

เครื่องมือวิเคราะห์ข้อมูลการใช้งานของลูกค้าบริษัททอมสัน รอยเตอร์

THOMSON REUTERS CLIENT USAGE ANALYSIS TOOLS



สหกิจศึกษาเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ปีการศึกษา 2560
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THOMSON REUTERS CLIENT USAGE ANALYSIS TOOLS




A COOPERATIVE EDUCATION SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)
DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดลอกข้อมูลใดๆ ของเอกสารทุกครั้งที่มีการนำไปใช้

ACADEMIC YEAR 2017

หัวข้อสหกิจศึกษา เครื่องมือวิเคราะห์ข้อมูลการใช้งานของลูกค้าบริษัททอมสัน รอยเตอร์
 Thomson Reuters Client Usage Analysis Tools
 ชื่อนักศึกษา นายเอกณรงค์ อภิพุทธิกุล รหัสนักศึกษา 57050366
 ปริญญา วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)
 ภาควิชา วิทยาการคอมพิวเตอร์
 ปีการศึกษา 2560
 อาจารย์ที่ปรึกษา ผศ.ดร.อนันตพร หารราชคุณาฒย

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2560

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.วิสันต์ ตั้งวงษ์เจริญ ประธานกรรมการ	
ผศ.ดร.อนันตพร หารราชคุณาฒย กรรมการและอาจารย์ที่ปรึกษา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้สิทธิ์ของคณะวิทยาศาสตร์อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา	เครื่องมือวิเคราะห์ข้อมูลการใช้งานของลูกค้าบริษัททอมสัน รอยเตอร์
ชื่อนักศึกษา	นายเอกณรงค์ อภิพุทธิกุล รหัสนักศึกษา 57050366
ปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
คณะ	วิทยาศาสตร์
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา	2560
อาจารย์ที่ปรึกษา	ผศ.ดร.อนันตพร ทรราชคุณาฒย

บทคัดย่อ

ในปัจจุบันบริษัททอมสัน รอยเตอร์ ได้ให้บริการข้อมูลทางการเงินกับลูกค้า ประกอบไปด้วย 3 ระบบได้แก่ ระบบ Analytics Data Cloud (ADC), ระบบ Elektron Time Series (ETS) และระบบ Elektron Tick History (ETH) โดยในแต่ละระบบจะบันทึกรายละเอียดการเรียกใช้ข้อมูลของลูกค้า (Access Log) เพื่อนำมาใช้สำหรับการวิเคราะห์ข้อมูล แต่เนื่องด้วยชุดคำสั่งในการวิเคราะห์ข้อมูลในระบบ ADC ใช้เวลาในการประมวลผลและหน่วยความจำสำรองเป็นจำนวนมาก นักพัฒนาจึงได้พัฒนาชุดคำสั่งขึ้นมาใหม่เพื่อใช้แทนชุดคำสั่งเดิม โดยผลการวิเคราะห์อยู่ในรูปแบบของรายงานและสามารถแสดงผลในรูปแบบของกราฟบน Grafana ในส่วนของระบบ ETS นักวิเคราะห์ต้องการทราบถึงปริมาณการใช้งานของลูกค้าและความคุ้มค่าในการเก็บข้อมูลทันภายในระบบ นักพัฒนาได้พัฒนาชุดคำสั่งที่ทำงานอัตโนมัติสำหรับวิเคราะห์ข้อมูลดังกล่าว โดยผลการวิเคราะห์จะอยู่บนฐานข้อมูล MongoDB และสามารถแสดงผลในรูปแบบกราฟบน Tableau ในส่วนของระบบ ETH นักวิเคราะห์ต้องการทราบถึงปริมาณการใช้งานในแต่ละ Service และมีลูกค้าคนใดเรียกใช้ข้อมูลบ้าง นักพัฒนาจึงได้พัฒนาชุดคำสั่งที่ทำงานอัตโนมัติสำหรับวิเคราะห์ข้อมูลดังกล่าว โดยผลการวิเคราะห์อยู่ในรูปแบบของรายงานและสามารถแสดงผลในรูปแบบของกราฟบน Grafana นอกจากนี้ นักพัฒนาได้พัฒนาโปรแกรมส่วนติดต่อผู้ใช้สำหรับวิเคราะห์ Access Log จากฐานข้อมูล Elasticsearch เพื่อใช้ทดแทน Kibana ที่ไม่สามารถรองรับปริมาณข้อมูลเป็นจำนวนมากได้ จากผลการดำเนินงานนักวิเคราะห์สามารถหารูปแบบการใช้งานของลูกค้า ตรวจสอบการใช้งานที่ผิดปกติ หาสาเหตุที่ทำให้ระบบเกิดความผิดปกติ นำผลลัพธ์จากการวิเคราะห์ไปใช้ในการทดสอบระบบที่พัฒนาขึ้นมาใหม่และใช้สำหรับบริหารจัดการหน่วยความจำสำรองภายในระบบ

คำสำคัญ : การวิเคราะห์ข้อมูล ข้อมูลเชิงอนุกรมเวลา แหล่งข้อมูลขนาดใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Thomson Reuters Client Usage Analysis Tools
Students	Mr. Eknarong Aphiphutthikul Student ID 57050366
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2017
Advisor	Asst.Prof.Dr. Anantaporn Hanskunatai

Abstract

Reuters Software (Thailand) Ltd. is a company providing financial information to the customer. There are three systems in Reuters Software, Analytics Data Cloud (ADC), Elektron Time Series (ETS), and Elektron Tick History (ETH). Each of the systems will log the information when user access the data which is called Access log. The Access Log uses to analyse information. The process of analysing data in ADC takes a lot of time and processing units. Developer eliminates those problems by creating the new way of processing. The result of analysis will showed in report and a graph will showed on Grafana. For ETS system, the information analyst requires the information about customer data usage and examines which stock information should be kept on the system. The result of analysed data will be stored on MongoDB and a graph will showed on Tableau. For ETH system, the information analyst requires the information about the usage of each service and the data of the customer who access to the service. The result of analysed data will display by a graph on Grafana. In addition, the developer also developed an interface to analyse Access Log from Elasticsearch database. The new application is developed to replace Kibana since Kibana couldn't store the huge of data. With the new application, the data analyst can find the pattern usage of customers, detect the misuse of the system from customers, test the new system, and manage the disk resources of the system.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 Keywords: Data Analysis, Time Series Data, Big Data

กิตติกรรมประกาศ

จากการที่ข้าพเจ้าได้ไปทำสหกิจศึกษาหัวข้อ เครื่องมือวิเคราะห์ข้อมูลการใช้งานของลูกค้า บริษัททอมสัน รอยเตอร์ สามารถดำเนินงานสำเร็จได้ด้วยดีเนื่องจากได้รับความช่วยเหลือ การสนับสนุน และความกรุณาจากองค์กรและบุคคลหลายท่าน คณะผู้จัดทำขอกล่าวคำขอบคุณองค์กร และบุคคลดังต่อไปนี้

ขอขอบพระคุณ ผศ.ดร.อนันตพร หารรัชคุณาฉัย อาจารย์ที่ปรึกษาสหกิจศึกษา ที่ได้ให้ คำปรึกษาพร้อมทั้งแนวทางแก้ปัญหา รวมทั้งตรวจแก้สหกิจศึกษาฉบับนี้ให้มีความสมบูรณ์เพิ่มมากขึ้น

ขอขอบพระคุณอาจารย์ภาคิณีวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้ความรู้และประสบการณ์ตลอดระยะเวลาที่ได้รับ การศึกษา

ขอขอบพระคุณ นายวรัทม์ ชัยเกษตรสิน และ นางสาวนิกานต์ พัฒนวานิชย์สกุล ผู้ดูแล และให้คำปรึกษาข้าพเจ้าขณะฝึกงานอยู่ที่บริษัทรอยเตอร์ซอฟต์แวร์ (ประเทศไทย) จำกัด

ขอขอบพระคุณ บริษัทรอยเตอร์ซอฟต์แวร์ (ประเทศไทย) จำกัด ที่ให้โอกาสในการเข้ารับ การฝึกงานและได้รับประสบการณ์ในการทำงานในสภาพแวดล้อมจริง

เอกณรงค์ อภิพุทธิกุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญรูป.....	ซ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการสหกิจศึกษา.....	1
1.3 ขอบเขตของโครงการสหกิจศึกษา.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.5 เครื่องมือและภาษาที่ใช้ในโครงการสหกิจศึกษา.....	2
บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการพัฒนาระบบ.....	4
2.1 การวิเคราะห์ข้อมูล.....	4
2.1.1 การแยกประเภทข้อมูล.....	4
2.1.2 การสังเขปข้อมูล.....	4
2.1.3 การหาข้อสรุปเกี่ยวกับลักษณะต่างๆของข้อมูล.....	5
2.1.4 แนวโน้มของข้อมูล.....	6
2.1.5 การพยากรณ์ทางสถิติ.....	7
2.2 โครงสร้างข้อมูล.....	7
2.2.1 JSON.....	7
2.2.2 XML.....	8
2.3 ภาษาไพธอน.....	9
2.3.1 คุณลักษณะเด่นของภาษาไพธอน.....	9
2.3.2 ชนิดของตัวแปร.....	10
2.4 ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ.....	15
2.4.1 Elasticsearch.....	15
2.4.2 MongoDB.....	16
2.4.3 Graphite.....	18
2.4.4 Tableau.....	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.4.5 Grafana	20
2.4.6 Kibana	22
2.4.7 Sensu	24
2.4.8 PyCharm	24
2.4.9 AppStatsParser	25
2.4.10 mRemoteNG	25
2.4.11 pscp	26
2.4.12 Map Network Drive	27
2.4.13 Qt	28
บทที่ 3 ขั้นตอนในการพัฒนาระบบ	29
3.1 ระบบ Analytics Data Cloud (ADC)	29
3.1.1 ขั้นตอนการศึกษาการทำงานของระบบ ADC	29
3.1.2 ขั้นตอนการวิเคราะห์การทำงานรูปแบบเดิม	30
3.1.3 ขั้นตอนการวิเคราะห์และออกแบบการทำงานรูปแบบใหม่	31
3.1.4 ขั้นตอนการพัฒนาซอฟต์แวร์	36
3.2 ระบบ Elektron Time Series (ETS)	36
3.2.1 ขั้นตอนการศึกษาการทำงานของระบบ ETS	36
3.2.2 ขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์	38
3.2.3 ขั้นตอนการพัฒนาซอฟต์แวร์	43
3.3 ระบบ Elektron Tick History (ETH)	44
3.3.1 ขั้นตอนการศึกษาการทำงานของระบบ ETH	44
3.3.2 ขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์	45
3.3.3 ขั้นตอนการพัฒนาซอฟต์แวร์	48
3.4 โปรแกรม Access Log Analysis From Elasticsearch	48
3.4.1 ขั้นตอนการศึกษาและรวบรวมข้อมูล	48
3.4.2 ขั้นตอนการวิเคราะห์ข้อมูล	49
3.4.3 ขั้นตอนการออกแบบซอฟต์แวร์	50
3.4.4 ขั้นตอนการพัฒนาซอฟต์แวร์	56
บทที่ 4 ผลการดำเนินงาน	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 ระบบ Analytics Data Cloud (ADC)

57

สารบัญ (ต่อ)

	หน้า
4.2 ระบบ Elektron Time Series (ETS).....	63
4.3 ระบบ Elektron Tick History (ETH).....	66
4.4 โปรแกรม Access Log Analysis From Elasticsearch.....	69
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	79
5.1 สรุปผลการดำเนินงาน.....	79
5.2 ข้อจำกัด.....	80
5.3 ข้อเสนอแนะ	80
เอกสารอ้างอิง	81
ภาคผนวก.....	82
ภาคผนวก ก การติดตั้งเครื่องมือเพื่อใช้ในการพัฒนาโปรแกรม	83



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 เปรียบเทียบโครงสร้างระหว่าง RDBMS กับ Elasticsearch	15
2.2 เปรียบเทียบโครงสร้างระหว่าง RDBMS กับ MongoDB.....	16
4.1 เปรียบเทียบการใช้หน่วยความจำสำรองและเวลาในการประมวลผลระหว่างชุดคำสั่งเดิม กับชุดคำสั่งใหม่	57
4.2 การประเมินผลการพัฒนาระบบวิเคราะห์ Access Log ของระบบ ADC	62
4.3 การประเมินผลการพัฒนาระบบวิเคราะห์ข้อมูลของระบบ ETS	66
4.4 การประเมินผลการพัฒนาระบบวิเคราะห์ Access Log ของระบบ ETH	69
4.5 การประเมินผลการพัฒนาโปรแกรม Access Log Analysis From Elasticsearch	78



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 ตัวอย่าง JSON.....	8
2.2 ตัวอย่าง XML.....	9
2.3 ตัวอย่างคำสั่งสอบถามชนิดตัวแปร.....	10
2.4 ตัวอย่างคำสั่งตัวแปรชนิดตัวเลข.....	11
2.5 ตัวอย่างคำสั่งตัวแปรชนิดสายอักขร.....	12
2.6 ตัวอย่างคำสั่งตัวแปรชนิดลิสต์.....	13
2.7 ตัวอย่างคำสั่งตัวแปรชนิดทUPLE.....	14
2.8 ตัวอย่างคำสั่งตัวแปรชนิดดิกชันนารี.....	14
2.9 Logo ของ Elasticsearch.....	15
2.10 Logo ของ MongoDB.....	16
2.11 การทำงานของ MongoDB Connector for BI.....	17
2.12 โปรแกรม Robo 3T.....	17
2.13 สถาปัตยกรรมของ Graphite.....	18
2.14 สถาปัตยกรรมของ Tableau.....	19
2.15 Tableau Desktop.....	19
2.16 ตัวอย่างการแสดงผลกราฟจากหลายแหล่งข้อมูลใน Grafana.....	20
2.17 ตัวอย่างการแจ้งเตือนบน Dashboard ใน Grafana.....	21
2.18 ตัวอย่างฐานข้อมูลที่ Grafana รองรับการเชื่อมต่อ.....	21
2.19 ตัวอย่างการสร้าง Dashboard ใน Grafana.....	22
2.20 ตัวอย่างการใช้ Machine Learning ใน Kibana.....	23
2.21 ตัวอย่างการสร้าง Dashboard ใน Kibana.....	23
2.22 Logo ของ Senu.....	24
2.23 Logo ของ PyCharm.....	25
2.24 Logo ของ mRemoteNG.....	26
2.25 ตัวอย่างการถ่ายโอนไฟล์จาก Windows ไปสู่ Linux โดยใช้ pscp.....	27
2.26 การ Map Network Drive บนระบบปฏิบัติการ Windows.....	27
2.27 Logo ของ Qt.....	28
3.1 การจัดเก็บ Access Log ของระบบ ADC.....	29
3.2 การวิเคราะห์ Access Log ของระบบ ADC ในรูปแบบเดิม.....	30
3.3 การวิเคราะห์ Access Log ของระบบ ADC ที่ได้รับการพัฒนา.....	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีใช้งานเพื่อการศึกษาเท่านั้น โปรดอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.4 Use Case Diagram กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler และ กลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ของระบบ ADC.....	33
3.5 Sequence Diagram กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler ของ ระบบ ADC.....	34
3.6 Sequence Diagram กลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ของระบบ ADC	35
3.7 การจัดเก็บและการสรุปปริมาณข้อมูลหุนที่เข้าสู่ระบบ ETS.....	37
3.8 การจัดเก็บ Access Log ของระบบ ETS.....	37
3.9 ตัวอย่างไฟล์ Rule ของระบบ ETS	38
3.10 ตัวอย่าง Feed Log Analysis ของระบบ ETS	39
3.11 ตัวอย่างบันทึกการเรียกใช้ข้อมูลของลูกค้าจำนวน 1 แถวของระบบ ETS.....	39
3.12 การวิเคราะห์ข้อมูลของระบบ ETS	40
3.13 Use Case Diagram การวิเคราะห์ข้อมูลของระบบ ETS.....	41
3.14 Sequence Diagram การวิเคราะห์ข้อมูลของระบบ ETS.....	42
3.15 การจัดเก็บ Access Log ของระบบ ETH.....	44
3.16 การวิเคราะห์ Access Log ของระบบ ETH.....	45
3.17 Use Case Diagram การวิเคราะห์ Access Log ของระบบ ETH.....	46
3.18 Sequence Diagram การวิเคราะห์ Access Log ของระบบ ETH	47
3.19 Use Case Diagram ของโปรแกรม Access Log Analysis From Elasticsearch	51
3.20 Sequence Diagram ขั้นตอนการดาวน์โหลดและวิเคราะห์ Access Log จากฐานข้อมูล Elasticsearch ด้วยโปรแกรม Access Log Analysis From Elasticsearch.....	52
3.21 Context Diagram ของโปรแกรม Access Log Analysis From Elasticsearch.....	53
3.22 Data Flow Diagram level 0 ของโปรแกรม Access Log Analysis From Elasticsearch	54
3.23 Data Flow Diagram level 1 กระบวนการ Query Data ของโปรแกรม Access Log Analysis From Elasticsearch.....	55
3.24 Data Flow Diagram level 1 กระบวนการ Write Data ของโปรแกรม Access Log Analysis From Elasticsearch.....	56
4.1 ตัวอย่าง Daily Report กลุ่มผลิตภัณฑ์ ของระบบ ADC.....	58
4.2 ตัวอย่าง Daily Report กลุ่มลูกค้า ของระบบ ADC	59
4.3 การแบ่งกลุ่ม KPI Report ของระบบ ADC.....	59

สารบัญญรูป (ต่อ)

รูปที่	หน้า
4.4 ตัวอย่าง matrix ในไฟล์ KPI Report ของระบบ ADC.....	60
4.5 ตัวอย่าง Monthly Report ของระบบ ADC	60
4.6 ตัวอย่างการแสดงผลข้อมูลบน Grafana ของระบบ ADC.....	61
4.7 ตัวอย่างข้อมูลใน FLA_daily_Record Collection บนฐานข้อมูล MongoDB ของระบบ ETS.....	63
4.8 ตัวอย่างแผนภูมิแท่งแสดงผลข้อมูลใน FLA_daily_Record Collection โดยใช้ Tableau ของระบบ ETS	64
4.9 ตัวอย่างข้อมูลใน RuleRic Collection บนฐานข้อมูล MongoDB ของระบบ ETS	65
4.10 ตัวอย่างข้อมูลใน timeseries_daily_usage Collection บนฐานข้อมูล MongoDB ของระบบ ETS.....	65
4.11 ตัวอย่าง Daily Report กลุ่มเซิร์ฟเวอร์ของระบบ ETH.....	67
4.12 ตัวอย่าง Daily Report กลุ่มข้อมูลหุ้นของระบบ ETH	67
4.13 ตัวอย่าง KPI Report ของระบบ ETH	68
4.14 ตัวอย่างการแสดงผลข้อมูลบน Grafana ของระบบ ETH.....	68
4.15 Tab General ของโปรแกรม Access Log Analysis From Elasticsearch.....	70
4.16 Tab Config ของโปรแกรม Access Log Analysis From Elasticsearch	70
4.17 หน้าต่าง Add Compass Host ของโปรแกรม Access Log Analysis From Elasticsearch	71
4.18 ตัวอย่าง Compass Host ที่เพิ่มในตาราง Compass Host ของโปรแกรม Access Log Analysis From Elasticsearch.....	71
4.19 ตัวอย่างการ Add Log ของโปรแกรม Access Log Analysis From Elasticsearch.....	72
4.20 หน้าต่าง Edit Log ของโปรแกรม Access Log Analysis From Elasticsearch.....	73
4.21 ตัวอย่าง Log ที่เพิ่มในตาราง Log ของโปรแกรม Access Log Analysis From Elasticsearch	73
4.22 ตัวอย่างการกำหนดเงื่อนไขในส่วนของ query ใน Tab ‘General’.....	74
4.23 ตัวอย่างการเพิ่มข้อมูลลงในตาราง Report	75
4.24 หน้าต่างการเลือก Path และตั้งชื่อไฟล์ของการสร้างรายงาน	75
4.25 หน้าต่าง Process Working เพื่อแสดงสถานะการทำงานของการสร้าง Report	76
4.26 ตัวอย่าง Report ของโปรแกรม Access Log Analysis From Elasticsearch.....	76
4.27 ตัวอย่างการเพิ่ม Field ลงในตาราง Raw Data.....	76

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.28 หน้าต่างการเลือก Path, ตั้งชื่อไฟล์ และกำหนดชนิดของไฟล์ของการดาวน์โหลด Raw Data	77
4.29 หน้าต่าง Process Working เพื่อแสดงสถานะการทำงานของ การดาวน์โหลด	77
4.30 ตัวอย่าง Raw Data ประเภทไฟล์เป็น CSV ของการดาวน์โหลด Raw Data	77
4.31 ตัวอย่าง Raw Data ประเภทไฟล์เป็น JSON ของการดาวน์โหลด Raw Data	78
ก.1 หน้าต่างติดตั้งของภาษา Python	83
ก.2 หน้าต่างสำหรับกำหนด Path ในการติดตั้งภาษา Python	83
ก.3 หน้าต่างตั้งค่า Environment Variables	84
ก.4 การตั้งค่า Path ภาษา Python บนเครื่องคอมพิวเตอร์ระบบปฏิบัติการ Windows	84
ก.5 หน้าต่างติดตั้งของโปรแกรม PyCharm	85
ก.6 หน้าต่างสำหรับกำหนด Path ในการติดตั้งโปรแกรม PyCharm	85
ก.7 หน้าต่างติดตั้งของฐานข้อมูล MongoDB	86
ก.8 หน้าต่างเลือกประเภทของการติดตั้งฐานข้อมูล MongoDB	86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันบริษัท Thomson Reuters เป็นบริษัทที่ให้บริการข้อมูลทางการเงินกับลูกค้า ซึ่งความต้องการข้อมูลในแต่ละวันมีปริมาณมหาศาล ดังนั้นระบบที่ให้บริการข้อมูลกับลูกค้าต้องสามารถประมวลผลและรองรับการใช้งานของลูกค้าในปริมาณมากได้ ระบบที่ให้บริการข้อมูลนั้น ประกอบไปด้วย Analytics Data Cloud (ADC) เป็นระบบที่ให้บริการข้อมูลเกี่ยวกับรายละเอียดต่างๆของบริษัทที่มีหุ้นอยู่ในตลาด เช่น กำไร ขาดทุน จำนวนหุ้นที่มีอยู่ เป็นต้น เพื่อช่วยในการตัดสินใจซื้อขายหุ้นของลูกค้า Elektron Time Series (ETS) เป็นระบบที่ให้บริการข้อมูลการซื้อขายหุ้นต่างๆในตลาด โดยจะเก็บข้อมูลเพียง 3 เดือนนับจากปัจจุบันเท่านั้น Elektron Tick History (ETH) เป็นระบบที่ให้บริการข้อมูลเหมือนกับระบบ ETS แต่จะเก็บข้อมูลมาตั้งแต่ปี 1996 จนถึงปัจจุบัน ในแต่ละระบบได้จัดเก็บการเรียกใช้ข้อมูลของลูกค้า (Access Log) ทุกครั้งที่มีการเรียกใช้ข้อมูลผ่านระบบ ซึ่งมีรายละเอียดต่างๆ เช่น วันเวลาที่ลูกค้าเรียกใช้ข้อมูล ประเภทเครื่องมือที่ลูกค้าเรียกใช้ข้อมูล ข้อมูลที่ลูกค้าเรียกใช้ เป็นต้น

การนำ Access Log มาวิเคราะห์นั้น ในปัจจุบันยังไม่ครอบคลุมทุกระบบ และบางระบบมีปัญหาเกี่ยวกับวิธีการนำ Access Log มาวิเคราะห์ ซึ่งส่งผลให้ไม่สามารถวิเคราะห์ Access Log ได้ตามความต้องการ ดังนั้นจึงต้องมีการพัฒนาโปรแกรมและขั้นตอนการดำเนินงานเพื่อใช้ในการวิเคราะห์ Access Log ให้มีประสิทธิภาพและตอบสนองต่อความต้องการของนักวิเคราะห์ โดยจะนำผลการวิเคราะห์ Access Log ที่อยู่ในรูปแบบรายงานและกราฟมาใช้ในการตรวจสอบและ พัฒนาระบบให้มีประสิทธิภาพมากยิ่งขึ้น

1.2 วัตถุประสงค์ของโครงการสหกิจศึกษา

- 1) เพื่อพัฒนาขั้นตอนการดำเนินงานในการวิเคราะห์การใช้ข้อมูลของลูกค้า
- 2) เพื่อพัฒนาโปรแกรมที่ใช้ในการวิเคราะห์การใช้ข้อมูลของลูกค้า
- 3) เพื่อให้สามารถนำผลการวิเคราะห์ไปใช้ในการปรับปรุงและพัฒนาการทำงานของระบบหรือใช้ในการทดสอบระบบที่สร้างขึ้นใหม่ เพื่อให้สามารถรองรับการใช้งานของลูกค้าได้
- 4) เพื่อพัฒนาโปรแกรมที่ใช้ในการค้นหาข้อมูลแทน Kibana ที่ไม่สามารถรองรับปริมาณข้อมูลมหาศาลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงการงานสหกิจศึกษา

- 1) พัฒนาและปรับปรุง Scripts เพื่อใช้ในการวิเคราะห์ Access Log ในระบบ ADC โดยผลการวิเคราะห์ให้ออกมาในรูปแบบรายงานและแสดงผลบน Grafana
- 2) พัฒนา Scripts เพื่อใช้ในการวิเคราะห์ Access Log ในระบบ ETS โดยผลการวิเคราะห์ให้แสดงผลบน Tableau
- 3) พัฒนา Scripts เพื่อใช้ในการวิเคราะห์ Access Log ในระบบ ETH โดยผลการวิเคราะห์ให้ออกมาในรูปแบบรายงานและแสดงผลบน Grafana
- 4) พัฒนาโปรแกรมส่วนติดต่อผู้ใช้งาน (Graphic User Interface) เพื่อค้นหาและวิเคราะห์ข้อมูลจากฐานข้อมูล Elasticsearch

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ประโยชน์ต่อตนเอง
 - 1.1) มีความรู้ความเข้าใจระบบจัดการฐานข้อมูลขนาดใหญ่
 - 1.2) ได้ศึกษาและใช้ซอฟต์แวร์ที่ตนเองไม่เคยใช้
 - 1.3) ได้ประสบการณ์ในการทำงานจริง
- 2) ประโยชน์ต่อองค์กร
 - 2.1) ช่วยลดข้อผิดพลาดในการวิเคราะห์ข้อมูล
 - 2.2) ช่วยลดเวลาในการวิเคราะห์ข้อมูล
 - 2.3) ช่วยให้สามารถวิเคราะห์ข้อมูลได้ตรงตามความต้องการมากขึ้น

1.5 เครื่องมือและภาษาที่ใช้ในโครงการงานสหกิจศึกษา

- 1) ฮาร์ดแวร์
 - 1.1) เครื่องคอมพิวเตอร์
 - หน่วยประมวลผลกลาง (CPU) Intel(R) Core(TM) i5-3340M @2.70 GHz
 - หน่วยความจำหลัก (RAM) 8 GB
 - หน่วยความจำสำรอง 120 GB
 - การ์ดจอ (Graphic processor) Intel HD Graphics 4000
 - 1.2) เครื่องเซิร์ฟเวอร์
 - หน่วยประมวลผลกลาง (CPU) Intel(R) Core(TM) i7-3770 @ 3.4 GH
 - หน่วยความจำหลัก (RAM) 24 GB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ซอฟต์แวร์

- 2.1) JetBrains PyCharm Community Edition 2017.1.3 ใช้ในการเขียนโปรแกรม
- 2.2) mRemote NG ใช้ในการเข้าถึงเครื่องเซิร์ฟเวอร์
- 2.3) Microsoft Office Excel 2007 ใช้ในการอ่านไฟล์ csv และ ไฟล์ excel
- 2.4) Kibana, Dashboard Grafana, Tableau ใช้ในการแสดงผลข้อมูลการใช้
- 2.5) baretail ใช้ในการอ่านไฟล์ Log
- 2.6) MongoDB ใช้เป็นฐานข้อมูล
- 2.7) Robo 3T ใช้จัดการฐานข้อมูล MongoDB
- 2.8) Cisco AnyConnect Secure Mobility Client ใช้เชื่อมต่อ Virtual Private Network
- 2.9) draw.io ใช้ออกแบบระบบในรูปแบบของ Diagram

3) ภาษา

- 3.1) Python 2.7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทฤษฎีและเทคโนโลยีที่ใช้ในการพัฒนาระบบ

ในการพัฒนาโปรแกรมและขั้นตอนการดำเนินงานเพื่อใช้ในการวิเคราะห์ Access Log ผู้พัฒนาได้ทำการศึกษาหลักการทฤษฎีต่างๆ และเทคโนโลยีที่เกี่ยวข้องที่สามารถนำไปประยุกต์ใช้งาน พร้อมทั้งนำความรู้ที่ได้ไปใช้ควบคู่กับการพัฒนาโปรแกรม โดยแบ่งเป็นหัวข้อต่างๆ ดังต่อไปนี้

- 2.1 การวิเคราะห์ข้อมูล (Data Analysis)
- 2.2 โครงสร้างข้อมูล
- 2.3 ภาษาไพธอน (Python)
- 2.4 ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ

2.1 การวิเคราะห์ข้อมูล (Data Analysis)

การวิเคราะห์ข้อมูล (Data Analysis) คือ การเก็บรวบรวมข้อมูล จัดการข้อมูลด้วยวิธีการต่างๆ เช่น การแยกประเภท การจัดชั้น การสังเขป การหาข้อสรุปเกี่ยวกับลักษณะต่างๆ ของข้อมูล การพิจารณาหาว่า ข้อมูลที่เก็บรวบรวมมาได้ มีความสัมพันธ์กับข้อมูลอื่นอย่างไร ตลอดจนการพยากรณ์เหตุการณ์ในอนาคต ซึ่งกระบวนการต่างๆ ของการวิเคราะห์ข้อมูล แยกกล่าวเป็นข้อๆ ได้ดังต่อไปนี้

2.1.1 การแยกประเภทข้อมูล (Classification)

ข้อมูลที่เก็บรวบรวมมาได้ อาจมีมากประเภท หรือน้อยประเภท แล้วแต่เรื่องที่ต้องการศึกษา ข้อมูลแต่ละประเภทอาจจำแนกเป็นชนิดย่อยออกไปอีก ทั้งนี้เพื่อจะได้ศึกษาในรายละเอียดปลีกย่อยต่างๆ ให้มากขึ้นตามความต้องการ ซึ่งการแยกประเภทข้อมูล อาจเป็นขั้นตอนที่พอเพียงสำหรับการวิเคราะห์ข้อมูลบางอย่าง ซึ่งไม่ต้องการศึกษาข้อมูลในชั้นลึกซึ้งนัก แต่สำหรับการศึกษาบางอย่าง การแยกประเภทข้อมูลเป็นเพียงขั้นเตรียมงานเท่านั้น ซึ่งในประการหลังนี้ ลักษณะต่างๆ ของข้อมูลทั้งที่เป็นรายข้อมูล และในส่วนรวมจะต้องได้รับการพิจารณา และศึกษาอย่างละเอียด

2.1.2 การสังเขปข้อมูล (Condensation)

ข้อมูลที่เก็บรวบรวมมาได้ในขั้นแรกจะอยู่ในสภาพที่ไม่เป็นระเบียบเรียบร้อย ยังไม่สามารถทำการหาข้อสรุปของลักษณะต่างๆ ของข้อมูลได้ ข้อมูลที่อยู่ในรูปเช่นนี้มีชื่อเรียกว่า "ข้อมูลดิบ" (Raw data) ดังนั้น เพื่อให้ข้อมูลอยู่ในสภาพพร้อม ที่จะสามารถหาข้อสรุป หรือทำการวิเคราะห์โดยวิธีอื่นๆ ได้ จึงอาจดำเนินการสังเขปข้อมูลดิบ หรือจัดข้อมูลดิบทั้งสิ้น ให้อยู่ในรูปแบบใหม่ ซึ่งเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระเปียบเรียบร้อย และมีขนาดกะทัดรัดสะดวกต่อการดำเนินการวิเคราะห์ในขั้นต่อไป การสังเขปข้อมูลตามที่กล่าวมานี้เรียกว่า "การแจกแจงความถี่" (Frequency distribution) ในกรณีที่ข้อมูลมีจำนวนไม่มากนัก การแจกแจงความถี่อาจแสดงเป็นรายข้อมูลที่เก็บรวบรวมมาได้ พร้อมทั้งความถี่หรือจำนวนที่ซ้ำๆ กันของข้อมูลตัวนั้นๆ ถ้าข้อมูลมีจำนวนมาก ทำให้ต้องเสียเนื้อที่กระดาษ และไม่สะดวก ที่จะทำการวิเคราะห์ในขั้นต่อไป นอกจากนี้ถ้าจะพิจารณาในด้านการกระจายของข้อมูลว่า ข้อมูลส่วนใหญ่มีค่าระหว่างเท่าไรถึงเท่าไร หรือรูปลักษณะการกระจายข้อมูลเป็นอย่างไร ก็จะไม่สามารถแลเห็นรูปลักษณะของการกระจายได้ชัดเจน ดังนั้น จึงอาจจะจัดทำการแจกแจงความถี่ของข้อมูลให้อยู่ในรูปแบบใหม่ กล่าวคือ จัดข้อมูลให้รวมอยู่เป็นกลุ่มๆ แล้วดูว่าข้อมูลค่าต่างๆ อยู่ในกลุ่มหรือชั้นใดบ้าง ให้นับจำนวนข้อมูลเหล่านั้นว่า อยู่ในชั้นใดเป็นจำนวนเท่าใด จำนวนข้อมูลในแต่ละชั้นเรียกว่า ความถี่

อย่างไรก็ตาม ถ้าจัดกลุ่มข้อมูลแต่ละกลุ่มให้มีขนาดใหญ่มาก จำนวนกลุ่ม หรือจำนวนชั้นก็จะน้อย และจำนวนข้อมูลในแต่ละกลุ่ม ก็จะอัดกันแน่นมากเกินไป ทำให้ไม่สามารถแลเห็นลักษณะการกระจายของข้อมูลได้ดี แต่ถ้าหากจัดแต่ละกลุ่มให้มีขนาดเล็กมาก จำนวนกลุ่ม หรือจำนวนชั้นก็จะมาก ซึ่งจะทำให้การกระจายของข้อมูลมีลักษณะแผ่กว้างมากเกินไป ไม่ช่วยให้แลเห็นลักษณะการกระจายที่ดี ดังนั้น การแจกแจงความถี่แบบจัดข้อมูลให้เป็นกลุ่มๆ นี้ จึงต้องให้แต่ละกลุ่มมีขนาดพอดี ที่จะทำให้แลเห็นลักษณะการกระจายของข้อมูลได้อย่างเด่นชัด กล่าวคือ ข้อมูลในกลุ่มจะต้องไม่อัดแน่นมากเกินไป และก็ต้องไม่แผ่กระจายมากจนเกินไปด้วย

2.1.3 การหาข้อสรุปเกี่ยวกับลักษณะต่างๆ ของข้อมูล (Sumarization)

ข้อมูลแต่ละชุดที่เก็บรวบรวมมาได้ อาจมีลักษณะการแจกแจงความถี่แตกต่างกันไป ดังได้กล่าวแล้วในข้อ 2.1.2 ในการวิเคราะห์ข้อมูล เราจำเป็นต้องศึกษาอย่างละเอียดว่า ข้อมูลชุดนั้นๆ บอกอะไรแก่เราบ้าง เช่น สมมติว่า มีข้อมูลเกี่ยวกับรายได้ต่อปีของคนจำนวนหนึ่ง ซึ่งเป็นตัวอย่างของประชากรทั้งประเทศ สิ่งต่างๆ ที่อาจต้องการทราบก็คือ ประชากรมีรายได้ต่อปีเฉลี่ยคนละเท่าไร รายได้ของคนมี และคนยากจนแตกต่างกันมากหรือไม่ และถ้าคนส่วนใหญ่ค่อนข้างยากจน คนเหล่านี้มีมากเพียงไร ค่าเหล่านี้คือ ค่าซึ่งบอกลักษณะต่างๆ ของข้อมูล ซึ่งเป็นค่าสถิติอย่างหนึ่ง และสามารถคำนวณหาได้ ดังนั้นการหาค่าที่บอกลักษณะสำคัญของข้อมูลที่มีการกระจาย ได้แก่ การหาค่ากลาง และการหาค่าการกระจายของข้อมูล ซึ่งจะได้กล่าวเป็นเรื่องๆ ต่อไป

1) การหาค่ากลางของข้อมูล

- มัชฌิมเลขคณิต คือ ค่าเฉลี่ยของข้อมูลชุดนั้น ซึ่งอาจเขียนให้อยู่ในรูปสูตรได้ดังนี้ มัชฌิมเลขคณิต = ผลบวกของทุกค่าของข้อมูล/จำนวนข้อมูล
- มัธยฐาน คือ คะแนนหรือค่า ณ ตำแหน่งกึ่งกลางของข้อมูลชุดนั้น ซึ่งข้อมูลทั้งชุดได้รับการเรียงลำดับแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฐานนิยม คือ คะแนนหรือค่าที่มีความถี่สูงสุด หรือซ้ำกันมากที่สุดของข้อมูลชุดนั้น

2) การหาค่าการกระจายของข้อมูล

การหาค่าการกระจายของข้อมูลที่ใช้กันทั่วไป คือ ส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation) ซึ่งหาได้จากการเอาผลต่างระหว่างข้อมูลแต่ละค่า และมีชดเชยเลขคณิตของข้อมูลนั้นมายกกำลังสอง แล้วทำการถัวเฉลี่ยค่ากำลังสองเหล่านั้น จากนั้นจึงถอดกรณฑ์ที่สองของค่าเฉลี่ยที่ได้

2.1.4 แนวโน้มของข้อมูล (Trend)

ข้อมูลบางชนิดแสดงถึงเหตุการณ์ที่เกิดขึ้นเป็นรายคาบเวลา ข้อมูลประเภทนี้เรียกว่า ข้อมูลอนุกรมเวลา (Time-series data) ถ้านำข้อมูลประเภทนี้ที่เกิดขึ้นในช่วงเวลาหนึ่งซึ่งยาวนานพอสมควรมาลงจุดจะได้เส้นกราฟ ซึ่งมีลักษณะโดยส่วนรวมอาจชันขึ้น หรือลาดลง หรือมีทั้งชันขึ้นหรือลาดลงในช่วงเวลาหนึ่งเช่นในรอบ 1 ปี เป็นต้น ลักษณะโดยส่วนรวมที่ชันขึ้นหรือลาดลงของเส้นกราฟในช่วงเวลายาวนานนี้ เรียกว่า แนวโน้มของข้อมูล วิธีการหาแนวโน้มของข้อมูลอาจแบ่งได้เป็น 2 วิธีใหญ่ๆ คือ

1) การลากเส้นอย่างอิสระ (Freehand method)

การสร้างแนวโน้มด้วยวิธีนี้ คือ การลากเส้น ซึ่งเป็นแนวเรียบผ่านไปในระหว่างเส้นกราฟของข้อมูล ซึ่งปกติจะมีบางตอนหักเหขึ้น และบางตอนหักเหลง เส้นซึ่งเป็นแนวเรียบนี้แสดงถึงความเป็นไปโดยส่วนรวมในระยะยาวของเหตุการณ์ทั้งหมดที่เกิดขึ้น ซึ่งเราเรียกว่า แนวโน้มของข้อมูล การลากเส้นอย่างอิสระนี้ไม่มีกฎเกณฑ์ใดๆ ทั้งสิ้น นอกจากคอยระมัดระวังให้แนวโน้ม แสดงถึงเหตุการณ์ที่เกิดขึ้นในระยะยาวได้ถูกต้องเท่านั้น

2) การถัวเฉลี่ยเคลื่อนที่ (Moving average method)

สมมติว่า มีข้อมูลจำนวนหนึ่ง ซึ่งเก็บรวบรวมไว้เป็นรายเดือน ถ้าเราหาค่ามัชฌิมเลขคณิตของข้อมูลในช่วงเวลาหนึ่ง เช่น ในรอบ 3 เดือน เป็นต้น แล้วจดค่านี้ไว้ จากนั้นก็เลื่อนช่วงเวลา 3 เดือนนี้ต่อไป โดยตัดข้อมูลตัวแรกออก และเพิ่มข้อมูลตัวที่ 4 เข้ามา แล้วก็หามัชฌิมเลขคณิตของช่วงเวลาดังกล่าวนี้อีก และจดค่าที่คำนวณได้ไว้ ทำเช่นนี้เรื่อยๆ ไป ก็จะได้ข้อมูลชุดใหม่ ซึ่งเป็นค่ามัชฌิมเลขคณิตของข้อมูลทุกๆ 3 เดือนต่อเนื่องกัน วิธีการเช่นนี้เรียกว่า การถัวเฉลี่ยเคลื่อนที่ 3 เดือน ในข้อมูลบางชุด อาจทำการถัวเฉลี่ยเคลื่อนที่ทุก 5 เดือน หรือทุก 3 ปี หรือทุก 5 ปีก็ได้ ทั้งนี้ขึ้นอยู่กับชนิด และลักษณะของข้อมูล เมื่อนำค่ามัชฌิมเลขคณิตเคลื่อนที่ที่ได้นี้มาลงจุด แล้วโยงจุดต่างๆ เหล่านี้ ก็จะได้เส้นซึ่งแสดงแนวโน้มของข้อมูลเป็นรายเดือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การพยากรณ์ทางสถิติ

เป็นการพยากรณ์โดยใช้ข้อมูลสถิติประเภทที่เรียกว่า ข้อมูลอนุกรมเวลาเป็นเครื่องมือ การพยากรณ์โดยวิธีนี้ จะต้องศึกษาถึงพฤติกรรมของเรื่องนั้นๆ ที่เกิดขึ้นในอดีตว่ามีลักษณะอย่างไรเสียก่อน แล้วจึงทำการพยากรณ์ ข้อมูลอนุกรมเวลาจะบอกให้ทราบถึงพฤติกรรมนั้นๆ ตัวอย่างของการพยากรณ์ทางสถิติในเรื่องที่เกี่ยวกับเศรษฐกิจ และธุรกิจ ได้แก่ การพยากรณ์จำนวนประชากรของประเทศ การพยากรณ์ผลผลิตทางการเกษตร ปริมาณการขาย ระดับราคาสินค้า ฯลฯ เป็นต้น

การพยากรณ์ทางสถิติจะทำได้ต่อเมื่อพฤติกรรมที่เกิดขึ้นในอดีตมีความแปรผันตามปกติ เช่น แต่ละปีที่ผ่านมาไป จำนวนประชากรของโลกมีแนวโน้มเพิ่มขึ้นเรื่อยๆ หรืออัตราการตายของเด็กอายุต่ำกว่า 1 ปี มีแนวโน้มลดลงอย่างสม่ำเสมอ หรือสินค้าเครื่องกันหนาวจะขายได้มากในฤดูหนาว แต่จะขายได้น้อยในฤดูอื่นๆ เป็นต้น แต่ถ้าพฤติกรรมที่เกิดขึ้นในอดีต มีความแปรผันผิดปกติ เช่น สินค้าเครื่องกันหนาว แทนที่จะขายได้น้อยในฤดูอื่นนอกจากฤดูหนาวเป็นประจำทุกปี กลับกลายเป็นว่า บางปีขายได้มาก บางปีก็ขายได้น้อย เอาแน่นอนอะไรไม่ได้ พฤติการณ์ทำนองนี้การพยากรณ์ทางสถิติไม่สามารถทำได้

ทั้งนี้เพราะการพยากรณ์ทางสถิติ ตั้งอยู่บนสมมติฐานที่ว่า "สิ่งอื่นไม่เปลี่ยนแปลง" นั่นคือว่า เหตุการณ์อื่นๆ ที่จะเกิดขึ้นในช่วงเวลาของการพยากรณ์ เหมือนกันกับที่เกิดขึ้นในช่วงเวลาที่แล้วๆ มา แต่ความเป็นจริงมิได้เป็นเช่นนั้น เพราะเหตุการณ์ที่จะเกิดขึ้นในช่วงเวลาของการพยากรณ์ น่าจะคล้ายคลึงกับเหตุการณ์ที่เพิ่งเกิดขึ้นเมื่อเร็วๆ นี้มากกว่าเหตุการณ์ที่เกิดขึ้นเมื่อนานมาแล้ว ยิ่งกว่านั้นในขณะที่จะทำการพยากรณ์หากบังเอิญไปอยู่ในวัฏจักร (Cycle) ที่เศรษฐกิจกำลังรุ่งเรืองถึงขีดสุด (Peak) ค่าที่พยากรณ์ได้จะสูงกว่าความเป็นจริง เพราะปกติวิสัยของเศรษฐกิจในช่วงเวลาถัดไป จะเริ่มตกต่ำ (Contraction) และในทางตรงข้าม ถ้าอยู่ในวัฏจักรที่เศรษฐกิจกำลังตกต่ำถึงที่สุด (Trough) ค่าที่พยากรณ์ได้ ก็จะต่ำกว่าความเป็นจริง เพราะปกติวิสัยของเศรษฐกิจในช่วงเวลาถัดไป จะเริ่มฟื้นตัวดีขึ้น (Expansion)

2.2 โครงสร้างข้อมูล

2.2.1 JSON

JSON (JavaScript Object Notation) คือ รูปแบบของข้อมูลชนิดหนึ่งที่ใช้สำหรับแลกเปลี่ยนข้อมูล ใช้ syntax ของภาษาจาวาสคริปต์ แต่ไม่ถูกมองว่าเป็นภาษาโปรแกรม แต่จัดว่าเป็นภาษาที่ใช้ในการแลกเปลี่ยนข้อมูลเท่านั้น สามารถทำความเข้าใจได้ง่าย JSON เป็น Text ธรรมดาที่อยู่ในรูปแบบของ Object เรียกว่า JSON Object เริ่มจากเครื่องหมายปีกกาเปิด “{” และสิ้นสุดที่เครื่องหมายปีกกาปิด “}” ภายใน Object จะมีข้อมูลที่เรียกว่า member ซึ่งจะอยู่ในรูปแบบของการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จับคู่ระหว่าง name กับ value สามารถเขียนได้ดังนี้ {"name": "value"} หากมี member มากกว่า 1 ชุดให้คั่นด้วยเครื่องหมาย comma (",") ดังนี้ {"name": "value", "name": "value", ...} name ใช้สำหรับการอ้างอิง value ส่วน value คือข้อมูลที่เป็นไปได้ทั้ง String, Number, Object, Array, Boolean หรือ Null

ข้อดีของ JSON ได้แก่ ถูกออกแบบมาเพื่อให้ง่ายสำหรับการแลกเปลี่ยนข้อมูล, ลักษณะข้อมูลสามารถอ่านและแก้ไขได้ง่าย, มีภาษาหลายๆ ภาษาที่รองรับการใช้งาน JSON เป็นต้น

```
{
  "Name": "Eknarong",
  "Lastname": "Aphiphutthikul",
  "Age": 22,
  "Address": {
    "Street": "1600 Pennsylvania Ave",
    "City": "Washington",
    "State": "DC",
    "Zip": 20500
  },
  "Sex": "Male"
}
```

รูปที่ 2.1 ตัวอย่าง JSON

2.2.2 XML

XML ย่อมาจาก Extensible Markup Language เป็นภาษาที่ใช้สำหรับการเขียนเอกสาร markup (markup document) โดยที่เอกสาร markup นั้นมีการใช้ metadata หรือ tags เพื่อบอกหน้าที่และประเภทของข้อมูลในส่วนต่างๆของเอกสาร การเพิ่ม metadata เข้าไปในเอกสารสามารถทำให้โครงสร้างของเอกสารชัดเจนขึ้น ทำให้การประมวลผลเอกสารเป็นไปได้ง่าย

XML เป็นภาษาที่เหมาะสมกับการแลกเปลี่ยนข้อมูลผ่านทางเครือข่ายคอมพิวเตอร์ เนื่องจาก XML ไม่ได้ขึ้นอยู่กับโปรแกรมหรือระบบปฏิบัติการใด นอกจากนี้ยังมีความยืดหยุ่น เนื่องจากสามารถที่จะกำหนดและตั้งค่า metadata ให้เหมาะสมตามที่ต้องการได้อย่างอิสระ และยังสามารถเพิ่มเติม metadata ในภายหลังได้โดยไม่มีผลกระทบต่อโปรแกรมที่มีอยู่

XML ถือว่าเป็นส่วนหนึ่งของ Standard Generalized Markup (SGML) ที่เป็นข้อกำหนดในการสร้างหรือจัดทำเอกสารในรูปแบบอิเล็กทรอนิกส์ที่กำหนดโดย W3C หรือ World Wide Web Consortium ที่มีโครงสร้างและรูปแบบที่เปิดให้แอปพลิเคชันต่างๆ สามารถเรียกไปใช้งานได้ จึงทำให้การจัดการข้อมูลหรือเรียกใช้ข้อมูลจากแอปพลิเคชันต่างๆ นั้นอยู่ในมาตรฐานเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>

```

รูปที่ 2.2 ตัวอย่าง XML

2.3 ภาษาไพธอน (Python)

Python เป็นภาษา Dynamic Object-Oriented Programming ที่ใช้กันอย่างกว้างขวาง ถูกพัฒนาขึ้นโดย Guido van Rossum ที่ Centrum Wiskunde & Informatica (CWI) ในประเทศ เนเธอร์แลนด์และในปัจจุบันดูแลโดยมูลนิธิซอฟต์แวร์ไพธอน (Python Software Foundation : PSF) โดยเวอร์ชันล่าสุดคือ Python 3.5 และ Python 2.7 ซึ่ง Python ถูกพัฒนาขึ้นมาจากภาษา C มีตัวแปลภาษาคือไพธอนอินเทอร์พรีเตอร์ (Python Interpreter) การประมวลผลจะทำในรูปแบบ Interpreter คือ จะประมวลผลไปที่ละบรรทัดและปฏิบัติตามคำสั่งที่ได้รับ

2.3.1 คุณลักษณะเด่นของภาษาไพธอน

- พัฒนาให้ไม่ยึดติดกับแพลตฟอร์ม กล่าวคือ สามารถทำงานได้หลายระบบ เช่น ระบบ Unix, Linux, Windows เป็นต้น
- ไวยากรณ์ของภาษา Python ถูกออกแบบมาเพื่อให้เข้าใจได้ง่าย ไม่ซับซ้อน โดยจะใช้คำในภาษาอังกฤษมาเป็นคำสั่ง
- เป็นซอฟต์แวร์ที่เปิดเผยแพร่ให้ต้นฉบับ (Open Source) ซึ่งทำให้สามารถพัฒนาโปรแกรมได้โดยไม่เสียค่าใช้จ่ายใดๆ
- สนับสนุนแนวคิดแบบเชิงวัตถุ หรือ Object Oriented Programming (OOP)
- รองรับการเขียนส่วนติดต่อกับผู้ใช้ (User Interface)
- เป็น Dynamic typing คือ สามารถเปลี่ยนชนิดข้อมูลได้ง่ายและสะดวก
- มี Built-in Object Types คือ โครงสร้างของข้อมูลที่สามารถใช้ได้ ใน Python ประกอบไปด้วย ลิสต์ (List), ดิกชันนารี (Dictionary), สตริง (String) ที่ง่ายต่อการใช้งานและมีประสิทธิภาพสูง
- มีโมดูล (Module) สำหรับจัดการ Regular Expression

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีโมดูลที่สร้างขึ้นจากนักพัฒนาที่สนับสนุนมากมาย
- สามารถจัดการหน่วยความจำอย่างอัตโนมัติและมีประสิทธิภาพ
- อนุญาตให้ฝังชุดคำสั่งของ Python เอาไว้ภายในโคดของภาษา C และ C ++
- มีฟังก์ชันสนับสนุนฐานข้อมูล เช่น ODBC, MySQL และอื่นๆ
- มีไลบรารีสนับสนุนด้านปัญญาประดิษฐ์

2.3.2 ชนิดของตัวแปร

ชนิดของตัวแปรในภาษาไพธอน แบ่งออกเป็น 5 ชนิดใหญ่ ๆ ได้แก่ number, string, list, tuple, dictionary ซึ่งเป็นตัวแปรทั่ว ๆ ไป แต่ภาษาไพธอนยอมให้มีตัวแปร list, tuple, dictionary ที่ผสมกันได้ เรียกว่าชนิด complex ถ้าหากต้องการทราบว่าตัวแปรที่ประกาศใช้นั้นเป็นชนิดใด เราสามารถสอบถามชนิดตัวแปรได้จากคำสั่ง `type(var)` ดังคำสั่งในรูปแบบต่อไปนี้

```
>>> pi = 3.14159
>>> type(pi)
<type 'float'>
>>> n = 100
>>> type(n)
<type 'int'>
>>> id = '504244001'
>>> type(id)
<type 'str'>
>>> 12j+1
(1+12j)
>>> type(12j+1)
<type 'complex'>
```

รูปที่ 2.3 ตัวอย่างคำสั่งสอบถามชนิดตัวแปร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ตัวแปรชนิดตัวเลข

ตัวแปรชนิดตัวเลข (number) ใช้สำหรับเก็บข้อมูลที่เป็นจำนวน เช่น เลขจำนวนนับ ซึ่งเลขจำนวนนับนี้มีคุณสมบัติสามารถเพิ่มค่าได้ คำนวณได้ และเปลี่ยนแปลงค่าได้ ประกอบด้วยเลขจำนวนเต็มได้แก่ ชนิด integer ซึ่งรับข้อมูลได้ตั้งแต่ -2^{31} ถึง $2^{31}-1$ และชนิด long integer ซึ่งรับข้อมูลตัวเลขที่มากกว่า integer ที่รับได้โดยไม่จำกัด ขึ้นอยู่กับเครื่องคอมพิวเตอร์แต่ละเครื่องว่ามีหน่วยความจำเสมือน (virtual memory) เท่าไร ในการประกาศตัวแปรชนิด long integer นั้นเพียงใส่ตัวอักษร L ต่อท้ายตัวเลขข้อมูลที่ต้องการเก็บเท่านั้น ถือว่าเป็นการประกาศตัวแปรชนิด long integer แล้ว สำหรับวิธีการใช้ตัวแปรสามารถเก็บข้อมูลในรูปแบบทั้งเลขฐานสิบ เลขฐานแปดโดยใส่เลข 0 ด้านหน้าตัวเลขฐานแปด และเลขฐานสิบหกโดยใส่เลข 0x ด้านหน้าเลขฐานสิบหก ได้ทั้งสองชนิด ดังนี้

```
>>> longvar = 417324698473L
>>> print longvar
417324698473
>>> type(longvar)
<type 'long'>
>>> decimalvar = 17
>>> print decimalvar
17
>>> octalvar = 017
>>> print octalvar
15
>>> hexvar = 0xff
>>> print hexvar
255
```

รูปที่ 2.4 ตัวอย่างคำสั่งตัวแปรชนิดตัวเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ตัวแปรชนิด number ยังประกอบด้วยตัวแปรชนิดเลขจำนวนจริง (floating point) ตัวแปรชนิดเลขจำนวนจริงสามารถรับค่าได้ตั้งแต่ $+10^{308.25}$ ถึง $-10^{308.25}$ ในการแสดงผลตัวเลขจำนวนจริงจะมีตัวอักษร e แสดงขึ้นมาในส่วนตรงกลางระหว่างตัวเลข มีความหมายว่า ตัวเลขในส่วนด้านหน้าอักษร e คือตัวเลขฐานสิบธรรมดา ส่วนตัวข้างหลังหมายถึงเลขยกกำลัง เช่น $32.3+e28$ มีค่าเท่ากับ 32.3^{28} เป็นต้น

2) ตัวแปรชนิดสายอักษร

ตัวแปรชนิดสายอักษร (string) ใช้สำหรับเก็บข้อมูลที่เป็นข้อความ หรือประโยค เพื่อนำมาเก็บเป็นข้อมูลต่างๆ ไป เช่น ชื่อ-สกุล ที่อยู่ และอื่น ๆ ซึ่งตัวแปรชนิดนี้ไม่สามารถนำมาคำนวณได้ ในการประกาศตัวแปรชนิดนี้ ข้อความที่ต้องการเก็บในตัวแปรจะต้องอยู่ภายใต้เครื่องหมาย (" ") หรือเครื่องหมาย (') กำกับอยู่ เช่น `name = 'Taweerat'` หรือ `name = "Taweerat"` ดังนั้นในกรณีที่มีการเก็บตัวเลขดังรูปแบบ '1234' จึงมีความหมายเป็นเพียงสายอักษร ไม่สามารถนำมาคำนวณได้ แต่ถ้าหากต้องการให้คำนวณได้จำเป็นต้องใช้ฟังก์ชันเพื่อเปลี่ยนชนิดตัวแปรเป็นชนิดตัวเลขใหม่ วิธีการเขียนคำสั่งเพื่อเก็บข้อมูลชนิดตัวแปรสายอักษร มี ดังนี้

```
>>> str1 = 'This is a literal string'
>>> str2 = "This is another string"
>>> str3 = "I'm a teacher"
>>> str4 = 'I don\'t like VB'
>>> str5 = "This is an example of \
two lines string"
>>> print str1
This is a literal string
>>> print str2
This is another string
>>> print str3
I'm a teacher
>>> print str4
I don't like VB
>>> print str5
This is an example of two lines string
```

รูปที่ 2.5 ตัวอย่างคำสั่งตัวแปรชนิดสายอักษร

3) ตัวแปรชนิดลิสต์

ตัวแปรชนิดลิสต์ (list) เป็นตัวแปรที่มีขึ้นในโปรแกรมภาษาใหม่ ๆ ซึ่งโปรแกรมภาษารุ่นเก่า ๆ จะเรียกว่า อะเรย์ (array) ซึ่งประกอบด้วยตัวแปร 1 ตัว มีข้อมูลเก็บได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลาย ๆ จำนวนในลักษณะที่ต่อเนื่องกัน การเรียกใช้ข้อมูลภายในลิสต์จะต้องระบุถึงดัชนีลำดับของข้อมูลที่เก็บเอาไว้ โดยเริ่มต้นจาก 0 เช่นเดียวกับบอเรีย แต่ลิสต์สามารถเรียกดัชนีที่เป็นค่าลบได้ นั่นคือ ถ้าเป็น -1 หมายถึง ข้อมูลลำดับสุดท้าย ดังตัวอย่างต่อไปนี้

```
>>> fruits = ["banana", "papaya", "orange", "apple", "mango"]
>>> fruits
['banana', 'papaya', 'orange', 'apple', 'mango']
>>> fruits[0]
'banana'
>>> fruits[1]
'papaya'
>>> fruits[-1]
'mango'
>>> fruits[-3]
'orange'
```

รูปที่ 2.6 ตัวอย่างคำสั่งตัวแปรชนิดลิสต์

4) ตัวแปรชนิดทUPLE

ตัวแปรชนิดทUPLE (tuple) เป็นตัวแปรที่มีลักษณะคล้ายกับตัวแปรชนิดลิสต์ เพียงแต่ตัวแปรชนิดทUPLEไม่สามารถนำมาเพิ่มเติมหรือเปลี่ยนแปลงข้อมูลได้เลยเมื่อได้สร้างขึ้นมาแล้ว ดังนั้น การเรียกใช้ข้อมูลจึงมีการใช้ตัวเลขดัชนีเช่นเดียวกัน และที่แตกต่างกันอีกอย่างหนึ่งคือ ในขณะที่สร้างตัวแปร ตัวแปรชนิดทUPLEจะมีข้อมูลอยู่ภายใต้เครื่องหมาย '()' ในขณะที่ตัวแปรชนิดลิสต์จะมีข้อมูลอยู่ภายใต้เครื่องหมาย '[]' ดังตัวอย่างต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

>>> animals = ("monkey", "rabbit", "cat", "kangaroo", "chicken")
>>> animals[0]
'monkey'
>>> animals[1]
'rabbit'
>>> animals[-1]
'chicken'
>>> animals[-3]
'cat'

```

รูปที่ 2.7 ตัวอย่างคำสั่งตัวแปรชนิดทUPLE

5) ตัวแปรชนิดดิกชันนารี

ตัวแปรชนิดดิกชันนารี (dictionary) เป็นตัวแปรที่มีลักษณะเก็บข้อมูลได้หลาย ๆ ค่า แต่ข้อมูลที่เก็บจะต้องเป็นการจับคู่ระหว่างคีย์และข้อมูล สามารถเปลี่ยนแปลงข้อมูลที่เก็บอยู่ได้ ตัวแปรชนิดนี้สามารถเก็บข้อมูลที่มีชนิดแตกต่างกันในตัวแปรเดียวกันได้ ดังตัวอย่างต่อไปนี้

```

>>> name={'Dad': 'Somchai', 'Mom': 'Somsri', 'Bro': 'John'}
>>> name
{'Dad': 'Somchai', 'Bro': 'John', 'Mom': 'Somsri'}
>>> name['Dad']
'Somchai'
>>> name['Mom']
'Somsri'

>>> age={'Dad': 42, 'Mom': 40}
>>> age['Dad']
42

```

รูปที่ 2.8 ตัวอย่างคำสั่งตัวแปรชนิดดิกชันนารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ

2.4.1 Elasticsearch

Elasticsearch คือ ฐานข้อมูลแบบ NoSQL ถูกสร้างขึ้นเมื่อปี ค.ศ. 2010 โดยเวอร์ชัน 0.4 เป็นเวอร์ชันแรกทีปล่อยออกมาให้ใช้งาน พัฒนาด้วยภาษาจาวาโดย Shay Banon Elasticsearch เป็น Open Source ซึ่งสามารถโหลดมาใช้งานได้ โดยไม่เสียค่าใช้จ่ายใดๆ จุดเด่นของ Elasticsearch ก็คือ การค้นหาและวิเคราะห์ข้อมูลได้อย่างรวดเร็วและมีประสิทธิภาพ, รองรับข้อมูลขนาดใหญ่, สามารถกระจายการเก็บข้อมูลในหลายๆเครื่องและสามารถขยายจำนวนเครื่องได้ง่าย, มีการสำรองข้อมูลให้อัตโนมัติโดยการกระจายข้อมูลไปเก็บบนเครื่องอื่นๆ ทำให้มั่นใจได้ว่าหากมีเครื่องใดเครื่องหนึ่งไม่สามารถใช้งานได้ ก็ยังสามารถค้นหาข้อมูลและวิเคราะห์ข้อมูลได้อย่างถูกต้องสมบูรณ์, รองรับการทำงานในรูปแบบของ RESTful API ซึ่งช่วยให้สามารถติดต่อกับฐานข้อมูลได้สะดวกยิ่งขึ้น, เก็บข้อมูลในรูปแบบของ JSON ซึ่งช่วยให้สามารถเข้าถึงข้อมูลได้ง่ายและเพิ่มความยืดหยุ่นในการเก็บมากยิ่งขึ้น เพราะเป็นการเก็บแบบไม่มีข้อกำหนดทางด้านโครงสร้าง หรือก็คือไม่จำเป็นต้องเก็บข้อมูลให้ตรงตามที่กำหนดเอาไว้ ไม่เหมือนกับฐานข้อมูลที่เก็บแบบมีโครงสร้าง (RDBMS) เช่น MySQL ซึ่งต้องเก็บข้อมูลตามที่โครงสร้างกำหนดไว้เท่านั้น ในการเปรียบเทียบโครงสร้างระหว่าง RDBMS กับ Elasticsearch ได้แสดงในตารางที่ 2.1

ตารางที่ 2.1 เปรียบเทียบโครงสร้างระหว่าง RDBMS กับ Elasticsearch

RDBMS	Elasticsearch
Database	Index
Table	Type
Row	Document
Column	Field



elasticsearch

รูปที่ 2.9 Logo ของ Elasticsearch

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 MongoDB

MongoDB คือ ฐานข้อมูลแบบ NoSQL จัดเก็บข้อมูลในรูปแบบของ JSON รองรับการทำให้ Full Index ทำให้การค้นหาข้อมูลขนาดใหญ่ทำได้รวดเร็ว สามารถจัดเก็บข้อมูลแบบกระจายเครื่องได้ ส่งผลให้มีความเร็วที่เพิ่มขึ้นเนื่องจากแต่ละเครื่องสามารถทำงานพร้อมๆกันได้ สามารถขยายเครื่องที่ใช้เก็บข้อมูลได้ง่าย มีระบบสำรองข้อมูลเพื่อป้องกันไม่ให้ข้อมูลสูญหาย สามารถเขียนชุดคำสั่งจัดการฐานข้อมูลที่มีการทำขั้นตอนซ้ำๆเอาไว้เรียกใช้งานในภายหลังได้ เก็บข้อมูลด้วยระบบ GridFS เป็นระบบการเก็บไฟล์บนพื้นที่ Harddisk ที่เก็บข้อมูลเป็นก้อนๆ และรองรับการเพิ่มหรือลดของปริมาณข้อมูลได้ สามารถวิเคราะห์ข้อมูลและสรุปผลของข้อมูลได้ ในการเปรียบเทียบโครงสร้างระหว่าง RDBMS กับ MongoDB ได้แสดงในตารางที่ 2.2

ตารางที่ 2.2 เปรียบเทียบโครงสร้างระหว่าง RDBMS กับ MongoDB

RDBMS	MongoDB
Table	Collection
Row	Document
Column	Field
Joins	Embedded document, linking



mongoDB

รูปที่ 2.10 Logo ของ MongoDB

1) MongoDB Connector for BI

MongoDB Connector for Business Intelligence (BI) ช่วยให้เราสามารถเชื่อมต่อ ฐานข้อมูล MongoDB กับ เครื่องมือทาง Business Intelligence ได้ เช่น Tableau, MicroStrategy เป็นต้น โดยเครื่องมือเหล่านี้คาดหวังว่าจะเชื่อมต่อกับฐานข้อมูลประเภท SQL จึงเป็นความท้าทายเป็นอย่างมากที่จะเชื่อมต่อกับฐานข้อมูล MongoDB ที่เป็นฐานข้อมูลประเภท NoSQL โดย MongoDB Connector for BI ทำหน้าที่เป็นตัวกลางในการส่งผ่านคำสั่งค้นหาและข้อมูลระหว่าง mongod กับ เครื่องมือทาง Business Intelligence โดย MongoDB Connector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

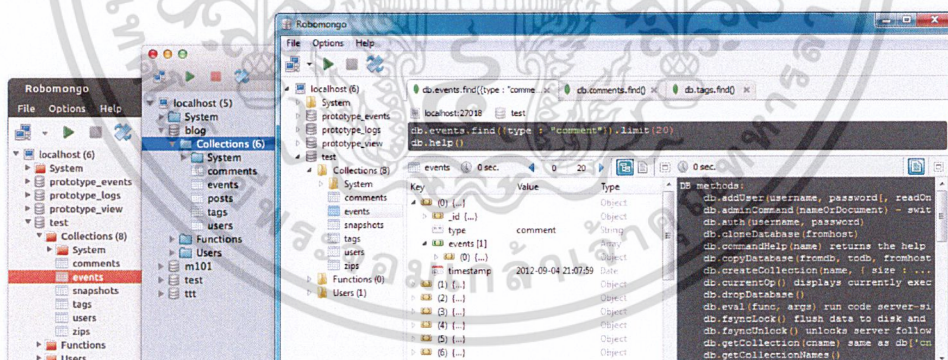
for BI จะมี mongosqlد รับคำสั่งค้นหาข้อมูลที่เป็น ภาษา SQL จาก Business Intelligence จากนั้นจะแปลงคำสั่งดังกล่าวเป็นคำสั่งค้นหาข้อมูลของ MongoDB โดยใช้ไฟล์ schema.druid ช่วยในการแปลง เมื่อ MongoDB ส่งข้อมูลกลับมาจะทำการส่งข้อมูลต่อไปยังเครื่องมือ Business Intelligence ดังรูปที่ 2.11



รูปที่ 2.11 การทำงานของ MongoDB Connector for BI

2) Robo 3T

Robo 3T เป็นส่วนติดต่อ MongoDB กับผู้ใช้ในรูปแบบของ User Interface เพื่ออำนวยความสะดวกในการจัดการกับฐานข้อมูล MongoDB โดยสามารถใช้คำสั่งได้เหมือนกับใช้ Mongo Shell ทุกประการ อีกทั้งยังมี Autocompletion เพื่อช่วยแสดงคำสั่งต่างๆที่เกี่ยวข้องกับสิ่งที่ผู้ใช้ต้องการอีกด้วย



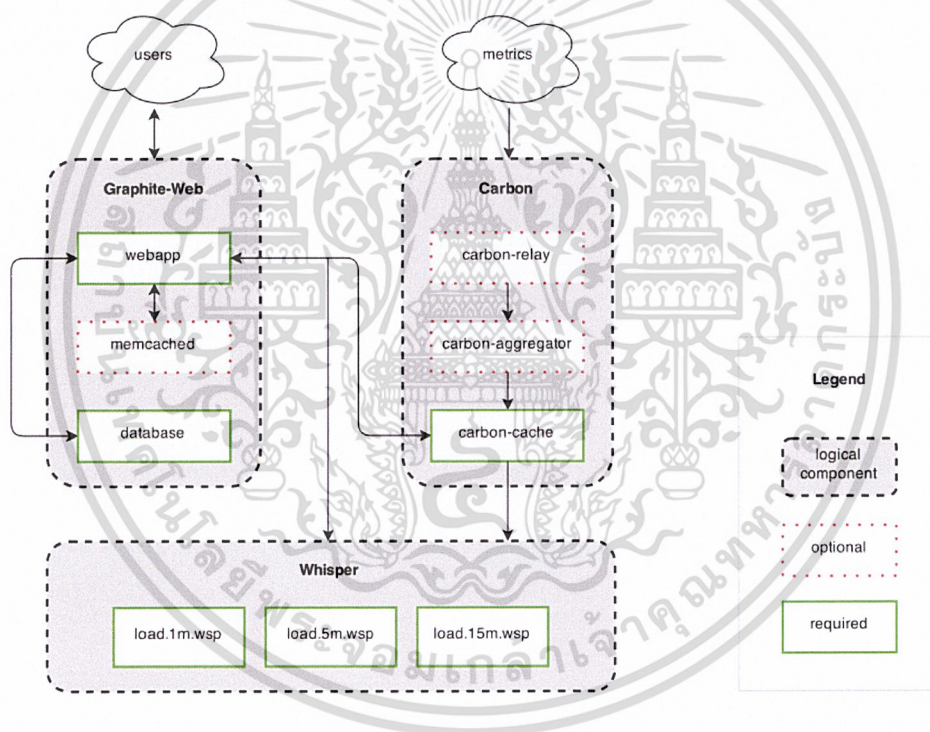
รูปที่ 2.12 โปรแกรม Robo 3T

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 Graphite

Graphite ถูกออกแบบและเขียนขึ้นโดย Chris Davis ที่ Orbitz ในปี 2006 และปล่อยออกมาภายใต้ใบอนุญาต Apache 2.0 แบบ Open Source เป็นเครื่องมือที่ใช้ในการ monitoring ซึ่งทำงานได้บน hardware หรือ Cloud สามารถใช้ในการติดตามประสิทธิภาพของ websites, applications, business services และ network servers เป็นเครื่องมือที่ใช้ในการจัดเก็บ ค้นหา แชร หรือแสดงข้อมูลตัวเลขออกมาเป็นกราฟ Graphite นั้นไม่สามารถเก็บรวบรวมข้อมูลเองได้ จะต้องมีการเขียนโค้ดเพื่อส่งข้อมูลเข้ามาเก็บ

Graphite ประกอบด้วย 3 ซอฟต์แวร์หลักๆคือ Carbon เป็นเซิร์ฟเวอร์ที่ใช้ในการรับข้อมูล, Whisper เป็น library database อย่างง่าย ที่ใช้สำหรับเก็บข้อมูลแบบ time-series, Graphite-web เป็นส่วนติดต่อกับผู้ใช้งาน และ API สำหรับการสร้างกราฟและแดชบอร์ด



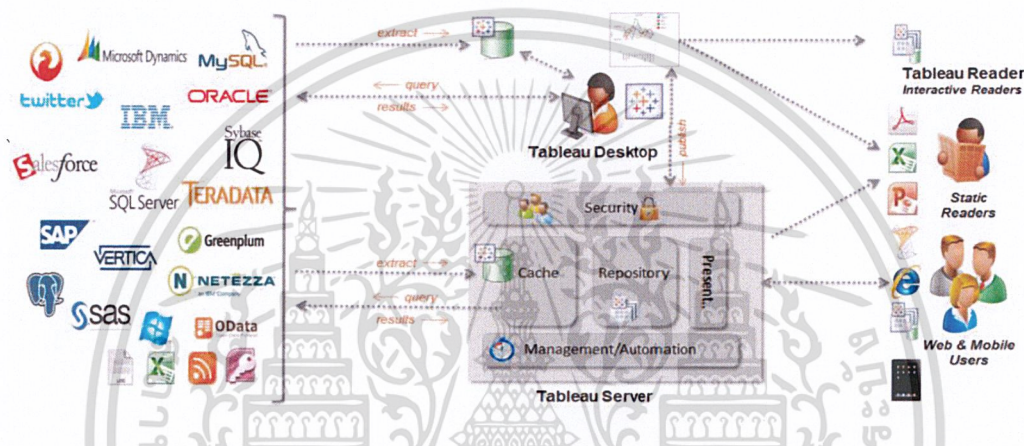
รูปที่ 2.13 สถาปัตยกรรมของ Graphite

2.4.4 Tableau

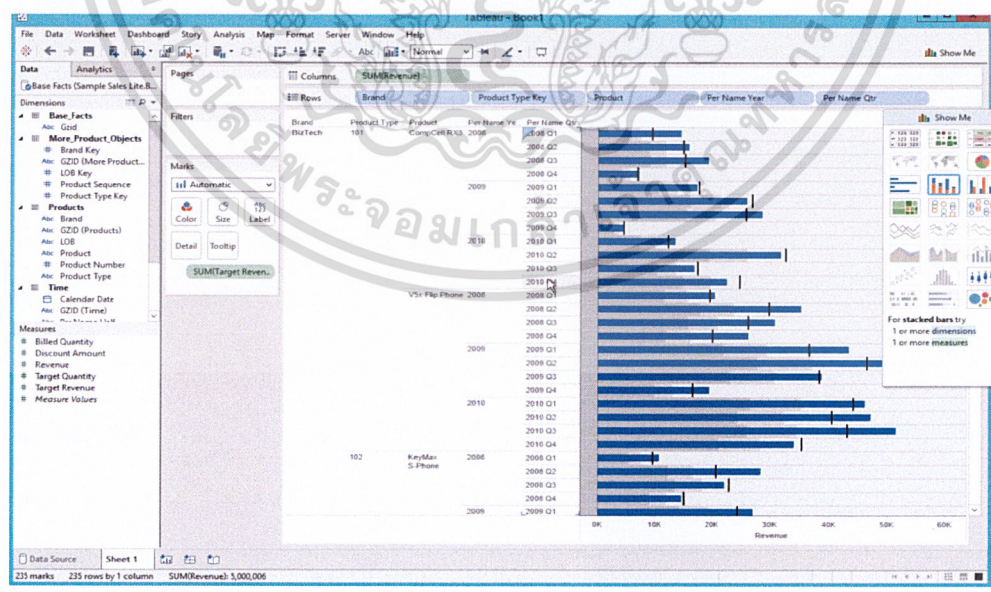
Tableau เป็นโปรแกรมสำหรับการวิเคราะห์ข้อมูล และหาคำตอบในเชิงธุรกิจ โดยการใช้ Tableau จะทำให้ผู้ใช้งานมีความเข้าใจในข้อมูลมากยิ่งขึ้น รวมถึงผู้ใช้งานสามารถนำข้อมูลที่มีอยู่มาวิเคราะห์และแสดงผลได้อย่างมีประสิทธิภาพ Tableau จะนำข้อมูลมาสร้างเป็นภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นามธรรม (Visualization) ต่างๆตามที่เราต้องการและสามารถสร้างเป็น Dashboard ได้ พร้อมทั้ง Share Content ต่างๆได้อย่างรวดเร็วด้วย Tableau Server Tableau ถูกพัฒนามาจาก VizQL เป็นภาษาที่ใช้สำหรับวิเคราะห์ข้อมูลจากฐานข้อมูลและแสดงผลในรูปแบบกราฟิก (Graphic) ได้ในเวลาเดียวกัน Tableau ได้พัฒนาระบบ Data Engine ขึ้นมาเพื่อทำให้การวิเคราะห์ข้อมูลของ CPU, L1 Cache และ L2 Cache ทำได้เต็มประสิทธิภาพ โดยสามารถเข้าถึงข้อมูลจาก SQL Database ต่างๆได้โดยตรง ซึ่งสำหรับการเชื่อมต่อกับฐานข้อมูลประเภท NoSQL จะต้องอาศัยตัวกลางที่ใช้ในการแปลงคำสั่งเพื่อเข้าถึงข้อมูล นอกจากการเชื่อมต่อกับ Database แล้ว Tableau ยังสามารถอ่านข้อมูลจากไฟล์ประเภทเชิงสถิติ (Statistics) ได้อีกด้วย



รูปที่ 2.14 สถาปัตยกรรมของ Tableau



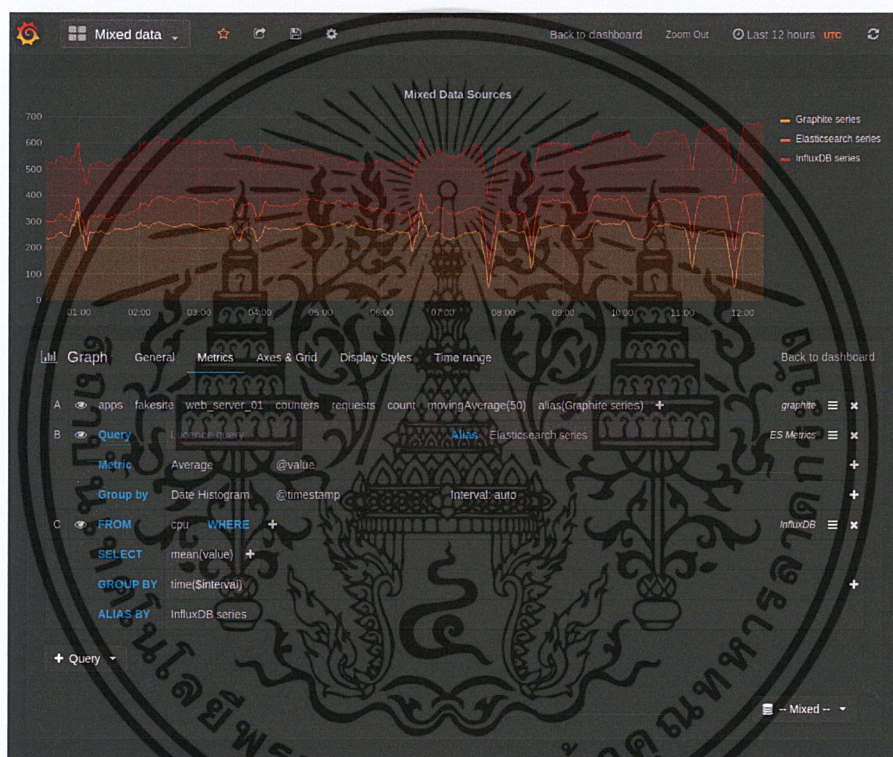
รูปที่ 2.15 Tableau Desktop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.5 Grafana

Grafana คือเครื่องมือที่เป็น Open Source ใช้ในการวิเคราะห์ข้อมูลและแสดงผล เพื่อให้เข้าใจข้อมูลและช่วยในการตัดสินใจทางธุรกิจ โดยคุณลักษณะของ Grafana สามารถสรุปได้ดังนี้

1) Visualize โดย Grafana สามารถแสดงข้อมูลด้วยรูปแบบต่างๆได้มากมาย เช่น กราฟเส้น (Line chart), แผนภูมิแท่ง (Bar Graph), Singlestat Panel, ตาราง (Table), Heatmap Panel เป็นต้น ซึ่งภายในรูปแบบหนึ่งสามารถเชื่อมต่อกับแหล่งข้อมูลมากกว่าหนึ่งแหล่ง เพื่อแสดงผลร่วมกันได้



รูปที่ 2.16 ตัวอย่างการแสดงกราฟจากหลายแหล่งข้อมูลใน Grafana

2) Alerting โดย Grafana มีระบบการแจ้งเตือน โดยผู้ใช้งานสามารถกำหนดได้ว่า จะให้มีการแจ้งเตือนด้วยเงื่อนไขใด ยกตัวอย่าง ในทุกวันจะมีการเก็บจำนวนการขอข้อมูลจากลูกค้ามายังเว็บเซิร์ฟเวอร์ ซึ่งหากวันใดมีจำนวนการขอข้อมูลมากเกินกว่าจุดที่เรากำหนดไว้เป็นเงื่อนไข จะทำให้มีการแจ้งเตือนเกิดขึ้น ซึ่งการแจ้งเตือนนั้นสามารถแจ้งเตือนผ่านหน้า Dashboard, slack, pagerduty, VictorOps, OpsGenie, Email หรือ webhook

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ตัวอย่างการแจ้งเตือนบน Dashboard ใน Grafana

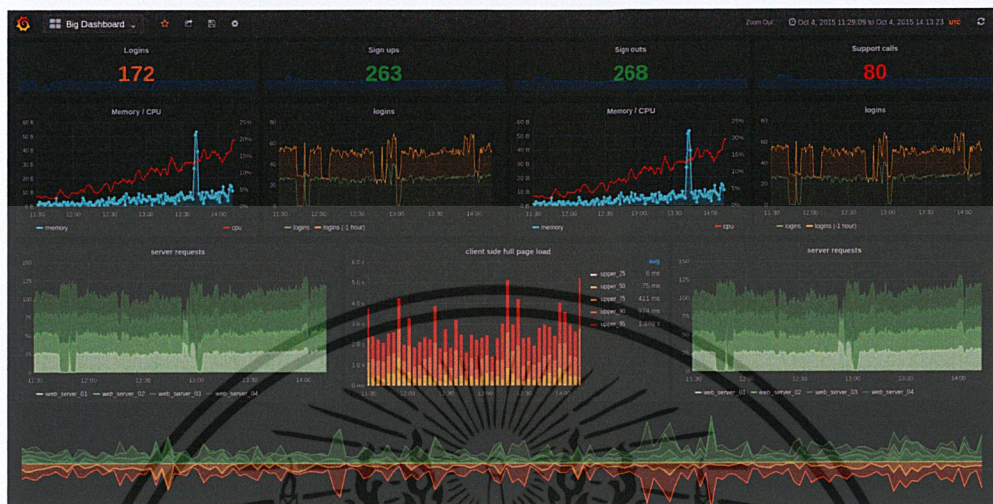
3) Data source โดย Grafana รองรับการเชื่อมต่อกับฐานข้อมูลมากกว่า 30 ฐานข้อมูล เช่น Graphite, influxdb, Prometheus, Elasticsearch, AWS CloudWatch, OpenTSDB, Ambari Metrics, AppDynamics, BlueFlood, ClickHouse, Cloudera Manager, Crate, DalmatinerDB, DataDog, Druid, Finance, Gnocchi, GoogleCalendar, Hawkular, Heroic, IBM APM, KairosDB เป็นต้น



รูปที่ 2.18 ตัวอย่างฐานข้อมูลที่ Grafana รองรับการเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) Dashboard Features โดย Grafana รองรับการสร้าง Dashboard ซึ่งภายในหนึ่งหน้าสามารถแสดงผลแบบ Visualize ได้หลากหลายแบบ สามารถแชร์ Dashboard เพื่อให้บุคคลอื่นสามารถเห็นข้อมูลต่างๆได้



รูปที่ 2.19 ตัวอย่างการสร้าง Dashboard ใน Grafana

2.4.6 Kibana

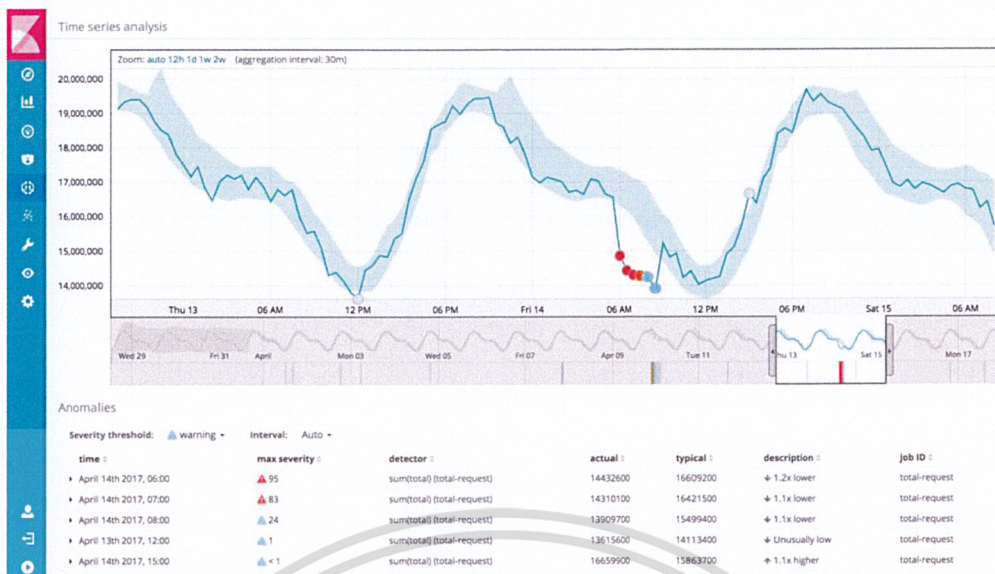
Kibana เป็น Open Source ที่ใช้ในการวิเคราะห์ข้อมูลและแสดงผลจากฐานข้อมูล Elasticsearch โดยสามารถเข้าใช้งานได้ผ่านทางเว็บเบราว์เซอร์ localhost:5601 Kibana มีคุณลักษณะต่างๆมากมายโดยสามารถสรุปได้ดังนี้

1) Visualize โดย Kibana สามารถแสดงผลข้อมูลได้หลากหลายรูปแบบ เช่น การแสดงข้อมูลบนแผนที่โดยการใส่ข้อมูลทางภูมิศาสตร์ เพื่อให้สามารถเห็นข้อมูลเชิงพื้นที่ หรือจะแสดงกราฟโดยใช้ข้อมูลที่เป็น time series เพื่อให้เห็นแนวโน้มของข้อมูล เป็นต้น

2) Metric Aggregations เป็นส่วนของคำสั่งที่ช่วยในการประมวลผลข้อมูล โดยมีคำสั่งต่างๆมากมายสามารถสรุปโดยสังเขปได้ดังต่อไปนี้ Count, Average, Sum, Min, Max, Unique Count, Standard Deviation, Percentiles, Percentile Rank เป็นต้น ซึ่งอำนวยความสะดวกในการวิเคราะห์ข้อมูลเป็นอย่างมาก

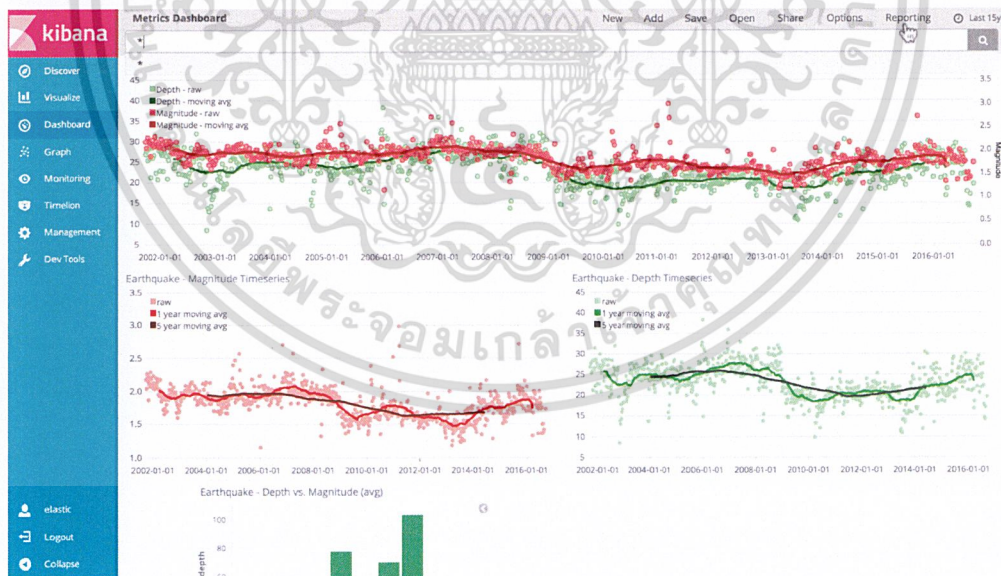
3) Machine Learning เนื่องจากข้อมูลที่มีขนาดใหญ่และมีความซับซ้อนมากยิ่งขึ้น ทำให้มนุษย์ไม่สามารถที่จะค้นหาสิ่งผิดปกติได้ ดังนั้น Kibana จึงช่วยวิเคราะห์ข้อมูลและค้นหาสิ่งผิดปกติโดยผ่านทาง Machine Learning

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ตัวอย่างการใช้ Machine Learning ใน Kibana

4) Dashboard โดย Kibana รองรับการสร้าง Dashboard และสามารถแชร์ Dashboard โดยส่ง URL ของ Dashboard นั้นให้กับบุคคลที่ต้องการได้



รูปที่ 2.21 ตัวอย่างการสร้าง Dashboard ใน Kibana

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7 Sensu

Sensu เป็นระบบตรวจสอบและติดตามโครงสร้างพื้นฐาน (Monitoring System) ประเภท agent based ซึ่งมีบริการรูปแบบการตรวจสอบ (framework) เตรียมไว้ให้ สนับสนุนเครื่องมือในการวัดประสิทธิภาพของธุรกิจ (business KPIs) เพื่อให้ตอบสนองต่อความต้องการของภาคธุรกิจ Sensu ประกอบไปด้วย 4 ส่วน คือ Sensu Server, Sensu Client, RabbitMQ และ Sensu API

1) Sensu Server จะทำหน้าที่รับข้อมูลต่างๆจาก RabbitMQ และจัดเก็บข้อมูลนั้นลงในฐานข้อมูล (Data Store) ต่างๆที่ได้ตั้งค่าไว้ เช่น Graphite, Librato, InfluxDB, Redis เป็นต้น โดยที่ Redis จะเป็นค่าเริ่มต้น

2) Sensu Client จะถูกติดตั้งไว้บนเครื่องที่ต้องการ monitoring ทำหน้าที่ในการรวบรวมข้อมูลต่างๆตามที่ได้ตั้งค่าไว้ เช่น ไฟล์, ข้อมูลตัวชี้วัด (KPI) เป็นต้น แล้วส่งไปยัง RabbitMQ

3) RabbitMQ เป็นเสมือนตัวกลางในการรับส่งข้อมูลระหว่าง Sensu Server กับ Sensu Client

4) Sensu API เป็นบริการที่ Sensu เตรียมไว้เพื่อใช้สำหรับเข้าถึงข้อมูลและเรียกใช้ฟังก์ชันต่างๆ



รูปที่ 2.22 Logo ของ Sensu

2.4.8 PyCharm

PyCharm เป็น Integrated Development Environment (IDE) ที่ใช้ในการเขียนโปรแกรมภาษา Python ถูกพัฒนาขึ้นโดย บริษัท JetBrains ช่วยให้สามารถเขียนโปรแกรมภาษา Python ได้รวดเร็วและมีความถูกต้องมากยิ่งขึ้น เพราะมีระบบที่ช่วยแสดงคำสั่งต่างๆที่เกี่ยวข้องขณะพิมพ์พร้อมทั้งมีระบบตรวจสอบความถูกต้องติดตั้งอยู่ด้วย ซึ่งจะช่วยตรวจสอบความถูกต้องของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่ผู้ใช้ได้เขียน มี Refactor ที่ช่วยในการปรับเปลี่ยนโปรแกรมที่เขียนได้อย่างถูกต้องและปลอดภัย

นอกจากการสนับสนุนภาษา Python แล้ว PyCharm ยังให้การสนับสนุนการพัฒนาเว็บไซต์อีกด้วย ได้แก่ ภาษา JavaScript, CoffeeScript, TypeScript, HTML/CSS, Angular JS, Node.js และอีกมากมาย

PyCharm สามารถทำงานได้บนระบบปฏิบัติการ Windows, Mac OS และ Linux โดยหน้าต่างของโปรแกรม PyCharm นักพัฒนาสามารถจัดสรรพื้นที่และปรับเปลี่ยนได้ตามความต้องการ เช่น สี แถบเครื่องมือ เป็นต้น



รูปที่ 2.23 Logo ของ PyCharm

2.4.9 AppStatsParser

คือโปรแกรมที่ถูกพัฒนาขึ้นด้วยภาษา Python ภายในบริษัททอมสันรอยเตอร์ซอฟต์แวร์ (ประเทศไทย) จำกัด โดย Krissada Choksinanun ใช้สำหรับส่งข้อมูลเชิงสถิติไปฐานข้อมูล Graphite ผ่านทาง Sensu Client

2.4.10 mRemoteNG

mRemoteNG เป็นโปรแกรมที่ใช้ในการเชื่อมต่อและควบคุมคอมพิวเตอร์ระยะไกล โดยสามารถ login เข้าไปควบคุมเครื่องคอมพิวเตอร์ที่ต้องการได้ โดยสามารถมองเห็นหน้าจอ เสมือนว่ากำลังนั่งอยู่หน้าคอมพิวเตอร์เครื่องนั้น สามารถติดตั้งและใช้งานได้เฉพาะระบบปฏิบัติการ Windows เท่านั้นและรองรับการคัดลอกไฟล์ระหว่างเครื่องคอมพิวเตอร์ที่ถูกควบคุมกับคอมพิวเตอร์ที่ใช้ควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 Logo ของ mRemoteNG

2.4.11 pscp

ในปัจจุบันไฟล์ข้อมูลมีความสำคัญต่อองค์กรอย่างมาก จึงมีความจำเป็นที่ต้องนำไฟล์ข้อมูลเหล่านั้นมาวิเคราะห์เพื่อให้เกิดประโยชน์ แต่การที่จะได้ไฟล์ข้อมูลมานั้น อาจจะมีข้อจำกัด เช่น ระบบปฏิบัติการไม่เหมือนกัน หรือ ขั้นตอนในการเข้าถึงไฟล์ข้อมูลที่ซับซ้อน ซึ่งข้อจำกัดเหล่านี้ส่งผลให้ การนำไฟล์ข้อมูลมาวิเคราะห์นั้นเป็นไปได้ยากลำบาก ซึ่งวิธีการที่จะช่วยให้สามารถนำไฟล์ข้อมูลมาวิเคราะห์ได้สะดวกสบายขึ้นนั้นมีหลายวิธี แต่จะเลือกใช้วิธีการใดต้องพิจารณาถึงความเหมาะสมของแต่ละองค์กร

ในการถ่ายโอนไฟล์ข้อมูลระหว่างระบบปฏิบัติการ Windows กับ ระบบปฏิบัติการ Linux นั้น เราสามารถใช้ pscp ในการจัดการได้โดยง่าย โดยทำการดาวน์โหลด โปรแกรม pscp.exe ลงมาบนเครื่องคอมพิวเตอร์ของเรา หากต้องการถ่ายโอนไฟล์จากเครื่องระบบปฏิบัติการ Windows ไปยังเครื่องระบบปฏิบัติการ Linux ให้ทำการเปิด Command Prompt แล้วพิมพ์ path ของโปรแกรม pscp.exe เว้นวรรคตามด้วย path ของไฟล์ที่ต้องการถ่ายโอน เว้นวรรคตามด้วย username@host:path ของเครื่องปลายทาง หากต้องการถ่ายโอนไฟล์จากเครื่องระบบปฏิบัติการ Linux มายังเครื่องระบบปฏิบัติการ Windows ก็สามารถทำได้ในลักษณะเดียวกับการถ่ายโอนไฟล์จากเครื่องระบบปฏิบัติการ Windows ไปยังเครื่องระบบปฏิบัติการ Linux โดยทำการเปิด Command Prompt แล้วพิมพ์ path ของโปรแกรม pscp.exe เว้นวรรคตามด้วย username@host:path ของไฟล์ที่ต้องการถ่ายโอน เว้นวรรคตามด้วย path ที่ต้องการเก็บไฟล์ที่ถ่ายโอน

รูปที่ 2.25 เป็นการแสดงการถ่ายโอนไฟล์ win-data.txt บนระบบปฏิบัติการ Windows ไปสู่ระบบปฏิบัติการ Linux

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

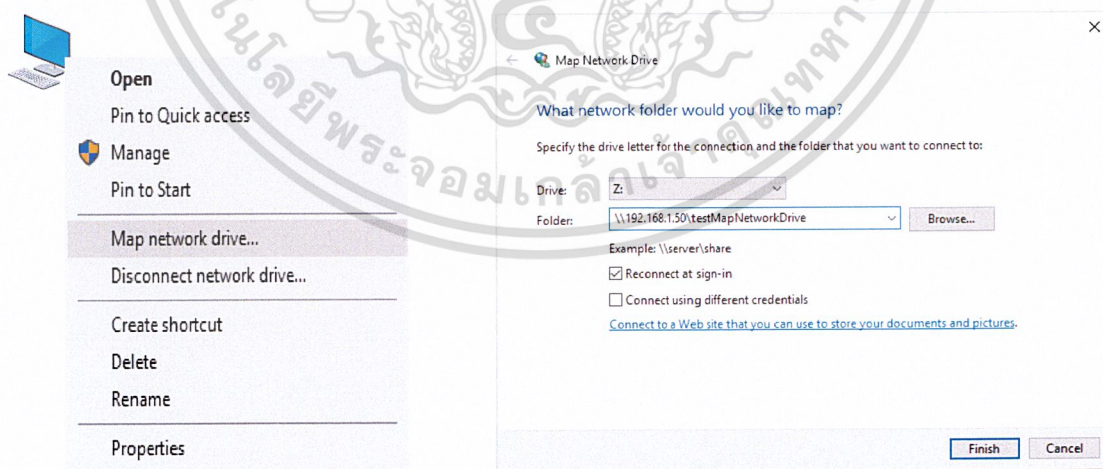
```
G:\Documents and Settings\editor\Desktop>pscp win-data.txt root@192.168.5.62:
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 2048 42:b2:98:95:57:14:aa:0a:6c:fa:e8:3a:7b:d8:0b:63
If you trust this host, enter "y" to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter "n".
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) y
root@192.168.5.62's password:
win-data.txt          | 0 kB |  0.0 kB/s | ETA: 00:00:00 | 100%
G:\Documents and Settings\editor\Desktop>
```

รูปที่ 2.25 ตัวอย่างการถ่ายโอนไฟล์จาก Windows ไปสู่ Linux โดยใช้ pscp

2.4.12 Map Network Drive

Map Network Drive คือการจำลอง Drive ขึ้นมาบนเครื่อง เพื่อเข้าถึง path บนเครื่องอื่นที่มีการเข้าใช้งานบ่อยๆ เพื่อความรวดเร็วในการเข้าถึงและสามารถคัดลอกไฟล์ที่ต้องการเพื่อนำกลับมาเก็บไว้บนเครื่องได้อีกด้วย Map Network Drive สามารถใช้งานได้กับระบบปฏิบัติการ Windows เท่านั้น ซึ่งมีขั้นตอนในการทำไม่ซับซ้อนมาก ดังนี้

- 1) คลิกขวาที่ My Computer เลือก Map network drive... ดังรูปที่ 2.26 (1)
- 2) เลือกชื่อ Drive ที่ใช้ในการอ้างอิง path บนเครื่องอื่น และกรอก path ที่ต้องการให้เข้าถึงในช่อง Folder ดังรูปที่ 2.26 (2)
- 3) คลิกซ้ายที่ปุ่ม Finish



(1)

(2)

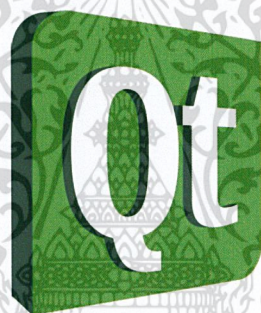
รูปที่ 2.26 การ Map Network Drive บนระบบปฏิบัติการ Windows

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.13 Qt

Qt เป็นเครื่องมือในการสร้างแอปพลิเคชัน และ Graphical User Interface (GUI) ซึ่งสามารถทำงานบน Desktop Personal Computer (Desktop PC), Smart Phone และ Embedded System สามารถทำงานได้หลายระบบปฏิบัติการ (Operating System : OS) หรือเรียกว่า Cross-platform ซึ่งหมายความว่า เมื่อมีโปรแกรมที่ทำงานบน OS หนึ่ง ไม่จำเป็นต้องเขียนโปรแกรมใหม่หรือแก้ไขโปรแกรม ซึ่งสามารถนำโปรแกรมไป Compile ให้สามารถทำงานบน OS อื่นได้ เมื่อโปรแกรมทำงาน รูปร่างหน้าตาของโปรแกรมจะเปลี่ยนไปตามสิ่งแวดล้อม (Environment) ของ OS นั้นๆ โดยอัตโนมัติ

Qt จะมี API และ Library ต่างๆ ที่เขียนด้วยภาษา C++ ทำให้สามารถพัฒนาแอปพลิเคชันแบบ GUI และยังสามารถพัฒนาของภาษา C++, Java, Python, Perl, Pascal และ PHP โดยผู้พัฒนาสามารถเลือกภาษาเพื่อปรับใช้ให้เหมาะสมกับงาน นอกเหนือจากส่วน GUI แล้ว Qt ยังสามารถติดต่อกับฐานข้อมูล SQL, อ่านข้อมูลที่เป็น XML และจัดการไฟล์ได้อีกด้วย



รูปที่ 2.27 Logo ของ Qt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ขั้นตอนในการพัฒนาระบบ

ในการพัฒนาโปรแกรมและขั้นตอนการดำเนินงานเพื่อใช้ในการวิเคราะห์การเรียกใช้ข้อมูลของลูกค้า (Access Log) ของบริษัท Thomson Reuters จะแบ่งออกเป็น 4 หัวข้อ ดังต่อไปนี้

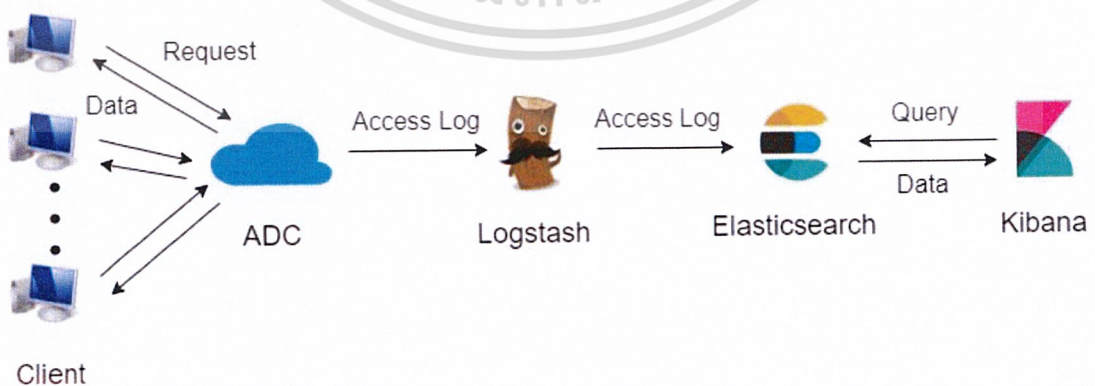
- 3.1 ระบบ Analytics Data Cloud (ADC)
- 3.2 ระบบ Elektron Time Series (ETS)
- 3.3 ระบบ Elektron Tick History (ETH)
- 3.4 โปรแกรม Access Log Analysis From Elasticsearch

3.1 ระบบ Analytics Data Cloud (ADC)

ระบบ ADC เป็นระบบที่ให้บริการข้อมูลเกี่ยวกับรายละเอียดต่างๆของบริษัทที่มีหุ้นอยู่ในตลาด เช่น กำไร ขาดทุน จำนวนหุ้นที่มีอยู่ เป็นต้น เพื่อช่วยในการตัดสินใจซื้อขายหุ้นของลูกค้า โดยผู้พัฒนาได้แบ่งขั้นตอนในการดำเนินงานออกเป็น 4 ขั้นตอน ดังต่อไปนี้

3.1.1 ขั้นตอนการศึกษาการทำงานของระบบ ADC

ระบบ ADC มีอยู่ทั้งหมด 2 ที่ได้แก่ ทวีปอเมริกาและทวีปยุโรป โดยขั้นตอนในการจัดเก็บ Access Log ของระบบ ADC เริ่มจากเมื่อลูกค้า (Client) เรียกใช้ข้อมูล (Request) มาที่ระบบ ADC ระบบจะส่งข้อมูล (Data) กลับไปยัง Client พร้อมกับส่ง Access Log ไปยัง Logstash จากนั้น Logstash ส่ง Access Log ไปจัดเก็บไว้ที่ฐานข้อมูล Elasticsearch ในส่วนของ Kibana นักวิเคราะห์สามารถค้นหา (Query) Access Log จากฐานข้อมูล Elasticsearch เพื่อแสดงผลในรูปแบบตารางหรือกราฟ ดังรูปที่ 3.1



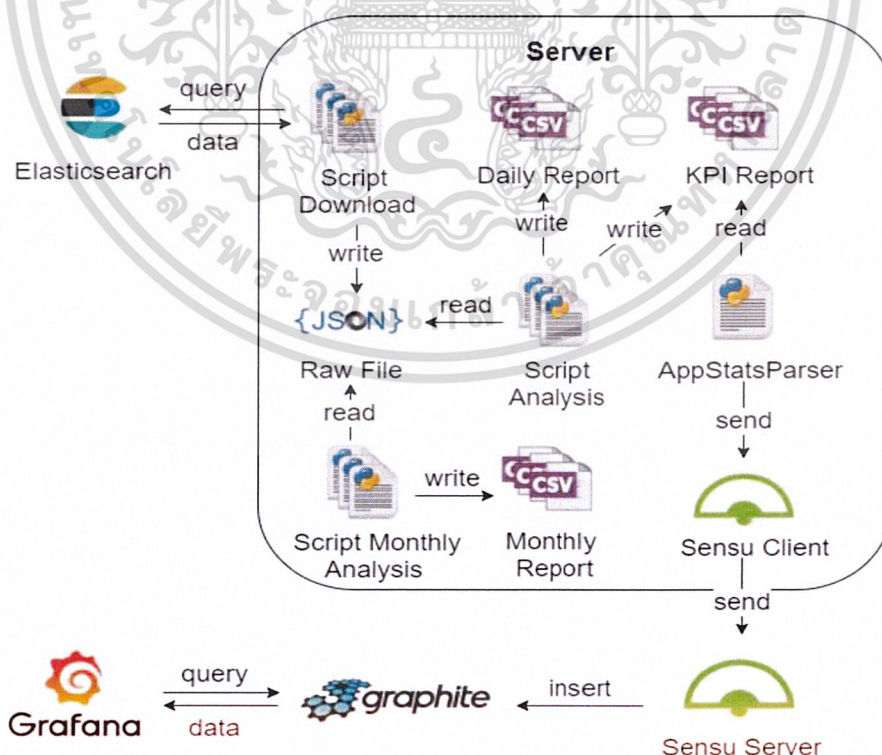
เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.1 การจัดเก็บ Access Log ของระบบ ADC ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ขั้นตอนการวิเคราะห์การทำงานรูปแบบเดิม

ถึงแม้ว่านักวิเคราะห์จะสามารถค้นหา Access Log เพื่อแสดงผลในรูปแบบตารางหรือกราฟผ่านทาง Kibana ได้แต่ผลลัพธ์ที่ได้ยังไม่เป็นไปตามความต้องการของนักวิเคราะห์ ดังนั้นวิธีการวิเคราะห์ Access Log ในรูปแบบเดิมได้ดำเนินการผ่านกลุ่มชุดคำสั่งจำนวน 2 กลุ่ม ได้แก่ กลุ่มที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler และกลุ่มที่สั่งทำงานโดยนักวิเคราะห์

1) กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler จะทำงานทุกวัน ประกอบไปด้วยชุดคำสั่ง (Scripts) จำนวน 2 ชุดได้แก่ ชุดที่ทำการดาวน์โหลด Access Log (Script Download) จากฐานข้อมูล Elasticsearch มาเก็บบนเครื่องเซิร์ฟเวอร์ในรูปแบบไฟล์ JSON (Raw File) และชุดที่ทำการวิเคราะห์ Access Log (Script Analysis) โดยอ่านจาก Raw File ที่ดาวน์โหลดมา และผลลัพธ์ถูกเขียนเป็นกลุ่มไฟล์ CSV จำนวน 2 กลุ่ม ได้แก่ กลุ่มรายงานประจำวัน (Daily Report) และกลุ่มรายงานตัวชี้วัด (KPI Report) ซึ่งข้อมูลในกลุ่ม KPI Report จะถูก AppStatsParser อ่านและส่งไปจัดเก็บไว้ที่ฐานข้อมูล Graphite ผ่านทาง Sensu Client และ Sensu Server จากนั้นจึงใช้ Grafana แสดงผลข้อมูล

2) กลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ ประกอบไปด้วย Scripts จำนวน 1 ชุด ได้แก่ ชุดที่ทำการวิเคราะห์ Access Log เป็นประจำทุกเดือน (Script Monthly Analysis) โดยจะอ่านจาก Raw File ที่ดาวน์โหลดมาและผลลัพธ์ถูกเขียนเป็นไฟล์ CSV (Monthly Report) จำนวน 1 ไฟล์ต่อเดือน

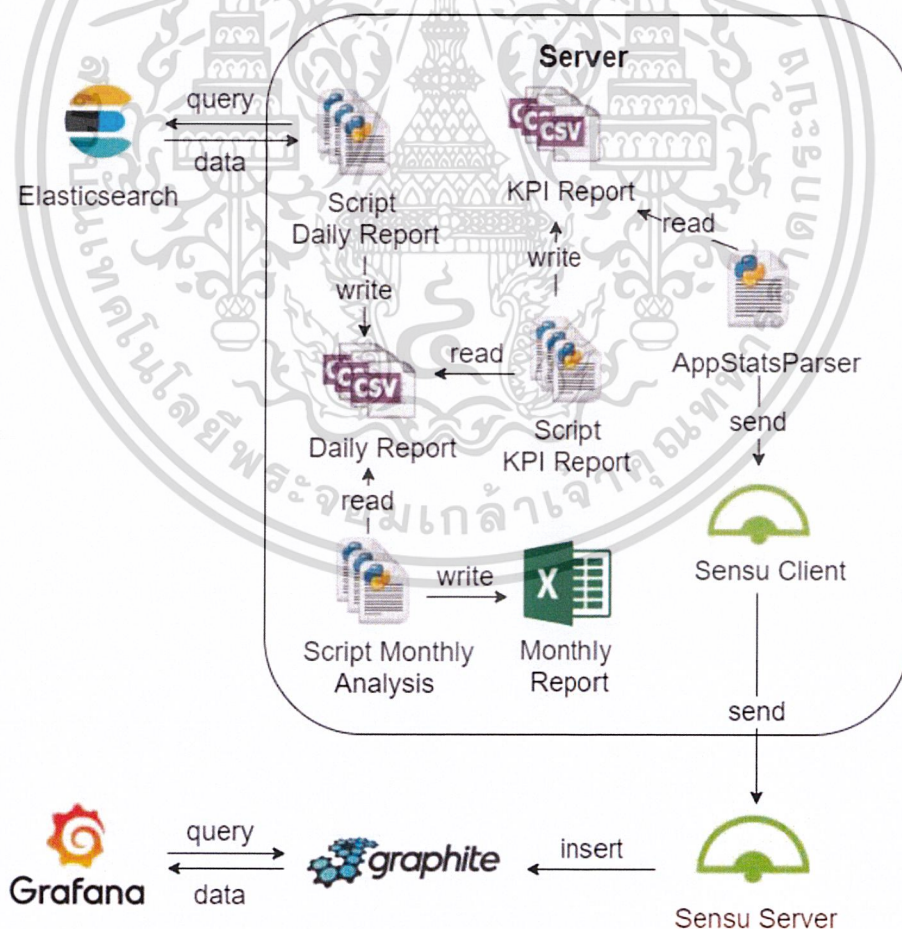


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้นรูปที่ 3.2 การวิเคราะห์ Access Log ของระบบ ADC ในรูปแบบเดิมซึ่งมีการนำไปใช้

ในปัจจุบัน Access Log มีปริมาณมากทำให้มีการใช้หน่วยความจำสำรอง (Harddisk) เป็นจำนวนมากและระยะเวลาในการประมวลผลนาน ทำให้ในบางวันกลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler ทำงานไม่สำเร็จ อันเนื่องมาจากปัจจัยต่างๆ เช่น ระบบเครือข่ายมีปัญหา เป็นต้น และนักวิเคราะห์มีความต้องการเพิ่มรายละเอียดของการวิเคราะห์ Access Log เป็นผลให้ Scripts ไม่ตอบสนองต่อความต้องการของนักวิเคราะห์

3.1.3 ขั้นตอนการวิเคราะห์และออกแบบการทำงานรูปแบบใหม่

จากปัญหาที่พบในการทำงานรูปแบบเดิมจึงมีการปรับปรุงและพัฒนาขั้นตอนการวิเคราะห์ Access Log ของระบบ ADC ขึ้นมาใหม่ จากที่ต้องดาวน์โหลด Access Log มาเก็บไว้บนเครื่องเซิร์ฟเวอร์ (Server) ได้เปลี่ยนเป็นพัฒนา Script Daily Report โดยใช้คำสั่งของฐานข้อมูล Elasticsearch เพื่อช่วยในการวิเคราะห์ Access Log ทำให้ได้ Daily Report เป็นผลลัพธ์ ในส่วนของการวิเคราะห์เพื่อให้ได้ KPI Report และ Monthly Report ได้มีการพัฒนา Scripts จำนวน 2 ชุด ได้แก่ Script KPI Report และ Script Monthly Analysis โดย Script ทั้ง 2 ชุดอ่านข้อมูลจาก Daily Report ดังรูปที่ 3.3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูผู้สอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.3 การวิเคราะห์ Access Log ของระบบ ADC ที่ได้รับการพัฒนา
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย Script Daily Report และ Script KPI Report จัดเป็นกลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler และ Script Monthly Analysis จัดเป็นกลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ รายละเอียดในการวิเคราะห์ Access Log ของแต่ละ Script มีดังต่อไปนี้

- Script Daily Report เป็น Script ที่จะทำงานโดยอัตโนมัติเป็นประจำทุกวัน ผ่านการตั้งเวลาการทำงานด้วย Task Scheduler โดย Script จะวิเคราะห์ข้อมูลจากฐานข้อมูล Elasticsearch ที่มีอยู่ 2 แห่ง ได้แก่ ในทวีปอเมริกาและในทวีปยุโรป โดยการวิเคราะห์จะแบ่งออกเป็น 2 กลุ่ม ได้แก่ กลุ่มของลูกค้าและกลุ่มของผลิตภัณฑ์ โดยในแต่ละกลุ่มจะวิเคราะห์ปริมาณจุดเวลา (Datapoints) และปริมาณการเรียกใช้ข้อมูล (Request) ซึ่งปริมาณจุดเวลา คือ จำนวนวันที่ลูกค้าขอข้อมูลใน 1 request และในส่วนของผลิตภัณฑ์ คือ ซอฟต์แวร์ที่ลูกค้าใช้ในการเรียกใช้ข้อมูล

- Script KPI Report เป็น Script ที่ทำงานโดยอัตโนมัติเป็นประจำทุกวันผ่านการตั้งเวลาการทำงานด้วย Task Scheduler โดย Script จะอ่านไฟล์ Daily Report แล้วสรุปปริมาณจุดเวลาและปริมาณการร้องขอตามประเภทของผลิตภัณฑ์และตามผลิตภัณฑ์

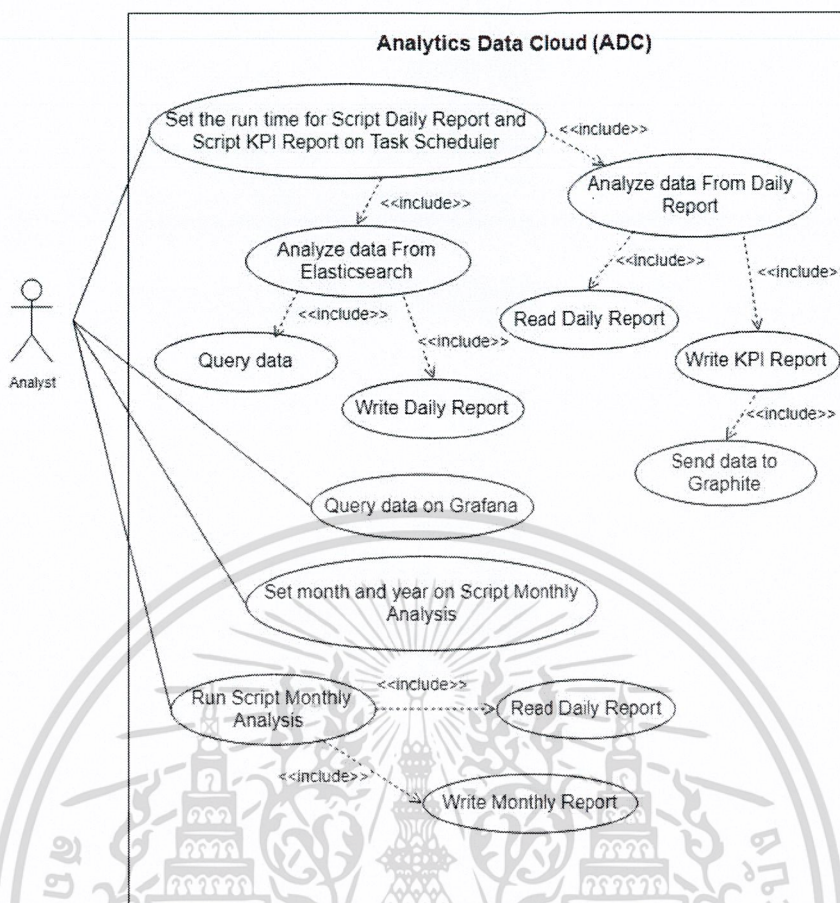
- Script Monthly Analysis เป็น Script ที่ทำงานโดยการสั่งจากนักวิเคราะห์ผ่าน Command Prompt โดย Script จะอ่านไฟล์ Daily Report ของทุกวันที่อยู่ในเดือนที่นักวิเคราะห์กำหนดและที่อยู่ในเดือนก่อนหน้า โดยจะเปรียบเทียบปริมาณการร้องขอและปริมาณจุดเวลาตามประเภทของผลิตภัณฑ์, ตามผลิตภัณฑ์ และตามลูกค้า

ขั้นตอนการทำงานของชุดคำสั่งจะอธิบายด้วยแผนภาพ (Diagram) เพื่อให้สามารถมองเห็นภาพรวมการทำงานของชุดคำสั่ง ดังนี้

1) แผนภาพการทำงานของผู้ใช้ระบบ (Use Case Diagram)

เนื่องจาก Script ได้ถูกแบ่งออกเป็น 2 กลุ่ม ได้แก่ กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler และ กลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ ทำให้กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler นักวิเคราะห์จะต้องตั้งเวลาการทำงานของ Script ผ่านทาง Task Scheduler และดูผลลัพธ์จากการวิเคราะห์ผ่านทาง Dashboard ของ Grafana, Daily Report หรือ KPI Report ในส่วนกลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ นักวิเคราะห์จะสามารถตั้งค่าเดือนและปีที่ต้องการจะวิเคราะห์ได้จากนั้นสั่งทำงานผ่านทาง Command Prompt และดูผลลัพธ์ได้จากไฟล์ที่ถูกเขียนขึ้นมา ดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Use Case Diagram กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler และกลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ของระบบ ADC

จากรูปที่ 3.4 สามารถอธิบายได้ดังนี้

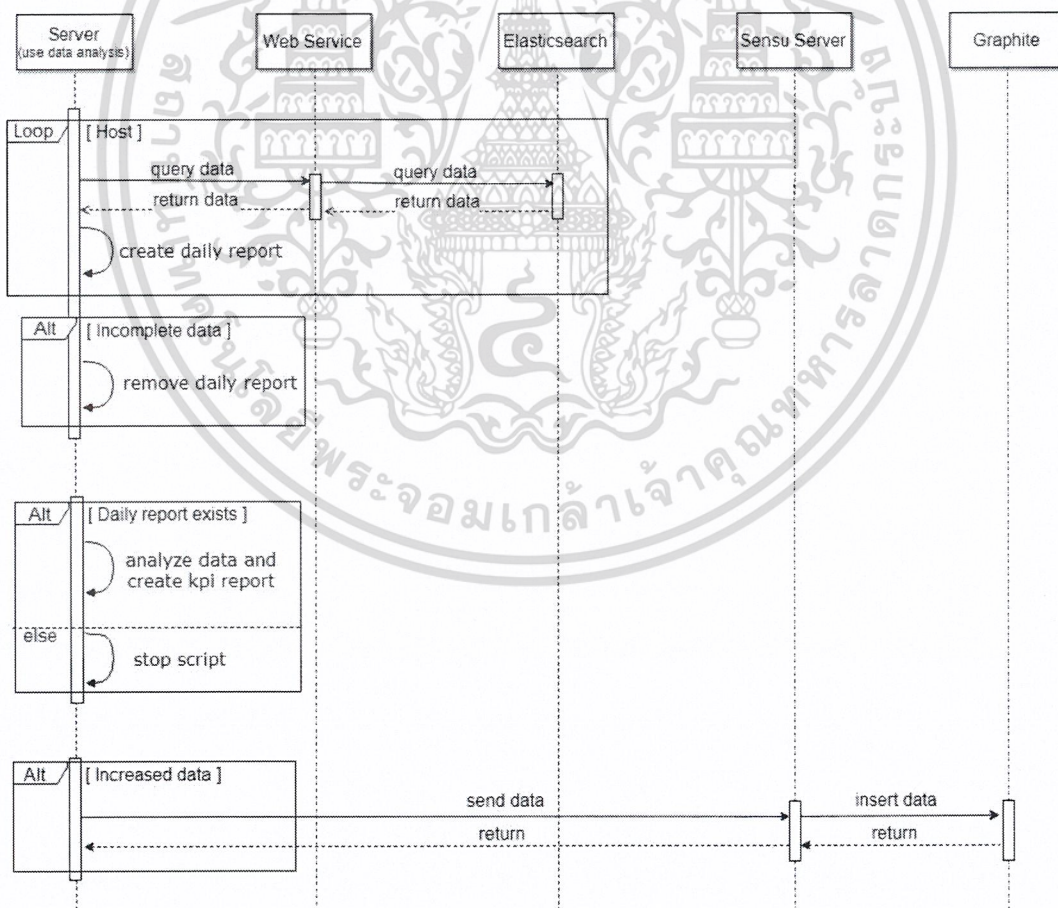
1. Set the run time for Script Daily Report and Script KPI Report on Task Scheduler คือ การตั้งเวลาการทำงานของ Script Daily Report และ Script KPI Report ผ่านทาง Task Scheduler โดยนักวิเคราะห์ ซึ่งทั้ง 2 Script จะทำงานเป็นประจำทุกวัน
2. Analyze data From Elasticsearch คือ การวิเคราะห์ข้อมูลจากการ query data จาก Elasticsearch แล้วเขียนเป็น Daily Report โดยจะทำงานก็ต่อเมื่อ Script Daily Report ทำงาน
3. Analyze data From Daily Report คือ การวิเคราะห์ข้อมูลจากการอ่าน Daily Report แล้วเขียนเป็น KPI Report ตามที่นักวิเคราะห์ต้องการ โดยจะทำงานก็ต่อเมื่อ Script KPI Report ทำงาน เมื่อมีการเขียน KPI Report จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้ทำให้ AppStatsParser ส่งข้อมูลไปที่ฐานข้อมูล Graphite โดยผ่านทางกรค่าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้ง Sensu Client ึ่งเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Query data on Grafana คือ นักวิเคราะห์สามารถค้นหาข้อมูลที่เก็บอยู่บนฐานข้อมูล Graphite แล้วนำมาแสดงเป็นกราฟต่างๆได้
5. Set month and year on Script Monthly Analysis คือ การกำหนดเดือนและปีให้กับ Script Monthly Analysis เพื่อวิเคราะห์ข้อมูลตามที่นักวิเคราะห์ต้องการ
6. Run Script Monthly Analysis คือ การสั่งให้ Script Monthly Analysis ทำงานผ่านทาง Command Prompt โดยนักวิเคราะห์ ซึ่ง Script จะอ่าน Daily Reports ของทุกวันที่อยู่ในเดือนที่ได้กำหนดไว้และที่อยู่ในเดือนก่อนหน้าที่กำหนด จากนั้นจะเขียนเป็นไฟล์ Monthly Report

2) แผนภาพความสัมพันธ์ระหว่างวัตถุ (Sequence Diagram)

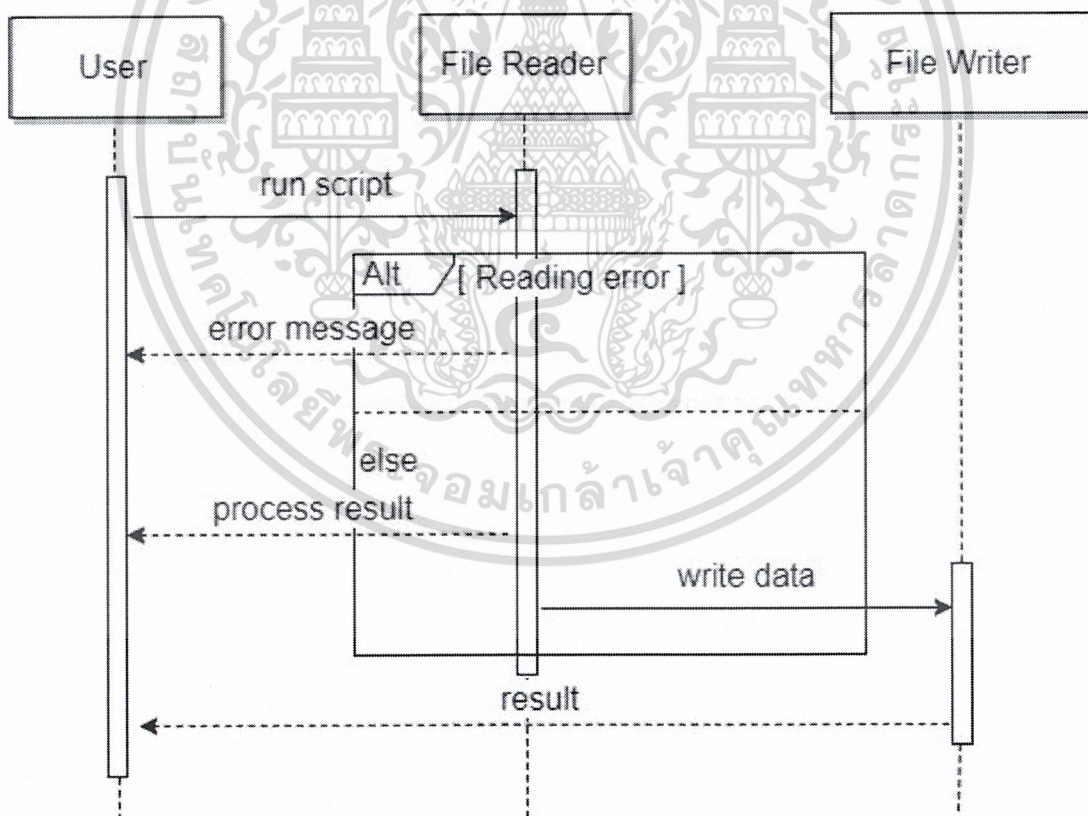
เป็นการออกแบบขั้นตอนการทำงานกลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler และกลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ เพื่อแสดงถึงลำดับขั้นตอน เวลา และโปรแกรมที่เกี่ยวข้องในการทำงาน ดังรูปที่ 3.5 และรูปที่ 3.6 ตามลำดับ



เอกสารรูปที่ 3.5 Sequence Diagram กลุ่มชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler การค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาของระบบ ADC อิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.5 สามารถอธิบายได้ดังนี้

- เมื่อ Script Daily Report ทำงานจะส่งคำสั่งค้นหาข้อมูล (query data) ไปที่ฐานข้อมูล Elasticsearch ในทวีปอเมริกาและทวีปยุโรป โดยผ่านทาง Host ซึ่งเปรียบเสมือนที่อยู่ที่ใช้ในการติดต่อ โดยเมื่อคำสั่งค้นหาข้อมูลถูกส่งออกไปจะผ่าน Web Service ของแต่ละ Host ก่อนจะเข้าถึงฐานข้อมูล Elasticsearch เมื่อได้รับข้อมูลกลับมาแล้วจะเขียนเป็นไฟล์ Daily Report จากนั้นตรวจสอบความสมบูรณ์ของข้อมูล หากไม่สมบูรณ์จะลบไฟล์ Daily Report
- เมื่อ Script KPI Report ทำงานจะตรวจสอบก่อนว่ามีไฟล์ Daily Report ตรงกับวันที่ต้องการหรือไม่ หากมีไฟล์จะทำการวิเคราะห์ข้อมูลและเขียนเป็นไฟล์ KPI Report ถ้าไม่มีไฟล์จะจบการทำงานของ Script
- เมื่อ AppStatsParser ทำงานจะตรวจสอบก่อนว่า KPI Report มีข้อมูลเพิ่มขึ้นหรือไม่ ถ้ามีจะส่งข้อมูลส่วนที่เพิ่มไปเก็บบนฐานข้อมูล Graphite ผ่านทาง Sensu



รูปที่ 3.6 Sequence Diagram กลุ่มชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ของระบบ ADC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.6 สามารถอธิบายได้ดังนี้

เมื่อนักวิเคราะห์สั่ง Script Monthly Analysis ทำงานผ่าน Command Prompt Script จะอ่านไฟล์ Daily Report ของทุกวันที่อยู่ในเดือนที่ได้กำหนดไว้และที่อยู่ในเดือนก่อนหน้าที่กำหนด หากการอ่านไฟล์เกิดข้อผิดพลาดจะส่งข้อความแจ้งข้อผิดพลาดให้นักวิเคราะห์ทราบ แต่หากไม่เกิดข้อผิดพลาดจะวิเคราะห์ข้อมูล จากนั้นส่งผลการทำงานให้นักวิเคราะห์ทราบและเขียนไฟล์ผลลัพธ์ Monthly Report

3.1.4 ขั้นตอนการพัฒนาซอฟต์แวร์

จากขั้นตอนการศึกษาการทำงานของระบบ ADC ขั้นตอนการวิเคราะห์การทำงานรูปแบบเดิม และขั้นตอนการวิเคราะห์และออกแบบการทำงานรูปแบบใหม่ ทำให้ผู้พัฒนาทราบถึงปัญหาและขั้นตอนต่างๆในการพัฒนาชุดคำสั่งเพื่อใช้ในการวิเคราะห์ Access Log ผู้พัฒนาได้ใช้เครื่องมือต่างๆในการพัฒนาชุดคำสั่งและอำนวยความสะดวกต่อนักวิเคราะห์ในการเข้าถึงข้อมูล ดังนี้

- โปรแกรม JetBrains PyCharm Community Edition เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมด้วยภาษา Python
- AppStatsParser และ Sensus ใช้ในการส่งข้อมูลที่ผ่านการวิเคราะห์แล้วไปเก็บไว้ในฐานข้อมูล Graphite
- Grafana ใช้ในการแสดงผลข้อมูลที่เก็บอยู่ในฐานข้อมูล Graphite ในรูปแบบของกราฟ

3.2 ระบบ Elektron Time Series (ETS)

ระบบ ETS เป็นระบบที่ให้บริการข้อมูลการซื้อขายหุ้นต่างๆในตลาด โดยจะเก็บข้อมูลเพียง 3 เดือนนับจากปัจจุบันเท่านั้น โดยผู้พัฒนาได้แบ่งขั้นตอนในการดำเนินงานออกเป็น 3 ขั้นตอนดังต่อไปนี้

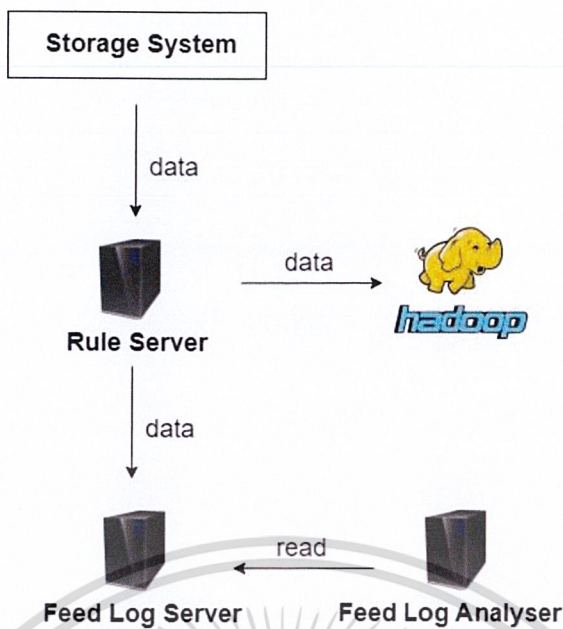
3.2.1 ขั้นตอนการศึกษาการทำงานของระบบ ETS

ระบบ ETS มีอยู่ทั้งหมด 2 ที่ได้แก่ ทวีปอเมริกาและทวีปยุโรป โดยจะขออธิบายการทำงานของระบบออกเป็น 2 ส่วน ได้แก่ ส่วนการจัดเก็บข้อมูลหุ้นลงในระบบและส่วนการจัดเก็บ Access Log

ส่วนการจัดเก็บข้อมูลหุ้นลงในระบบ เริ่มจากระบบจัดเก็บข้อมูล (Storage System) ส่งข้อมูลหุ้นมาที่ Rule Server เพื่อจัดกลุ่มของหุ้น จากนั้นส่งข้อมูลดังกล่าวไปที่ฐานข้อมูล Hadoop และ Feed Log Server โดยฐานข้อมูล Hadoop จะเป็นฐานข้อมูลหลักในการจัดเก็บข้อมูลของระบบ ETS ส่วน Feed Log Server จะทำหน้าที่สำรองข้อมูล ในขณะเดียวกันจะมีระบบ

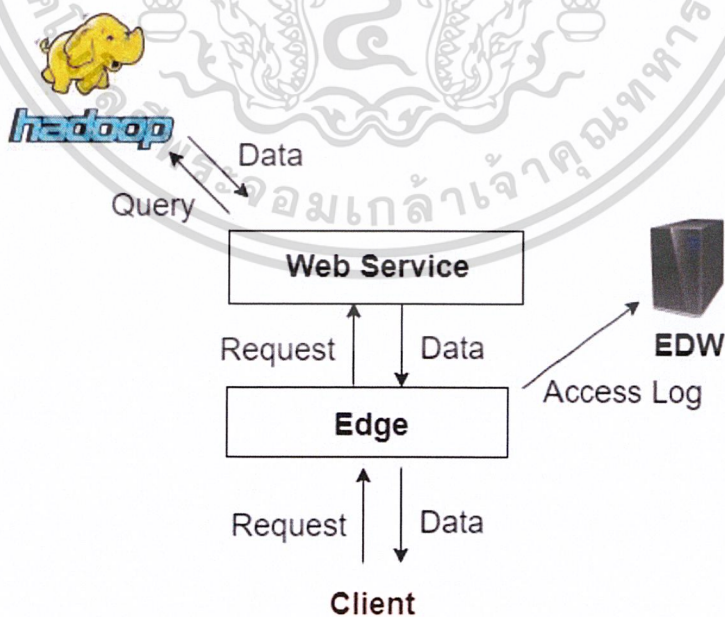
Feed Log Analyser ทำการอ่านข้อมูลที่ Feed Log Server เพื่อสรุปว่าในแต่ละวันมีการจัดเก็บหุ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
แม้ว่ากรรมใดๆ หักสน ออกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 การจัดเก็บและการสรุปปริมาณข้อมูลทันทีเข้าสู่ระบบ ETS

ส่วนการจัดเก็บ Access Log เริ่มจากเมื่อลูกค้า (Client) เรียกใช้ข้อมูล (Request) มาที่ระบบ ETS Request นั้นจะผ่าน Edge ซึ่งมีหน้าที่ 2 ประการได้แก่ ประการที่ 1 ส่ง Request ต่อไปให้ Web Service จากนั้น Web Service ทำการค้นหา (Query) ข้อมูลจากฐานข้อมูล Hadoop และส่งข้อมูลกลับไปให้ Edge Edge ส่งข้อมูลกลับไปให้ Client ประการที่ 2 ส่ง Access Log ไปเก็บไว้ที่ EDW ดังรูปที่ 3.8



รูปที่ 3.8 การจัดเก็บ Access Log ของระบบ ETS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์

ในการวิเคราะห์ข้อมูลในระบบ ETS นั้นนักวิเคราะห์ต้องการทราบถึงปริมาณการใช้งานของลูกค้าในลักษณะต่างๆ เช่น ใช้แอปพลิเคชันอะไรในการติดต่อขอข้อมูล ข้อมูลที่ลูกค้าขอจัดอยู่ในกลุ่มหุ้นอะไร เป็นต้น นอกจากนี้แล้วนักวิเคราะห์ต้องการทราบถึงพื้นที่ในการจัดเก็บข้อมูลหุ้นว่ามีความคุ้มค่าหรือไม่ โดยข้อมูลที่ใช้ในการวิเคราะห์มีทั้งหมด 3 แหล่ง ได้แก่ Rule Server, Feed Log Analyser และ EDW ซึ่งแต่ละแหล่งจะได้ข้อมูลดังนี้

- Rule เป็นไฟล์ XML ที่อยู่ใน Rule Server โดยปกติจะมีการเปลี่ยนแปลงทุกวัน ไฟล์ Rule มีรายละเอียดเกี่ยวกับกลุ่มของหุ้น เช่น ชื่อตลาดหุ้น ประเทศ เป็นต้น โดยข้อมูลที่เราสนใจในไฟล์ Rule คือ RuleID และ RuleName ดังรูปที่ 3.9 ตัวอย่างไฟล์ Rule

```
<ruleset_publication xmlns="...">
  <publication_info execution_time="2017-12-11T23:00:00Z" by="3753" time="2017-12-11T20:27:22Z" version="2051.0">
    <details>[ tsfcpReferenceSessionRulesApp , 1339.0 | tsfcpReferenceDefaultViewRulesApp , 1339.0 | tsfcpR
  </publication_info>
  <batch ruleset><rule schema xmlns="..." xmlns:xsi="...">
    <identity RuleID="12229" RuleName="AE ABD EQ 113 1 1" RuleVersionID="13.0"/>
    <schedule start_time="2008-01-30T17:50:24Z"/>
    <context_key>
      <part key_pos="3">
        <value>38</value>
        <value>2</value>
      </part>
      <part key_pos="5">
        <value>113</value>
      </part>
      <part key_pos="1">
        <value>460</value>
      </part>
    </context_key>
    <rule_clauses>
      <data_filter clause_message_type="update">
        <clause_header clause_number="1" process_stage="DataCapture" process_stage_group="1"/>
        <dmmt_annotations>TAS record for Regular Trades - Default</dmmt_annotations>
        <accept_conditions operator="or">
          <unary_condition condition_operator="exists">
            <operand>
              <value fid="6" type="currentValue_Fid"/>
            </operand>
          </unary_condition>
          <unary_condition condition_operator="exists">
            <operand>
              <value fid="178" type="currentValue_Fid"/>
            </operand>
          </unary_condition>
        </accept_conditions>
      </data_filter>
    </rule_clauses>
  </batch ruleset>
</ruleset_publication>
```

รูปที่ 3.9 ตัวอย่างไฟล์ Rule ของระบบ ETS

- Feed Log Analysis เป็นกลุ่มของไฟล์ Excel ที่ถูกเขียนขึ้นโดย Feed Log Analyser เป็นประจำทุกวัน โดย 1 ไฟล์ก็คือ 1 RuleID และรายละเอียดในแต่ละไฟล์จะประกอบไปด้วยชื่อหุ้น(RicName) และผลรวมปริมาณข้อมูลที่เข้ามาในระบบตามตัวชี้วัดของบริษัท ดังรูปที่ 3.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12837#1.0#2056.0#C994RUCDPUL01.csv

12838#1.0#2056.0#C994RUCDPUL01.csv

12840#5.0#2056.0#C994RUCDPUL01.csv

12841#8.0#2056.0#C994RUCDPUL01.csv

12842#4.0#2056.0#C994RUCDPUL01.csv

12843#1.0#2056.0#C994RUCDPUL01.csv

12844#2.0#2056.0#C994RUCDPUL01.csv

	A	B	C	D	E	F	G	H	I	J
	RicName	#TAS	#TAQ	#OOC	#CanCor	#Scaling	#RicRenan	#RuleChar	#TimeSer	#BatchRule
1	NOK35C2M=R	0	187	1	0	0	0	0	0	0
2	NOK35P2M=R	0	187	1	0	0	0	0	0	0
3	DKKNOK20P2Y=R	0	132	1	0	0	0	0	0	0
4	DKKNOK45PSW=R	0	132	1	0	0	0	0	0	0
5	DKKNOK45P2W=R	0	132	1	0	0	0	0	0	0
6	DKKNOK20P2W=R	0	132	1	0	0	0	0	0	0
7	DKKNOK20PSW=R	0	132	1	0	0	0	0	0	0
8	DKKNOK45P2Y=R	0	132	1	0	0	0	0	0	0
9	IDR30C1Y=R	0	25	1	0	0	0	0	0	0
10	IDR30P1Y=R	0	25	1	0	0	0	0	0	0
11										

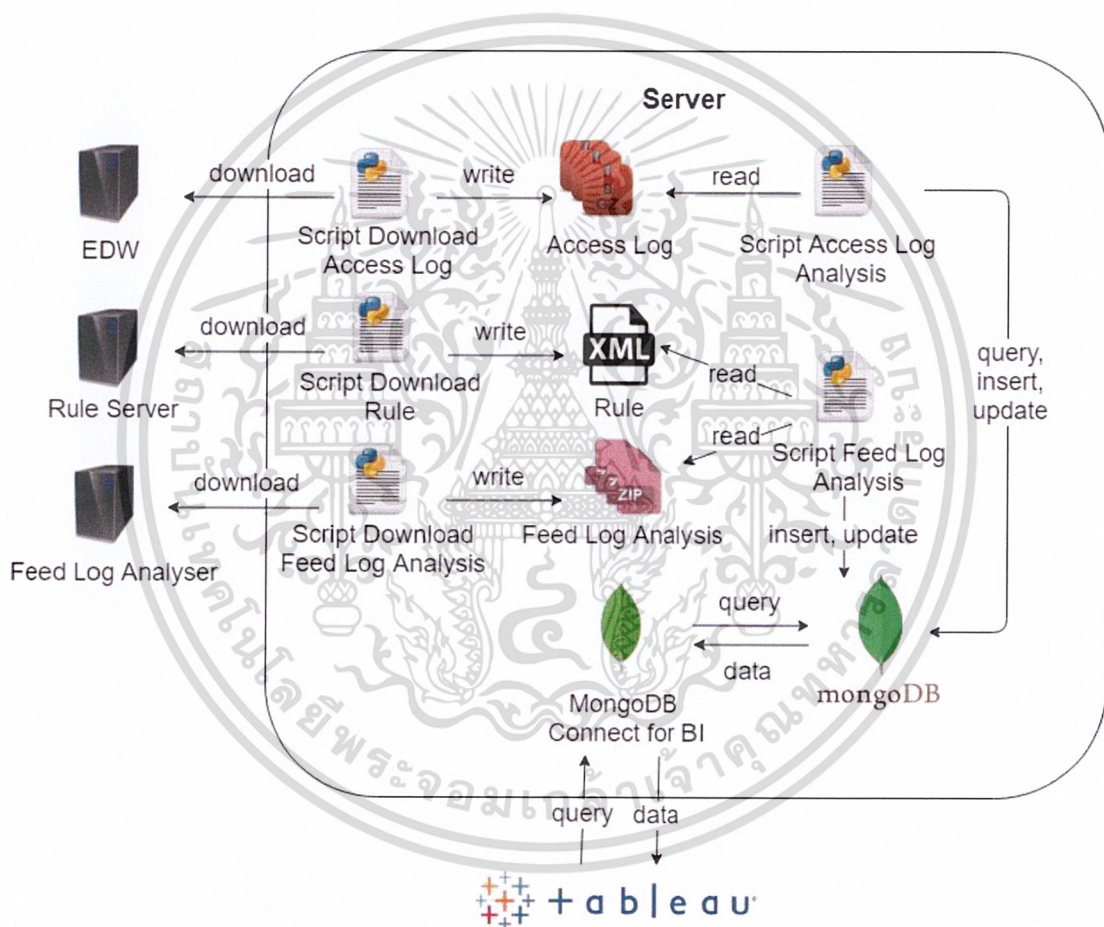
รูปที่ 3.10 ตัวอย่าง Feed Log Analysis ของระบบ ETS

- Access Log เป็นกลุ่มของไฟล์ที่มีการบีบอัดข้อมูล (GZ) เป็นประจำทุกวัน ซึ่งภายในประกอบไปด้วยไฟล์ JSON และภายในไฟล์ JSON มีบันทึกการเรียกใช้ข้อมูลของลูกค้าอยู่เป็นจำนวนมาก ดังรูปที่ 3.11 แสดงตัวอย่างบันทึกการเรียกใช้ข้อมูลของลูกค้าจำนวน 1 แถว

```
{
  "points": null,
  "frequency": null,
  "targetRole": ["ActiveCache"],
  "requestReceivedTime": 1514030400.011,
  "instrumentCode": ["GDMOF.PK"],
  "requestId": "5D17D8F6-C322-4BC3-B225-DC07B589D270",
  "freqMult": null,
  "CapabilityName": "TSfCP",
  "factList": [],
  "adjustment": [],
  "CapabilityVersion": "2.0",
  "features": [],
  "applicationId": "TSI7Service",
  "endTime": null,
  "blendingRetryCount": null,
  "hostname": "c406wuhdpd01",
  "blending": null,
  "EventType": "TSReq",
  "qoS": null,
  "businessProcessId": "Unknown",
  "event": null,
  "soapMethod": "\"tns:GetMetaData_1\"",
  "uuid": null,
  "tsiVersion": "6.4",
  "startTime": null
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ว่าระบบควรใช้ระบบเพื่อการศึกษาเท่านั้น โปรดปฏิบัติตามเงื่อนไขการใช้งาน
รูปที่ 3.11 ตัวอย่างบันทึกการเรียกใช้ข้อมูลของลูกค้าจำนวน 1 แถวของระบบ ETS
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการวิเคราะห์ข้อมูลของระบบ ETS ประกอบไปด้วย Scripts 5 ชุดที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler ได้แก่ ชุดที่ดาวน์โหลด Access Log จาก EDW (Script Download Access Log), ชุดที่ดาวน์โหลด Rule จาก Rule Server (Script Download Rule), ชุดที่ดาวน์โหลด Feed Log Analysis จาก Feed Log Analyser (Script Download Feed Log Analysis), ชุดที่วิเคราะห์ข้อมูลจาก Access Log (Script Access Log Analysis) และ ชุดที่วิเคราะห์ข้อมูลจาก Rule และ Feed Log Analysis (Script Feed Log Analysis) โดยผลลัพธ์ในการวิเคราะห์ข้อมูลจะเก็บอยู่ในฐานข้อมูล MongoDB จากนั้นใช้ Tableau ค้นหาข้อมูล (Query) ในฐานข้อมูล MongoDB โดยผ่าน MongoDB Connect for BI เพื่อแสดงผลข้อมูลในรูปแบบกราฟ ดังรูปที่ 3.12

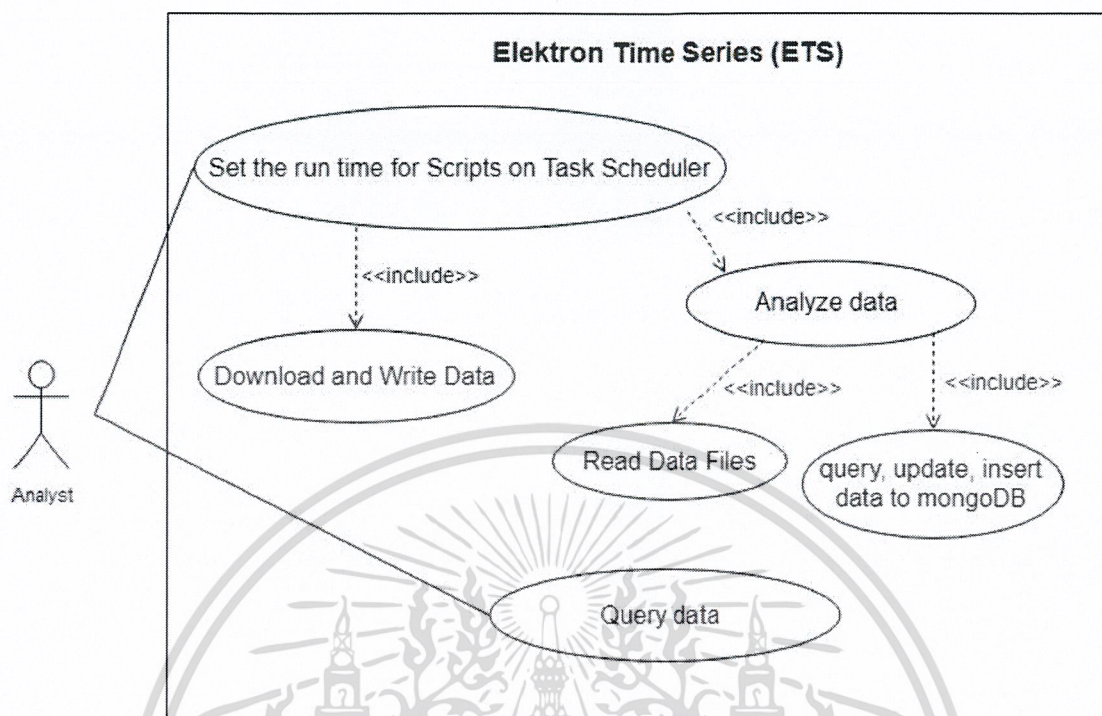


รูปที่ 3.12 การวิเคราะห์ข้อมูลของระบบ ETS

ขั้นตอนการทำงานของชุดคำสั่งจะอธิบายด้วยแผนภาพ (Diagram) เพื่อให้สามารถมองเห็นภาพรวมการทำงานของชุดคำสั่ง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) แผนภาพการทำงานของผู้ใช้ระบบ (Use Case Diagram)



รูปที่ 3.13 Use Case Diagram การวิเคราะห์ข้อมูลของระบบ ETS

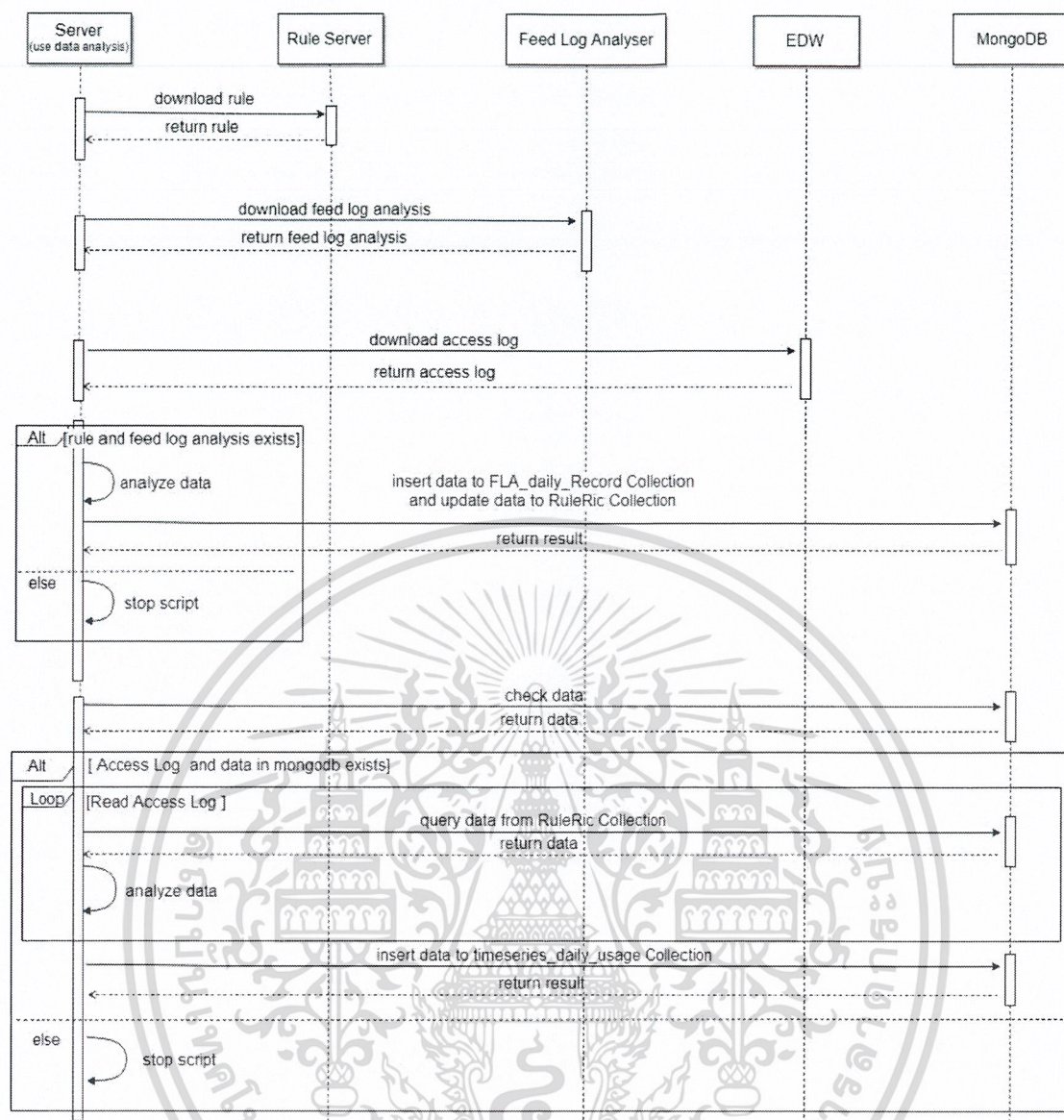
จากรูปที่ 3.13 สามารถอธิบายได้ดังนี้

1. Set the run time for Scripts on Task Scheduler คือ การตั้งเวลาการทำงานของ Script ทั้ง 5 ชุด
2. Download and Write Data คือ การดาวน์โหลดข้อมูลที่ใช้ในการวิเคราะห์ โดยใช้ Script Download Rule, Script Download Access Log และ Script Download Feed Log Analysis
3. Analyze data คือ การวิเคราะห์ข้อมูลที่ได้จากการดาวน์โหลดโดยเริ่มจากการอ่านไฟล์ข้อมูล จากนั้นวิเคราะห์ข้อมูลตามความต้องการของนักวิเคราะห์ เมื่อวิเคราะห์เสร็จแล้วนำผลการวิเคราะห์เก็บเข้าสู่ฐานข้อมูล MongoDB
4. Query data คือ นักวิเคราะห์สามารถแสดงข้อมูลที่เก็บอยู่ในฐานข้อมูล MongoDB ออกมาในรูปแบบกราฟผ่านทาง Tableau

2) แผนภาพความสัมพันธ์ระหว่างวัตถุ (Sequence Diagram)

เป็นการออกแบบขั้นตอนการทำงานของ Script ทั้ง 5 ชุด เพื่อแสดงถึง

เอกสารที่ออกสารที่ส่งไปใช้กับระบบที่เกี่ยวข้องกับการดำเนินงานด้านไปรษณีย์และไปรษณีย์ภัณฑ์
ลำดับขั้นตอน เวลา และโปรแกรมที่เกี่ยวข้องในการทำงาน ดังรูปที่ 3.14
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 Sequence Diagram การวิเคราะห์ข้อมูลของระบบ ETS

จากรูปที่ 3.14 สามารถอธิบายได้ดังนี้

- เมื่อ Script Download Rule ทำงานจะทำการดาวน์โหลดไฟล์ Rule ซึ่งเป็นไฟล์ XML จาก Rule Server มาเก็บไว้บนเครื่องที่ใช้ในการวิเคราะห์
- เมื่อ Script Download Feed Log Analysis ทำงานจะทำการดาวน์โหลดไฟล์ Feed Log Analysis จาก Feed Log Analyser มาเก็บไว้บนเครื่องที่ใช้ในการวิเคราะห์
- เมื่อ Script Download Access Log ทำงานจะทำการดาวน์โหลด Access Log จาก EDW มาเก็บไว้บนเครื่องที่ใช้ในการวิเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อ Script Feed Log Analysis ทำงานจะทำการตรวจสอบว่ามีไฟล์ Rule กับไฟล์ Feed Log Analysis ตรงกับวันที่ต้องการวิเคราะห์หรือไม่ หากมี จะดำเนินการอ่านไฟล์ Rule และไฟล์ Feed Log Analysis ตามวันที่ ต้องการ จากนั้นทำการวิเคราะห์โดยคำนวณปริมาณข้อมูลที่เข้ามาในระบบ ตามตัวชี้วัดของบริษัทและทำการจัดซื้อหุ้นตามกลุ่มของหุ้นแล้วเก็บข้อมูล เข้าฐานข้อมูล MongoDB ที่ FLA_daily_Record Collection และ RuleRic Collection ตามลำดับ
- เมื่อ Script Access Log Analysis ทำงานจะทำการตรวจสอบ 2 อย่าง คือ ตรวจสอบว่าข้อมูลใน FLA_daily_Record Collection และใน RuleRic Collection มีการเพิ่มข้อมูลตรงกับวันที่ต้องการวิเคราะห์หรือยัง และ ตรวจสอบว่าไฟล์ Access Log มีอยู่หรือไม่ หากการตรวจสอบผ่านทั้ง 2 อย่างจะเริ่มอ่านไฟล์ Access Log โดยขณะเดียวกันจะมีการค้นหาข้อมูลใน RuleRic Collection เพื่อดูว่าหุ้นที่ลูกค้าขอข้อมูลอยู่ในกลุ่มของหุ้นอะไร และทำการวิเคราะห์โดยคำนวณปริมาณการใช้งานของลูกค้าในลักษณะ ต่างๆ เมื่อทำการวิเคราะห์ข้อมูลเสร็จแล้วจะนำผลการวิเคราะห์เพิ่มเข้าไป ในฐานข้อมูล MongoDB ที่ timeseries_daily_usage Collection

3.2.3 ขั้นตอนการพัฒนาซอฟต์แวร์

จากขั้นตอนการศึกษาการทำงานของระบบ ETS ขั้นตอนการวิเคราะห์และ ออกแบบซอฟต์แวร์ ทำให้ผู้พัฒนาทราบถึงขั้นตอนต่างๆในการพัฒนาชุดคำสั่งเพื่อใช้ในการวิเคราะห์ ปริมาณการใช้งานของลูกค้าในลักษณะต่างๆและการใช้พื้นที่ในการจัดเก็บข้อมูลหุ้นว่ามีความคุ้มค่า หรือไม่ ผู้พัฒนาได้ใช้เครื่องมือต่างๆในการพัฒนาชุดคำสั่งและอำนวยความสะดวกต่อนักวิเคราะห์ใน การเข้าถึงข้อมูล ดังนี้

- โปรแกรม JetBrains PyCharm Community Edition เป็นเครื่องมือที่ใช้ในการ พัฒนาโปรแกรมด้วยภาษา Python
- โปรแกรม pscp ใช้ในการถ่ายโอนไฟล์ Access Log และไฟล์ Rule
- ฐานข้อมูล MongoDB ใช้สำหรับเก็บผลการวิเคราะห์
- MongoDB Connect for BI เป็นตัวกลางสำหรับให้ Tableau สามารถค้นหา ข้อมูลในฐานข้อมูล MongoDB ได้
- Tableau ใช้ในการแสดงผลข้อมูลที่เก็บอยู่ในฐานข้อมูล MongoDB ในรูปแบบ ของกราฟ

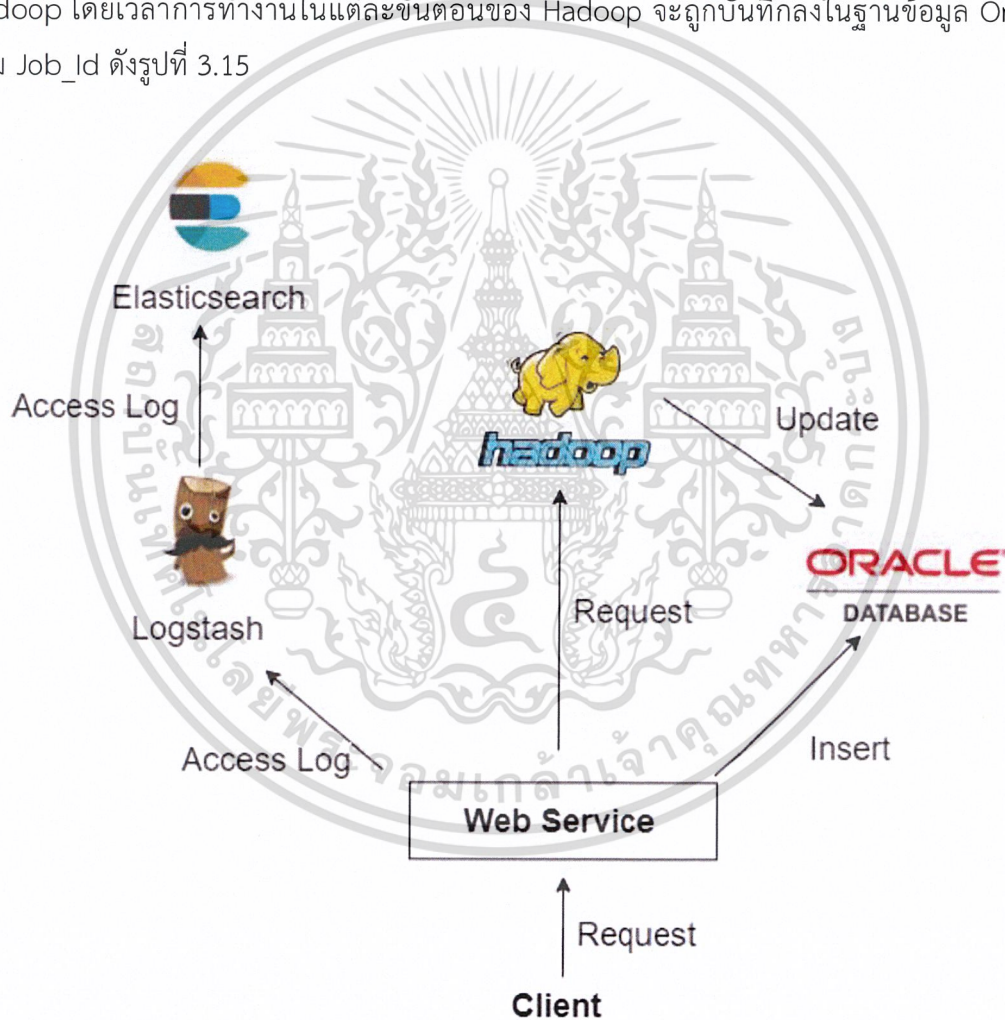
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ระบบ Elektron Tick History (ETH)

ระบบ ETH เป็นระบบที่ให้บริการข้อมูลการซื้อขายหุ้นต่างๆในตลาด โดยจะเก็บข้อมูลมาตั้งแต่ปี 1996 จนถึงปัจจุบัน ผู้พัฒนาได้แบ่งขั้นตอนในการดำเนินงานออกเป็น 3 ขั้นตอน ดังต่อไปนี้

3.3.1 ขั้นตอนการศึกษาการทำงานของระบบ ETH

ระบบ ETH มีอยู่ที่ทวีปอเมริกา โดยขั้นตอนการจัดเก็บ Access Log ของระบบ ETH เริ่มจากลูกค้า (Client) เรียกใช้ข้อมูล (Request) มาที่ Web Service เมื่อ Web Service ได้รับ Request จะมีการทำงานอยู่ 2 ส่วน ได้แก่ ส่วนที่ 1 ส่ง Access Log ไปจัดเก็บที่ฐานข้อมูล Elasticsearch โดยผ่าน Logstash และส่วนที่ 2 สร้าง Job_Id แล้วบันทึกลงในฐานข้อมูล Oracle พร้อมกับเวลาที่ Request ติดต่อเข้ามาที่ Web Service จากนั้นส่งรายละเอียดการขอข้อมูลไปที่ Hadoop โดยเวลาการทำงานในแต่ละขั้นตอนของ Hadoop จะถูกบันทึกลงในฐานข้อมูล Oracle ตาม Job_Id ดังรูปที่ 3.15



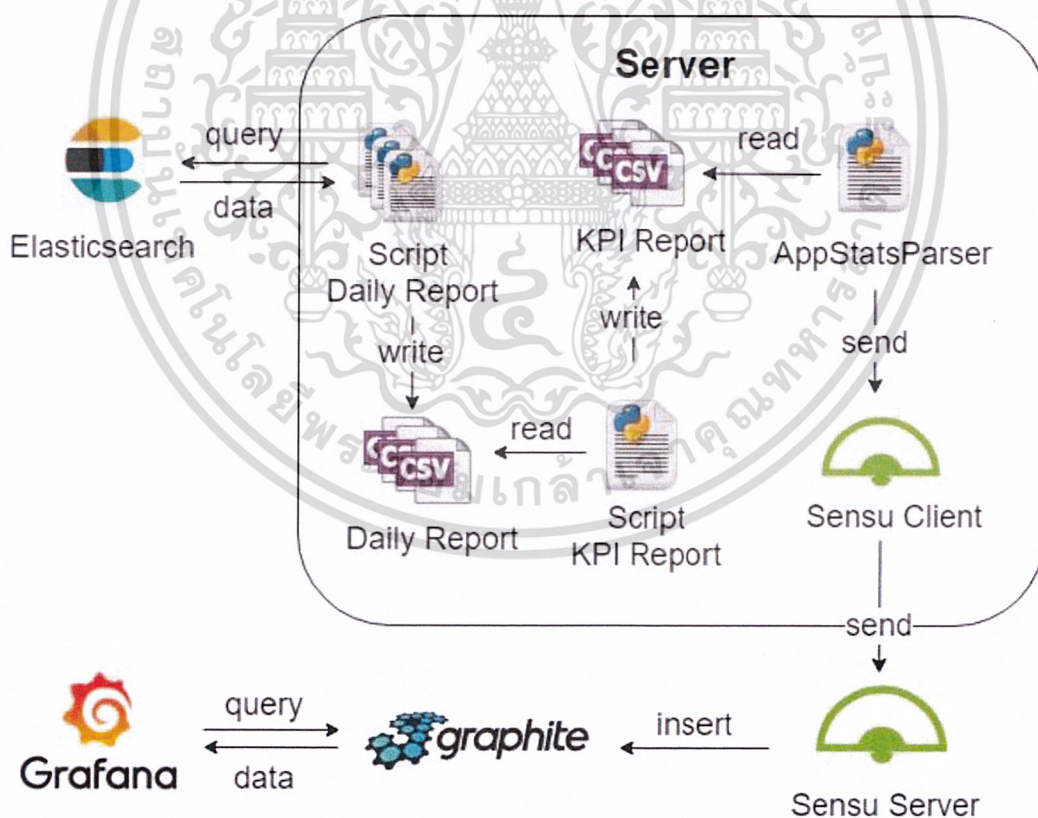
รูปที่ 3.15 การจัดเก็บ Access Log ของระบบ ETH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์

ในการวิเคราะห์ Access Log ของระบบ ETH นั้นนักวิเคราะห์ต้องการทราบว่าในแต่ละวันมีลูกค้าคนใดบ้างเข้ามาขอข้อมูลในระบบ โดยลูกค้าแต่ละคนใช้ Service อะไร มีปริมาณการเรียกใช้ข้อมูล (Request) เป็นจำนวนเท่าไร หากวันใดตรวจพบการขอข้อมูลที่ผิดปกตินักวิเคราะห์จะสามารถทราบได้ทันทีว่าลูกค้าคนใดเป็นคนขอข้อมูล ซึ่งขั้นตอนในการวิเคราะห์ประกอบไปด้วย Scripts จำนวน 2 ชุดได้แก่ Script Daily Report และ Script KPI Report ซึ่งตั้งเวลาการทำงานผ่านทาง Task Scheduler ให้ทำงานเป็นประจำทุกวัน

Script Daily Report เป็น Script ที่ใช้ในการค้นหา (query) Access Log จากฐานข้อมูล Elasticsearch แล้วสรุปข้อมูลเขียนเป็นรายงาน (Daily Report) ในรูปแบบไฟล์ CSV ในส่วนของ Script KPI Report เป็น Script สำหรับอ่าน Daily Report แล้วทำการเขียนเป็นกลุ่มของรายงาน (KPI Reports) ในรูปแบบไฟล์ CSV เมื่อข้อมูลในกลุ่มของ KPI Report เพิ่มขึ้น AppStatsParser จะทำการส่งข้อมูลในส่วนที่เพิ่มไปเก็บที่ฐานข้อมูล Graphite ผ่านทาง Sensu โดยนักวิเคราะห์สามารถใช้ Grafana ในการ query ข้อมูลในฐานข้อมูล Graphite ออกมาแสดงในรูปแบบของกราฟได้ ดังรูปที่ 3.16

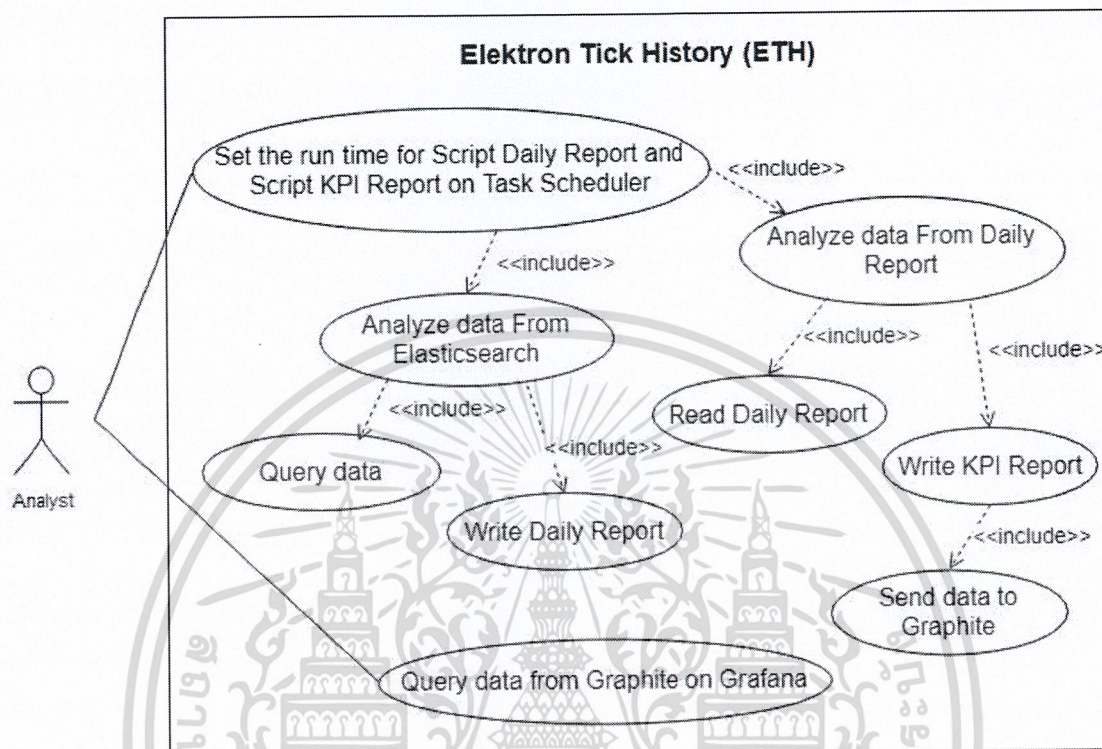


รูปที่ 3.16 การวิเคราะห์ Access Log ของระบบ ETH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงานของชุดคำสั่งจะอธิบายด้วยแผนภาพ (Diagram) เพื่อให้สามารถมองเห็นภาพรวมการทำงานของโปรแกรม ดังนี้

1) แผนภาพการทำงานของผู้ใช้ระบบ (Use Case Diagram)



รูปที่ 3.17 Use Case Diagram การวิเคราะห์ Access Log ของระบบ ETH

จากรูปที่ 3.17 สามารถอธิบายได้ดังนี้

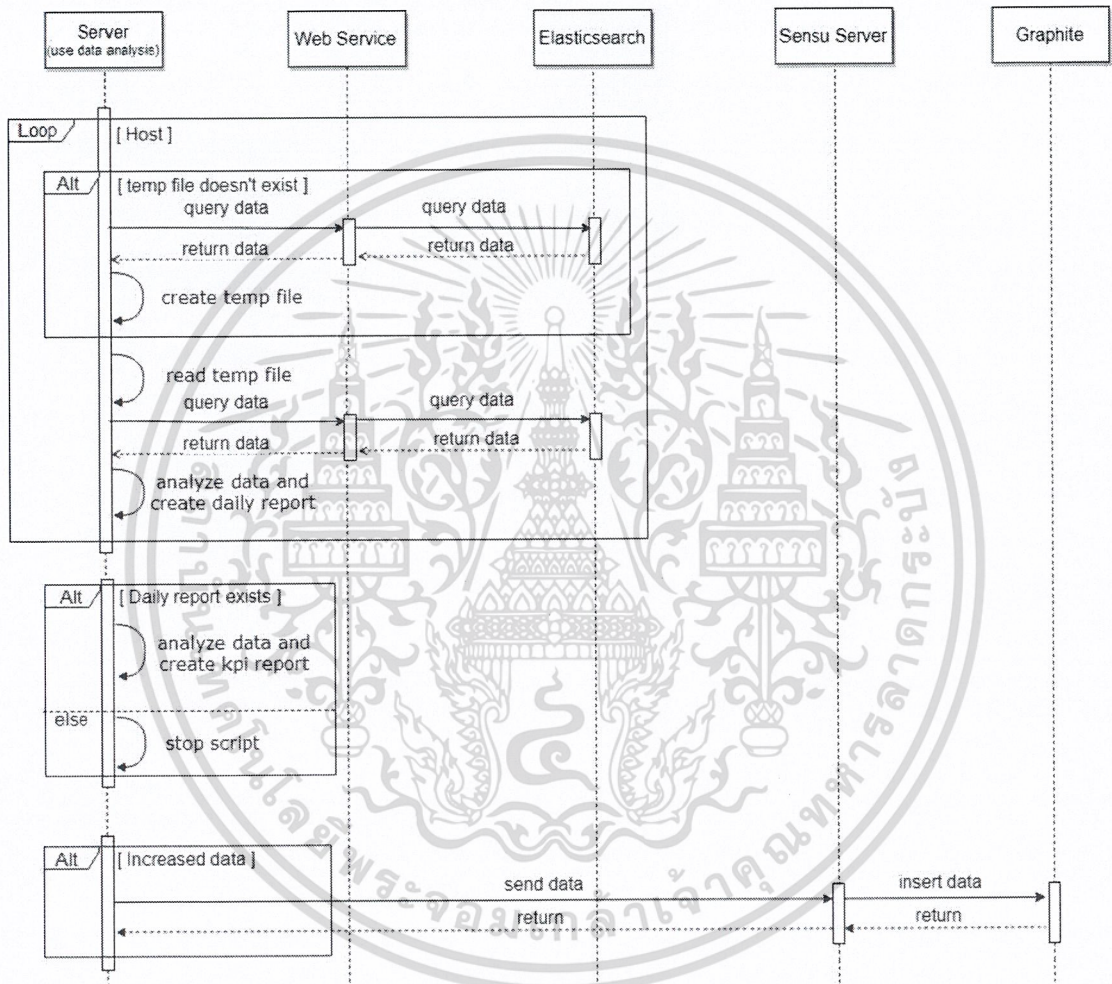
1. Set the run time for Script Daily Report and Script KPI Report on Task Scheduler คือ การตั้งเวลาการทำงานของ Script Daily Report และ Script KPI Report ผ่านทาง Task Scheduler โดยนักวิเคราะห์ ซึ่งทั้ง 2 Script จะทำงานเป็นประจำทุกวัน
2. Analyze data From Elasticsearch คือ การวิเคราะห์ข้อมูลจากการ query data จาก Elasticsearch แล้วเขียนเป็น Daily Report โดยจะทำงานก็ต่อเมื่อ Script Daily Report ทำงาน
3. Analyze data From Daily Report คือ การวิเคราะห์ข้อมูลจากการอ่าน Daily Report แล้วเขียนเป็น KPI Report ตามที่นักวิเคราะห์ต้องการ โดยจะทำงานก็ต่อเมื่อ Script KPI Report ทำงาน เมื่อมีการเขียน KPI Report จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่สู่สาธารณะโดยไม่ได้รับอนุญาต
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้ง Sensu Client เนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Query data from Graphite on Grafana คือ นักวิเคราะห์สามารถค้นหาข้อมูลที่เกิดขึ้นบนฐานข้อมูล Graphite แล้วนำมาแสดงเป็นกราฟต่างๆได้

2) แผนภาพความสัมพันธ์ระหว่างวัตถุ (Sequence Diagram)

เป็นการออกแบบขั้นตอนการทำงานของ Script ทั้ง 2 ชุด เพื่อแสดงถึงลำดับขั้นตอน เวลา และโปรแกรมที่เกี่ยวข้องในการทำงาน ดังรูปที่ 3.18



รูปที่ 3.18 Sequence Diagram การวิเคราะห์ Access Log ของระบบ ETH

จากรูปที่ 3.18 สามารถอธิบายได้ดังนี้

- เมื่อ Script Daily Report ทำงานจะทำการตรวจสอบว่ามี temp file อยู่หรือไม่ ซึ่ง temp file คือ Access Log ที่ไม่สามารถใช้คำสั่งพื้นฐานของฐานข้อมูล Elasticsearch ในการวิเคราะห์และคำนวณข้อมูลในเชิงสถิติได้ ดังนั้นจึงต้องดาวน์โหลด Access Log มาวิเคราะห์เอง หากตรวจสอบแล้วมี

เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้เฉพาะภายในระบบเอกสารของหน่วยงานนี้ ไม่สามารถนำไปใช้ประโยชน์อื่นใดได้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

temp file ตามที่ต้องการจะทำการอ่าน temp file ดังกล่าว จากนั้นส่งคำสั่งค้นหาข้อมูล (query data) ไปที่ฐานข้อมูล Elasticsearch ผ่าน Web Service เมื่อได้รับข้อมูลกลับมาแล้วจะทำการวิเคราะห์ข้อมูลร่วมกันและเขียนเป็นไฟล์ Daily Report

- เมื่อ Script KPI Report ทำงานจะตรวจสอบก่อนว่ามีไฟล์ Daily Report ตรงกับวันที่ต้องการหรือไม่ หากมีไฟล์จะทำการวิเคราะห์ข้อมูลและเขียนเป็นไฟล์ KPI Report ถ้าไม่มีไฟล์จะจบการทำงานของ Script
- เมื่อ AppStatsParser ทำงานจะตรวจสอบก่อนว่า KPI Report มีข้อมูลเพิ่มขึ้นหรือไม่ ถ้ามีจะส่งข้อมูลส่วนที่เพิ่มไปเก็บที่ฐานข้อมูล Graphite ผ่านทาง Sensu

3.3.3 ขั้นตอนการพัฒนาซอฟต์แวร์

จากขั้นตอนการศึกษาการทำงานของระบบ ETH ขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์ ทำให้ผู้พัฒนาทราบถึงขั้นตอนต่างๆในการพัฒนาชุดคำสั่งเพื่อใช้ในการวิเคราะห์ว่าในแต่ละวันมีลูกค้าคนใดบ้างเข้ามาขอข้อมูลในระบบ ลูกค้าแต่ละคนใช้ Service อะไร มีปริมาณการเรียกใช้ข้อมูล (Request) เป็นจำนวนเท่าไร ผู้พัฒนาได้ใช้เครื่องมือต่างๆในการพัฒนาชุดคำสั่งและอำนวยความสะดวกต่อนักวิเคราะห์ในการเข้าถึงข้อมูล ดังนี้

- โปรแกรม JetBrains PyCharm Community Edition เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมด้วยภาษา Python
- AppStatsParser และ Sensu ใช้ในการส่งข้อมูลที่ผ่านการวิเคราะห์แล้วไปเก็บไว้ในฐานข้อมูล Graphite
- Grafana ใช้ในการแสดงผลข้อมูลที่เก็บอยู่ในฐานข้อมูล Graphite ในรูปแบบของกราฟ

3.4 โปรแกรม Access Log Analysis From Elasticsearch

เป็นโปรแกรมส่วนติดต่อผู้ใช้ (Graphic User Interface) เพื่อใช้สำหรับดาวน์โหลดและวิเคราะห์ Access Log จากฐานข้อมูล Elasticsearch โดยผู้พัฒนาได้แบ่งขั้นตอนในการดำเนินงานออกเป็น 4 ขั้นตอน ดังต่อไปนี้

3.4.1 ขั้นตอนการศึกษาและรวบรวมข้อมูล

1) ศึกษาการใช้ฐานข้อมูล Elasticsearch ภายในองค์กร เพื่อให้เข้าใจถึงจุดประสงค์การใช้งาน โดยแบ่งลักษณะของการศึกษาได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับศึกษาลักษณะข้อมูลที่เก็บลงในฐานข้อมูล Elasticsearch ระยะเวลาในการดำเนินการคำนวณว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ศึกษาลักษณะการใช้งานของนักวิเคราะห์ในการวิเคราะห์ข้อมูล
- ศึกษาคำสั่งพื้นฐานที่ช่วยในการวิเคราะห์ข้อมูลในฐานข้อมูล

Elasticsearch

2) ศึกษาภาษาที่ใช้ในการพัฒนาโปรแกรม

- ศึกษาหลักการเขียนโปรแกรมด้วยภาษา Python ที่เกี่ยวข้องกับการวิเคราะห์ข้อมูล
- ศึกษาเครื่องมือต่างๆของภาษา Python ที่ใช้ในการเขียนส่วนติดต่อผู้ใช้งานและการติดต่อฐานข้อมูล Elasticsearch

3.4.2 ขั้นตอนการวิเคราะห์ข้อมูล

จากการศึกษาพบว่าบริษัททอมสันรอยเตอร์(ประเทศไทย)จำกัด ในส่วนของระบบที่ให้บริการข้อมูลทางด้านการเงินกับลูกค้า เช่น ระบบ ADC และระบบ ETH ได้มีการจัดเก็บ Access Log ไว้ในฐานข้อมูล Elasticsearch และในอนาคตนอกจาก 2 ระบบที่ได้กล่าวไปข้างต้นยังมีระบบอื่นที่จะปรับเปลี่ยนการเก็บ Access Log มาจัดเก็บไว้ในฐานข้อมูล Elasticsearch เหมือนกัน

โดยปกตินักวิเคราะห์จะมี Script สำหรับวิเคราะห์ Access Log ในลักษณะที่เป็นภาพรวมของแต่ละระบบเป็นประจำทุกวันโดยอัตโนมัติ แต่เนื่องด้วยบางครั้งนักวิเคราะห์ต้องการผลการวิเคราะห์ที่เฉพาะเจาะจง ทำให้ผลการวิเคราะห์ที่ได้จาก Script ไม่ตอบสนองต่อความต้องการของนักวิเคราะห์ เป็นผลให้นักวิเคราะห์ต้องค้นหาข้อมูลที่ต้องการผ่านทาง Kibana แต่เนื่องด้วยข้อจำกัดของ Kibana ที่ไม่สามารถจัดการข้อมูลที่มีปริมาณมากได้และไม่สามารถวิเคราะห์ข้อมูลที่มีความสัมพันธ์ของคอลัมภ์หลายๆคอลัมภ์ได้ จึงทำให้นักวิเคราะห์ต้องเสียเวลามากในการวิเคราะห์ข้อมูลในกรณีดังกล่าว

ผู้พัฒนาได้ศึกษาคำสั่งพื้นฐานของฐานข้อมูล Elasticsearch พบว่าสามารถใช้ในการวิเคราะห์ข้อมูลที่มีความสัมพันธ์ของคอลัมภ์หลายๆคอลัมภ์ได้ ดังนั้นผู้พัฒนาจึงได้พัฒนาโปรแกรมส่วนติดต่อกับผู้ใช้ เพื่อให้ให้นักวิเคราะห์สามารถค้นหาและวิเคราะห์ข้อมูลในฐานข้อมูล Elasticsearch ได้มีประสิทธิภาพมากยิ่งขึ้น โดยหน้าที่หลักในการทำงานของโปรแกรมมีดังต่อไปนี้

1) Functional Requirement

- มีหน้าจอส่วนติดต่อผู้ใช้ โดยผู้ใช้สามารถกำหนดข้อมูลต่างๆเองได้
- ผู้ใช้สามารถกำหนดค่า Name Username Password และ Host เพื่อใช้ในการติดต่อกับฐานข้อมูล Elasticsearch
- ผู้ใช้สามารถกำหนดเงื่อนไขที่บ่งบอกถึงแต่ละระบบที่ให้บริการข้อมูล
- โปรแกรมสามารถค้นหาและจดจำ Field ของแต่ละระบบที่ให้บริการ

ข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผู้ใช้สามารถกำหนด Field และเงื่อนไขในการคัดกรองข้อมูลได้
- ผู้ใช้สามารถเลือกที่จะดาวน์โหลดข้อมูลดิบ (Raw Data) มาเก็บอยู่ในรูปแบบไฟล์ CSV หรือ ไฟล์ JSON ได้
- ผู้ใช้สามารถเลือกที่จะวิเคราะห์ข้อมูลโดยใช้คำสั่งพื้นฐานของ Elasticsearch ในการคำนวณเชิงสถิติได้ โดยผลลัพธ์จะเก็บอยู่ในรูปแบบไฟล์ CSV
- ผู้ใช้สามารถกำหนดเส้นทาง (path) ที่ใช้สำหรับเก็บไฟล์ผลลัพธ์ต่างๆได้
- ผู้ใช้สามารถบันทึกและเปิดการตั้งค่าต่างๆในโปรแกรมได้
- โปรแกรมสามารถติดต่อ ค้นหา และวิเคราะห์ข้อมูลเชิงสถิติกับข้อมูลในฐานข้อมูล Elasticsearch ได้

2) Non-Functional Requirement

- ส่วนติดต่อผู้ใช้งานสามารถใช้งานได้ง่าย ไม่ซับซ้อน
- โปรแกรมต้องไม่ผิดพลาดอันเนื่องมาจากการการประมวลผล
- ผลลัพธ์ของการวิเคราะห์สามารถเข้าใจง่าย
- โปรแกรมใช้เวลาในการประมวลผลน้อยที่สุด

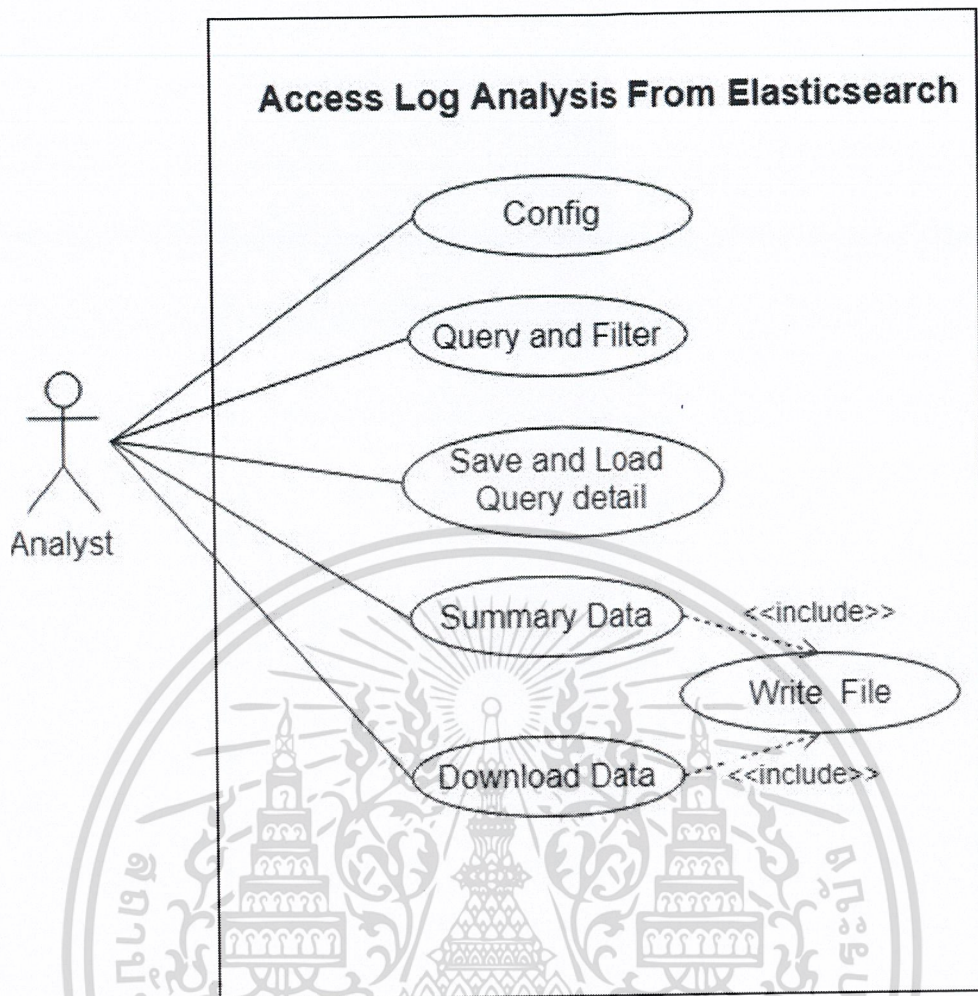
3.4.3 ขั้นตอนการออกแบบซอฟต์แวร์

เป็นขั้นตอนหลังจากผ่านการศึกษาค้นคว้าข้อมูลและวิเคราะห์ระบบเป็นที่เรียบร้อยแล้ว โดยขั้นตอนนี้จะเป็นการออกแบบลักษณะและขั้นตอนการทำงานของซอฟต์แวร์ เพื่อให้ผู้พัฒนาสามารถมองเห็นภาพรวมของระบบ ว่าแต่ละส่วนมีการทำงานอย่างไรบ้าง มีฟังก์ชันการทำงานใดบ้าง มีผู้ใดที่เกี่ยวข้องกับระบบบ้าง ทั้งนี้เพื่อให้การพัฒนาซอฟต์แวร์ดำเนินไปอย่างเป็นขั้นเป็นตอนและมีประสิทธิภาพมากที่สุด ผู้พัฒนาจะอธิบายด้วยแผนภาพ (Diagram) ดังนี้

1) แผนภาพการทำงานของผู้ใช้ระบบ (Use Case Diagram)

เป็นแผนภาพที่แสดงการทำงานของผู้ใช้ระบบว่ามีบุคคลใดบ้างที่มีปฏิสัมพันธ์กับระบบและแสดงความสัมพันธ์ของระบบย่อยภายในซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 Use Case Diagram ของโปรแกรม Access Log Analysis From Elasticsearch

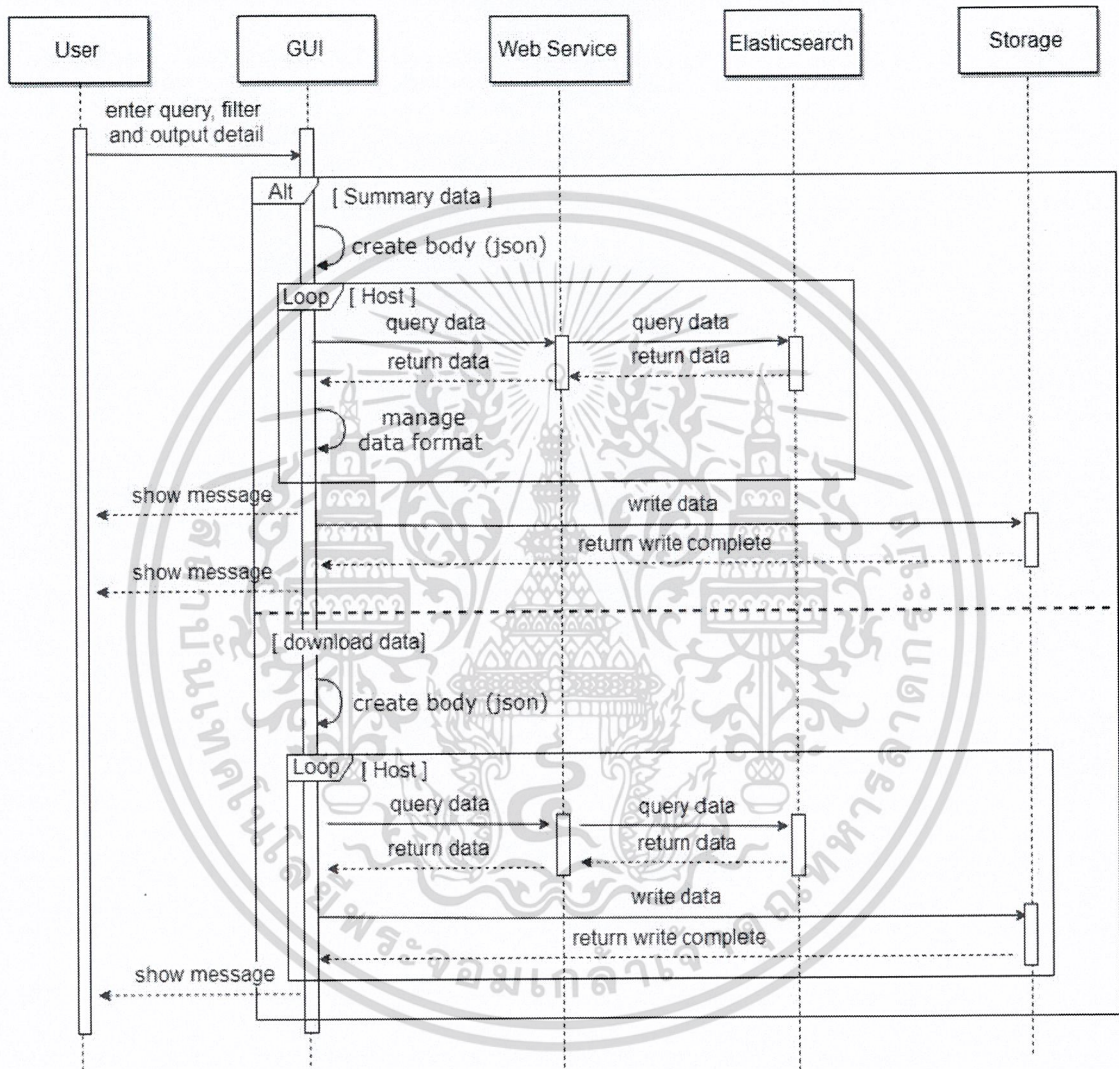
จากรูปที่ 3.19 สามารถอธิบายได้ดังนี้

1. Config คือ ส่วนที่ให้นักวิเคราะห์กำหนดค่า Name Username Password Host และเงื่อนไขที่บ่งบอกถึงระบบแต่ละระบบ
2. Query and Filter คือ ส่วนที่ให้นักวิเคราะห์กำหนดรายละเอียดต่างๆในการ query และ Filter ข้อมูล
3. Save and Load Query detail คือ ส่วนที่ให้นักวิเคราะห์บันทึกและเปิดการตั้งค่าต่างๆในโปรแกรม
4. Summary Data คือ ส่วนที่ให้นักวิเคราะห์กำหนดการคำนวณเชิงสถิติว่าจะคำนวณอะไรบ้าง พร้อมทั้งกำหนดเส้นทาง (path) ที่ใช้ในการเขียนไฟล์ผลลัพธ์ CSV
5. Download Data คือ ส่วนที่ให้นักวิเคราะห์กำหนดรายละเอียดที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ความปลอดภัยของข้อมูล
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) แผนภาพความสัมพันธ์ระหว่างวัตถุ (Sequence Diagram)

เป็นการออกแบบขั้นตอนการทำงานของโปรแกรม Access Log Analysis From Elasticsearch เพื่อแสดงถึงลำดับขั้นตอน เวลา และโปรแกรมที่เกี่ยวข้องในการทำงาน ดังรูปที่ 3.20



รูปที่ 3.20 Sequence Diagram ขั้นตอนการดาวน์โหลดและวิเคราะห์ Access Log จากฐานข้อมูล Elasticsearch ด้วยโปรแกรม Access Log Analysis From Elasticsearch

จากรูปที่ 3.20 สามารถอธิบายได้ดังนี้

- เมื่อนักวิเคราะห์กรอกรายละเอียดคัดกรองข้อมูลและเลือกการสรุปผล

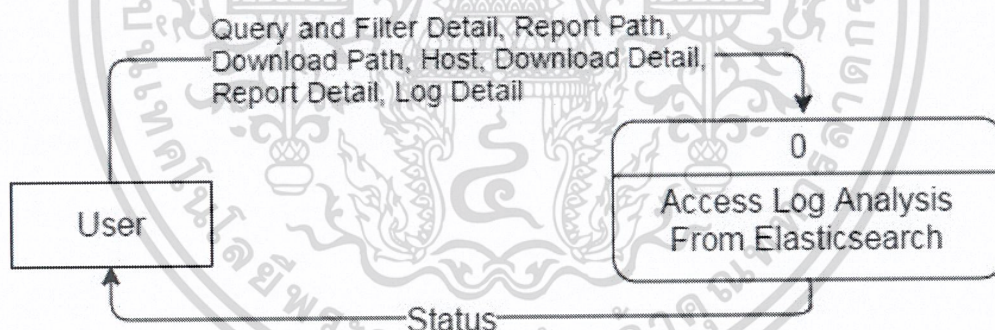
ข้อมูล โปรแกรมจะสร้างโครงสร้างข้อมูล JSON โดยเรียกว่า body ซึ่งเป็นเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อเก็บข้อมูล เช่น ข้อมูลเกี่ยวกับราคาค่าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีข้อมูล วิเคราะห์ข้อมูล และคำนวณข้อมูลเชิงสถิติ เมื่อ

body ถูกสร้างสำเร็จแล้ว โปรแกรมจะส่ง body ไปที่ฐานข้อมูล Elasticsearch ผ่านทาง Web Service จากนั้น Elasticsearch จะทำการวิเคราะห์ คำนวณ และรวบรวมข้อมูล เมื่อดำเนินการเสร็จจะส่งข้อมูลที่เป็นผลลัพธ์กลับมาที่โปรแกรม โปรแกรมจะทำการจัดรูปแบบของข้อมูลแล้วเขียนเป็นไฟล์ CSV เก็บไว้ที่ path ที่นักวิเคราะห์ระบุ โดยทุกขั้นตอนในการดำเนินการจะรายงานให้กับนักวิเคราะห์ทราบผ่านทางหน้าจอ

- ในกรณีที่นักวิเคราะห์เลือกที่จะดาวน์โหลดข้อมูลดิบ ขั้นตอนการทำงานจะเหมือนกับการสรุปผลข้อมูล แต่คำสั่งที่อยู่ใน body จะไม่เหมือนกันพร้อมทั้งรูปแบบของผลลัพธ์ที่ได้จากฐานข้อมูล Elasticsearch จะไม่เหมือนกัน เมื่อได้รับผลลัพธ์กลับมาแล้วจะเขียนเป็นไฟล์ CSV หรือ JSON ตามที่นักวิเคราะห์ระบุ

3) แผนผังบริบท (Context Diagram)

ในการพัฒนาโปรแกรม Access Log Analysis From Elasticsearch มีบุคคลที่เกี่ยวข้องกับระบบ ได้แก่ นักวิเคราะห์ระบบ โดยนักวิเคราะห์ระบบจะต้องกรอกรายละเอียดที่เกี่ยวข้องกับการคัดกรองข้อมูลและผลลัพธ์ที่ต้องการ ดังรูปที่ 3.21



รูปที่ 3.21 Context Diagram ของโปรแกรม Access Log Analysis From Elasticsearch

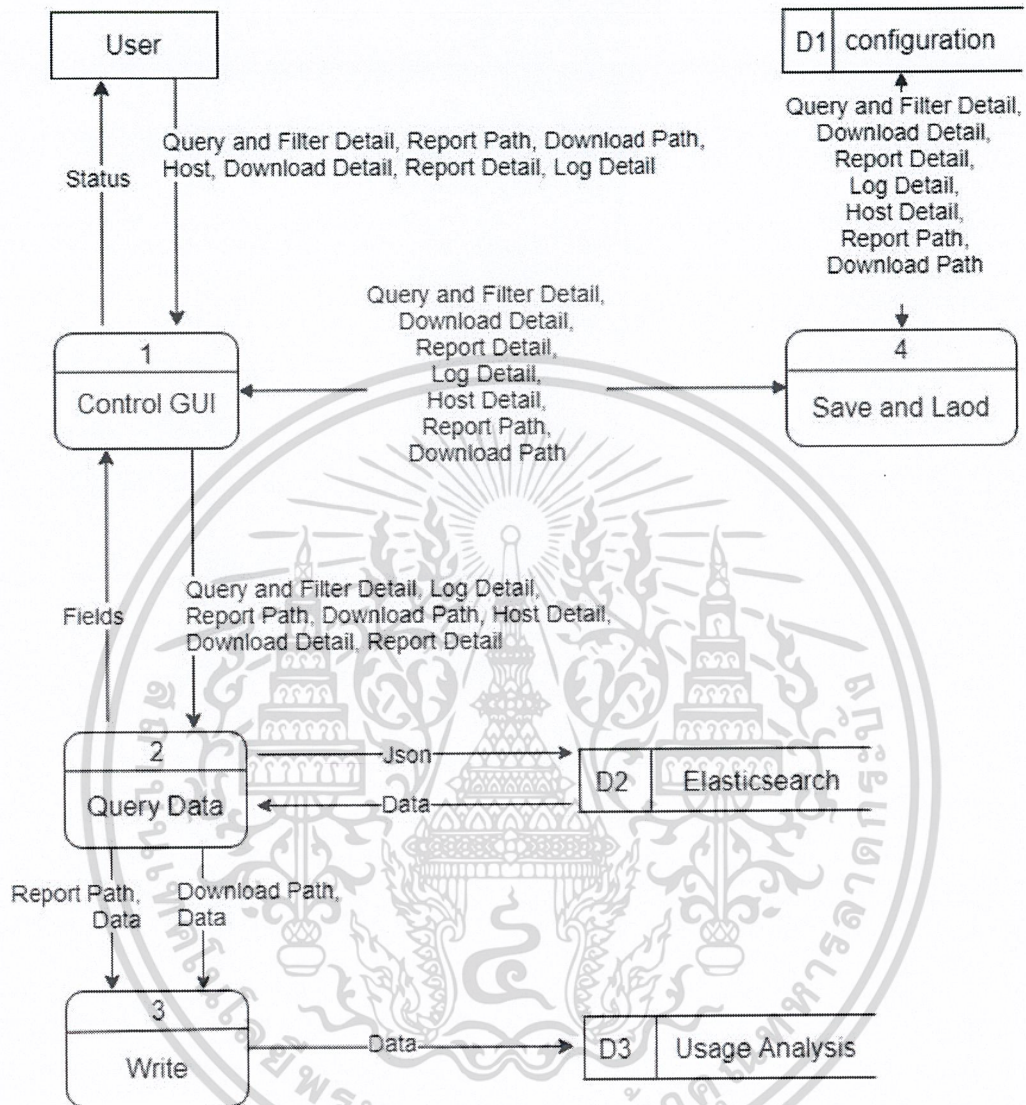
4) แผนภาพการไหลของข้อมูล (Data Flow Diagram: DFD level 0)

เป็นการนำเสนอแผนภาพกระแสข้อมูลที่แสดงให้เห็นกระบวนการทำงานที่มีรายละเอียดที่เพิ่มขึ้น และแสดงการไหลของกระแสข้อมูลว่ามาจากบุคคลใดหรืออุปกรณ์ใด เกิดจากกระบวนการใด ถูกส่งต่อไปกระบวนการใด และถูกนำไปจัดเก็บที่ใด เป็นต้น

แผนภาพการไหลของข้อมูล (Data Flow Diagram: DFD level 0) ของ

โปรแกรม Access Log Analysis From Elasticsearch มีกระบวนการทำงานหลัก 4 กระบวนการ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า คือ กระบวนการควบคุมการแสดงผลหน้าจอ (Control GUI) กระบวนการค้นหาข้อมูล (Query ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data) กระบวนการเขียนไฟล์ผลลัพธ์ (Write) และกระบวนการบันทึกและเปิดการตั้งค่าต่างๆของโปรแกรม (Save and Load) ดังรูปที่ 3.22



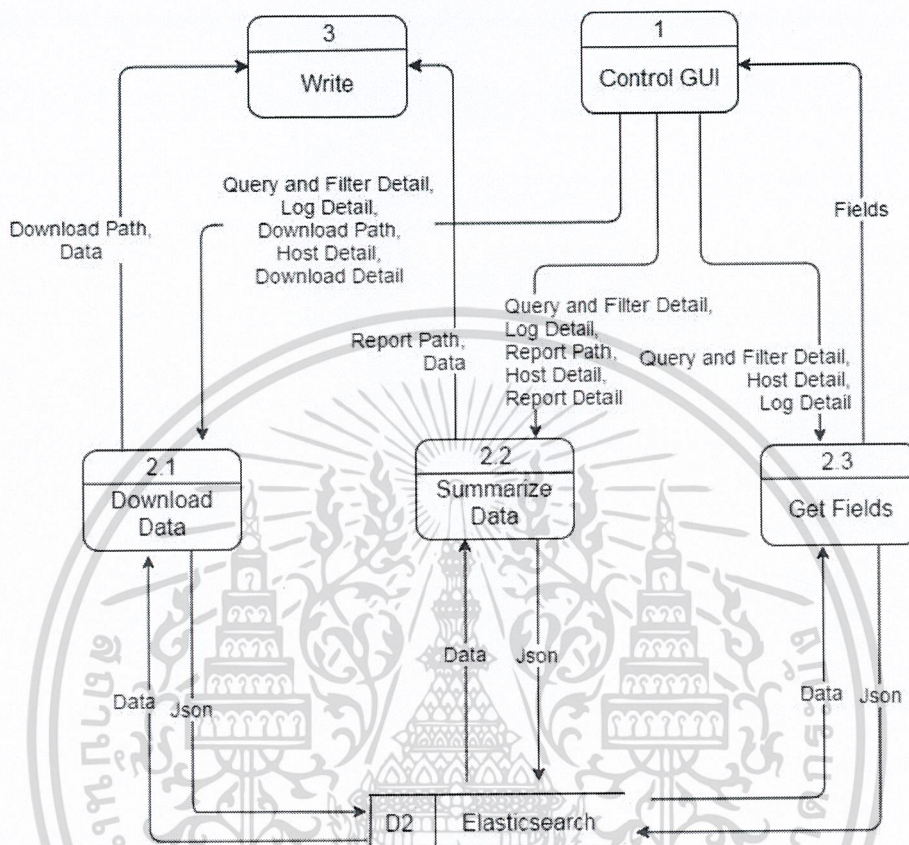
รูปที่ 3.22 Data Flow Diagram level 0 ของโปรแกรม Access Log Analysis From Elasticsearch

5) แผนภาพการไหลของข้อมูล (Data Flow Diagram: DFD level 1)

แผนภาพกระแสข้อมูล กระบวนการที่ 2 Query Data ของโปรแกรม Access Log Analysis From Elasticsearch มีกระบวนการย่อย 3 กระบวนการ ได้แก่ กระบวนการดาวน์โหลดข้อมูล (Download Data) จะทำงานเมื่อนักวิเคราะห์เลือกที่จะดาวน์โหลดข้อมูลดิบ กระบวนการสรุปผลข้อมูล (Summarize Data) จะทำงานเมื่อนักวิเคราะห์เลือกที่จะคำนวณและ

วิเคราะห์ข้อมูลเชิงสถิติโดยใช้คำสั่งพื้นฐานของฐานข้อมูล Elasticsearch และกระบวนการสุดท้ายคือ ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

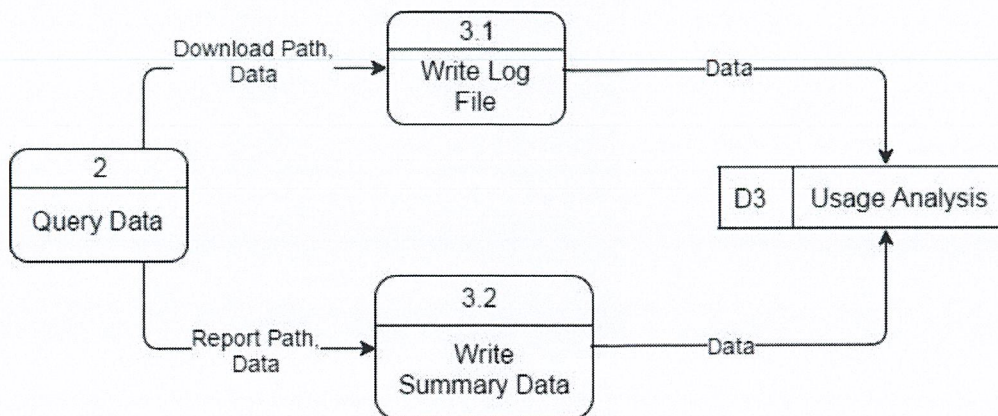
กระบวนการค้นหาฟิลด์ (Get Fields) จะทำงานเมื่อนักวิเคราะห์กำหนดเงื่อนไขเพื่อป้องกันตั้งแต่ระบบที่ให้บริการข้อมูล โดยทั้ง 3 กระบวนการย่อยจะมีการสร้างโครงสร้างข้อมูล JSON ซึ่งเป็นคำสั่งที่ใช้ในการควบคุม Elasticsearch ให้ทำตามคำสั่งที่ระบุ ดังรูปที่ 3.23



รูปที่ 3.23 Data Flow Diagram level 1 กระบวนการ Query Data ของโปรแกรม Access Log Analysis From Elasticsearch

แผนภาพกระแสข้อมูล กระบวนการที่ 3 Write ของโปรแกรม Access Log Analysis From Elasticsearch มีกระบวนการย่อย 2 กระบวนการคือ กระบวนการเขียนไฟล์ CSV หรือ JSON ที่ได้จากการดาวน์โหลดข้อมูลดิบมาจากฐานข้อมูล Elasticsearch (Write Log File) และกระบวนการเขียนไฟล์ CSV ที่ได้จากการวิเคราะห์และคำนวณข้อมูลเชิงสถิติ (Write Summary Data) โดยไฟล์ที่ได้จากกระบวนการทั้ง 2 กระบวนการจะถูกเก็บไว้ที่ path ตามที่นักวิเคราะห์ระบุ ดังรูปที่ 3.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 Data Flow Diagram level 1 กระบวนการ Write Data ของโปรแกรม Access Log Analysis From Elasticsearch

3.4.4 ขั้นตอนการพัฒนาซอฟต์แวร์

จากขั้นตอนการศึกษาและรวบรวมข้อมูล ขั้นตอนการวิเคราะห์ข้อมูล และขั้นตอนการออกแบบซอฟต์แวร์ ทำให้ผู้พัฒนาทราบถึงขั้นตอนต่างๆในการพัฒนาโปรแกรม Access Log Analysis From Elasticsearch ผู้พัฒนาได้ใช้เครื่องมือต่างๆในการพัฒนาโปรแกรม ดังนี้

- โปรแกรม JetBrains PyCharm Community Edition เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมด้วยภาษา Python
- Library PyQT เป็นเครื่องมือที่ช่วยในการเขียนส่วนติดต่อผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

ในบทนี้จะกล่าวถึงผลการพัฒนาชุดคำสั่งหรือโปรแกรมที่วิเคราะห์บันทึกการเรียกใช้ข้อมูลของลูกค้า (Access Log) ในระบบให้บริการข้อมูลทางการเงินของบริษัท โดยจะแบ่งผลการดำเนินงานออกเป็น 4 หัวข้อ ดังต่อไปนี้

- 4.1 ระบบ Analytics Data Cloud (ADC)
- 4.2 ระบบ Elektron Time Series (ETS)
- 4.3 ระบบ Elektron Tick History (ETH)
- 4.4 โปรแกรม Access Log Analysis From Elasticsearch

4.1 ระบบ Analytics Data Cloud (ADC)

จากปัญหาการใช้หน่วยความจำสำรอง (Harddisk) เป็นจำนวนมากและเวลาในการประมวลผลนาน ทำให้ชุดคำสั่งเดิมทำงานไม่สำเร็จ นักพัฒนาได้ปรับปรุงและพัฒนาขั้นตอนการวิเคราะห์ Access Log ขึ้นมาใหม่ โดยการเขียนชุดคำสั่งใหม่จำนวน 3 ชุด ได้แก่ Script Daily Report Script KPI Report และ Script Monthly Analysis ซึ่งช่วยลดการใช้หน่วยความจำสำรองและเวลาในการประมวลผล ดังตารางที่ 4.1

ตารางที่ 4.1 เปรียบเทียบการใช้หน่วยความจำสำรองและเวลาในการประมวลผลระหว่างชุดคำสั่งเดิมกับชุดคำสั่งใหม่

	Raw Data	Daily Report		Monthly Report	
	ชุดคำสั่งเดิม	ชุดคำสั่งเดิม	Script Daily Report	ชุดคำสั่งเดิม	Script Monthly Analysis
ขนาดหน่วยความจำสำรองที่ใช้	~ 2 GB	~ 23 KB	~ 23 KB	~ 4 MB	~ 13 MB
เวลาในการประมวลผล	~ 4 hr	~ 4 hr	~ 5 secs	~ 2 days	~ 10 min

จากตารางที่ 4.1 จะเห็นว่าเมื่อใช้ชุดคำสั่งใหม่ การใช้หน่วยความจำสำรองโดยรวมนั้นลดลง เนื่องจากไม่มีการดาวน์โหลดข้อมูลดิบ (Raw Data) มาเก็บไว้บนเครื่องเซิร์ฟเวอร์ ในส่วนของเวลาใน

การประมวลผลนั้นลดลงอย่างเห็นได้ชัด เนื่องจากในการวิเคราะห์ Access Log ให้ได้ Daily Report กับ Monthly Report โดยใช้ชุดคำสั่งเดิมนั้นจะต้องอ่านจาก Raw Data ซึ่งใช้เวลาเป็นจำนวนมาก แต่ Script Daily Report นั้นใช้คำสั่งพื้นฐานของฐานข้อมูล Elasticsearch ช่วยในการวิเคราะห์ ข้อมูลทำให้ใช้เวลาในการประมวลผลลดลง ส่วน Script Monthly Analysis นั้นเปลี่ยนจากอ่าน Raw Data มาเป็นอ่าน Daily Report แทน ทำให้ลดการอ่านข้อมูลที่ไม่จำเป็นส่งผลให้เวลาในการประมวลผลนั้นลดลง




จากการวิเคราะห์ Access Log โดย Script Daily Report Script KPI Report และ Script Monthly Analysis ทำให้ได้กลุ่มไฟล์ผลการวิเคราะห์ ดังต่อไปนี้

1) Daily Report แบ่งออกเป็น 2 กลุ่ม ได้แก่ กลุ่มผลิตภัณฑ์ (Product) ดังรูปที่ 4.1 และ กลุ่มลูกค้า (UUID) ดังรูปที่ 4.2 โดยในแต่ละกลุ่มจะวิเคราะห์จุดเวลา (Datapoints) และปริมาณการเรียกใช้ข้อมูล (Request) ตามสถานที่ที่ระบบติดตั้งอยู่ (Site)

Name	Date modified	Type	Size
ProdID_2017-06-04T00-00_2017-06-04T23...	7/13/2017 9:32 AM	Microsoft Office E...	14 KB
ProdID_2017-06-05T00-00_2017-06-05T23...	7/4/2017 10:24 AM	Microsoft Office E...	22 KB
ProdID_2017-06-06T00-00_2017-06-06T23...	7/3/2017 1:19 PM	Microsoft Office E...	24 KB
ProdID_2017-06-07T00-00_2017-06-07T23...	7/3/2017 1:22 PM	Microsoft Office E...	24 KB

From 2017-06-04T00-00 until 2017-06-04T23-59				
Date	Site	ProductID	Datapoints	Total
6/4/2017	amers	WEBCHECK:CPVIEWS	4318884	1079722
6/4/2017	amers	DE:UNITY	85141191	875513
6/4/2017	amers	ESTVIEWS:UNITY	99797730	66919
6/4/2017	amers	MSF_CORP:ETIV:CPVIEWS	7092622	41110
6/4/2017	amers	AGGHISTORICAL:PERSISTENTANALYTICS	37300710	26406
6/4/2017	amers	IMAP:CPVIEWS	25406097	24310
6/4/2017	amers	HISTCACHE:CPVIEWS	0	24306
6/4/2017	amers	MONITOR:CPVIEWS	17879096	21510
6/4/2017	amers	PRODUCTALARMS:CPVIEWS	15139796	18587
6/4/2017	amers	MSF_EKNAPPFUNDDetails:CPVIEWS	455893	18514
6/4/2017	amers	THINMONITOR:CPVIEWS	25040223	17298
6/4/2017	amers	HTMLCHART:CPVIEWS	220323	14993
6/4/2017	amers	STARMINEWACC:PERSISTENTANALYTICS	210156477	13592
6/4/2017	amers	TRACS_PREFETCH:CPVIEWS	23031346	11901
6/4/2017	amers	PA:CPVIEWS	614776	11657
6/4/2017	amers	RESEARCH_STARMINE:CPVIEW	13906436	10488
6/4/2017	amers	LIPPERAUTOMATION:CPVIEWS	0	9107
6/4/2017	amers	AGGCURRENT:PERSISTENTANALYTICS	8846237970	7753
6/4/2017	amers	ETIV:CPVIEWS	953013	6428
6/4/2017	amers	CP:CPVIEWS	125730	6100







เอกสารนี้เป็นเอกสารที่รูปที่ 4.1 ตัวอย่าง Daily Report กลุ่มผลิตภัณฑ์ ของระบบ ADC
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Name	Date modified	Type	Size
 UUID_2017-10-23T00-00_2017-10-23T23...	24/10/2560 12:30	Microsoft Excel Co...	1,291 KB
 UUID_2017-11-24T00-00_2017-11-24T23...	25/11/2560 12:00	Microsoft Excel Co...	989 KB
 UUID_2017-12-26T00-00_2017-12-26T23...	27/12/2560 12:00	Microsoft Excel Co...	508 KB

From 2017-10-23T00-00 until 2017-10-23T23-59				
Date	Site	UUID	Datapoints	Total
23/10/2017	emea	SL1-5IZYP86	916346	1656135
23/10/2017	emea	PADACT-002	87011306	978967
23/10/2017	emea	SL1-88V0Y05	516511	619716
23/10/2017	emea	SL1-88V0XY	449433	467441
23/10/2017	emea	SL1-7V112DW	153918	406148
23/10/2017	emea	SL1-7ZF6O3N	210158	394982
23/10/2017	emea	SL1-8JI7VWA	318289	175363
23/10/2017	emea	SL1-6PSSVKD	187732	164522
23/10/2017	emea	SL1-3LBC673	855561	152931
23/10/2017	emea	GEDTC-82081	125425	125451
23/10/2017	emea	SL1-6PT2XE6	581323	124974
23/10/2017	emea	SL1-80EM29J	104868	106176
23/10/2017	emea	SL1-5LM4KM6	666856	99859

รูปที่ 4.2 ตัวอย่าง Daily Report กลุ่มลูกค้า ของระบบ ADC

2) KPI Report แบ่งออกเป็น 6 กลุ่ม ได้แก่ กลุ่มประเภทผลิตภัณฑ์ที่คำนวณจุดเวลา (CapPoint) กลุ่มประเภทผลิตภัณฑ์ที่คำนวณปริมาณการเรียกใช้ข้อมูล (CapReq) กลุ่มผลิตภัณฑ์ที่คำนวณจุดเวลา (ProdIDPoint) กลุ่มผลิตภัณฑ์ที่คำนวณปริมาณการเรียกใช้ข้อมูล (ProdIDReq) กลุ่มคำนวณจุดเวลาทั้งหมด (TotalPoint) และกลุ่มคำนวณปริมาณการเรียกใช้ข้อมูลทั้งหมด (TotalReq) ดังรูปที่ 4.3 โดยบนสุดในแต่ละคอลัมภ์ของไฟล์แต่ละกลุ่มจะต้องมี matrix ในการอ้างอิงข้อมูลบนฐานข้อมูล Graphite ดังรูปที่ 4.4

Name	Date modified	Type
 CapPoint	22/2/2561 10:06	File folder
 CapReq	22/2/2561 10:06	File folder
 ProdIDPoint	22/2/2561 10:06	File folder
 ProdIDReq	22/2/2561 10:06	File folder
 TotalPoint	22/2/2561 10:06	File folder
 TotalReq	22/2/2561 10:06	File folder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ไปยังผู้อื่นและต้องอ้างอิงถึงชื่อเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 การแบ่งกลุ่ม KPI Report ของระบบ ADC

Timestamp	prodPoint.amers.AGGCURRENT-PERSISTENTANALYTICS.count
4/6/2017 0:00	8846237970
5/6/2017 0:00	8801624716
6/6/2017 0:00	8814093014
7/6/2017 0:00	8772355178
8/6/2017 0:00	8798473958
9/6/2017 0:00	8847215111
10/6/2017 0:00	8922706598
11/6/2017 0:00	8859887446
12/6/2017 0:00	8755349596
13/6/2017 0:00	8767762336
14/6/2017 0:00	8812554349
15/6/2017 0:00	8802876695
16/6/2017 0:00	8840910988

รูปที่ 4.4 ตัวอย่าง matrix ในไฟล์ KPI Report ของระบบ ADC

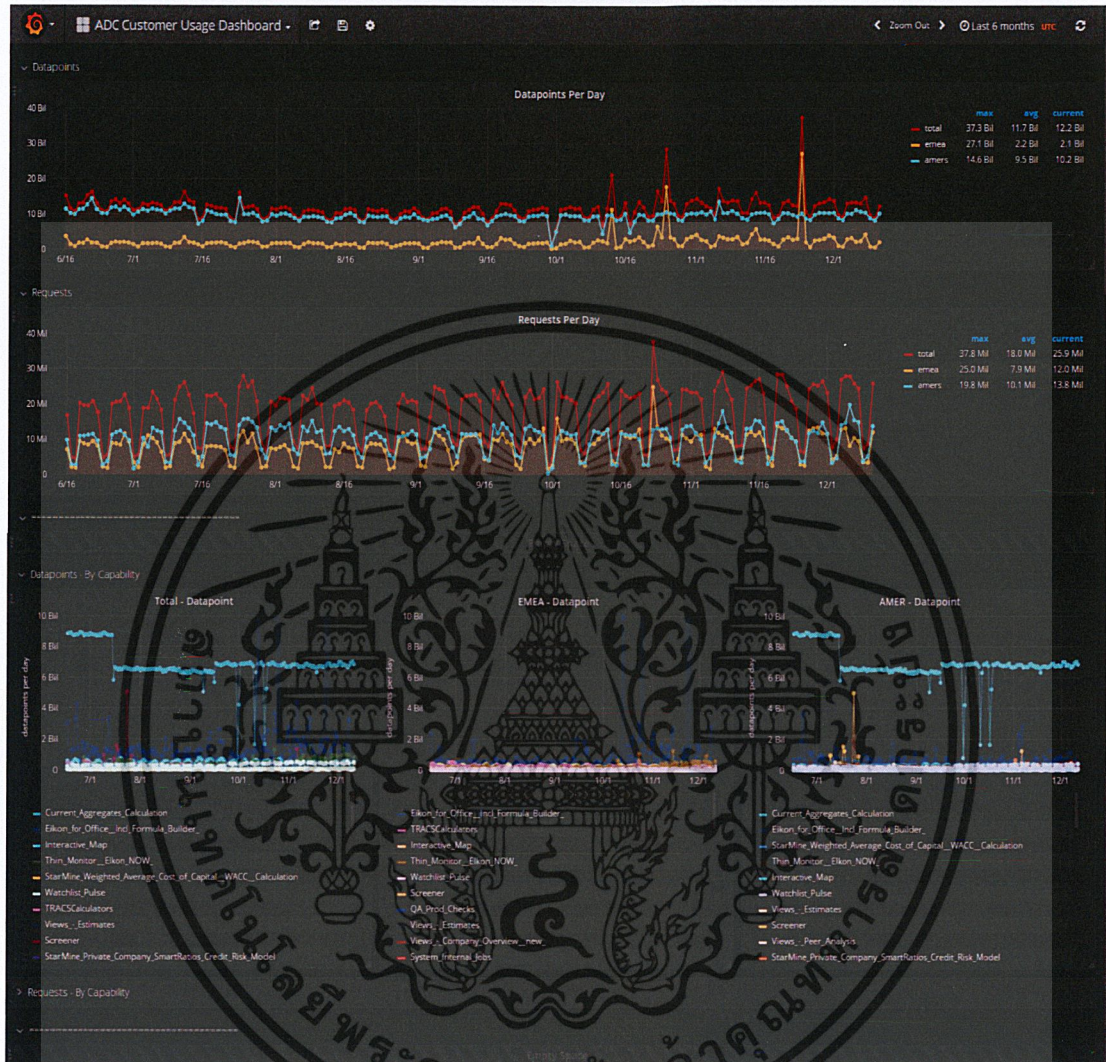
3) Monthly Report แบ่งออกเป็น 6 กลุ่ม ได้แก่ กลุ่มประเภทผลิตภัณฑ์ที่คำนวณจุดเวลา (Capability DataPoint) กลุ่มประเภทผลิตภัณฑ์ที่คำนวณปริมาณการเรียกใช้ข้อมูล (Capability Request) กลุ่มผลิตภัณฑ์ที่ยังไม่ถูกจัดประเภทโดยคำนวณจุดเวลา (Non-Group Datapoint) กลุ่มผลิตภัณฑ์ที่ยังไม่ถูกจัดประเภทโดยคำนวณปริมาณการเรียกใช้ข้อมูล (Non-Group Request) กลุ่มลูกค้าที่คำนวณจุดเวลา (UUID DataPoint) และกลุ่มลูกค้าที่คำนวณปริมาณการเรียกใช้ข้อมูล (UUID Request) โดยในแต่ละกลุ่มจะคำนวณค่าสูงสุด (Maximum) และค่าเฉลี่ย (Average) ของข้อมูลเป็นเวลา 2 เดือนเพื่อเปรียบเทียบการเปลี่ยนแปลงระหว่างเดือน ดังรูปที่ 4.5

Capability	Daily DataPoint	July-2017	August-2017	%Chg
API Garden	Average	290645.5238	1039793.435	257.7531218
API Garden	Maximum	464816	1883343	305.1803294
Aggregates Matrix App	Average	58028912.52	48172274.22	-16.98573673
Aggregates Matrix App	Maximum	101416745	77124240	-23.95314995
Answers	Average	1158741.238	2662917.261	129.8112101
Answers	Maximum	2479060	3109638	25.43617339
Backtesting	Average	39336.38095	39494.08696	0.400916404
Backtesting	Maximum	41752	40040	-4.100402376
Bond Liquidity Board	Average	70503.52381	56971.21739	-19.19380151
Bond Liquidity Board	Maximum	168826	146258	-13.36760925
Buyside Order Workflow	Average	31874.38095	20059.56522	-37.06680846
Buyside Order Workflow	Maximum	109889	142348	29.53798833
CIM	Average	575244	147772.5	-74.31133571
CIM	Maximum	1418939	154242	-89.12976527
Charts	Average	4713920.333	4211335.609	-10.66171444
Charts	Maximum	5653083	5978491	5.756292628
Company Codes	Average	67731.33333	80067.73913	18.21373534
Company Codes	Maximum	102008	110220	8.148380519

รูปที่ 4.5 ตัวอย่าง Monthly Report ของระบบ ADC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากกลุ่มไฟล์ผลการวิเคราะห์แล้วข้อมูลใน KPI Report ได้ถูกส่งไปเก็บที่ฐานข้อมูล Graphite เพื่อใช้เป็นแหล่งข้อมูลในการสร้าง Dashboard บน Grafana ซึ่งทำให้เห็นภาพรวมและแนวโน้มของข้อมูลง่ายขึ้น สามารถเห็นความผิดปกติของการเรียกใช้ข้อมูลของลูกค้า ดังรูปที่ 4.6



รูปที่ 4.6 ตัวอย่างการแสดงผลข้อมูลบน Grafana ของระบบ ADC

จากผลการดำเนินงานในการพัฒนาระบบการวิเคราะห์ Access Log ของระบบ ADC ได้มีการให้ผู้ใช้งานประเมินผล โดยผู้ประเมินผลคือ นายวรัทม ชัยเกษตรสิน ตำแหน่ง Capacity Team Manager ดังตารางที่ 4.2 โดยในช่องคะแนน 5 = พึงพอใจมากที่สุด, 4 = พึงพอใจมาก, 3 = ปานกลาง, 2 = น้อย และ 1 = น้อยที่สุดหรือไม่พอใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 การประเมินผลการพัฒนาระบบวิเคราะห์ Access Log ของระบบ ADC

ลำดับ	รายการการประเมิน	คะแนนการทำงาน รูปแบบเก่า	คะแนนการทำงาน รูปแบบใหม่
1	การตั้งค่าและสั่งให้โปรแกรมทำงาน สามารถทำได้โดยง่าย	3	5
2	โปรแกรมสามารถประมวลผลข้อมูลได้ อย่างถูกต้อง	4	5
3	ระยะเวลาที่โปรแกรมประมวลผลข้อมูล	1	5
4	ผลลัพธ์ Daily Report ตรงกับความ ต้องการของผู้ใช้	3	5
5	สามารถนำผลลัพธ์ Daily Report ไปใช้ งานต่อได้โดยง่าย	3	5
6	ผลลัพธ์ KPI Report ตรงกับ ความ ต้องการของผู้ใช้	3	5
7	สามารถนำผลลัพธ์ KPI Report ไป แสดงผลบน Grafana ได้โดยง่าย	3	5
8	ผลลัพธ์ Monthly Report ตรงกับความ ต้องการของผู้ใช้	2	5
9	สามารถนำผลลัพธ์ Monthly Report ไปใช้งานต่อได้โดยง่าย	3	5
10	ความสวยงามของผลลัพธ์ Monthly Report	3	5
11	ความคุ้มค่าของการใช้หน่วยความจำ สำรอง	1	5
	เฉลี่ย	2.64	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ระบบ Elektron Time Series (ETS)

จากการวิเคราะห์กลุ่มไฟล์ Access Log ไฟล์ Rule และกลุ่มไฟล์ Feed Log Analysis โดยใช้ Script Access Log Analysis และ Script Feed Log Analysis ในการวิเคราะห์ทำให้ได้ข้อมูลทั้งหมด 3 Collection ในฐานข้อมูล MongoDB ดังนี้

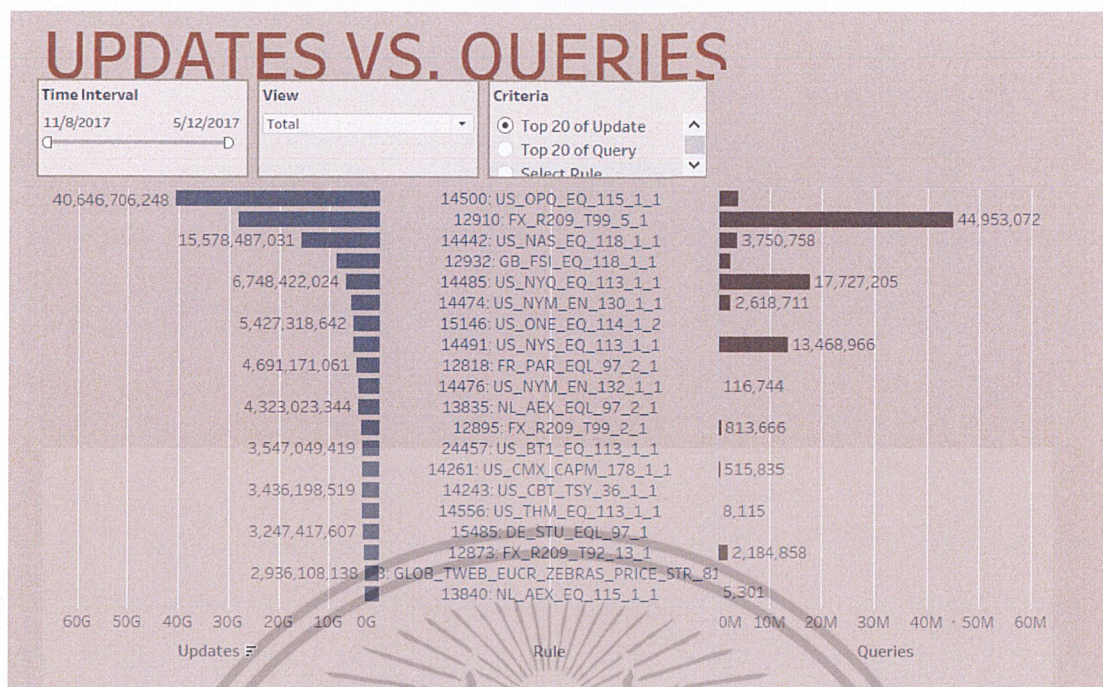
1) FLA_daily_Record เป็นที่จัดเก็บข้อมูลเกี่ยวกับกลุ่มของหุ้น (Rule) เพื่ออธิบายว่าในแต่ละ Rule มีปริมาณข้อมูลเท่าใดที่เข้ามาในระบบตามตัวชี้วัดและมีการเรียกใช้ข้อมูลของลูกค้ำตาม Rule เป็นปริมาณเท่าไรในแต่ละกลุ่มของการเรียกใช้งาน ดังรูปที่ 4.7

Key	Value	Type
(6) ObjectId("5a0a51e842ee365f8c643db0")	{ 31 fields }	Object
_id	ObjectId("5a0a51e842ee365f8c643db0")	ObjectId
1_MINUTE	18545.0	Double
TickTAS_PrePost_Predefined	0	Int32
TickTAS_Predefined	0	Int32
TickTAS_Full	0	Int32
CanCor	0	Int32
TAS	0	Int32
TAS_Raw	0	Int32
Scaling	0	Int32
RicRename	0	Int32
Date	2017-11-11 00:00:00.000Z	Date
RIC	0	Int32
TickTAS_PrePost_Full	0	Int32
TickTAS_PrePost	0	Int32
1_MINUTE_PrePost	12.0	Double
PARTIALBAR_CoraxAdjusted	0	Int32
TickTAS_PrePost_Predefined	0	Int32
Rule	12232	String
Context	0	Int32
PARTIALBAR	187.0	Double
30_MINUTE	66.0	Double
TickTAS_Predefined	1.0	Double
TASCORRECTION	1.0	Double
TickTAS_Full	0	Int32
OOC	0	Int32
FRImage	0	Int32
TAS_Raw	1.0	Double
TAS_Cooked	0	Int32
RuleName	AE_DBX_EQ_113_1.1	String
30_MINUTE_PrePost	23.0	Double
(7) ObjectId("5a0a51e842ee365f8c643db1")	{ 31 fields }	Object
(8) ObjectId("5a0a51e842ee365f8c643db2")	{ 31 fields }	Object
(9) ObjectId("5a0a51e842ee365f8c643db3")	{ 31 fields }	Object
(10) ObjectId("5a0a51e842ee365f8c643db4")	{ 31 fields }	Object
(11) ObjectId("5a0a51e842ee365f8c643db5")	{ 31 fields }	Object
(12) ObjectId("5a0a51e842ee365f8c643db6")	{ 31 fields }	Object
(13) ObjectId("5a0a51e842ee365f8c643db7")	{ 31 fields }	Object
(14) ObjectId("5a0a51e842ee365f8c643db8")	{ 31 fields }	Object

รูปที่ 4.7 ตัวอย่างข้อมูลใน FLA_daily_Record Collection บนฐานข้อมูล MongoDB ของระบบ ETS

ซึ่งข้อมูลที่อยู่ใน FLA_daily_Record Collection นั้นสามารถบอกได้ว่ากลุ่มของหุ้นในแต่ละกลุ่มมีการใช้เนื้อที่ในการจัดเก็บข้อมูลเป็นปริมาณเท่าใดโดยดูจากผลรวมปริมาณข้อมูลที่เข้ามาในระบบตามตัวชี้วัดของบริษัทและสามารถหาความคุ้มค่าของเนื้อที่ที่ใช้ในการเก็บข้อมูลในแต่ละกลุ่มของหุ้นได้โดยเปรียบเทียบกับปริมาณการเรียกใช้ข้อมูลของลูกค้ำ ดังรูปที่ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

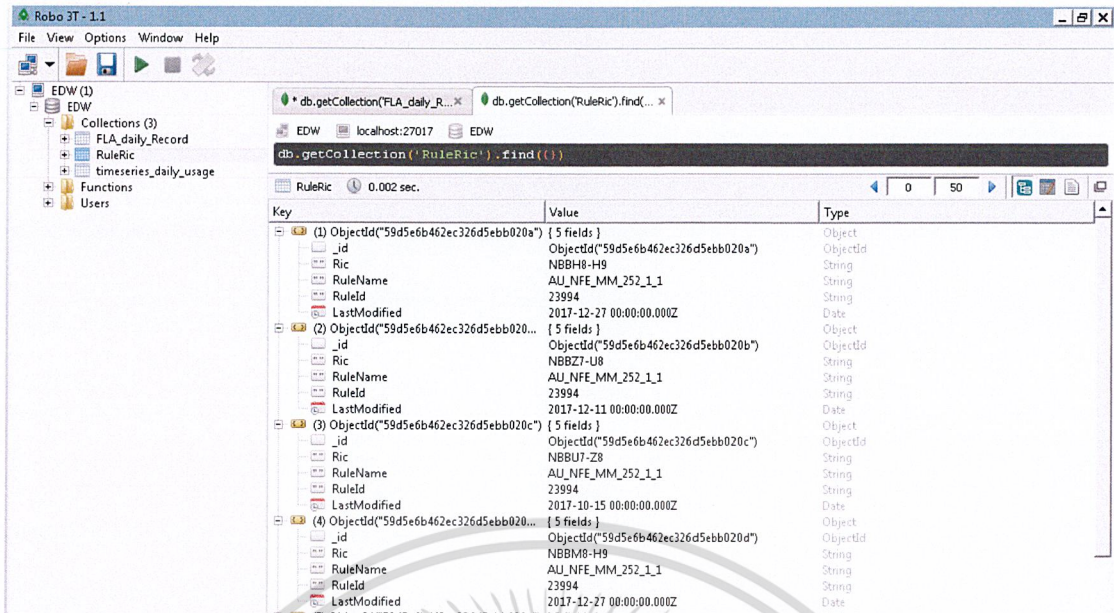


รูปที่ 4.8 ตัวอย่างแผนภูมิแท่งแสดงข้อมูลใน FLA_daily_Record Collection โดยใช้ Tableau ของระบบ ETS

จากรูปที่ 4.8 เป็นแผนภูมิที่แสดงข้อมูลตั้งแต่วันที่ 11/8/2017 ถึง 5/12/2017 โดยแผนภูมิแท่งทางด้านซ้ายแทนผลรวมปริมาณข้อมูลที่เข้ามาในระบบ แผนภูมิแท่งทางด้านขวาแทนปริมาณการเรียกใช้ข้อมูลของลูกค้า ให้พิจารณาที่ Rule 15146:US_ONE_EQ_114_1_2 จะมีผลรวมปริมาณข้อมูลที่เข้ามาในระบบคือ 5,427,318,642 แถว แต่ไม่มีการเรียกใช้ข้อมูลของลูกค้าเลย จากข้อมูลตรงนี้ทำให้นักวิเคราะห์สามารถรู้ได้ว่า Rule ใดมีการใช้เนื้อที่ในการเก็บข้อมูลไม่คุ้มค่าและสามารถนำข้อมูลตรงนี้ไปช่วยตัดสินใจในการบริหารพื้นที่เก็บข้อมูลได้

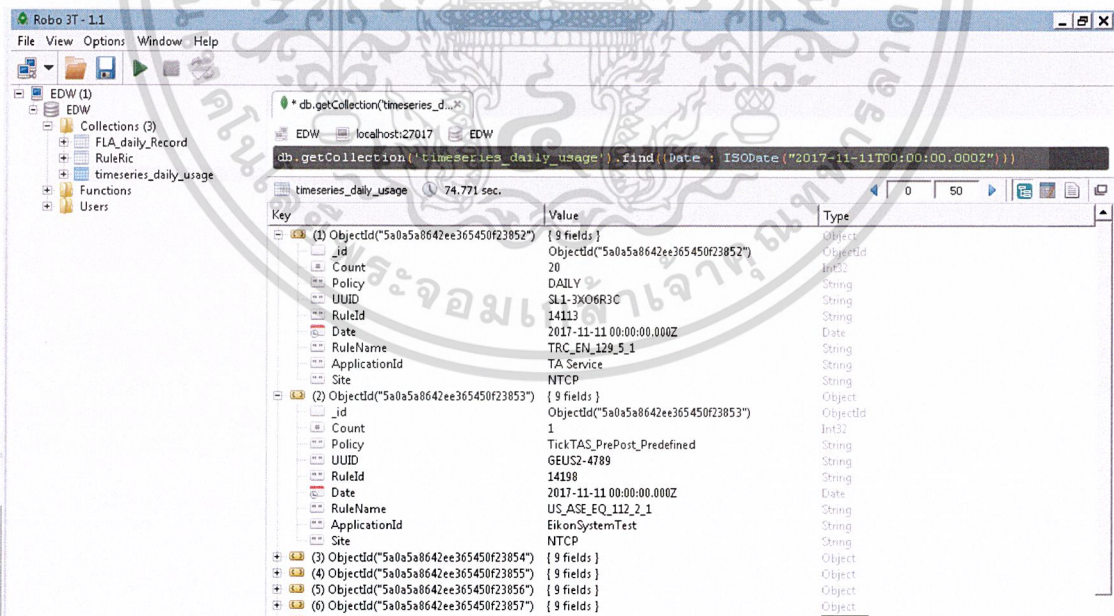
2) RuleRic เป็นที่จัดเก็บข้อมูลเกี่ยวกับกลุ่มของหุ้น (Rule) และหุ้น (Ric) ซึ่งนักวิเคราะห์จะสามารถรู้ได้ว่าในแต่ละ Rule มี Ric อะไรบ้างและในแต่ละ Ric มีการใช้งานล่าสุดวันใด ดังรูปที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 ตัวอย่างข้อมูลใน RuleRic Collection บนฐานข้อมูล MongoDB ของระบบ ETS

3) timeseries_daily_usage เป็นที่จัดเก็บข้อมูลเกี่ยวกับการเรียกใช้ข้อมูลของลูกค้า โดยจะบอกถึงลูกค้าคือใคร (UUID) ใช้แอปพลิเคชันอะไร (ApplicationId) รูปแบบการติดต่อคืออะไร (Policy) ปริมาณการติดต่อเป็นจำนวนเท่าไร (Count) ดังรูปที่ 4.10



รูปที่ 4.10 ตัวอย่างข้อมูลใน timeseries_daily_usage Collection บนฐานข้อมูล MongoDB

ของระบบ ETS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการดำเนินงานในการพัฒนาระบบการวิเคราะห์ข้อมูลของระบบ ETS ได้มีการให้
ผู้ใช้งานประเมินผล โดยผู้ประเมินผลคือ นางสาวณิกานต์ พัฒนาวณิชย์สกุล ตำแหน่ง Associate
Capacity Engineer ดังตารางที่ 4.3 โดยในช่องคะแนน 5 = พึงพอใจมากที่สุด, 4 = พึงพอใจมาก,
3 = ปานกลาง, 2 = น้อย และ 1 = น้อยที่สุดหรือไม่พอใจ

ตารางที่ 4.3 การประเมินผลการพัฒนาระบบวิเคราะห์ข้อมูลของระบบ ETS

ลำดับ	รายการการประเมิน	คะแนน
1	การสั่งให้โปรแกรมทำงานสามารถทำได้โดยง่าย	4
2	โปรแกรมสามารถประมวลผลข้อมูลได้อย่างถูกต้อง	5
3	โปรแกรมสามารถจัดการกับข้อผิดพลาดที่เกิดขึ้นได้	4
4	ระยะเวลาโดยรวมที่โปรแกรมประมวลผลข้อมูล	4
5	ผลลัพธ์ที่ได้ตรงกับความต้องการของผู้ใช้	5
6	ผู้ใช้นำผลลัพธ์ไปแสดงผลบน Tableau ได้โดยง่าย	5
7	ผลลัพธ์ที่ได้สามารถตอบคำถามทางธุรกิจได้	5
	เฉลี่ย	4.57

4.3 ระบบ Elektron Tick History (ETH)

จากการวิเคราะห์ Access Log โดย Script Daily Report และ Script KPI Report ทำให้
ได้กลุ่มไฟล์ผลการวิเคราะห์ ดังต่อไปนี้

1) Daily Report แบ่งออกเป็น 2 กลุ่ม ได้แก่ กลุ่มเซอร์วิส ดังรูปที่ 4.11 และกลุ่มข้อมูลหุ้น
ดังรูปที่ 4.12 โดยกลุ่มเซอร์วิสจะบอกรายละเอียดเกี่ยวกับการเรียกใช้ข้อมูลของลูกค้า ได้แก่ ไอดี
ลูกค้า (User ID) ไอดีประเภทลูกค้า (Client ID) เซอร์วิสที่ใช้ (Service Name) ปริมาณการเรียกใช้
ข้อมูลของลูกค้า (Count Request) ปริมาณคำสั่ง (Count Returned Record) ปริมาณชุดคำสั่ง
(Count Item) ในส่วนของกลุ่มข้อมูลหุ้นจะบอกรายละเอียดเกี่ยวกับปริมาณการขอข้อมูลในแต่ละหุ้น
ของลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Site	Service Name	Client ID	User ID	View	Count Request	Count Returned Record	Count Item
HDC	SampleService	DSS	9004812	raw	3	3	0
HDC	SampleService	GoldmanSachs	xyz	normalised	2	11000	0
HDC	SampleService	ETH	E2EServiceCheckTest	raw	192	3840	0
HDC	SampleService	DSS	9010445	raw	3	300	0
HDC	SampleService	test	test	raw	1	1000	0
PLN	SampleService	DSS	9012033	raw	2	200	0
PLN	SampleService	ETH	E2EServiceCheckTest	raw	188	3760	0
PLN	SampleService	GoldmanSachs	xyz	raw	4	10300	0
HDC	SampleService	DSS	9012033	raw	3	300	0
PLN	SampleService	GoldmanSachs	xyz	normalised	1	10000	0
PLN	SampleService	DSS	9010592	raw	1	100	0
PLN	SampleService	DSS	9001038	raw	1	100	0
PLN	SampleService	DSS	9001038	rawLL2	2	200	0
HDC	SampleService	GoldmanSachs	xyz	raw	6	11400	0

รูปที่ 4.11 ตัวอย่าง Daily Report กลุ่มเซอร์วิสของระบบ ETH

Date	Site	Ric	Count Item
10/10/2017	HDC	IT4966401=TWEB	2
10/10/2017	HDC	MCUV21	14
10/10/2017	HDC	WTCLM4	14
10/10/2017	HDC	JP00350068=TWEB	2
10/10/2017	HDC	TFSZ7	1
10/10/2017	HDC	912828D23=TWEB	2
10/10/2017	HDC	ZTF.L	1
10/10/2017	HDC	LCOV1	27
10/10/2017	HDC	LCOV0	27
10/10/2017	HDC	LCOV3	1
10/10/2017	HDC	LCOV2	1
10/10/2017	HDC	AU0000184=TWEB	2
10/10/2017	HDC	APAM.AS	1
10/10/2017	HDC	ES00122T3=TWEB	2
10/10/2017	HDC	LCOV9	29
10/10/2017	HDC	LCOV8	29
10/10/2017	HDC	XUIT.L	1
10/10/2017	HDC	MCUV24	13
10/10/2017	HDC	EJP3.AS	1
10/10/2017	HDC	DE000A0GNPZ3=AS	1

รูปที่ 4.12 ตัวอย่าง Daily Report กลุ่มข้อมูลหุ้นของระบบ ETH

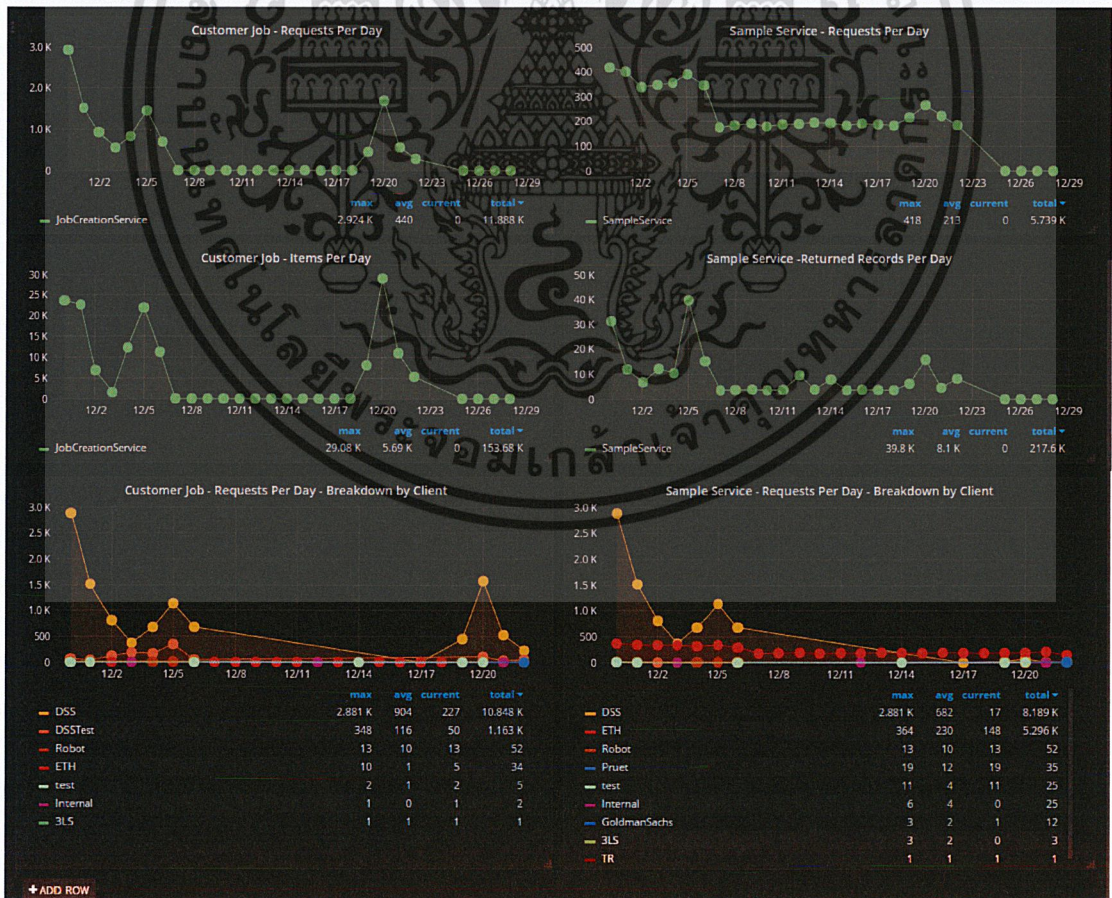
2) KPI Report จะบอกรายละเอียดเกี่ยวกับการเรียกใช้ข้อมูลของลูกค้า ได้แก่ ไอทีประเภทลูกค้า (Client ID) เซอร์วิสที่ใช้ (Service Name) ปริมาณการเรียกใช้ข้อมูลของลูกค้า (Count Request) ปริมาณคำสั่ง (Count Returned Record) ปริมาณชุดคำสั่ง (Count Item) โดยรูปแบบในรายงานด้านบนสุดของคอลัมภ์จะต้องมี matrix ในการอ้างอิงข้อมูลบนฐานข้อมูล Graphite ดังรูปที่ 4.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timestamp	HDC.DSSTest.SampleServi	HDC.DSSTest.SampleService	HDC.DSSTest.JobCreationServ	HDC.DSSTest.JobCreationSei
1/10/2017 0:00	13	0	13	22
2/10/2017 0:00	5	0	5	41
9/10/2017 0:00	23	0	23	32
10/10/2017 0:00	46	0	46	535
11/10/2017 0:00	54	0	54	680
12/10/2017 0:00	49	0	49	865
13/10/2017 0:00	41	0	41	383
14/10/2017 0:00	21	0	21	25
15/10/2017 0:00	20	0	20	30
16/10/2017 0:00	47	0	47	129
17/10/2017 0:00	69	0	69	957
18/10/2017 0:00	144	0	144	947
19/10/2017 0:00	77	0	77	519
20/10/2017 0:00	50	0	50	397
21/10/2017 0:00	46	0	46	420
22/10/2017 0:00	43	0	43	58
23/10/2017 0:00	47	0	47	96
24/10/2017 0:00	39	0	39	157

รูปที่ 4.13 ตัวอย่าง KPI Report ของระบบ ETH

นอกจากกลุ่มไฟล์ผลการวิเคราะห์แล้วข้อมูลใน KPI Report ได้ถูกส่งไปเก็บที่ฐานข้อมูล Graphite เพื่อใช้เป็นแหล่งข้อมูลในการสร้าง Dashboard บน Grafana ซึ่งทำให้เห็นภาพรวมและแนวโน้มของข้อมูลง่ายขึ้น สามารถเห็นความผิดปกติของการเรียกใช้ข้อมูลของลูกค้า ดังรูปที่ 4.14



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 รูปที่ 4.14 ตัวอย่างการแสดงผลข้อมูลบน Grafana ของระบบ ETH
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น

จากผลการดำเนินงานในการพัฒนาระบบการวิเคราะห์ Access Log ของระบบ ETH ได้มีการให้ผู้ใช้งานประเมินผล โดยผู้ประเมินผลคือ นายวรัทม์ ชัยเกษตรสิน ตำแหน่ง Capacity Team Manager ดังตารางที่ 4.4 โดยในช่องคะแนน 5 = พึงพอใจมากที่สุด, 4 = พึงพอใจมาก, 3 = ปานกลาง, 2 = น้อย และ 1 = น้อยที่สุดหรือไม่พอใจ

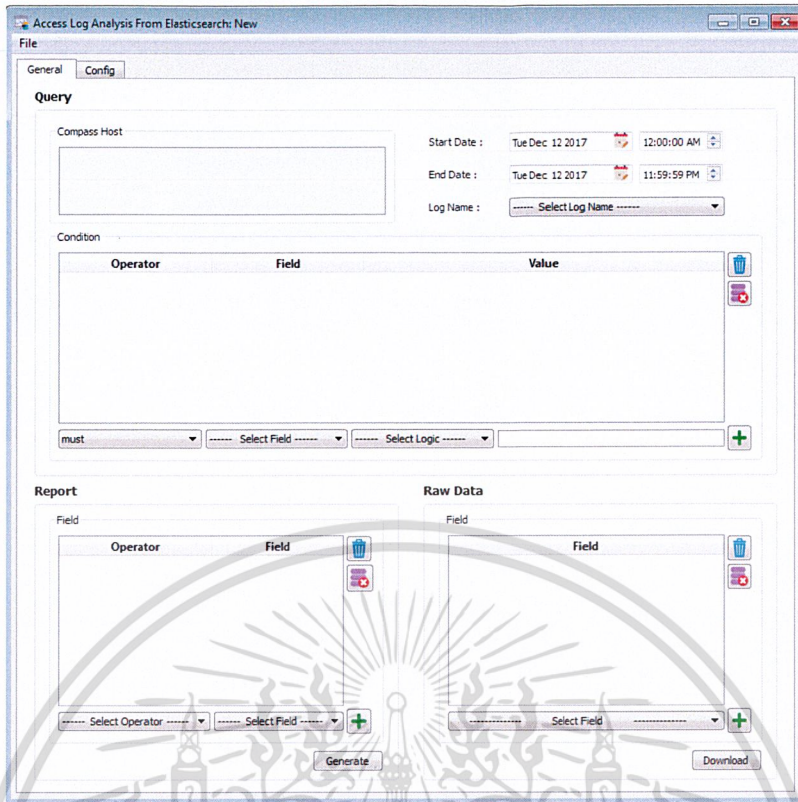
ตารางที่ 4.4 การประเมินผลการพัฒนาระบบวิเคราะห์ Access Log ของระบบ ETH

ลำดับ	รายการการประเมิน	คะแนน
1	การตั้งค่าและสั่งให้โปรแกรมทำงานสามารถทำได้โดยง่าย	5
2	โปรแกรมสามารถประมวลผลข้อมูลได้อย่างถูกต้อง	5
3	ระยะเวลาที่โปรแกรมประมวลผลข้อมูล	5
4	ผลลัพธ์ Daily Report ตรงกับความต้องการของผู้ใช้	5
5	สามารถนำผลลัพธ์ Daily Report ไปใช้งานต่อได้โดยง่าย	5
6	ผลลัพธ์ KPI Report ตรงกับความต้องการของผู้ใช้	5
7	สามารถนำผลลัพธ์ KPI Report ไปแสดงผลบน Grafana ได้โดยง่าย	5
	เฉลี่ย	5

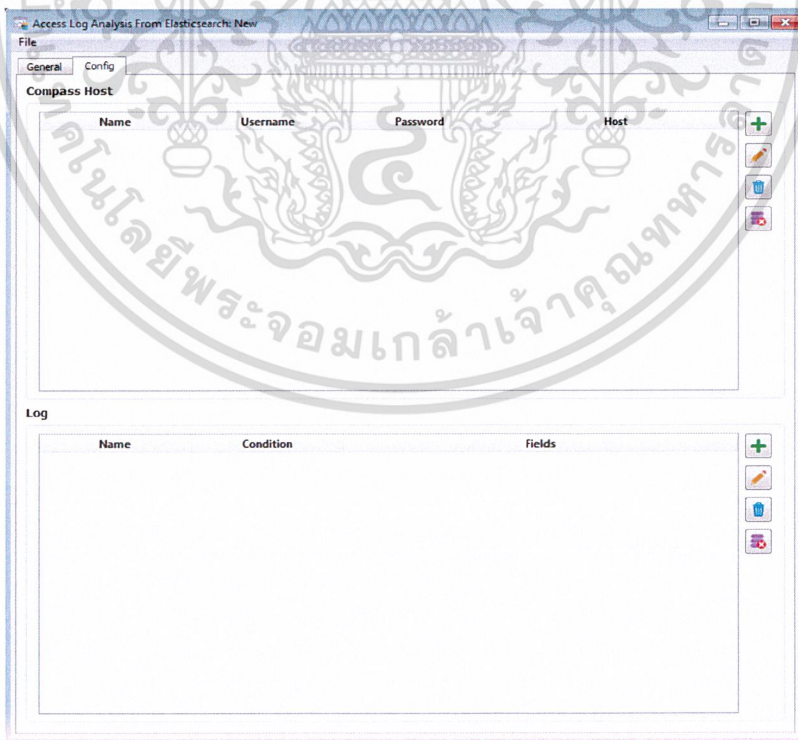
4.4 โปรแกรม Access Log Analysis From Elasticsearch

โปรแกรมส่วนติดต่อผู้ใช้ (Graphic User Interface) ใช้สำหรับวิเคราะห์ Access Log จากฐานข้อมูล Elasticsearch ในระบบบริการข้อมูลทางการเงินของบริษัททอมสัน รอยเตอร์ (ประเทศไทย) จำกัด โดยภายในโปรแกรมจะมี Tab 'General' ดังรูปที่ 4.15 และ Tab 'Config' ดังรูปที่ 4.16 โดย Tab 'General' จะเป็นส่วนในการกำหนดเงื่อนไขต่างๆในการวิเคราะห์ Access Log ได้แก่ สถานที่ที่ฐานข้อมูล Elasticsearch ติดตั้งอยู่ (Compass Host) ช่วงเวลาที่ต้องการวิเคราะห์ (Start Date และ End Date) ระบบที่ต้องการวิเคราะห์ (Log Name) เงื่อนไขในการกรองข้อมูล (Condition) และลักษณะของผลลัพธ์ (Report และ Raw Data) ในส่วนของ Tab 'Config' จะเป็นการตั้งค่าพื้นฐานต่างๆ ได้แก่ Username Password Host ของฐานข้อมูล Elasticsearch และเงื่อนไขที่บ่งบอกถึงระบบให้บริการข้อมูลทางการเงินในแต่ละระบบ โดยก่อนที่นักวิเคราะห์จะสร้างรายงานหรือดาวน์โหลด Raw Data จะต้องเพิ่มรายละเอียดในส่วนของ Compass Host และส่วนของ Log ใน Tab 'Config' ให้เรียบร้อยก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 Tab General ของโปรแกรม Access Log Analysis From Elasticsearch



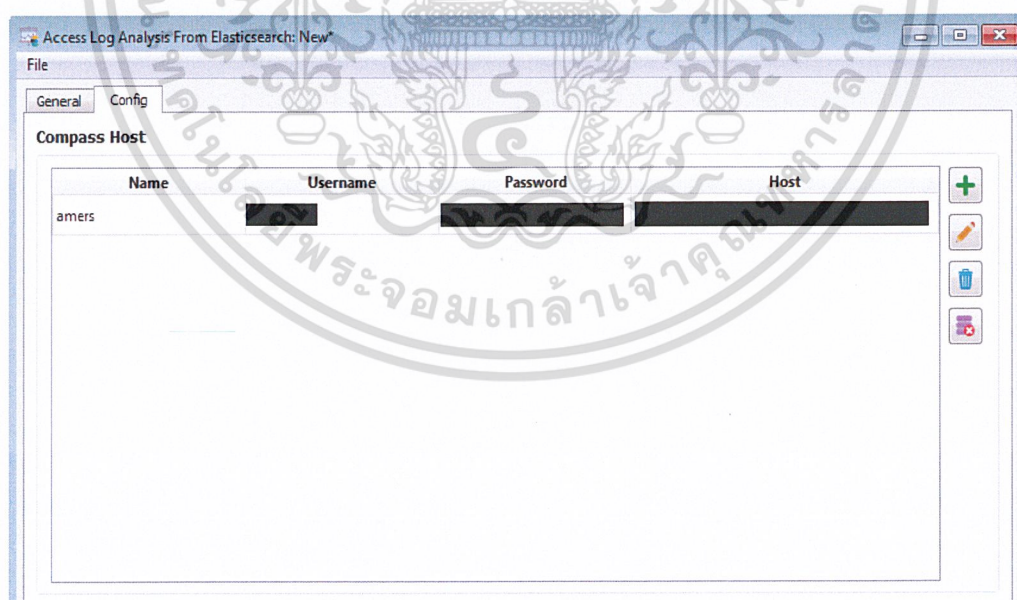
เอกสารนี้เป็นลิขสิทธิ์ของสำนักงานส่งเสริมการศึกษานอกระบบฯ สำนักงาน ก.ป.ค. ไม่สามารถนำข้อมูลไปเผยแพร่โดยไม่ได้รับอนุญาต
รูปที่ 4.16 Tab Config ของโปรแกรม Access Log Analysis From Elasticsearch
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.1 การกำหนดรายละเอียดที่ Tab 'Config'

1) การ Add Compass Host

ให้คลิกที่ปุ่ม + ด้านข้างของตาราง Compass Host จะมีหน้าต่าง Add Compass Host ปรากฏขึ้นมาดังรูปที่ 4.17 จากนั้นใส่ Name, Username, Password และ Host ให้เรียบร้อย และเมื่อคลิก OK จะเป็นการ add ข้อมูลเข้าสู่ตาราง ดังรูปที่ 4.18

รูปที่ 4.17 หน้าต่าง Add Compass Host ของโปรแกรม Access Log Analysis From Elasticsearch



รูปที่ 4.18 ตัวอย่าง Compass Host ที่เพิ่มในตาราง Compass Host ของโปรแกรม Access Log Analysis From Elasticsearch

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การ Add Log

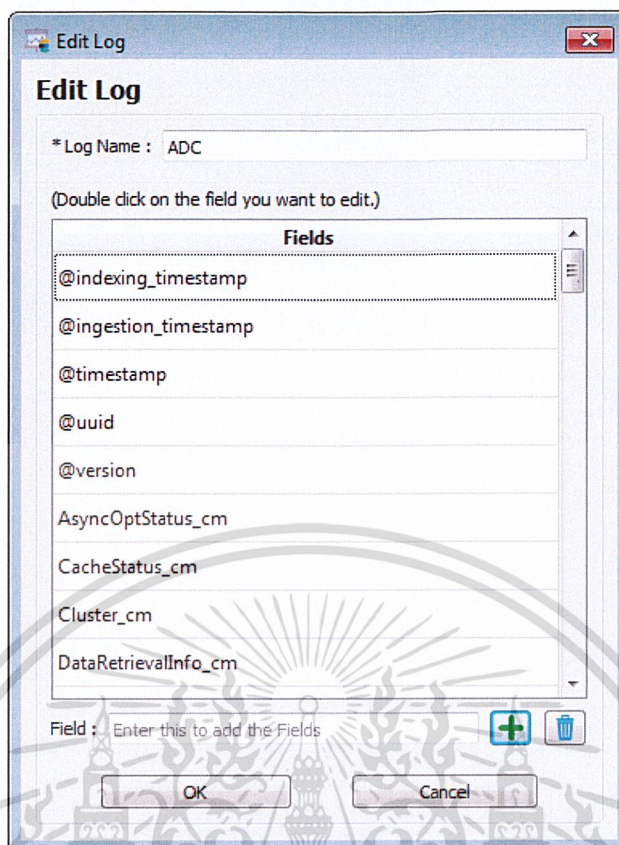
ให้คลิกที่ปุ่ม + ด้านข้างของตาราง Log จะมีหน้าต่าง Add Log ปรากฏขึ้นมา จากนั้นใส่ Log Name, เลือก Compass Host และ add เงื่อนไข (Condition) ดังรูปที่ 4.19

Operator	Field	Value
must	type	json

รูปที่ 4.19 ตัวอย่างการ Add Log ของโปรแกรม Access Log Analysis From Elasticsearch

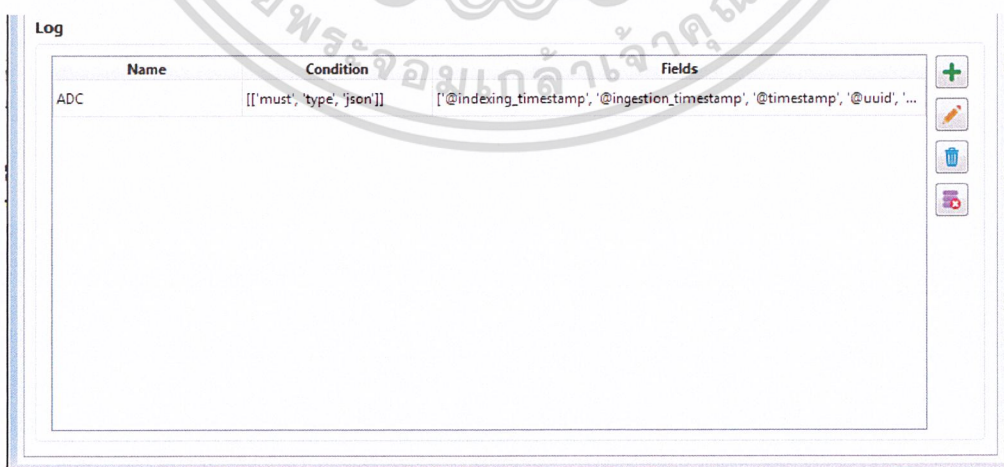
เมื่อกำหนดรายละเอียดของ Log เรียบร้อยแล้วหลังจากนั้นกดที่ปุ่ม Add โปรแกรมจะค้นหา Field ที่ตรงกับเงื่อนไขและแสดงออกมาที่หน้าต่าง Edit Log ดังรูปที่ 4.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 หน้าต่าง Edit Log ของโปรแกรม Access Log Analysis From Elasticsearch

หน้าต่าง Edit Log ผู้ใช้สามารถที่จะเปลี่ยน Log Name, ลบ Field ที่ไม่ต้องการหรือเพิ่ม Field ที่ต้องการได้ เมื่อกดปุ่ม OK จะเป็นการเพิ่ม Log ลงใน ตาราง Log ดังรูปที่ 4.21



เอกสารนี้เป็นรูปที่ 4.21 ตัวอย่าง Log ที่เพิ่มในตาราง Log ของโปรแกรม Access Log Analysis ด้านการคำ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาเอกสารนี้ไปถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
From Elasticsearch

ผู้ใช้สามารถแก้ไข Log Name, เพิ่มหรือลบ Field ได้ในภายหลัง โดยเลือก Log ที่ต้องการ จากนั้นกดปุ่ม Edit ที่ด้านข้างตาราง

3) การลบข้อมูลที่เลือกและลบข้อมูลทั้งหมดในตาราง

ด้านข้างของตารางในโปรแกรม Access Log Analysis From Elasticsearch มีปุ่ม Delete และปุ่ม Clear ผู้ใช้สามารถเลือกรายการที่ต้องการ จากนั้นกดปุ่ม Delete หรือหากต้องการลบทั้งหมดให้กดปุ่ม Clear

4.4.2 การกำหนดรายละเอียดที่ Tab 'General'

1) การกำหนดเงื่อนไขในส่วนของ query

ในส่วนของ query ที่ต้องกำหนดเงื่อนไข ได้แก่ เลือก Compass Host, กำหนด Start Date, กำหนด End Date, เลือก Log Name และเพิ่ม Condition ในตาราง ดังรูป 4.22 การเพิ่ม Condition ผู้ใช้จะต้องเลือก Log Name ก่อนเสมอ เพื่อเป็นการกำหนด Field ให้กับส่วนของการเลือก Field

ในการกำหนดรายละเอียดในส่วนนี้ จะเป็นเงื่อนไขในการกรองข้อมูล โดยข้อมูลที่ตรงกับ Condition จะถูกนำไปใช้ในส่วนของการสร้างรายงานหรือส่วนของการดาวน์โหลด Raw Data ต่อไป

Operator	Field	Value
must	ProdId_cm	text: DE:UNITY

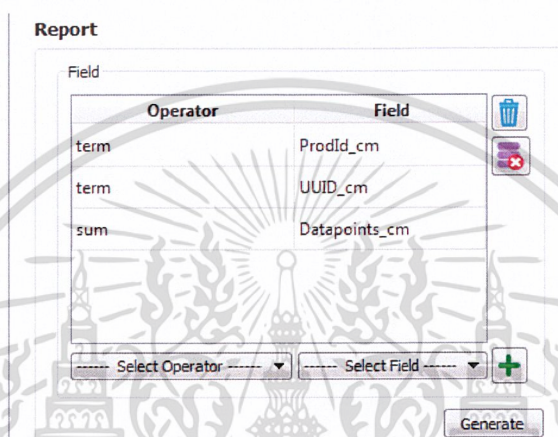
รูปที่ 4.22 ตัวอย่างการกำหนดเงื่อนไขในส่วนของ query ใน Tab 'General'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

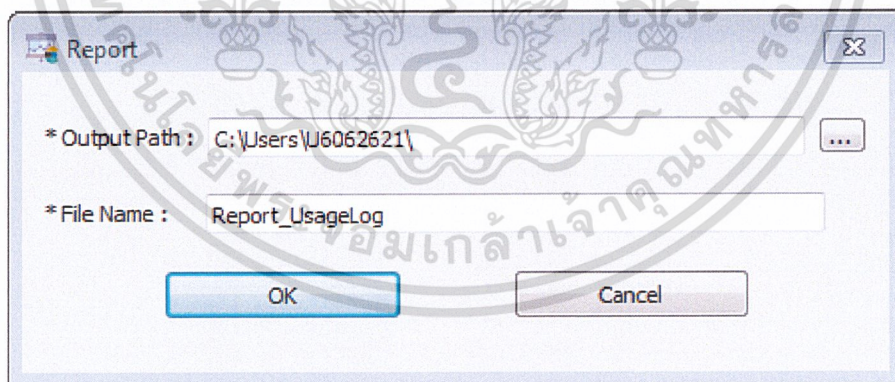
2) การสร้างรายงาน (Report)

ในการสร้างรายงานจะมีการเลือก 2 ส่วนได้แก่ ส่วนที่ 1 Operator คือ ชุดคำสั่งทางสถิติ ประกอบไปด้วย avg (Average), cardinality, date histogram, extended Stats, histogram, max, min, missing, stats, sum และ term ส่วนที่ 2 รายการของ Field ต่างๆ

เมื่อเพิ่ม Operator และ Field ลงในตาราง Report ตามที่ต้องการแล้ว ดังรูปที่ 4.23 ให้กดปุ่ม Generate จะปรากฏหน้าต่างให้เลือก Path และตั้งชื่อไฟล์ ดังรูปที่ 2.24



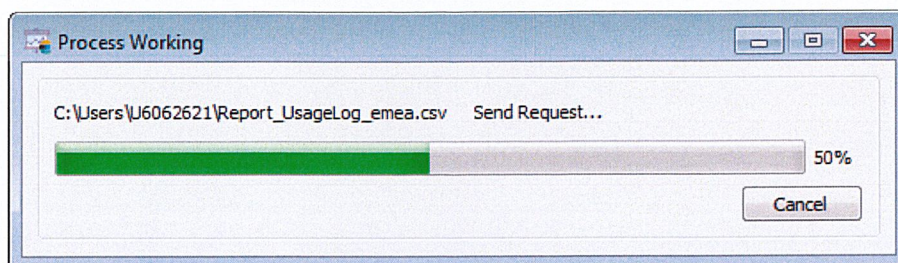
รูปที่ 4.23 ตัวอย่างการเพิ่มข้อมูลลงในตาราง Report



รูปที่ 4.24 หน้าต่างการเลือก Path และตั้งชื่อไฟล์ของการสร้างรายงาน

เมื่อเลือก Path และ ตั้งชื่อไฟล์เรียบร้อยแล้วให้กดปุ่ม OK จะปรากฏหน้าต่าง Process Working เพื่อแสดงสถานะการทำงาน ดังรูปที่ 4.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



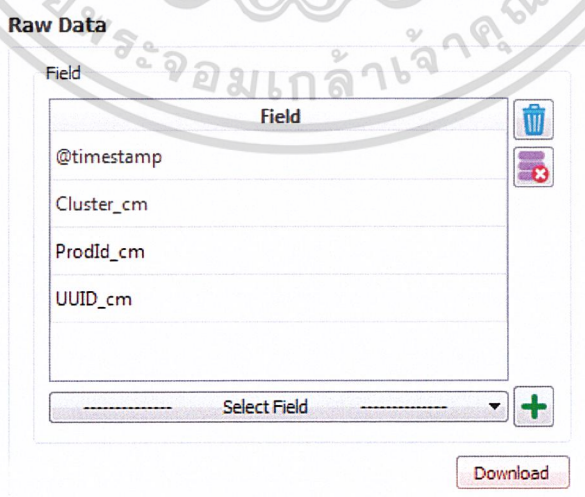
รูปที่ 4.25 หน้าต่าง Process Working เพื่อแสดงสถานะการทำงานของการสร้าง Report

	A	B	C	D
1	terms_ProdId_cm	terms_UUID_cm	sum_Datapoints_cm	count
2	DE:UNITY	R12-1143663	1338480	458137
3	DE:UNITY	SL1-45YBFKW	13455	320400
4	DE:UNITY	SL1-8331J5X	2806479	279678
5	DE:UNITY	SL1-6XBPEPT	1765873	214680
6	DE:UNITY	GESG1-17726	77632	191463
7	DE:UNITY	GENTC-17050	116643	182708
8	DE:UNITY	SL1-7AE3RKW	143267	150355
9	DE:UNITY	SL1-3LBC673	809563	148282

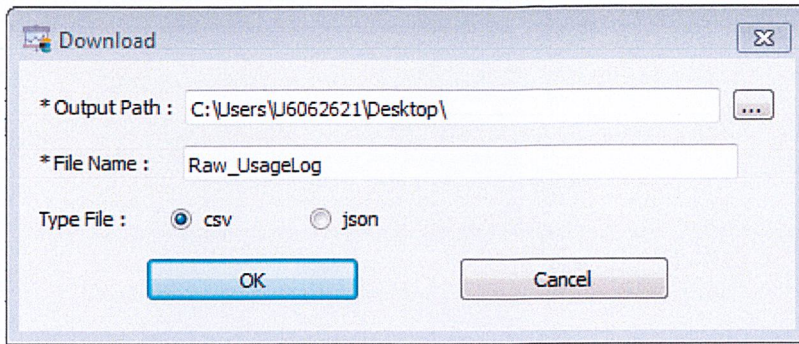
รูปที่ 4.26 ตัวอย่าง Report ของโปรแกรม Access Log Analysis From Elasticsearch

3) การดาวน์โหลด Raw Data

การดาวน์โหลด Raw Data จะมีการเลือก Field ที่ต้องการดาวน์โหลด เมื่อเลือก Field เรียบร้อยแล้ว ดังรูปที่ 4.27 ให้กด Download จะปรากฏหน้าต่างเลือก Path, ตั้งชื่อไฟล์ และกำหนดประเภทของไฟล์ ดังรูปที่ 4.28

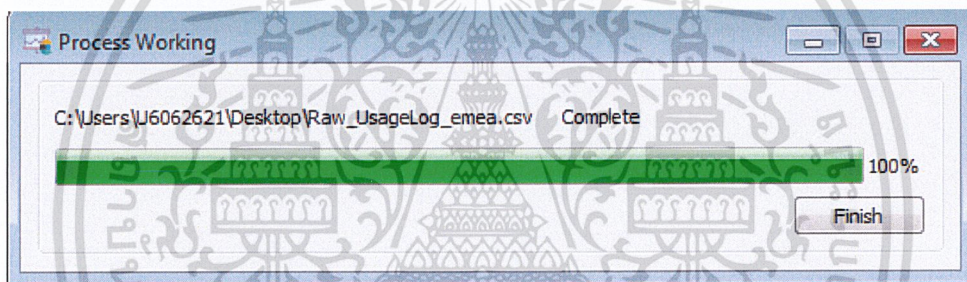


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.27 ตัวอย่างการเพิ่ม Field ลงในตาราง Raw Data
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีเมลห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.28 หน้าต่างการเลือก Path, ตั้งชื่อไฟล์ และกำหนดชนิดของไฟล์
ของการดาวน์โหลด Raw Data

เมื่อกดปุ่ม OK จะปรากฏหน้าต่าง Process Working เพื่อแสดงสถานะการทำงานของการดาวน์โหลด Raw Data ดังรูปที่ 4.29



รูปที่ 4.29 หน้าต่าง Process Working เพื่อแสดงสถานะการทำงานของการดาวน์โหลด

	A	B	C	D
1	@timestamp	Cluster_cm	ProdId_cm	UUID_cm
2	2017-12-07T00:03:42.158Z	hdc.excel	DE:UNITY	R12-1143663
3	2017-12-07T00:00:10.328Z	hdc.excel	DE:UNITY	SL1-4B9M5OE
4	2017-12-07T00:00:16.096Z	hdc.excel	DE:UNITY	GESG1-1777
5	2017-12-07T00:00:16.409Z	hdc.excel	DE:UNITY	GESG1-1777
6	2017-12-07T00:00:16.258Z	hdc.excel	DE:UNITY	GESG1-1777
7	2017-12-07T00:00:16.180Z	hdc.excel	DE:UNITY	GESG1-1777
8	2017-12-07T00:00:27.458Z	hdc.excel	DE:UNITY	GENTC-11880
9	2017-12-07T00:00:27.867Z	hdc.excel	DE:UNITY	SL1-3LBC673
10	2017-12-07T00:00:27.890Z	hdc.excel	DE:UNITY	SL1-3LBC673
11	2017-12-07T00:00:31.161Z	hdc.excel	DE:UNITY	SL1-783JGZE
12	2017-12-07T00:00:31.164Z	hdc.excel	DE:UNITY	SL1-783JGZE
13	2017-12-07T00:00:29.197Z	hdc.excel	DE:UNITY	SL1-783JGZE
14	2017-12-07T00:00:29.153Z	hdc.excel	DE:UNITY	SL1-783JGZE
15	2017-12-07T00:00:32.133Z	hdc.excel	DE:UNITY	SL1-783JGZE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.30 ตัวอย่าง Raw Data ประเภทไฟล์เป็น CSV ของการดาวน์โหลด Raw Data
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการพัฒนาชุดคำสั่งและโปรแกรมส่วนติดต่อผู้ใช้สำหรับวิเคราะห์บันทึกการเรียกใช้ข้อมูลของลูกค้า (Access Log) ในระบบต่างๆที่ให้บริการข้อมูลทางการเงิน สามารถสรุปผลการดำเนินงานได้ดังนี้

1) ชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler จำนวน 2 ชุดและชุดคำสั่งที่สั่งทำงานโดยนักวิเคราะห์ผ่านทาง Command Promp จำนวน 1 ชุดใช้สำหรับวิเคราะห์ Access Log ในระบบ Analytics Data Cloud (ADC) โดยผลการวิเคราะห์ที่อยู่ในรูปแบบของ Daily Report, Monthly Report, KPI Report และแสดงผลในรูปแบบกราฟบน Grafana ซึ่งผลการประเมินจากผู้ใช้งานได้ค่าเฉลี่ย 5.0

2) ชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler จำนวน 5 ชุดใช้สำหรับวิเคราะห์ Access Log ในระบบ Elektron Time Series (ETS) โดยผลการวิเคราะห์อยู่ในฐานข้อมูล MongoDB และแสดงผลในรูปแบบกราฟบน Tableau ซึ่งผลการประเมินจากผู้ใช้งานได้ค่าเฉลี่ย 4.57

3) ชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler จำนวน 2 ชุดใช้สำหรับวิเคราะห์ Access Log ในระบบ Elektron Tick History (ETH) โดยผลการวิเคราะห์อยู่ในรูปแบบของ Daily Report, KPI Report และแสดงผลในรูปแบบกราฟบน Grafana ซึ่งผลการประเมินจากผู้ใช้งานได้ค่าเฉลี่ย 5.0

4) โปรแกรม Access Log Analysis From Elasticsearch ใช้สำหรับดาวน์โหลดและวิเคราะห์ Access Log จากฐานข้อมูล Elasticsearch โดยผลการดาวน์โหลดอยู่ในรูปแบบของไฟล์ประเภท JSON หรือ CSV และผลการวิเคราะห์อยู่ในรูปแบบของไฟล์ CSV ซึ่งผลการประเมินจากผู้ใช้งานได้ค่าเฉลี่ย 4.64

จากผลการดำเนินงานในแต่ละระบบทำให้นักวิเคราะห์สามารถหารูปแบบการใช้งานของลูกค้า ตรวจสอบการใช้งานที่ผิดปกติ หาสาเหตุที่ทำให้เกิดความผิดปกติกับระบบได้รวดเร็วยิ่งขึ้น สามารถนำข้อมูลไปใช้ในการทดสอบระบบที่พัฒนาขึ้นมาใหม่ได้ รวมไปถึงสามารถนำข้อมูลที่ได้จากการวิเคราะห์ไปใช้ตัดสินใจในการบริหารที่จัดเก็บข้อมูลภายในระบบให้มีความคุ้มค่ามากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ข้อจำกัด

1) โปรแกรม Access Log Analysis From Elasticsearch

- สามารถใช้งานได้บนระบบปฏิบัติการ Windows เท่านั้น
- หาก Field ในฐานข้อมูล Elasticsearch มีการเปลี่ยนแปลง Field ที่อยู่บนโปรแกรมไม่สามารถเปลี่ยนแปลงตามได้ แต่สามารถแก้ไขได้ โดยการ Add Log ใหม่
- การกรองช่วงเวลาสามารถกรองได้เฉพาะ Field ที่ชื่อ '@timestamp' บนฐานข้อมูล Elasticsearch เท่านั้น
- ในการสร้างรายงานสามารถใช้ Operator ได้เฉพาะ avg (Average), cardinality, date histogram, extended Stats, histogram, max, min, missing, stats, sum และ term เท่านั้น

2) ชุดคำสั่งที่ตั้งเวลาการทำงานผ่านทาง Task Scheduler

- หากถึงเวลาที่ Scripts ทำงานแล้วข้อมูลที่ใช้ในการวิเคราะห์ไม่ครบ Scripts จะไม่ทำงานต่อ ผู้ใช้จะต้องสั่ง run Scripts แก้ไขในช่วงเวลาดังกล่าวเอง
- ผู้ใช้จะต้องอัปเดตข้อมูลบน Dashboard ของ Tableau เอง

5.3 ข้อเสนอแนะ

เนื่องจากในอนาคตนักวิเคราะห์อาจมีความต้องการข้อมูลจากการวิเคราะห์เพิ่มมากขึ้น ดังนั้นควรมีการพัฒนาโปรแกรมที่ใช้ในการวิเคราะห์ข้อมูลโดยสามารถตั้งเวลาการทำงานได้ หรือสามารถสร้างชุดคำสั่งขึ้นมาเพื่อให้นักวิเคราะห์นำไปใช้งานต่อได้สะดวก

เอกสารอ้างอิง

- กรรณิกา ทิตาราม, ส่งศักดิ์ ทิตาราม. 2525. การวิเคราะห์และตีความข้อมูล. สืบค้นเมื่อวันที่ 24 กันยายน, 2560, [Online]. เข้าถึงได้จาก: <http://kanchanapisek.or.th/kp6/sub/book/book/book.php?book=6&chap=11&page=t6-11-infodetail06.html>
- ทวีรัตน์ นวลช่วย. 2558. Python Programming. สืบค้นเมื่อวันที่ 5 มิถุนายน, 2560, [Online]. เข้าถึงได้จาก: <https://sites.google.com/site/dotpython/input-and-output/variable-type>
- งานบริการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ มหาวิทยาลัยราชภัฏเพชรบูรณ์. 2557. JSON คืออะไร. สืบค้นเมื่อวันที่ 5 มิถุนายน, 2560, [Online]. เข้าถึงได้จาก: <http://comservice.pcru.ac.th/home/index.php/2015-06-24-04-36-41/126-json>
- วรพงษ์ น่วมอินทร์. 2554. แบบจำลองเครื่องอนุमानความรู้ด้วยเทคนิค Ripple Down Rules ที่นำกลับมาใช้ใหม่. วิทยานิพนธ์ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.
- Elastic. 2558. Elasticsearch. สืบค้นเมื่อวันที่ 5 มิถุนายน, 2560, [Online]. เข้าถึงได้จาก: <https://www.elastic.co/products/elasticsearch>
- Elastic. 2558. Kibana. สืบค้นเมื่อวันที่ 5 มิถุนายน, 2560, [Online]. เข้าถึงได้จาก: <https://www.elastic.co/products/kibana>
- mongoDB. 2557. สืบค้นเมื่อวันที่ 20 กันยายน, 2560, [Online]. เข้าถึงได้จาก: <https://www.mongodb.com/>
- graphite. 2556. สืบค้นเมื่อวันที่ 10 มิถุนายน, 2560, [Online]. เข้าถึงได้จาก: <https://graphiteapp.org/>
- Tableau. 2556. สืบค้นเมื่อวันที่ 25 กันยายน, 2560, [Online]. เข้าถึงได้จาก: <https://www.tableau.com/>
- Grafana. 2557. สืบค้นเมื่อวันที่ 22 มิถุนายน, 2560, [Online]. เข้าถึงได้จาก: <https://grafana.com/>
- Qt. 2560. สืบค้นเมื่อวันที่ 25 ตุลาคม, 2560, [Online]. เข้าถึงได้จาก: <https://www.qt.io/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

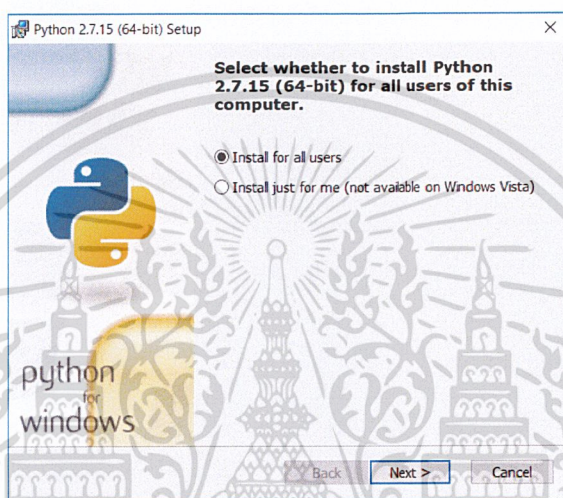
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

การติดตั้งเครื่องมือเพื่อใช้ในการพัฒนาโปรแกรม

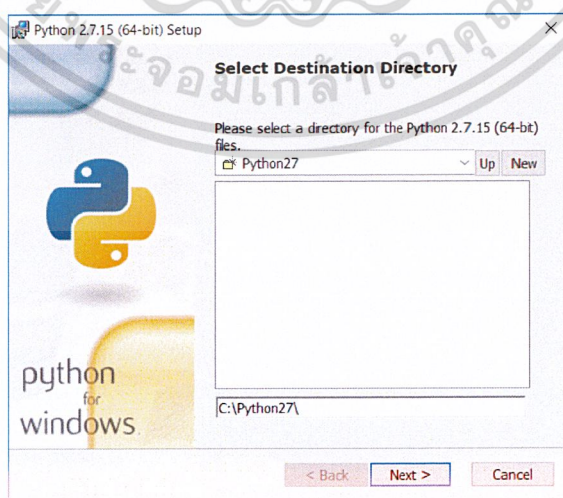
ก.1 การติดตั้งภาษา Python

- 1) ดาวน์โหลดตัวติดตั้ง Python 2.7 64 บิตจาก <https://www.python.org/downloads/>
- 2) เมื่อดาวน์โหลดตัวติดตั้งสำเร็จแล้วให้เปิดตัวติดตั้ง Python ดังรูปที่ ก.1 แล้วกด Next



รูปที่ ก.1 หน้าต่างติดตั้งของภาษา Python

- 3) ทำการกำหนด Path ไปที่ C:\Python27 แล้วกด Next เพื่อเริ่มการติดตั้ง ดังรูปที่ ก.2

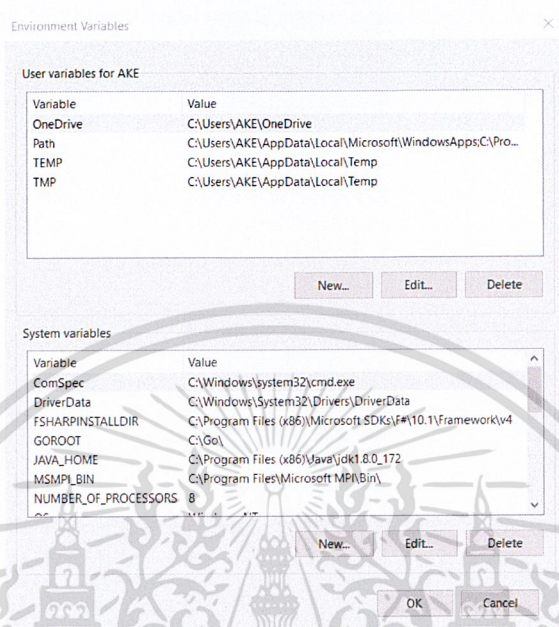


รูปที่ ก.2 หน้าต่างสำหรับกำหนด Path ในการติดตั้งภาษา Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับคุณใช้เฉพาะเพื่อการศึกษานั้น ไม่ขอเอาใจให้อ่างไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

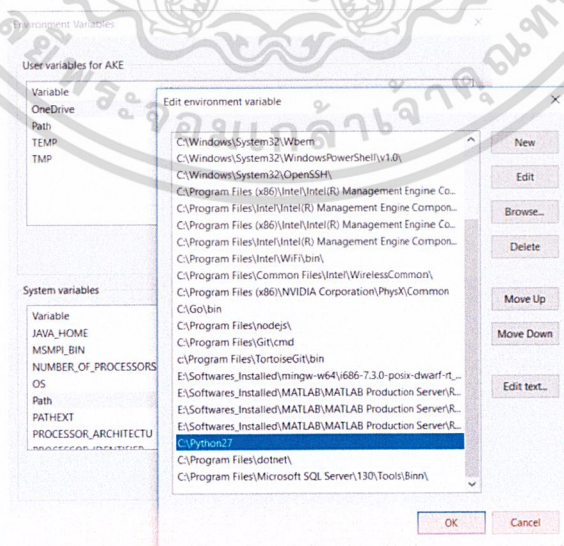
4) เมื่อติดตั้งสำเร็จให้กด Finish

5) คลิกขวาที่ This PC > Properties > Advanced system settings > Environment Variables... เพื่อทำการตั้งค่า Environment Variables ดังรูปที่ ก.3



รูปที่ ก.3 หน้าต่างตั้งค่า Environment Variables

6) ทำการเพิ่ม Path C:\Python27 ลงใน Variable Path ในส่วนของ System variables ดังรูปที่ ก.4

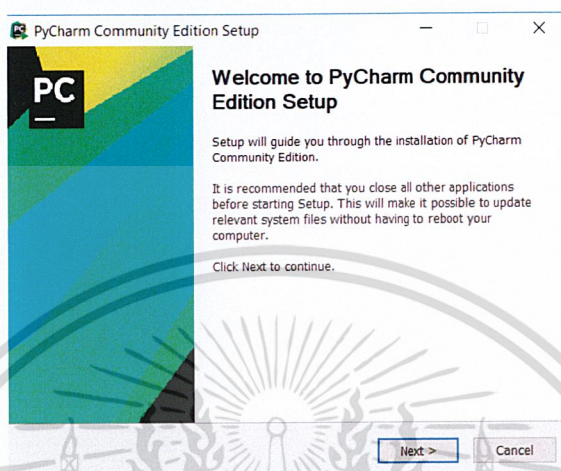


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ปรับปรุงแก้ไขโดยไม่ได้รับอนุญาต
 รูปที่ ก.4 การตั้งค่า Path ภาษา Python บนเครื่องคอมพิวเตอร์ระบบปฏิบัติการ Windows
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.2 การติดตั้งโปรแกรม PyCharm

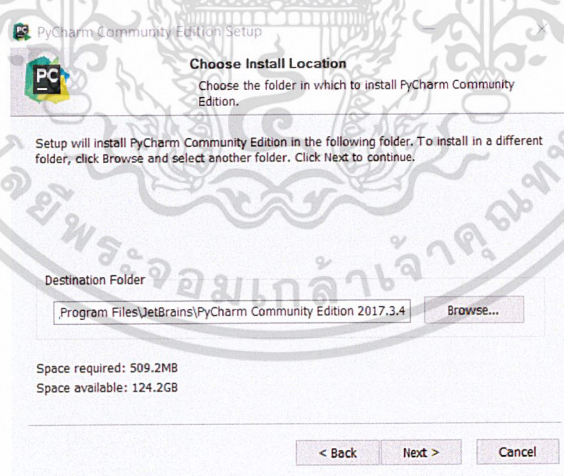
1) ดาวน์โหลดไฟล์ติดตั้ง PyCharm จาก <https://www.jetbrains.com/pycharm/download/#section=windows>

2) เมื่อดาวน์โหลดไฟล์ติดตั้งสำเร็จแล้วให้เปิดไฟล์ติดตั้ง PyCharm ดังรูปที่ ก.5 แล้วกด Next



รูปที่ ก.5 หน้าต่างติดตั้งของโปรแกรม PyCharm

3) กำหนด Path สำหรับที่อยู่ของโปรแกรม ดังรูปที่ ก.6



รูปที่ ก.6 หน้าต่างสำหรับกำหนด Path ในการติดตั้งโปรแกรม PyCharm

4) เมื่อติดตั้งโปรแกรมเสร็จให้กด Finish

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.3 การติดตั้งฐานข้อมูล MongoDB

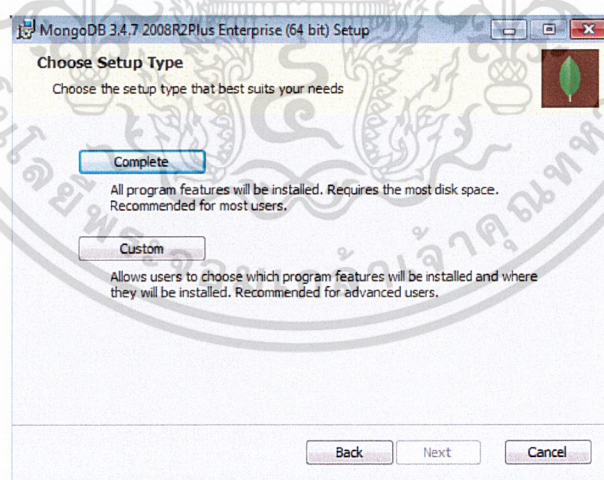
1) ดาวน์โหลดตัวติดตั้ง MongoDB จาก <https://www.mongodb.com/download-center?jmp=nav#atlas>

2) เมื่อดาวน์โหลดตัวติดตั้งสำเร็จแล้วให้เปิดตัวติดตั้ง MongoDB ดังรูปที่ ก.7 แล้วกด Next



รูปที่ ก.7 หน้าต่างติดตั้งของฐานข้อมูล MongoDB

3) เลือกประเภทของการติดตั้งเป็น Complete ดังรูปที่ ก.8



รูปที่ ก.8 หน้าต่างเลือกประเภทของการติดตั้งฐานข้อมูล MongoDB

4) เมื่อติดตั้งฐานข้อมูลเสร็จให้กด Finish

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้