



ระบบอนุมัติสินเชื่อเข้าซื้อรถยนต์และสินเชื่อธุรกิจขนาดย่อม

CREDIT APPROVAL PROCESS

นายกนัตสร เจนจิรวัดนา

โครงการสหกิจนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศทางธุรกิจ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

พ.ศ. 2559

ระบบอนุมัติสินเชื่อเข้าซื้อรถยนต์และสินเชื่อธุรกิจขนาดย่อม

CREDIT APPROVAL PROCESS

นายกนต์ธร เจนจิรวัฒนา

โครงการสหกิจนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศทางธุรกิจ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

พ.ศ. 2559

คณะกรรมการสอบ

..... ประธานการสอบ

(ดร.สรมย์พร เจริญพิทย์)

..... กรรมการสอบ

(ผศ.เกษม ทิพย์ธาราจันทร์)

..... กรรมการและอาจารย์ที่ปรึกษา

(อาจารย์นุชนารถ พงษ์พานิช)

..... ประธานสหกิจศึกษาสาขาวิชา BI

(อาจารย์นุชนารถ พงษ์พานิช)

ชื่อโครงการ	ระบบอนุมัติสินเชื่อเช่าซื้อรถยนต์และสินเชื่อธุรกิจขนาดเล็ก
	CREDIT APPROVAL PROCESS
ผู้เขียน	นายกนต์ธร เจริญวัฒนา
คณะวิชา	เทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศทางธุรกิจ
อาจารย์ที่ปรึกษา	อาจารย์ นุชนารถ พงษ์พานิช
พนักงานที่ปรึกษา	คุณอาทิตย์ เลิศยนต์ชีพ
ชื่อบริษัท	บริษัท ออฟฟแลนด์ จำกัด
ประเภทธุรกิจ/สินค้า	บริษัทผู้ให้เช่าบริการทางด้านโซลูชันระบบสารสนเทศ

บทสรุป

งานที่ได้รับมอบหมายนั้นเป็นส่วนหนึ่งของระบบอนุมัติสินเชื่อเช่าซื้อรถยนต์และสินเชื่อธุรกิจขนาดเล็ก คือส่วนของการนำเข้าตรวจสอบและนำออกข้อมูล โดยส่วนนี้มีหลักการทำงานโดยการนำไฟล์เข้าในฐานข้อมูลก่อนนำไปประมวลผลกับข้อมูลในฐานข้อมูลตามเงื่อนไขของข้อกำหนดทางธุรกิจ โดยจะต้องทำทั้ง Front-End และ Back-End เพื่อให้ได้กระบวนการทำงานและผลลัพธ์ที่ถูกต้อง โดยงานที่ทำนั้นจะต้องทำภายใต้พื้นฐานการเขียนโค้ดของบริษัทเพื่อให้ได้รูปแบบการเขียนโค้ดที่สามารถรับช่วงต่อได้ทันที เมื่อผู้จัดทำจบการสหกิจแล้วต้องเข้ามาแก้ไขในส่วนงาน และพื้นฐานภาษาที่ต้องใช้ในการทำงานนั้น ได้แก่ Java, TypeScript, SQL, CSS, HTML เป็นต้น

จากการดำเนินงานผลที่ได้คือ สามารถสร้างระบบนำเข้าตรวจสอบและนำออกข้อมูลได้อย่างมีประสิทธิภาพ และสามารถประมวลผลข้อมูลได้รวดเร็ว ใช้ทรัพยากรของระบบน้อย โดยจะต้องสามารถใช้งานได้จริง และทางผู้จัดทำยังได้ความรู้ทางภาษาต่างๆมากมาย และการประสานงานในการทำงานที่ต้องไปทำร่วมกับคนอื่นๆ ที่ทำอยู่ในโครงการเดียวกับทางผู้จัดทำ และได้ประสบการณ์ในชีวิตจริงมากมาย

กิตติกรรมประกาศ

ขอขอบคุณบริษัท ออฟฟแลนท์ จำกัด ที่มอบโอกาสให้ข้าพเจ้าได้มีโอกาสในการทำงานจริงของบริษัท ทำให้ข้าพเจ้าได้รับประสบการณ์จริงจากการทำงานที่ไม่สามารถหาได้ในชั้นเรียนและเปิดโอกาสให้เรียนรู้ประสบการณ์ต่างๆภายในสภาพแวดล้อมของบริษัทที่เป็นประสบการณ์อันมีค่ายิ่งของข้าพเจ้า ทั้งรูปแบบการเข้าประชุมงาน การทำงานร่วมกันหลายคนภายในชิ้นงานเดียวที่แบ่งออกเป็นหลายๆระบบย่อยและแต่ละคนรับผิดชอบในแต่ละระบบย่อย

ขอขอบคุณพี่พนักงานทุกคนที่คอยให้ความช่วยเหลือในด้านต่างๆ ค่อยให้ความรู้ และให้คำปรึกษาตลอดระยะเวลา 4 เดือนที่ผ่านมาข้าพเจ้าได้รับการดูแลเป็นอย่างดีและที่สำคัญขอขอบพระคุณพนักงานที่ปรึกษา คุณอาทิตย์ เลิศยนต์ชีพ ที่ให้คำปรึกษาและความเอ็นดูมาตลอด รวมถึงอาจารย์ที่ปรึกษา อาจารย์ นุชนารถ พงษ์พานิช ที่ให้คำแนะนำต่างๆตลอดมา จึงขอขอบคุณทุกท่านที่คอยให้การช่วยเหลือข้าพเจ้ามาตลอด ข้าพเจ้ารู้สึกซาบซึ้งใจเป็นอย่างยิ่ง และหากข้าพเจ้าได้กระทำการอันใดที่เป็นเรื่องไม่ดีทั้งหลายข้าพเจ้าขออภัยมา ณ ที่นี้ด้วย



TNi

สารบัญ

หน้า

บทสรุป

ก

กิตติกรรมประกาศ

ข

สารบัญ

ค

สารบัญภาพประกอบ

ง

บทที่

บทที่ 1 บทนำ

1

1.1 ชื่อและสถานที่ตั้งของสถานประกอบการ

1

1.1.1 ชื่อสถานประกอบการ

1

1.1.2 ที่ตั้งสถานประกอบการ

1

1.2 ลักษณะธุรกิจของสถานประกอบการหรือการให้บริการหลักขององค์กร

2

1.3 รูปแบบการจัดการและบริหารองค์กร

4

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

4

1.5 พนักงานที่ปรึกษาและตำแหน่งของพนักงานที่ปรึกษา

4

1.6 ระยะเวลาที่ปฏิบัติงาน

4

1.7 ที่มาและความสำคัญของปัญหา

5

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของการปฏิบัติงาน

5

1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

5

1.10 นิยามศัพท์เฉพาะ

6

บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

7

2.1 ทฤษฎี

7

2.1.1 OOP: Object-Oriented Programming

7

2.1.2	ภาษา Hypertext markup language (HTML)	7
2.1.3	ภาษา Java	8
2.1.4	ภาษา JavaScript	9
2.1.5	ภาษา Typescript	11
2.1.6	ภาษา SQL	12
2.2	เทคโนโลยี	14
2.2.1	Angular2	14
2.2.2	JavaScript Object Notation (JSON)	15
2.2.3	Hibernate Framework	16
2.2.4	Visual Studio Code	18
2.2.5	Rational Application Developer for Web Sphere	18
2.2.6	NodeJS	19

บทที่ 3 แผนปฏิบัติงานและขั้นตอนการดำเนินงาน 21

3.1	แผนงานและระยะเวลาในการปฏิบัติงาน	21
3.2	รายละเอียดของโครงการที่ได้รับมอบหมาย	22
3.2.1	ส่วน Front End	22
3.2.2	ส่วน Back End	22
3.2.3	Database	22
3.3	ขั้นตอนการดำเนินงาน	23
3.3.1	เตรียม Environment	23
3.3.2	ศึกษาใช้งานโปรแกรมที่ได้รับ และ Coding rule	23
3.3.3	สร้าง หน้าเว็บ ตามต้นแบบที่รับมา	23
3.3.4	สร้าง Web Service ต่างๆให้กับไฟล์ .html และ .ts	24
3.3.5	นำข้อมูลลง Database	25
3.3.6	เชื่อมต่อ Web Service ให้กลับไฟล์ .ts	25
3.3.7	นำโค้ดทั้งหมดอัปข้อมูลใน Server	26

บทที่ 4 ผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ	27
4.1 ขั้นตอนและผลการดำเนินงาน	27
4.2 ผลการวิเคราะห์ข้อมูล	37
4.3 วิจัยข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายของการจัดทำ โครงการ	38
บทที่ 5 บทสรุปและข้อเสนอแนะ	39
5.1 สรุปผลการดำเนินงาน	39
5.2 แนวทางแก้ปัญหา	39
5.3 ข้อเสนอแนะจากการดำเนินงาน	39
เอกสารอ้างอิง	40
ประวัติผู้จัดทำโครงการ	42



TNI

สารบัญภาพประกอบ

หน้า

ภาพที่ 1.1 โลโก้บริษัท อฟวาแลนท์ จำกัด.....	1
ภาพที่ 1.2 แผนที่ตั้งบริษัท อฟวาแลนท์ จำกัด ที่มา : http://www.avalant.co.th	2
ภาพที่ 1.3 แสดงผลิตภัณฑ์ของทางบริษัท	3
ภาพที่ 1.4 โครงสร้างบริษัท.....	4
ภาพที่ 2.1 รูปโลโก้ HTML5.....	8
ภาพที่ 2.2 ภาพโลโก้ของ Angular 2	14
ภาพที่ 2.3 ภาพสถาปัตยกรรมของ Angular 2.....	15
ภาพที่ 2.4 ตัวอย่างโค้ดที่ใช้ Format JSON	16
ภาพที่ 2.5 โครงสร้าง Hibernate	17
ภาพที่ 2.6 ตัวอย่างหน้าต่างภายในโปรแกรม Visual Studio Code	18
ภาพที่ 2.7 รูปหน้าต่างโปรแกรมตอนเริ่มเข้าใช้งาน	19
ภาพที่ 2.8 รูปตัวอย่างหน้าต่างภายในโปรแกรม.....	19
ภาพที่ 2.9 ตัวอย่างคำสั่งสำหรับการนำ Package หรือ Module คนอื่นมาใช้.....	20
ภาพที่ 3.1 ตัวอย่างโค้ด HTML5 ที่ใช้สร้างหน้าเว็บ	24
ภาพที่ 3.2 ตัวอย่างโค้ด TypeScript ของ Front-End	25
ภาพที่ 3.3 ตัวอย่างโค้ดของไฟล์ Controller ที่ใช้สำหรับเรียกฟังก์ชันจาก Back-End	26
ภาพที่ 4.1 ตัวอย่างโค้ดฟังก์ชันที่ใช้สำหรับการนำออกข้อมูลมาเป็นไฟล์ Excel	27
ภาพที่ 4.2 ตัวอย่าง โค้ดฟังก์ชันที่ใช้สำหรับประมวลผลตามเงื่อนไข.....	28
ภาพที่ 4.3 ตัวอย่าง โค้ดของฟังก์ชันการดึงข้อมูลจากฐานข้อมูล	28
ภาพที่ 4.4 ตัวอย่าง โค้ดControllerของ Back-End ที่ใช้เชื่อมกับ Front-End	29
ภาพที่ 4.5 ขั้นตอนการไฟล์ Create Web Service.....	29
ภาพที่ 4.6 ขั้นตอนการ Test Web Service ด้วย Post Man.....	30
ภาพที่ 4.7 การอัปโหลดขึ้น Server	30
ภาพที่ 4.8 ความสัมพันธ์ของฐานข้อมูล.....	31
ภาพที่ 4.9 หน้าจอการนำไฟล์เข้าระบบ	32
ภาพที่ 4.10 หน้าจอการ Upload ไฟล์.....	32
ภาพที่ 4.11 หน้าจอเลือกไฟล์จากฐานข้อมูลของคอมพิวเตอร์.....	33

ภาพที่ 4.12 หน้าจอเลือกไฟล์ที่ต้องการและรอการ Upload ขึ้นฐานข้อมูล.....	33
ภาพที่ 4.13 หน้าจอ Upload ไฟล์ลงฐานข้อมูลเสร็จสิ้น.....	34
ภาพที่ 4.14 แสดงตำแหน่งของไฟล์เมื่อถูกอัปโหลดแล้ว.....	34
ภาพที่ 4.15 แสดงการเลือกไฟล์ที่ต้องการ	35
ภาพที่ 4.16 แสดงไฟล์ที่ถูกเปลี่ยนสถานะเป็น “P” เมื่อถูกนำเข้าระบบ.....	35
ภาพที่ 4.17 แสดงไฟล์ที่ถูกเปลี่ยนสถานะเป็น “A” เมื่อประมวลผลเสร็จแล้ว	36
ภาพที่ 4.18 แสดงไฟล์ที่นำออกเมื่อกดที่ปุ่มไอคอน	36
ภาพที่ 4.19 ตัวอย่างไฟล์ที่ถูกนำออกมาตรวจสอบ.....	37



บทที่ 1

บทนำ

1.1 ชื่อและสถานที่ตั้งของสถานประกอบการ



ภาพที่ 1.1 โลโก้บริษัท อฟวาแลนท์ จำกัด

1.1.1 ชื่อสถานประกอบการ

ชื่อภาษาไทย : บริษัท อฟวาแลนท์ จำกัด

ชื่อภาษาอังกฤษ : Avalant Company Limited

1.1.2 ที่ตั้งสถานประกอบการ

ที่อยู่ : 20 ถนนสาทรเหนือ ชั้น 15 อาคารบุปผชาติ แขวงสีลม เขตบางรัก

กรุงเทพฯ 10500 ประเทศไทย

โทรศัพท์ : +66-2-633-9367-9

โทรสาร : +66-2-633-8174

อีเมล : mkt-avalant@avalant.co.th

เว็บไซต์ : <http://www.avalant.co.th>



ภาพที่ 1.2 แผนที่ตั้งบริษัท อฟวาแลนท์ จำกัด

ที่มา : <http://www.avalant.co.th>

1.2 ลักษณะธุรกิจของสถานประกอบการหรือการให้บริการหลักขององค์กร

บริษัท อฟวาแลนท์ จำกัด (Avalant Company Limited) เป็นบริษัทผู้ให้คำปรึกษาทางด้านโซลูชันระบบสารสนเทศ (Information Technology) เพื่อตอบสนองการดำเนินงานต่างๆ ภายในองค์กร โดยมีวัตถุประสงค์เพื่อช่วยให้การดำเนินงานต่างๆ ภายในองค์กรมีประสิทธิภาพ และประสิทธิผลที่ดีขึ้น ส่งผลสืบเนื่องถึงผลกำไรที่มากขึ้นขององค์กร

บริษัทฯ มีความเชี่ยวชาญในธุรกิจหลากหลายประเภททั้งในด้านธุรกิจประกันภัย ธุรกิจการเงิน ธุรกิจค้าปลีก รวมไปถึงธุรกิจทางด้านโทรคมนาคม โดยการให้บริการนั้นครอบคลุมตั้งแต่การวิเคราะห์และออกแบบกระบวนการทางธุรกิจ การวางแผนเชิงกลยุทธ์ การบริหารโครงการ การวิเคราะห์ความต้องการ การออกแบบและพัฒนาความต้องการ การตรวจสอบและรับประกันคุณภาพ การพัฒนาโครงสร้างพื้นฐานที่สำคัญ (Infrastructure) และการออกแบบโซลูชันที่เหมาะสม (Solution Architect) ให้แก่องค์กร

วิสัยทัศน์ (Vision)

บริษัทฯ ต้องการเป็นบริษัทชั้นนำในการให้คำปรึกษาทางด้านโซลูชันระบบสารสนเทศ (Information Technology) ในระดับภูมิภาคเอเชีย

พันธกิจ (Mission)

เพื่อช่วยเหลือลูกค้าของเราออกแบบโซลูชันทั้งทางด้านธุรกิจและด้านระบบสารสนเทศ (Information Technology) เพื่อให้เกิดผลตอบแทนจากการลงทุนผ่านผลกำไรและผลการดำเนินงานที่ดีขึ้น

ผลิตภัณฑ์และการบริการ

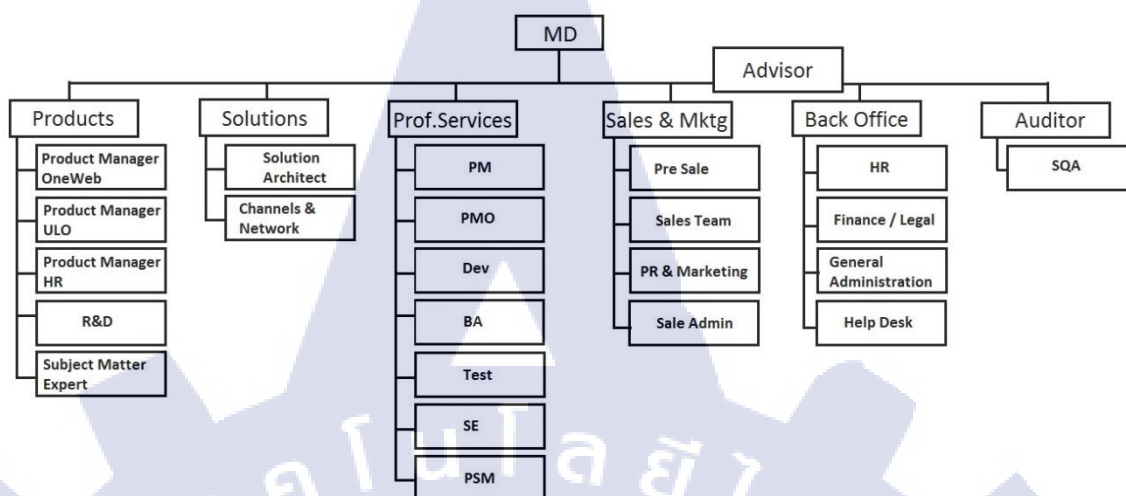
บริษัทมีการวางแผนและจัดการให้ระบบสารสนเทศสอดคล้องกับทางธุรกิจ โดยบริษัทได้ทำงานร่วมกับผู้บริหารในหลายประเภทธุรกิจโดยเฉพาะธุรกิจประกันภัยและธุรกิจการเงินเพื่อส่งมอบกระบวนการดำเนินงานและเทคโนโลยีเพื่อปรับปรุงพัฒนากระบวนการทางธุรกิจและสนับสนุนทิศทางกลยุทธ์ธุรกิจขององค์กร บริษัทฯ เชื่อมมั่นในแนวทางที่เน้นการปฏิบัติโดยให้ความสำคัญกับความต้องการทางด้านธุรกิจของลูกค้า ก่อนที่จะพัฒนาโซลูชันเพื่อตอบสนองความต้องการของลูกค้า ดังนั้นบริษัทฯ จึงเน้นการทำงานโดยเริ่มที่การสื่อสารกับผู้บริหารและผู้ที่เกี่ยวข้องในองค์กรของลูกค้าเพื่อที่จะกำหนดเป้าหมายในการพัฒนากระบวนการทางธุรกิจ (Business Process Improvement: BPI) ซึ่งวิธีการดังกล่าวจะส่งผลให้สามารถวัดผลตอบแทนจากการลงทุนได้จริง

บริษัท อีฟวอเลท์ จำกัด นั้นเน้นการทำผลิตภัณฑ์ที่ตอบโจทย์ให้แก่ธุรกิจ โดยเน้นแนวความคิดในเรื่องของ “One web” ซึ่งในแนวความคิดนี้จะทำให้ผลิตภัณฑ์ที่จะส่งมอบให้แก่ธุรกิจสามารถนำไปใช้งานในทุก Platform



ภาพที่ 1.3 แสดงผลิตภัณฑ์ของทางบริษัท

1.3 รูปแบบการจัดการและบริหารองค์กร



ภาพที่ 1.4 โครงสร้างบริษัท

บริษัทมีโครงสร้างองค์กรเป็นแบบการแบ่งตามสายงานด้านต่างๆ โดยในแต่ละสายงานจะมีหัวหน้าใหญ่ในแต่ละสายงานนั้น ซึ่งหัวหน้าในแต่ละสายงานจะมีการรายงานผลต่างๆ ให้กับ MD (Managing Director) ได้โดยตรง

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ตำแหน่ง

Internship

หน้าที่งานที่ได้รับมอบหมาย

Java Developer

1.5 พนักงานที่ปรึกษาและตำแหน่งของพนักงานที่ปรึกษา

พนักงานที่ปรึกษา: คุณอาทิตย์ เลิศยนต์ชีพ

ตำแหน่ง:

Manager

1.6 ระยะเวลาที่ปฏิบัติงาน

เริ่มต้นปฏิบัติงานสหกิจศึกษาตั้งแต่ที่ 1 มิถุนายน 2559 สิ้นสุดภาคปฏิบัติงานวันที่ 30 กันยายน 2559 (รวมเป็นระยะเวลา 4 เดือน)

1.7 ที่มาและความสำคัญของปัญหา

ในปัจจุบันเทคโนโลยีสารสนเทศเข้ามามีบทบาทมากในโลกของธุรกิจในทุกๆด้านไม่ว่าจะเป็นด้านการเงิน การขนส่ง การคลัง ฯลฯ บริษัทได้เล็งเห็นความสำคัญของปัญหาที่เกิดขึ้นเกี่ยวกับการเงิน จึงมีการจัดทำและออกแบบการทำสินเชื่อและเช่าซื้อของรถยนต์เพื่อประโยชน์ของบริษัทที่จะทำให้การทำงานเป็นไปอย่างคล่องแคล่วมากขึ้นและมีความถูกต้องมากที่สุดทำให้บริษัทที่ได้ใช้งาน เกิดประโยชน์สูงสุดต่อบริษัทเอง ทั้งนี้ยังสามารถเข้าตรวจสอบข้อมูลย้อนหลังได้อย่างรวดเร็ว สะดวกต่อการตรวจสอบงานต่างๆให้มีประสิทธิภาพมากขึ้น

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของการปฏิบัติงาน

1. เพื่อให้มีโอกาสการเรียนรู้และได้รับประสบการณ์ชีวิตการทำงานที่แท้จริง
2. เพื่อฝึกการเป็นนักวิจัยค้นคว้าและพัฒนาอย่างแท้จริง
3. เพื่อนำประสบการณ์ที่ได้รับจากการสหกิจศึกษามาประยุกต์ใช้ในการทำงานในอนาคต
4. ศึกษาองค์ความรู้ในการเขียนโปรแกรมภาษาต่างๆ
5. สามารถเข้าใจหลักการงานและถ่ายทอดองค์ความรู้ได้
6. นำองค์ความรู้มาฝึกปฏิบัติงานได้จริง
7. จัดทำเอกสารอธิบายหลักการงาน

1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

1. สามารถค้นคว้าและแก้ไขปัญหาได้ด้วยตัวเอง
2. วางแผนจัดการและแบ่งเวลาการทำงานได้อย่างเหมาะสม
3. พัฒนาทักษะการนำเสนอผลงาน
4. ได้รับองค์ความรู้เกี่ยวกับการเขียนโปรแกรมภาษาต่างๆ
5. สามารถนำองค์ความรู้ไปถ่ายทอดและพัฒนาต่อยอดได้
6. เรียนรู้วัฒนธรรมภายในองค์กร
7. ฝึกการแก้ปัญหาอย่างเป็นขั้นตอน
8. ปรับตัวให้เข้ากับกฎระเบียบในสังคมการทำงานรวมถึงเพื่อนร่วมงานได้

1.10 นิยามศัพท์เฉพาะ

1. Business Rule : กฎเกณฑ์ทางธุรกิจ, กฎการทำงานของธุรกิจ
2. System Requirement Specification (SRS) : ข้อกำหนดความต้องการทางซอฟต์แวร์
3. Credit Approval Process (CAP) หมายถึง กระบวนการอนุมัติสินเชื่อซึ่งเป็นตัวควบคุมการตรวจสอบ และดำเนินการเป็นกิจกรรมที่แยกจากกิจกรรมการขาย
4. Web Application หมายถึง โปรแกรมประยุกต์ที่เข้าถึงด้วยโปรแกรมค้นดูเว็บผ่านเครือข่ายคอมพิวเตอร์
5. Front-End หมายถึง ส่วนหน้าจอที่แสดงผลสำหรับติดต่อกับผู้ใช้
6. Back-End หมายถึง ส่วนที่มีหน้าที่ติดต่อกับฐานข้อมูลและประมวลผลข้อมูล
7. Internet หมายถึง เครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่ประกอบไปด้วยเครือข่ายเล็กๆหลายเครือข่าย
8. Tag หมายถึง รหัสที่ใช้ในแฟ้มข้อมูล หรือใช้ในเรื่องของโครงสร้างข้อมูล (Data Structure) ที่จะเป็นตัวบ่งบอกว่าจะต้องมีการตีความเรื่องนั้นๆอย่างไร ซอฟต์แวร์ต่างๆนิยมใช้ป้ายระบุนี้ด้วยวัตถุประสงค์ต่างๆกัน คำสั่ง HTML ก็นิยมใช้รหัสนี้ด้วยเช่นกัน ดู HTML ประกอบ
9. Compile หมายถึง กระบวนการแปลภาษาโปรแกรมระดับสูงเป็นภาษาเครื่อง เพื่อให้คอมพิวเตอร์เข้าใจและสามารถทำงานตามได้
10. Runtime หมายถึง เวลาที่ใช้ไปในขณะที่คอมพิวเตอร์ปฏิบัติการตามโปรแกรมใดโปรแกรมหนึ่ง
11. Run หมายถึง กระบวนการที่สั่งให้คอมพิวเตอร์ทำงานตามคำสั่งของโปรแกรม
12. Native Code หมายถึง โปรแกรมที่ Compile ให้อยู่ในรูปของภาษาเครื่อง
13. Tool หมายถึง เครื่องมือที่ช่วยอำนวยความสะดวกต่างๆ
14. Web Browser หมายถึง โปรแกรมสำหรับเข้าใช้เว็บไซต์ผ่านเครือข่ายคอมพิวเตอร์
15. Cookies หมายถึง ข้อมูลชั่วคราวที่ถูกสร้างและจัดเก็บที่เกิดจากการเข้าใช้เว็บไซต์
16. Platform หมายถึง ที่รองรับระบบปฏิบัติการ
17. Client หมายถึง ฝ่ายของผู้ใช้งาน
18. Server หมายถึง ฝ่ายของผู้ให้บริการ

บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.1 ทฤษฎี

2.1.1 OOP: Object-Oriented Programming

OOP คือแนวคิดการมองหน่วยต่างๆ ของภาษาโปรแกรม (Entity) ให้เป็นวัตถุ (Object) ดังเช่นที่มนุษย์ปฎิชนธรรมชาติมองสิ่งต่างๆ ในชีวิตประจำวันเป็นวัตถุ

ทุก Object จะประกอบไปด้วย 2 ส่วนคือ

1. Attribute (Object Data) – เป็นส่วนที่ใช้ในการเก็บข้อมูลต่างๆ ของ Object เช่น รถมี attribute คือ ยี่ห้อรถ ชื่อรุ่น และสี เป็นต้น
2. Method (Object behavior) – เป็นสิ่งที่ Object นั้นสามารถทำได้ เช่น รถสามารถเพิ่มความเร็วได้ หรือเปลี่ยนเกียร์ได้ เป็นต้น

2.1.2 ภาษา Hypertext markup language (HTML)

HTML คือ ภาษาหลักที่ใช้ในการเขียนเว็บเพจ โดยใช้ Tag ในการกำหนดการแสดงผล HTML ย่อมาจากคำว่า Hypertext markup language โดย Hypertext หมายถึงข้อความที่เชื่อมต่อกันผ่านลิงก์ (Hyperlink) Markup language หมายถึงภาษาที่ใช้ Tag ในการกำหนดการแสดงผลสิ่งต่างๆ ที่แสดงอยู่บนเว็บเพจ ดังนั้น HTML จึงหมายถึง ภาษาที่ใช้ Tag ในการกำหนดการแสดงผลเว็บเพจที่ต่างก็เชื่อมถึงกันใน Hyperspace ผ่าน Hyperlink โดยปัจจุบัน HTML ถูกพัฒนามาถึงเวอร์ชันที่ 5 ซึ่งถูกเรียกว่า HTML5



ภาพที่ 2.1 รูปโลโก้ HTML5

2.1.3 ภาษา Java

Java หรือ Java programming language คือ ภาษาโปรแกรมเชิงวัตถุ พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่บริษัท ซัน ไมโครซิสเต็มส์ ภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส C++ โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นโอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แล้วภายหลังจึงเปลี่ยนไปใช้ชื่อ "จาวา" ซึ่งเป็นชื่อกาแฟแทน จุดเด่นของภาษา Java อยู่ที่ผู้เขียนโปรแกรมสามารถใช้หลักการของ Object-Oriented Programming มาพัฒนาโปรแกรมของตนด้วย Java ได้

ภาษา Java เป็นภาษาสำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP : Object-Oriented Programming) โปรแกรมที่เขียนขึ้นถูกสร้างภายในคลาส ดังนั้น คลาส คือ ที่เก็บเมทอด (Method) หรือพฤติกรรม (Behavior) ซึ่งมีสถานะ (State) และรูปพรรณ (Identity) ประจำพฤติกรรม (Behavior)

ข้อดีของ ภาษา Java

1. ภาษา Java เป็นภาษาที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุแบบสมบูรณ์ ซึ่งเหมาะสำหรับพัฒนาระบบที่มีความซับซ้อน การพัฒนาโปรแกรมแบบวัตถุจะช่วยให้เราสามารถใช้อำนาจหรือชื่อ ต่าง ๆ ที่มีอยู่ในระบบงานนั้นมาใช้ในการออกแบบโปรแกรมได้ ทำให้เข้าใจได้ง่ายขึ้น
2. โปรแกรมที่เขียนขึ้นโดยใช้ภาษา Java จะมีความสามารถทำงานได้ในระบบปฏิบัติการที่แตกต่างกัน ไม่จำเป็นต้องดัดแปลงแก้ไขโปรแกรม เช่น หากเขียนโปรแกรมบนเครื่อง Sun โปรแกรมนั้นก็สามารถูก compile และ run บนเครื่องพีซีธรรมดาได้

3. ภาษาจาวามีการตรวจสอบข้อผิดพลาดทั้งตอน compile time และ runtime ทำให้ลดข้อผิดพลาดที่อาจเกิดขึ้นในโปรแกรม และช่วยให้ debug โปรแกรมได้ง่าย

4. ภาษาจาวามีความซับซ้อนน้อยกว่าภาษา C++ เมื่อเปรียบเทียบ code ของโปรแกรมที่เขียนขึ้นโดยภาษา Java กับ C++ พบว่า โปรแกรมที่เขียนโดยภาษา Java จะมีจำนวน code น้อยกว่าโปรแกรมที่เขียนโดยภาษา C++ ทำให้ใช้งานได้ง่ายกว่าและลดความผิดพลาดได้มากขึ้น

5. ภาษาจาวาถูกออกแบบมาให้มีความปลอดภัยสูงตั้งแต่แรก ทำให้โปรแกรมที่เขียนขึ้นด้วยจาวามีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้น ด้วยภาษาอื่น เพราะ Java มี security ทั้ง low level และ high level ได้แก่ electronic signature, public and private key management, access control

6. มี IDE, application server, และ library ต่าง ๆ มากมายสำหรับจาวาที่เราสามารถใช้งานได้โดยไม่ต้องเสียค่าใช้จ่าย ทำให้เราสามารถลดค่าใช้จ่ายที่ต้องเสียไปกับการซื้อ tool และ s/w ต่าง ๆ

ข้อเสียของ ภาษา Java

1. ทำงานได้ช้ากว่า native code (โปรแกรมที่ compile ให้อยู่ในรูปของภาษาเครื่อง) หรือโปรแกรมที่เขียนขึ้นด้วยภาษาอื่น อย่างเช่น C หรือ C++ ทั้งนี้ก็เพราะว่าโปรแกรมที่เขียนขึ้นด้วยภาษาจาวาจะถูกแปลงเป็นภาษากลางก่อน แล้วเมื่อโปรแกรมทำงานคำสั่งของภาษากลางนี้จะถูกเปลี่ยนเป็นภาษาเครื่องอีกทีหนึ่ง ทีละคำสั่ง (หรือกลุ่มของคำสั่ง) ณ runtime ทำให้ทำงานช้ากว่า native code ซึ่งอยู่ในรูปของภาษาเครื่องแล้วตั้งแต่ compile โปรแกรมที่ต้องการความเร็วในการทำงานจึงไม่นิยมเขียนด้วยจาวา

2. Tool ที่มีในการใช้พัฒนาโปรแกรมจาวามักไม่ค่อยเก่ง ทำให้หลายอย่างโปรแกรมเมอร์จะต้องเป็นคนทำเอง ทำให้ต้องเสียเวลาทำงานในส่วนที่ tool ทำไม่ได้ ถ้าเราดู tool ของ MS จะใช้งานได้ง่ายกว่า และพัฒนาได้เร็วกว่า (แต่เราต้องซื้อ tool ของ MS และก็ต้องรันบน platform ของ MS)

2.1.4 ภาษา JavaScript

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า

"สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปที่ละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจ็กต์โอเรียนเต็ล (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server)

JavaScript ถูกพัฒนาขึ้น โดย เน็ตสเคป คอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคป จึงได้ร่วมมือกับ บริษัทซัน ไมโครซิสเต็มส์ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript ที่สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่น ต่าง ๆ มากมาย และยังสามารตอบโต้กับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้น จึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของ JavaScript จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ตาม สิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชันใหม่ๆ ออกมาด้วย (ปัจจุบันคือรุ่น 1.5) ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้

ความสามารถของ JavaScript

1. JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่ายๆ ได้ โดยไม่ต้องพึ่งภาษาอื่น
2. JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่น เมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าต่างใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น นี่คือข้อดีของ JavaScript เลขที่กว่าได้ที่ทำให้เว็บไซต์ต่างๆ ทั้งหลายหันมาใช้
3. JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือ สามารถ

เปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่ายๆนั่นเอง

4. JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์ เช่น Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างพ๊อปขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง เป็นต้น
5. JavaScript สามารถใช้ในการตรวจสอบผู้ใช้ได้ เช่น ตรวจสอบว่าผู้ใช้ ใช้ web browser อะไร
6. JavaScript สร้าง Cookies (เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เอง) ได้

ข้อดีและข้อเสียของ JavaScript

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าคุณจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านั้นเท่านั้น อย่างไรก็ตาม จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือ ไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อยู่

2.1.5 ภาษา Typescript

Typescript คือ ภาษาโปรแกรมที่ทำงานได้เสมือน Javascript เวอร์ชัน ES6 โดยมีสิ่งที่เพิ่มขึ้นมาคือ สนับสนุน Type System รวมถึงคุณสมบัติอื่นๆที่เพิ่มขึ้น เช่น Enum และความสามารถที่เพิ่มขึ้นของการโปรแกรมเชิงวัตถุ Typescript นั้นเป็น Transpiler ซึ่งนั่นคือ Typescript จะแปลงโปรแกรมที่เขียนขึ้นมาเป็นภาษา Javascript อีกครั้ง

ข้อดีของการใช้ TypeScript

1. TypeScript ทำให้คุณใช้ JavaScript สมัยใหม่ได้ในปัจจุบัน ความสามารถของ ES2015 และอื่นๆ ได้รวมไว้แล้วใน TypeScript

2. ตัวแปรที่คุณประกาศแล้วใน TypeScript จะเปลี่ยนชนิดข้อมูลไม่ได้อีกต่อไป
ข้อผิดพลาดในโปรแกรมของคุณจะน้อยลงเพราะคุณไม่มีโอกาสพลาดในการใส่ข้อมูลผิดชนิดเป็นแน่
3. TypeScript มีการตรวจสอบโค้ดในช่วง compile time ทำให้คุณดักจับข้อผิดพลาดได้
แต่ดันไม่ปล่อยให้ข้อผิดพลาดไปโผล่ในตอนทำงานจริง (runtime)
4. IDE และ Text Editor ที่ดีเยี่ยมสนับสนุนให้คุณใช้งาน TypeScript ได้อย่างสมบูรณ์

ข้อควรรู้ทั่วไปเกี่ยวกับ TypeScript

1. โค้ด JavaScript ธรรมดา ก็คือ โค้ด TypeScript ด้วยเช่นกัน
2. เราสามารถระบุหรือไม่ระบุชนิดข้อมูลใน TypeScript ก็ได้ แต่เมื่อประกาศตัวแปรแล้วเราจะเปลี่ยนชนิดข้อมูลไม่ได้
3. แม้เราจะใส่ค่าตัวแปรผิดชนิดข้อมูลใน TypeScript แต่ TypeScript ก็ยังใจดีแปลงโค้ดเป็น JavaScript ให้อยู่ดี

2.1.6 ภาษา SQL

SQL ย่อมาจาก structured query language คือภาษาที่ใช้ในการเขียนโปรแกรม เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ เป็นภาษามาตรฐานบนระบบฐานข้อมูลเชิงสัมพันธ์และเป็นระบบเปิด (open system) หมายถึงเราสามารถใส่คำสั่ง SQL กับฐานข้อมูลชนิดใดก็ได้ และ คำสั่งงานเดียวกันเมื่อส่งงานผ่าน ระบบฐานข้อมูลที่แตกต่างกันจะได้ผลลัพธ์เหมือนกัน ทำให้เราสามารถเลือกใช้ฐานข้อมูล ชนิดใดก็ได้โดยไม่ติดขัดกับฐานข้อมูลใดฐานข้อมูลหนึ่ง นอกจากนี้แล้ว SQL ยังเป็นชื่อโปรแกรมฐานข้อมูล ซึ่งโปรแกรม SQL เป็นโปรแกรมฐานข้อมูลที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน มีประสิทธิภาพการทำงานสูง สามารถทำงานที่ซับซ้อนได้โดยใช้คำสั่งเพียงไม่กี่คำสั่ง โปรแกรม SQL จึงเหมาะที่จะใช้กับระบบฐานข้อมูลเชิงสัมพันธ์ และเป็นภาษาหนึ่ง ซึ่งแบ่งการทำงานได้เป็น 4 ประเภท ดังนี้

1. Select query ใช้สำหรับดึงข้อมูลที่ต้องการ
2. Update query ใช้สำหรับแก้ไขข้อมูล

3. Insert query ใช้สำหรับการเพิ่มข้อมูล

4. Delete query ใช้สำหรับลบข้อมูลออกไป

ปัจจุบันมีซอฟต์แวร์ระบบจัดการฐานข้อมูล (DBMS) ที่สนับสนุนการใช้คำสั่ง SQL เช่น Oracle , DB2, MS-SQL, MS-Access

นอกจากนี้ภาษา SQL ถูกนำมาใช้เขียนร่วมกับโปรแกรมภาษาต่างๆ เช่น ภาษา C/C++ , Visual Basic และ Java

ประโยชน์ของภาษา SQL

1. สร้างฐานข้อมูลและ ตาราง
2. สนับสนุนการจัดการฐานข้อมูล ซึ่งประกอบด้วย การเพิ่ม การปรับปรุง และการลบข้อมูล
3. สนับสนุนการเรียกใช้หรือ ค้นหาข้อมูล

ประเภทของคำสั่งภาษา SQL

1. ภาษานิยามข้อมูล(Data Definition Language : DDL) เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล กำหนดโครงสร้างข้อมูลว่ามี Attribute ใด ชนิดของข้อมูล รวมทั้งการเปลี่ยนแปลงตาราง และการสร้างดัชนี คำสั่ง : CREATE,DROP,ALTER

2. ภาษาจัดการข้อมูล (Data Manipulation Language: DML) เป็นคำสั่งที่ใช้ในการเรียกใช้ เพิ่ม ลบ และเปลี่ยนแปลงข้อมูลในตาราง คำสั่ง: SELECT, INSERT, UPDATE, DELETE

3. ภาษาควบคุมข้อมูล (Data Control Language: DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิการอนุญาต หรือ ยกเลิก การเข้าถึงฐานข้อมูล เพื่อป้องกันความปลอดภัยของฐานข้อมูล คำสั่ง: GRANT, REVOKE

2.2 เทคโนโลยี

2.2.1 Angular2

Angular2 คือ แพลตฟอร์มการพัฒนาแอปพลิเคชันบนมือถือ เดสก์ท็อป และเว็บ โดยรันบนพื้นฐานภาษา Typescript ซึ่งมีความคล้ายคลึงกับภาษา Javascript และ Angular2 ทำงานได้หลายแพลตฟอร์ม



ภาพที่ 2.2 ภาพโลโก้ของ Angular 2

สถาปัตยกรรมแอปพลิเคชันของ Angular2

โมดูล คือ แนวคิดการรวมข้อมูลเป็นกลุ่ม เหมือนการแปะป้ายใส่กล่องไว้ เพื่อแยกประเภทเพื่อให้ง่ายต่อการเข้าถึง และ ดูแลรักษาได้ง่ายขึ้น หรือสรุปง่ายๆ ก็คือกลุ่มข้อมูลที่มีความเกี่ยวข้องกัน

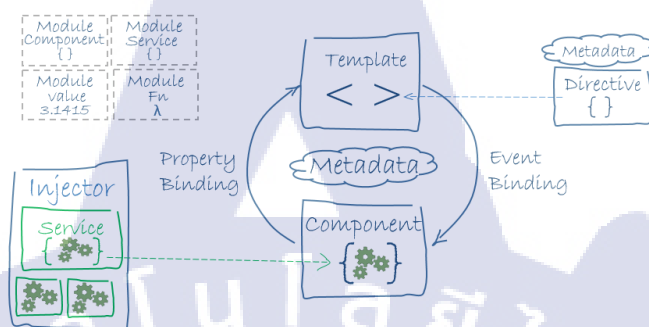
ไลบรารี คือ โมดูลที่มีความสามารถพิเศษเพื่อให้โมดูลอื่นๆ ใช้งาน หรือก็คือเครื่องมือที่ทำให้สามารถสร้างความสามารถต่างๆ แอปพลิเคชันนั่นเอง

คอมโพเนน คือ สิ่งที่ควบคุมการแสดงผลทางหน้าจอโดยส่วนนี้จะสัมพันธ์กับส่วน เทมเพลต และ เซอร์วิส

เทมเพลต คือ ตัวที่แสดงผลเองเลยโดยมี คอมโพเนนควบคุม

เมตาดาต้า คือ สิ่งที่ใช้บอกอธิบายคลาสว่ามีการตั้งค่าอย่างไรบ้าง และทำอะไรบ้าง

เซอว์วิส คือส่วนที่เป็นฟังก์ชัน เกิดขึ้นมาเพื่อทำงานเฉพาะทาง เช่น การดึงข้อมูลจากเซอว์เวอร์ เพื่อมาแสดงผลเป็นต้น



ภาพที่ 2.3 ภาพสถาปัตยกรรมของ Angular 2

2.2.2 JavaScript Object Notation (JSON)

JavaScript Object Notation หรือ JSON เป็นฟอร์แมตสำหรับแลกเปลี่ยนข้อมูลคอมพิวเตอร์ ฟอร์แมต JSON นั้นอยู่ในรูปข้อความธรรมดา (Plain Text) ที่ทั้งมนุษย์และโปรแกรมคอมพิวเตอร์สามารถอ่านเข้าใจได้ ซึ่งปัจจุบันนิยมใช้ใน Web Application

โครงสร้างของฟอร์แมต

เจสัน เป็นรูปแบบสายอักขระ (String) ชนิดหนึ่งที่ถูกจัดเก็บในรูปแบบที่สามารถอ่านและเข้าใจได้ง่าย ถ้าจะให้เข้าใจง่าย ๆ ก็คือเจสันเปรียบเสมือนรูปแบบของ อาร์เรย์ (Array) ชนิดหนึ่งที่ใช้รับส่งข้อมูลผ่านแอแจ็กซ์เพราะซึ่งปกติแล้วถ้าเราต้องการรับ-ส่งข้อมูลผ่านแอแจ็กซ์ต่าง ๆ นั้น จะต้องรับ-ส่งมาในรูปแบบของสายอักขระทั้งก่อน และเมื่อฝั่งแอแจ็กซ์ทำการรับค่าที่ทำการส่งค่ากลับมาจากเซิร์ฟเวอร์ ก็จะต้องนำสายอักขระ เหล่านั้นมาตัดตำแหน่งที่ต้องการ เพื่อเอาสายอักขระตัวที่ต้องการมาใช้ แต่สำหรับเจสันแล้ว สามารถรับส่งชุดค่าตัวแปรได้ทั้งฝั่งไคลเอนต์ (Client) และฝั่งเซิร์ฟเวอร์ (Server) โดยทั้ง 2 ฝั่งสามารถทำการเข้ารหัสและถอดรหัสโดยใช้เจสันเอนโค้ด (Json Encode) และ เจสันดีโค้ด (Json

Decode) เพื่ออ่านค่าตัวแปรเหล่านั้น และจะเรียกใช้งานมันได้อย่างไร ซึ่งปกติแล้วจะอยู่ในรูปแบบของอาร์เรย์และสำหรับตัวแปรเจสันนั้นไม่จำกัดแค่รับส่งข้อมูลผ่านเว็บเบราว์เซอร์เท่านั้น แต่ยังสามารถนำเจสันไปประยุกต์กับการรับส่งข้อมูลในรูปแบบอื่น ๆ ได้ เช่นการจับเก็บข้อมูลในรูปแบบของ สายอักขระในข้อความ หรือการรับส่งผ่านตัวให้บริการเว็บไซต์ (Web Service) ก็สามารทำได้เช่นเดียวกัน

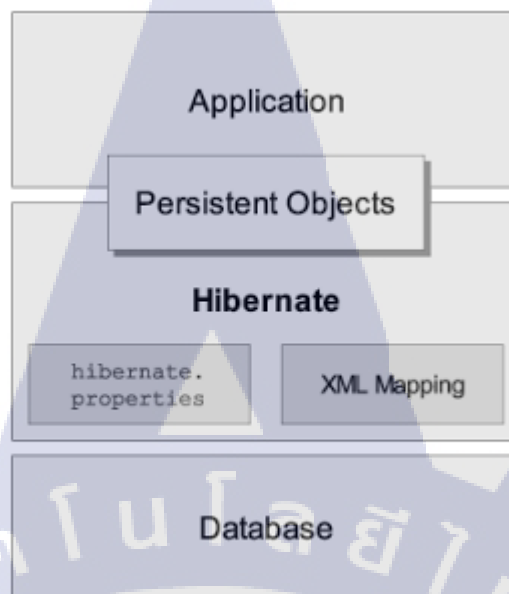
```
{
  "Students": [
    {
      "Name": "Amit Goenka",
      "Major": "Physics"
    },
    {
      "Name": "Smita Pallod",
      "Major": "Chemistry"
    },
    {
      "Name": "Rajeev Sen",
      "Major": "Mathematics"
    }
  ]
}
```

ภาพที่ 2.4 ตัวอย่างโค้ดที่ใช้ Format JSON

2.2.3 Hibernate Framework

Hibernate เป็น Framework Java ใช้ในการจัดการข้อมูลแบบ ORM (Object/Relation Mapping Object กับ ข้อมูลจากฐานข้อมูลแบบอัตโนมัติกลับไปกลับมาได้) เพื่อความสะดวกในการทำงานต่างๆ เช่น การเข้าถึงข้อมูล การเรียกค้นข้อมูล เป็นต้น

โครงสร้าง Hibernate



ภาพที่ 2.5 โครงสร้าง Hibernate

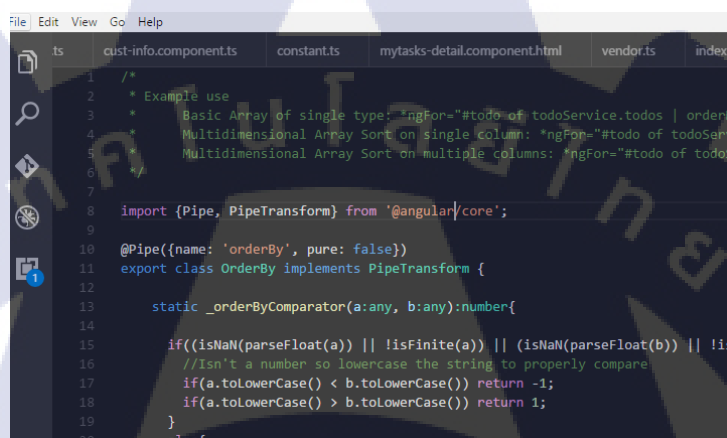
จากภาพที่ 2.2.7.1 เป็นรูปโครงสร้างระดับบนสุดของ Hibernate ซึ่งทำหน้าที่เป็นตัวกลางในการสื่อสารระหว่าง Application กับ Database

ประโยชน์ของ Hibernate

1. มีเวลาในการโฟกัส Business Logic ของโปรแกรมได้อย่างเต็มที่
2. ทำให้การบำรุงรักษาง่ายขึ้นเนื่องจากโค้ดที่เขียนน้อยลง สะอาด และเข้าใจง่าย
3. ไม่ยึดติดกับฐานข้อมูล เราสามารถใช้ฐานข้อมูลได้หลากหลาย
4. มี APIs ที่เรียบง่ายที่ใช้ในการบันทึกและอ่านข้อมูลจากฐานข้อมูลได้โดยตรงผ่าน Java Objects
5. เข้าถึงฐานข้อมูลได้อย่างมีประสิทธิภาพ ทำให้ลดภาระของ Database Server ซึ่งส่งผลให้การเชื่อมต่อเร็วขึ้น

2.2.4 Visual Studio Code

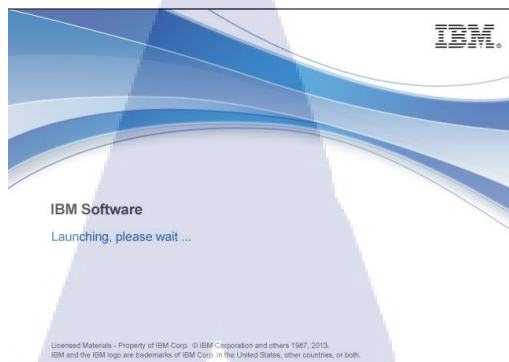
Visual Studio Code เป็นโปรแกรมที่ถูกพัฒนาขึ้นโดยบริษัท Microsoft โดยโปรแกรม Visual Studio Code นี้เป็นโปรแกรมสำหรับแก้ไขและปรับแต่งโค้ดที่ตัดความสามารถของ Visual Studio รุ่นปกติออกไปเหลือเพียง Editor อย่างเดียวซึ่งสามารถทำงานข้ามแพลตฟอร์ม ทั้งบนวินโดวส์ แมค และลินุกซ์ โดยเป็นโปรแกรมแจกฟรีให้ทุกคนสามารถเข้าไปดาวน์โหลดได้



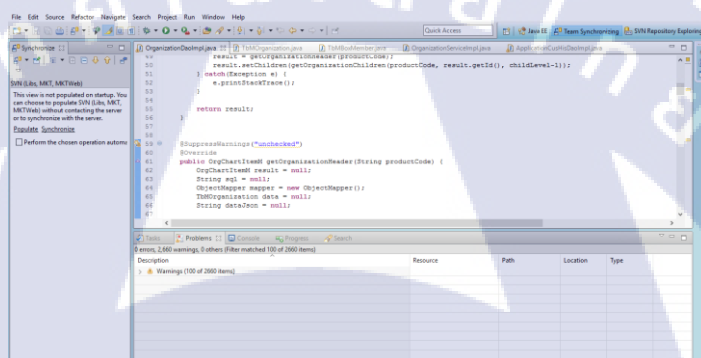
ภาพที่ 2.6 ตัวอย่างหน้าต่างภายในโปรแกรม Visual Studio Code

2.2.5 Rational Application Developer for Web Sphere

Rational Application Developer หรือ RAD เป็นโปรแกรมที่พัฒนาขึ้นมาโดยบริษัท IBM โดยมีพื้นฐานการทำงานมาจากโปรแกรม Eclipse ซึ่งสำหรับใช้พัฒนาเว็บไซต์และโปรแกรมที่สามารถเขียนโดยใช้ภาษา Java นอกจากนี้ยังมีเครื่องมือช่วยเหลือเพื่อให้สามารถทำงานได้ไวขึ้น อาทิ ส่วนของการ Synchronize SVN เพื่อให้สามารถอัปเดตไฟล์ที่พัฒนาเข้าเซิร์ฟเวอร์กลางเพื่อให้สามารถคนอื่นๆสามารถนำไปพัฒนาต่อ ตามที่ได้รับมอบหมาย และสิ่งที่เป็นจุดเด่นอีกอย่างคือ Web Sphere ซึ่งมาพร้อมกับโปรแกรมนี้ โดยมีไว้สำหรับในการใช้เป็นเซิร์ฟเวอร์ทดสอบบนเครื่องผู้พัฒนาได้เลย



ภาพที่ 2.7 รูปหน้าต่างโปรแกรมตอนเริ่มเข้าใช้งาน



ภาพที่ 2.8 รูปตัวอย่างหน้าต่างภายในโปรแกรม

2.2.6 NodeJS

Node.js คือ Cross Platform Runtime Environment สำหรับฝั่ง Server และเป็น Open Source ซึ่งเขียนด้วยภาษา JavaScript สรุปรวมๆ NodeJS ก็คือ Platform ตัวหนึ่ง ที่เขียนด้วย JavaScript สำหรับเป็น Web Server นั่นเอง ซึ่งภายในโปรเจกต์นี้กระผมจะเน้นใช้ คือส่วนที่เป็น **npm** หรือ Node Package Manager ซึ่งก็คือตัวช่วยที่คอยทำหน้าที่จัดการกับ Package ของ Node ให้เรา อย่างเช่น Application/package หรือ module ต่างๆ ที่มีคนเขียนไว้ ซึ่ง npm นั้นสามารถนำมาใช้ได้อย่างง่ายดาย โดย Package/Module จะถูกติดตั้งไว้ในโฟลเดอร์ `node_modules` ภายในโปรเจกต์ของเรา

```
npm install PACKAGE_NAME
```

ภาพที่ 2.9 ตัวอย่างคำสั่งสำหรับการนำ Package หรือ Module คนอื่นมาใช้



บทที่ 3

แผนปฏิบัติงานและขั้นตอนการดำเนินงาน

3.1 แผนงานและระยะเวลาในการปฏิบัติงาน

หัวข้องาน	เดือนที่ 1	เดือนที่ 2	เดือนที่ 3	เดือนที่ 4
Expense Tab				
CAP : UI Screen	■	■		
CAP : Front Controller	■	■		
CAP : Date Access Object		■	■	
CAP : Backend Validate			■	
CAP : Service		■	■	
CAP : Entity			■	■
MKT : Screen			■	■
MKT : Front Controller				■
MKT : Date Access Object				■
MKT : Backend Validate				■

ตารางที่ 3.1 ตารางการแผนการทำงานและปฏิบัติจริง

3.2 รายละเอียดของโครงการที่ได้รับมอบหมาย

โครงการชิ้นนี้เป็นโครงการที่เกิดขึ้นจากที่ธนาคารมีชื่อแห่งหนึ่งมาว่าจ้างให้สร้างระบบ และพัฒนาระบบระบบอนุมัติสินเชื่อเช่าซื้อรถยนต์และสินเชื่อธุรกิจขนาดย่อมในรูปแบบของ Web Application และได้มอบหมายให้นักศึกษาสหกิจเข้าร่วมการพัฒนา ระบบกับทางบริษัท ที่บริษัทเป็นผู้รับผิดชอบพัฒนา โดยได้มอบหมายส่วนของงาน การนำเข้าข้อมูลมาตรวจสอบต้นขั้วรถยนต์และ ค้นหา/แก้ไข/นำออกข้อมูล และสามารถแบ่งส่วนของการทำงานได้ดังนี้

3.2.1 ส่วน Front End

Front-End คือ ส่วนที่ติดต่อกับผู้ใช้ (User Interface) เปรียบได้กับหน้าบ้านหรือเรียกว่า เป็นส่วนงานด้านการทำงานในฝั่งของ Browser นั้นเอง โดยการทำงานในจะใช้หลักการของ Web Server ที่ทำให้เราให้เห็นถึงหน้าเว็บต่าง ๆ นั้น มีขั้นตอนในการทำงานโดยเราจะทำการกรอก URL ของเว็บที่เราต้องการเข้าไปใช้งานเช่น localhost:3000/login กรอกลงบริเวณช่อง Address ของ Browser ที่เราใช้เช่น Chrome โปรแกรมเหล่านี้เราเรียกว่า Client ที่หน้าที่ส่งคำขอหรือ Request ไปยัง ที่ Server ที่กรอกที่อยู่ลงไป ก็คือ localhost:3000/login ตัว Browser ก็จะทำการส่ง Request ไปยัง Server ที่อยู่ของ Web ที่เรากรอกไว้ เมื่อ Server ได้รับคำขอหรือ Request ของเรา Server จะทำการประมวลผลตามคำขอที่เราส่งไปว่าเราต้องการอะไรบ้าง เมื่อประมวลผลเสร็จ Server จะส่ง Code HTML และ JavaScript กลับมาให้เราและแสดง Interface ขึ้นบน Browser ที่เราใช้

3.2.2 ส่วน Back End

Back-End คือ ระบบจัดการเว็บไซต์ เปรียบได้กับหลังบ้าน ถือเป็นส่วนงานของการจัดการ เนื้อหาต่างของเว็บไซต์ รวมถึงโครงสร้างของเว็บไซต์ด้วยเช่นกัน ส่วนนี้ User จะไม่สามารถเข้ามา เห็นได้ซึ่งแตกต่างกับ Front-End หรือหน้าบ้านที่จะแสดง User Interface ให้ User ได้เห็น ดังนั้น Back-End หรือหลังบ้าน จะเป็นหน้าที่ที่ผู้พัฒนาจะต้องเขียน Code ขึ้นมาเพื่อใช้ในการจัดการเนื้อหา ต่างๆที่จะแสดงขึ้นบนหน้า Web Browser โดยในส่วนนี้การปฏิบัติงานจะใช้เป็น Web Service

3.2.3 Database

Database (ฐานข้อมูล) เป็นกลุ่มของข้อมูลซึ่งมีจำนวนมหาศาล ที่มีการจัดการข้อมูลเพื่อให้เข้าถึง และปรับปรุงได้ง่าย ฐานข้อมูลแบ่งออกเป็นหลายประเภท แต่ที่นิยมใช้กันคือ Relation database ซึ่งเป็นฐานข้อมูลที่เก็บในรูปแบบของตาราง ทำให้การเข้าถึงและจัดการข้อมูลได้หลายวิธี

3.3 ขั้นตอนการดำเนินงาน

3.3.1 เตรียม Environment

เนื่องจากบริษัทใช้โปรแกรม RAD (Rational Application Developer) เป็นโปรแกรมหลักที่ใช้งานภายในบริษัท จึงมีเจ้าหน้าที่ของบริษัทมาทำการติดตั้ง Config โปรแกรมให้ที่เครื่องคอมพิวเตอร์ของเราเพื่อให้ใช้งานได้สะดวกและถูกต้อง นอกจากนี้ยังมีโปรแกรมอื่นๆที่ใช้ในการทำงานอีกเช่น Hoo Win Tails โปรแกรมที่ใช้ในการดู Log ต่างๆ , Visual studio Code โปรแกรมที่ใช้ในการเขียน HTML และ TypeScript , Oracle SQL โปรแกรมใช้จัดการฐานข้อมูลของโครงการ โดยโปรแกรมเหล่านี้ทางบริษัทจะจัดการเอาไฟล์ติดตั้งวางไว้ที่ Shared Document ของบริษัทโดยให้เข้าไปนำมาติดตั้งในเครื่องของเราและจะมีเจ้าหน้าที่มาสอนการใช้งานโปรแกรมต่อไป

3.3.2 ศึกษาใช้งานโปรแกรมที่ได้รับ และ Coding rule

ศึกษารูปแบบของ Code ที่ใช้ในโครงการของบริษัท เพื่อที่จะสามารถเขียนออกมาให้เป็นรูปแบบเดียวกันหรือใกล้เคียงกันมากที่สุดเท่าที่จะเป็นไปได้ เพราะจะได้สะดวกในการรับช่วงต่อของพนักงานคนอื่นจะเข้ามาทำงานต่อจากเราหรือเอาของเราไปพัฒนาต่อ

3.3.3 สร้าง หน้าเว็บ ตามต้นแบบที่รับมา

สร้างหน้าเว็บไซต์โดยใช้โปรแกรม Visual Studio Code ใช้ภาษา HTML และ JavaScript ในการเขียนรูปร่างหน้าตาของหน้าเว็บไซต์ วางโครงสร้างให้ถูกต้องตามต้นแบบที่รับมา ซึ่งในขั้นตอนนี้เป็นเพียงการวาดหน้าตาของเว็บไซต์เพียงอย่างเดียว ยังไม่สามารถใช้งานได้เพราะยังไม่ได้เขียนคำสั่งลงไปที่ปุ่มต่างๆ หรือส่วนในการแสดงข้อมูลบนหน้าเว็บไซต์ ไฟล์ที่ได้รับจากขั้นตอนนี้จะเป็นไฟล์ .html และ .ts ที่จะต้องนำไปเขียนคำสั่งต่างๆมาใส่ในส่วนต่างๆต่อไป

```

1 <section class="content">
2 <div class="card kk">
3
4 <div class="card-header">ค้นหา</div>
5
6
7 <div class="card-block" id="text">
8
9 <div class="row">
10 <div class="col-md-6"><input type="text" name="import-datetime-from" label="วันที่นำเข้า(จาก)" [(ngModel)]="importDateFromSearch" [cssClass]="cssClass"></div>
11 <div class="col-md-6"><input type="text" name="import-datetime-to" label="วันที่นำเข้า(ถึง)" [(ngModel)]="importDateToSearch" [cssClass]="cssClass"></div>
12 </div>
13 <div class="row">
14 <div class="col-md-6"><input type="text" name="import-user" label="ผู้นำเข้า" [(ngModel)]="importUserSearch" [cssClass]="cssClass"></div>
15 <div class="col-md-6"><input type="text" name="import-status" label="สถานะ" [(ngModel)]="importStatusSearch" [cssClass]="cssClass" [master]="importStatus"></div>
16 </div>
17 <div class="text-md-center">
18 <button type="button" class="btn btn-primary" (click)="search()">ค้นหา</button>
19 <button type="button" class="btn btn-primary" (click)="clearFilter()">ล้างข้อมูล</button>
20 </div>
21 </div>
22 </div>
23 <div class="card kk">
24 <div class="card-header" style="text-align:left;">รายการนำเข้าผลการตรวจสอบ</div>
25 <div class="card-block">
26 <div class="row">
27 <div class="col-xs-8">
28 <button type="button" class="btn btn-primary" (click)="openUploader()">นำเข้า</button>
29 <button type="button" class="btn btn-primary" (click)="saveImportFile()">นำเข้าระบบ</button>
30 </div>
31 <div class="col-xs-4">
32 <span class="pull-right">
33 <button type="button" class="btn btn-primary" (click)="deleteImportFile()">ลบไฟล์</button>
34 </span>
35 </div>
36 </div>
37 <!--rowTable-->
38 <div class="table-responsive">
39 <table class="table table-hover table-header-nowrap">

```

ภาพที่ 3.1 ตัวอย่างโค้ด HTML5 ที่ใช้สร้างหน้าเว็บ

3.3.4 สร้าง Web Service ต่างๆให้กับไฟล์ .html และ .ts

สร้าง Web Service โดยใช้โปรแกรม RAD (Rational Application Developer) ในขั้นตอนนี้จะทำการสร้าง Function เพื่อไปใช้ที่ Front-End โดยใช้ภาษา Java ในการเขียนคำสั่งขึ้นจัดการข้อมูล ประมวลผลข้อมูลที่รับมาจากฐานข้อมูลเพื่อให้หน้าเว็บแสดงข้อมูลที่เรต้องการ

```

106 + }
107
108 + updateImportFile(attachmentDocument:AttachDocument) {
109 +   console.log('--updateImportFile--');
110 +   let attachmentDocumentId = attachmentDocument.attachDocumentId;
111 +   let importFileAdd: ImportFile = new ImportFile();
112 +   importFileAdd.createUser = this.authService.getCurrentUser().username;
113 +   importFileAdd.fileName = attachmentDocument.fileName;
114 +   importFileAdd.filePath = attachmentDocument.filePath;
115 +   importFileAdd.importCategory = ("LCD0");
116 +   importFileAdd.importUser = this.authService.getCurrentUser().username;
117 +   importFileAdd.isActive = 'Y';
118 +   importFileAdd.updateUser = this.authService.getCurrentUser().username;
119 +   this.lcdoDataImportService.saveImportFile(importFileAdd).
120 +   subscribe((importFileResult: ImportFile) => {
121 +     console.log('LcdoImportResult = ', importFileResult);
122 +     //Object.assign(this.carshieldImportList,carshieldImportResult);
123 +     console.debug('importFileResult',importFileResult);
124 +     this.importFileList.push(importFileResult);
125 +     //this.carshieldImportList = carshieldImportList;
126 +     //this.appHistoryModel = applicationHistores[0];
127 +   });
128
129 + }
130
131 + saveImportFile() {
132 +   let dataImport: number[] = [];
133 +   this.importFileList.forEach(
134 +     importFileResult => {
135 +       if (importFileResult.checked) {
136 +         console.debug('importFileResult.importFileId = ',importFileResult.importFileId);
137 +         dataImport.push(importFileResult.importFileId);
138 +         importFileResult.importStatus = "P";
139 +       }
140 +     }
141 +   );
142 +   this.lcdoDataImportService.importResult(dataImport).
143 +   subscribe((genericResponse: GenericResponse) => {
144 +     console.debug('genericResponse = ',genericResponse);
145 +     console.log('genericResponse = ', genericResponse);

```

ภาพที่ 3.2 ตัวอย่างโค้ด TypeScript ของ Front-End

3.3.5 นำข้อมูลลง Database

นำข้อมูลลงในฐานข้อมูลเพื่อใช้ในการทดสอบ Service ที่เราเขียนขึ้นมาได้นั้นสามารถใช้งานได้ตรงตามความต้องการของส่วนงานที่ได้รับมอบหมาย

3.3.6 เชื่อมต่อ Web Service ให้อัปโหลดไฟล์ .ts

เชื่อมต่อ Web service โดยใช้ Ajax เรียก URL และข้อมูลที่ได้มาจาก Function ที่เราเขียนขึ้นมาใส่ลงใน TAG ที่เราเขียนไว้ที่ Front-End ซึ่งจะมี Service ที่เขียนขึ้นในฝั่งของ Front-End เป็นตัวเชื่อมที่จะคอยเอาข้อมูลมาจัดการเพื่อนำไปแสดงบนหน้า Browser

```

14
15 - constructor(
16 -   private configService: ConfigService,
17 -   private http: Http
18 - ) {
19 -   console.debug('Lcdo Import Service : construct...');
20 - }
21 - private hostname: string = this.configService.get('service.hostname');
22 - private url = {
23 -   'getLcdoDataImportByImportFileId': this.hostname + 'datamart',
24 -   'getLcdoDataImportByValidationId': this.hostname + 'datamart/by/validation',
25 -   'getLcdoDataImportByImportDataId': this.hostname + 'datamart/by/importFile',
26 -   'saveImportFile': this.hostname + 'lcdoDataImport',
27 -   'searchImportFile': this.hostname + 'lcdoDataImport/search',
28 -   'importResult': this.hostname + 'lcdoDataImport/importResult',
29 -   'exportResult': this.hostname + 'lcdoDataImport/export',
30 -   'deleteImportFile': this.hostname + 'datamart/delete-datamart',
31 -   'deleteImportFileList': this.hostname + 'lcdoDataImport/deletebylist'
32 - };
33
34 - private importFileModel: ImportFile[];
35
36 - private handleError(error: any) {
37 -   let errMsg = (error.message) ? error.message :
38 -     error.status ? `${error.status} - ${error.statusText}` : 'Server error';
39 -   console.error(errMsg); // log to console instead
40 -   return Observable.throw(errMsg);
41 - }
42
43 - public getLcdoDataImportByImportFileId(id: number): Observable<ImportFile[]> {
44 -   let url = this.url.getLcdoDataImportByImportFileId + "/" + id;
45
46 -   return this.http.get(url)
47 -     .map((res: Response) => {
48 -       // Mapping response data
49 -       let body = res.json();
50 -       this.importFileModel = body;
51 -       return body || {};
52 -     })
53 -     .catch(this.handleError);

```

ภาพที่ 3.3 ตัวอย่างโค้ดของไฟล์ Controller ที่ใช้สำหรับเรียกฟังก์ชันจาก Back-End

3.3.7 นำโค้ดทั้งหมดอัปข้อมูลใน Server

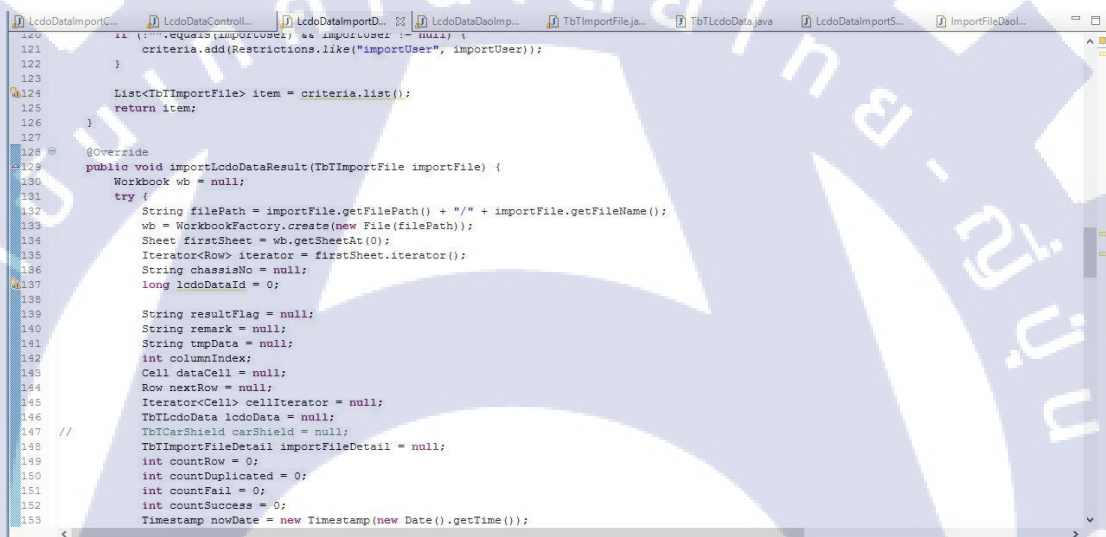
นำไฟล์ทั้งหมดอัปขึ้น Server ในขั้นตอนนี้จะต้องมีความละเอียดเป็นพิเศษ ก่อนจะทำการอัปโค้ดขึ้นบน Server นั้นจะต้องมีการตรวจสอบโค้ดของเราด้วยตัวเองว่าโค้ดของเราไม่เสียหาย ซึ่งจะส่งผลกระทบต่อ Deploy จะทำให้ไม่สามารถ Deploy ได้ ดังนั้นการอัปโค้ดทุกครั้งจะต้องตรวจสอบให้ดีกว่านั้นว่าโค้ดของเราใช้งานได้และทำการใส่ comment ถึงในส่วนที่เราอัปขึ้น Server ว่าเราอัปในส่วนไหนขึ้น Server ไป

บทที่ 4

ผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ

4.1 ขั้นตอนและผลการดำเนินงาน

1. ศึกษาความต้องการของระบบ (Requirement) โครงสร้าง (Structure) กระบวนการทำงาน (Business Process Flow)
2. พัฒนาระบบตามเอกสารต้นแบบ (Prototype) ที่มีการออกแบบและวิเคราะห์ไว้
3. สร้าง Web service เพื่อรองรับกระบวนการทำงานของระบบ และ ใส่ข้อมูลลงในฐานข้อมูล (Database)



```
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

ภาพที่ 4.1 ตัวอย่างโค้ดฟังก์ชันที่ใช้สำหรับการนำออกข้อมูลมาเป็นไฟล์ Excel

```

216
217 logger.debug("chassisNo >> " + chassisNo + " resultFlag >> " + resultFlag + " remark >> " + remark);
218
219 if ((("Y".equalsIgnoreCase(resultFlag) || "N".equalsIgnoreCase(resultFlag) || "C".equalsIgnoreCase(resultFlag) || resultFlag == null)
220     lcdData = lcdDataDao.getLcdDataByChassisNo(chassisNo);
221     logger.debug("ChassisNo >> " + chassisNo);
222     if (lcdData == null) {
223         countFail++;
224         importFileDetail.setImportType("LCDO_IMPORT_FAIL");
225     } else {
226         logger.debug("LcdDataResult ***** >> " + chassisNo);
227         lcdData = lcdDataDao.getLcdDataByChassisNo(chassisNo);
228         logger.debug("LcdDataResult >> " + lcdData);
229         if (lcdData != null && "Y".equalsIgnoreCase(lcdData.getResultFlag()) || "N".equalsIgnoreCase(lcdData.getResultFlag()))
230             logger.debug(" <<<< LCDO_IMPORT_DUPLICATED >>>> ");
231             importFileDetail.setImportType("LCDO_IMPORT_DUPLICATED");
232             countDuplicated++;
233     }
234     } else if (lcdData != null && lcdData.getResultFlag() == null){
235         logger.debug(" <<<< LCDO_IMPORT_SAVE >>>> ");
236         logger.debug("Insert LcdDataResult : chassisNo >> " + chassisNo + " ResultFlag >> " + resultFlag );
237         lcdData.setRemark(remark);
238         lcdData.setResultFlag(resultFlag);
239         this.save(importFile);
240         lcdDataDao.save(lcdData);
241         logger.debug(" <<<<<<<< SAVE >>>>>>>> " + lcdData);
242         countSuccess++;
243         importFileDetail.setImportType("LCDO_IMPORT_SUCCESS");
244         logger.debug("LCDO_IMPORT_SUCCESS");
245     }
246 }
247
248 } else {
249

```

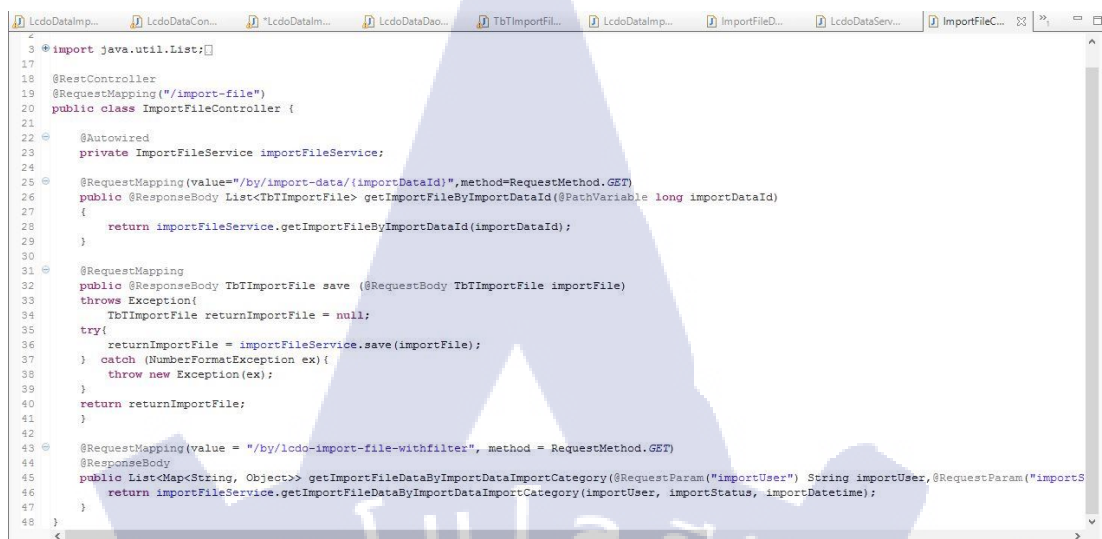
ภาพที่ 4.2 ตัวอย่างโค้ดฟังก์ชันที่ใช้สำหรับประมวลผลตามเงื่อนไข

```

91
92 @SuppressWarnings("unchecked")
93 @Override
94 @Transactional
95 public List<Map<String, Object>> getImportFileDataByImportDataImportCategory(String importUser, String importStatus, String importDatetime) {
96     String filter_sql = "";
97
98     if(!importUser.equalsIgnoreCase("null"))
99     {
100         filter_sql = filter_sql.concat(" AND im.IMPORT_USER='" + importUser.toString() + "' ");
101     }
102     if(!importStatus.equalsIgnoreCase("null"))
103     {
104         filter_sql = filter_sql.concat(" AND im.IMPORT_STATUS='" + importStatus.toString() + "' ");
105     }
106     if(!importDatetime.equalsIgnoreCase("null"))
107     {
108         filter_sql = filter_sql.concat(" AND im.IMPORT_DATETIME='" + importDatetime.toString() + "' ");
109     }
110
111     logger.debug("Filter_SQL => " + filter_sql);
112
113     String sql = "SELECT im.FILE_NAME as 'fileName', im.IMPORT_USER as 'importUser', im.TOTAL_RECORD as 'totalRecord', "
114         + "im.SUCCESS_RECORD as 'successRecord', im.DUPLICATE_RECORD as 'duplicateRecord', FAIL_RECORD as 'failRecord', "
115         + "im.IMPORT_DATETIME as 'importDatetime', im.IMPORT_STATUS as 'importStatus' "
116         + "FROM TB_I_IMPORT_FILE im "
117         + "LEFT JOIN TB_C_IMPORT_DATA id ON im.IMPORT_CATEGORY = id.IMPORT_CATEGORY "
118         + "WHERE im.IMPORT_CATEGORY = 'LCDO' " + filter_sql;
119     SQLQuery sqlQuery = getSession().createSQLQuery(sql);
120     sqlQuery.setResultTransformer(Criteria.ALIAS_TO_ENTITY_MAP);
121     return sqlQuery.list();
122 }
123

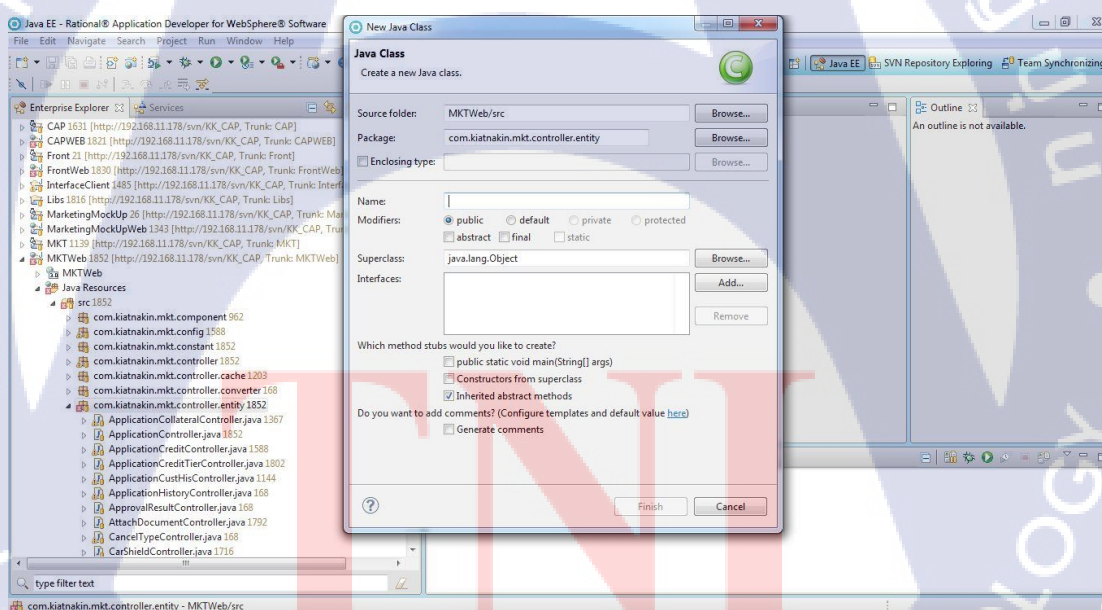
```

ภาพที่ 4.3 ตัวอย่างโค้ดของฟังก์ชันการดึงข้อมูลจากฐานข้อมูล

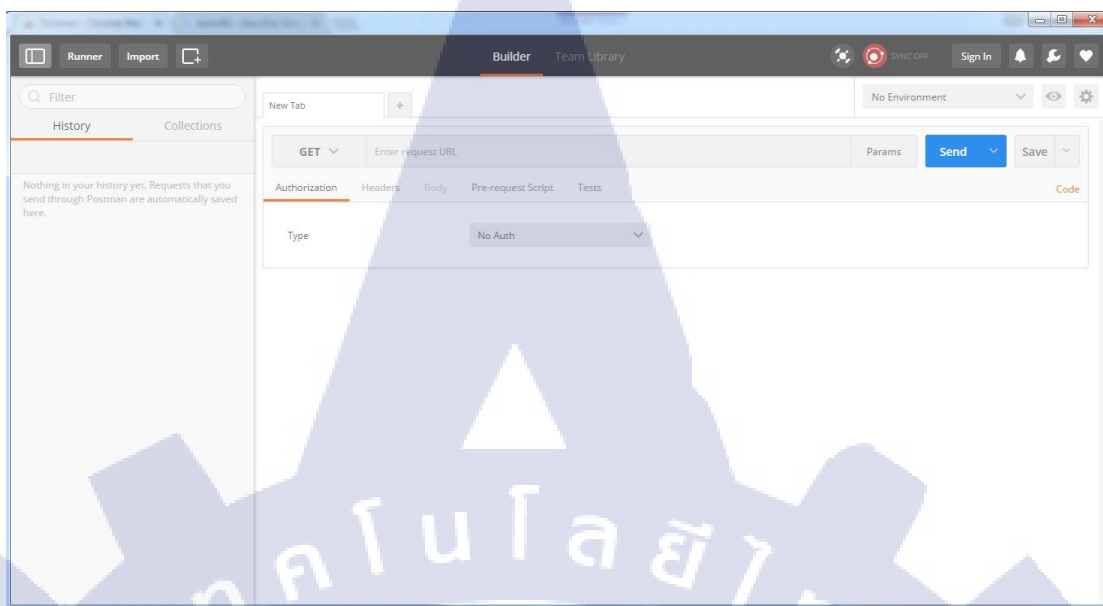


ภาพที่ 4.4 ตัวอย่างโค้ด Controller ของ Back-End ที่ใช้เชื่อมกับ Front-End

4. Generate Web service ให้กับ Front-End

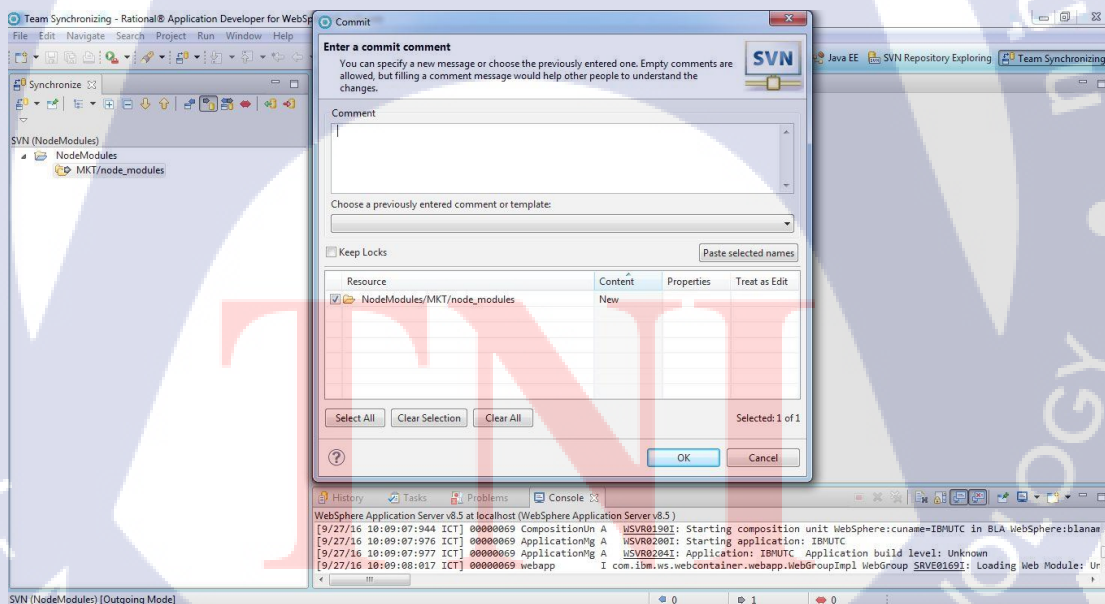


ภาพที่ 4.5 ขั้นตอนการสร้างไฟล์ Create Web Service

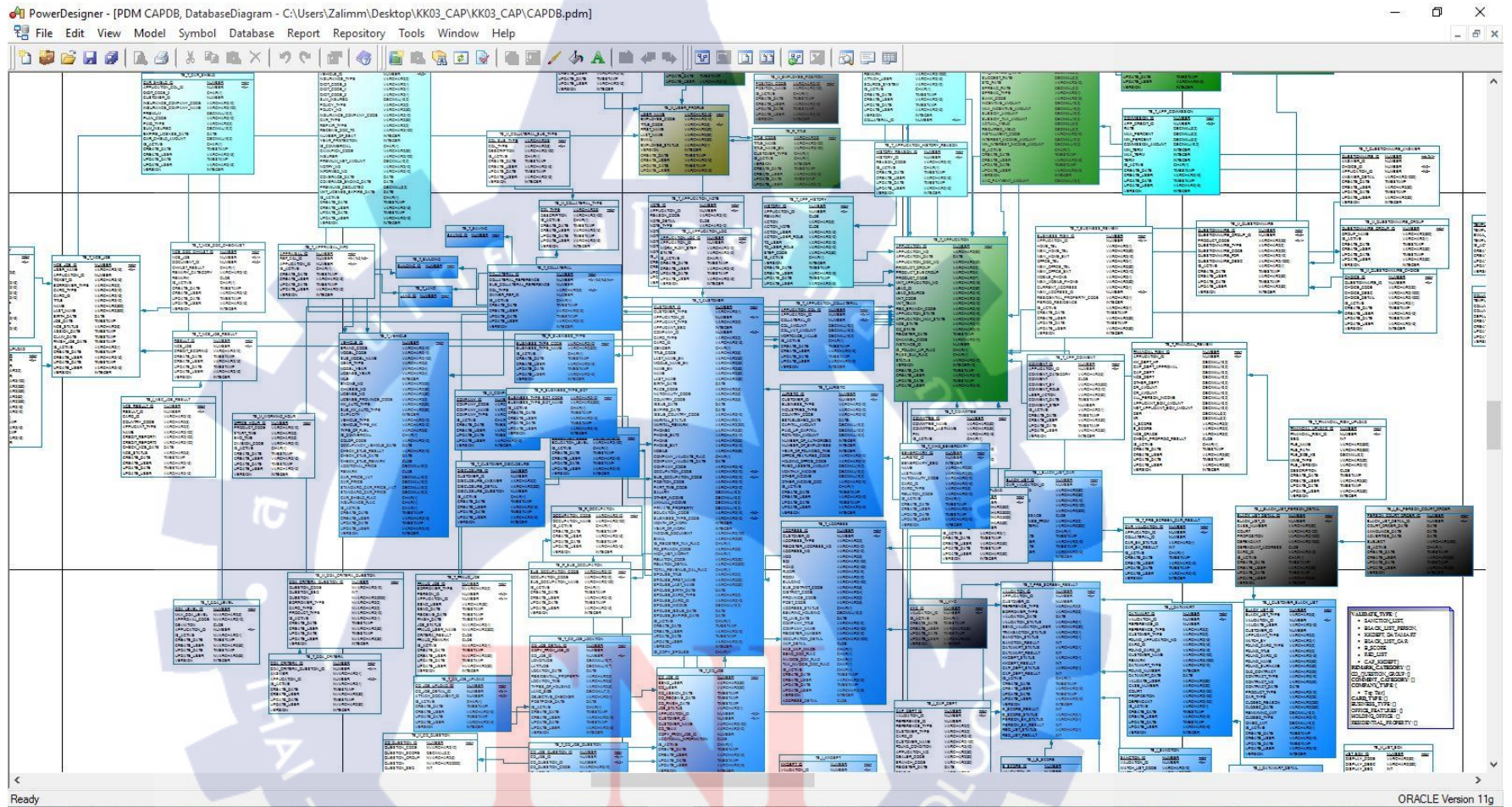


ภาพที่ 4.6 ขั้นตอนการ Test Web Service ด้วย Post Man

5. อัปเดตขึ้นเซิร์ฟเวอร์ (Server) เพื่อรอการ Deploy โดยพนักงานประจำ



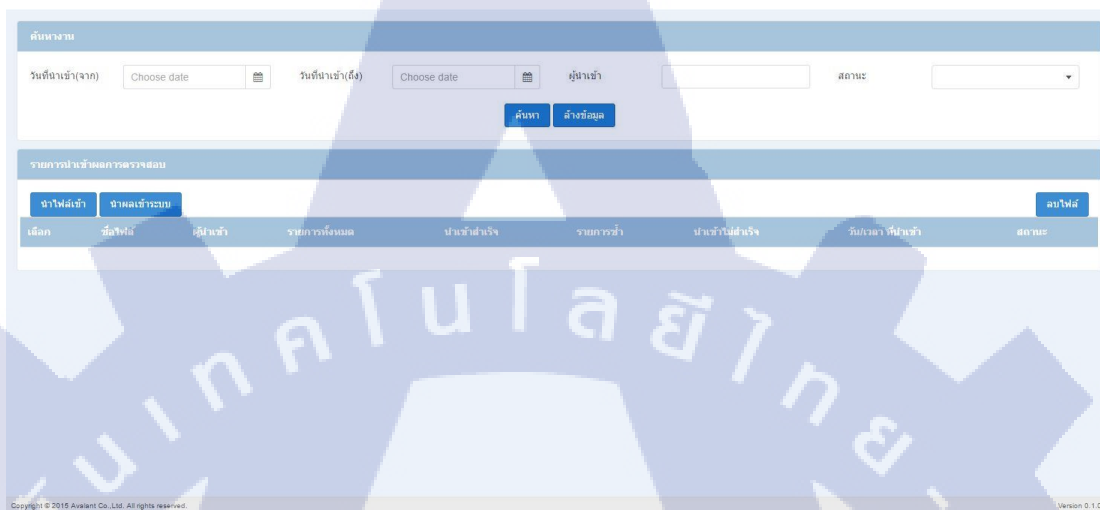
ภาพที่ 4.7 การอัปไฟล์ขึ้น Server



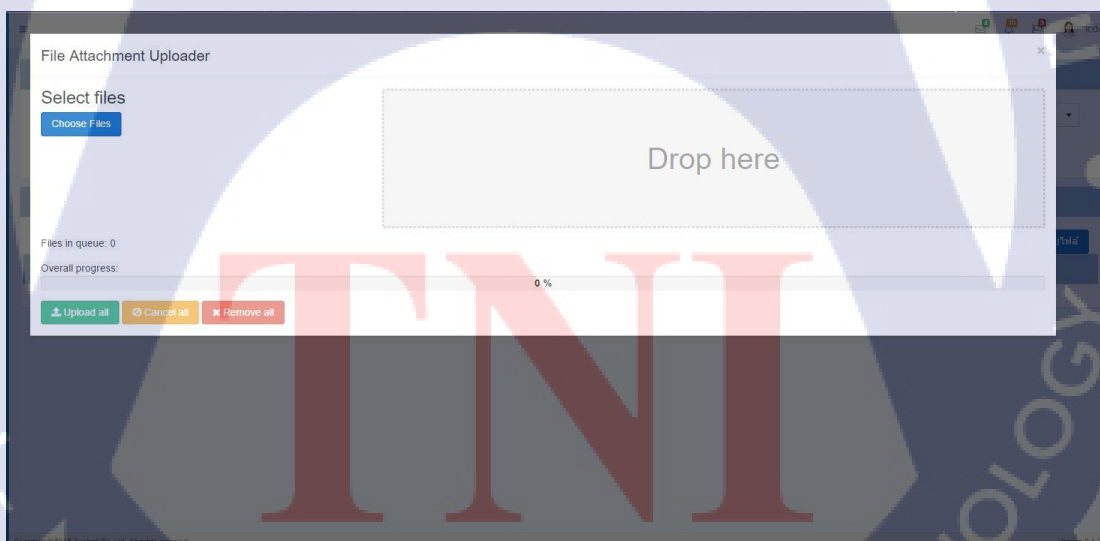
ภาพที่ 4.8 ความสัมพันธ์ของฐานข้อมูล

6. ขั้นตอนการทดสอบระบบด้วยตนเอง (Unit test)

6.1. เข้าใช้งานหน้า Web Application ผ่าน Path บริเวณช่อง Address

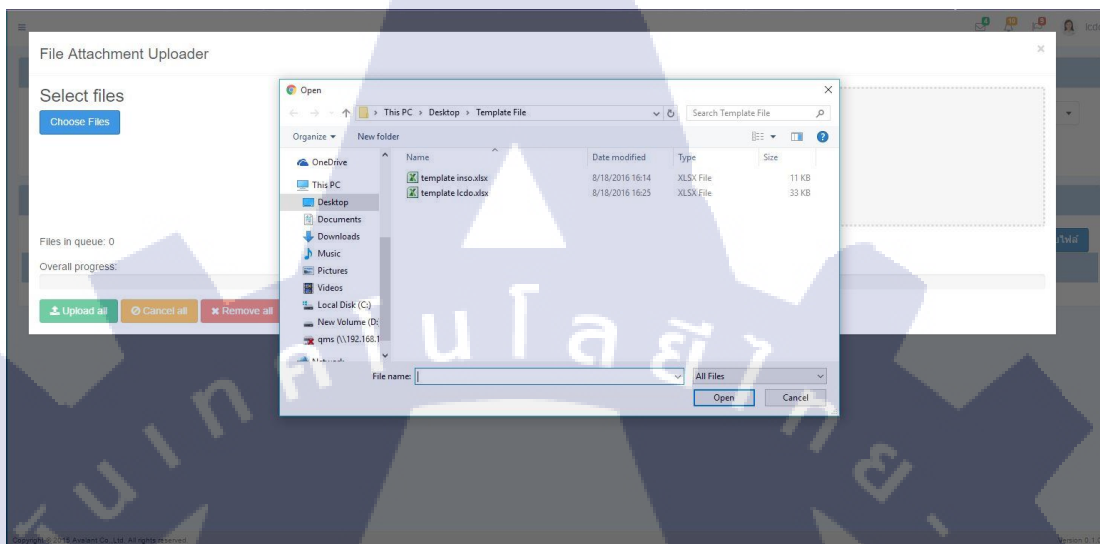


ภาพที่ 4.9 หน้าจอการนำไฟล์เข้าระบบ

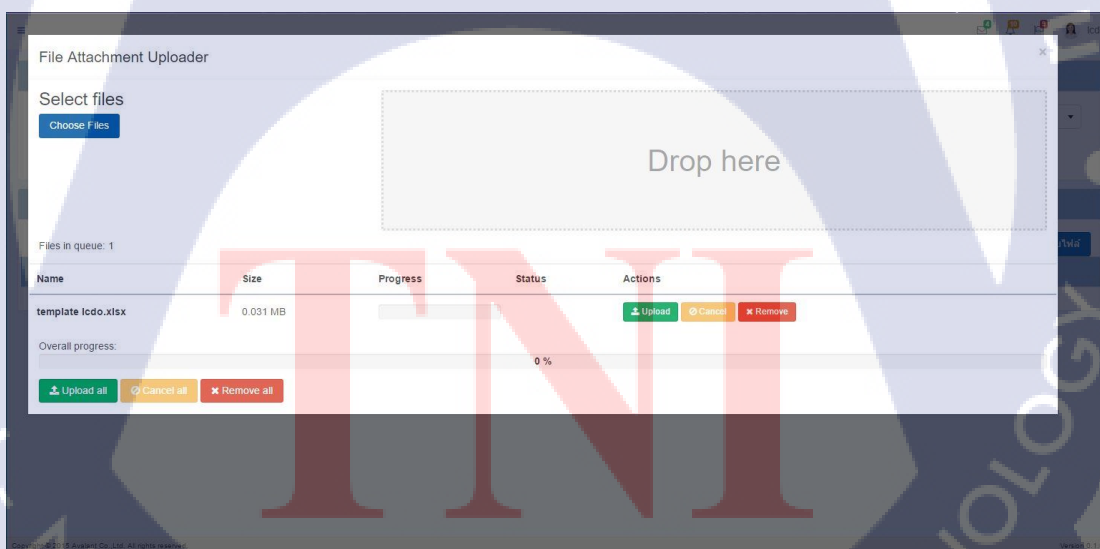


ภาพที่ 4.10 หน้าจอการ Upload ไฟล์

- 6.2. กดปุ่มนำไฟล์เข้าเมื่อกดไปที่ปุ่มแล้วจะปรากฏหน้าต่าง File Attachment Uploader ดังภาพที่ x.xx ในหน้าจะสามารถนำไฟล์เข้าได้ 2 วิธี โดยวิธีที่ 1 เป็นการกดที่ปุ่ม Choose File ดังภาพที่ x.xx อีกวิธีคือลากไฟล์ที่ต้องการมาบริเวณที่ Drop Here

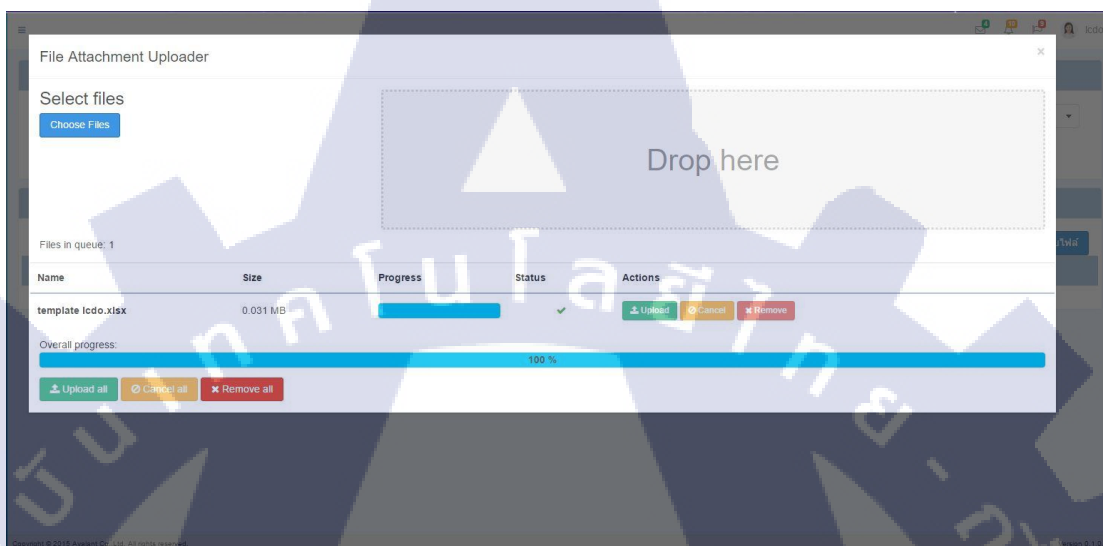


ภาพที่ 4.11 หน้าจอเลือกไฟล์จากฐานข้อมูลของคอมพิวเตอร์

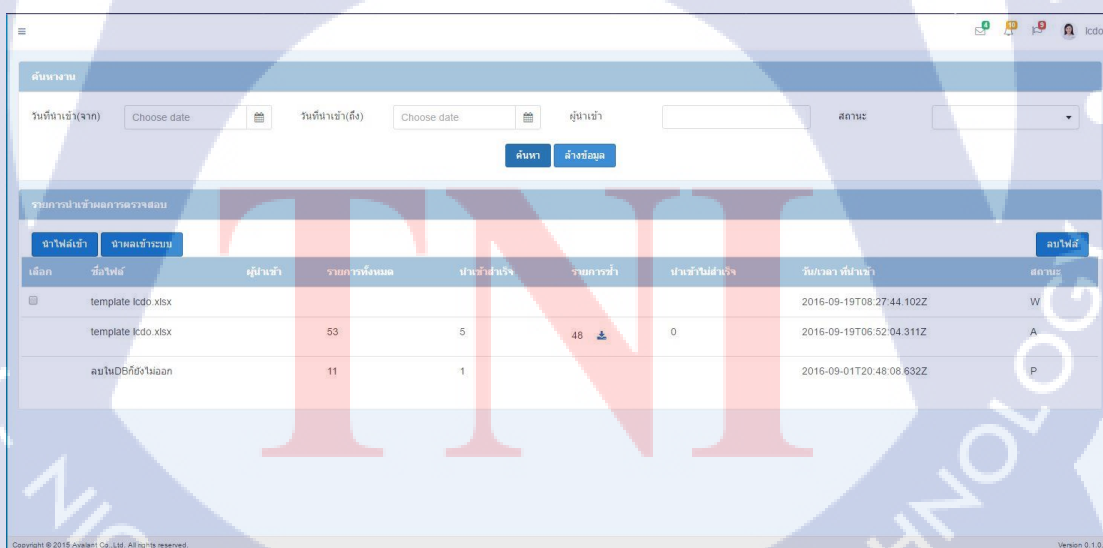


ภาพที่ 4.12 หน้าจอเลือกไฟล์ที่ต้องการและรอการ Upload ขึ้นฐานข้อมูล

- 6.3. เมื่อนำไฟล์เข้ามารอไว้ที่ระบบไฟล์จะถูกนำมาเตรียมไว้รอการ Upload ไฟล์เข้าฐานข้อมูล (Database) ดังภาพ x.xx ถ้าไฟล์ถูกต้องให้กดที่ปุ่ม Upload หากผิดไฟล์ให้กดปุ่ม Cancel ไฟล์ที่ถูกอัปโหลดแล้ว Status ของไฟล์จะถูกใส่เครื่องหมายถูกต้อง ดังภาพที่ x.xx

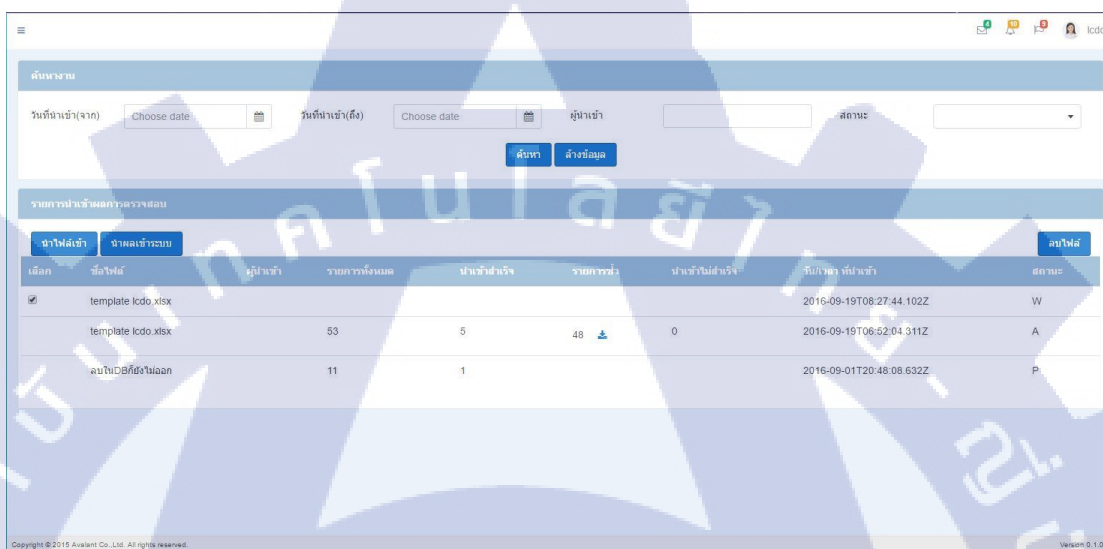


ภาพที่ 4.13 หน้าจอ Upload ไฟล์ลงฐานข้อมูลเสร็จสิ้น



ภาพที่ 4.14 แสดงตำแหน่งของไฟล์เมื่อถูกอัปโหลดแล้ว

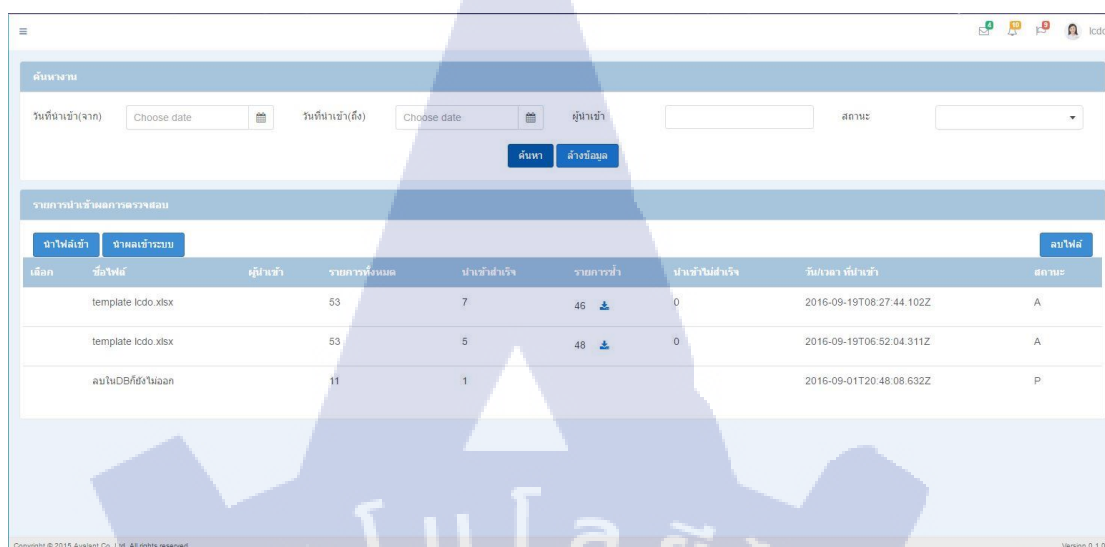
- 6.4. จากนั้นไฟล์ที่ถูก Upload แล้วจะปรากฏขึ้นที่ตารางด้านล่าง และปรากฏช่อง Check box บริเวณด้านหน้าสุดดังภาพที่ x.xx ในขั้นตอนนี้เราจะพบว่ามีไฟล์ที่ไม่มีช่อง Check box อยู่ด้วยนั่นหมายความว่าไฟล์นั้นได้ถูกนำเข้าระบบไปแล้วสถานะของไฟล์จะถูกเปลี่ยนไปและไม่สามารถกระทำการใดๆกับไฟล์ได้



ภาพที่ 4.15 แสดงการเลือกไฟล์ที่ต้องการ



ภาพที่ 4.16 แสดงไฟล์ที่ถูกเปลี่ยนสถานะเป็น “P” เมื่อถูกนำเข้าระบบ



เลือก	ชื่อไฟล์	ผู้ใช้งาน	รายการที่ยอม	นำเข้าสำเร็จ	รายการสำเร็จ	นำเข้าสำเร็จ	เวลาที่นำเข้า	สถานะ
	template lcoo.xlsx		53	7	46	0	2016-09-19T08:27:44.102Z	A
	template lcoo.xlsx		53	5	48	0	2016-09-19T06:52:04.311Z	A
	ลบในDBที่ยังไม่อ่าน		11	1			2016-09-01T20:48:08.632Z	P

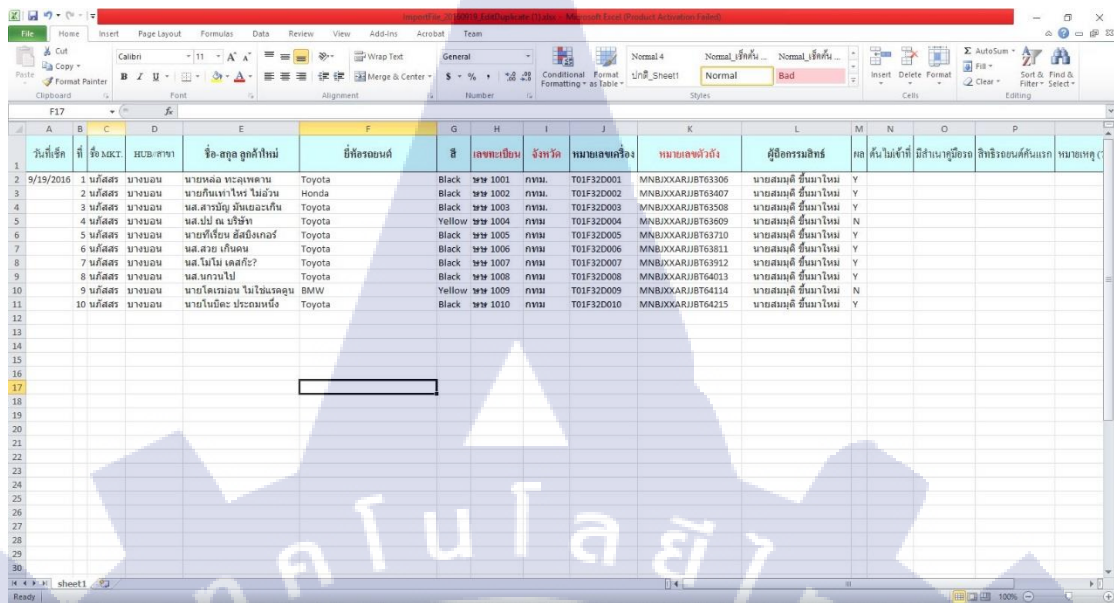
ภาพที่ .17 แสดงไฟล์ที่ถูกเปลี่ยนสถานะเป็น “A” เมื่อประมวลผลเสร็จแล้ว

6.5. จากนั้นไฟล์เลือกไฟล์ที่เราต้องการจะประมวลผล แล้วกดที่ปุ่มนำเข้าไฟล์ในระบบ ในขั้นตอนนี้ระบบจะทำการประมวลผลข้อมูลภายในไฟล์ที่เรานำเข้า ไปตามเงื่อนไขต่างๆที่เราเขียนไว้ที่ Back-End เมื่อระบบประมวลผลอยู่ Status ของไฟล์จะถูกเปลี่ยนเป็น “P” หมายถึง ประมวลผลอยู่ ดังภาพที่ x.xx เมื่อระบบประมวลผลข้อมูลเสร็จสิ้นแล้ว Status ของไฟล์จะถูกเปลี่ยนเป็น “A” หมายถึงเสร็จสิ้น โดยอัตโนมัติ



ภาพที่ 4.18 แสดงไฟล์ที่นำออกเมื่อกดที่ปุ่มไอคอน

6.6. หากกดที่ไอคอนในตารางช่องรายการซ้ำ จะเป็นการนำไฟล์ออกมาในรูปแบบของ Excel ที่ถูกตั้ง Template ที่เป็นรูปแบบเดียวกันให้ง่ายต่อการอ่านไฟล์ ดังภาพที่ x.xx



วันที่ซื้อ	ชื่อ	ตำแหน่ง	ชื่อ-สกุล	ยี่ห้อรถ	สี	เลขทะเบียน	จังหวัด	หมายเลขเครื่อง	หมายเลขตัวถัง	ผู้ถือกรรมสิทธิ์	สถานะ	อื่น ๆ
9/19/2016	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1001	กทม.	T01F32D001	MNBXXXXXJBT63306	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Honda	Black	ขข 1002	กทม.	T01F32D002	MNBXXXXXJBT63407	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1003	กทม.	T01F32D003	MNBXXXXXJBT63508	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Yellow	ขข 1004	กทม.	T01F32D004	MNBXXXXXJBT63609	นายสมชาย ใจดี	N	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1005	กทม.	T01F32D005	MNBXXXXXJBT63710	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1006	กทม.	T01F32D006	MNBXXXXXJBT63811	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1007	กทม.	T01F32D007	MNBXXXXXJBT63912	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1008	กทม.	T01F32D008	MNBXXXXXJBT64013	นายสมชาย ใจดี	Y	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	BMW	Yellow	ขข 1009	กทม.	T01F32D009	MNBXXXXXJBT64114	นายสมชาย ใจดี	N	
	นายสมชาย ใจดี	นายช่างเทคนิค	นายสมชาย ใจดี	Toyota	Black	ขข 1010	กทม.	T01F32D010	MNBXXXXXJBT64215	นายสมชาย ใจดี	Y	

ภาพที่ 4.19 ตัวอย่างไฟล์ที่ถูกนำออกมาตรวจสอบ

6.7. ไฟล์ Excel ที่ได้รับมาจะถูกนำออกไปสำหรับการตรวจสอบว่ามีกลุ่มของข้อมูล (Record) ใดบ้างที่มีข้อมูลซ้ำอยู่ในฐานข้อมูลอยู่แล้ว หรือข้อมูลที่มีผลของการตรวจสอบไม่ตรงกับในฐานข้อมูล

4.2 ผลการวิเคราะห์ข้อมูล

1. การออกแบบหน้าจอ Interface มีผลต่อการวางโครงสร้าง (Structure) ของ Front-End ทั้งหมด
2. การศึกษา Business Rule ให้เข้าใจก่อนเริ่มการเขียนโค้ด มีผลโดยตรงต่อการสร้างฟังก์ชันขึ้นมาเพื่อการทำงานของระบบที่ต้องการและให้ผลข้อมูลตามต้องการ
3. การศึกษาฟังก์ชันที่เป็นมาตรฐานเดียวกันภายในบริษัทส่งผลให้การรับงานต่อจากงานที่เราทำไว้เป็นไปได้อย่างขึ้นและงานจะมีความรวดเร็วขึ้น
4. การสร้างฟังก์ชันต้องทำให้มีการทำงานให้น้อยที่สุดเพื่อลดการใช้ทรัพยากร ส่งผลต่อความเร็วในการประมวลผลของระบบ
5. การศึกษาความสัมพันธ์ภายในฐานข้อมูลทำให้ประมวลผลข้อมูลเป็นไปได้อย่างรวดเร็วขึ้น

4.3 วิจารณ์ข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายของการจัดทำ

โครงการ

โครงการชิ้นนี้มีวัตถุประสงค์เพื่อศึกษาและพัฒนา Web Application โดยใช้ความรู้ด้านภาษา TypeScript, Java, SQL, CSS เป็นหลัก เพื่อตอบสนองความต้องการของผู้ใช้งานหรือลูกค้า และมีประสิทธิภาพในการทำงานให้มากที่สุด และมีขั้นตอนของการทำงานที่ง่ายต่อการใช้งานของผู้ใช้งานให้มากที่สุด รวมไปถึงการแสดงผลที่เข้าใจมากที่สุด ทำให้สามารถทำงานที่ซับซ้อนได้ในเวลาที่รวดเร็วและง่ายด้วยการใช้การออกแบบหน้าจอผู้ใช้งาน (User Interface Design) มาพัฒนาให้ดียิ่งขึ้น

ทั้งนี้ในการใช้งานจริงนั้นมีความสามารถในการทำงานได้เป็นที่น่าพึงพอใจ ตรงกับความต้องการของลูกค้า (Requirement) แต่ยังพบข้อผิดพลาดในบางจุดที่ยังไม่สามารถแก้ไขได้ทันในเวลา เนื่องจากมีการปรับ แก้ไข ความต้องการของลูกค้าอยู่บ่อยครั้งเพื่อให้ทิศทางการทำงานตรงกับความต้องการการใช้งานของลูกค้าให้มากที่สุดและไม่ผิดทิศทางตามที่ควรจะเป็น

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ผลการดำเนินงานจริงสามารถปฏิบัติได้ตรงตามที่คาดหวังไว้แม้จะไม่ตรงตามแผนการทำงานบางส่วน เพราะการทำงานในบางส่วนของงานต้องใช้เวลาในการทำงานมากกว่าที่วางแผนไว้ เนื่องจากทักษะการทำงานของเราไม่มีความสามารถเท่าที่ควร จึงส่งผลให้มีความล่าช้าในการทำงานและมีการปรับเปลี่ยนไฟล์ต้นแบบและฐานข้อมูล ให้มีความเหมาะสมในกระบวนการทำงาน ฐานข้อมูลที่ถูกเปลี่ยนเพื่อลดความซับซ้อนของข้อมูล เพื่อให้ง่ายต่อการพัฒนา จึงมีการปรึกษากับทางพนักงานที่ปรึกษาดังโครงสร้างที่ถูกปรับเปลี่ยนและการพัฒนาระบบเพื่อให้ตรงกับ SRS

5.2 แนวทางแก้ปัญหา

1. ศึกษา SRS ให้มีความเข้าใจในกระบวนการทำงานของระบบให้ดี
2. Log ข้อมูลในการทำงานในทุกขั้นตอนเพื่อดูกระบวนการทำงาน ตัวแปรที่ถูกส่งและรับ ว่าตรงตามที่ต้องการหรือไม่
3. Log Debug ออกมาเพื่อดูข้อผิดพลาดของฟังก์ชัน เพื่อเป็นแนวทางในการแก้ไขปัญหาที่พบ
4. สำรองข้อมูลในการทำงานในทุกขั้นตอนเพื่อป้องกันการเกิดเหตุการณ์ไม่พึงประสงค์ต่อข้อมูลที่อาจจะส่งผลให้ข้อมูลงานที่สูญหายได้ อาทิเช่น ไวรัสมัลแวร์, เครื่องคอมพิวเตอร์เสียหาย, ไฟดับก่อนการเซฟงาน เป็นต้น
5. ปรึกษานักงานที่ปรึกษาดังสิ่งที่เราไม่แน่ใจ ไม่ควรทำเองเพราะจะส่งผลให้ระบบการทำงานผิดพลาดหรือได้ข้อมูลไม่ตรงตามความต้องการได้

5.3 ข้อเสนอแนะจากการดำเนินงาน

1. เนื่องจากในส่วนงานที่ทำมีความซับซ้อนมา จึงอยากให้มีการสอนงานให้มีความเข้าใจภาษาของงานที่เราเริ่มทำให้ดีก่อนการทำงาน จะส่งผลให้งานเสร็จได้รวดเร็ว และถูกต้องมากขึ้น
2. นักศึกษาควรกล้าแสดงออกที่จะถามถึงสิ่งที่ไม่เข้าใจ ในการทำงานและขอความช่วยเหลือในการหาความรู้ในการทำงานมากยิ่งขึ้น เพื่อเพิ่มทักษะในการทำงานให้มีประสิทธิภาพในการทำงาน

เอกสารอ้างอิง

1. Coding Basic [Online], Available: <http://www.codingbasic.com/html.html> [2016, August 31].
2. Mindphp [Online], Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2185-java-คืออะไร.html> [2016, August 31].
3. 2013, Srisunthorn's Blog [Online], Available: <https://srisunthorn.wordpress.com/2013/03/13/1056/> [2016, August 31].
4. Mindphp [Online], Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2187-java-javascript-คืออะไร.html> [2016, August 31].
5. Nuttavut Thongjor, 2016, Babel Coder [Online], Available: <https://www.babelcoder.com/blog/posts/typescript-data-types#TypeScript-คืออะไร> [2016, August 31].
6. Nuttavut Thongjor, 2016, Babel Coder [Online], Available: <https://www.babelcoder.com/blog/posts/angular2-introduction-to-angular2-and-architecture-overview> [2016, August 31].
7. 2016, Wikipedia [Online], Available: <https://th.wikipedia.org/wiki/%E0%B9%80%E0%B8%88%E0%B8%8B%E0%B8%B1%E0%B8%99> [2016, August 31].
8. Nopphanan Mayoe, 2016, Programming Hunter [Online], Available: <http://programminghunter.blogspot.com/2015/07/what-is-hibernate-framework.html> [2016, August 31].
9. Mk, 2015, Blognone [Online], Available: <https://www.blognone.com/node/68057> [2016, August 31].
10. IBM [Online], Available: <http://www-03.ibm.com/software/products/en/application> [2016, August 31].
11. Chai Phonbopit, 2015, Devahoy [Online], Available: <http://devahoy.com/posts/getting-started-with-nodejs/> [2016, August 31].
12. Mindphp [Online], Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2088-sql-คืออะไร.html> [2016, August 31].

13. เกร็ดความรู้ [Online], Available: <http://www.xn--12cg1cxchd0a2gzc1c5d5a.net/css/>
[2016, September 27].



ประวัติผู้จัดทำโครงการ



ชื่อ-สกุล

นายกนดรร เจนจิรวัดนา

วัน เดือน ปีเกิด

14 มิถุนายน พ.ศ.2536

ประวัติการศึกษา

ระดับประถมศึกษา

โรงเรียนอัสสัมชัญ ศรีราชา

สำเร็จการศึกษาปี 2550

ระดับมัธยมศึกษา

โรงเรียนบดินทรเดชา (สิงห์ สิงหเสนี)

สำเร็จการศึกษาปี 2555

ระดับอุดมศึกษา

สถาบันเทคโนโลยี ไทย-ญี่ปุ่น

ทุนการศึกษา

- ไม่มี -

ประวัติการฝึกอบรบ

1. Technology to AEC บริษัท AVALANT CO.,LTD
2. Space Technology บริษัท Japan Aerospace Exploration Agency
3. Logistics and Manufacturing (TAN LAND) บริษัท Ichitan Group Public Company Limited
4. Cloud Computing for Business ThaiNichi Institute of Technology