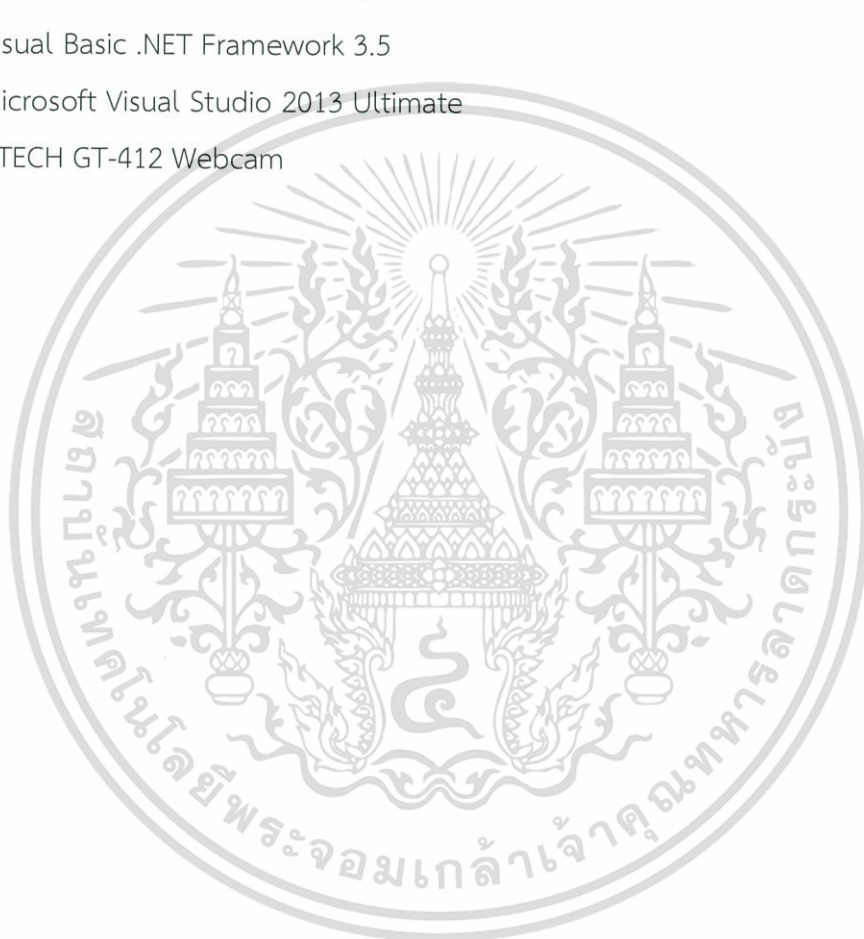
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 ข้อจำกัดของโปรแกรมที่พัฒนา

- 1) โปรแกรมที่พัฒนารองรับเฉพาะระบบปฏิบัติการ Windows เท่านั้น
- 2) ไม่สามารถป้องกันการปิดเครื่องหรือ logout ได้ ในระบบปฏิบัติการ Windows 8 ขึ้นไป
- 3) ไม่สามารถเชื่อมต่อ กล้อง (Webcam) พร้อมกันหลายๆตัวได้

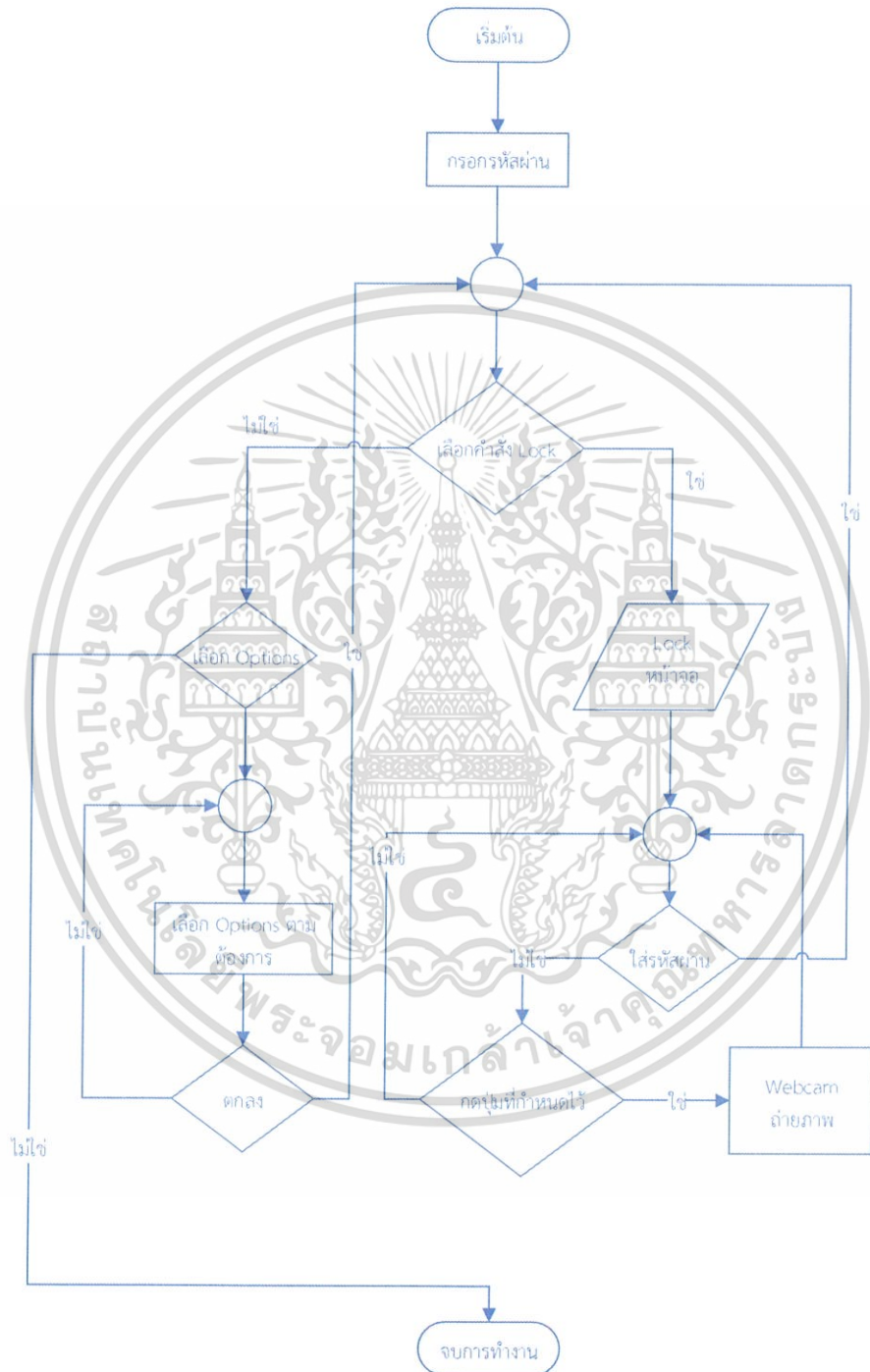
3.1.5 ภาษาและเครื่องมือที่ใช้ในการพัฒนา

- 1) Visual Basic .NET Framework 3.5
- 2) Microsoft Visual Studio 2013 Ultimate
- 3) GTECH GT-412 Webcam



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการทำงานของโปรแกรม



รูปที่ 3.1 ขั้นตอนการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 แผนภาพ UML (Unified Modeling Language Diagram)

3.3.1 Use Case Diagram

โนโปรแกรมมีผู้ใช้ระบบ (User) กับผู้ประสงค์ร้าย (Attacker) โดยแยกเป็นระบบย่อยอย่างชัดเจน คือ

- 1) Lock หน้าจอ
- 2) ตัวเลือก (Options)
- 3) ออกจากโปรแกรม (Exit)

โดยระบบ “Lock หน้าจอ” จะมีการเรียกใช้งาน (include) ระบบย่อย ได้แก่ “ใส่รหัสผ่าน” โดยมีรายละเอียดการทำงานคือ เมื่อมีการ Lock หน้าจอ จะเรียก ใส่รหัสผ่าน มาทำงาน

ระบบ “ใส่รหัสผ่าน” จะมีการเรียกใช้งานในบางครั้ง (Extend) ได้แก่ การถ่ายภาพจากกล้อง (Webcam Capture) โดยเมื่อ Attacker เข้ามาใช้งานจะเจอกับการใส่รหัสผ่าน เมื่อ Attacker ใส่รหัสผ่านผิด หรือกดปุ่มที่กำหนดเอาไว้ก็จะเรียกใช้งาน การถ่ายภาพจากกล้อง (Webcam Capture)



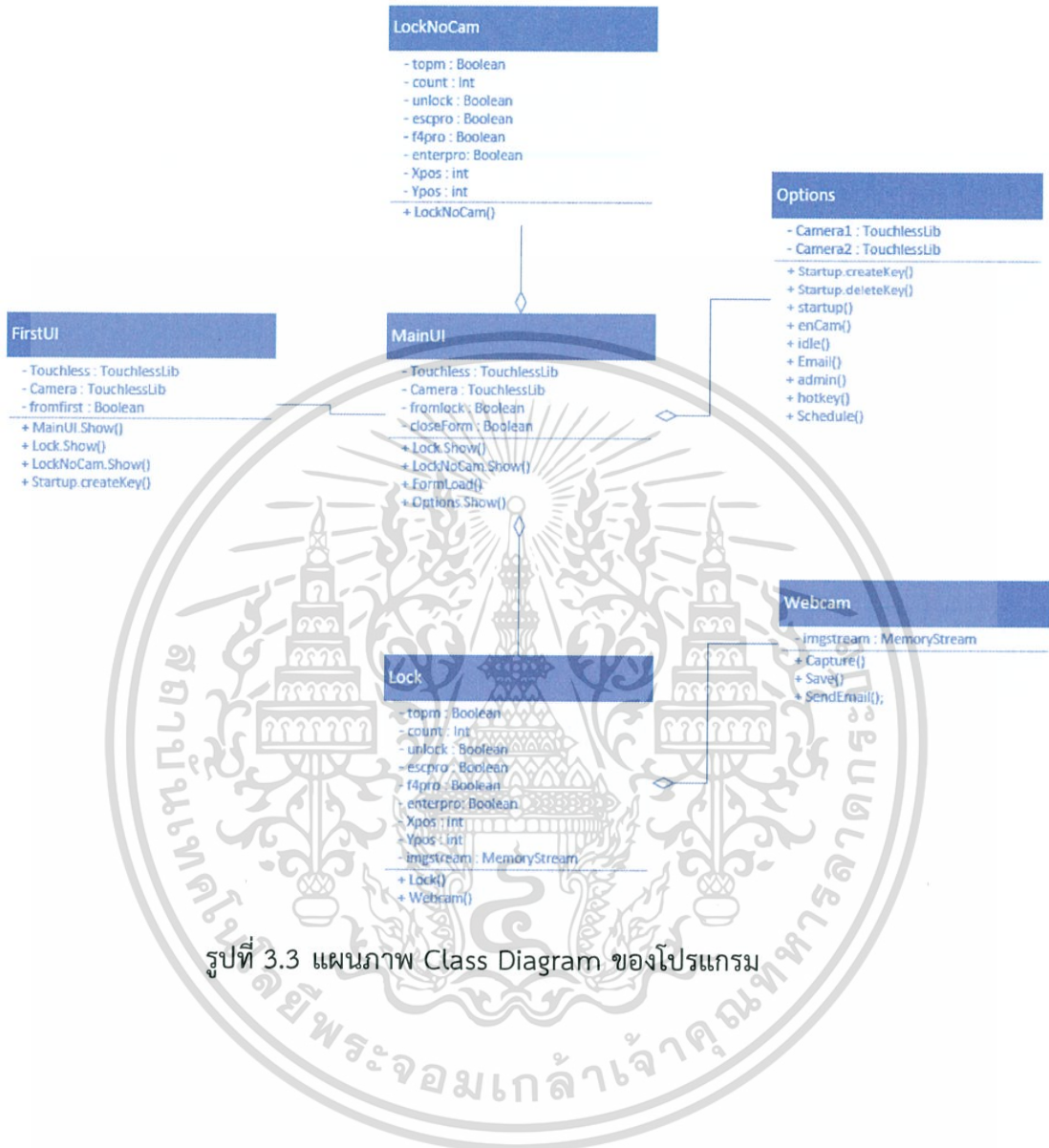
รูปที่ 3.2 แผนภาพ Use Case สำหรับการใช้งานโปรแกรม

3.3.2 Class Diagram

ประกอบไปด้วยคลาสและออบเจกต์ต่างๆดัง3.3 โดยมีรายละเอียดดังนี้

- 1) Class FirstUI ส่วนติดต่อผู้ใช้แบบกราฟิกเมื่อเปิดโปรแกรมครั้งแรก
- 2) Class MainUI ส่วนติดต่อผู้ใช้หลักของโปรแกรม
- 3) Class Options ตั้งค่าการใช้งานต่างๆของโปรแกรม
- 4) Class LockNoCam ทำการ Lock หน้าจอคอมพิวเตอร์
- 5) Class Lock ทำการ Lock หน้าจอคอมพิวเตอร์พร้อมทั้งระบบกล้อง (Webcam)
- 6) Class Webcam บันทึกภาพจากกล้อง (Webcam)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แผนภาพ Class Diagram ของโปรแกรม

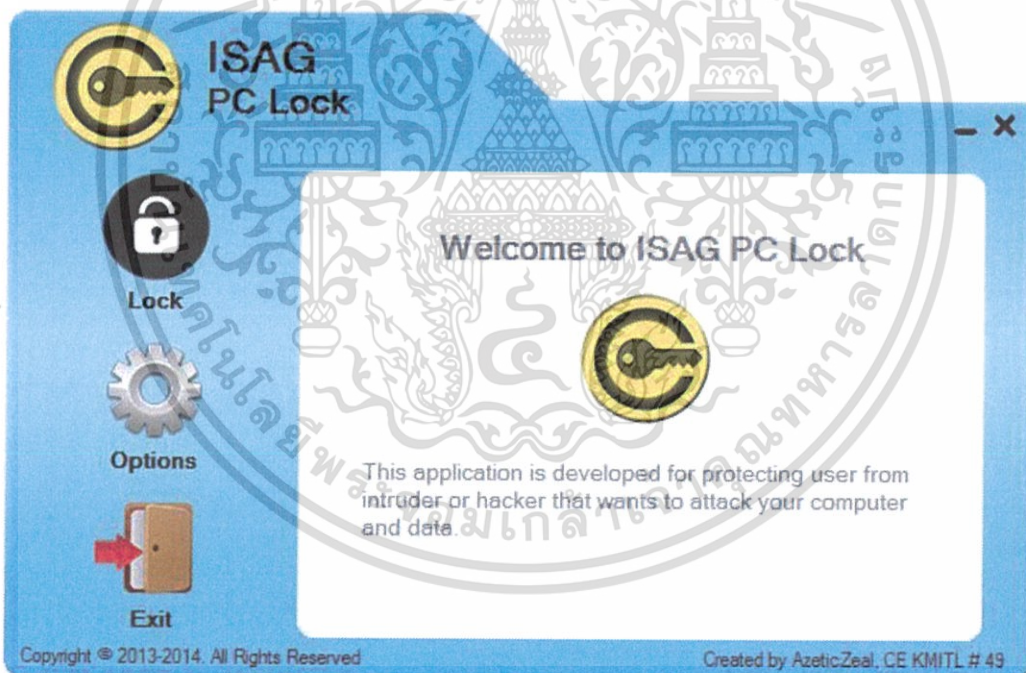
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบส่วนติดต่อกับผู้ใช้งาน

ผู้พัฒนาได้ออกแบบส่วนติดต่อผู้ใช้ของโปรแกรมไว้ดัง 3.4 – 3.11 ดังนี้

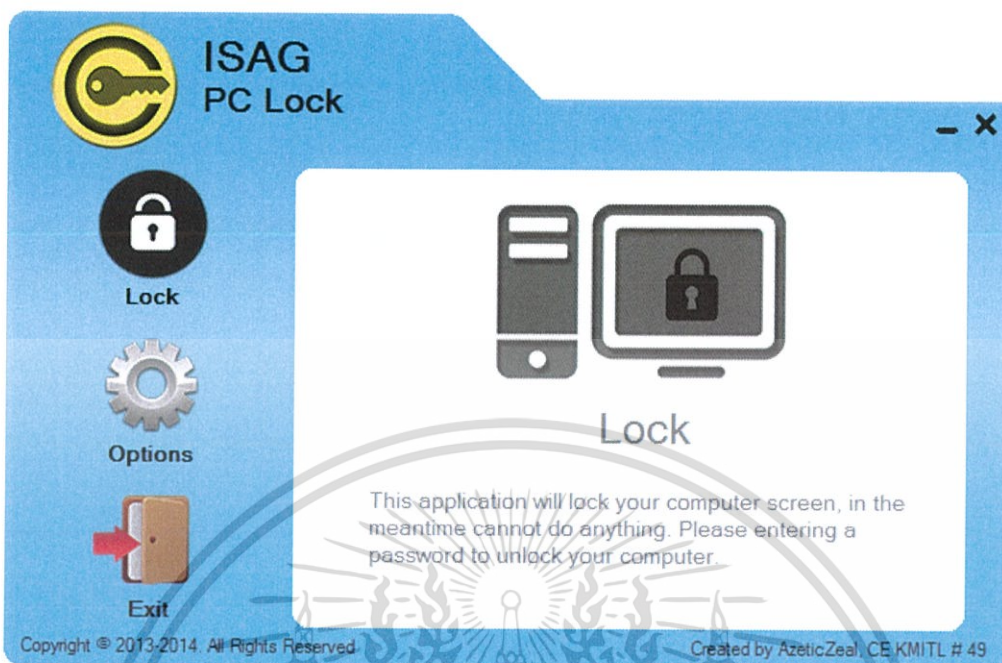


รูปที่ 3.4 หน้าแรกของโปรแกรม

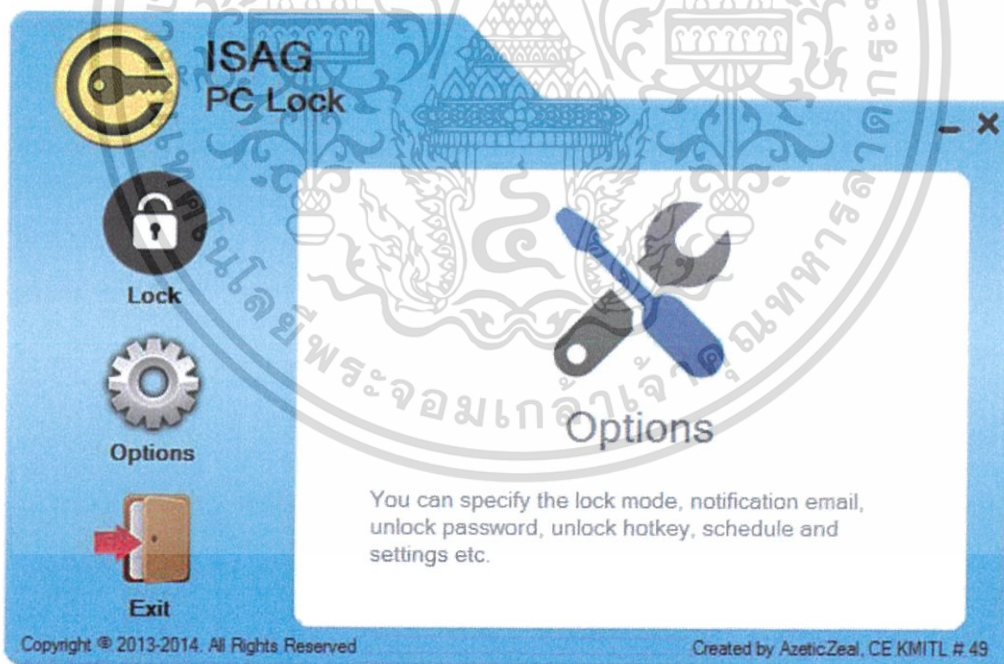


รูปที่ 3.5 หน้าหลักของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

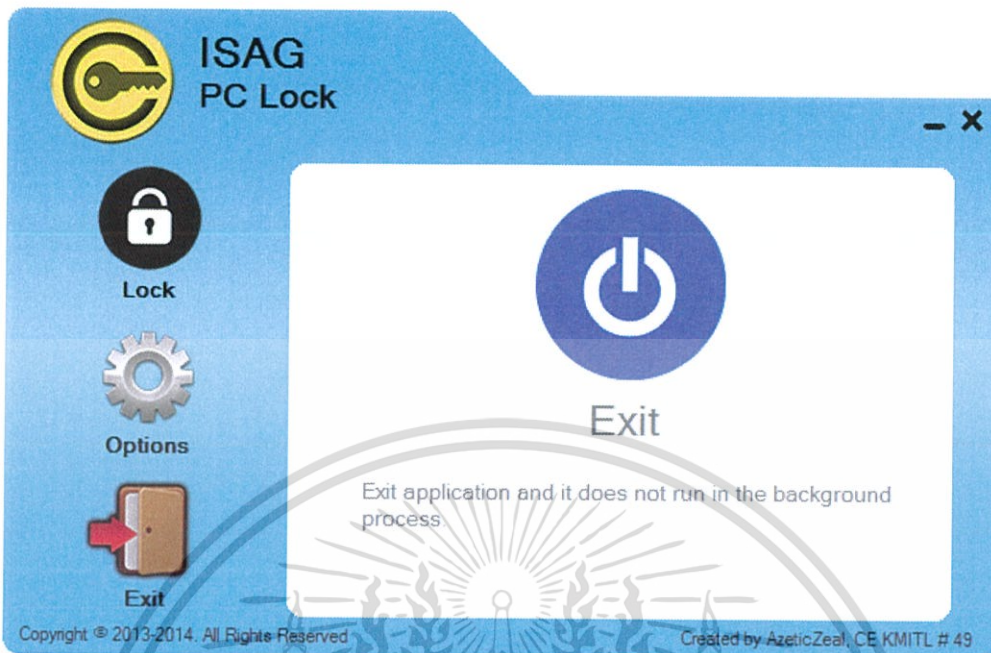


รูปที่ 3.6 หน้าหลักของโปรแกรมเมื่อนำเมาส์ไปวางที่ปุ่ม Lock

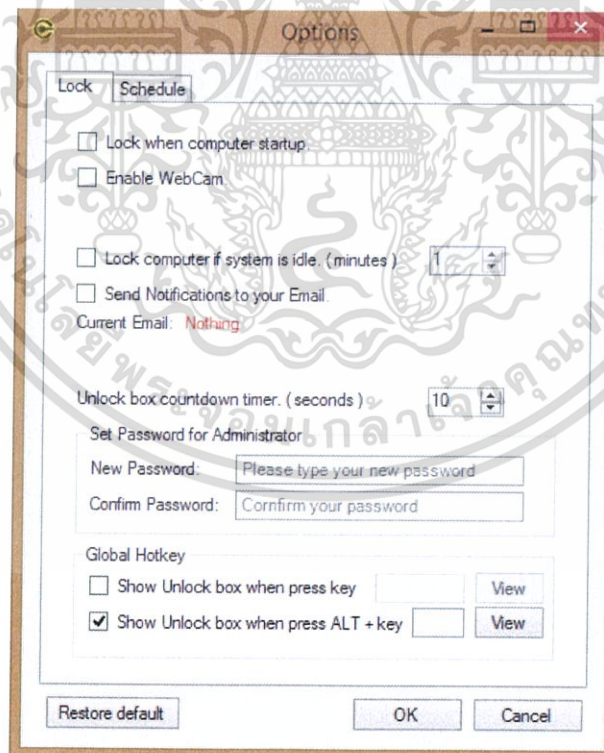


รูปที่ 3.7 หน้าหลักของโปรแกรมเมื่อนำเมาส์ไปวางที่ปุ่ม Options

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

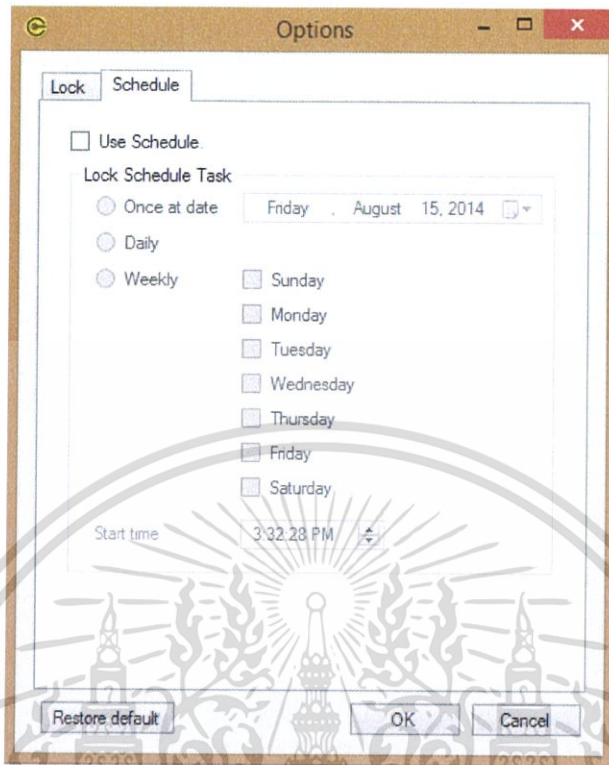


รูปที่ 3.8 หน้าหลักของโปรแกรมเมื่อนำเมาส์ไปวางที่ปุ่ม Exit



รูปที่ 3.9 หน้า Options ของโปรแกรม (TabPage1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 หน้า Options ของโปรแกรม (TabPage2)



รูปที่ 3.11 หน้า Lock Screen ของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทำงาน

ในบทนี้จะกล่าวถึงการทดลองใช้งานโปรแกรมที่ได้พัฒนาขึ้น อธิบายระบบต่างๆของโปรแกรม รายงานสรุปผลการทดลอง และคู่มือการใช้งาน

4.1 การทดลองใช้โปรแกรมเบื้องต้น

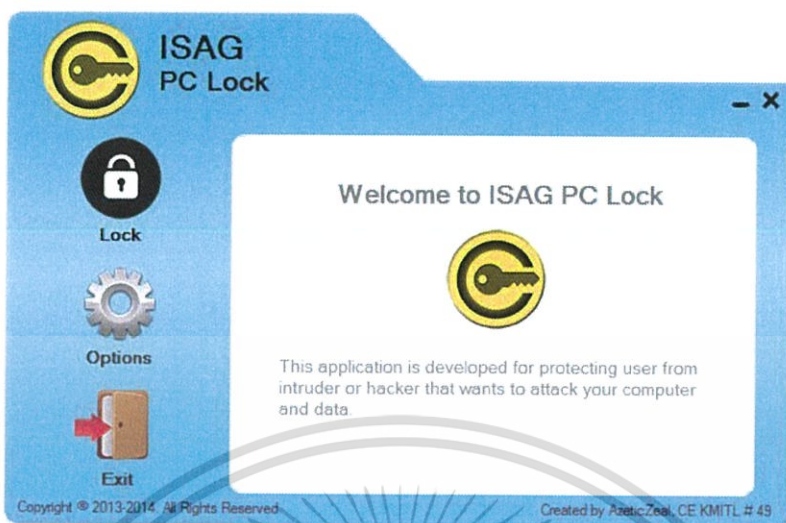
เมื่อใช้งานโปรแกรมจะแสดงหน้าต่างดังรูปที่ 4.1



รูปที่ 4.1 หน้าต่างแรกของโปรแกรม

จากนั้นผู้ใช้ต้องทำการพิมพ์รหัสผ่านเพื่อใช้งานโปรแกรม โดยรหัสผ่านตั้งต้นคือ 1234 โดยผู้ใช้จะสามารถกำหนดรหัสเป็นอะไรก็ได้ เมื่อกดปุ่ม Login จะเข้าสู่หน้าต่างหลักของโปรแกรมดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

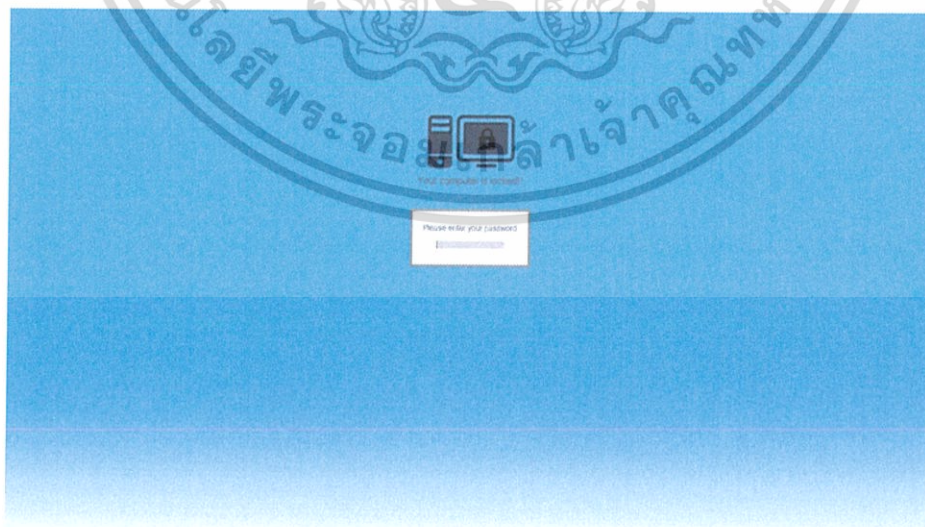


รูปที่ 4.2 หน้าต่างหลักของโปรแกรม

เมื่อเข้ามาหน้าต่างหลักแล้วจะมีปุ่มให้กดอยู่ 3 ปุ่มด้วยกัน คือ Lock Options และ Exit ซึ่งแต่ละปุ่มจะมีการทำงานดังนี้

4.1.1 การใช้ปุ่ม Lock

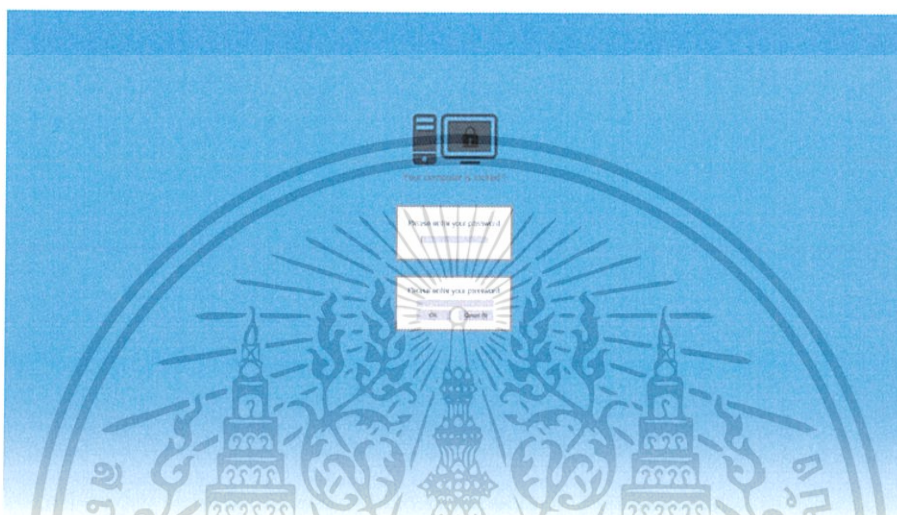
เมื่อกดปุ่มนี้จะเป็นการสั่งให้โปรแกรมล๊อคหน้าจอคอมพิวเตอร์ จะไม่สามารถทำอะไรได้นอกจากพิมพ์รหัสผ่านเพื่อทำการปลดล๊อคหน้าจอ



รูปที่ 4.3 หน้า Lock Screen ของโปรแกรม

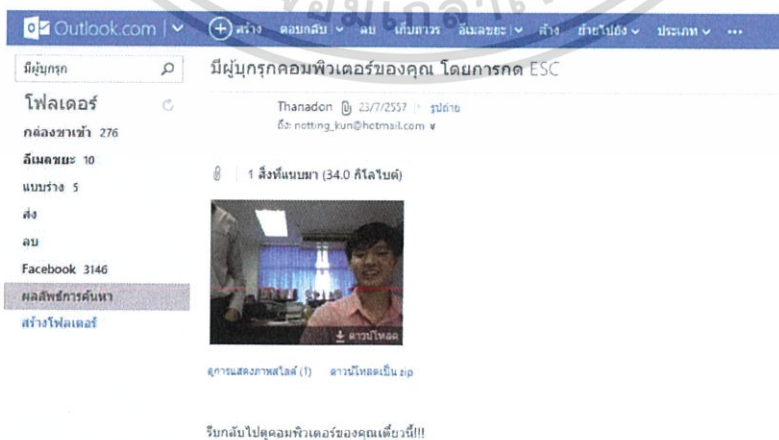
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยการปลดล็อคหน้าจอ นั้น ผู้ใช้จะต้องใส่ Hotkey ลงไปเพื่อที่จะให้ช่องใส่รหัสผ่านที่แท้จริงปรากฏขึ้น ซึ่งช่องใส่รหัสผ่านที่ปรากฏให้เห็นจะเป็นเพียงช่องหลอกเท่านั้น โดยปุ่มที่เป็นค่าตั้งต้นที่ทำให้ช่องใส่รหัสผ่านที่แท้จริงปรากฏขึ้นคือ ALT+S ซึ่งผู้ใช้สามารถกำหนดเป็น ALT+? อักษรภาษาอังกฤษตัวใดก็ได้ เมื่อกดแล้ว จะได้ดังรูปที่ 4.4



รูปที่ 4.4 หน้า Lock Screen แสดงช่องใส่รหัสผ่านที่แท้จริง

ส่วนในช่องหลอกนั้น หากมีผู้ใดมาพิมพ์รหัสผ่านใส่แล้วกด Enter โปรแกรมก็จะสั่งให้ Webcam ถ่ายรูปไว้ทันที รวมถึงกดปุ่ม ALT+F4, ESC ก็จะสามารถถ่ายไว้เช่นเดียวกัน โดยรูปที่ถ่ายไว้จะไปเก็บไว้ในโฟลเดอร์ Pictures นอกเหนือจากนี้ ผู้ใช้ยังสามารถตั้งค่าให้รูปที่ถ่ายได้ ส่งไปยัง Email ของผู้ใช้ได้อีกด้วย โดยมีตัวอย่างดังรูปที่ 4.5

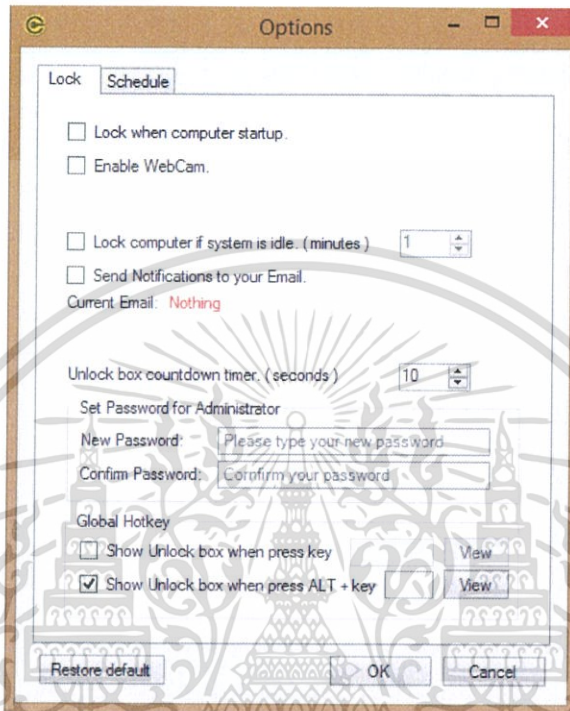


รูปที่ 4.5 Email แจ้งเตือนว่ามีผู้บุกรุก

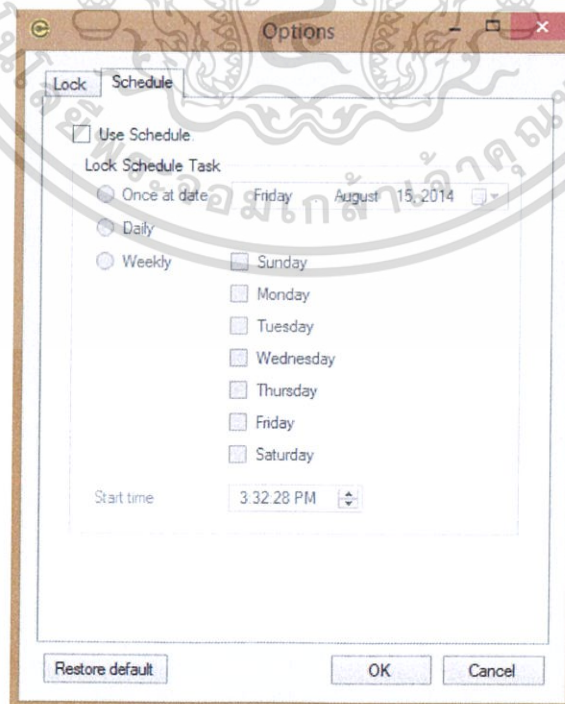
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การใช้งานปุ่ม Options

โดยปุ่มนี้จะเป็นการปรับการตั้งค่าต่างๆในโปรแกรม โดยมีหน้าต่างดังรูปที่ 4.6 และ 4.7



รูปที่ 4.6 หน้าต่าง Options ของโปรแกรม 1



รูปที่ 4.7 หน้าต่าง Options ของโปรแกรม 2

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับใช้ในเพียงการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน Options จะมีฟังก์ชันการใช้งานให้ตั้งค่าหลักๆดังนี้

- 1) Lock when computer startup
- 2) Enable WebCam
- 3) Lock computer if system is idle
- 4) Send Notification to your Email
- 5) Unlock box countdown timer
- 6) Set Password for Administrator
- 7) Global Hotkey
- 8) Use Schedule
- 9) Restore default

4.1.2.1 ฟังก์ชัน Lock when computer startup

ในฟังก์ชันนี้เมื่อผู้ใช้เลือกใช้งานแล้ว โปรแกรมจะทำการสร้าง Path ไว้ใช้ในการทำ Program Startup ของ Windows ลงไปใน Registry ของ Windows และทุกครั้งที่เปิดเครื่องคอมพิวเตอร์มา และรันหน้า Windows โปรแกรมก็จะทำการล๊อคหน้าจอทันที

4.1.2.2 Enable WebCam

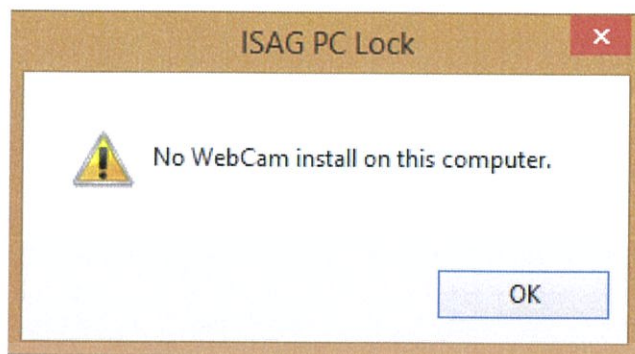
ในฟังก์ชันนี้เมื่อผู้ใช้เลือกใช้งานแล้ว จะมีรายชื่อของ WebCam ที่เชื่อมต่ออยู่กับคอมพิวเตอร์ขึ้นมาให้เลือกดังรูปที่ 4.8



รูปที่ 4.8 Enable WebCam

แต่ถ้าหากคอมพิวเตอร์ไม่มี WebCam เชื่อมต่ออยู่ จะไม่สามารถใช้งานฟังก์ชันนี้ได้ โดยโปรแกรมจะแสดง Dialog box ดังรูปที่ 4.9

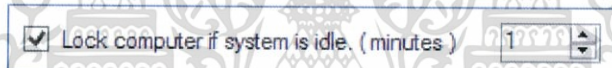
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 ไม่ได้ติดตั้ง WebCam

4.1.2.3 Lock computer if system is idle

ในฟังก์ชันนี้เมื่อผู้ใช้เลือกใช้งานแล้ว โปรแกรมจะให้กำหนดเวลาว่าจะใช้เวลากี่นาทีในการล็อคหน้าจอในขณะที่ไม่ได้ใช้งาน โดยค่าเริ่มต้นกำหนดไว้ที่ 1 นาที และเป็นค่าที่น้อยที่สุดที่สามารถตั้งค่าได้

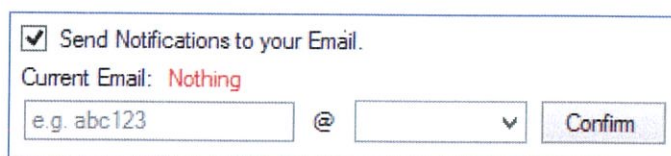


รูปที่ 4.10 Lock computer if system is idle

หมายเหตุ ถ้าเปิดโปรแกรมหรือแอปพลิเคชันที่ Full Screen ไว้ โปรแกรมจะไม่ล็อคหน้าจอ เช่น Youtube, Windows Media Player เป็นต้น

4.1.2.4 Send Notification to your Email

ในฟังก์ชันนี้เมื่อผู้ใช้เลือกใช้งานแล้ว โปรแกรมจะมีช่องให้ผู้ใช้ใส่ Email ที่ต้องการให้ส่งไป ดังรูปที่ 4.11



รูปที่ 4.11 Send Notification to your Email

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Current Email จะเป็นการบอกว่าเราใส่ Email ไปแล้วหรือยัง โดยในฟังก์ชันนี้สนับสนุนแค่ Hotmail กับ Gmail เท่านั้น เมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้วกดปุ่ม Confirm จะมีหน้าต่างให้ใส่รหัสผ่านของ Email ดังรูปที่ 4.12

รูปที่ 4.12 หน้าต่างใส่รหัสผ่าน Email

เมื่อกรอกรหัสผ่านเรียบร้อยแล้วกด OK จะมีหน้าต่างยืนยันอีกรอบดังรูปที่ 4.13

รูปที่ 4.13 Confirm your Email

จากนั้น Current Email ก็จะเปลี่ยนไปเป็น Email ที่กรอก ดังรูปที่ 4.14

รูปที่ 4.14 หน้าต่างแสดงเมื่อใส่ Email แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2.5 Unlock box countdown timer

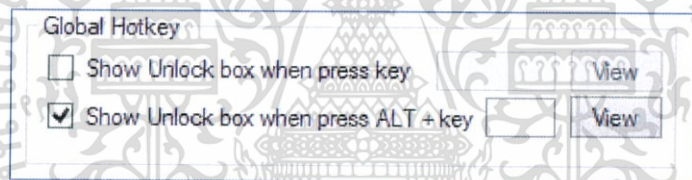
ในฟังก์ชันนี้ผู้ใช้สามารถกำหนดเวลานับถอยหลังให้กับช่องใส่รหัสผ่านที่แท้จริงได้ โดยค่าเริ่มต้นกำหนดไว้ที่ 10 วินาที ซึ่งเป็นค่าที่ต่ำที่สุดที่จะกำหนดได้ โดยผู้ใช้สามารถกำหนดได้สูงสุดที่ 100 วินาที

4.1.2.6 Set Password for Administrator

ในฟังก์ชันนี้ผู้ใช้สามารถตั้งรหัสผ่านของโปรแกรมได้ตามต้องการ

4.1.2.7 Global Hotkey

ในฟังก์ชันนี้บังคับให้ผู้ใช้ต้องเลือกอันใดอันหนึ่งเป็นอย่างน้อย หรือจะเลือกทั้งสองอันเลยก็ได้ โดยค่าเริ่มต้นจะตั้งค่าไว้ที่ ALT+key ซึ่ง Key ที่ตั้งเป็นค่าเริ่มต้นคือ S โดยเมื่อผู้ใช้กด Global Hotkey ที่หน้าล๊อคจอแล้ว จะทำให้ช่องใส่รหัสผ่านที่แท้จริงปรากฏขึ้น

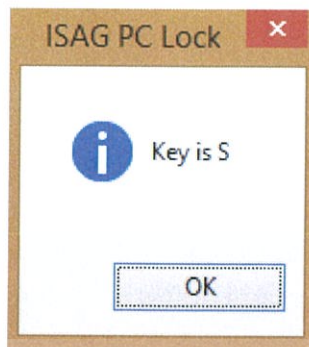


รูปที่ 4.15 Global Hotkey

โดย Show Unlock box when press key นั้นจะสามารถใส่เป็น String ก็ได้ หรือเป็นตัวอักษรตัวเดียวก็ได้เช่น ใส่คำว่า “knot” ลงไป เมื่ออยู่ในหน้าล๊อคจอ พิมพ์คำว่า “knot” ก็จะทำให้ปรากฏช่องใส่รหัสผ่านที่แท้จริงขึ้นมา

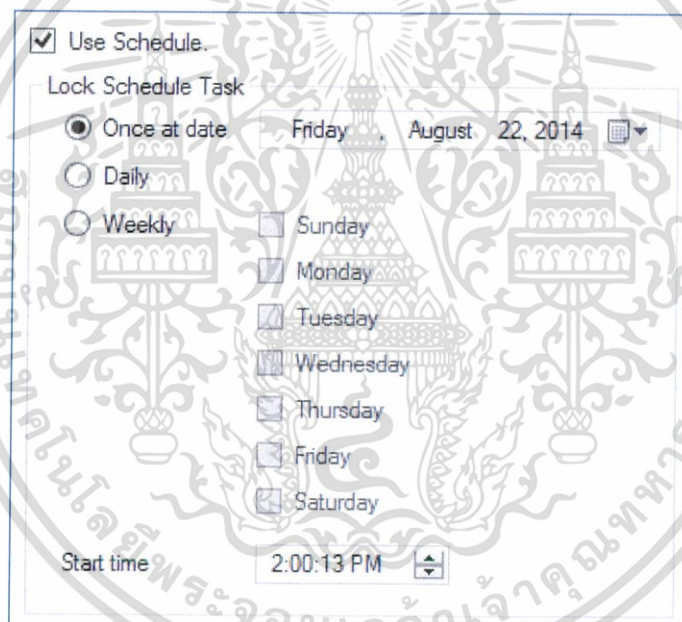
ส่วน Show Unlock box when press ALT + key นั้น สามารถตั้ง key เป็นอักษรภาษาอังกฤษ หรือตัวเลขตัวใดก็ได้

โปรแกรมจะไม่บอกว่าเราพิมพ์อะไรลงไปในช่วงโดยตรง เพื่อป้องกันไม่ให้คนอื่นมาเห็นได้เมื่อใส่เสร็จแล้ว ถ้าต้องการทราบว่าเป็น key อะไร ให้กดปุ่ม View จะมีหน้าต่างให้กรอกรหัสผ่านของโปรแกรม เมื่อกรอกแล้วโปรแกรมก็จะแสดง key นั้นๆออกมา ดังรูปที่ 4.16



รูปที่ 4.16 แสดง Key จากปุ่ม View

4.1.2.8 Use Schedule



รูปที่ 4.17 Use Schedule

ในฟังก์ชันนี้เมื่อผู้ใช้เลือกใช้งานแล้ว จะเป็นการกำหนด Task ให้โปรแกรม หรือกำหนดเวลาว่า เวลาไหน วันที่เท่าไรจะสั่งให้โปรแกรมล๊อคหน้าจอ โดยต้องเลือกอันใดอันหนึ่งเท่านั้น โดยมีให้เลือกทั้งหมดสามแบบ

- 1) Once at date คือการกำหนดไปเลยว่าจะสั่งให้โปรแกรมล๊อคหน้าจอวันที่เท่าไร โดยเวลาจะขึ้นกับ Start time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

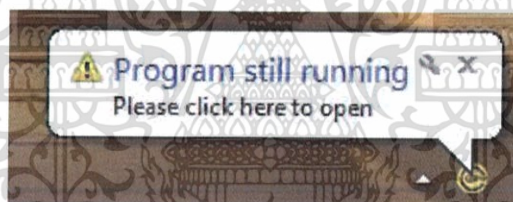
- 2) Daily คือการกำหนดว่าให้ลือคหน้าจอรทุกวัน โดยเวลาจะขึ้นกับ Start time
- 3) Weekly คือการกำหนดว่าใน 1 สัปดาห์จะให้ลือคหน้าจอรวันไหนบ้าง โดยสามารถเลือกได้หลายวัน เวลาจะขึ้นกับ Start time

4.1.2.9 Restore default

เมื่อกดปุ่ม Restore default จะเป็นการ Reset การตั้งค่าทั้งหมด โดยค่าเริ่มต้นของโปรแกรมแสดงดังรูปที่ 4.6 และ 4.7

4.1.3 การใช้งานปุ่ม Exit

ปุ่ม Exit นี้จะเป็นการออกจากโปรแกรม โดยโปรแกรมจะหยุดการทำงานทุกอย่าง ซึ่งต่างจากการกดปุ่ม x ที่มุมขวาบนของหน้าต่างหลัก ซึ่งการกด x นั้น โปรแกรมจะยังคงทำงานอยู่ในสถานะ Background Process ซึ่งจะมี Balloon Tips ขึ้นมาบอกดังรูปที่ 4.18



รูปที่ 4.18 Program still running

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุปของโครงการ

โปรแกรมป้องกันผู้ละเมิดใช้งานเครื่องคอมพิวเตอร์ เป็นโปรแกรมที่พัฒนามาจากภาษา Visual Basic .NET Framework 3.5 ซึ่งออกแบบมาเพื่อเพิ่มประสิทธิภาพในการจับผู้ร้ายที่จะมาขโมยข้อมูลในเครื่องคอมพิวเตอร์ หรือแอบใช้เครื่องคอมพิวเตอร์ในทางที่ผิด ออกแบบมาซึ่งสามารถที่จะถ่ายภาพของคนร้ายและส่งไปยังอีเมลล์ของเจ้าของเครื่องคอมพิวเตอร์ด้วย

ในโครงการนี้ได้ศึกษาเกี่ยวกับการทำงานของระบบปฏิบัติการ Windows เบื้องต้น ว่ามีการเรียกใช้งานโปรแกรมอัตโนมัติอย่างไร และมีวิธีการป้องกันการพิมพ์คีย์บอร์ดอย่างไร เพื่อให้เข้าใจและนำมากำหนดขอบเขตของโครงการ การทดลองนี้เป็นแนวทางในการพัฒนาโปรแกรมให้มีประสิทธิภาพสูงขึ้นไป และสามารถนำไปใช้ให้เกิดประโยชน์ได้จริง

5.2 ปัญหา อุปสรรค และแนวทางการแก้ไข

- 1) เนื่องจากเป็นโปรแกรมที่ต้องติดต่อผ่านฮาร์ดแวร์ ทำให้มีความยากลำบากในการพัฒนาในบางส่วน ผู้จัดทำเลยต้องนำ Open Source ที่เปิดให้ศึกษาฟรี มาช่วยเสริมให้ใช้งานได้ง่ายขึ้น
- 2) ระบบปฏิบัติการ Windows ในแต่ละรุ่นนั้น มีความแตกต่างกันบางส่วน ทำให้การพัฒนาโปรแกรมเพียงเวอร์ชันเดียวไม่เพียงพอ และไม่สามารถใช้งานได้เต็มที่ประสิทธิภาพ จึงต้องทำการพัฒนาโปรแกรม ให้รองรับกับ Windows ทุกรุ่น

5.3 แนวทางในการพัฒนาต่อ

- 1) พัฒนาให้มีประสิทธิภาพมากยิ่งขึ้น เพิ่มในส่วนของ Mobile Application ให้มีระบบแจ้งเตือนและสามารถเปิดดูได้จากในมือถือ
- 2) พัฒนาส่วนติดต่อผู้ใช้ให้มีความหลากหลาย สวยงาม และใช้งานได้ง่ายขึ้นกว่าเดิม
- 3) เพิ่มประสิทธิภาพในการถ่ายภาพคนร้ายให้ทำงานได้อย่างรวดเร็วยิ่งขึ้นกว่าเดิม
- 4) สามารถถ่ายภาพจากกล้อง (Webcam) ได้พร้อมกันหลายๆตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] นัฐพงศ์ สงเนียม “การใช้งานโปรแกรม Visual Studio 2010 เบื้องต้น” [Online]. Available : <http://www.siam2dev.com> 2011.
- [2] thaischool. “พื้นฐานการเขียนโปรแกรมด้วยภาษา Visual Basic 2010” [Online]. Available : http://www.thaischool1.in.th/site/download.php?doc_id=1264&school_id=40150001&title=40150001_0_20140728-094450.pdf&pathfile=/home/thschool1/domains/thaischool1.in.th/public_html/files_school/40150001/document/40150001_0_20140728-094450.pdf&url=N 2011.
- [3] mie “ประวัติ Visual Basic.NET อย่างย่อ” [Online]. Available : <http://greatfriends.biz/webboards/msg.asp?id=27407> 2006.
- [4] CodePlex. “การใช้งาน Touchless Library” [Online]. Available : <http://touchless.codeplex.com> 2008.