

แอปพลิเคชัน ฟิกซี

ยติชาติ หวังกุล¹ และ ชัยพร เขมะภาคะพันธ์²

¹คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิต

²ภาควิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิต

Emails: w.atichart@gmail.com, chaiyaporn@gmail.com

บทคัดย่อ

บทความนี้นำเสนอการพัฒนาแอปพลิเคชัน Fixzy ซึ่งเป็นแอปพลิเคชันช่วยจัดการงานซ่อมและการให้บริการในที่พักอาศัย (Home Services Application) โดยแอปพลิเคชันนี้ทำงานแบบออนไลน์เพื่อการนัดหมายช่าง มีการถ่ายภาพบริเวณที่มีปัญหาและต้องการซ่อมแซม ซึ่งจะมีการรับส่งข้อมูลผ่านเว็บเซิร์ฟเวอร์ โดยข้อมูลจะถูกบีบอัดและเข้ารหัสข้อมูลระหว่างการรับส่ง วิธีนี้จะสามารถแก้ปัญหาการรับส่งข้อมูลที่มีความล่าช้า โดยเฉพาะอย่างยิ่งกรณีที่มีข้อมูลมีขนาดใหญ่ เช่น รูปภาพ นอกจากนี้ยังเพิ่มความปลอดภัยด้วยการเข้ารหัสข้อมูลอีกด้วย

โดยผลการทดลองใช้งานจริงเพื่อแก้ปัญหาที่กล่าวมาข้างต้น แอปพลิเคชันที่นำเสนอสามารถลดเวลาในการรับส่งข้อมูลประเภทตัวอักษรได้ประมาณร้อยละ 80 และข้อมูลประเภทรูปภาพได้ประมาณร้อยละ 8 เมื่อเทียบกับแอปพลิเคชันในปัจจุบันที่ไม่ได้ใช้เทคนิคการบีบอัดและการเข้ารหัสข้อมูล ซึ่งผลจากการพัฒนาแอปพลิเคชันนี้สามารถลดค่าใช้จ่ายและปริมาณการใช้งานแบนด์วิดท์ของผู้ใช้ สามารถรองรับการขยายตัวของผู้ใช้งานแอปพลิเคชันได้เป็นอย่างดี เพิ่มความปลอดภัยของข้อมูล และลดค่าใช้จ่ายในส่วนที่เกี่ยวข้องกับระบบเว็บเซิร์ฟเวอร์ได้อย่างดี

คำสำคัญ— แอปพลิเคชันออนไลน์, การให้บริการในที่พักอาศัย, เว็บเซิร์ฟเวอร์

Abstract

This paper presents the development of Fixzy Application. Is an Application for management, maintenance and services in households. Applications are available online to make an appointment technician. The imaging a specific area that has a problems and need to repair. This technically have been data transmission to the Web Service. The data will be compressed and encrypted during transmission. It can solve the data transmission delay problem, especially case of the data is larger such as a pictures. It also increasing security by data encryption.

The experimental result has been use in real of the solutions mentioned above. This application is presented this can reduce the time for data transmission types of characters to

approximately 80 percent, and image data of approximately 8 percent compared with the current version of the application that does not use compression and encryption. The result of development an application can reduce costs and bandwidth usage of users. This can encourage the expansion of the users as well. This program will help the system reducing the cost of Web Service as well.

Keywords— Online Application; Web Service;

1. บทนำ

ปัจจุบัน การค้นหาช่างซ่อมหรือการให้บริการในที่พักอาศัยเป็นเรื่องยากเนื่องจากเวลาเกิดเหตุอุปกรณ์ในที่พักอาศัยชำรุด ไม่สามารถทราบได้ว่า ณ เวลานั้นจะหาช่างผู้ชำนาญงานได้จากแหล่งใด หรือถ้าเกิดมีข้อมูลการติดต่อช่างผู้นั้นแล้ว ก็ไม่สามารถทราบได้ว่าเวลาที่ต้องการรับบริการนั้นช่างสามารถเข้ารับบริการได้หรือไม่ ทำให้ผู้ต้องการรับบริการไม่มีทางเลือกในการหาช่างมากนัก อาจจะต้องเดินทางไปหาช่างตามตามเสาไฟฟ้า ตู้โทรศัพท์สาธารณะ แม้กระทั่งค้นหาจากสมุดหน้าเหลืองหรืออินเทอร์เน็ต ซึ่งผู้ต้องการรับบริการไม่สามารถทราบตัวตนและความชำนาญในการให้บริการงานนั้นๆได้ ทำให้เกิดความเสี่ยงต่อผู้รับบริการ ทางผู้จัดทำแอปพลิเคชัน Fixzy ได้เล็งเห็นถึงปัญหาที่กล่าวมาข้างต้น จึงได้ดำเนินการพัฒนาปรับปรุงแก้ไข เพื่อต้องการยกระดับการให้บริการ สร้างมาตรฐานและเพิ่มประสิทธิภาพการทำงานของระบบ อีกทั้งยังมีการแจ้งหรือระบุข้อมูลความสามารถของช่างที่มีอยู่ในระบบต่อผู้ใช้งานเพื่อเป็นข้อมูลในการตัดสินใจเลือกใช้บริการ

เนื่องจากปัจจุบัน โทรศัพท์เคลื่อนที่สมาร์ทโฟนเข้ามามีบทบาทในการดำรงชีวิตของบุคคลทั่วไปมากขึ้น ซึ่งเป็นปัจจัยหลักในการเจริญเติบโตของธุรกิจด้านแอปพลิเคชัน จากข้อมูลสำนักงานสถิติแห่งชาติ [1] สถิติผู้ใช้โทรศัพท์เคลื่อนที่สมาร์ทโฟนในประเทศไทยปี 2558 พบว่ามีจำนวนผู้ใช้โทรศัพท์เคลื่อนที่สมาร์ทโฟนมากกว่า 20 ล้านคน ดังตารางที่ 1 และแอปพลิเคชันบนมือถือได้กลายเป็นส่วนสำคัญร่วมกับการดำเนินธุรกิจในปัจจุบันมากขึ้น ผู้จัดทำแอปพลิเคชัน Fixzy จึงได้เกิดแนวคิดที่จะใช้แอปพลิเคชันเพื่อเป็นการต่อยอดในการทำธุรกิจซึ่งแอปพลิเคชัน Fixzy เป็นแอปพลิเคชันที่ช่วยจัดการงานซ่อมแซมและการให้บริการในที่พักอาศัยด้วยการทำงานระบบออนไลน์ เพื่อใช้ในการ

เรียกงานบริการและการนัดหมายช่าง โดยการนัดหมายการเข้าให้บริการนั้นจะมีการถ่ายภาพบริเวณที่เสียหายหรือชำรุดและต้องการซ่อมแซม ซึ่งโดยเทคนิคแล้วจะมีการรับส่งข้อมูลผ่านเว็บเซอร์วิสแบบ REST ในกรณีที่มีข้อมูลรูปภาพมีขนาดใหญ่ อาจทำให้เกิดความล่าช้าระหว่างการรับส่งข้อมูลผ่านเว็บเซอร์วิสได้ ดังนั้น แอปพลิเคชัน Fixzy จึงใช้เทคนิคในการบีบอัดและการเข้ารหัสชุดข้อมูลนั้น เพื่อแก้ปัญหาที่กล่าวมาข้างต้น ทำให้ข้อมูลที่ได้รับส่งผ่านเว็บเซอร์วิสนั้นมีขนาดเล็กลงและยังปลอดภัยอีกด้วย

ตาราง 1 จำนวนผู้ใช้งานโทรศัพท์เคลื่อนที่สมาร์ตโฟน

จำนวนผู้ใช้งานโทรศัพท์เคลื่อนที่สมาร์ตโฟน				
อายุ	ไม่มีเครื่อง	1 เครื่อง	2 เครื่อง	3 เครื่องขึ้นไป
รวม	25,882,027	23,402,431	283,585	26,095
6-10ปี	347,429	302,330	-	64
11-14ปี	671,322	1,331,156	1,552	688
15-19ปี	906,540	3,487,255	9,686	881
20-24ปี	1,148,667	3,505,607	24,946	2,950
25-29ปี	1,178,640	3,290,503	39,212	938
30-34ปี	1,579,617	3,007,003	50,637	4,008
35-39ปี	2,319,989	2,595,373	51,086	1,771
40-49ปี	6,451,160	3,463,652	61,857	6,282
50-59ปี	6,118,023	1,779,641	31,430	5,278
60ปีขึ้นไป	5,160,640	639,911	13,179	3,236

เพื่อรองรับการขยายตัวของผู้ใช้งานแอปพลิเคชันในอนาคต ซึ่งส่งผลกระทบต่อเจริญเติบโตของธุรกิจ ทำให้มีการใช้งานแบนด์วิดท์ฝั่งเว็บเซอร์วิสเพิ่มมากขึ้น ส่งผลให้เกิดค่าใช้จ่ายเพิ่มขึ้น งานวิจัยนี้จึงได้นำวิธีการบีบอัดข้อมูลมาใช้ในการส่งข้อมูลระหว่างแอปพลิเคชันกับเว็บเซอร์วิสเพื่อลดต้นทุนค่าใช้จ่ายทางด้านเว็บเซอร์วิส และช่วยให้ผู้ใช้ลดการใช้งานแบนด์วิดท์ได้อีกด้วย

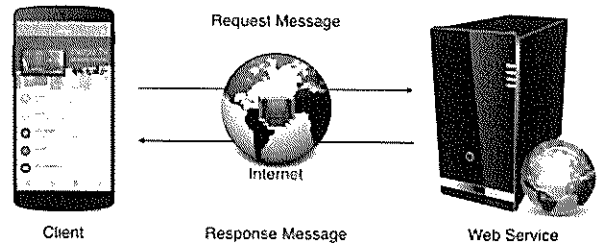
2. เครื่องมือและวิธีการที่ใช้ในการวิจัย

2.1 เว็บเซอร์วิส (Web Service) คือระบบซอฟต์แวร์ที่ทำงานโดยการแลกเปลี่ยนข้อมูล ระหว่างแอปพลิเคชันผ่านระบบเครือข่ายดังรูปที่ 1 โดยการให้บริการของเว็บเซอร์วิสนั้นจะถูกเรียกใช้งานจากแอปพลิเคชันอื่นๆ ในรูปแบบ RPC (Remote Procedure Call) โดยเป็นตัวกลางในการทำให้ติดต่อสื่อสารกันได้ง่ายและช่วยให้แอปพลิเคชันสามารถติดต่อสื่อสารกันได้ แม้จะพัฒนาโดยใช้ภาษาที่ต่างกันออกไป โดยการพัฒนาเว็บเซอร์วิสนั้นสามารถเลือกพัฒนาได้ 2 รูปแบบคือ

1. SOAP (Simple Object Access Protocol) เป็นโพรโทคอลมาตรฐานที่ใช้ร่วมกับเว็บเซอร์วิส สามารถทำงานบนโพรโทคอลใดๆก็ได้ อีกทั้งยังสนับสนุนความปลอดภัยทั้งการยืนยันตัวตน การอนุญาต และการเข้ารหัสข้อมูล แต่ยากต่อการพัฒนาเนื่องจากสนับสนุนรูปแบบข้อมูล XML เพียงอย่างเดียว และเนื่องจากมีโครงสร้างหลายส่วน ทำให้มี Overhead สูง ทำให้ใช้แบนด์วิดท์มาก จึงเหมาะกับการพัฒนาเพื่อใช้ใน ตลาด Enterprise โดยเฉพาะ [2]

2. REST (Representational State Transfer) เป็นรูปแบบการทำงานที่ใช้มาตรฐานการทำงานอยู่บนโพรโทคอล HTTP ทำให้สามารถพัฒนาได้ง่าย และสนับสนุนรูปแบบข้อมูลหลายประเภท เช่น XML JSON Plain Text เป็นต้น นอกจากนี้ยังสามารถรองรับการขยายระบบได้ง่าย และ

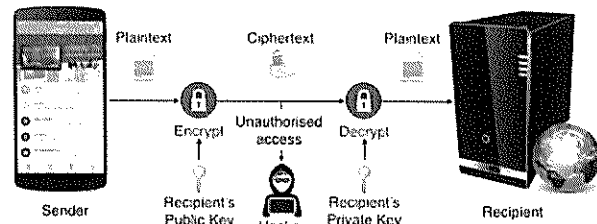
ยังรองรับเรื่องการ Caching ข้อมูลอีกด้วย แต่การทำงานรูปแบบ REST มีข้อจำกัดที่จะทำงานได้เฉพาะโพรโทคอล HTTP เท่านั้น และจะต้องจัดการเรื่องความปลอดภัยและความน่าเชื่อถือด้วยระบบเอง



รูปที่ 1 การทำงานระหว่างแอปพลิเคชันกับเว็บเซอร์วิส

2.2 JSON (Java Script Object Notation) เป็นวิธีการที่ทำให้ JavaScript สามารถแลกเปลี่ยนข้อมูลกับ Server ได้ ซึ่ง JSON เป็นภาษาที่มนุษย์สามารถเข้าใจได้ โดยที่ JSON จะใช้ “[]” แทน Array และใช้ “{}” แทน Hash โดยแต่ละสมาชิกจะคั่นด้วย Comma “,” และแต่ละชื่อสมาชิกจะคั่นด้วย Colon “:”

2.3 SSL (Secure Socket Layer) เป็นโพรโทคอลในการรักษาความปลอดภัยระหว่างการรับส่งข้อมูลบนระบบเครือข่าย เพื่อยืนยันตัวตนของเว็บไซต์นั้นๆ โดยการรับส่งข้อมูลระหว่างแอปพลิเคชันกับเว็บเซอร์วิสนั้นจะมีการเข้ารหัสเพื่อป้องกันการดักจับข้อมูล ซึ่งเหมาะแก่การใช้งานเพื่อปกป้องข้อมูลสำคัญ เช่น ข้อมูลบัตรเครดิต รหัสผ่าน ดังแสดงในรูปที่ 2



รูปที่ 2 การทำงานของ SSL

2.4 MVC (Model-View-Controller) เป็นสถาปัตยกรรมซอฟต์แวร์ซึ่งแบ่งโครงสร้างการทำงานเป็น 3 ส่วน คือ

1. Model ทำงานเกี่ยวกับข้อมูล โดยการรับคำสั่งมาจาก Controller เช่น Insert, Update หรือ Delete เป็นต้น
2. View ทำหน้าที่เป็นส่วนแสดงผล หรือที่เรียกว่า User Interface
3. Controller เป็นส่วนที่ใช้ประมวลผล หรือวิเคราะห์เงื่อนไข และส่งข้อมูลไปยัง Model หรือ View

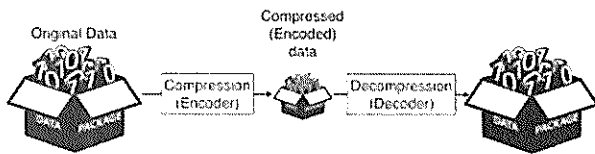
2.5 งานวิจัยที่เกี่ยวข้อง

การบีบอัดข้อมูลจากรูปที่ 3 คือ วิธีการในการจัดเก็บข้อมูลซึ่งทำให้สามารถใช้เนื้อที่ในการจัดเก็บข้อมูลน้อยลง ดังนั้นการบีบอัดข้อมูลนั้นจึงมีความสำคัญในระบบการสื่อสารมาก เนื่องจากจะทำให้รับส่งข้อมูลได้มากขึ้นโดยใช้แบนด์วิดท์เท่าเดิม ซึ่งสามารถแบ่งการบีบอัดข้อมูลได้เป็นสองประเภท ได้แก่

1. การบีบอัดข้อมูลแบบไม่สูญเสีย (Lossless Data Compression) คือ รูปแบบของการบีบอัดข้อมูลในแบบดิจิทัล โดยจะไม่มีการสูญเสียข้อมูลใดๆ ทุกๆ การบีบอัด เมื่อทำการขยายข้อมูล จะได้ข้อมูลที่สมบูรณ์เหมือนข้อมูลต้นฉบับ เช่น การบีบอัดภาพดิจิทัลแบบ PNG (Portable Network Graphics) การบีบอัดเสียงแบบ AIFF (Audio Interchange File Format) เป็นต้น

2. การบีบอัดข้อมูลแบบมีการสูญเสีย (Lossy Data Compression) คือ การบีบอัดข้อมูลที่จะตัดส่วนของข้อมูลที่ซ้ำซ้อน โดยจะไม่นำส่วนนั้นมาเข้ารหัส ซึ่งการใช้เทคนิคนี้จำเป็นต้องพิจารณาอย่างระมัดระวังและต้องคำนึงถึงการใช้งานจริงเป็นกรณีไป เช่นการลดคุณภาพสีของภาพถ่าย หากนำไปใช้ในทางด้านสื่อ อาจละทิ้งได้ในบางกรณี แต่หากนำไปใช้ในเชิงการแพทย์ การลดคุณภาพของภาพนั้นเพียงเล็กน้อย อาจทำให้การวินิจฉัยโรคของแพทย์ผิดพลาดได้ เป็นต้น

จากงานวิจัย High Performance Lossless Data Compression FPGAs using OpenCL [3] ได้นำเสนอเทคนิคและวิธีการบีบอัดข้อมูลเพื่อเพิ่มประสิทธิภาพในการลดขนาดของข้อมูลที่ส่งผ่านระบบเครือข่าย เพื่อให้ใช้งานบนตัวชิปได้อย่างมีประสิทธิภาพ โดยการวิจัย [3] ได้มีการเทคนิคในการบีบอัดข้อมูลแบบ Gzip และทำการเข้ารหัส LZ77 Huffman ซึ่งเป็นอัลกอริทึมส่วนหนึ่งของการบีบอัดข้อมูลที่เรียกว่า Lossless Compression (รูปแบบการบีบอัดข้อมูลดิจิทัล ซึ่งจะไม่มีการสูญเสียข้อมูลใด ๆ ในการบีบอัด) จากนั้นนำมาทำการเปรียบเทียบการใช้งานในเชิงพาณิชย์ระหว่าง FPGA ASIC และ CPU โดยผลการทดลองที่ดีที่สุดในการบีบอัดข้อมูลสามารถแลกเปลี่ยนข้อมูลได้เพิ่มขึ้นจากเดิม 12 เท่า



รูปที่ 3 ขั้นตอนการทำงานของบีบอัดข้อมูล

งานวิจัย Improve Performance of Advance Encryption Standard using Parallel Computing [4] ได้นำเสนอการดำเนินการเข้ารหัสเพื่อความปลอดภัยของข้อมูล โดยใช้อัลกอริทึม AES (Advanced Encryption Standard) และใช้การคำนวณแบบขนานโดยใช้ JPPF (Java Parallel Programming Framework) ซึ่งมีความยืดหยุ่นในการปรับปรุงประสิทธิภาพการทำงานในแง่ของความเร็ว ในส่วนของขั้นตอนการดำเนินงานแบ่งเป็น 3 ส่วน ได้แก่

1. Client ไว้สำหรับรับส่งข้อมูลผ่าน API
2. Server เป็นตัวกลางที่รับข้อมูลส่งไปยังโหนด
3. Nodes เป็นส่วนของการดำเนินงานในตระกะทางธุรกิจ

3. วิธีการดำเนินการวิจัย

งานวิจัยนี้ นำเสนอการพัฒนาแอปพลิเคชัน Fixzy ซึ่งเป็นแอปพลิเคชันจัดการงานซ่อมและงานบริการภายในที่พักอาศัย ด้วยระบบออนไลน์ โดยการนัดหมายงานซ่อมและบริการแต่ละครั้งนั้นจะมีการระบุ

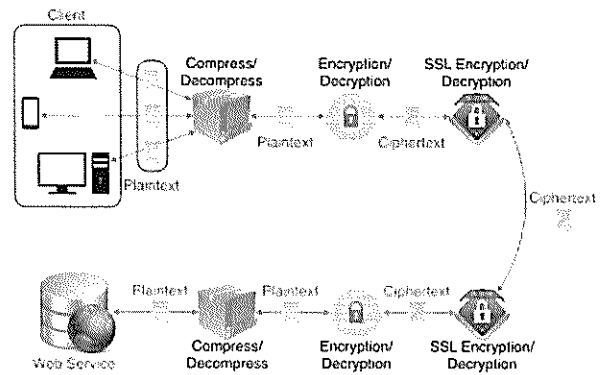
รายละเอียดและถ่ายภาพจุดที่ต้องการซ่อมแซม ซึ่งทางเทคนิคแล้วจะต้องมีการรับส่งข้อมูลระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์ ถ้าเกิดกรณีข้อมูลชุดนั้นมีขนาดใหญ่ อาจทำให้เกิดปัญหาหรือเกิดความล่าช้าขณะรับส่งข้อมูลได้ ดังนั้นงานวิจัยนี้จึงต้องการแก้ปัญหาที่กล่าวมาข้างต้นด้วยเทคนิคการบีบอัดข้อมูล [5], [6] เพื่อลดขนาดของชุดข้อมูลก่อนการรับส่งระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์ และเพื่อให้การทำงานเกิดประสิทธิภาพมากที่สุด งานวิจัยนี้จึงได้นำเทคนิคการเข้ารหัสข้อมูลมาใช้ร่วมด้วย เพื่อเพิ่มความปลอดภัยของข้อมูลผู้ใช้ [7]

ขั้นตอนการทำงานของแอปพลิเคชัน Fixzy สามารถแบ่งการแจ้งบริการออกเป็นหลายหมวด เช่น

- 1.หมวดงานไฟ เป็นหมวดที่ให้บริการงานซ่อมด้านระบบไฟฟ้า เช่น งานเดินสายไฟ แก้ปัญหาไฟช็อต เป็นต้น
- 2.หมวดงานน้ำ เป็นหมวดที่ให้บริการงานซ่อมด้านระบบน้ำ เช่น งานน้ำรั่วซึม ท่อตัน ปัมป์น้ำเสีย เป็นต้น
- 3.หมวดงานเครื่องใช้ไฟฟ้า เป็นหมวดที่ให้บริการงานซ่อมด้านอุปกรณ์อิเล็กทรอนิกส์หรือเครื่องใช้ไฟฟ้า เช่น งานซ่อมคอมพิวเตอร์ ซ่อมแอร์ ซ่อมเครื่องทำน้ำอุ่น เป็นต้น

เลือกหมวดงานบริการแล้ว จะต้องมีการระบุรายละเอียดเบื้องต้น เพื่อใช้เป็นข้อมูลในการจัดการ เช่น วันและเวลาที่ต้องการรับบริการ สถานที่ที่ต้องการรับบริการ เบอร์โทรศัพท์ติดต่อกลับ และรูปถ่ายบริเวณที่ต้องการซ่อมแซม เป็นต้น จากนั้นจึงทำการเรียกใช้บริการและรอช่างที่อยู่บริเวณใกล้เคียงตอบรับงานเพื่อทำการนัดหมายเข้าให้บริการต่อไป

ภาพรวมของแอปพลิเคชัน Fixzy ประกอบด้วยการทำงานสองส่วน ได้แก่ ส่วนการทำงานของแอปพลิเคชันและส่วนของเว็บเซิร์ฟเวอร์ โดยขั้นตอนการทำงานของแอปพลิเคชัน Fixzy แสดงดังรูปที่ 4

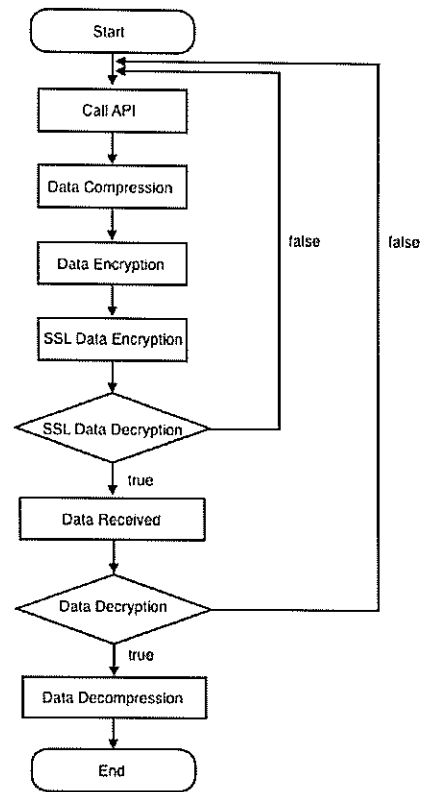


รูปที่ 4 ภาพรวมของแอปพลิเคชัน Fixzy

ขั้นตอนการทำงานของแอปพลิเคชัน Fixzy ด้วยการรับส่งข้อมูลระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์ ซึ่งขั้นตอนการทำงานมีดังนี้

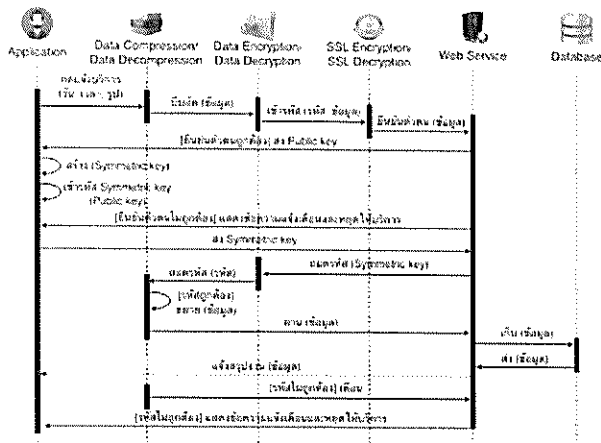
1. ผู้ใช้งานแอปพลิเคชันเลือกหมวดงานที่ต้องการแจ้งบริการ และระบุรายละเอียด วันและเวลาที่ต้องการรับบริการ สถานที่ที่ต้องการรับบริการ เบอร์โทรศัพท์ติดต่อกลับ รายละเอียดเพิ่มเติม เช่น ขนาด BTU (British Thermal Unit) ของแอร์ เป็นต้น และถ่ายภาพบริเวณที่เกิดความเสียหาย หรือจุดที่ต้องการซ่อมแซม จากนั้นทำการแจ้งบริการ

2. หลังจากผู้ใช้ทำการกดแจ้งบริการแล้ว จะได้ชุดข้อมูลมาชุดหนึ่งเพื่อรอการบีบอัดข้อมูล โดยการบีบอัดข้อมูลนี้จะใช้เทคนิคการบีบอัดข้อมูลในรูปแบบ Gzip
 3. ขั้นตอนถัดไปจะนำชุดข้อมูลที่ทำการบีบอัดข้อมูลแล้วมาเข้ารหัสเพื่อเพิ่มความปลอดภัยให้กับข้อมูลด้วยวิธีการเข้ารหัสแบบ AES (Advanced Encryption Standard)
 4. จากนั้นแอปพลิเคชันจะทำการร้องขอการส่งข้อมูลผ่าน Protocol HTTPS (Hypertext Transfer Protocol Secure) ซึ่งมีการเข้ารหัสข้อมูลในการระหว่างการส่งชุดข้อมูลนั้น และมีการทำ Web Authentication เพื่อตรวจสอบการยืนยันตัวตนของเว็บเซิร์ฟเวอร์
 5. เมื่อเว็บเซิร์ฟเวอร์ได้รับการร้องขอแล้วจะทำการตอบกลับด้วย Certificate เพื่อเป็นการยืนยันว่าเว็บเซิร์ฟเวอร์นี้เป็นเว็บเซิร์ฟเวอร์ที่ใช้งานกับแอปพลิเคชันนี้จริง และได้ทำการส่ง Public Key ของเว็บเซิร์ฟเวอร์ไปยังแอปพลิเคชันเพื่อใช้ในการเข้ารหัสด้วย
 6. หลังจากนั้น แอปพลิเคชันจะทำการตรวจสอบ Certificate ที่ได้รับจากเว็บเซิร์ฟเวอร์ว่าเว็บเซิร์ฟเวอร์นี้เป็นเว็บเซิร์ฟเวอร์ที่ใช้งานร่วมกับแอปพลิเคชันนี้จริง และทำการสร้าง Symmetric Key ขึ้นมาหนึ่งชุดไว้สำหรับการเข้ารหัสข้อมูลที่ต้องการรับส่งระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์
 7. เมื่อแอปพลิเคชันทำการสร้าง Symmetric Key เสร็จเรียบร้อยแล้ว จะทำการเข้ารหัส Symmetric Key นั้นด้วย Public Key ที่เว็บเซิร์ฟเวอร์ได้ส่งมาให้
 8. จากนั้นแอปพลิเคชันจะทำการส่ง Symmetric Key ที่ทำการเข้ารหัสกลับไปยังเว็บเซิร์ฟเวอร์
 9. เว็บเซิร์ฟเวอร์จะทำการถอดรหัส Symmetric Key ที่ส่งมาจากแอปพลิเคชันด้วย Private Key ของเว็บเซิร์ฟเวอร์ แล้วจึงเริ่มกระบวนการเชื่อมต่อแบบ SSL
 10. เมื่อทำการกระบวนการเชื่อมต่อ SSL เสร็จแล้ว ข้อมูลที่ถูกส่งมาจากแอปพลิเคชันจะถูกถอดรหัสด้วย Symmetric Key
 11. หลังจากผ่านขั้นตอนในการรับส่งข้อมูลแล้ว เว็บเซิร์ฟเวอร์จะทำการถอดรหัสด้วย Secret Key ที่ตรงกับ Secret Key ของแอปพลิเคชัน
 12. เมื่อเว็บเซิร์ฟเวอร์ทำการถอดรหัสแล้ว จะนำชุดข้อมูลที่ได้มาทำการขยายเพื่ออ่านข้อมูลต่อไป
 13. หลังจากได้ชุดข้อมูลที่สมบูรณ์แล้ว เว็บเซิร์ฟเวอร์จะทำการเก็บข้อมูลนั้นลงฐานข้อมูลการแจ้งงานของผู้ใช้ และทำการส่งผลตอบรับกลับไปยังแอปพลิเคชันด้วยวิธีการเดิม
 14. เมื่อแอปพลิเคชันได้รับผลตอบรับกลับมาแล้ว ถือเป็นอันสิ้นสุดกระบวนการแจ้งบริการ และรอการตอบรับของช่างที่จะเข้าไปให้บริการต่อไป
- แอปพลิเคชัน Fixzy มีขั้นตอนการพิจารณาการจัดการหลายขั้นตอน เพื่อพัฒนาประสิทธิภาพด้านเวลา และเพิ่มความปลอดภัยให้กับข้อมูลที่ต้องการรับส่งระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์ โดยมีขั้นตอนการทำงานดังรูปที่ 5



รูปที่ 5 ขั้นตอนการพิจารณาการจัดการแอปพลิเคชัน Fixzy

- ขั้นตอนการพิจารณาการจัดการแอปพลิเคชัน Fixzy ด้วยการจำลองข้อมูลประเภทรูปถ่ายขนาด 102 กิโลไบต์ KB มีดังนี้
1. สมมติให้แอปพลิเคชันร้องขอข้อมูลจากเว็บเซิร์ฟเวอร์พร้อมทั้งส่งข้อมูลประเภทรูปถ่ายขนาด 102 กิโลไบต์ KB ออกไปด้วย
 2. นำชุดข้อมูลที่ทำการส่งออกมาด้วยนั้นทำการบีบอัดเพื่อให้มีขนาดเล็กลง
 3. เข้ารหัสข้อมูลที่ได้จากการบีบอัดข้อมูลเพื่อความปลอดภัย
 4. ส่งข้อมูลด้วยกระบวนการ SSL โดยมีการเข้ารหัสเพื่อเพิ่มความปลอดภัยในการยืนยันตัวตนระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์
 5. เว็บเซิร์ฟเวอร์ทำการถอดรหัสจากกระบวนการ SSL
 6. หากถอดรหัสถูกต้อง เว็บเซิร์ฟเวอร์จะทำการรับข้อมูลที่ถูกส่งมานั้นไว้ หากถอดรหัสไม่ถูกต้อง เว็บเซิร์ฟเวอร์จะส่งผลตอบรับกลับไปยังแอปพลิเคชันว่าไม่สามารถรับข้อมูลได้ เพื่อเริ่มกระบวนการร้องขอใหม่ต่อไป
 7. เว็บเซิร์ฟเวอร์นำข้อมูลที่ได้รับมาทำการถอดรหัส
 8. ถ้ารหัสตรงกัน เว็บเซิร์ฟเวอร์จะทำการเก็บข้อมูลไว้เพื่อรอทำขั้นตอนต่อไป ถ้ารหัสไม่ตรงกัน เว็บเซิร์ฟเวอร์จะส่งผลตอบรับกลับไปยังแอปพลิเคชันว่าไม่สามารถรับข้อมูลได้ เพื่อเริ่มกระบวนการร้องขอใหม่ต่อไป
 9. เว็บเซิร์ฟเวอร์นำข้อมูลที่ถอดรหัสแล้วมาทำการขยายเพื่ออ่านข้อมูล
 10. เว็บเซิร์ฟเวอร์อ่านข้อมูลที่ส่งมานั้น และส่งผลตอบรับไปยังแอปพลิเคชันว่าได้รับข้อมูลแล้ว

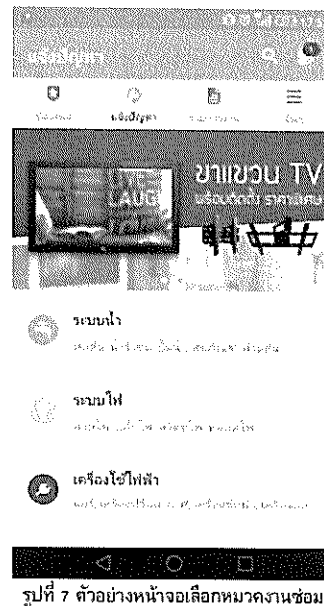


รูปที่ 6 การจำลอง Sequence Diagram การทำงานของแอปพลิเคชัน Fixzy

จากรูปที่ 6 แสดงการจำลองภาพการทำงานของแอปพลิเคชัน Fixzy ด้วย Sequence Diagram ดังนี้

1. ผู้ใช้งานกดแจ้งบริการผ่านแอปพลิเคชันโดยระบุอินพุต วัน เวลา และรูปภาพ
2. นำอินพุตที่รับมาทำการบีบอัดข้อมูล
3. ทำการเข้ารหัสข้อมูลที่ทำการบีบอัดมาแล้ว
4. ส่งข้อมูลไปยังเว็บเซอร์วิสผ่าน HTTPS Protocol เพื่อยืนยันตัวตน
5. ถ้ายืนยันตัวตนไม่ถูกต้อง เว็บเซอร์วิสจะส่งผลตอบรับไปยังแอปพลิเคชัน และยกเลิกงานแจ้งบริการนั้น
6. ถ้ายืนยันตัวตนถูกต้อง เว็บเซอร์วิสจะส่ง Public key กลับไปให้แอปพลิเคชัน
7. แอปพลิเคชันทำการสร้าง Symmetric key ขึ้นมา
8. ทำการเข้ารหัส Symmetric key ด้วย Public key ที่ส่งมาจากเว็บเซอร์วิส
9. แอปพลิเคชันจะส่ง Symmetric key ไปยังเว็บเซอร์วิส
10. เว็บเซอร์วิสทำการถอดรหัสข้อมูลที่รับมาจากข้อ 4 ด้วย Symmetric key
11. เว็บเซอร์วิสทำการถอดรหัสข้อมูลด้วยรหัสผ่านที่ตรงกันกับรหัสผ่านในแอปพลิเคชัน
12. หากรหัสผ่านถูกต้องเว็บเซอร์วิสจะทำการขยายข้อมูล
13. เว็บเซอร์วิสทำการอ่านข้อมูลหลังจากขยายข้อมูลแล้ว
14. เว็บเซอร์วิสส่งข้อมูลไปเก็บยังฐานข้อมูล
15. เว็บเซอร์วิสจะดึงข้อมูลแจ้งสรุปงานได้ฐานข้อมูลออกมา
16. เว็บเซอร์วิสส่งข้อมูลแจ้งสรุปงานไปยังแอปพลิเคชัน
17. หากรหัสผ่านไม่ถูกต้อง เว็บเซอร์วิสจะส่งผลตอบรับไปยังแอปพลิเคชัน และยกเลิกงานแจ้งบริการนั้น

จากรูปที่ 7 แสดง GUI (Graphic User Interface) หน้าเลือกหมวดงานซ่อม โดยในด้านบนเป็นส่วนของโฆษณา และถัดมาคือส่วนของหมวดงาน



รูปที่ 7 ตัวอย่างหน้าจอเลือกหมวดงานซ่อม

จากรูปที่ 8 แสดง GUI (Graphic User Interface) หน้ารายละเอียดเบื้องต้นก่อนการแจ้งบริการและหลังการแจ้งบริการ โดยรูปซ้ายมือแสดงช่องกรอกรายละเอียดการบริการเพิ่มเติม ภาพถ่ายบริเวณที่ต้องการซ่อมแซม และปุ่มกดใช้รหัสส่วนลด ส่วนรูปขวามือแสดงป้ายรายการ (Tab) ช่างที่รับงาน และรายละเอียดการบริการ



รูปที่ 8 ตัวอย่างหน้าจอระบุข้อมูลก่อนการแจ้งบริการและหลังการแจ้งบริการ

4. ผลการดำเนินงาน

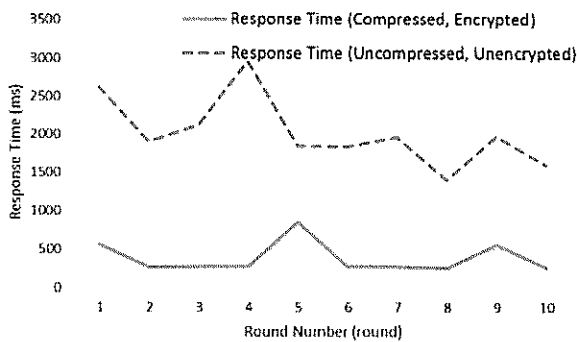
การดำเนินการทดลองจับเวลาหาค่า Response Time (ความเร็วในการตอบสนอง) ของการรับส่งข้อมูลระหว่างแอปพลิเคชัน Fixzy กับเว็บเซอร์วิสโดยการจำลองข้อมูล 2 ประเภทคือ ข้อมูลตัวอักษรมีขนาด 103 กิโลไบต์ (KB) และข้อมูลรูปภาพมีขนาด 102 กิโลไบต์ (KB) มาบีบอัดและเข้ารหัสข้อมูล เปรียบเทียบกับชุดข้อมูลชุดเดียวกันโดยไม่ใช้เทคนิคการบีบอัดและเข้ารหัสข้อมูล และทดลองจับเวลาเป็นจำนวนทั้งหมด 10 ครั้ง ได้เวลาเฉลี่ย Response Time ของข้อมูลตัวอักษรอยู่ที่ 386.8

มิลลิวินาที (ms) ดังรูปที่ 9 และได้เวลาเฉลี่ย Response Time ของข้อมูลรูปภาพอยู่ที่ 12738.7 มิลลิวินาที (ms) ดังรูปที่ 10

สภาพแวดล้อมที่ใช้ในการทดลองรับส่งข้อมูลระหว่างแอปพลิเคชันกับเว็บเซิร์ฟเวอร์ครั้งนี้ได้แก่

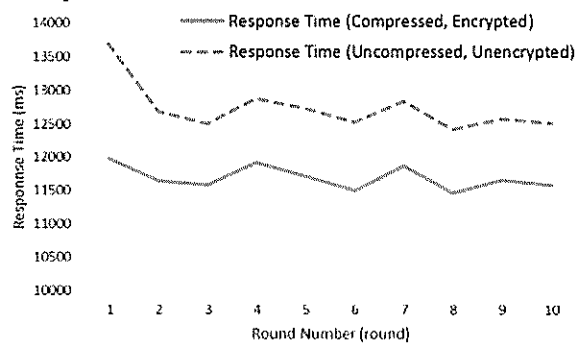
1. อินเทอร์เน็ตความเร็วสูงใยแก้วนำแสง (Optical Fiber) แบนด์วิดท์ดาวน์โหลด 10 เมกะไบต์ (MB) อัปโหลด 2 เมกะไบต์ (MB)

จากรูปที่ 9 แสดงผลการทดลองวัดผลจากข้อมูลประเภทตัวอักษรขนาด 103 กิโลไบต์ (KB) โดยเปรียบเทียบกันระหว่างแอปพลิเคชันที่ใช้เทคนิคการบีบอัดและเข้ารหัสข้อมูล กับแอปพลิเคชันในปัจจุบันที่ไม่ได้ใช้เทคนิคการบีบอัดและการเข้ารหัสข้อมูล กับข้อมูลที่ไม่ได้ทำการบีบอัดและเข้ารหัส เป็นจำนวน 10 ครั้ง ผลปรากฏว่าข้อมูลที่ทำการบีบอัดและเข้ารหัส มีประสิทธิภาพทางเวลาดีกว่าข้อมูลต้นฉบับประมาณร้อยละ 80



รูปที่ 9 ผลการทดลองวัดค่า Response Time ของข้อมูลประเภทตัวอักษร

จากรูปที่ 10 แสดงผลการทดลองวัดผลจากข้อมูลประเภทรูปภาพขนาด 102 กิโลไบต์ (KB) โดยเปรียบเทียบกันระหว่างแอปพลิเคชันที่ใช้เทคนิคการบีบอัดและเข้ารหัสข้อมูล กับแอปพลิเคชันในปัจจุบันที่ไม่ได้ใช้เทคนิคการบีบอัดและการเข้ารหัสข้อมูล เป็นจำนวน 10 ครั้ง ผลปรากฏว่าข้อมูลที่ทำการบีบอัดและเข้ารหัสมีประสิทธิภาพทางเวลาดีกว่าข้อมูลต้นฉบับประมาณร้อยละ 8



รูปที่ 10 ผลการทดลองวัดค่า Response Time ของข้อมูลประเภทรูปภาพ

5. สรุป

Fixzy เป็นแอปพลิเคชันช่วยจัดการงานซ่อมและการให้บริการในที่พักอาศัย (Home Services Application) ด้วยการทำงานแบบออนไลน์เพื่อการนัดหมายช่าง โดยการนัดหมายช่างแต่ละครั้งจะมีการรับส่งข้อมูลผ่านเว็บเซิร์ฟเวอร์ ซึ่งอาจเกิดปัญหาการรับส่งข้อมูลมีความล่าช้า

เนื่องจากชุดข้อมูลนั้นมีขนาดใหญ่ งานวิจัยนี้จึงได้นำเทคนิคการบีบอัดและเข้ารหัสข้อมูลมาใช้เพื่อแก้ไขปัญหานั้นได้กล่าวมาข้างต้น ซึ่งจากการทดลองสามารถสรุปได้ว่าการนำวิธีการบีบอัดและการเข้ารหัสข้อมูลสามารถนำมาประยุกต์ใช้ในแอปพลิเคชัน Fixzy ได้อย่างมีประสิทธิภาพ โดยการทดลองรับส่งข้อมูลประเภทตัวอักษร สามารถลดเวลาในการรับส่งได้ประมาณร้อยละ 80 และสามารถลดเวลาในการรับส่งข้อมูลประเภทรูปภาพได้ประมาณร้อยละ 8 จากข้อมูลต้นฉบับ และข้อมูลที่ทำการบีบอัดระหว่างเว็บเซิร์ฟเวอร์ REST กับแอปพลิเคชันนั้น ถึงแม้จะมีผู้ไม่หวังดีติดโจรกรรมข้อมูลระหว่างทาง ก็จะไม่สามารถนำข้อมูลนั้นมาใช้ได้เลย เนื่องจากข้อมูลนั้นมีการเข้ารหัสไว้ ทำให้ข้อมูลนั้นมีความปลอดภัย อีกทั้งยังสามารถลดค่าใช้จ่ายการใช้งานแบนด์วิดธ์ของผู้ใช้สามารถรองรับการขยายตัวของผู้ใช้งานแอปพลิเคชันและลดค่าใช้จ่ายในส่วนที่เกี่ยวข้องกับระบบเว็บเซิร์ฟเวอร์ได้อีกด้วย

เอกสารอ้างอิง

- [1] National Statistical Office. Smartphone Owner in Thailand by age group [Online]. Available: <http://service.nso.go.th>
- [2] Hataikam Sirichaisuttikorn, Anurak Choeichum, Chaiyaporn Kemaphalaphun, "Consumer Trading for Smart Grid on Cloud" 2015 SEDT International Symposium on Electrical, Electronic Engineering and Digital Technology.
- [3] Mohamed S. Abdelfattah, Andrei Hagiescu and Deshanand Singh, "High Performance Lossless Data Compression FPGAs using OpenCL" 2014 IWOCCL Proceedings of the International Workshop on OpenCL 2013 & 2014.
- [4] Vishal Pachori, Gunjan Ansañ, Neha Chaudhary, "Improve Performance of Advance Encryption Standard using Parallel Computing" 2012 IJERA International Journal of Engineering Research and Applications.
- [5] Abdullah M. AlFararjeh, Ammany M. Abu Jabal, "Recommendations to improve performance of an enterprise web-based application" 2010 ISWSA Proceedings of the 1st International Conference on Intelligent Semantic Web-Services and Applications.
- [6] Anjali Chauhan, Nimesh Sakhale, Chinmay Desai, Sanket Deore, Rajesh Gaikwad, "Email Sender with Data Compression", [Online]. Available at: www.researchpublish.com
- [7] Brijender Kahanwal, Kanishak Dua and Girish Pal Singh, "Java File Security System (JFSS)" 2012 Global journals Inc.(USA) Global Journal of Computer Science and Technology Network, Web & Security.