รายงานสหกิจศึกษาฉบับสมบูรณ์

การลงลายมือดิจิทัลในไฟล์โค้ดและเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

Code Signing and Release Approver Information Web Portal

นายชัชวาลย์ บำรุงไทยวรกุล

ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

รายงานสหกิจศึกษาฉบับสมบูรณ์

การลงลายมือดิจิทัลในไฟล์โค้ดและเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ
Code Signing and Release Approver Information Web Portal

นายชัชวาลย์ บำรุงไทยวรกุล

ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

ชื่อโครงงานสหกิจศึกษา การลงลายมือดิจิทัลในไฟล์โค้ดและเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ
ชื่อ-สกุล นักศึกษา นายชัชวาลย์ บำรุงไทยวรกุล
คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมคอมพิวเตอร์ **สาขาวิชา** วิศวกรรมสารสนเทศ
ชื่อ-สกุล อาจารย์นิเทศ ผศ.ดร.สุธีรา พันธุ์ธีรานุรักษ์
ชื่อ-สกุล ผู้นิเทศงาน คุณเกรียงไกร ตรัยไชยาพร
สถานประกอบการ บริษัท เอ็กซอนโมบิล จำกัด

บทคัดย่อ

ในปัจจุบันนั้นได้มีการโจรกรรมข้อมูลทางโลกไซเบอร์มากมาย ซึ่งอาจจะก่อให้เกิดความเสียหายต่อระบบและองค์กรทั้งในด้านค่าใช้จ่ายที่สูงขึ้นและชื่อเสียงองค์กรดังนั้นปริญญานิพนธ์ฉบับนี้จึงได้พัฒนาระบบเพิ่มความปลอดภัยให้กับข้อมูลขององค์กรโดยให้ชื่อว่า การลงลายมือดิจิทัลลงในไฟล์โค้ด (Code Signing) ซึ่งวิธีการที่นำมาประยุกต์ใช้สามารถทำงานร่วมกับแอปพลิเคชันขององค์กรที่มีอยู่แล้วได้ โดยวิธีการก็คือ เพื่อยืนยันว่าโค้ดชุดนั้นได้รับการรองรับจากผู้พัฒนาและไม่ถูกปลอมแปลงโดยมิจฉาชีพ อีกทั้งยังสามารถตรวจสอบ ได้ว่าบุคคลเป็นผู้พัฒนาขึ้นมาหรือไฟล์โค้ดได้รับการเปลี่ยนแปลงแก้ไขหรือไม่ ซึ่งวิธีดังกล่าวนำมาซึ่งความปลอดภัยที่มากขึ้นให้กับองค์กร การนำโค้ดขึ้นสู่ระบบ หากไม่มีผู้อนุมัติมาตรวจสอบอาจจะทำให้ระบบเกิดความเสียหายได้ เพื่อยืนยันว่าโค้ดที่นำขึ้นสู่ระบบ ควรตรวจสอบได้ว่าบุคคลเป็นผู้อนุมัติ แต่แอปพลิเคชันที่ใช้อยู่นั้น ยังตรวจสอบผู้อนุมัติได้ยาก จึงได้จัดทำเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นสู่ระบบ ทำให้สามารถตรวจสอบรายชื่อผู้อนุมัติได้ง่ายและรวดเร็ว ส่งผลให้ระบบขององค์กรทำงานได้อย่างมีประสิทธิผลและเสถียรภาพมากขึ้น

คำสำคัญ: การลงลายมือดิจิทัลลงในไฟล์โค้ด ดิจิทัลซิกเนเจอร์ รีรีสเดฟพินิชัน การจัดการรีรีส ใบรับรอง กุญแจส่วนตัว ไฟล์พีเอฟเอกซ์

Co-operative Title: Code Signing and Release Approver Information Web Portal
Student Intern Name: Mr.Chatchawarn Bumrungtaivorakul
Faculty: Engineering **Department:** Computer Engineering **Program:** Information Engineering
Advisor Name: Asst.Prof.De.Sutheera Puntheeranurak
Mentor Name: Kriangkrai Traichaiyaporn
Company: ExxonMobil Limited

ABSTRACT

In recent years, cyber attacks become more frequent which lead to severe financial and operational impacts to societies and organizations. This thesis aims at implementing "Code Signing" that is the digitally signing process to prevent a tampering attack. The signed code or executables are guaranteed that it has not been altered or corrupted. Moreover, code signing detects the software owner and ensure the right copy of the application. Hence, the code signing enhances security and effectiveness in organizations by mitigating risk from cyber attacks and ensuring the software copyright.

Furthermore, if code deployment does not have any approvers, the code may cause the problem to a system. However, it is hard to check who is the checker in the existing application. So, we created "Release Approver Information Portal" to show who is the approver. The web portal could increase speed and efficiency for validation checking process which benefits the organization.

Keywords: Code Signing; Digital Signature; Release Definition; Release Management; Certificate; Private Key; PFX file

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี ด้วยความช่วยเหลือของคุณเกรียงไกร ตรัยไชยาพร และ คุณประภาส ติวารี พนักงานที่ปรึกษาสหกิจศึกษา ซึ่งได้ให้คำแนะนำ สอนความรู้ และให้ความเห็นต่าง ๆ อันเป็นประโยชน์อย่างยิ่งในการทำงาน อีกทั้งยังช่วยแก้ปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างดำเนินงานวิจัยอีกด้วย ขอขอบคุณ คุณทิฆัมพร อุ่นเธียรนันท์ ที่ให้คำปรึกษาด้านการทำงาน ผู้อนุมัติขั้นตอนต่าง ๆ ในการดำเนินการวิจัย ขอขอบคุณผศ.ดร.สุธีรา พันธุ์ธีรานุรักษ์ อาจารย์นิเทศศึกษา ที่คอยให้คำแนะนำเกี่ยวกับการฝึกงาน การทำสหกิจศึกษากับทางบริษัท ตลอดจนถึงการปรับตัวให้เหมาะสมกับที่ทำงาน ขอขอบคุณเพื่อน ๆ ร่วมงานทุกคนที่ช่วยเสนอแนะแนวทางการแก้ไขปัญหา เป็นกำลังใจในการทำปริญญานิพนธ์เรื่องนี้

สุดท้ายนี้ผู้วิจัยขอขอบคุณบิดามารดา และครอบครัว ซึ่งเปิดโอกาสให้ได้รับการศึกษาเล่าเรียน ตลอดจนคอยช่วยเหลือและให้กำลังใจผู้ทำวิจัยเสมอมาจนสำเร็จการศึกษา

ชัชวาลย์ บำรุงไทยวรกุล

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

ตารางที่ **หน้า**

สารบัญภาพ

สารบัญภาพ (ต่อ)

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

บริษัท เอ็กซอนโมบิล จำกัด (ExxonMobil Limited) เป็นองค์กรขนาดใหญ่ด้านพลังงานที่มีหลายธุรกิจอาทิเช่น ธุรกิจก๊าซธรรมชาติ น้ำมันปิโตรเลียม เคมีภัณฑ์ ต่าง ๆ อีกทั้งยังเป็นองค์กรมหาชนที่ใหญ่ที่สุดในโลกอีกด้วย องค์กรมีการทำงานแบบครบวงจรโดยเริ่มตั้งแต่ขุดเจาะน้ำมัน นำน้ำมันที่ขุดเจาะได้มาผ่านกระบวนกลั่นลำดับส่วนเพื่อให้ได้น้ำมันตามจุดประสงค์ของลูกค้า ก่อนจะส่งต่อเพื่อจำหน่ายให้ลูกค้าตามปั๊มน้ำมันต่าง ๆ ปั๊มน้ำมันของบริษัท เอ็กซอนโมบิล จำกัด ในแต่ละสาขาอยู่ภายใต้เครื่องหมายการค้า “เอสโซ่” ในประเทศไทยและภายใต้เครื่องหมายการค้าอื่น ๆ ในอีกหลายประเทศ ด้วยเหตุนี้ จึงทำให้บริษัท เอ็กซอนโมบิล จำกัด มีจำนวนพนักงานหรือบุคลากรต่าง ๆ เป็นจำนวนมากมหาศาลที่ต้องถูกจัดเก็บไว้ในระบบคอมพิวเตอร์ขององค์กร รวมไปถึงรายละเอียดของบุคคลภายนอกที่มีความเกี่ยวของกับคนในบริษัททั้งโดยตรงและโดยอ้อมอีกด้วย นอกจากนี้ยังมีข้อมูลทางธุรกิจอีกมากมาย อาทิเช่น ข้อมูลทางการเงิน สินค้า ตลาดหุ้น และอื่น ๆ อีกมากมาย ซึ่งเป็นข้อมูลที่มีความสำคัญต่อการดำเนินงานในทุกแผนกขององค์กรที่ถูกจัดเก็บไว้ในเช่นกัน การรักษาความปลอดภัยของข้อมูลจึงเป็นสิ่งสำคัญมาก ถ้าหากข้อมูลไม่มีความปลอดภัยนั้น อาจจะทำให้เกิดความเสียหายต่อองค์กรได้ ดังนั้นความปลอดภัยของข้อมูลจึงเป็นสิ่งที่สำคัญอันดับต้น ๆ เช่นกัน

การเข้าร่วมโครงการสหกิจศึกษากับทางบริษัท เอ็กซอนโมบิล จำกัดในแผนกอีมิท (EMIT หรือ ExxonMobil Information Technology) ในทีมเอแอลเอ็ม หรือแอปพลิเคชัน ไลฟไซเคิล แมเนจเมนต์ (ALM หรือ Application Lifecycle Management) ซึ่งเป็นแผนกย่อยในแผนกอีเอเอส หรือ อีมิทไอทีแอปพลิเคชันซัปพอร์ต (EAS หรือ EMIT IT Application Support) มีหน้าที่หลักในการจัดการขั้นตอนการทำซอฟต์แวร์แอปพลิเคชันจากเริ่มต้นการวางแผนจนไปถึงการนำไปใช้จริง ประกอบไปด้วยสี่ขั้นตอนหลักภายในทีม ขั้นตอนแรกคือการวางแผน (Plan) คือการเก็บรวบรวมข้อมูลความต้องการของผู้ใช้งานต่าง ๆ โดยใช้กระดานกันบัน (Kanban Board) ในโปรแกรมทีเอฟเอส (TFS หรือ Team Foundation Server) ขั้นตอนที่สองคือการพัฒนาและการทดสอบ (Develop and Test) คือการแก้ไขแอปพลิเคชันเป็นเวอร์ชันต่าง ๆ การคอมไพล์แอปพลิเคชัน และหาข้อผิดพลาดก่อนจะนำแอปพลิเคชันไปใช้จริง ขั้นตอนที่สามคือการทดสอบในสภาพแวดล้อม (Release) คือการนำเอาแอปพลิเคชันที่สร้างขึ้นมาแล้ว นำไปทดสอบในสภาพแวดล้อมต่าง ๆ ที่ไม่เหมือนกันเพื่อดูการทำงาน ขั้นตอนที่สี่คือมอนิเตอร์และเรียนรู้ (Monitor and Learn) คือการบันทึกผลจากการทดสอบในแต่ละสภาพแวดล้อมต่าง ๆ เพื่อนำไปปรับปรุงพัฒนาให้ดีขึ้น แอปพลิเคชันหลักของทีมที่ดูแลคือโปรแกรมทีเอฟเอสเป็นโปรแกรมที่รวมขั้นตอนต่าง ๆ ทั้งหมดของทีม เพื่อให้ผู้ใช้งานที่เกี่ยวข้องเข้ามาใช้ได้ ซึ่งปัญหาแรกที่เจอคือไฟล์โค้ดที่ผู้ใช้งานสร้างนั้น ไม่มีความปลอดภัยอาจจะถูกโจรกรรม แก้ไข เปลี่ยนแปลง อาจจะมีมัลแวร์ที่สามารถ

สร้างอันตรายให้กับระบบหรือองค์กรได้ และอีกหนึ่งปัญหาคือการนำโค้ดขึ้นระบบ ควรจะต้องมีผู้อนุมัติก่อนในทุกครั้ง เพื่อป้องกันการเสียหายของระบบหลังจากที่ไฟล์โค้ดทำงานแล้ว

จากปัญหาที่กล่าวมาจากข้างต้นนำมาซึ่งการพัฒนาระบบด้านความปลอดภัย จะอยู่ในขั้นตอนที่สองของทีมคือการพัฒนาและการทดสอบ ให้มีความปลอดภัยมากยิ่งขึ้นได้โดยการเพิ่มฟังก์ชันความปลอดภัย ป้องกันการโจรกรรมของไฟล์โค้ดที่ ไม่ให้ถูกเปลี่ยนแปลงหรือแก้ไขโดยผู้ไม่หวังดี จึงทำให้ระบบมีประสิทธิภาพและเสถียรภาพมากขึ้น ซึ่งวิธีดังต่อไปนี้คือเรียกว่าการลงลายมือดิจิทัลในไฟล์โค้ด (Code Signing) และจากปัญหาที่สอง สามารถแก้ได้โดยการสร้างหน้าเว็บขึ้นแสดงรายชื่อของผู้อนุมัติการนำโค้ดขึ้นระบบ เพื่อง่ายต่อการตรวจสอบและตั้งค่าผู้อนุมัติได้อย่างรวดเร็ว

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพิ่มความปลอดภัยให้กับไฟล์โค้ด

1.2.2 ลดภาระงานของพนักงานที่เกี่ยวข้อง

1.2.3 ไฟล์โค้ดมีความถูกต้องมากยิ่งขึ้น

1.2.4 ไฟล์โค้ดมีมาตรฐานเดียวกันและน่าเชื่อถือยิ่งขึ้น

1.2.5 ลดอัตราการเสียหายของระบบและเซิร์ฟเวอร์

1.2.6 เพิ่งฟังก์ชันเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

1.3 ขอบเขตของการวิจัย

ขอบเขตงานตลอดระยะเวลาเข้าร่วมโครงการสหกิจศึกษาที่ได้รับมอบหมายให้รับผิดชอบจากบริษัทแบ่งออกเป็น 2 ส่วนดังนี้

1.3.1 การลงลายมือดิจิทัลในไฟล์โค้ด

1.3.2 เว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

1.4 วิธีการดำเนินวิจัย

การวางแผนการทำงานในขั้นตอนต่าง ๆ ในการดำเนินงานวิจัยสามารถจำแนกออกได้ตามแต่ละส่วนของขอบเขตงานวิจัยดังกล่าวไว้ข้างต้นได้ดังนี้

1.4.1 การลงลายมือดิจิทัลในไฟล์โค้ด

ขอบเขตงานวิจัยในส่วนนี้เป็นส่วนที่ต้องศึกษาค้นคว้าหาข้อมูลเกี่ยวกับทฤษฎีความปลอดภัยทางไซเบอร์ การเข้ารหัสเพื่อเพิ่มระดับความปลอดภัยของไฟล์โค้ดและทำการทดลองเพื่อเลือกใช้เครื่องมือที่เหมาะสมที่สุด วิธีการดำเนินการวิจัยในขอบเขตงานวิจัยนี้สามารถจัดลำดับได้ ดังนี้

1.4.1.1 ศึกษาทฤษฎีการเข้ารหัสถอดรหัสข้อมูลดิจิทัล และทดลองการลงลายมือดิจิทัล

1.4.1.2 ทำคู่มือการลงลายมือดิจิทัลให้กับไฟล์โค้ด

1.4.2 เว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

เว็บแสดงรายชื่อของผู้อนุมัติในส่วนของการจัดการรีรีส (Release Management) ในโปรแกรมทีเอฟเอส โดยตรวจสอบว่ามีรายชื่อผู้อนุมัติอยู่ในรีรีสเดฟฟินิชัน (Release Definition) ใดบ้าง ให้แสดงผลออกมาเป็นตาราง โดยวิธีการดำเนินการวิจัยในขอบเขตงานวิจัยนี้สามารถจัดลำดับได้ ดังนี้

1.4.2.1 ศึกษาการใช้ภาษาฟรอนต์เอนด์โปรแกรมมิง

1.4.2.2 สร้างหน้าเว็บแสดงแสดงรายชื่อผู้อนุมัติ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่ได้รับในการพัฒนาระบบที่ได้รับมอบหมายในการเข้าร่วมโครงการสหกิจศึกษากับทางบริษัทเอ็กซอนโมบิล จำกัด สามารถจำแนกได้ออกเป็น 3 ส่วน ดังนี้

1.5.1 ประโยชน์ต่อบริษัท

1.5.1.1 ระบบมีประสิทธิภาพและเสถียรภาพมากขึ้น

1.5.1.2 ข้อมูลมีความปลอดภัยและถูกต้องมากขึ้น

1.5.1.3 ลดรายจ่ายส่วนหนึ่งของบริษัทได้

1.5.2 ประโยชน์ต่อพนักงานที่เกี่ยวข้อง

1.5.2.1 ลดขั้นตอนการทำงานของพนักงานที่เกี่ยวข้อง

1.5.2.2 ลดโอกาสการเข้าไปแก้ไขในไฟล์โค้ดและในระบบ

1.5.2.3 ศึกษาหาความรู้ด้านใหม่เพื่อคนในทีม

1.5.3 ประโยชน์ต่อผู้วิจัย

1.5.3.1 ได้รับความรู้ความเข้าใจหลักการทำงานของโปรแกรมทีเอฟเอส โปรแกรมเมคเสิท (Makecert) โปรแกรมพีวีเคทูพีเอฟเอ็ก (Pvk2pfx) โปรแกรมไซน์ทูล (Signtool)

1.5.3.2 เรียนรู้การเขียนคำสั่งผ่านคอมมานไลน์ (Command Line) โดยใช้ภาษาพาวเวอร์เชลล์ (PowerShell)

1.5.3.3 เรียนรู้ภาษาเกี่ยวกับการออกแบบเว็บไซต์ เช่น ภาษาเอชทีเอ็มแอล (HTML) ภาษาจาวาคริป (JavaScript) ภาษาซีเอสเอส (CSS) เจคิวรี่ (JQuery) เจสัน (JSON) เป็นต้น

1.5.3.4 มีความเข้าใจในการทำงานจริงและสามารถปรับตัวได้ในสภาวะแวดล้อมใหม่ ๆ ในชีวิตจริงได้ดีขึ้น

1.5.3.5 สามารถนำความรู้ที่ได้รับไปเป็นแนวทางในการเตรียมความพร้อมในด้านต่าง ๆ ในอนาคต

บทที่ 2
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

งานวิจัยชิ้นนี้เป็นการศึกษา พัฒนาระบบความปลอดภัยของไฟล์โค้ดและเว็บแสดงรายชื่อผู้อนุมัติก่อนนำโค้ดขึ้นระบบ ผู้วิจัยจึงต้องทำการศึกษาค้นคว้าทฤษฎีทางคอมพิวเตอร์ที่เกี่ยวข้อง ไม่ว่าจะเป็นภาษาคอมพิวเตอร์ (Computer Language) ที่ใช้ในการควบคุมการทำงานหรือซอฟต์แวร์ (Software) ทฤษฎีทางการเข้ารหัสต่าง ๆ ที่จะถูกนำมาใช้ในงานวิจัยนี้แล้วนำทฤษฎีเหล่านั้นมาประยุกต์ใช้ในการทำงานที่ได้รับมอบหมายให้สำเร็จสมบูรณ์ โดยแนวคิดทฤษฎีและงานวิจัยที่เกี่ยวข้องที่นำมาศึกษาในการทำวิจัยมีรายละเอียดดังต่อไปนี้

2.1 ทฤษฎีที่เกี่ยวข้องด้านภาษาคอมพิวเตอร์

2.1.1 ภาษาพาวเวอร์เชลล์ (PowerShell) [1]

ภาษาพาวเวอร์เชลล์ เป็นภาษาที่มีพื้นฐานการพัฒนามาจาก ดอตเน็ตเฟรมเวิร์ก (.Net Framework) และผ่านการออกแบบมาเพื่อให้ผู้ดูแลระบบสามารถนำไปใช้งานในการเขียนโปรแกรมในรูปแบบของชุดคำสั่งหรือสคริป (Script) ต่าง ๆ ได้ เพื่อบริหารจัดการระบบต่าง ๆ ของวินโดว์เซิร์ฟเวอร์ (Window Server) ได้อย่างมีประสิทธิภาพ เนื่องจากมีคำสั่งจำนวนมาก และสามารถเรียกใช้งานได้ง่าย

รูปแบบของคำสั่งในวินโดว์ พาวเวอร์เชลล์ มีลักษณะการทำงานที่อ้างอิงมาจาก ดอตเน็ตเฟรมเวิร์ก และมีรูปแบบการทำงานแบบอ็อบเจกต์ (Object) อย่างสมบูรณ์ โดยแต่ละคำสั่งของพาวเวอร์เชลล์ นั้นจะเรียกว่า “คอม-มาน-เล็ท” (cmdlet) โดยแต่ละคำสั่งนั้นจะมีองค์ประกอบ 2 ส่วนด้วยกัน ได้แก่

1. คำกริยา (Verb) เป็นส่วนที่บ่งบอกกระทำว่าต้องการดำเนินการอะไรกับอ็อบเจกต์ที่ต้องการ เช่นตารางที่ 2.1

ตารางที่ 2.1 ตารางแสดงตัวอย่างคำสั่งในภาษาพาวเวอร์เชลล์ในส่วนคำกริยา

ลำดับ	คำสั่ง	ความหมาย
1.	New	การสร้างอ็อบเจกต์ใหม่
2.	Get	การดึงค่าข้อมูลต่าง ๆ ของอ็อบเจกต์
3.	Set	การกำหนดค่าของข้อมูลต่าง ๆ ของอ็อบเจกต์
4.	Remove	การลบอ็อบเจกต์ออก

2. คำนาม (Noun) เป็นการกำหนดชนิดของอ็อบเจกต์ที่ต้องการดำเนินการด้วย เช่นตารางที่ 2.2

ตารางที่ 2.2 ตารางแสดงตัวอย่างคำสั่งในภาษาพาวเวอร์เชลล์ในส่วนคำนาม

ลำดับ	คำสั่ง	ความหมาย
1.	ADUser	ต้องการทำงานกับผู้ใช้งานอ็อบเจกต์ในระบบแอ็กทิฟไดเรกทอรีโดเมนเซอร์วิส (Active Directory Domain Service)
2.	Process	ต้องการทำงานกับโพรเซส (Process) ต่าง ๆ ที่ทำงานอยู่ในเครื่องคอมพิวเตอร์ที่ระบุ

ดังนั้นการอธิบายความหมายของคำสั่งจึงสามารถเข้าใจได้โดยง่ายจากการดูจากคำกริยาและคำนามตามที่ระบุ ดังตารางที่ 2.3 ที่เป็นคำสั่งพาวเวอร์เชลล์โดยสมบูรณ์

ตารางที่ 2.3 ตารางแสดงตัวอย่างคำสั่งในภาษาพาวเวอร์เชลล์

ลำดับ	คำสั่ง	ความหมาย
1.	New-ADUser	การสร้างผู้ใช้งานอ็อบเจกต์ในระบบแอ็กทิฟไดเรกทอรีโดเมนเซอร์วิส
2.	Get-ADUser	การดึงค่าลักษณะต่าง ๆ ของผู้ใช้งานอ็อบเจกต์ในระบบแอ็กทิฟไดเรกทอรีโดเมนเซอร์วิส

2.1.2 ภาษาเอชทีเอ็มแอล (HTML) [2]

ภาษาเอชทีเอ็มแอล (HTML หรือ Hyper Text Markup Language) เป็นภาษาประเภทมาร์กอัป (Markup Language) ที่ใช้ในการสร้างเว็บเพจ (Web Page) มีต้นแบบมาจากภาษาเอสจีเอ็มแอล (SGML หรือ Standard Generalized Markup Language) ที่ลดความสามารถบางส่วนออก เพื่อให้สามารถทำความเข้าใจและเรียนรู้ได้ง่าย ปัจจุบันมีการพัฒนาและกำหนดมาตรฐานโดยองค์กรดับเบิลยูทรีซี (W3C หรือ World Wide Web Consortium)

ภาษาเอชทีเอ็มแอล ได้ถูกพัฒนาขึ้นอย่างต่อเนื่องตั้งแต่เอชทีเอ็มแอลระดับ 1, เอชทีเอ็มแอล 2.0, เอชทีเอ็มแอล 3.0 ,เอชทีเอ็มแอล 3.2 และ เอชทีเอ็มแอล 4.0 ในปัจจุบันองค์กรดับเบิลยูทรีซี ได้พัฒนารูปแบบของภาษาเอชทีเอ็มแอลแบบใหม่ ที่เรียกว่าเอกซ์เอชทีเอ็มแอล (XHTML หรือ Extensible Hypertext Markup Language) ซึ่งเป็นลักษณะของโครงสร้างเอกซ์เอ็มแอล (XML) แบบหนึ่ง ที่มีหลักในการกำหนดโครงสร้างของโปรแกรมที่มีรูปแบบและมาตรฐานมากกว่า มาทดแทนใช้เอชทีเอ็มแอล รุ่น 4.01 ที่ใช้กันอยู่ในปัจจุบัน

เอชทีเอ็มแอล มีโครงสร้างการเขียนโดยอาศัยแท็ก (Tag) ในการควบคุมการแสดงผลของข้อความ รูปภาพ หรือวัตถุอื่น ๆ แต่ละแท็กอาจจะมีส่วนขยาย เรียกว่า แอตทริบิวต์ (Attribute) สำหรับจัดรูปแบบเพิ่มเติม

การสร้างเว็บเพจ โดยใช้ภาษาเอชทีเอ็มแอล สามารถทำโดยใช้โปรแกรมเท็กซ์เอดิเตอร์ (Text Editor) ต่าง ๆ เช่น โน้ตแพด (Notepad), เอดิตพลัส (EditPlus) หรือจะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น ไมโครซอฟท์ ฟรอนต์เพจ (Microsoft FrontPage), ดรีมวีเวอร์ (Dream Weaver) ซึ่งอำนวยความสะดวกในการสร้างหน้าเอชทีเอ็มแอล ในลักษณะดับเบิลยูวายเอสไอดับเบิลยูวายจี (WYSIWYG หรือ What You See Is What You Get)

การเรียกใช้งานหรือทดสอบการทำงานของเอกสารเอชทีเอ็มแอล จะใช้โปรแกรม อินเทอร์เน็ตเว็บเบราว์เซอร์ (Internet Web Browser) เช่น อินเทอร์เน็ต เอกซ์พลอเรอร์ (IE หรือ Internet Explorer), โมซิลลาไฟร์ฟอกซ์ (Mozilla Firefox), ซาฟารี (Safari) และกูเกิลโครม (Google Chrome) เป็นต้นโครงสร้างหลักของเอชทีเอ็มแอล จะเริ่มด้วยคำสั่งเอชทีเอ็มแอล โดยคำสั่งจะอยู่ภายในเครื่องหมายวงเล็บสามเหลี่ยมเสมอ ซึ่งชุดโครงสร้างหลักที่ใช้จะแยกเป็น 2 ส่วนคือ

1. เฮด (Head) คำสั่งที่อยู่ในส่วนนี้จะใช้บรรยายรายละเอียดเกี่ยวกับเว็บเพจ ซึ่งจะไม่แสดงผลที่เว็บเพจ โดยส่วนของเฮดเป็นส่วนที่ใช้อธิบายเกี่ยวกับข้อมูลเฉพาะของหน้าเว็บนั้น ๆ เช่น ชื่อเรื่องของหน้าเว็บ (Title), ชื่อผู้จัดทำเว็บ (Author), คีย์เวิร์ดสำหรับการค้นหา (Keyword) เป็นต้น

2. บอดี (Body) คำสั่งที่อยู่ในส่วนนี้จะใช้ในการจัดรูปแบบตัวอักษร จัดหน้า ใส่รูปภาพ ซึ่งตัวอักษรในส่วนนี้จะแสดงที่เว็บเบราว์เซอร์โดยตรง โดยส่วนของบอดี จะเป็นส่วนเนื้อหาหลักของหน้าเว็บ ซึ่งการแสดงผลจะต้องใช้แท็กจำนวนมาก ขึ้นอยู่กับลักษณะของข้อมูล เช่น ข้อความ, รูปภาพ, เสียง, วีดิโอ หรือไฟล์ต่าง ๆ

2.1.3 ภาษาจาวาสคริปต์ (JavaScript) [3]

จาวาสคริปต์ คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง จาวาสคริปต์เป็นภาษาสคริปต์เชิงวัตถุ ที่เรียกกันว่า "สคริปต์" ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช่ร่วมกับเอชทีเอ็มแอล) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปทีละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจ็กโอเรียล

เต็ด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษาเอชทีเอ็มแอล สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษาเอชทีเอ็มแอล และภาษาจาวา ได้ทั้งทางฝั่งไคลเอนต์ (Client) และทางฝั่งเซิร์ฟเวอร์

จาวาสคริปต์ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่าไลฟ์สคริป (Live Script) ออกมาพร้อมกับเน็ตสเคป เนวิเกเตอร์ 2.0 (Netscape Navigator2.0) เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบไลฟ์ไวร์ (Live Wire) ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อใช้งานกับภาษาจาวาได้ และได้ปรับปรุงไลฟ์สคริปต์ ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า จาวาสคริปต์สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่นต่าง ๆ มากมาย และยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจากจาวาสคริปต์ ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่บุคคลก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดยอีซีเอ็มเอ (ECMA การทำงานของ จาวาสคริปต์ จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ (ฝั่งผู้ใช้งาน) ดังนั้น จาวาสคริปต์จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุนจาวาสคริปต์แล้ว อย่างไรก็ดี สิ่งที่ต้องระวังคือ จาวาสคริปต์ มีการพัฒนาเป็นเวอร์ชันใหม่ ๆ ออกมาด้วย (ปัจจุบันคือรุ่น 1.5) ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิดได้ปัญหาได้

จาวาสคริปต์ไม่ใช่ภาษาจาวา (Java) แต่อย่างใด จาวาเป็นภาษาที่ถูกพัฒนาโดยซันไมโครซิสเต็ม (Sun Microsystems) เป็นภาษาประเภทโปรแกรมมิง (Programming) สำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ คล้ายกับภาษาซีหรือซีพลัสพลัส (C, C++)

2.1.4 ภาษาซีเอสเอส (CSS) [4]

ซีเอสเอส (CSS หรือ Cascading Style Sheet) มักเรียกโดยย่อว่า "สไตล์ชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสารเอชทีเอ็มแอลโดยที่ซีเอสเอสกำหนดกฎเกณฑ์ในการระบุรูปแบบ (Style) ของเนื้อหาในเอกสาร ได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ นี้ใช้หลักการของการแยกเนื้อหาเอกสารเอชทีเอ็มแอลออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลัพธ์ของเอกสารเอชทีเอ็มแอลโดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสารเอชทีเอ็มแอลมีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ เอกสารเอชทีเอ็มแอลถูกเพิ่มเข้ามาครั้งแรกในเอชทีเอ็มแอล เวอร์ชัน 4.0 เมื่อปีพ.ศ. 2539 ที่กำหนดโดย องค์กรดับเบิลยูทรีซี

ซีเอสเอส กับ เอชทีเอ็มแอลและเอกซ์เอชทีเอ็มแอล นั้นทำหน้าที่คนละอย่างกัน โดย เอชทีเอ็มแอล และเอกซ์เอชทีเอ็มแอล จะทำหน้าที่ในการวางโครงร่างเอกสารอย่างเป็นรูปแบบ ถูกต้อง เข้าใจง่าย ไม่เกี่ยวข้องกับการแสดงผล ส่วนซีเอสเอสจะทำหน้าที่ในการตกแต่งเอกสารให้สวยงาม เรียกได้ว่า เอชทีเอ็มแอลและเอกซ์เอชทีเอ็มแอล คือส่วนโค้ดดิง ส่วนซีเอสเอสคือส่วนการออกแบบ (Design)

2.1.5 ภาษาเจคิวรี (JQuery) [5]

เจคิวรีเป็นจาวาสคริปต์ไลบรารี (JavaScript Library) ที่มีการรวบรวมฟังก์ชันของจาวาสคริปต์ต่าง ๆ ให้อยู่ในรูปแบบแพตเทิร์นเฟรมเวิร์ก (Patterns Framework) ที่สะดวกและง่ายต่อการใช้งาน มีความยืดหยุ่นรองรับต่อการใช้งานข้ามเบราว์เซอร์คือไม่ว่าจะใช้งานบนเว็บเบราว์เซอร์ใด ในไลบรารีของเจคิวรี จะมีการเลือกใช้ฟังก์ชันที่เหมาะสมต่อการทำงานและแสดง ผลในเว็บเบราว์เซอร์ที่กำลังรันอยู่ ซึ่งช่วยลดปัญหาการทำงานที่ผิดพลาดในฝั่งของผู้ใช้งานได้จาก ปัญหาก่อนนี้ นักโปรแกรมเมอร์ทั้งหลายในสมัยก่อน ๆ มักจะทดสอบโปรแกรมและพัฒนาบนเบราว์เซอร์ไออี (เป็นเว็บเบราว์เซอร์ ที่คนใช้มากที่สุดเกือบ 95% เมื่อสมัย 5-6 ปี)

ปัจจุบันมีหลายเว็บเบราว์เซอร์ ได้เกิดขึ้นมากมาย เช่น โครม, ไฟร์ฟอกซ์ หรือ ซาฟารี และบางคำสั่งของจาวาสคริปต์จะไม่ทำงานหรือไม่รองรับในเว็บเบราว์เซอร์บางตัว ด้วยเหตุผลนี้เอง การใช้เจคิวรีมาเป็นทางเลือกก็สามารถช่วยแก้ปัญหานี้เป็นได้อย่างดี ทั้งยังสะดวกต่อการใช้งาน เพราะเป็นรูปแบบที่เข้าใจง่าย และเขียนได้ในรูปแบบที่สั้น ๆ รองรับการทำงานทั้งในเอชที เอ็มแอลรูปแบบเดิม หรือ ซีเอสเอส เอเลเมนต์ (element) ดีโอเอ็ม เอเลเมนต์ (DOM Element), ผลกระทบการจัดการอีเวนต์ (Event) ต่าง ๆ หรือแม้กระทั่งการพัฒนาเอแจ็ก (Ajax) ด้วยเจคิวรีก็สามารถ ทำได้อย่างง่ายดาย โดยรูปแบบเหล่านี้ยังคงทำงานอยู่ภายใต้คำสั่งของภาษาจาวาสคริปต์ แต่การเรียกใช้งานโครงร่างหรือฟังก์ชันต่าง ๆ จะถูกกำหนดรูปแบบโดยแพตเทิร์นที่ได้ถูกออกแบบไว้ในไลบรารีของเจคิวรี

2.1.6 ภาษาเจสัน (JSON) [6]

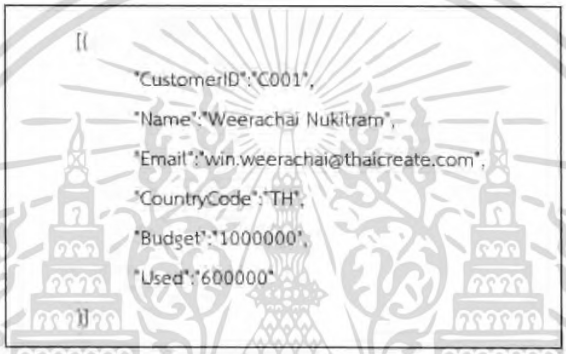
เจสัน หรือ จาวาสคริปต์ อ็อบเจกต์ โนเทชัน (JSON หรือ Java Script Object Notation) ซึ่งหากนิยามในภาษาไทยคือเครื่องหมายที่ใช้แทนวัตถุที่เป็นข้อมูลที่สามารถทำงานได้กับภาษาจาวาสคริปต์

ภาษาจาวาสคริปต์เป็นภาษาสำหรับการโปรแกรมบนเว็บเบราเซอร์ (Web Browser) ซึ่งเป็นภาษาประเภททำงานบนเครื่องลูกข่าย (Client-Side Programming) ซึ่งภาษาจาวาสคริปต์ ช่วยให้นักพัฒนาสามารถโปรแกรมจัดการข้อมูลบนหน้าเว็บไซต์ได้อย่างสะดวกและมีประสิทธิภาพ

เจสันเป็นโครงสร้างข้อมูลชนิดหนึ่ง ที่สามารถทำงานร่วมกับภาษาจาวาสคริปต์ได้อย่างดีและมีประสิทธิภาพ ซึ่งภาษาเจสันเป็นโครงสร้างสำหรับการจัดเก็บข้อมูล และใช้ในการแลกเปลี่ยนข้อมูลผ่าน เครือข่ายอินเทอร์เน็ตได้ อีกทั้งภาษาเจสันสามารถแปลงให้เป็นโครงสร้างของภาษาเอกซ์เอ็มแอล (eXtension Markup Language) ได้อย่างสะดวกรวดเร็ว

ในภาษาจาวาสคริปต์ ถูกออกแบบมาให้เป็นภาษาเชิงวัตถุ ซึ่งจะมองข้อมูลหรือฟังก์ชันการทำงานเปรียบเสมือนวัตถุซึ่งวัตถุหนึ่ง ๆ สามารถทำงานหรือสามารถปรับแต่งได้อย่างหลากหลาย เช่นเดียวกันกับภาษาที่เป็นรูปแบบโครงสร้างข้อมูลที่เข้ากับแนวคิดเชิงวัตถุ และรูปแบบโครงสร้างของภาษาเจสัน สามารถทำงานร่วมกับภาษาจาวาสคริปต์ ได้เป็นอย่างดี และนอกจากนี้ภาษาจาวาสคริปต์ มีตัวแปลงเจสัน หรือที่เรียกว่า พาร์เซอร์ (Parser) เพื่อใช้ในการแปลงโครงสร้างเจสันให้ทำงานกับจาวาสคริปต์หรือ แปลงให้เป็นโครงสร้างเอกซ์เอ็มแอล ก็สามารถทำได้เช่นกัน

ตัวอย่าง เจสันดังภาพที่ 2.1 เป็นข้อมูลที่อยู่ในรูปเจสันของพนักงานคนหนึ่ง ที่มีข้อมูลของเลขประจำตัว ชื่อ อีเมล ตัวย่อประเทศ งบประมาณ และเงินที่ใช้ไป



ภาพที่ 2.1 ตัวอย่างภาษาเจสัน

2.2 ทฤษฎีที่เกี่ยวข้องในด้านซอฟต์แวร์คอมพิวเตอร์

2.2.1 โปรแกรมทีเอฟเอสหรือทีมฟาวน์เดชันเซิร์ฟเวอร์ (TFS หรือ Team Foundation Server) [7]

ทีมฟาวน์เดชันเซิร์ฟเวอร์ หรือ ทีเอฟเอส เป็นผลิตภัณฑ์ของไมโครซอฟท์ที่สามารถบริการการเก็บโค้ด (Source code management) การจัดทำรายงาน การจัดการความต้องการ (Requirement) การจัดการโปรเจกต์ (Project Management) ทั้งในรูปแบบการพัฒนาซอฟต์แวร์แบบ อไจล์ (Agile) และวอเตอร์ฟอล (Waterfall) การสร้าง การทดสอบ การจัดการรีรีส (Release Management) ทีเอฟเอสยังสามารถดูแลทั้งวัฏจักรการสร้างพัฒนาแอปพลิเคชัน และยังสามารถเป็นหลังบ้าน (Back-End) ในการพัฒนาหลาย ๆ สภาพแวดล้อม เหมาะสำหรับวิชวล สตูดิโอและอีคลิปซ์ (Eclipse) ในทุก ๆ แพลทฟอร์ม

การสร้างซอฟต์แวร์ในปัจจุบันนี้จัดเป็นเรื่องที่มีความท้าทายอย่างมาก เนื่องจากแม้แต่โครงสร้างที่มีขนาดเล็กที่สุดก็ยังต้องการให้สมาชิกหลาย ๆ คนมาทำงานร่วมกัน อีกทั้งส่วนใหญ่แล้วสมาชิกที่อยู่ในทีมยังอาจไม่ได้อยู่ในสถานที่เดียวกันอีกด้วย ทีเอฟเอส สามารถช่วยได้โดยการสร้างสภาพแวดล้อมที่มีระบบควบคุมรหัสต้นฉบับ (Source Code) แบบเบ็ดเสร็จอย่างที่นักพัฒนาต้องการ รวมทั้งมีระบบติดตามการแก้ปัญหาอย่างที่

ผู้จัดการโครงการและนักทดสอบต้องการอีกด้วย เพื่อช่วยให้การสื่อสารของทีมงานเป็นไปอย่างราบรื่น ซึ่งผลลัพธ์ที่เกิดขึ้นก็คือสภาพแวดล้อมที่รองรับการทำงานร่วมกันแบบครบวงจรสำหรับทีมงานทั้งหมดนั่นเอง

ฟังก์ชันหลักที่ได้ใช้งานจากโปรแกรมทีเอฟเอสคือ

2.2.1.1 ระบบควบคุมเวอร์ชันและรวบรวมโค้ดเข้าด้วยกัน

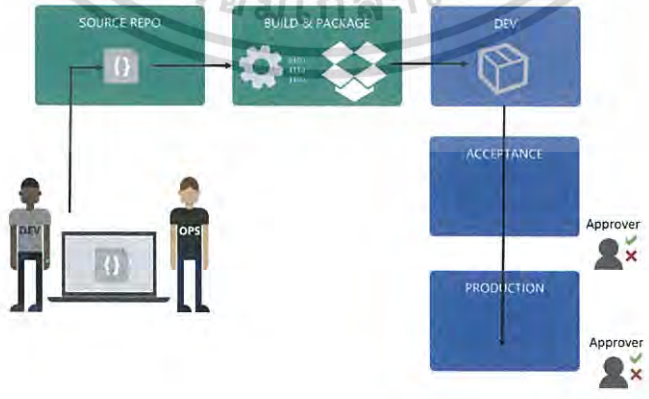
คุณสมบัติของ ทีเอฟเอส ที่ชื่อ เวอร์ชันคอนโทรล (Version Control) มีฟังก์ชันที่ทันสมัยในตัวมากมาย อาทิเช่น ระบบสื่อสารที่อิงกับ เอกซ์เอ็มแอลเว็บเซอร์วิส และสามารถผสานการทำงานกับองค์ประกอบส่วนอื่น ๆ ของ ทีเอฟเอส ได้อย่างคล่องตัว ในขณะที่ประสิทธิภาพและเสถียรภาพของ เอสคิวแอล เซิร์ฟเวอร์ จะทำให้ฟังก์ชันเหล่านี้ทำงานได้ดีขึ้นกว่าเดิมอีกด้วย

2.2.1.2 การสร้างและทดสอบ (Build and Test)

หัวใจสำคัญของโครงการซอฟต์แวร์ทุกโครงการก็คือการบิลด์และเทส (Build and Test) คือการแปลงรหัสต้นฉบับให้กลายเป็นแอปพลิเคชันนั่นเอง ดังนั้น ทีเอฟเอส จึงได้จัดเตรียมฟังก์ชันบิลด์ (Build) แบบเบ็ดเสร็จเอาไว้ ผ่านทางคุณสมบัติบิลด์ที่รองรับการทำงานของทีมงานทั้งหมดได้ รูปแบบดังกล่าวจะช่วยให้ผู้ใช้งานจัดตารางเวลาและประมวล ผลการบิลด์ของโซลูชันหลาย ๆ ชุดได้ ซึ่งการบิลด์จะต้องการผ่านการทดสอบและการวิเคราะห์โค้ดอย่างเข้มงวดเสีย ก่อนจึงจะเผยแพร่ผลลัพธ์ออกไปได้

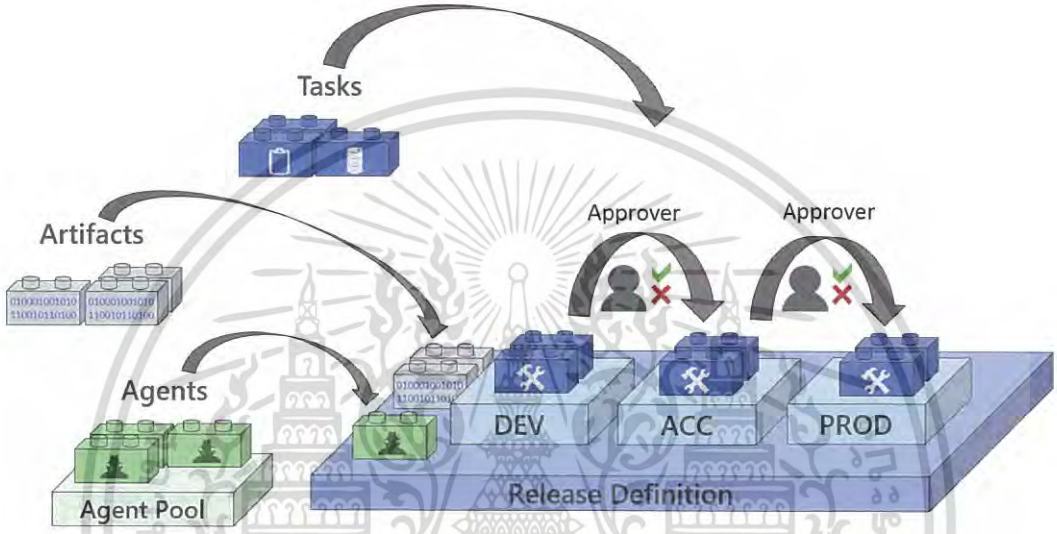
2.2.1.3 การจัดการรีรีส (Release Management) [8]

การจัดการรีรีส เป็นบริการหนึ่งในโปรแกรมทีเอฟ ที่จะช่วยในการเคลื่อนย้ายซอฟต์แวร์ต่าง ๆ อัตโนมัติไปสู่สภาพแวดล้อมอื่น ๆ และทดสอบซอฟต์แวร์ในหลาย ๆ สภาพแวดล้อม การใช้การจัดการรีรีสสามารถเลือกได้ว่าจะให้เคลื่อนย้ายอัตโนมัติไปสู่เซิร์ฟเวอร์จริง (Production) หรือจะมีการตั้งค่าผู้อนุมัติ (Approval) ก่อนนำขึ้นเซิร์ฟเวอร์จริง เป็นสิ่งสำคัญที่จะช่วยทีมเคลื่อนย้ายซอฟต์แวร์ได้อย่างต่อเนื่องถึงผู้ใช้งานด้วยความเร็วที่เพิ่มขึ้นและความเสี่ยงที่ลดลง มีการทำงานดังภาพที่ 2.2 เริ่มจากนักพัฒนาเขียนโปรแกรมขึ้นมา จากนั้นเอาไฟล์ไปไว้ในระบบเก็บโค้ด และให้ระบบการจัดการรีรีส เคลื่อนย้ายไฟล์นั้นไปยังสภาพแวดล้อมต่าง ๆ



ภาพที่ 2.2 การทำงานของการจัดการรีรีส [9]

รีรีสเดฟฟินิชัน (Release Definition) คือที่เก็บสภาพแวดล้อมต่าง ๆ ที่สามารถนำไฟล์โค้ดหรือซอฟต์แวร์ขึ้นไปเพื่อทำงานบนสภาพแวดล้อมนั้น ๆ ได้ สามารถใส่คำสั่งอัตโนมัติให้กับแต่ละรีรีสเดฟฟินิชันได้ด้วย ในแต่ละรีรีสเดฟฟินิชัน สามารถกำหนดผู้อนุมัติการนำโค้ดขึ้นระบบให้กับแต่ละสภาพแวดล้อม ดังภาพที่ 2.3 มีฐานของรีรีสไว้รองรับหลายสภาพแวดล้อม มีการใส่ทาสก์เข้าไป และสามารถกำหนดผู้อนุมัติในแต่ละสภาพแวดล้อมได้



ภาพที่ 2.3 ภาพของการทำการจัดการรีรีสในส่วนของรีรีสเดฟฟินิชัน [10]

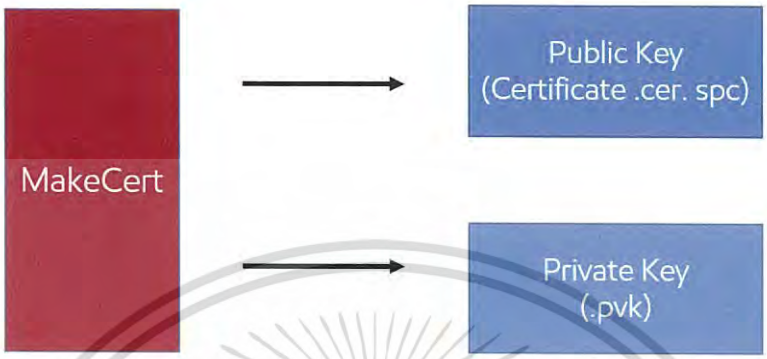
2.2.2 โปรแกรมเมคเสิท (MakeCert) [11]

โปรแกรมเมคเสิทเป็นเครื่องมือสร้างใบรับรองสร้างใบรับรองเอกซ์.509 (X.509) เพื่อทำการทดสอบเท่านั้น โดยจะสร้างกุญแจคู่สาธารณะและส่วนตัวสำหรับการลงลายมือดิจิทัล และจัดเก็บในแฟ้มใบรับรอง เครื่องมือนี้ยังเชื่อมโยงคู่ของกุญแจที่มีชื่อของผู้สร้าง และสร้างใบรับรองเอกซ์.509 ที่ผูกผู้ใช้ระบุชื่อไปส่วนของคู่กุญแจสาธารณะ

โปรแกรมเมคเสิทจะรวมตัวเลือกพื้นฐานและเพิ่มเติม ตัวเลือกพื้นฐานจะนิยมใช้กันมากที่สุดเพื่อสร้างใบรับรอง ตัวเลือกเพิ่มเติมจะเพิ่มความยืดหยุ่นให้กับใบรับรอง

โปรแกรมเมคเสิทนี้จะมีการติดตั้งโดยอัตโนมัติกับวิชวลสตูดิโอและวินโดว์เอสดีเค (Windows SDK) เรียกใช้เครื่องมือ แนะนำให้ใช้ซีเอ็มดีพร้อมท์ (CMD Prompt) รับคำสั่งสตูดิโอวิชวล (Studio Visual) หรือวินโดว์เอสดีเค คำสั่ง (CMD Shell) ยูทิลิตี้เหล่านี้ช่วยให้คุณสามารถเรียกใช้เครื่องมือง่าย ๆ โดยไม่ต้องไปที่โฟลเดอร์การติดตั้ง สำหรับข้อมูลเพิ่มเติม ดูวิชวลสตูดิโอและพร้อมท์คำสั่งเอสดีเคของวินโดว์

การทำงานของโปรแกรมเมคเสิท มีการทำงานโดยเรียกโปรแกรมผ่านคอมมานไลน์เพื่อสร้างกุญแจสาธารณะและกุญแจส่วนตัวได้เพื่อใช้ในการลงลายมือดิจิทัลต่อไป ดังภาพที่ 2.4



ภาพที่ 2.4 การทำงานของโปรแกรมเมคเสิท

รูปแบบคำสั่ง (Syntax)

MakeCert.exe [Options] OutputCertificateFile

ตัวอย่างคำสั่งพื้นฐาน ดังตารางที่ 2.4

ตารางที่ 2.4 ตารางอธิบายคำสั่งพื้นฐานในโปรแกรมเมคเสิท

ลำดับ	คำสั่ง	คำอธิบาย
1.	-n "Name"	ชื่อผู้สร้างใบรับรอง. ชื่อควรสอดคล้องกับมาตรฐาน X.500 รูปแบบคำสั่งที่ง่ายที่สุดคือ "CN=MyName" .ตัวอย่าง: -n "CN=Test".
2.	-pe	กุญแจส่วนตัวสามารถนำออกมาได้.
3.	-ss SubjectCertStoreName	ชื่อของสถานที่เก็บใบรับรองและที่อยู่ของที่เก็บใบรับรอง
4.	-# SerialNumber	เลขของใบรับรอง. ค่าที่มากที่สุดคือ 2^31. ค่าเริ่มต้นถูกสร้างขึ้นด้วยเครื่องมือในการรับประกันความเป็นเอกลักษณ์

ตารางที่ 2.4 ตารางอธิบายคำสั่งพื้นฐานในโปรแกรมเมคเสิท (ต่อ)

ลำดับ	คำสั่ง	คำอธิบาย
5.	-$ CertificateAuthority	ประเภทของใบรับรอง. สิทธิของใบรับรองควรจะตั้งค่าเป็นทางการค้า (commercial สำหรับใบรับรองที่ใช้ในซอฟต์แวร์ทางการค้า) หรือทางเฉพาะ (individual สำหรับใบรับรองที่ใช้โดยใบรับรองซอฟต์แวร์เฉพาะ)
6.	-?	แสดงคำสั่งพื้นฐาน.
7.	-!	แสดงคำสั่งเพิ่มเติม.
8.	-a algorithm	กำหนดลายมืออัลกอริทึม. อัลกอริทึมควรเป็น md5, sha1 (ค่าเริ่มต้น), sha256, sha384, or sha512.
9.	-b mm/dd/yyyy	กำหนดวันออกใบรับรอง โดยค่าเริ่มต้นเป็นวันที่สร้าง
10.	-cy certType	กำหนดประเภทของใบรับรอง ค่าที่ถูกต้องสุดสำหรับเอนทิตีปลายและอำนาจในการรับรอง
11.	-e mm/dd/yyyy	กำหนดวันหมดอายุของใบรับรอง. โดยค่าเริ่มต้นจะเป็น 12/31/2039 11:59:59 GMT.
12.	-ic file	กำหนดคนออกใบรับรอง
13.	-ik keyName	กำหนดชื่อกุญแจคอนเทนเนอร์
14.	-in name	กำหนดชื่อทั่วไปสำหรับใบรับรอง
15.	-ip provider	กำหนดชื่อคนเข้ารหัสเอพีไอ สำหรับข้อมูลเกี่ยวกับการเข้ารหัสเอพีไอ (CryptoAPI)
16.	-is store	กำหนดชื่อที่เก็บใบรับรอง
17.	-iv pvkFile	กำหนดกุญแจส่วนตัว

ตารางที่ 2.4 ตารางอธิบายคำสั่งพื้นฐานในโปรแกรมเมคเสิท (ต่อ)

ลำดับ	คำสั่ง	คำอธิบาย
18.	**-len** number	กำหนดความยาวในการสร้างกุญแจในรูปของบิท
19.	**-m** number	กำหนดช่วงเวลาวันหมดอายุในรูปแบบเดือน
20.	-r	กำหนดว่าใบรับรองเป็นแบบสร้างด้วยตนเอง
21.	**-sc** file	กำหนดหัวข้อใบรับรอง
22.	**-sv** pvkFile	กำหนดหัวข้อกุญแจส่วนตัว ไฟล์นี้จะถูกสร้างถ้าไม่ถูกค้นพบ

2.2.3 โปรแกรมพีวีเคทูพีเอฟเอ็ก (pvk2pfx) [12]

โปรแกรมพีวีเคทูพีเอฟเอกซ์ เป็นเครื่องมือคอมมานไลน์ (command line) ไว้คัดลอกกุญแจสาธารณะ (ใบรับรอง) คือไฟล์ประเภทเซอร์ (.cer) กับกุญแจส่วนตัว คือไฟล์ประเภทพีวีเค (.pvk) เพื่อแปลงเป็นไฟล์พีเอฟเอกซ์ (.pfx หรือ Personal Information Exchange)

รูปแบบคำสั่ง

pvk2pfx.exe /pvk pvkfilename.pvk [/pi pvkpassword] /spc spcfilename.ext [/pfx] pfxfilename.pfx [/po pfxpassword] [/f]]

ตัวอย่างคำสั่งพื้นฐาน ดังตารางที่ 2.5

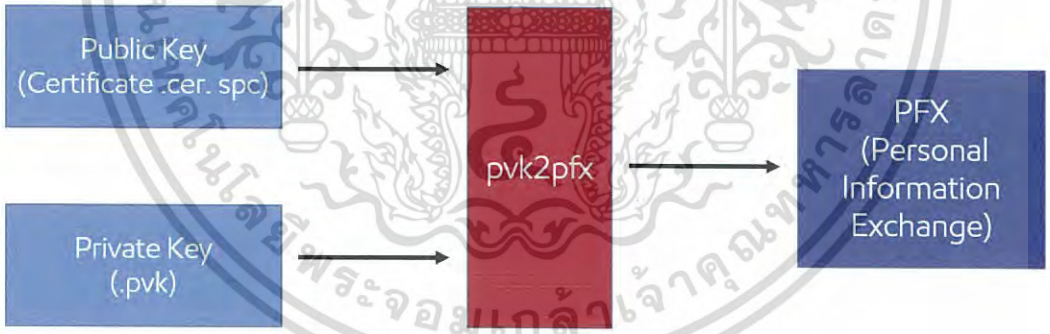
ตารางที่ 2.5 ตารางอธิบายคำสั่งพื้นฐานในโปรแกรมพีวีเคทูพีเอฟเอกซ์

ลำดับ	คำสั่ง	คำอธิบาย
1.	**/pvk** pvkfilename.pvk	กำหนดชื่อของกุญแจส่วนตัว
2.	**/spc** spcfilename.ext	กำหนดชื่อใบรับรอง
3.	**/pfx** pfxfilename.pfx	กำหนดชื่อของไฟล์พีเอฟเอกซ์

ตารางที่ 2.5 ตารางอธิบายคำสั่งพื้นฐานในโปรแกรมพีวีเคทูพีเอฟเอกซ์ (ต่อ)

ลำดับ	คำสั่ง	คำอธิบาย
4.	**/pfx** pfxfilename.pfx	กำหนดชื่อของไฟล์พีเอฟเอกซ์
5.	**/pi** pvkpassword	กำหนดรหัสของกุญแจส่วนตัว
6.	**/po** pfxpassword	กำหนดรหัสของไฟล์พีเอฟเอกซ์ ถ้ารหัสของไฟล์พีเอฟเอกซ์ ไม่ได้ถูกกำหนด ค่าเริ่มต้นจะเป็นรหัสเดียวกับรหัสของกุญแจส่วนตัว
7.	**/f**	ให้สร้างไฟล์พีเอฟเอกซ์ ไฟล์ใหม่ทับของไฟล์พีเอฟเอกซ์ไฟล์เดิม ถ้าไฟล์พีเอฟเอกซ์ ไฟล์ใหม่มีที่ชื่อเหมือนกับ ไฟล์พีเอฟเอกซ์ไฟล์เดิม

การทำงานของโปรแกรมพีวีเคทูพีเอฟเอกซ์ มีการทำงานโดยการนำกุญแจสาธารณะและกุญแจส่วนตัวมารวมกันเป็นไฟล์เดียวกันที่เรียกว่าไฟล์พีวีเคทูพีเอฟเอกซ์ เพื่อให้มีฟอแมตตรงตามโปรแกรมไซน์ทูลกำหนดเอาไว้ในการลงลายมือดิจิทัล เป็นไปดังภาพที่ 2.5



ภาพที่ 2.5 การทำงานของโปรแกรมพีวีเคทูพีเอฟเอกซ์

2.2.4 โปรแกรมไซน์ทูล (Signtool) [13]

โปรแกรมไซน์ทูลเป็นเครื่องมือคำสั่งคอมมานด์ไลน์ใช้ในการเซ็นรับรองไฟล์ดิจิทัล ตรวจสอบลายมือดิจิทัล หรือเวลาในการเซ็นไฟล์ ประเภทไฟล์ที่โปรแกรมไซน์ทูลรองรับมีดังนี้

รูปแบบคำสั่ง (Syntax)

Signtool.exe [Command][Options][FileName ...]

ตัวอย่างคำสั่งพื้นฐาน ดังตารางที่ 2.6

ตารางที่ 2.6 ตารางอธิบายคำสั่งพื้นฐานในโปรแกรมไซน์ทูล

ลำดับ	คำสั่ง	คำอธิบาย
1.	**catdb**	เพิ่มหรือลบไฟล์ในแค็ตตาล็อกจากฐานข้อมูล
2.	**sign**	เซ็นไฟล์ดิจิทัล
3.	**signwizard**	คำสั่งนี้ไม่รองรับบนระบบวินโดว์วิสต้า (Window Vista) และเวอร์ชันที่เก่ากว่า ส่งมาจากไซน์นิงวิซาร์ด (signing wizard) เฉพาะไฟล์เดี่ยวที่สามารถกำหนดชื่อไฟล์ในตัวแปรคอมมานไลน์
4.	**timestamp**	ที่อยู่ไทม์แสตมป์เซิร์ฟเวอร์
5.	**verify**	พิสูจน์ลายมือดิจิทัลในไฟล์
6.	**/f** SignCertFile	กำหนดไฟล์ใบรับรองเพื่อการเซ็นในรูปแบบ .pfx เท่านั้น สามารถใช้โปรแกรมพีวีเคทูพีเอฟเอกซ์ ในการแปลงกุญแจส่วนตัวและใบรับรองเป็นไฟล์ .pfx
7.	**/p** Password	กำหนดรหัสเพื่อเปิดไฟล์ .pfx กำหนดไฟล์ .pfx ได้ด้วยคำสั่ง /f
8.	**/t** URL	กำหนดยูอาร์แอล (URL) สำหรับไทม์สแตมป์เซิร์ฟเวอร์ ถ้าคำสั่งนี้ไม่ปรากฏ ไฟล์ที่ได้รับการเซ็นจะไม่มีไทม์สแตมป์ การแจ้งเตือนจะถูกสร้างขึ้นถ้าไทม์สแตมป์ล้มเหลว

ประเภทไฟล์ที่โปรแกรมไซน์ทูลรองรับดังตารางที่ 2.7

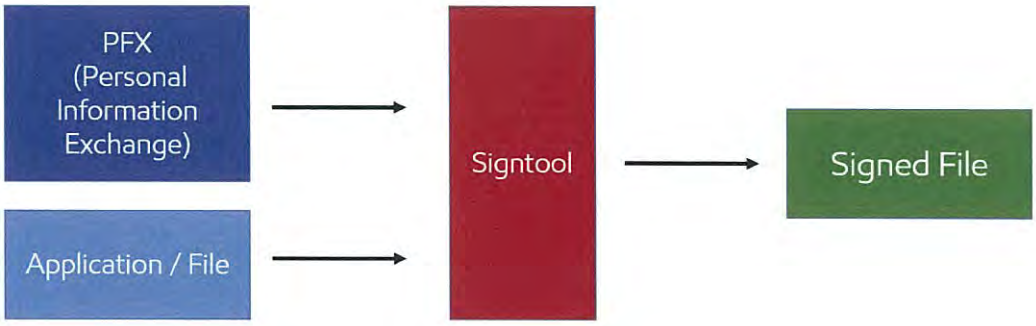
ตารางที่ 2.7 ประเภทของไฟล์ที่โปรแกรมไซน์ทูลรองรองรับ

ลำดับ	ประเภทไฟล์ที่โปรแกรมไซน์ทูลรองรับ
1.	.appx
2.	.cab

ตารางที่ 2.7 ประเภทของไฟล์ที่โปรแกรมไซน์ทูลรองรองรับ (ต่อ)

ลำดับ	ประเภทไฟล์ที่โปรแกรมไซน์ทูลรองรับ
3.	.cat
4.	.dll
5.	.exe
6.	.js
7.	.vbs
8.	.wsf
9.	.msi
10.	.msp
11.	.mst
12.	.ocx
13.	.Javascript
14.	.ps1
15.	.stl
16.	.sys

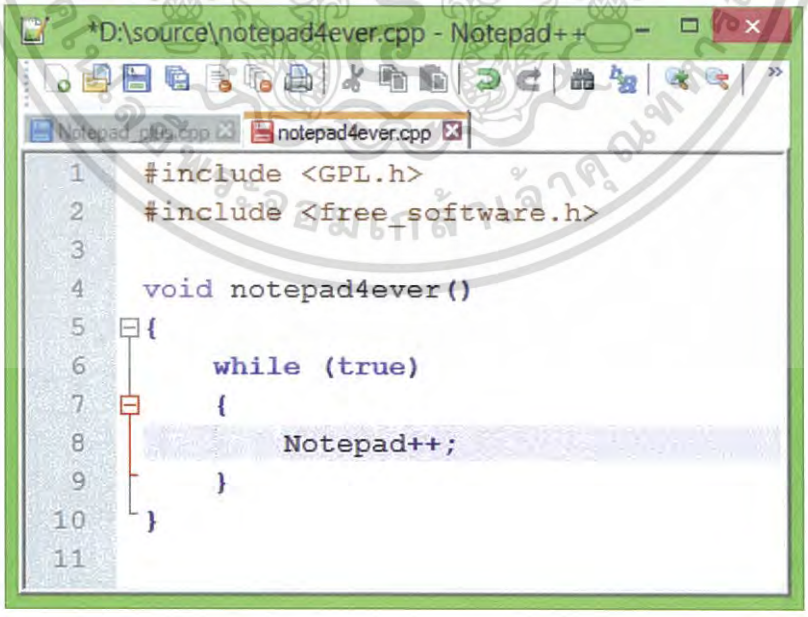
การทำงานของโปรแกรมไซน์ทูล มีการทำงานโดยมีการนำไฟล์โค้ดหรือแอปพลิเคชันมาเข้ารหัสลงลายมือดิจิทัลโดยเรียกโปรแกรมไซน์ทูลผ่านคอมมานไลน์ หลังจากลงลายมือดิจิทัลแล้วจะได้เป็นไฟล์ที่มีลายมือดิจิทัล ดังภาพที่ 2.6



ภาพที่ 2.6 การทำงานของโปรแกรมไซน์ทูล

2.2.5 โปรแกรมโน้ตแพดพลัสพลัส (Notepad++) [14]

โปรแกรมโน้ตแพดพลัสพลัส คือเทกซ์เอดิเตอร์ (Text Editor) เป็นเครื่องมือประเภทเดียวกันกับโน้ตแพดที่มาพร้อมกับวินโดว์ แต่ความสามารถของโน้ตแพดพลัสพลัสนั้นสูงกว่า โน้ตแพดพลัสพลัสสามารถแก้ไขไฟล์ต่าง ๆ ได้หลากหลายนามสกุล หลายรูปแบบ สามารถเปิดได้หลายไฟล์ในคราวเดียวกันโดยจะแยกเป็นแท็บของบุคคลของมันอย่างชัดเจน ทำให้สะดวกต่อการใช้งาน อีกทั้งมีการกำหนดสีที่กำหนดให้กับแต่ละโค้ด ทำให้ง่ายต่อการเขียน และแก้ไขโค้ดจึงเหมาะสำหรับนักพัฒนาซอฟต์แวร์ หรือนักพัฒนาเว็บเป็นอย่างยิ่ง โดยจุดเด่นอีกอย่างของ โน้ตแพดพลัสพลัสคือมีขนาดไฟล์ที่เล็กเพียงแค่ 8 เมกะไบต์ใช้งานทรัพยากรเครื่องต่ำ ไม่มีอาการค้างแม้จะเปิดหลายไฟล์พร้อมกัน ที่สำคัญที่สุดคือเจ้าโปรแกรมนี้รองรับภาษาไทย และเวอร์ชันยังฟรีอีกด้วย ภาพที่ 2.7 คือตัวอย่างโปรแกรมโน้ตแพดพลัสพลัส



ภาพที่ 2.7 ภาพโปรแกรมโน้ตแพดพลัสพลัส

2.2.6 โปรแกรมไมโครซอฟท์เวิร์ด (Microsoft Word) [15]

ไมโครซอฟท์ เวิร์ด เป็นโปรแกรมประมวลคำเพื่อการค้า ออกแบบโดยไมโครซอฟท์ เปิดตัวเป็นครั้งแรกในปี ค.ศ. 1983 ภายใต้ชื่อ มัลติ-ทูล เวิร์ด สำหรับระบบปฏิบัติการซีนิกซ์ (Xenix) โดยมีเวอร์ชันอื่น ๆ ออกมาอีกภายหลังเพื่อทำงานเขียนสำหรับแพลตฟอร์มอื่น ๆ อาทิเช่น ไอบีเอ็มพีซีรันบนดอส (1983), แอปเปิล แมคอินทอช (1984), เอทีแอนที ยูนิกซ์ พีซี (AT&T Unix PC (1985)), และไมโครซอฟท์ วินโดวส์ (1989) โดยเป็นองค์กอบหนึ่งของซอฟต์แวร์ระบบไมโครซอฟท์ ออฟฟิศ ซึ่งเป็นผลิตภัณฑ์ที่ขายแยกต่างหาก และรวมอยู่ในไมโครซอฟท์ เวิร์ก สูท เวอร์ชันปัจจุบัน คือ ไมโครซอฟท์ เวิร์ด 2010 สำหรับวินโดว์ และ 2011 สำหรับแมค

โปรแกรมไมโครซอฟท์เวิร์ด ซึ่งเป็นโปรแกรมประมวลผลคำแบบพิเศษ ช่วยให้สร้างเอกสารแบบมืออาชีพอย่างมีประสิทธิภาพและประหยัด เช่น เหมาะกับงานด้านการพิมพ์เอกสารทุกชนิด สามารถพิมพ์เอกสารออกมาเป็นชุด ๆ ซึ่งเอกสารอาจเป็นจดหมาย บันทึกข้อความ รายงาน บทความ ประวัติย่อ และยังสามารถตรวจสอบ ทบทวน แก้ไข ปรับปรุงความถูกต้องในการพิมพ์เอกสารได้อย่างง่ายดาย สามารถตรวจสอบ สะกดคำและหลักไวยากรณ์ เพิ่มตาราง เพิ่มกราฟิก ในเอกสารได้อย่างง่ายดาย หรือเพิ่มเติมข้อมูลได้ตลอดเวลา สามารถใช้ลักษณะของการจัดพิมพ์ด้วยคอมพิวเตอร์แบบตั้งโต๊ะ (Desktop Publishing) เพื่อสร้างโบชัวร์ (Brochures) ด้านสื่อโฆษณา (Advertisements) และจดหมายข่าว (Newsletters) ได้ด้วยโปรแกรมประมวลผลคำ (Word Processor)

2.3 ทฤษฎีการเขียนผังงาน (Flowchart) [16]

ผังงาน (Flowchart) คือ รูปภาพหรือสัญลักษณ์ที่ใช้เขียนแทนคำอธิบาย ข้อความหรือคำพูดที่ใช้ในอัลกอริทึม เพราะการที่จะเข้าใจขั้นตอนได้ง่ายและตรงกันนั้น การใช้คำพูดหรือข้อความอาจทำได้ยากกว่าการใช้รูปภาพหรือสัญลักษณ์ ผังงานสามารถแบ่งออกเป็น 2 ประเภทใหญ่ ๆ คือ

1. ผังงานระบบ (System Flowchart)
2. ผังงานโปรแกรม (Program Flowchart)

2.3.1 ผังงานระบบ (System Flowchart)

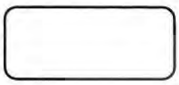
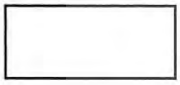
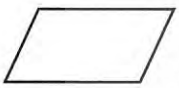
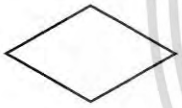
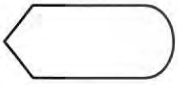
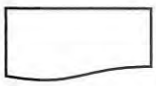
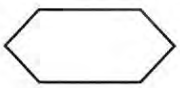
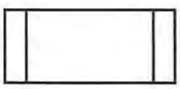
เป็นผังแสดงขั้นตอนการทำงานภายในระบบ คำว่าระบบงาน หมายถึง ส่วนต่าง ๆ ที่เกี่ยวข้องกับงานทั้งหมด ทั้งวัสดุ เครื่องจักร อุปกรณ์ และ บุคลากร แสดงขั้นตอนเริ่มต้นว่ามีเอกสารเบื้องต้นจากส่วนใดของระบบงาน ผ่านไปยังหน่วยงานใด มีกิจกรรมอะไรในหน่วยงานนั้น ส่งงานต่อไปที่ใดจึงจะเสร็จสิ้น บางส่วนจะเกี่ยวกับคน บางส่วนเกี่ยวกับคอมพิวเตอร์ ต้องนำส่วนที่เกี่ยวกับคอมพิวเตอร์มาเขียนโปรแกรมทั้งแสดงรายละเอียดการทำงานแยกเป็นผังงานโปรแกรม

2.3.2 ผังงานโปรแกรม (Program Flowchart)

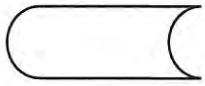
เป็นผังแสดงลำดับขั้นตอนการทำงานในโปรแกรม มีส่วนแสดงการทำงานในขั้นการรับข้อมูล การคำนวณหรือการประมวลผล และการแสดงผล เรียกอีกอย่างหนึ่งว่า ผังเขียนโปรแกรม หรือ ผังงาน

2.3.3 สัญลักษณ์ของผังงาน (Flowchart Symbol) ดังตารางที่ 2.10

ตารางที่ 2.10 สัญลักษณ์ของผังงาน

สัญลักษณ์	ความหมาย
	Terminator ใช้แสดงจุดเริ่มต้นและจุดสิ้นสุดของโปรแกรม
	Process ใช้ในการประมวลผลข้อมูล กำหนดค่า หรือการคำนวณทางคณิตศาสตร์
	Input/Output หรือ I/O ใช้ในการรับข้อมูล แสดงผลข้อมุลโดยไม่ระบุอุปกรณ์
	Manual Input ใช้ในการรับข้อมูลจากแป้นพิมพ์
	Decision Symbol ใช้ในการเปรียบเทียบเงื่อนไขหรือตัดสินใจ
	Display ใช้เมื่อต้องการระบุให้แสดงข้อมูลบนจอภาพ
	Document Symbol ใช้เมื่อต้องการระบุให้แสดงข้อมูลบนเครื่องพิมพ์
	Preparation การเตรียมทำงานลำดับถัดไป
	Predefined Process โปรแกรมย่อย หรือโมดูลเริ่มทำงานหลังจากจบคำสั่งในโปรแกรมย่อยแล้ว จะกลับมาทำคำสั่งต่อไป

ตารางที่ 2.10 สัญลักษณ์ของผังงาน (ต่อ)

สัญลักษณ์	ความหมาย
	Online Storage แหล่งเก็บข้อมูลออนไลน์ หรือหน่วยความจำสำรอง
	Connector หรือ On-page Connector จุดเชื่อมผังงานในหน้าเดียวกัน
	Connector หรือ Off-page Connector ชุดเชื่อมผังงานที่อยู่หน้าต่างกัน

2.3.4 ประโยชน์ของผังงาน

1. ทำให้เข้าใจและแยกปัญหาต่าง ๆ ได้ง่ายขึ้น
2. ผู้เขียนโปรแกรมมองเห็นลำดับการทำงาน รู้ว่าสิ่งใดควรทำก่อน สิ่งใดควรทำหลัง
3. สามารถหาข้อผิดพลาดของโปรแกรมได้ง่าย
4. ทำให้ผู้อื่นเข้าใจการทำงานได้ง่ายกว่าการดูจาก ซอร์สโค้ด
5. ไม่ขึ้นกับภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง ผู้อื่นสามารถเรียนรู้เข้าใจได้ง่าย

2.3.5 ข้อจำกัดของผังงาน

ผู้เขียนโปรแกรมบางคนไม่นิยมเขียนผังงานก่อนการเขียนโปรแกรม เพราะเห็นว่าเสียเวลา นอกจากนี้แล้ว ยังมีข้อจำกัดอื่น ๆ อีกคือ

1.ผังงานเป็นการสื่อสารความหมายระหว่างบุคคลกับบุคคลมากกว่าที่สื่อความหมายระหว่างเครื่อง เพราะผังงานไม่ขึ้นกับภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง ทำให้เครื่องไม่สามารถรับและเข้าใจได้ว่าในผังงานนั้นต้องการให้ทำอะไร

2. ในบางครั้ง เมื่อพิจารณาจากผังงาน จะไม่สามารถทราบได้ว่า ขั้นตอนการทำงานใดสำคัญกว่ากัน เพราะทุก ๆ ขั้นตอนจะใช้รูปภาพหรือสัญลักษณ์ในลักษณะเดียวกัน

3. การเขียนผังงานเป็นการสิ้นเปลือง เพราะจะต้องใช้กระดาษและอุปกรณ์อื่น ๆ เพื่อประกอบการเขียนภาพ ซึ่งไม่สามารถเขียนด้วยมือได้อย่างเดียว และในบางครั้ง การเขียนผังงานอาจจะต้องใช้กระดาษมากกว่า 1 แผ่น หรือ 1 หน้า ซึ่งถ้าเป็นข้อความอธิบายอาจใช้เพียง 2-3 บรรทัดเท่านั้น

2.4 ทฤษฎีที่เกี่ยวข้องด้านการเข้ารหัส [17]

การเข้ารหัสข้อมูล (Encryption)

การเข้ารหัสเป็นวิธีป้องกันข้อมูลจากการโจรกรรม ในขณะที่มีการรับและส่งข้อมูลผ่านทางเครือข่าย โดยข้อมูลทั้งหมดจะถูกแปลงเป็นรหัสที่ไม่สามารถอ่านได้ด้วยวิธีปกติ เรียกว่า การเข้ารหัส ดังนั้นแม้ว่าจะมีการโจรกรรมข้อมูลไปได้ แต่หากไม่สามารถถอดรหัส (Decryption) ก็ไม่สามารถเข้าใจข้อมูลเหล่านั้นได้ การเข้ารหัสอธิบายได้ดังภาพที่ 2.8 คือมีข้อความสาร จากนั้นนำไปเข้ารหัส จึงได้ออกมาเป็นข้อความที่เข้ารหัสแล้ว

ภาพที่ 2.8 ภาพทฤษฎีการเข้ารหัส

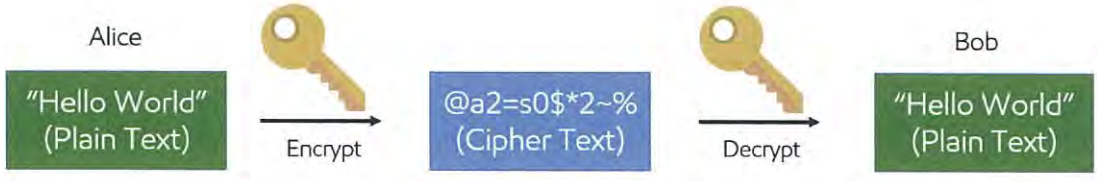
องค์ประกอบของการเข้ารหัส (Cryptography)

- ข้อความต้นฉบับ (Plain text) คือ ข้อมูลต้นฉบับซึ่งเป็นข้อความที่สามารถอ่านแล้วเข้าใจ
- อัลกอริทึมการเข้ารหัสลับ (Encryption Algorithm) คือ ขั้นตอนวิธีในโปรแกรมคอมพิวเตอร์ที่ใช้ในการแปลงข้อมูลต้นฉบับเป็นข้อมูลที่ได้รับการเข้ารหัส
- ข้อความไซเฟอร์ (Ciphertext) คือ ข้อมูลหรือข่าวสารที่ได้รับการเข้ารหัส ทำให้อ่านไม่รู้เรื่อง
- กุญแจลับ (Key) คือ เป็นกุญแจที่ใช้ร่วมกับ อัลกอริทึมในการเข้ารหัส และถอดรหัส
- อัลกอริทึมการถอดรหัสลับ (Decryption Algorithm) คือ กระบวนการหรือขั้นตอนในการแปลงข้อความไซเฟอร์ให้กลับเป็นข้อความต้นฉบับ โดยอาศัยกุญแจลับดอกเดียวกัน

การเข้ารหัสมีด้วยกัน 2 ลักษณะ คือ

2.4.1 การเข้ารหัสแบบสมมาตร (Symmetric Encryption)

การเข้ารหัสแบบสมมาตรคือการเข้ารหัสและถอดรหัสโดยใช้กุญแจรหัสตัวเดียวกันคือ ผู้ส่งและผู้รับจะต้องมีกุญแจรหัสที่เหมือนกันเพื่อใช้ในการเข้ารหัสและถอดรหัส ดังภาพที่ 2.9



ภาพที่ 2.9 การเข้ารหัสแบบสมมาตร

2.4.2 การเข้ารหัสแบบไม่สมมาตร (Asymmetric Encryption)

2.4.2.1 เข้ารหัสแบบไม่สมมาตรแบบปกติ

การเข้ารหัสและถอดรหัสโดยใช้กุญแจรหัสคนละตัวกัน การส่งจะมีกุญแจรหัสตัวหนึ่งในการเข้ารหัส และผู้รับก็จะมีกุญแจรหัสอีกตัวหนึ่งเพื่อใช้ในการถอดรหัส ผู้ใช้รายหนึ่ง ๆ จึงมีกุญแจรหัส 2 ค่าเสมอคือ กุญแจสาธารณะ (Public key) และกุญแจส่วนตัว (Private key) ผู้ใช้จะประกาศให้ผู้อื่นทราบถึงกุญแจสาธารณะของตนเองเพื่อให้นำไปใช้ในการ เข้ารหัสและส่งข้อมูลที่เข้ารหัสแล้วมาให้ ข้อมูลที่เข้ารหัสดังกล่าวจะถูกถอดออกได้โดยกุญแจส่วนตัวเท่านั้น ดังภาพที่ 2.10

ภาพที่ 2.10 การเข้ารหัสแบบไม่สมมาตร

2.4.2.2 การเข้ารหัสแบบลายมือดิจิทัล (Digital Signature)

1) ใบรับรองดิจิทัล (Digital Certificate)

การเข้ารหัสและลายมือชื่อดิจิทัล ในการทำธุรกรรม สามารถรักษาความลับของข้อมูล สามารถรักษาความถูกต้องของข้อมูลและสามารถระบุตัวบุคคลได้ระดับหนึ่ง เพื่อเพิ่มระดับความปลอดภัยในการระบุตัวบุคคลโดยสร้างความเชื่อถือมากขึ้นด้วย ใบรับรองดิจิทัล ซึ่งออกโดยองค์กรกลางที่เป็นที่เชื่อถือเรียกว่า องค์กรรับรองความถูกต้อง (Certification Authority) จะถูกนำมาใช้สำหรับยืนยันในตอนทำธุรกรรมว่าเป็นบุคคลนั้น ๆ จริง ตามที่ได้อ้างไว้

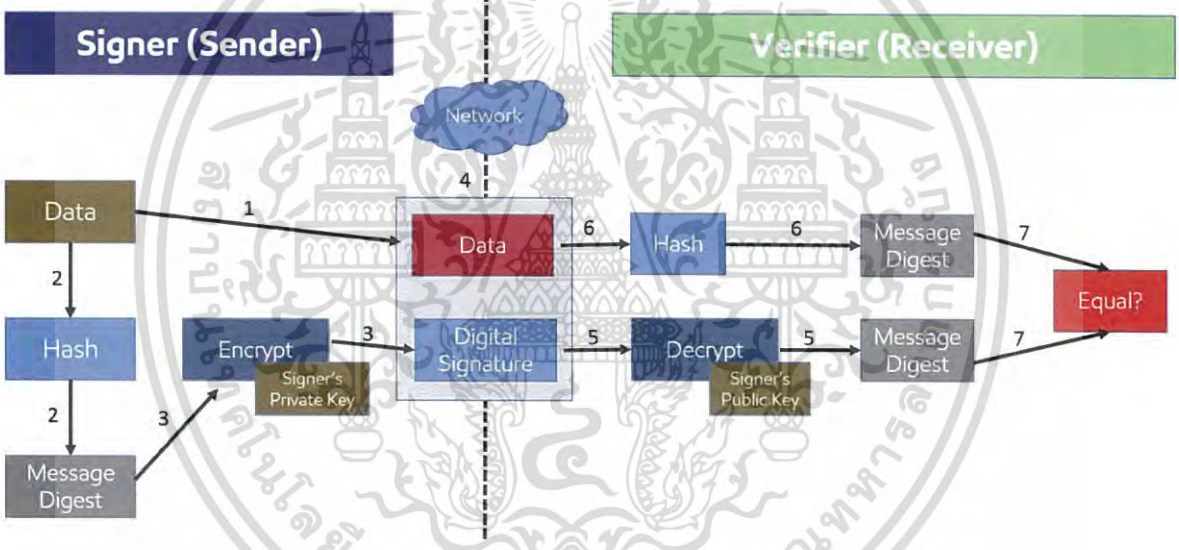
2) ลายมือดิจิทัล

ใบรับรองดิจิทัลเป็นสิ่งที่แสดงยืนยันตัวบุคคล เป็นข้อมูลที่แนบไปกับข้อความที่ส่งไป เพื่อเป็นการแสดงตัวตน (Authentication) ว่าผู้ส่งข้อความเป็นบุคคล โดยข้อมูลนั้นได้ถูกส่งมาจากผู้ส่งคนนั้นจริง ๆ และข้อความไม่ได้ถูกเปลี่ยนแปลงและแก้ไข ใช้กับการพิสูจน์ความถูกต้องของเอกสารตามกฎหมาย เช่น ด้านการเงิน การทำสัญญา และเอกสารอื่น ๆ ว่าเป็นของแท้นั้น สามารถทำได้โดยการตรวจสอบความถูกต้องของลายมือของผู้มีอำนาจอนุมัติ

ข้อความที่มีลายมือส่งไปยังอีกฝ่ายหนึ่ง จะมีลักษณะต่อไปนี้

1. ผู้รับสามารถพิสูจน์เอกลักษณ์ของผู้ที่อ้างนั้นว่าเป็นคนส่งข่าวสารจริง ๆ
2. ผู้ส่งไม่สามารถบอกปัดสิ่งที่เขียนลงไปในข้อความ
3. ผู้รับไม่สามารถที่จะประกอบและเปลี่ยนแปลงข้อความที่ตนส่งมาด้วยตนเองได้

การเข้ารหัสข้อความที่ยาวนั้น ค่อนข้างเสียเวลา เนื่องจากขั้นตอนการเข้ารหัสต้องใช้การคำนวณเป็นอย่างมาก จึงมีการสร้างขั้นตอนที่คำนวณได้อย่างรวดเร็ว โดยเปลี่ยนข้อความทั้งหมดให้เหลือเพียงข้อความสั้น ๆ เรียกว่า ข้อความไดเจส (Digest Message) ซึ่งจะถูกสร้างขึ้นด้วยกระบวนการเข้ารหัสยอดนิยมที่เรียกว่า วันเวย์แฮชฟังก์ชัน (One-way hash function) จะใช้ ข้อความไดเจสนี้ในการเข้ารหัสเพื่อเป็นลายมือดิจิทัล โดยจะแจกกุญแจสาธารณะ ไปยังผู้ที่ต้องการติดต่อ



ภาพที่ 2.11 ลักษณะการลงลายมือดิจิทัล

จากภาพที่ 2.11 การลงลายมือดิจิทัลมีวิธีการการดังนี้

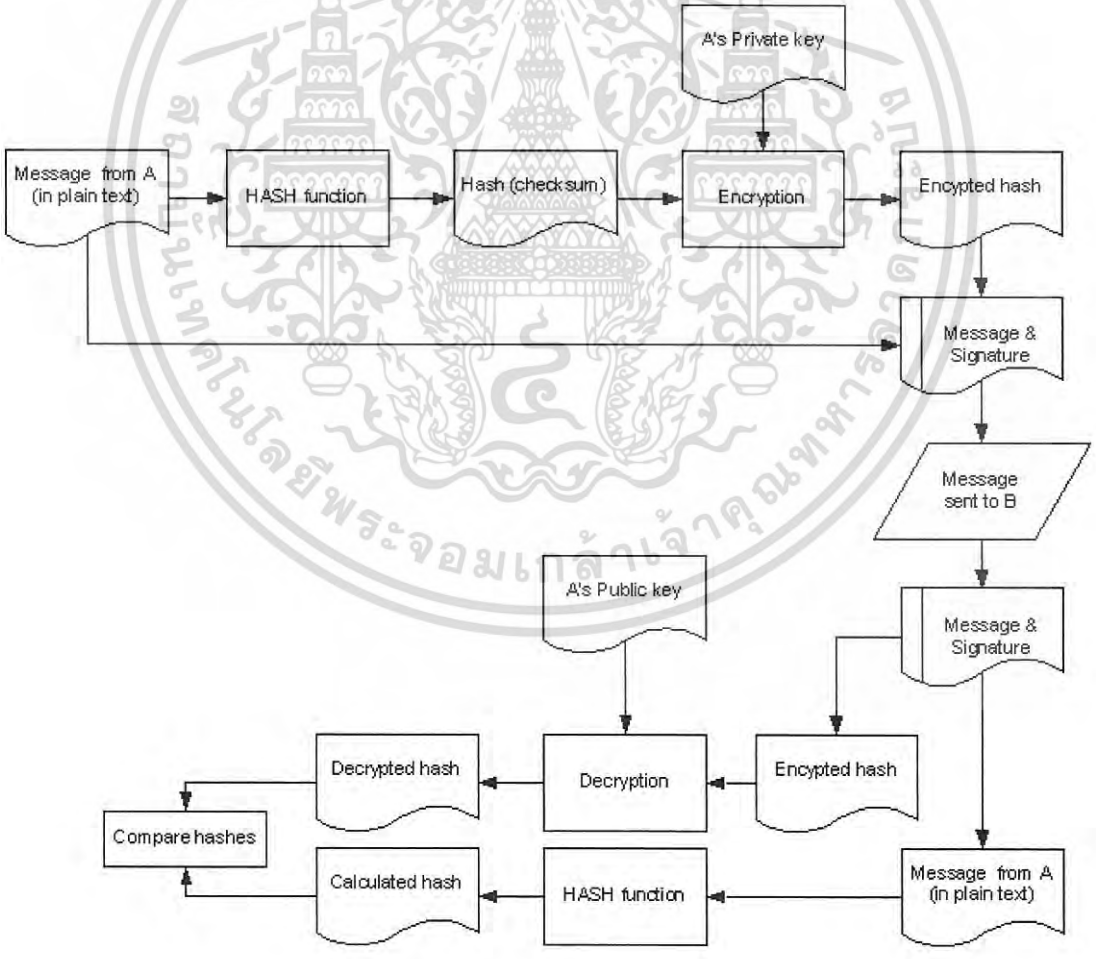
ด้านผู้ส่ง (Sender)

1. ผู้ส่งส่งข้อความต้นฉบับอาจจะเป็นไฟล์โค้ดหรือแอปพลิเคชันต่างๆ
2. ผู้ส่งนำข้อความต้นฉบับมาผ่านแฮชฟังก์ชันได้เป็นข้อความไดเจส
3. ผู้ส่งนำข้อความไดเจสมาเข้ารหัสด้วยกุญแจส่วนตัวของผู้ส่งจะได้ลายมือดิจิทัล
4. ส่งข้อความต้นฉบับและลายมือดิจิทัลไปสู่ผู้รับผ่านเครือข่าย

ด้านผู้รับ (Receiver)

5. ผู้รับนำลายมือดิจิทัลมาถอดรหัสด้วยกุญแจสาธารณะของผู้ส่งจะได้ข้อความไดเจส
6. ผู้รับนำข้อความต้นฉบับมาผ่านแฮชฟังก์ชันจะได้เป็นข้อความไดเจส
7. นำข้อความไดเจสทั้งสองมาเปรียบเทียบกัน

แผนผังการทำงานการลงลายมือดิจิทัลดังภาพที่ 2.12 มีวิธีการคือมีข้อความผู้ส่ง นำไปเข้าแฮชฟังก์ชัน จะได้ข้อความไดเจสออกมา และใช้กุญแจส่วนตัวของผู้ส่งในการเข้ารหัส จะได้เป็นข้อความที่มีการเข้ารหัสที่เรียกดิจิทัลซิกเนเจอร์แล้วส่งดิจิทัลซิกเนเจอร์ไปพร้อมกับข้อความต้นฉบับไปยังผู้รับ ผู้รับเมื่อได้รับข้อความ ผู้รับนำไปถอดรหัสด้วยกุญแจสาธารณะของผู้ส่ง จะได้ข้อความไดเจส และผู้รับยังนำข้อความต้นฉบับที่ได้รับมาเข้าแฮชฟังก์ชัน จะได้เป็นอีกหนึ่งข้อความไดเจส นำข้อความไดเจสนี้มาเทียบกัน ถ้าเหมือนกันแสดงว่ามีความถูกต้อง ถ้าไม่เหมือนกัน แสดงว่าข้อความนั้นได้รับการแก้ไขมา

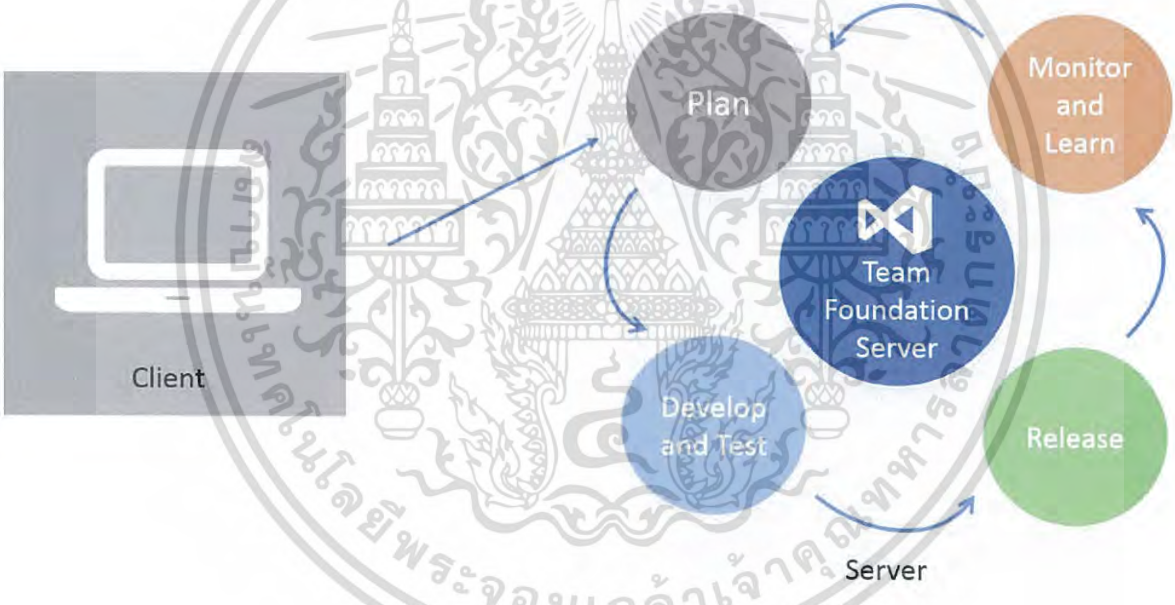


ภาพที่ 2.12 แผนผังการทำงานการลงลายมือดิจิทัล

บทที่ 3
วิธีการดำเนินการวิจัย

การศึกษาวิจัยในครั้งนี้เป็นการศึกษาวิจัยเชิงพัฒนา มีการศึกษาถึงปัจจัยต่าง ๆ ที่ส่งผลต่อกระบวนการทำงานของโปรแกรม โดยงานวิจัยชิ้นนี้ถูกแบ่งออกเป็น 2 ส่วน คือ การลงลายมือดิจิทัลในไฟล์โค้ด และเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ นำผลการวิเคราะห์มาพัฒนาระบบในโปรแกรมทีเอฟเอสให้มีประสิทธิภาพและความปลอดภัยที่มากขึ้นให้กับบริษัท เอ็กซอนโมบิล จำกัด ซึ่งความสัมพันธ์ระหว่างสิ่งต่าง ๆ ถูกแสดงออกมาในรูปแบบโครงสร้างทางสถาปัตยกรรมของระบบ (System Architecture) ดังต่อไปนี้

3.1 โครงสร้างทางสถาปัตยกรรมของระบบ



ภาพที่ 3.1 การทำงานของโปรแกรมทีเอฟเอส

จากภาพที่ 3.1 โครงสร้างทางสถาปัตยกรรมของระบบแสดงให้เห็นถึงการเชื่อมต่อในส่วนต่าง ๆ ของระบบทีเอฟเอสจะช่วยบริการกระบวนการสร้างแอปพลิเคชัน กระบวนการทำงานจะเริ่มต้นเมื่อผู้ใช้งานใช้โปรแกรมทีเอฟเอสในส่วนของระยะแรก คือการวางแผน (Plan) เก็บความต้องการของผู้ใช้งานเพื่อวางแผน กำหนดขอบเขตงานที่จะทำในอนาคต โดยใช้กระดานกันบัน เมื่อวางแผนเสร็จแล้วจะเข้าสู่ระยะที่สอง คือการพัฒนาและทดสอบ (Develop and Test) ในขั้นตอนนี้ผู้ใช้งานจะมีการทำเวอร์ชันของโปรแกรมต่าง ๆ การสร้าง (Build) และการตรวจสอบปัญหาของโปรแกรมก่อนนำไปทดสอบในสภาพแวดล้อมต่าง ๆ ในระยะที่สาม คือการทดสอบใน

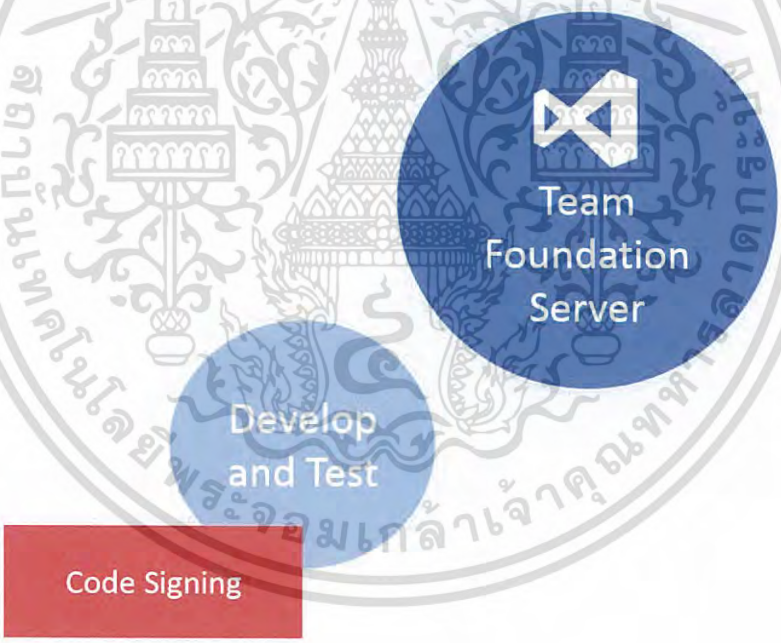
สภาพแวดล้อม (Release) คือการนำไปทดสอบในหลาย ๆ สภาพแวดล้อมได้โดยอัตโนมัติเพื่อหาข้อบกพร่องและแก้ไขในการทำงานของโปรแกรม ระยะที่สี่คือการมอนิเตอร์และเรียนรู้ (Monitor and Learn) คือการเก็บข้อมูลผลการทดลองต่าง ๆ เพื่อนำไปแก้ไข พัฒนา วางแผนใหม่เป็นวัฏจักรไปเรื่อย ๆ

3.2 กระบวนการทำงานของระบบ

งานวิจัยนี้ถูกแบ่งออกเป็น 2 ขอบเขตดังที่กล่าวไว้แล้วนั้น กระบวนการทำงานของแต่ละขอบเขตจะมีลักษณะที่แตกต่างกันออกไปตามหน้าที่การทำงานเฉพาะตัว แต่หนึ่งสิ่งที่มีความคล้ายคลึงกันของแต่ละระบบ นั่นคือการเชื่อมต่อการทำงานเข้ากับโปรแกรมทีเอฟเอส ซึ่งเป็นส่วนที่สำคัญที่สุดในกระบวนการทำงาน โดยระบบทำงานในแต่ละส่วนของชิ้นงานสามารถอธิบายได้ ดังนี้

3.2.1 การลงลายมือดิจิทัลในไฟล์โค้ด

3.2.1.1 ศึกษาทฤษฎีการเข้ารหัสถอดรหัสข้อมูลดิจิทัล และทดลองการลงลายมือดิจิทัล



ภาพที่ 3.2 ภาพการลงลายมือดิจิทัลในไฟล์โค้ดร่วมกับโปรแกรมทีเอฟเอส

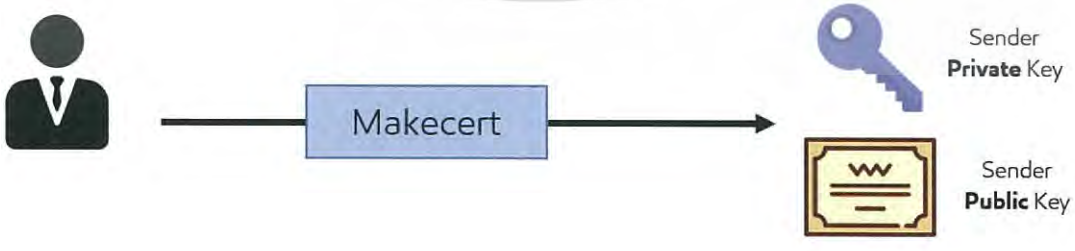
จากภาพที่ 3.2 โปรแกรมทีเอฟเอสให้บริการการทำงานไว้หลายฟังก์ชัน ฟังก์ชันหนึ่งของโปรแกรมทีเอฟเอสคือเป็นศูนย์รวมที่เก็บไฟล์โค้ดที่กำลังพัฒนาหรือพัฒนาสมบูรณ์แล้ว นักพัฒนาจากทีมต่างสามารถทำงานร่วมกันได้ ถึงแม้จะอยู่ไกลกัน กระบวนการทำงานในส่วนนี้เป็นการทำการในระยะที่สองของวัฏจักร

ซึ่งก็คือการพัฒนาและทดสอบ (Develop and Test) ฟังก์ชันที่จะเพิ่มเข้าไปในระยะนี้ก็คือการลงลายมือดิจิทัลลงบนไฟล์โค้ด โดยจะทำเข้ารหัสให้กับไฟล์โค้ดของผู้พัฒนา โดยใช้ใบรับรองของบริษัทเซ็นลงไปในไฟล์โค้ด ซึ่งถ้าเป็นคนที่ทำงานร่วมกันสามารถถอดรหัสออกมาได้ ซึ่งการทำงานก็คือ

ในด้านผู้ส่งนั้น เริ่มต้นโดยผู้ส่งเตรียมการส่งไฟล์ต้นฉบับหรือโค้ดไฟล์ต้นฉบับเอาไว้เพื่อเตรียมการลงลายมือดิจิทัล ดังภาพที่ 3.3 ขั้นตอนต่อมา ผู้ส่งทำการแฮชไฟล์ต้นฉบับอีกอันหนึ่งโดยใช้แฮชแบบทางเดียว จะได้เป็นข้อความไดเจส ดังภาพที่ 3.4 ผู้ส่งสร้างกุญแจส่วนตัวและกุญแจสาธารณะ (ใบรับรอง) ด้วยโปรแกรมเมคเสิท ดังภาพที่ 3.5

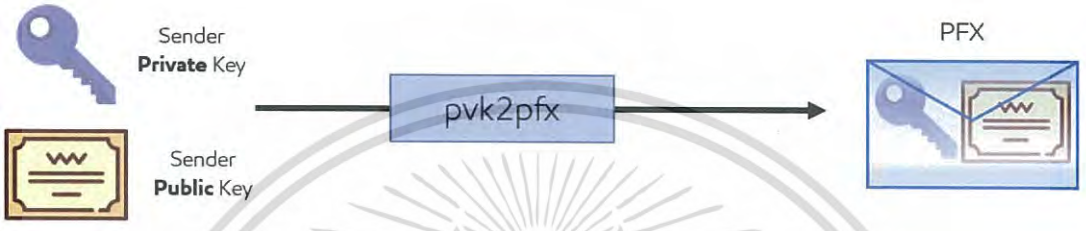
ภาพที่ 3.3 ผู้ส่งเตรียมการส่งไฟล์ต้นฉบับเอาไว้

ภาพที่ 3.4 ผู้ส่งทำการแฮชไฟล์ต้นฉบับอีกอันหนึ่ง จะได้เป็นข้อความไดเจส

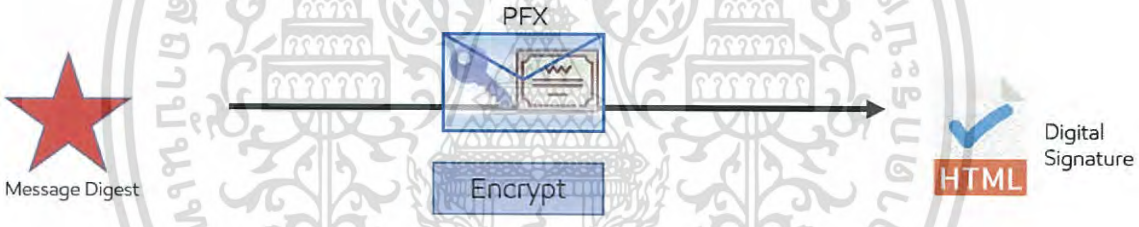


ภาพที่ 3.5 ผู้ส่งสร้างกุญแจส่วนตัวและกุญแจสาธารณะ (ใบรับรอง) ด้วยโปรแกรมเมคเสิท

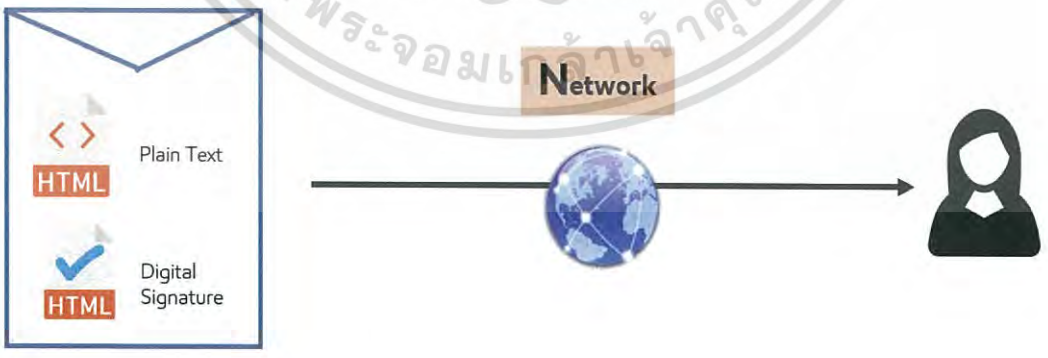
หลังจากที่ผู้ส่งสร้างกุญแจสาธารณะและใบรับรอง ผู้ส่งจะรวมกุญแจสาธารณะและกุญแจส่วนตัวเป็นไฟล์เดียวกันเรียกว่า ไฟล์พีเอฟเอกซ์ โดยใช้โปรแกรมพีวีเคทูพีเอฟเอกซ์ เรียกคำสั่งผ่านคอมมานไลน์ ดังภาพที่ 3.6 จากนั้นนำข้อความไดเจสมาเข้ารหัสหรือลงลายมือดิจิทัลด้วยไฟล์พีเอฟเอกซ์จะได้ลายมือดิจิทัลมา ดังภาพที่ 3.7 เป็นอันเสร็จสมบูรณ์ในการลงลายมือดิจิทัล ขั้นตอนต่อไปคือผู้ส่งส่งไฟล์ทั้งสองไปยังผู้รับโดยอาจจะผ่านเครือข่ายอินเทอร์เน็ตเป็นต้น ดังภาพที่ 3.8



ภาพที่ 3.6 การรวมกุญแจส่วนตัวและใบรับรองไว้ในไฟล์เดียวกันผ่านโปรแกรมพีวีเคทูพีเอฟเอกซ์

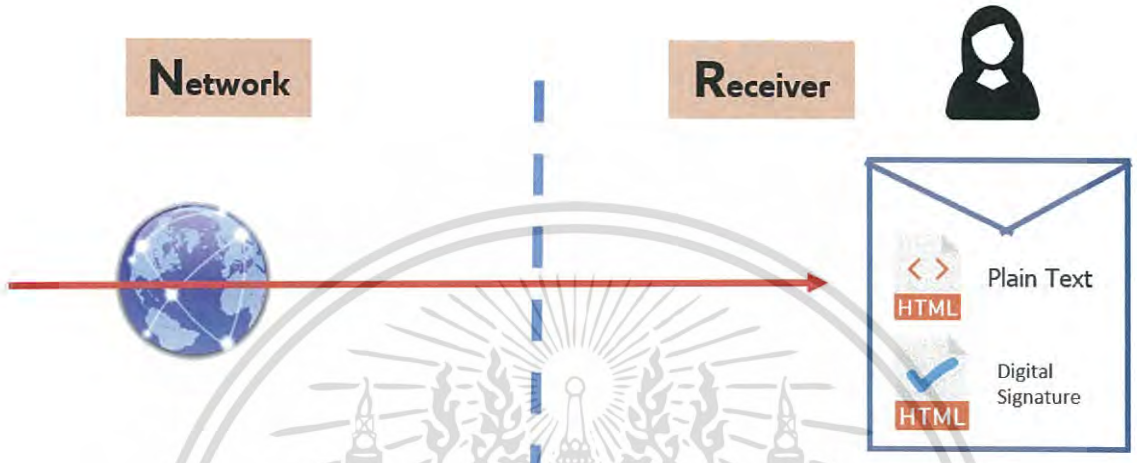


ภาพที่ 3.7 การเข้ารหัสหรือลงลายมือดิจิทัลบนข้อความไดเจสด้วยไฟล์พีเอฟเอกซ์ได้เป็นลายมือดิจิทัล

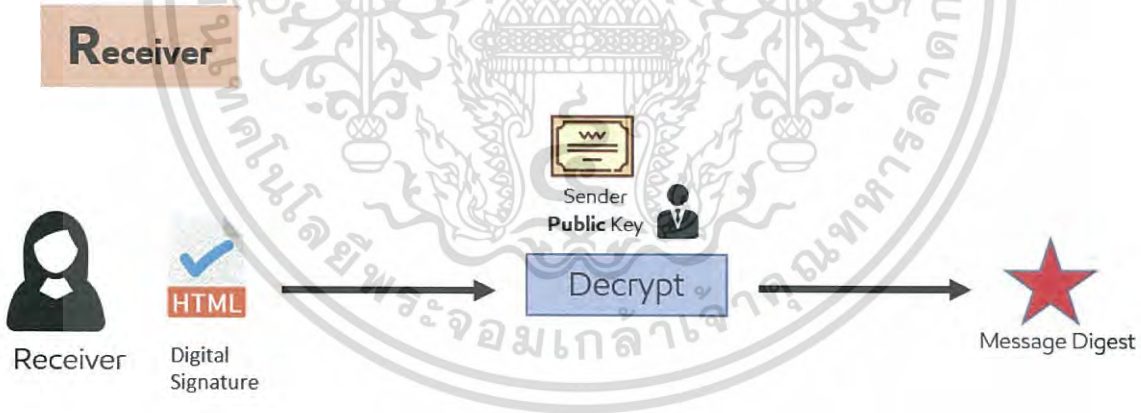


ภาพที่ 3.8 ภาพการส่งไฟล์ทั้งสองไปยังผู้รับ

ในด้านผู้รับ ผู้รับจะได้รับไฟล์ต้นฉบับและดิจิทัลซิกเนเจอร์ที่ส่งผ่านเครือข่ายมาจากผู้ส่ง ดังภาพที่ 3.9 ผู้รับนำลายมือดิจิทัลมาถอดรหัสด้วยกุญแจสาธารณะของผู้ส่งจะได้เป็นข้อความไดเจส ดังภาพที่ 3.10 เพื่อที่จะเตรียมการตรวจสอบว่าไฟล์โค้ดได้รับการแก้ไขมาก่อนหรือไม่



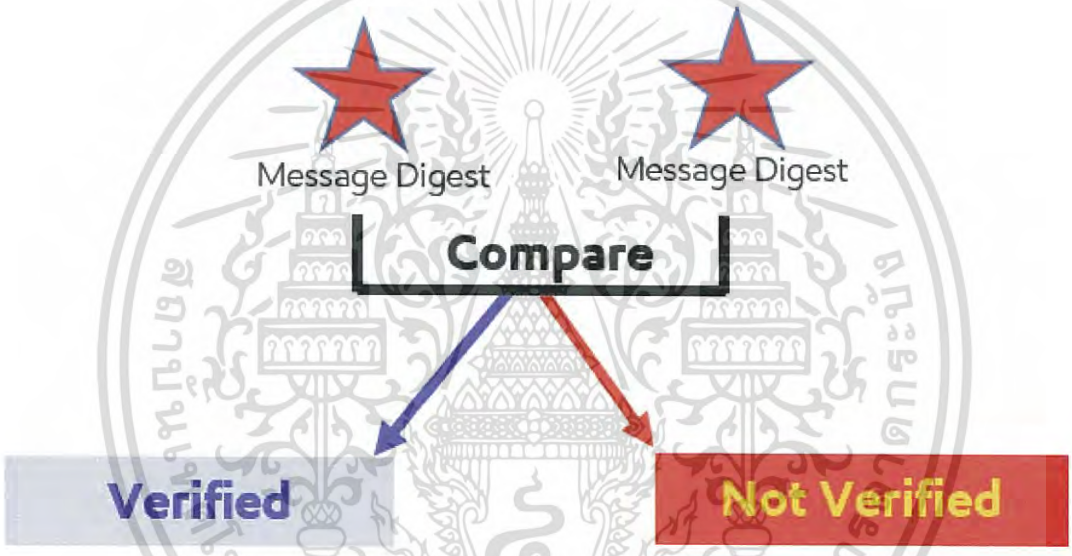
ภาพที่ 3.9 ผู้รับได้รับไฟล์มาจากฝั่งผู้ส่ง



ภาพที่ 3.10 ผู้รับนำลายมือดิจิทัลมาถอดรหัสด้วยกุญแจสาธารณะของผู้ส่งจะได้เป็นข้อความไดเจส

ผู้รับนำไฟล์โค้ดต้นฉบับมาทำการแฮชโดยใช้การแฮชแบบทางเดียวจะได้เป็นข้อความไดเจส ดังภาพที่ 3.11 จากนั้นในขั้นตอนสุดท้าย ผู้รับนำข้อความไดเจสมาเปรียบเทียบกัน ถ้ามีความเหมือนกัน สามารถตรวจสอบความเป็นจริงได้ (Verify) หากไม่เหมือนกัน โค้ดชุดนั้นอาจจะถูกการแก้ไข โจรกรรม และไม่มีความน่าเชื่อถือ ดังภาพที่ 3.12

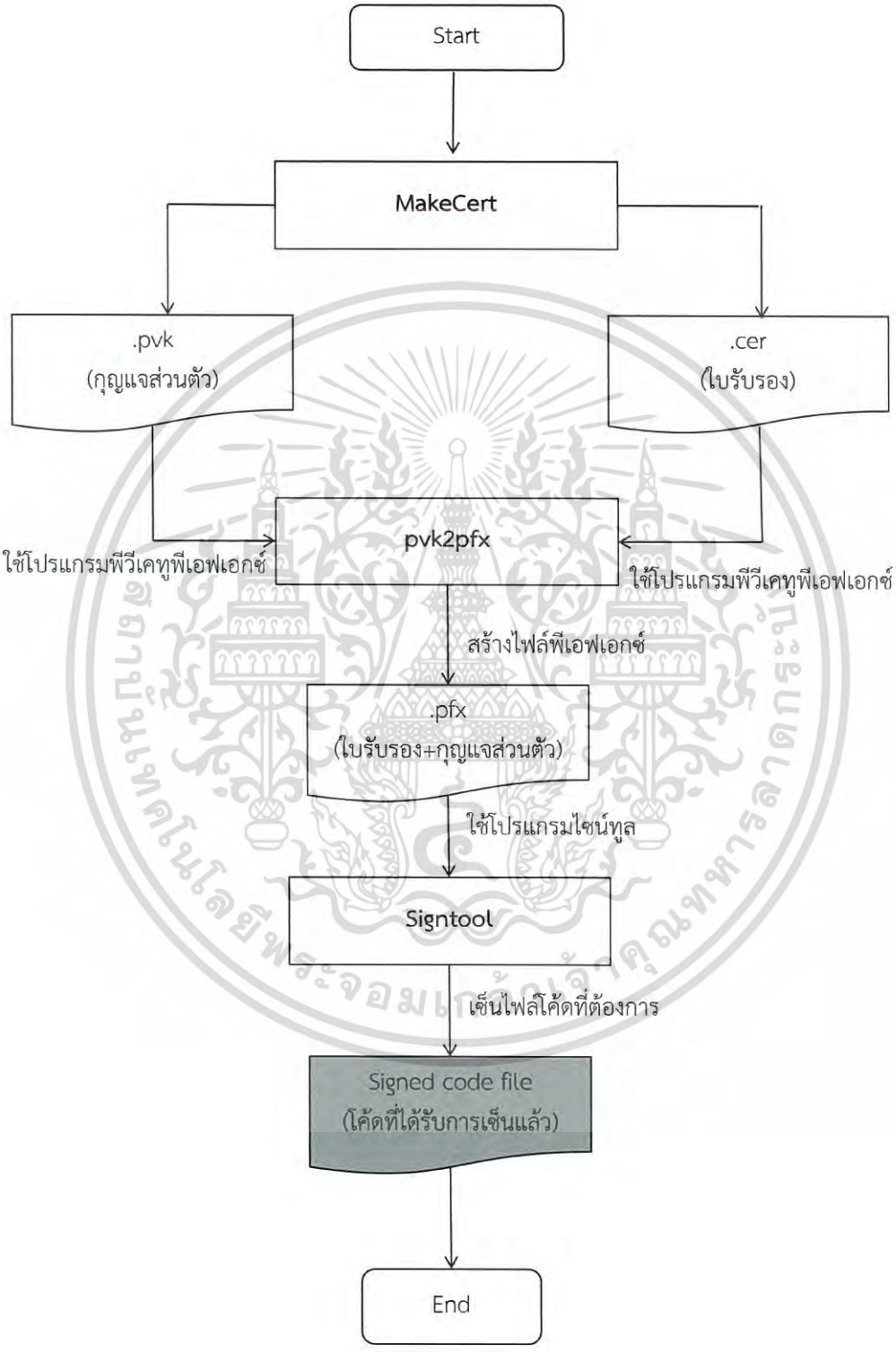
ภาพที่ 3.11 ผู้รับนำข้อความต้นฉบับมาแฮชจะได้ข้อความไดเจส



ภาพที่ 3.12 ผู้รับนำข้อความไดเจสมาเปรียบเทียบกัน

ภาพที่ 3.13 เป็นแผนผังการทำงานตั้งแต่ต้นจนจบของการลงลายมือดิจิทัล เริ่มจากใช้โปรแกรมเมคเสิทในการสร้างกุญแจส่วนตัวและใบรับรองโดยเรียกคำสั่งผ่านคอมมานไลน์ จากนั้นนำไปไฟล์ทั้งสองที่สร้างขึ้นมานั้นก็คือ กุญแจส่วนตัวและกุญแจสาธารณะหรือที่เรียกว่าใบรับรอง นำมารวมกันเป็นไฟล์เดียวกันที่เรียกว่าไฟล์พีเอฟเอกซ์โดยใช้โปรแกรมพีวีเคทูพีเอฟเอกซ์ จากนั้นเป็นการลงลายมือดิจิทัลโดยการนำไฟล์โค้ดที่ต้องการมาลงลายมือดิจิทัล และเรียกใช้โปรแกรมไซน์ทูลในการลงลายมือดิจิทัล จะได้ไฟล์โค้ดที่มีลายมือดิจิทัล สามารถทำไปตรวจสอบต่อได้ว่าโค้ดชุดนั้นได้รับการเปลี่ยนแปลงแก้ไขหรือไม่

แผนผังการลงลายมือดิจิทัลผ่านโปรแกรมต่าง ๆ ดังภาพที่ 3.13



ภาพที่ 3.13 แผนผังการทำงานการทำลายมือดิจิทัล

3.2.1.2 ทำคู่มือการลงลายมือดิจิทัลให้กับไฟล์โค้ด

เมื่อทำการทดลองเสร็จเรียบร้อยแล้ว จึงนำไปจัดทำเป็นเอกสารคู่มือขั้นตอนการทำการลงลายมือดิจิทัลลงในไฟล์โค้ดหรือโค้ดไซน์นิงเป็นภาษาอังกฤษโดยใช้โปรแกรมไมโครซอฟท์ เวิร์ด ให้กับบริษัท โดยจะอธิบายตั้งแต่โปรแกรมที่ต้องใช้ คำสั่งที่ต้องใช้ในการสร้าง ข้อจำกัดของโปรแกรม แนวทางแก้ไขของโปรแกรม เพื่อให้ผู้อื่นที่สนใจจากทั่วโลก นำไปพัฒนาต่อในระยะยาวได้

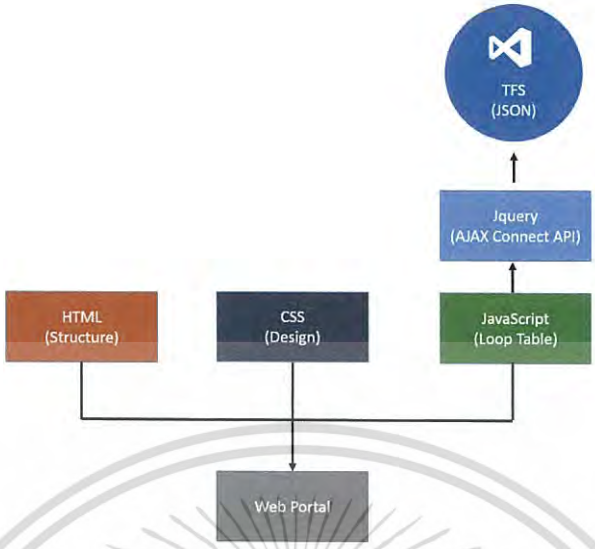
3.2.2 เว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

ภาพที่ 3.14 ภาพการทำเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

จากภาพที่ 3.14 ในส่วนนี้จะเป็นการออกแบบหน้าเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ ในส่วนการจัดการรีรีส ในโปรแกรมทีเอฟเอส มีขั้นตอนการทำงานดังนี้

3.2.2.1 ศึกษา เรียนรู้ ทดลองการทำงานเบื้องต้นของแต่ละภาษา

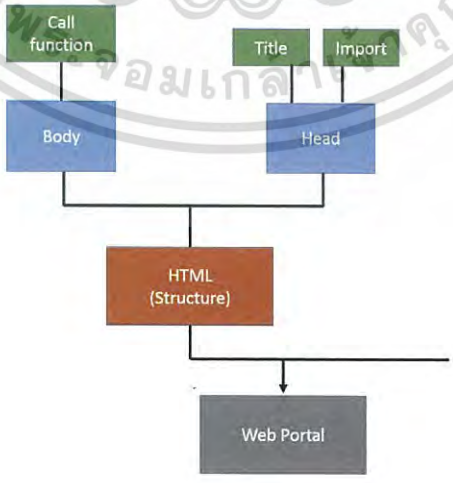
3.2.2.2 นำภาษาต่าง ๆ ให้มาทำงานร่วมกัน



ภาพที่ 3.15 ภาพการทำงานโดยรวมเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

จากภาพที่ 3.15 ภาพการทำงานโดยรวมเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ มีการใช้ทำงานร่วมกันของหลายภาษาในการสร้างหน้าเว็บนี้ขึ้นมา มีการใช้ภาษาเอชทีเอ็มแอลในการกำหนดโครงสร้างและแสดงผลหน้าเว็บ ภาษาซีเอสเอสในการตกแต่งหน้าเว็บเช่น ฟอนต์ สีฟอนต์ สีพื้นหลังต่างๆ ภาษาจาวาสคริปต์เพื่อการสร้างฟังก์ชันต่าง ๆ ที่ต้องใช้ในการสร้างหน้าเว็บให้เว็บดูมีความน่าสนใจมากขึ้น ใช้ภาษาเจคิวรีในการเชื่อมต่อเอพีไอเพื่อดึงข้อมูลจากโปรแกรมทีเอฟเอส ซึ่งข้อมูลที่เก็บอยู่ในโปรแกรมทีเอฟเอส ถูกจัดเก็บอยู่ในรูปแบบภาษาเจสัน

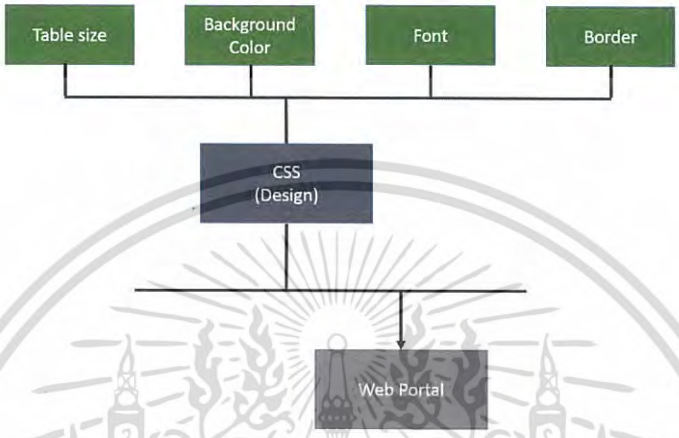
1) ภาษาเอชทีเอ็มแอล



ภาพที่ 3.16 ภาพการทำงานของภาษาเอชทีเอ็มแอล

จากภาพที่ 3.16 ภาษาเอชทีเอ็มแอล ใช้ในการวางโครงสร้างให้หน้าเว็บประกอบด้วยส่วนบอดี ซึ่งจะใช้เรียกฟังก์ชันหรือแสดงผลต่าง ๆ บนหน้าเว็บ และส่วนเฮดจะแสดงชื่อของเว็บไซต์ (Title) และอิมพอร์ตการเรียกใช้ภาษาอื่น ๆ ร่วมกับภาษาเอชทีเอ็มแอล

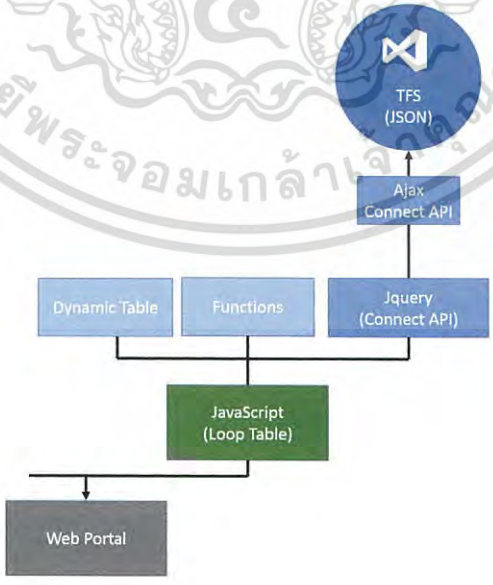
2) ภาษาซีเอสเอส



ภาพที่ 3.17 ภาพการทำงานของภาษาซีเอสเอส

จากภาพที่ 3.17 การทำงานของภาษาซีเอสเอสใช้ในการตกแต่งหน้าเว็บเพจ ใส่ความหนา ใส่สี ตัวอักษร ฟอนต์ต่าง ๆ ให้กับตารางและส่วนอื่น ๆ เพื่อให้หน้าเว็บเพจมีความสวยงามมากขึ้น

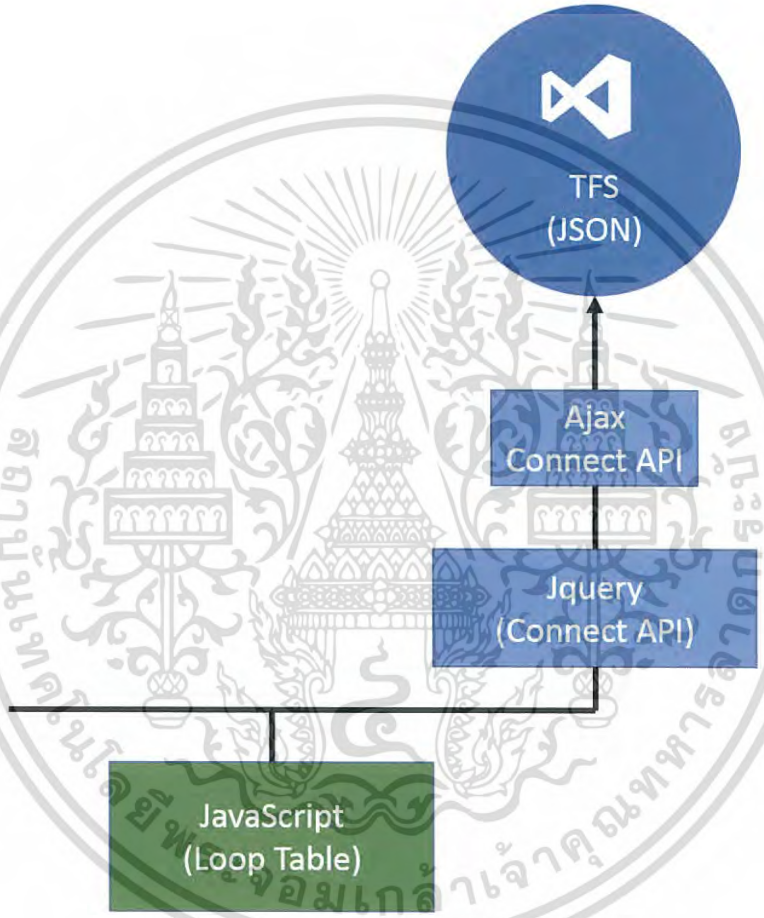
3) ภาษาจาวาสคริปต์



ภาพที่ 3.18 ภาพการทำงานของภาษาจาวาสคริปต์

จากภาพที่ 3.18 ภาษาจาวาสคริปต์ ทำหน้าที่ประมวลผลต่าง ๆ เพื่อที่จะนำไปแสดงผลในหน้าของเอชทีเอ็มแอล มีการใช้ภาษาเจคิวรีเพื่อเชื่อมต่อไปยังโปรแกรมทีเอฟเอส มีการเรียกฟังก์ชันสร้างตารางแบบไดนามิกเพื่อนำข้อมูลที่ได้กลับมาจากโปรแกรมทีเอฟเอส แสดงผลในตาราง

4) เจคิวรีเฟรมเวิร์ค



ภาพที่ 3.19 ภาพการทำงานของเจคิวรีเฟรมเวิร์ค

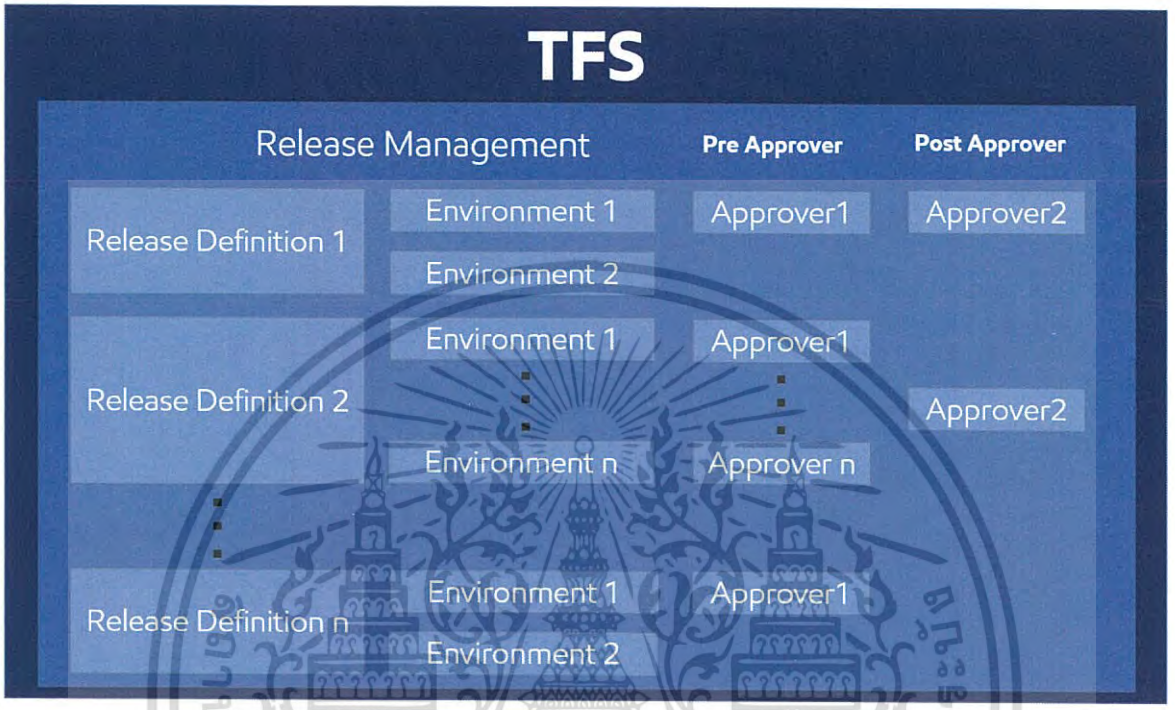
จากภาพที่ 3.19 เจคิวรีเฟรมเวิร์ค จะถูกเขียนอยู่ในไฟล์จาวาสคริปต์ มีการเรียกใช้ฟังก์ชันเอแจ็ค เพื่อการเชื่อมต่อเอพีไอ รับส่งข้อมูลไปยังโปรแกรมทีเอฟเอส ซึ่งข้อมูลในโปรแกรมทีเอฟเอสจะถูกเก็บอยู่ในรูปแบบเจสัน และจะใช้ภาษาจาวาสคริปต์ในการจัดการกับข้อมูลเจสันให้แสดงผลออกมาเป็นตาราง

ภาพที่ 3.20 เป็นแผนผังการทำงานจะเริ่มจากเข้าไปในโปรแกรมทีเอฟเอส แล้วเลือกในหมวดหมู่ของการจัดการรีรีส คลิกไปที่ลิงค์ที่ได้สร้างขึ้นไว้ จะแสดงรายชื่อของรีรีสเดฟฟินิชันทั้งหมด พร้อมทั้งสภาพแวดล้อมและรายชื่อผู้อนุมัติทั้งหมด หากไม่มีผู้อนุมัติในรีรีสเดฟฟินิชันใด สามารถคลิกไปที่ชื่อรีรีสเดฟฟินิชันนั้น ๆ เพื่อลิงค์ไปยังหน้าการตั้งค่าในแต่ละรีรีสเดฟฟินิชัน

ภาพที่ 3.20 แผนผังการทำงานของเว็บ

3.2.2.3 สร้างหน้าเว็บแสดงรายชื่อผู้อนุมัติ

เมื่อศึกษาเสร็จเรียบร้อยแล้ว ก็เริ่มพัฒนาสร้างเว็บแสดงรายชื่อผู้อนุมัติ



ภาพที่ 3.21 ภาพการทำงานของเว็บ

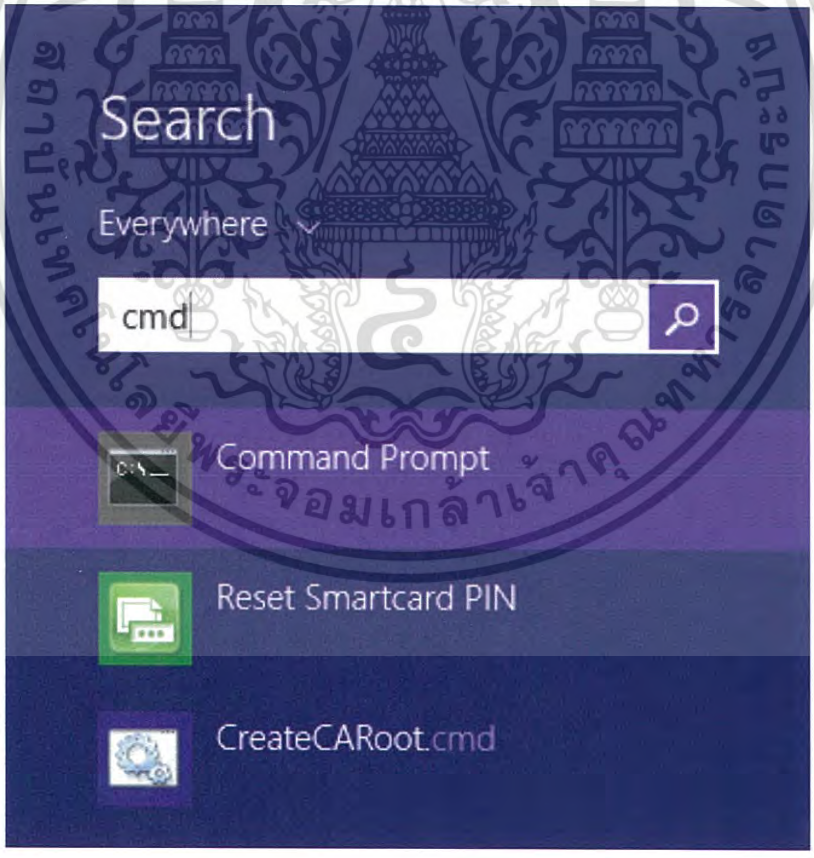
จากภาพที่ 3.21 การทำงานก็คือในแต่ละรีรีสเดฟฟินิชัน จะมีได้หลายสภาพแวดล้อม ซึ่งการจะส่งไฟล์โค้ดหรือซอฟต์แวร์ผ่านไปแต่ละสภาพแวดล้อม ต้องมีการตั้งค่าผู้อนุมัติ ซึ่งหน้าเว็บที่ได้พัฒนามานี้จะช่วยให้การตรวจสอบรายชื่อของผู้อนุมัติเป็นไปได้ง่ายขึ้น โดยจะแยกรีรีสเดฟฟินิชันที่มีผู้อนุมัติและไม่มีผู้อนุมัติออกจากกัน อีกทั้งยังสามารถคลิกไปบนรายชื่อของรีรีสเดฟฟินิชันเพื่อเชื่อมต่อไปยังหน้าที่ตั้งค่าได้ทันที

บทที่ 4
ผลการวิจัย

จากการทำวิจัยตลอดจนสิ้นสุดโครงการสหกิจศึกษานี้ ผลการวิจัยที่ได้สามารถแสดงให้เห็นถึงขั้นตอนการทำงานต่าง ๆ ของโปรแกรมที่ใช้ในการสร้างใบรับรอง กุญแจสาธารณะ กุญแจส่วนตัว โปรแกรมสร้างลายมือดิจิทัล รูปร่างหน้าตาของไฟล์โค้ดที่ได้รับการลงลายมือดิจิทัล (Signed file) การตรวจสอบไฟล์ที่มีลายมือดิจิทัล (Verify) อีกทั้งยังได้แสดงเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบได้อีกด้วย

4.1 การลงลายมือดิจิทัลในไฟล์โค้ด

ในส่วนแรกนี้จะเริ่มต้นจากการสร้างใบรับรองและกุญแจส่วนตัวผ่านโปรแกรมเมคเสิท เพื่อที่จะนำไปใช้ในการลงลายมือดิจิทัลให้กับไฟล์โค้ดที่ต้องการ โดยมีขั้นตอนดังต่อไปนี้ ขั้นตอนที่หนึ่งเปิดคอมมานด์พรอมต์ขึ้นมา กดปุ่มวินโดว์บนคีย์บอร์ด แล้วค้นหาคำว่าซีเอ็มดี (CMD) ดังภาพที่ 4.1



ภาพที่ 4.1 ค้นหาคำว่าซีเอ็มดี เพื่อเรียกคอมมานด์พรอมต์

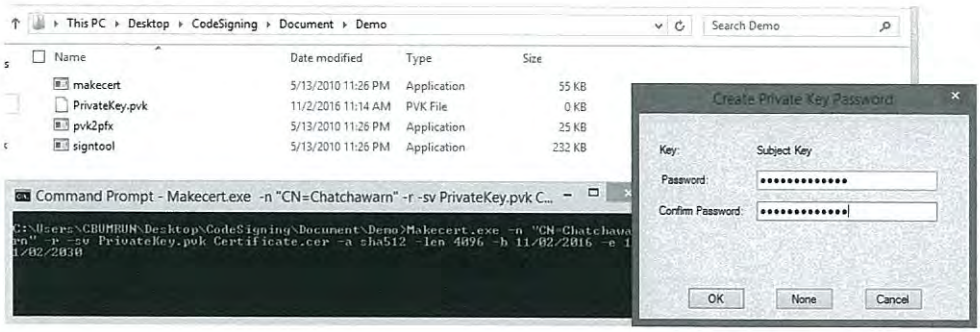
ขั้นตอนที่สอง เรียกคำสั่งลงในคอมมานไลน์เพื่อสร้างใบรับรองและกุญแจส่วนตัว ดังภาพที่ 4.2
ตัวอย่างคำสั่งที่เรียกใช้

Makecert.exe -n "CN=Chatchawarn" -r -sv PrivateKey.pvk Certificate.cer -a sha512-len 4096

คำอธิบายคำสั่งในแต่ละส่วน ดังตารางที่ 4.1

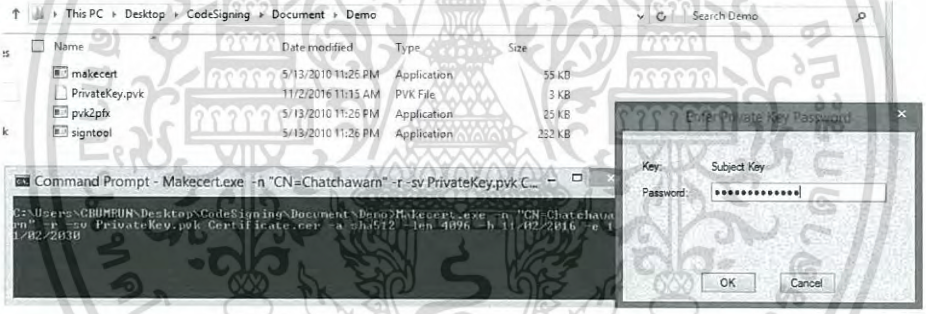
ตารางที่ 4.1 ตารางอธิบายตัวอย่างคำสั่งในโปรแกรมเมคเสิท

คำสั่ง	ความหมาย
Makecert.exe	เรียกโปรแกรมเมคเสิท
CN=Chatchawarn	ชื่อของผู้สร้างเช่น “Chatchawarn”
-r	กำหนดให้เป็นการสร้างใบรับรองด้วยตนเอง
-sv PrivateKey.pvk	ชื่อของกุญแจส่วนตัว เช่น “PrivateKey.pvk”. กุญแจส่วนตัวจะถูกสร้างหลังจากคำสั่งนี้ทำงาน
Certificate.cer	ชื่อของใบรับรอง เช่น “Certificate.cer”ใบรับรองจะถูกสร้างหลังคำสั่งนี้ทำงาน
-a sha512	อัลกอริทึมในการเข้ารหัส
-len 4096	ความยาวของบิท
-b 11/02/2016	วันออกใบรับรอง
-e 11/02/2030	วันหมดอายุของใบรับรอง

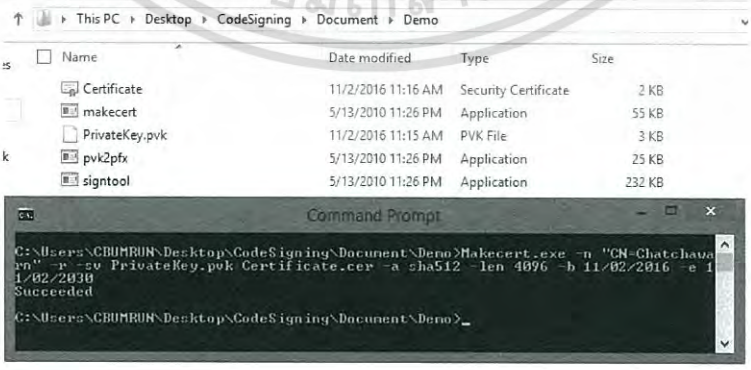


ภาพที่ 4.2 การสร้างรหัสผ่านให้กุญแจส่วนตัว

ขั้นตอนที่สาม ใส่รหัสผ่านเพื่อยืนยันการสร้างกุญแจส่วนตัว ดังภาพที่ 4.3 เพื่อเอารหัสผ่านนี้ไปใช้ในการยืนยันตัวตนระหว่างการลงลายมือดิจิทัล จะได้ไฟล์ขึ้นมาสองชนิดคือไฟล์นามสกุลพีวีเค (.pvk) และไฟล์ใบรับรองที่มีนามสกุลเซอร์ (.cer) เมื่อสร้างใบรับรองและกุญแจสำเร็จจะได้ดังภาพที่ 4.4



ภาพที่ 4.3 ภาพใส่รหัสเพื่อยืนยันกุญแจส่วนตัว



ภาพที่ 4.4 ใบรับรองและกุญแจส่วนตัวถูกสร้างเสร็จเรียบร้อย

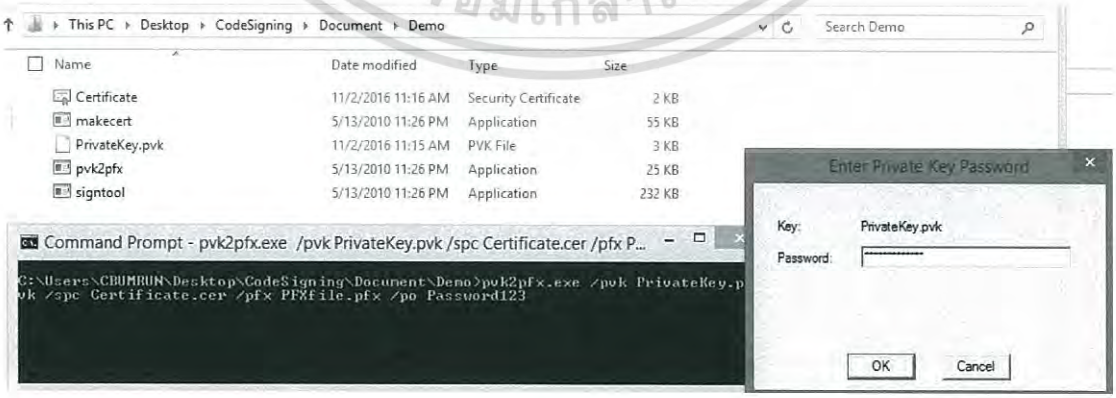
ขั้นตอนที่สี่ รวมกุญแจส่วนตัวและใบรับรองไว้ในไฟล์เดียวกันเรียกว่าไฟล์พีเอฟเอกซ์เพื่อให้ถูกรูปแบบในการลงลายมือดิจิทัลตามที่โปรแกรมไซน์ทูลกำหนด ดังภาพที่ 4.5 และไฟล์พีเอฟเอกซ์ จะถูกสร้างขึ้น ดังภาพที่ 4.6 โดยตัวอย่างคำสั่งที่เรียกใช้คือ

pvk2pfx.exe /pvk PrivateKey.pvk /spc Certificate.cer /pfx PFXfile.pfx /po Password123

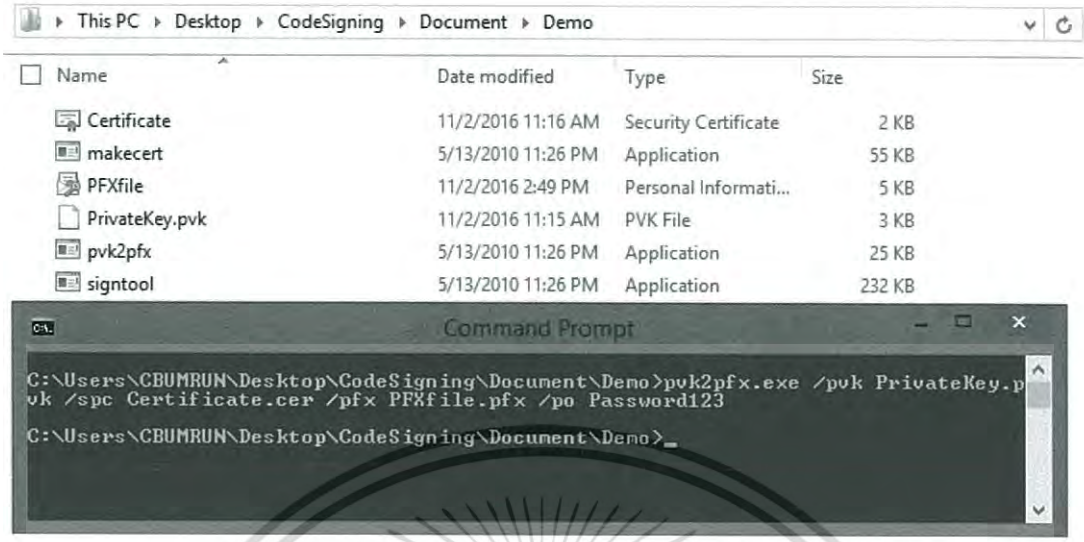
คำอธิบายคำสั่งในแต่ละส่วน ดังตารางที่ 4.2

ตารางที่ 4.2 ตารางอธิบายตัวอย่างคำสั่งในโปรแกรมพีวีเคทูพีเอฟเอกซ์

คำสั่ง	ความหมาย
pvk2pfx.exe	เรียกโปรแกรมพีวีเคทูพีเอฟเอกซ์
/pvk PrivateKey.pvk	ชื่อของกุญแจส่วนตัว ในที่นี้คือ “PrivateKey.pvk”
/spc Certificate.cer	ชื่อของใบรับรอง ในที่นี้คือ “Certificate.cer”
/pfx PFXfile.pfx	กำหนดชื่อของไฟล์ .pvk ในที่นี้ไฟล์ชื่อ “PFXFile.pfx” จะถูกสร้างหลังเรียกคำสั่ง
/po Password123	กำหนดรหัสผ่านสำหรับไฟล์ .pfx. ในที่นี้รหัสผ่านคือ “Password123”



ภาพที่ 4.5 ใส่รหัสผ่านของกุญแจส่วนตัว



ภาพที่ 4.6 ไฟล์พีเอฟเอกซ์ถูกสร้างขึ้น

ขั้นตอนแรกของการลงลายมือดิจิทัลบนไฟล์โค้ดด้วยโปรแกรมไซน์ทูลเรียกใช้คำสั่งผ่านคอมมานไลน์ ดังภาพที่ 4.7 ให้จัดเตรียมไฟล์โค้ดที่ต้องการลงลายมือดิจิทัลเอาไว้ โดยตัวอย่างที่เรียกใช้คือ

SignTool.exe sign /f PFXfile.pfx /p Password123 /t http://timestamp.verisign.com/scripts/timstamp.dll file1.ps1

คำอธิบายคำสั่งในแต่ละส่วน ดังตารางที่ 4.3

ตารางที่ 4.3 ตารางอธิบายตัวอย่างคำสั่งไซน์ในโปรแกรมไซน์ทูล

คำสั่ง	ความหมาย
SignTool.exe	เรียกโปรแกรมไซน์ทูล
Sign	คำสั่งให้ลงลายมือดิจิทัล
/f PFXfile.pfx	ชื่อไฟล์ PFX ในที่นี้คือ “PFXFile.pfx”
/p Password123	รหัสผ่านสำหรับไฟล์ PFX ในที่นี้คือ “Password123”

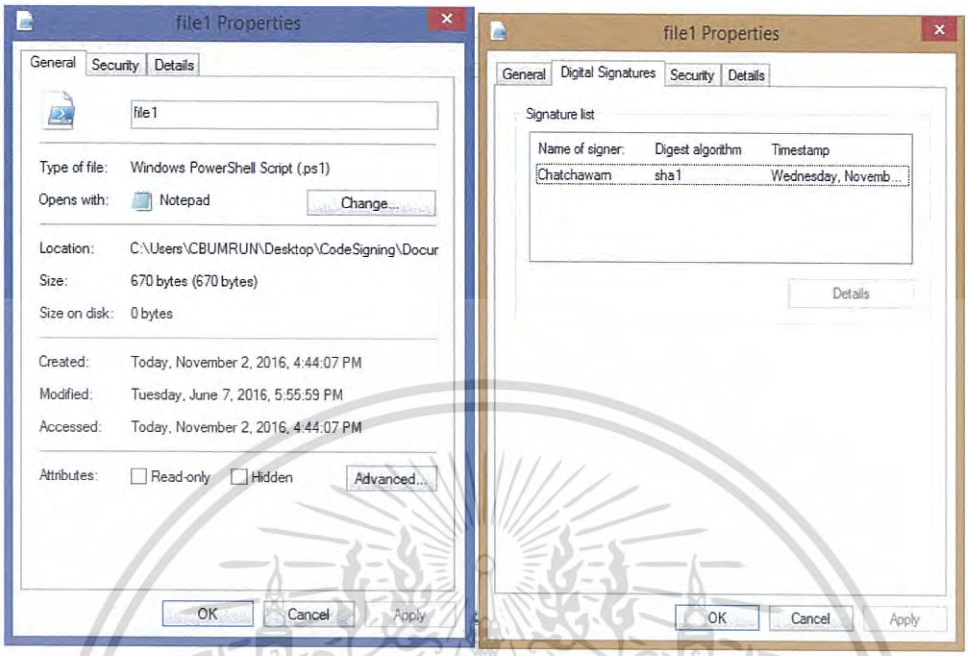
ตารางที่ 4.3 ตารางอธิบายตัวอย่างคำสั่งไซน์ในโปรแกรมไซน์ทูล (ต่อ)

คำสั่ง	ความหมาย
/t http://timestamp.verisign.com /scripts/timstamp.dll	ไทม์สแตมป์เซิร์ฟเวอร์ในที่นี้เป็นเซิร์ฟเวอร์ตัวอย่าง
File1.ps1	ไฟล์โค้ดที่ต้องการลงลายมือดิจิทัล ในที่นี้คือ “File1.ps1”

หลังจากที่เรียกคำสั่งเสร็จสมบูรณ์ จะขึ้นข้อความว่าสำเร็จ แสดงว่าไฟล์โค้ดที่เราต้องการลงลายมือดิจิทัลได้ลงลายมือเสร็จเรียบร้อยสมบูรณ์แล้ว

ภาพที่ 4.7 ไฟล์ได้รับการลงลายมือดิจิทัลสมบูรณ์

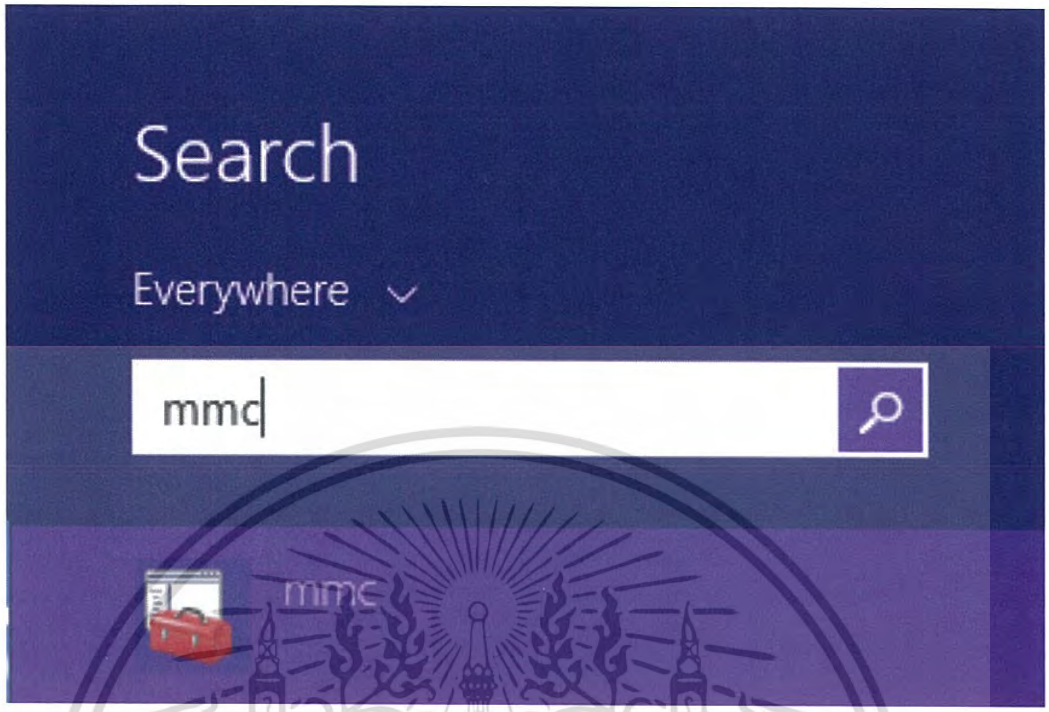
การตรวจสอบไฟล์โค้ดว่าพบลายมือดิจิทัลหรือไม่ ดังภาพที่ 4.8 หากไฟล์โค้ดของมีการลงลายมือดิจิทัลเอาไว้ จะมีแท็บดิจิทัลซิกเนเจอร์ปรากฏขึ้นมา หากไม่ได้ลงลายมือดิจิทัลเอาไว้ในไฟล์โค้ดก็จะไม่มีแท็บดิจิทัลซิกเนเจอร์ปรากฏขึ้นมาและถ้าในกรณีของไฟล์โค้ดสามารถตรวจสอบได้ ดังภาพที่ 4.9 หากมีการลงลายมือดิจิทัลเอาไว้เมื่อเลื่อนลงไปล่างสุดข้างหน้าโค้ดจะพบว่ามีภาษาที่อ่านไม่ออก เรียกว่าดิจิทัลซิกเนเจอร์



ภาพที่ 4.8 ไฟล์โค้ดที่ไม่ได้รับการลงลายมือดิจิทัลและไฟล์โค้ดที่ได้รับการลงลายมือดิจิทัล

ภาพที่ 4.9 กรณีของในไฟล์โค้ดว่ามีดิจิทัลซิกเนเจอร์ปรากฏหรือไม่

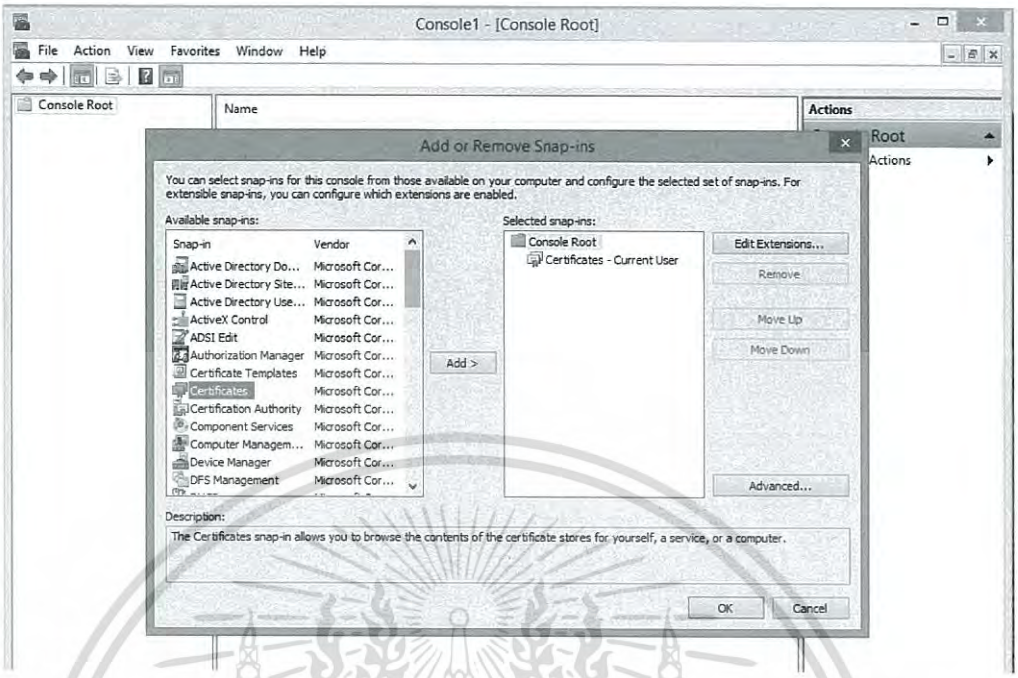
การตรวจสอบดิจิทัลซิกเนเจอร์ ขั้นตอนแรกคือติดตั้งใบรับรองในทรัสต์รูต (Trust Root) กดสตาร์ทแล้วพิมพ์คำว่า เอ็มเอ็มซี (MMC) ดังภาพที่ 4.10 ไปที่ไฟล์ จากนั้นไปที่ไฟล์ เลือกที่แอดรีมูฟสแน็ปอิน (Add/Remove Snap-in) ดังภาพที่ 4.11



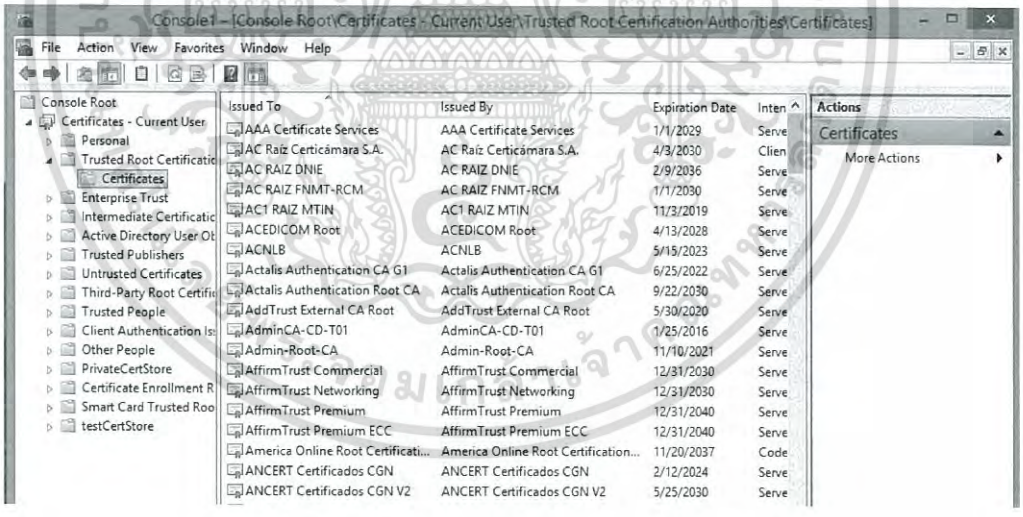
ภาพที่ 4.10 ค้นหาคำว่าเอ็มเอ็มซี

ภาพที่ 4.11 ไปที่ไฟล์ แล้วคลิกที่แอดรีมูฟสแน็ปอิน

จากนั้นเลือกใบรับรอง แล้วแอดลงในสแน็ปอิน ดังภาพที่ 4.12 ไปที่ทรัสต์รูตเซอร์ทิฟิเคชัน (Trusted Root Certification) แล้วเลือกที่ใบรับรอง (Certificates) ดังภาพที่ 4.13 จะเข้ามาสู่หน้าของใบรับรองทั้งหมดที่มีในคอมพิวเตอร์

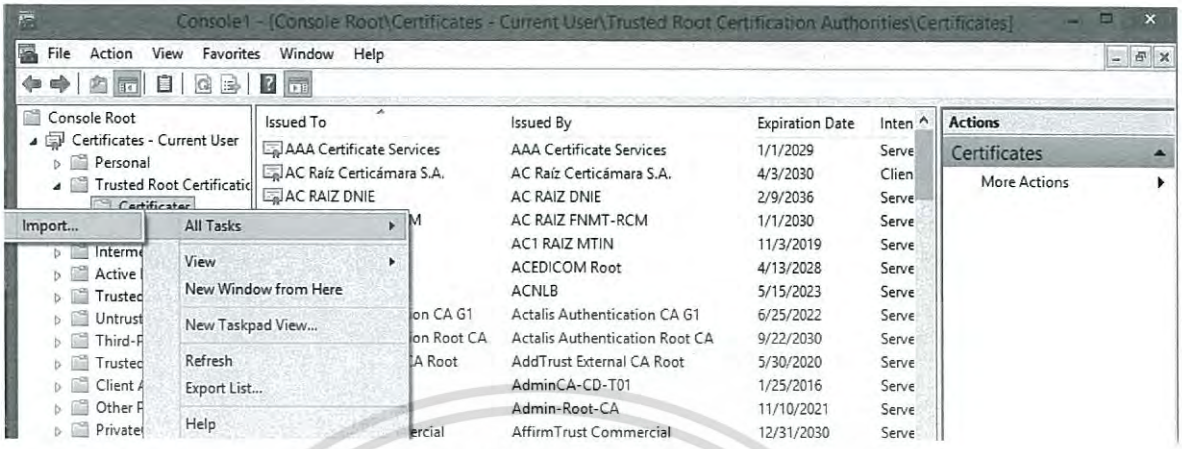


ภาพที่ 4.12 เลือกใบรับรอง แล้วเพิ่มลงในสแน็ปอิน

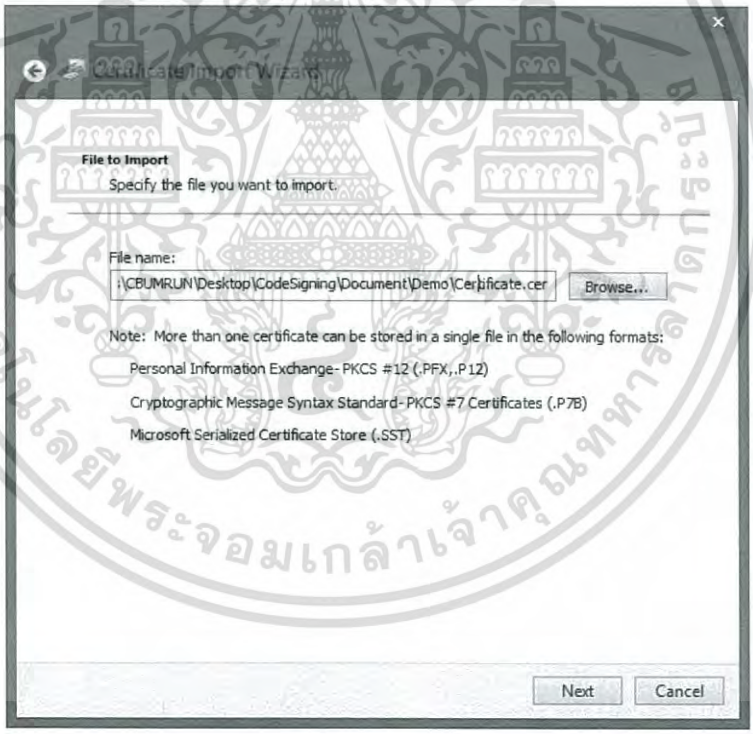


ภาพที่ 4.13 ไปที่ทรัสต์รูตเซอร์ทิฟิเคชัน (Trusted Root Certification) แล้วเลือกที่ใบรับรอง (Certificates)

ขั้นตอนต่อไปคลิปขวาที่ใบรับรอง (Certificates) เลือก ทาสก์ทั้งหมด แล้วกดอิมพอร์ต (Import) ดังภาพที่ 4.14 จะมีหน้าต่างขึ้นมาให้เลือกใบรับรองที่ต้องการติดตั้งในที่ทรัสต์รูตเซอร์ทิฟิเคชัน ดังภาพที่ 4.15

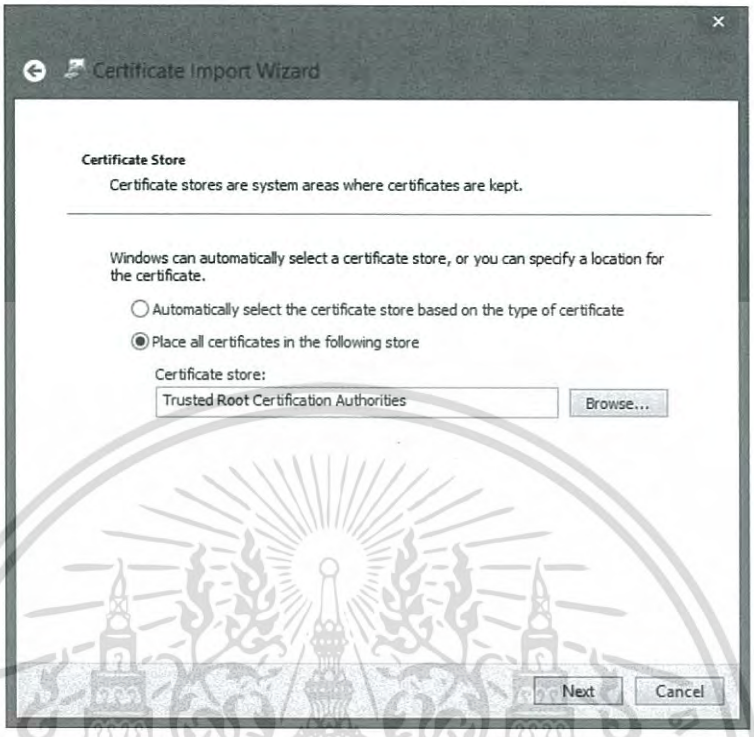


ภาพที่ 4.14 คลิปขวาที่ใบรับรอง (Certificates) เลือก ทาสก์ทั้งหมด แล้วกดอิมพอร์ต (Import)

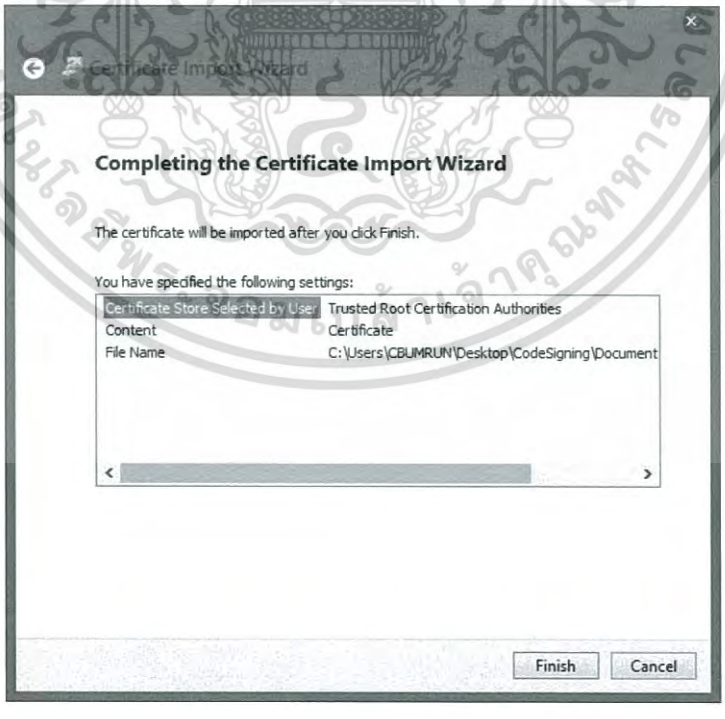


ภาพที่ 4.15 หน้าต่างเลือกใบรับรองที่เก็บไว้ในเครื่อง

กำหนดที่เก็บใบรับรองไว้ที่ทรัสต์รูตเซอร์ทิฟิเคชันเพื่อที่ต้องการติดตั้งใบรับรอง แล้วกดต่อไป ดังภาพที่ 4.16 จากนั้นจะมีหน้าต่างแสดงขึ้นมาเพื่อให้ยืนยันว่าที่อยู่ในการติดตั้งใบรับรองถูกต้องหรือไม่ ประเภท และไฟล์ใบรับรอง จากนั้นจึงกดเสร็จสิ้น (Finish) ดังภาพที่ 4.17

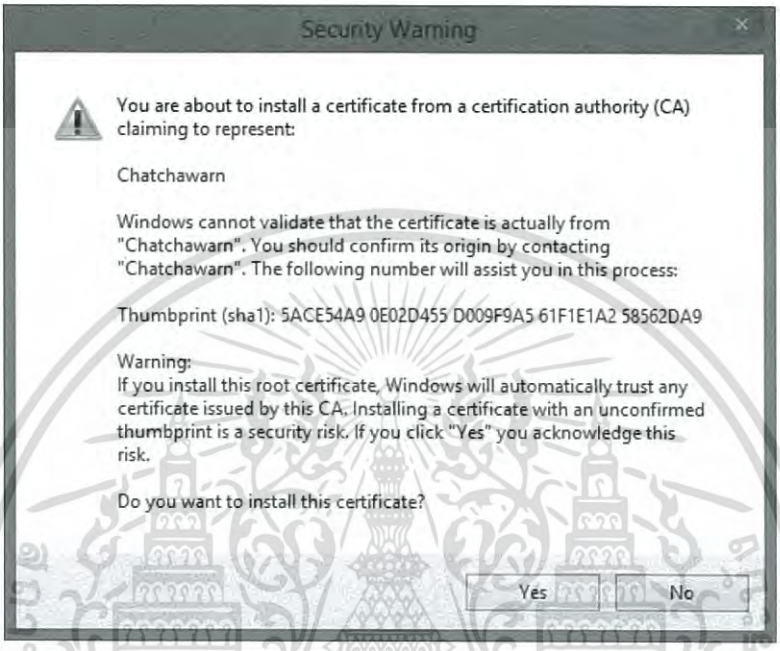


ภาพที่ 4.16 หน้าต่างเลือกที่เก็บใบรับรอง

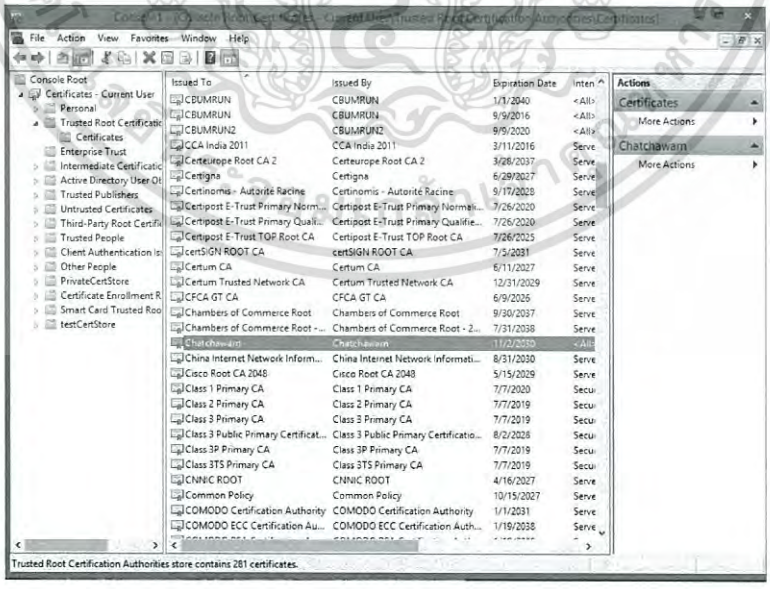


ภาพที่ 4.17 การติดตั้งใบรับรองเสร็จสิ้น

จากนั้นจะมีหน้าต่างให้กดยืนยันอีกรอบ ให้กดใช่ ดังภาพที่ 4.18 ใบรับรองจะถูกติดตั้งในใบรับรองจะถูกติดตั้งในทรัสต์รูตเซอร์ทิฟิเคชันเรียบร้อย สามารถถอนการติดตั้งหรือดูรายละเอียดข้อมูลใบรับรองได้ในทรัสต์รูตเซอร์ทิฟิเคชัน ดังภาพ 4.19



ภาพที่ 4.18 กดใช่เพื่อติดตั้งใบรับรอง



ภาพที่ 4.19 ใบรับรองจะถูกติดตั้งในทรัสต์รูตเซอร์ทิฟิเคชัน

หลังจากได้ทำการติดตั้งใบรับรองลงในทรัสต์รูตเซอร์ทิฟิเคชันแล้ว สามารถเรียกคำสั่งเพื่อตรวจสอบลายมือดิจิทัลในไฟล์โค้ดได้ ผ่านคอมมานไลน์ เมื่อสำเร็จจะได้ดังภาพ 4.20 จะแสดงรายละเอียดของไฟล์โค้ด ใครเป็นผู้ลงลายมือดิจิทัล รูปแบบการเข้ารหัส ตลอดจนถึงวันหมดอายุของใบรับรอง หากสำเร็จจะไม่แสดงค่าแอร์เรอร์และคำเตือน หรือมีค่าเป็น 0 หากไฟล์โค้ดได้รับการเปลี่ยนแปลงหรือแก้ไข จะแสดงค่าอื่นๆ หรือมีแอร์เรอร์เกิดขึ้น ดิจิทัลซิกเนเจอร์จะไม่มีความน่าเชื่อถือ

คำสั่งตัวอย่างที่เรียกใช้

Signtool verify /pa /v file1.ps1

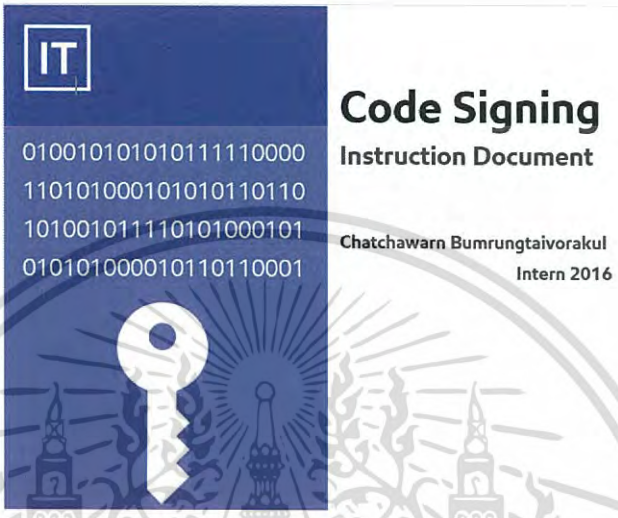
คำอธิบายคำสั่งในแต่ละส่วน ดังตารางที่ 4.4

ตารางที่ 4.4 ตารางอธิบายตัวอย่างคำสั่งตรวจสอบในโปรแกรมไซน์ทูล

คำสั่ง	ความหมาย
Signtool	เรียกโปรแกรมไซน์ทูล
Verify	คำสั่งเพื่อการตรวจสอบ
/pa	ให้โปรแกรมตรวจสอบว่าลายมือดิจิทัลของแต่ละไฟล์ที่ระบุ สอดคล้องกันตามอาร์กิวเมนต์ชื่อไฟล์
/v	แสดงผลลัพธ์การตรวจสอบว่าสำเร็จ ล้มเหลว หรือคำเตือน
File1.ps1	ไฟล์โค้ดที่ต้องการตรวจสอบในที่นี้คือ “File1.ps1”

ภาพที่ 4.20 ตรวจสอบสำเร็จ ข้อมูลถูกต้อง

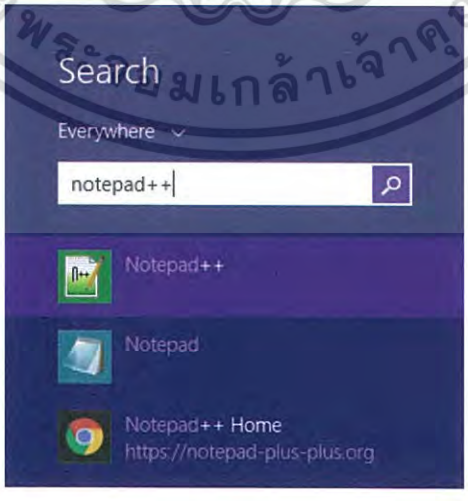
เมื่อทำการทดลองเสร็จเรียบร้อยแล้ว ได้ทำการสร้างเอกสารอธิบายขั้นตอนต่าง ๆ ของการลงลายมือดิจิทัล โดยใช้โปรแกรมไมโครซอฟท์เวิร์ดเป็นภาษาอังกฤษ ให้บริษัท เอ็กซอนโมบิล จำกัด หากมีผู้สนใจศึกษาต่อในอนาคต สามารถอ่านวิธีการทำลายมือดิจิทัลทั้งหมดได้ในเอกสารชุดนี้ ดังภาพ 4.21



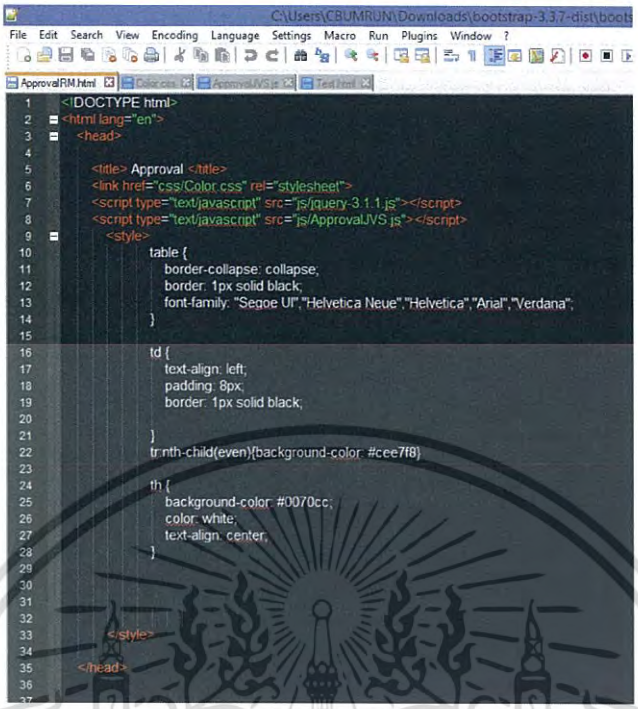
ภาพที่ 4.21 เอกสารขั้นตอนการลงลายมือดิจิทัลที่จัดทำขึ้น

4.2 เว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ

การทำเว็บแสดงรายชื่อผู้อนุมัติการนำโค้ดขึ้นระบบ สามารถทำได้ดังขั้นตอนแรก คือค้นหาและเข้าโปรแกรมโน้ตแพดพลัสพลัส (Notepad++) ดังภาพ 4.22 จากนั้น สร้างไฟล์เอชทีเอ็มแอลเพื่อเป็นโครงสร้างการแสดงผลของเว็บ ดังภาพ 4.23 จะมีคำสั่งต่าง ๆ มากมายเพื่อใช้ในการแสดงผลเป็นภาษาเอชทีเอ็มแอล

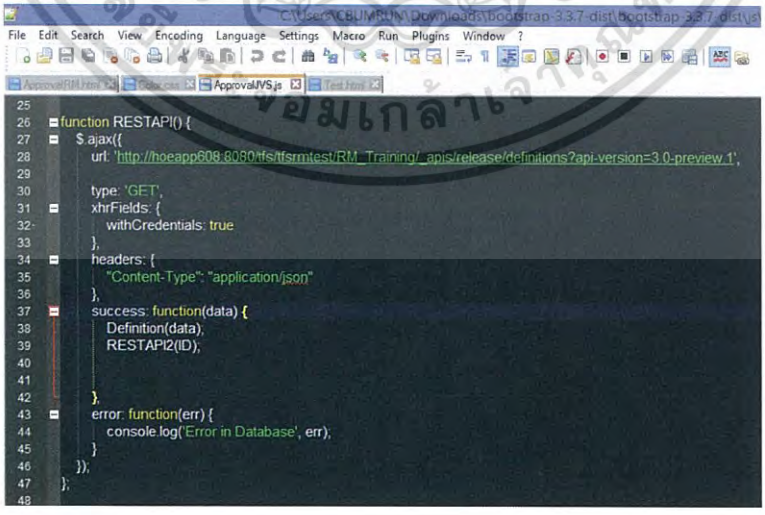


ภาพที่ 4.22 การค้นหาโปรแกรมโน้ตแพดพลัสพลัส



ภาพที่ 4.23 ไฟล์เอชทีเอ็มแอล

สร้างไฟล์จาวาสคริปต์ ขึ้นมาเพื่อเรียกใช้งานจาวาสคริปต์ สร้างตารางแสดงผลและใช้เจคิวรีในการเชื่อมต่อเอพีไอเพื่อดึงข้อมูลจากโปรแกรมทีเอฟเอส ซึ่งข้อมูลจัดเก็บอยู่ในรูปแบบของเจสัน ดังภาพ 4.24 จากนั้นสร้างไฟล์ซีเอสเอสขึ้นมา เพื่อใช้ในการตกแต่งหน้าเว็บแสดงผล ดังภาพ 4.25 มีการตกแต่งตารางแสดงผลว่าจะต้องมีขนาดเท่าไร สีอะไร ฟอนต์ที่ใช้แสดงผลคืออะไรเป็นต้น



ภาพที่ 4.24 สร้างไฟล์จาวาสคริปต์เพื่อเชื่อมต่อเอพีไอ

ภาพที่ 4.25 สร้างไฟล์ซีเอสเอส เพื่อใช้ในการตกแต่งหน้าเว็บ

เมื่อสร้างไฟล์ดังกล่าวพร้อมทั้งมีโค้ดคำสั่งต่าง ๆ แล้ว ให้เปิดไฟล์เอชทีเอ็มแอลขึ้นมา เบราเซอร์จะแสดงผลตารางแต่ละรีรีสเดฟฟินิชัน ว่ามีผู้อนุมัติ (Approval) หรือไม่มีผู้อนุมัติ (Non Approval) หากมีผู้อนุมัติมีในสภาพแวดล้อมไหน มีผู้ใดบ้าง สามารถคลิกไปที่ชื่อของรีรีสเดฟฟินิชันเพื่อลิงค์ไปยังหน้าการตั้งค่าของรีรีสเดฟฟินิชันเพื่อให้ได้ ดังภาพ 4.26

Non Approval

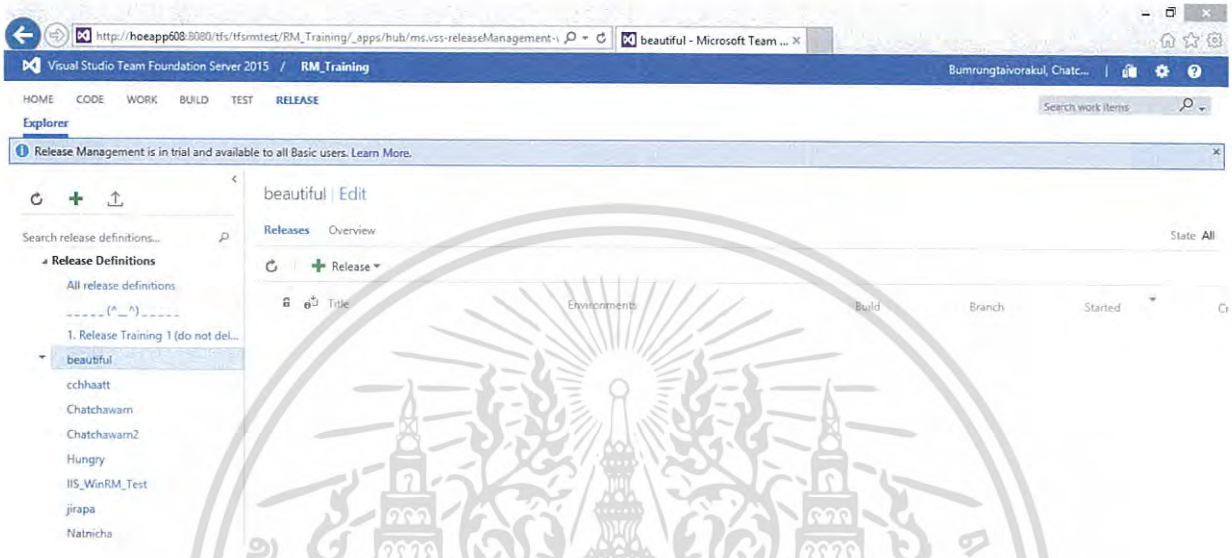
Release Definition
Workflow_Manager_ACC2PRD_ test
IIS_WinRM_Test
Natnicha
beautiful

Approval

Release Definition	Environments	Pre Approver	Post Approver
1. Release Training 1 (do not delete)	Acc		
	Prod		
-----(^_^)-----	Acceptance		
	Production		
Chatchawarn2	Acceptance	Sareekul, Natnicha C Tangjittanawath, Krissada C	Leedsuwanrut, Pijarin C
	Production	Prakartchai, Jirapa C	Suksanguan, Chawakorn C Jehsamoh, Fiklree C
Chatchawarn	Acceptance		Prakartchai, Jirapa C
TFS_ALEXPRESA	ACC		
TFS_WCV	Acc		
cchhaatt	ACCAAA	Prakartchai, Jirapa C	
Hungry	Starving		Sareekul, Natnicha C Leedsuwanrut, Pijarin C
	evil	Wieck, Benjamas C Boonlaetae, Pongpayak C	

ภาพที่ 4.26 เว็บแสดงรายชื่อผู้อนุมัติในแต่ละรีรีสเดฟฟินิชัน (Release Definition)

หลังจากคลิกที่ชื่อรีรีสเดฟฟินิชันแล้ว ก็จะเข้าไปยังหน้าเว็บตั้งค่าของรีรีสเดฟฟินิชันนั้น ๆ ได้เลยเพื่อง่ายต่อการตั้งค่ากำหนดผู้อนุมัติในสภาพแวดล้อมต่าง ๆ อาจจะกำหนดคนเดียว หลายคน หรือเป็นกลุ่ม หรือแม้กระทั่งกำหนดผู้อนุมัติก่อนหรือหลังก้ได้ ดังภาพ 4.27



ภาพที่ 4.27 โปรแกรมทีเอฟเอสในส่วนการตั้งค่าของแต่ละรีรีสเดฟฟินิชัน

บทที่ 5
บทสรุปและวิจารณ์ผลการดำเนินงาน

5.1 บทสรุปปริญญานิพนธ์

การเข้ารหัสด้วยการลงลายมือดิจิทัลบนไฟล์โค้ดที่ต้องการความปลอดภัย โดยใช้หลักการเข้ารหัสแบบไม่สมมาตร ประเภทลายมือดิจิทัล เข้ารหัสโดยการใช้กุญแจส่วนตัวของผู้ส่ง แล้วส่งผ่านเครือข่ายไปให้ผู้รับ ผู้รับสามารถถอดรหัสโดยใช้กุญแจสาธารณะของผู้ส่ง ตรวจสอบได้ว่า ไฟล์โค้ดที่ได้รับมานั้น ถูกแก้ไข ปลอมแปลง หรือมีความถูกต้องน่าเชื่อถือหรือไม่ อีกทั้งยังสามารถตรวจสอบว่าไฟล์โค้ดถูกลงลายมือดิจิทัลไว้เมื่อไร รวมไปถึงวันหมดอายุวัน ทำให้ไฟล์โค้ดความปลอดภัยมากขึ้น นอกจากนี้ยังได้มีการพัฒนาออกแบบหน้าเว็บของทีเอฟเอสในส่วนการจัดการรีรีสให้แสดงผู้อนุมัติว่ามีหรือไม่มี ซึ่งสามารถเพิ่มฟังก์ชันการทำงานให้กับโปรแกรมทีเอฟเอสได้ด้วย

5.2 ปัญหาที่พบในระหว่างการดำเนินงาน

- ขาดประสบการณ์ในการเรียกคำสั่งคอมมานไลน์ (Command Line) ซึ่งทำให้การทำงานในช่วงแรกเป็นไปด้วยความล่าช้า
- เอกสารที่นำมาศึกษาส่วนใหญ่เป็นภาษาอังกฤษ จึงยากต่อการทำความเข้าใจ
- บางขั้นตอนต้องรอดำเนินการช้า อาทิเช่นการขอใบรับรองต้องเป็นไปตามขั้นตอนของบริษัท ซึ่งใช้เวลานาน
- ภาษาคอมพิวเตอร์บางชนิด เป็นภาษาที่ไม่เคยเรียนในห้องเรียน จึงยากต่อการเรียนรู้ในเบื้องต้น

5.3 แนวทางการแก้ไข

- พยายามศึกษาตั้งแต่คำสั่งพื้นฐานไปจนถึงคำสั่งขั้นสูง ทำการทดลอง ใช้งานเป็นประจำจนรู้สึกคุ้นเคยกับคำสั่ง ทำให้สามารถนำไปปรับและประยุกต์ใช้กับงาน ให้ตรงกับความต้องการตนให้เร็วขึ้น
- ศึกษาหาความรู้จากอินเทอร์เน็ตเพิ่มเติม เมื่อเกิดความไม่เข้าใจในบางคำสั่งหรือการทำลองไม่เป็นไปตามความต้องการ
- ปรึกษาขอคำแนะนำจากพี่พนักงานที่ดูแลในการทำการทดลอง

5.4 แนวทางการพัฒนาต่อและนำไปใช้

- สามารถนำแนวทางนี้ ไปพัฒนาต่อในโปรแกรมทีเอฟเอสให้เป็นอีกฟังก์ชันหนึ่งในการทำการลงลายมือดิจิทัลอัตโนมัติเพื่อให้การนำไฟล์โค้ดขึ้นระบบมีความปลอดภัยมากยิ่งขึ้น

- สามารถนำแนวทางนี้ไปพัฒนาให้กับระบบอื่น ๆ ในบริษัทที่ต่างแพลตฟอร์มเพื่อเพิ่มความปลอดภัยและประสิทธิภาพของข้อมูลที่มากขึ้น

เอกสารอ้างอิง

[1] **ภาษาพาวเวอร์เชลล์คืออะไร** [ออนไลน์] เข้าถึงได้จาก :
http://thaiwinadmin.blogspot.sg/2008/05/kb2008218.html
(วันที่ค้นหาข้อมูล 1 ธันวาคม 2559)

[2] **เอชทีเอ็มแอลคืออะไร** [ออนไลน์] เข้าถึงได้จาก :
http://www.enjoyday.net/webtutorial/html/html_chapter01.html
http://krukikz.com/index.php?option=com_content&view=article&id=119&Itemid=152
(วันที่ค้นหาข้อมูล 3 ธันวาคม 2559)

[3] **จาวาสคริปต์คืออะไร** [ออนไลน์] เข้าถึงได้จาก :
http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2187-java-javascript-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.htm
(วันที่ค้นหาข้อมูล 3 ธันวาคม 2559)

[4] **ซีเอสเอสคืออะไร** [ออนไลน์] เข้าถึงได้จาก :
http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2193-css%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html
(วันที่ค้นหาข้อมูล 3 ธันวาคม 2559)

[5] **เจคิวรีคืออะไร** [ออนไลน์] เข้าถึงได้จาก :
http://www.100ydesign.com/column.php?id=000268
(วันที่ค้นหาข้อมูล 3 ธันวาคม 2559)

เอกสารอ้างอิง (ต่อ)

[6] **เจสันคืออะไร** [ออนไลน์] เข้าถึงได้จาก :
http://www.teacher.ssru.ac.th/nutthapat_ke/file.php/1/IntroJSON3_new.pdf
(วันที่ค้นหาข้อมูล 12 ธันวาคม 2559)

[7] **โปรแกรมทีเอฟเอส** [ออนไลน์] เข้าถึงได้จาก :
https://www.microsoft.com/thailand/visualstudio/vs2005/vsts/vsts_tfs.aspx
(วันที่ค้นหาข้อมูล 4 ธันวาคม 2559)

[8] **การจัดการรีรีส** [ออนไลน์] เข้าถึงได้จาก :
https://www.visualstudio.com/en-us/docs/release/getting-started/understand-rm
(วันที่ค้นหาข้อมูล 4 ธันวาคม 2559)

[9] **การทำงานของการจัดการรีรีส** [สไลด์การสอน] ที่มา:
สไลด์สื่อการสอนรีรีส แมแนจน์เมนต์ของ บริษัท เอ็กซอนโมบิล จำกัด

[10] **ภาพของการทำการจัดการรีรีสในส่วนของรีรีสเดฟฟินิชัน** [สไลด์การสอน] ที่มา:
สไลด์สื่อการสอนรีรีส แมแนจน์เมนต์ของ บริษัท เอ็กซอนโมบิล จำกัด

[11] **โปรแกรมเมคเสิท** [ออนไลน์] เข้าถึงได้จาก :
https://msdn.microsoft.com/en-us/library/bfsktky3(VS.100).aspx
(วันที่ค้นหาข้อมูล 4 ธันวาคม 2559)

[12] **โปรแกรมพีวีเคทูพีเอฟเอกซ์** [ออนไลน์] เข้าถึงได้จาก :
https://msdn.microsoft.com/windows/hardware/drivers/devtest/pvk2pfx
(วันที่ค้นหาข้อมูล 4 ธันวาคม 2559)

[13] **โปรแกรมไซน์ทูล** [ออนไลน์] เข้าถึงได้จาก :
https://msdn.microsoft.com/en-us/library/8s9b9yaz(v=vs.110).aspx
(วันที่ค้นหาข้อมูล 7 ธันวาคม 2559)

เอกสารอ้างอิง (ต่อ)

[14] **โปรแกรมโน้ตแพดพลัสพลัส** [ออนไลน์] เข้าถึงได้จาก :
http://www.mysmileeasy.com/notepad/
(วันที่ค้นหาข้อมูล 10 ธันวาคม 2559)

[15] **โปรแกรมไมโครซอฟท์เวิร์ด** [ออนไลน์] เข้าถึงได้จาก :
http://mookda25391205.blogspot.sg/2014/10/microsoft-word.html
(วันที่ค้นหาข้อมูล 15 ธันวาคม 2559)

[16] **การเขียนผังงานการทำงาน** [ออนไลน์]
https://stwannaporn.wordpress.com/2014/06/29/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%80%E0%B8%82%E0%B8%B5%E0%B8%A2%E0%B8%99%E0%B8%9C%E0%B8%B1%E0%B8%87%E0%B8%87%E0%B8%B2%E0%B8%99-flowchart/
(วันที่ค้นหาข้อมูล 20 ธันวาคม 2559)

[17] **ทฤษฎีการเข้ารหัส** [ออนไลน์] เข้าถึงได้จาก :
www.stech.ac.th/blogs/0398/wp.../chapter_6_security-system.ppt
(วันที่ค้นหาข้อมูล 29 พฤษจิกายน 2559)

ภาคผนวก ก
โปสเตอร์

Cooperative Education 2559
Code Signing

Chatchawarn Bumrungtaivorakul
Asst.Prof.Dr.Sutheera Puntheeranurak
Department of Computer Engineering
Email : chatzzkub@hotmail.com

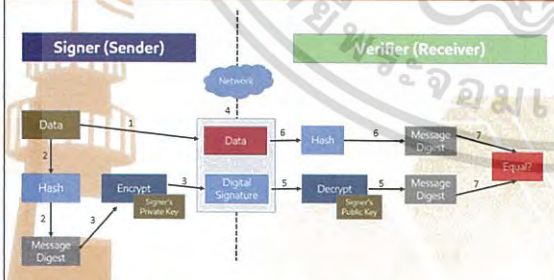
Abstract

In recent years, cyber attacks become more frequent which lead to severe financial and operational impacts to societies and organizations. This thesis aims at implementing "Code Signing" that is the digitally signing process to prevent a tampering attack. The signed code or executables is guaranteed that it has not been altered or corrupted. Moreover, code signing detects the software owner and ensure the right copy of the application. Hence, the code signing enhances security and effectiveness in organizations by mitigating risk from cyber attacks and ensuring the software right copy. This thesis explains industrial standard for code signing and current available applications, so that the organization can maintenance for long term.

Introduction

In the past, it was fine to keep code files without encryption. However, nowadays cyber crime widely spreads much more than the past. Although it is hard to secure code files, there are many ways to keep your code safe. "Code Signing" is one of techniques to help us make it safe and effective.

Methodology



1. Sender sends plaintext
2. Sender hash function to be the message digest
3. Let digest encrypt with sender private key for digital signature
4. Send both of plaintext and digital
5. Receiver takes digital signature decrypt with sender public key to be digest
6. Receiver gets plaintext convert Hash function to be digest
7. Receiver compares digest

Technology used

Microsoft

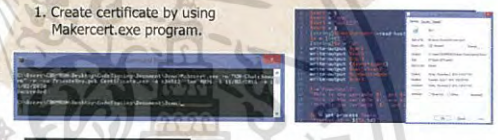
Makecert.exe
Create key, certificate

Pvk2pfx.exe
Merge key + certificate to PFX file

Signtool.exe
Sign and Verify

Result

1. Create certificate by using Makercert.exe program.



2. Combine the private key and the certificate to a single file which is called PFX file

3. Sign the file by using signtool.exe

4. Verify the file by using signtool.exe

Conclusion

According to the goals set, the solution can work as well. Code Signing can encrypt a code file by running easy command. It can help the user to increase security with a code file. However, there is a problem about type of code files. Not all types of code files support by Makecert program. Therefor, it can be useful for users who want to increase security with code file.

IT

ExxonMobil

ภาพที่ ก.1 โปสเตอร์