



การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob

ด้วย REST API

นายพัชรพล ห่วงไทย

รายงานสหกิจเล่มนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรบริหารธุรกิจบัณฑิต

สาขาวิชาระบบสารสนเทศ คณะบริหารธุรกิจและเทคโนโลยีสารสนเทศ

มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ

ปีการศึกษา 2565

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ

ชื่อ : นายพัชรพล ห่วงไทย
ชื่อรายงาน : การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
ที่ปรึกษา : อาจารย์ ไพฑูรย์ จันทรเรือง
ปีการศึกษา : 2565

บทคัดย่อ

โครงการการพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API พัฒนาขึ้นโดยมีวัตถุประสงค์
เพื่อ

1. เพื่อวิเคราะห์และออกแบบแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
2. เพื่อพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
3. เพื่อหาประสิทธิภาพของแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API

เครื่องมือที่ใช้ในการพัฒนาประกอบด้วย NodeJS เป็น Runtime ของภาษา Javascript ใช้ในการพัฒนา REST API ส่วนของฐานข้อมูลใช้เป็น Firebase โดยใช้ Firebase Firestore ที่เป็นฐานข้อมูลแบบ NoSql และ Firebase Storage เป็นตัวเก็บไฟล์รูปภาพ

ผลจากการพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST APIพบว่าสามารถทำงานเป็นไปตามที่ออกแบบไว้โดยมีผลการประเมินประสิทธิภาพอยู่ในเกณฑ์

อาจารย์ที่ปรึกษา

กิตติกรรมประกาศ

ผู้จัดทำรายงานการปฏิบัติงานสหกิจศึกษา ขอขอบพระคุณสถานประกอบการ พนักงานที่ปรึกษา พนักงานพี่เลี้ยง ที่คอยช่วยเหลือ สนับสนุน ทำให้รายงานปฏิบัติงานสหกิจศึกษาสำเร็จลุล่วงไปได้ทุกประการ

ขอขอบพระคุณคณะอาจารย์คณะบริหารธุรกิจและเทคโนโลยีสารสนเทศ สาขาวิชาการระบบสารสนเทศ ที่ได้สละเวลาตรวจสอบ แสดงความคิดเห็นและให้คำแนะนำ ทำให้รายงานปฏิบัติงานสหกิจศึกษามีเนื้อหาที่ถูกต้องและเหมาะสม

ขอขอบพระคุณเจ้าหน้าที่คณะบริหารธุรกิจและเทคโนโลยีสารสนเทศ สาขาวิชาการระบบสารสนเทศ ที่ได้ให้ความช่วยเหลือประสานงานในด้านต่างๆ เป็นอย่างดี

สุดท้ายนี้ ผู้จัดทำโครงการใคร่ขอกราบขอบพระคุณ คุณพ่อและคุณแม่ ผู้ให้ชีวิต และเป็นกำลังใจให้ผู้จัดทำรายงานตลอดมา อีกทั้งผู้มีส่วนเกี่ยวข้อง ที่ไม่ได้กล่าวนามไว้ในที่นี้ ซึ่งได้ให้การสนับสนุน จนทำให้รายงานการปฏิบัติงานสหกิจศึกษาเล่มนี้ สำเร็จลุล่วงไปได้ด้วยดีมา ณ โอกาสนี้

พัชรพล ห่วงไทย

สารบัญ

	หน้า
บทคัดย่อ	ข
กิตติกรรมประกาศ	ค
สารบัญตาราง	จ
สารบัญภาพ	ฉ
บทที่ 1 บทนำ	1
1.1 ประวัติและรายละเอียดของหน่วยงาน/สถานประกอบการ	1
1.2 รายละเอียดการปฏิบัติงานสหกิจศึกษา	2
1.3 โครงการพิเศษที่ได้รับมอบหมาย	2
บทที่ 2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง	4
2.1 เทคโนโลยีที่เกี่ยวข้อง	4
2.2 แนวคิดทฤษฎีที่เกี่ยวข้อง	7
2.3 เอกสารและงานวิจัยที่เกี่ยวข้อง	9
บทที่ 3 ขั้นตอนการดำเนินงาน	12
3.1 การวิเคราะห์และออกแบบระบบ	12
3.2 การวิเคราะห์ระบบ	13
บทที่ 4 ผลการดำเนินงาน	29
4.1 ผลการดำเนินงาน	29
4.2 ประโยชน์ที่ได้รับจากการปฏิบัติงานสหกิจศึกษา	33
บทที่ 5 สรุป อภิปรายรายผล และข้อเสนอแนะ	34
5.1 สรุปผลดำเนินงาน	34
5.2 ข้อเสนอแนะ	35
บรรณานุกรม	0
ภาคผนวก	0
แบบฟอร์มประเมิน	0
ใบประเมิน	0

สารบัญตาราง

ตารางที่	หน้า
1-1 แสดงระยะเวลาดำเนินงาน	3
3-1 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 1.0 การสมัครสมาชิก	22
3-2 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 2.0 การเข้าสู่ระบบ	23
3-3 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 3.0 การว่าจ้างงาน	23
3-4 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 4.0 การรับจ้างงาน	24
3-5 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 5.0 การจัดการข้อมูลพื้นฐาน	24
3-6 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 6.0 การจัดทำรายงาน	25
3-7 แสดงพจนานุกรมข้อมูลผู้ว่าจ้าง (employer)	27
3-8 แสดงพจนานุกรมข้อมูลผู้รับจ้าง (contractor)	27
3-9 แสดงพจนานุกรมข้อมูลการลงทะเบียนว่าจ้างงาน (Register employer)	28
3-10 แสดงพจนานุกรมข้อมูลการลงทะเบียนว่าจ้างงาน (Register contractor)	28

สารบัญภาพ

ภาพ	หน้า
2-1 NodeJS Icon	4
2-2 Javascript Icon	5
2-3 แสดงจำนวนและประเภทของ Web API ที่มีการใช้งาน	10
2-4 แสดงระยะเวลาที่ใช้ในการประมวลผล (ms) เปรียบเทียบกับจำนวนข้อมูล	11
3-1 แสดงขั้นตอนการดำเนินงาน	12
3-2 แสดงผังการไหลข้อมูลระดับที่ 0 การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API	14
3-3 แสดงผังการไหลข้อมูลระดับที่ 1 การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API	15
3-4 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 1 การสมัครสมาชิก	16
3-5 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 2 การเข้าสู่ระบบ	17
3-6 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 3 การว่าจ้างงาน	18
3-7 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 4 การรับจ้างงาน	19
3-8 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 5 การจัดการข้อมูลพื้นฐาน	20
3-9 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 6 การจัดทำรายงาน	21
4-1 การเรียกใช้ API แสดงข้อมูลประเภทงานทั้งหมด	30
4-2 ผลลัพธ์การเรียกใช้ API แสดงข้อมูลประเภทงานทั้งหมด	30
4-3 หน้าแสดงผลการเรียกใช้ API แสดงข้อมูลประเภทงานทั้งหมด	31
4-4 การเรียกใช้ API สร้างข้อมูลประเภทงาน	31
4-5 การเรียกใช้ API แก้ไขข้อมูลประเภทงาน	32
4-6 การเรียกใช้ API ลบข้อมูลประเภทงาน	32

บทที่ 1

บทนำ

1.1 ประวัติและรายละเอียดของหน่วยงาน/สถานประกอบการ

1.1.1 ชื่อสถานประกอบการ

ห้างหุ้นส่วนจำกัด เค.เอ็น. กรุ๊ป 888

1.1.2 สถานที่ตั้ง สถานประกอบการ

999/84 ซอย 6 ตำบลตลาด อำเภอเมือง จังหวัดนครราชสีมา 30310

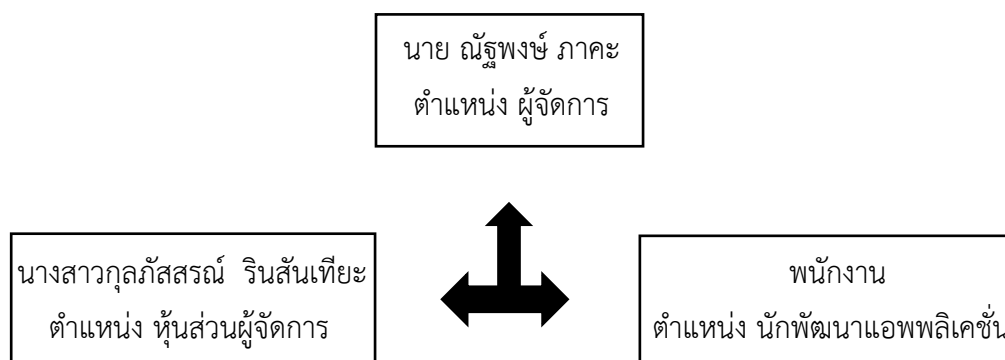
1.1.3 ประวัติความเป็นมาของหน่วยงาน / สถานประกอบการ

ผู้ก่อตั้งเริ่มจากที่อยากได้ประสบการณ์ใหม่ๆ ต้องการเป็นนายตัวเอง แล้วก็ทางผู้ก่อตั้งเริ่มรับงาน Outsource จนกระทั่งมาถึงการเปิดบริษัท เพื่อให้หน้าเชื่อถือมากขึ้น ก่อตั้งบริษัทเมื่อ 4 มกราคม 2565

1.1.4 ลักษณะงานของสถานประกอบการ

กิจกรรมการจัดทำโปรแกรมเว็บเพจและเครือข่ายตามวัตถุประสงค์ของผู้ใช้

1.1.5 การจัดองค์กรและบริหารงาน



1.2 รายละเอียดการปฏิบัติงานสหกิจศึกษา

1.2.1 ตำแหน่ง และหน้าที่ที่ได้รับมอบหมาย

Backend Developer โดยหน้าที่ที่ได้รับมอบหมายคือ พัฒนาโปรแกรมในส่วนของหลังบ้านรวมไปถึงการออกแบบ database ส่วนใหญ่จะทำงานร่วมกับนักพัฒนา Front End เพื่อให้ Code ทำงานได้อย่างสมบูรณ์แบบ

1.2.2 ประเภทของงานที่ได้รับมอบหมายให้รับผิดชอบ

พัฒนา API เพื่อเชื่อมต่อกับแอปพลิเคชันและเว็บไซต์รับสมัครงาน MeJob ให้สามารถเพิ่ม ลบ แก้ไข และเรียกใช้ข้อมูลในฐานข้อมูลได้

1.2.3 ลักษณะของงานที่รับผิดชอบ

จัดการข้อมูลที่บ้านต้องการใช้และส่งไปให้ รวมไปถึงการออกแบบ database ว่าต้องเก็บข้อมูลยังไงจะดึงข้อมูลยังไงให้เร็วที่สุด โดยจะส่งผ่าน API(Application Programming Interface)

1.2.4 ชื่อและตำแหน่งของพนักงานที่ปรึกษา

นาย อนุรักษ์ ภาคะ ตำแหน่ง ผู้จัดการ

1.2.5 ระยะเวลาการปฏิบัติงานที่สหกิจศึกษา

ระยะเวลาในการปฏิบัติงาน ตั้งแต่ 20 มิถุนายน 2565 ถึง 7 ตุลาคม 2565

1.3 โครงการพิเศษที่ได้รับมอบหมาย

1.3.1 ชื่อโครงการสหกิจศึกษา

การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API

1.3.2 วัตถุประสงค์ของโครงการ

1.3.2.1 เพื่อวิเคราะห์แอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API

1.3.2.2 เพื่อพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API

1.3.2.3 เพื่อหาประสิทธิภาพของแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API

1.3.3 ประโยชน์ที่ได้รับ

1.3.3.1 ได้ระบบหลังบ้านให้ดึงข้อมูลให้เร็วที่สุด

1.3.3.2 ได้ระบบหลังบ้านที่ทำงานร่วมกับหน้าบ้านได้ // ไม่กล่าวถึง API***

1.3.4 ประโยชน์ที่คาดว่าจะได้รับ (ต้องสอดคล้องกับความเป็นและความสำคัญของปัญหา)

1.3.4.1 ได้เรียนรู้และสามารถพัฒนา API ที่สามารถใช้ในการศึกษาค้นคว้าได้จริง

1.3.5 ขอบเขตของงาน

การออกแบบและพัฒนาระบบหลังบ้าน โดยโปรแกรม visual studio code ในการพัฒนา API ด้วย NodeJs โดยใช้ภาษา Javascript คอยรับคำสั่งจากฝั่งลูกข่าย (Client) ซึ่งก็คือ Application ต่าง ๆ เช่น Web Application, Mobile Application เมื่อฝั่ง ลูกข่าย (Client) ส่งคำสั่ง จะเรียกว่าการ Request จากนั้น ตัว API จะรับคำสั่งดังกล่าว นำไปประมวลผลและสรุปเป็นข้อมูลที่ตรงกับ Request และส่งข้อมูลเหล่านั้นกลับไปฝั่งลูกข่าย (Client) หรือ Application เพื่อนำไปใช้งานต่อไป

1.3.6 ระยะเวลาการปฏิบัติงาน

บทที่ 2

ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

ในการจัดทำรายงาน การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API และการประยุกต์ใช้งาน API ได้รวบรวม ทฤษฎีและเทคโนโลยีที่เกี่ยวข้องรวมถึงรายงานหรือระบบงานที่เกี่ยวข้องโดยมีรายละเอียดเนื้อหา ทั้งหมด ดังนี้

2.1 เทคโนโลยีที่เกี่ยวข้อง

2.2 แนวคิดทฤษฎีที่เกี่ยวข้อง

2.3 เอกสารและงานวิจัยที่เกี่ยวข้อง

2.1 เทคโนโลยีที่เกี่ยวข้อง

2.1.1 Node.js คือสภาพแวดล้อมการทำงานของภาษา JavaScript นอกเว็บเบราว์เซอร์ที่ทำงานด้วย V8 engine นั้นหมายความว่าเราสามารถใช้ Node.js ในการพัฒนาแอปพลิเคชันแบบ Command line แอปพลิเคชัน Desktop หรือแม้แต่เว็บเซิร์ฟเวอร์ได้ โดยที่ Node.js จะมี APIs ที่เราสามารถใช้สำหรับทำงานกับระบบปฏิบัติการ เช่น การรับค่าและการแสดงผล การอ่านเขียนไฟล์ และการทำงานกับเน็ตเวิร์ก เป็นต้น



ภาพที่ 2-1 NodeJS Icon

Node.js ถูกพัฒนาและทำงานด้วยใช้ Chrome V8 engine สำหรับคอมไพล์ภาษา JavaScript ให้เป็นภาษาเครื่องด้วยการคอมไพล์แบบ Just-in-time (JIT) เพื่อเพิ่มประสิทธิภาพการทำงานของภาษา JavaScript จากที่แต่เดิมมันเป็นภาษาที่มีการทำงานแบบ Interpreted

Node.js เป็นโปรแกรมที่สามารถใช้ได้ทั้งบน Windows, Linux และ Mac OS X นั้นหมายความว่าคุณสามารถเขียนโปรแกรมในภาษา JavaScript และนำไปรันได้ทุกระบบปฏิบัติการที่สนับสนุนโดย Node.js นี่เป็นแนวคิดของการเขียนครั้งเดียวแต่ทำงานได้ทุกที่ (Write once, run anywhere)

ประวัติของ Node.js ถูกพัฒนาขึ้นมาครั้งแรกโดยนักพัฒนาชาวอเมริกัน Ryan Dahl ในปี 2009 สำหรับใช้เป็นสภาพแวดล้อมเพื่อที่จะใช้ภาษา JavaScript ในการพัฒนาเว็บไซต์ในรูปแบบของ Server side ในช่วงแรกนั้น Node.js สนับสนุนการทำงานเพียงแค่นบน Linux และ Mac OS X เท่านั้น และสำหรับ Windows นั้นถูกเพิ่มการสนับสนุนเข้ามาในภายหลัง

แรงบันดาลใจในการสร้าง Node.js ขึ้นมาก็เพื่อใช้เป็นเว็บเซิร์ฟเวอร์ที่สามารถรองรับจำนวนการใช้งานได้มากกว่าเว็บเซิร์ฟเวอร์ในรูปแบบเดิมอย่าง Apache HTTP Server โดยใช้หลักของการเขียนโปรแกรมแบบ Asynchronous หรือ Non-blocking I/O ที่ใช้ฟังก์ชัน Callback และ Event loop ในการจัดการกับ Request แทนการสร้าง Thread และ Process ที่ต้องใช้ทรัพยากรของเว็บเซิร์ฟเวอร์เป็นจำนวนมาก

Node.js และภาษา JavaScript ในช่วงเริ่มแรกภาษา JavaScript ถูกพัฒนาขึ้นมาสำหรับเขียน Script บนเว็บเบราว์เซอร์เพื่อให้หน้าเว็บสามารถตอบสนองได้แบบ Real-time สิ่งหนึ่งที่ทำให้ Node.js เลือกภาษา JavaScript เป็นภาษาหลักในการเขียนโปรแกรมเนื่องจากเห็นประสิทธิภาพการทำงานของ V8 engine ซึ่งเป็น Engine ที่ใช้รันภาษา JavaScript บน Google Chrome พร้อมทั้งภาษา JavaScript ยังมีรูปแบบการทำแบบ Asynchronous I/O ซึ่งเป็นสิ่งที่ Node.js ต้องการอยู่แล้ว



ภาพที่ 2-2 Javascript Icon

นั่นหมายความว่าในปัจจุบัน ภาษา JavaScript ไม่ได้ถูกจำกัดสำหรับการเขียนโปรแกรมเพียงบนเว็บเบราว์เซอร์อีกต่อไป ในการมาของ Node.js เราสามารถเขียนโปรแกรม Command line สร้างเว็บเซิร์ฟเวอร์หรือเขียนโปรแกรมบน Desktop โดยการใช้ Framework อย่าง Electron หรือโปรแกรมทุกรูปแบบเหมือนกับภาษาอื่นๆ ได้

ข้อดีอีกอย่างหนึ่งในการใช้ภาษา JavaScript ของ Node.js คือทำให้การพัฒนาเว็บไซต์ทำได้อย่างขึ้นสำหรับนักพัฒนา เนื่องจากพวกเขาสามารถใช้ภาษา JavaScript สำหรับทั้ง Front-end และ Back-end ได้โดย

ไม่ต้องศึกษาภาษาเฉพาะในแต่ละด้าน ตัวอย่างของการพัฒนาเว็บไซต์ในรูปแบบนี้ เช่น React.js ซึ่งเป็นไลบรารี โดย Facebook

นอกจากนี้ คุณยังสามารถใช้ภาษาอื่นๆ สำหรับเขียนโปรแกรมบน Node.js ได้ เช่น ภาษา TypeScript เพื่อให้การเขียนโปรแกรมมีการเชื่อมโยงกับประเภทข้อมูลมากขึ้น และสามารถช่วยลดข้อผิดพลาดในการพัฒนาโปรแกรมได้ แต่เนื่องจาก Node.js สนับสนุนเพียงภาษา JavaScript ดังนั้นโปรแกรมที่เขียนโดยภาษาอื่นจะถูกแปลงเป็น JavaScript ก่อน

2.1.2 JSON ย่อมาจาก JavaScript Object Notation คือ Standard format อย่างหนึ่งที่เป็น text และสามารถอ่านออกได้ด้วยตาเปล่า ใช้ในการสร้าง object ขึ้นมาเพื่อส่ง ข้อมูลระหว่าง application หรือ Applications Program Interface(API) โดย format จะมีรูปแบบเป็น คู่ Key-Value หรือเป็นแบบ Array และสามารถนำมาใช้แทน XML format JSON เป็น format ที่ได้รับการใช้งานจาก JavaScript มาก่อน แต่ปัจจุบันมี ภาษา programming หลายชนิดที่เริ่มใช้งาน

2.1.3 Postman คือเครื่องมือสำหรับการพัฒนาและทดสอบ API service ซึ่งเป็นที่นิยมอย่างกว้างขวาง เพราะการใช้งานที่แสนจะง่ายดาย มี UI สวยงามเข้าใจง่าย ๆ และสามารถใช้ได้ฟรีฟังก์ชันทั่วไปที่คนใช้งานจะเป็นการทดสอบส่ง request เพื่อดู response ที่ได้กลับมาเป็นข้อมูล Action ต่าง ๆ แต่บทความนี้จะพูดถึงฟังก์ชันอื่น ๆ ที่คนทั่วไปอาจยังไม่เคยใช้หรือไม่เคยรู้มาก่อน เราจึงจะมาแนะนำ ๆ ที่มีโปรแกรมสำหรับการพัฒนาอยากทดสอบตัว API สามารถใช้งานโปรแกรม Postman ได้อย่างสะดวกอย่างมากไม่ยุ่งยาก

2.1.4 Visual Studio Code หรือ VS Code จากบริษัทไมโครซอฟต์ เป็นโปรแกรมประเภท Editor ใช้ในการแก้ไขโค้ดที่มีขนาดเล็ก แต่มีประสิทธิภาพสูง เป็น OpenSource โปรแกรมจึงสามารถนำมาใช้งานได้โดยไม่ มีค่า ใช้งานง่าย เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานหลายแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows , macOS และ Linux รองรับหลายภาษาทั้ง JavaScript, TypeScript และ Node.js ในตัว และสามารถเชื่อมต่อกับ Git ได้ง่าย สามารถนำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือและส่วนขยายต่าง ๆ ให้เลือกใช้มากมาย รองรับการใช้ภาษาอื่น ๆ ทั้ง ภาษา C++ , C# , Java , Python , PHP หรือ Go สามารถปรับเปลี่ยน Themes ได้ มีส่วน Debugger และ Commands เป็นต้น

2.1.5 npm หรือ Node Package Manager คือโปรแกรมสำหรับการแพ็คเกจของภาษา JavaScript และเป็นตัวจัดการแพ็คเกจที่มาพร้อมกับตัวติดตั้งของ Node.js มันเป็นเครื่องมือที่ช่วยอำนวยความสะดวกและแนะนำให้ใช้โดย Node.js สำหรับใช้จัดการกับแพ็คเกจและไลบรารีของภาษา JavaScript ที่คุณต้องการใช้งานสำหรับการเขียนโปรแกรมบน Node.js

2.1.6 Firebase คือฐานข้อมูลประเภท NoSQL ฐานข้อมูล MySQL MSSQL และฐานข้อมูลชนิด RDBMS ต่าง ๆ จะมีลักษณะเป็นตารางข้อมูล มีคอลัมน์ มีการกำหนดชนิดของข้อมูลไว้อย่างชัดเจน และใช้ภาษา SQL ในการติดต่อเพื่อขอใช้ข้อมูล (SELECT) เพิ่มข้อมูล (INSERT) และลบข้อมูล (DELETE) สามารถกรองเอาเฉพาะข้อมูลที่ต้องการได้ด้วยการใช้ WHERE และบางครั้งมีปัญหาเรื่องช่องโหว่ (SQL Injection ถือเป็นวิธีพื้นฐานที่นิยมใช้และได้ผลมากที่สุดในขณะนี้)

ฐานข้อมูลชนิด NoSQL จะไม่ใช้ภาษา SQL ในการจัดการข้อมูล และออกแบบให้มีความยืดหยุ่น และเน้นความเร็วในการใช้งานมากที่สุด ฐานข้อมูล NoSQL ที่นิยมใช้งานในปัจจุบันคือ MongoDB ซึ่งมีการเก็บข้อมูลเป็นชนิด JSON (เจสัน) มีตารางเหมือนเดิม แต่ไม่มีคอลัมน์ข้อมูลที่ตายตัว ใน 1 แถว สามารถเก็บข้อมูลได้ทั้งข้อความ (String) ตัวเลข (Number) และอื่น ๆ รวมไปถึงอาร์เรย์ และออบเจกต์

Firebase มีการทำงานคล้าย ๆ กับ MongoDB คือมีฐานข้อมูล แต่ไม่มีตาราง มีการเก็บข้อมูลในรูปแบบ JSON สามารถเพิ่มข้อมูลไปในออบเจกต์ใด ๆ ก็ได้ แต่เก็บเป็นอาร์เรย์ไม่ได้ ถ้าต้องการเพิ่มข้อมูลแบบอาร์เรย์ จะต้องใช้การ PUT ข้อมูลเข้าไปต่อท้ายเรื่อย ๆ ซึ่งจะมี Key ที่ Firebase สร้างให้เป็นตัวอ้างอิง

2.2 แนวคิดทฤษฎีที่เกี่ยวข้อง

2.2.1 ระบบหลังบ้าน(Backend) คือ ระบบจัดการเว็บไซต์ เปรียบได้กับหลังบ้าน ถือเป็นส่วนการจัดการเนื้อหา รวมถึงโครงสร้างของเว็บไซต์ สำหรับให้เว็บไซต์เข้ามาบริหารจัดการเว็บไซต์ ตรงข้ามกับส่วนระบบหน้าบ้าน (Frontend) ซึ่งเป็นส่วนที่แสดงผล เปรียบได้กับหน้าบ้าน ซึ่งถือเป็นส่วนที่ ผู้ใช้งาน(User) โดยทั่วไปสามารถเห็นและเข้ามาใช้งานได้ ซึ่ง Backend และ Frontend นี้ ถือเป็นส่วนหนึ่งของ ระบบ CMS (Content Management System) หรือ ระบบจัดการเนื้อหาของเว็บไซต์ นั่นเอง และในส่วน backend ควรต้องมีทักษะด้าน Web Developer (Javascript, CSS, HTML5, jQuery)

Front-end เปรียบได้กับเปรียบได้กับหน้าบ้าน หรือเป็นส่วนติดต่อผู้ใช้ (User interface) ไม่ว่าจะเป็น หน้าโฮม หน้าเว็บเพจ เนื้อหาต่างๆ รูปภาพ ลิงก์ เป็นต้น เป็นส่วนที่ user ทั่วไปสามารถเห็นและเข้ามาใช้งานได้ของเว็บไซต์ โดยความสำคัญของ front-end ส่วนนี้จะแสดงหน้าตาของเว็บไซต์ให้ผู้เข้าชมเห็น การออกแบบก็เป็นส่วนที่ช่วยดึงดูดและทำให้ผู้อื่นสนใจเว็บไซต์ ทั้งความสวยงาม การใช้งานเว็บไซต์ที่เข้าใจง่าย สะดวก และทำให้ผู้เข้าชมเห็นว่าเว็บไซต์มีการพัฒนาอยู่ตลอด

Front-end และ Back-end เกี่ยวข้องกันอย่างไร front-end กับ back-end จะมีความสำคัญต่อกันเป็นอย่างมาก หาก front-end แสดงผลช้าหรือเกิดข้อผิดพลาด ส่วนของ back-end ก็ต้องทำการแก้ไขปัญหา หรือถ้าหากว่า back-end ไม่ทำการปรับปรุงเว็บไซต์อย่างสม่ำเสมอ จะส่งผลต่อเว็บไซต์เป็นอย่างมาก ไม่ว่าจะเป็นเว็บไซต์ล้าหลัง ขาดความน่าสนใจ ผู้เยี่ยมชมลดลง ก่อให้เกิดผลกระทบหลาย ๆ ด้าน

โดยทั่วไปผู้ดูแลเว็บไซต์ ที่สามารถแก้ไขส่วนต่างๆภายในเว็บไซต์ได้นั้น มักจะมีหน้า log in เพื่อเข้าสู่ระบบจัดการเว็บไซต์ หน้าเว็บไซต์ในส่วนของการจัดการ ของเว็บที่ผู้ดูแลเท่านั้น จึงสามารถเข้าไปแก้ไขหรือเพิ่มเติมเนื้อหาต่างๆในเว็บไซต์ได้

Front-end และ Back-end อะไรสำคัญกว่ากัน ถ้าการใช้งานเว็บไซต์เน้นการแสดงผลข้อมูลที่ไม่เยอะ front-end จะต้อง design ให้ออกมาสวยและถูกใจคนเข้าชมเว็บไซต์ แต่ถ้าหากเว็บไซต์ข้อมูลเยอะ ส่วน back-end จะต้องจัดการข้อมูลและจัดการเว็บไซต์อย่างไรเพื่อไม่ให้ระบบนั้นช้าลงในการแสดงผล

2.2.2 Non-relational database คือ Non-relational database ถูกคิดค้นขึ้นในช่วงกลางทศวรรษ 1990 เนื่องจากในยุคนั้นมีการใช้งานอินเทอร์เน็ตที่หลากหลายขึ้นทำให้ การใช้งาน Relational database เริ่มไม่ตอบโจทย์ ทำให้มีการเริ่มคิดค้นตัว Non-relational หรือที่เรียกว่า NoSQL ขึ้นมา เพื่อให้สามารถแปลงข้อมูลที่มีความหลากหลายมากขึ้น รวมไปถึงการหลีกเลี่ยงความยุ่งยากของการใช้งาน Relational ซึ่งก็ยังมีการเพิ่มความยืดหยุ่นที่มากขึ้นด้วย

ตัวย่อ NoSQL ถูกใช้งานครั้งแรกในปี 1998 โดย Carlo Strozzi โดย NoSQL แพร่หลายมากขึ้นในปี 2015 เนื่องจากในปีเดียวกันมีการทำ Big Data มากขึ้น องค์กรต่างๆชั้นนำในขณะนั้น เช่น Facebook, Twitter, LinkedIn, และ Google เริ่มมีการปรับใช้ NoSQL เพราะว่า ตอบโจทย์ในเรื่องความเร็ว ความยืดหยุ่นและเหมาะกับองค์กรขนาดใหญ่ที่ต้องการประมวลผลข้อมูลพวกรูปภาพ วีดีโอ เสียงจำนวนมาก

Non-relational database คือ ฐานข้อมูลที่ไม่ใช่ Relational Database เป็น Unstructure ของ Database แบบ SQL ไว้ใช้สำหรับข้อมูลที่ไม่ได้อยู่เป็นรูปแบบของตาราง หรือมีความเชื่อมโยงสัมพันธ์กันโดยตรง เช่น ข้อมูลที่เป็นภาพ เป็นวิดีโอ เป็นเสียง หรืออาจจะเป็นข้อมูลที่เป็นตัวอักษรก็ได้

2.2.3 API ย่อมาจาก Application Programming Interface คือตัวที่ใช้ช่องทางการเชื่อมต่อหรือเป็นช่องทางการเชื่อมต่อที่ทาง Server API ที่ให้เชื่อมต่อได้ใช้เป็นตัวกลางในการเชื่อมต่อกับโปรแกรมต่อโปรแกรมหรือเชื่อมต่อกับระบบปฏิบัติการเข้าด้วยกัน เป็นตัวช่วยในการให้เข้าถึงการรับหรือการส่ง ข้อมูลได้ง่ายขึ้น และในบทความนี้จะมาเปรียบเทียบถึงประโยชน์ของ API ว่าทำอะไรได้บ้าง และใช้ประโยชน์อะไรบ้าง เพื่อให้เข้าใจได้ Web API คือ API ที่อยู่ในกลุ่มของ HTTP ที่ใช้เว็บไซต์มาเป็นตัวกลางในการเชื่อมต่อของเว็บไซต์

2.3 เอกสารและงานวิจัยที่เกี่ยวข้อง

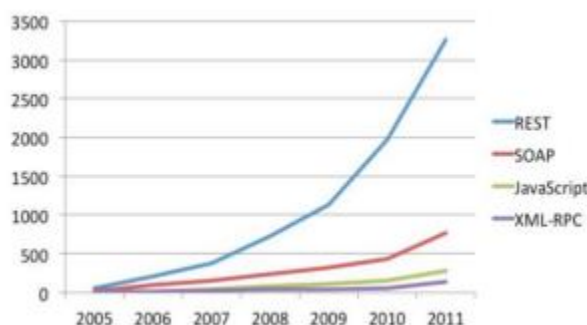
2.3.1 อานันท์เงินบำรุง : การเปรียบเทียบประสิทธิภาพระหว่างคลังข้อมูลเชิงสัมพันธ์และคลังข้อมูลเชิงเอกสาร ปริญญาวิทยาศาสตรมหาบัณฑิต (เทคโนโลยีสารสนเทศ) สาขาวิชาเทคโนโลยีสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์
 อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก: รองศาสตราจารย์กฤษณะ ไวยมัย, D.U.
 ปี การศึกษา 2563

ปัจจุบันประเภทของฐานข้อมูลมีมากมายหลายรูปแบบที่นิยมใช้กันมากขึ้น ซึ่งหนึ่งในฐานข้อมูลประเภทใหม่คือฐานข้อมูลแบบโนเอสคิวแอล ซึ่งงานวิจัยนี้ได้นำฐานข้อมูลแบบเอกสาร (Document database) ซึ่งเป็นฐานข้อมูลประเภทหนึ่งใน โนเอสคิวแอล โดยจะเปรียบเทียบกับ ฐานข้อมูลเชิงสัมพันธ์ (Relational database) ในการนำมาใช้เป็นคลังข้อมูล ซึ่งจะมีการวิเคราะห์การออกแบบ, ความเร็วในการสืบค้นข้อมูล (Querying) การนำเข้าข้อมูล (Time of importing data) และการใช้งานทรัพยากรเครื่องเช่น หน่วยประมวลผลกลาง หน่วยความจำ หลักการปริมาณการใช้พื้นที่เก็บข้อมูลในเครื่อง เพื่อนำข้อมูลในมุมมองดังกล่าวมาเปรียบเทียบ ประสิทธิภาพของคลังข้อมูลเชิงสัมพันธ์และคลังข้อมูลโนเอสคิวแอล โดยในงานวิจัยชิ้นนี้ได้ใช้การออกแบบของโครงสร้าง (Schema) ร่วมกัน ทั้งสองคลังข้อมูล ซึ่งผลของการเปรียบเทียบนั้นพบว่า การนำเข้าข้อมูลนั้นจะมีเครื่องมือช่วยในการนำเข้าข้อมูลในคลังข้อมูลทั้งสองประเภทเหมือนกัน ๆ กันทำให้ง่ายในการนำเข้าข้อมูล และในส่วนของคุณภาพเร็วใน Query นั้น พบว่าเมื่อ

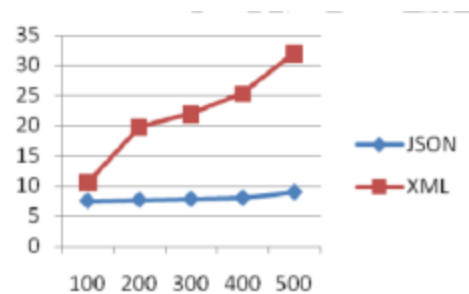
ลักษณะของ Query มีการเชื่อมโยงมากกว่าหนึ่งตาราง (multi-table queries) นั้น คลังข้อมูลเชิงสัมพันธ์ใช้เวลาในการ Query น้อยกว่าแต่เมื่อลักษณะของ Query เป็นแบบภายในตารางเดียว (single-table queries) และมีการค้นหาในลักษณะของการระบุข้อมูลบางส่วนที่เฉพาะเจาะจงรวมไปถึงเมื่อผลลัพธ์จากการค้นหามีจำนวนแถวที่มากนี้คลังโนเอสคิวแอลใช้เวลาในการสืบค้นน้อยกว่า จึงสามารถแนะนำได้ว่าถ้าหากเราใช้การออกแบบในลักษณะโครงสร้างรูปดาวแล้วการนำโนเอสคิวแอลมาใช้เก็บข้อมูลสำหรับเป็นคลังข้อมูลนั้นอาจจะมีความเหมาะสมมากกว่า คลังข้อมูลเชิงสัมพันธ์สำหรับกรณีที่รายงานที่ต้องการนั้นต้องการข้อมูลจากหนึ่งตารางเท่านั้น

2.3.2 การวิจัยติดตามการใช้งาน API พบว่ามีการเพิ่มจำนวนขึ้นเป็นอย่างมากจากในปี ค.ศ. 2005 พบจำนวน 105 APIs ในปี ค.ศ. 2011 พบจำนวนมากกว่า 5,000 APIs โดยพบว่าการใช้ 23 รูปแบบ REST มากกว่า SOAP ทั้งนี้อาจเนื่องมาจากสาเหตุที่ REST มียืดหยุ่นและมีความง่ายในการใช้งานมากกว่า SOAP ที่มีความเป็นมาตรฐานและใช้งานได้ยุ่งยากมากกว่า และจากการเปรียบเทียบประสิทธิภาพความเร็วในการทำงานของ SOAP เทียบกับ REST พบว่า REST มีการทำงานที่รวดเร็วกว่า SOAP และหากเปรียบเทียบในเรื่องข้อมูลที่ใช้ในการรับส่งข้อมูลพบว่า SOAP มีการรับส่งข้อมูลที่มากกว่า REST

เนื่องจาก SOAP มีข้อกำหนดในการรับส่งข้อมูลโดยใช้รูปแบบ XML ต่างจาก REST ที่มีได้หลายรูปแบบ ซึ่ง JSON เป็นรูปแบบหนึ่งที่เป็นที่นิยม มีการวิจัยที่เปรียบเทียบระยะเวลาของการรับส่งข้อมูลและการนำเอาข้อมูลมาใช้ โดยเป็นการเปรียบเทียบข้อมูลรูปแบบ XML กับ JSON พบว่า JSON ใช้ระยะเวลาน้อยกว่า XML



ภาพที่ 2-3 แสดงจำนวนและประเภทของ Web API ที่มีการใช้งาน

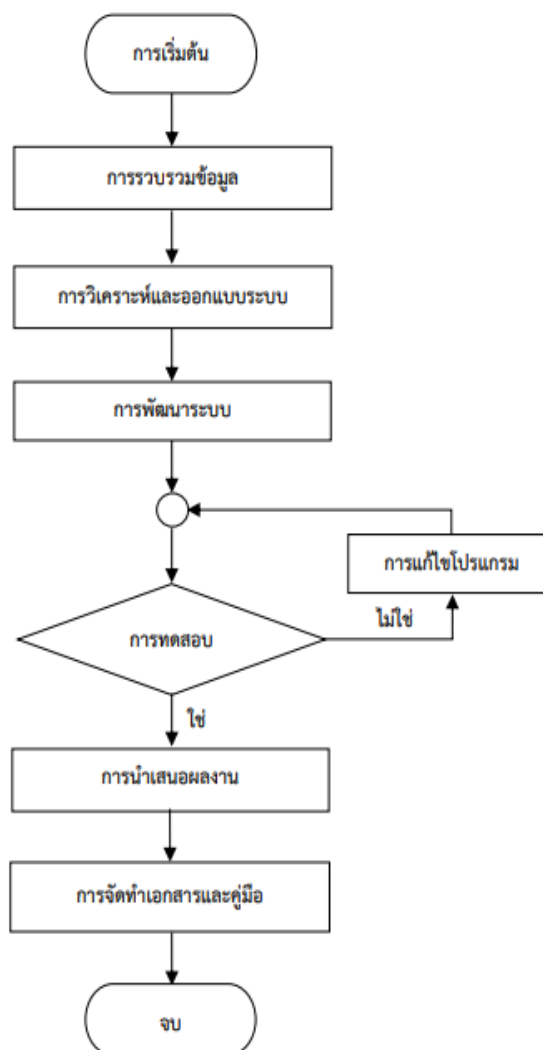


ภาพที่ 2-4 แสดงระยะเวลาที่ใช้ในการประมวลผล (ms) เปรียบเทียบกับจำนวนข้อมูล

บทที่ 3

การวิเคราะห์และออกแบบระบบ

ในการพัฒนาแอปพลิเคชันรับสมัครงาน MeJob และการประยุกต์ใช้งาน API ได้มีการจัดทำแผนการดำเนินงาน รวมถึงขั้นตอนต่างๆ ที่ใช้ในการพัฒนาระบบงาน ซึ่งมีขั้นตอนการดำเนินงาน ดังนี้



ภาพที่ 3-1 แสดงขั้นตอนการดำเนินงาน

3.1 การวิเคราะห์ระบบ

3.1.1 ผู้ที่เกี่ยวข้องกับระบบ

3.1.1.1 ผู้ใช้ทั่วไป

3.1.1.2 ผู้ว่าจ้าง

3.1.1.3 ผู้รับจ้าง

3.1.1.4 ผู้ดูแลระบบ

3.1.2 ผู้ใช้ทั่วไป

3.1.2.1 การสมัครเป็นสมาชิก

3.1.2.2 การสมัครเป็นช่าง

3.1.3 ผู้ว่าจ้าง

3.1.3.1 การเข้าสู่ระบบ

3.1.3.2 การว่าจ้างงาน

3.1.4 ผู้รับจ้าง

3.1.4.1 การเข้าสู่ระบบ

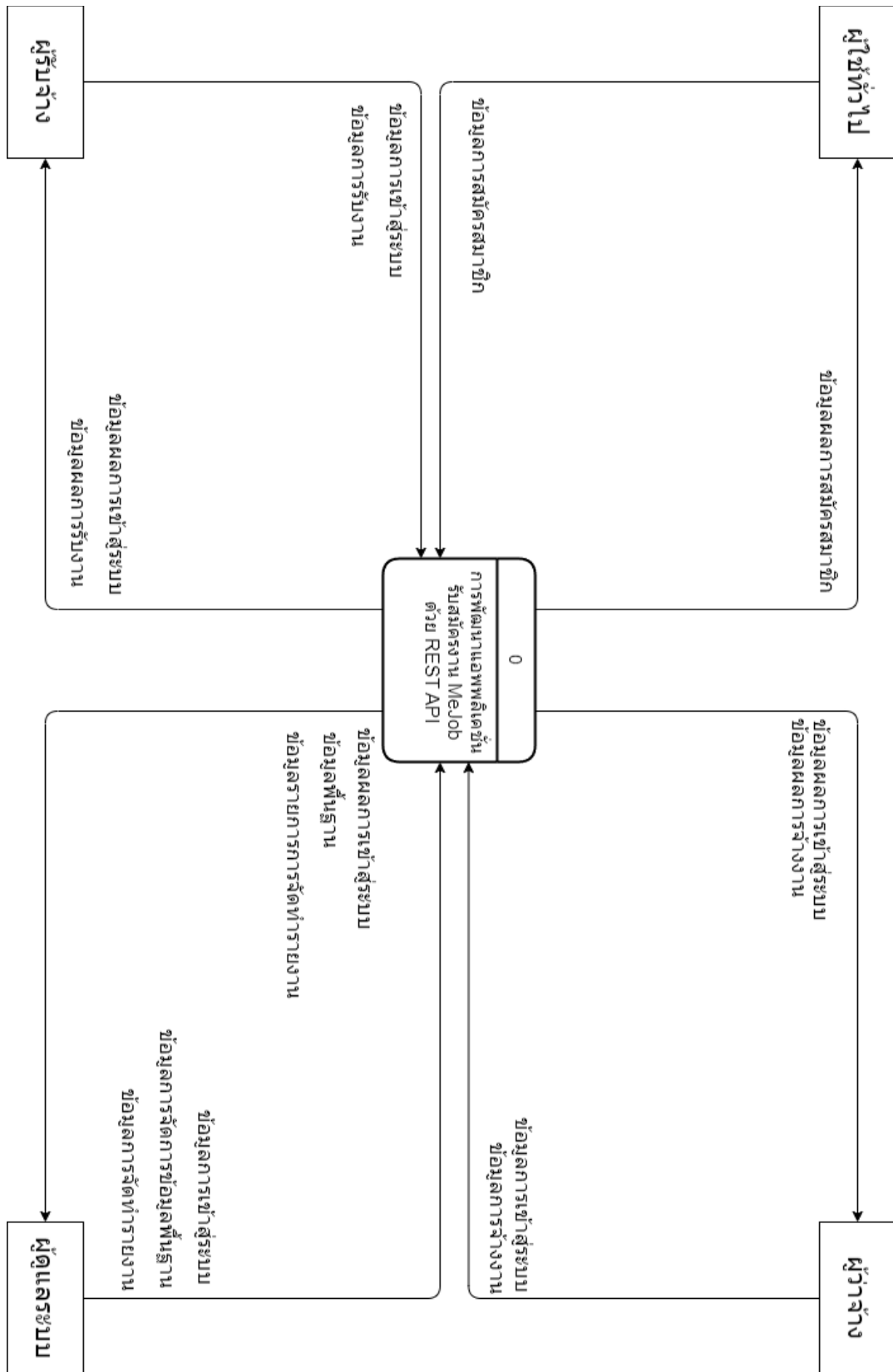
3.1.4.2 การรับจ้างงาน

3.1.5 ผู้ดูแลระบบ

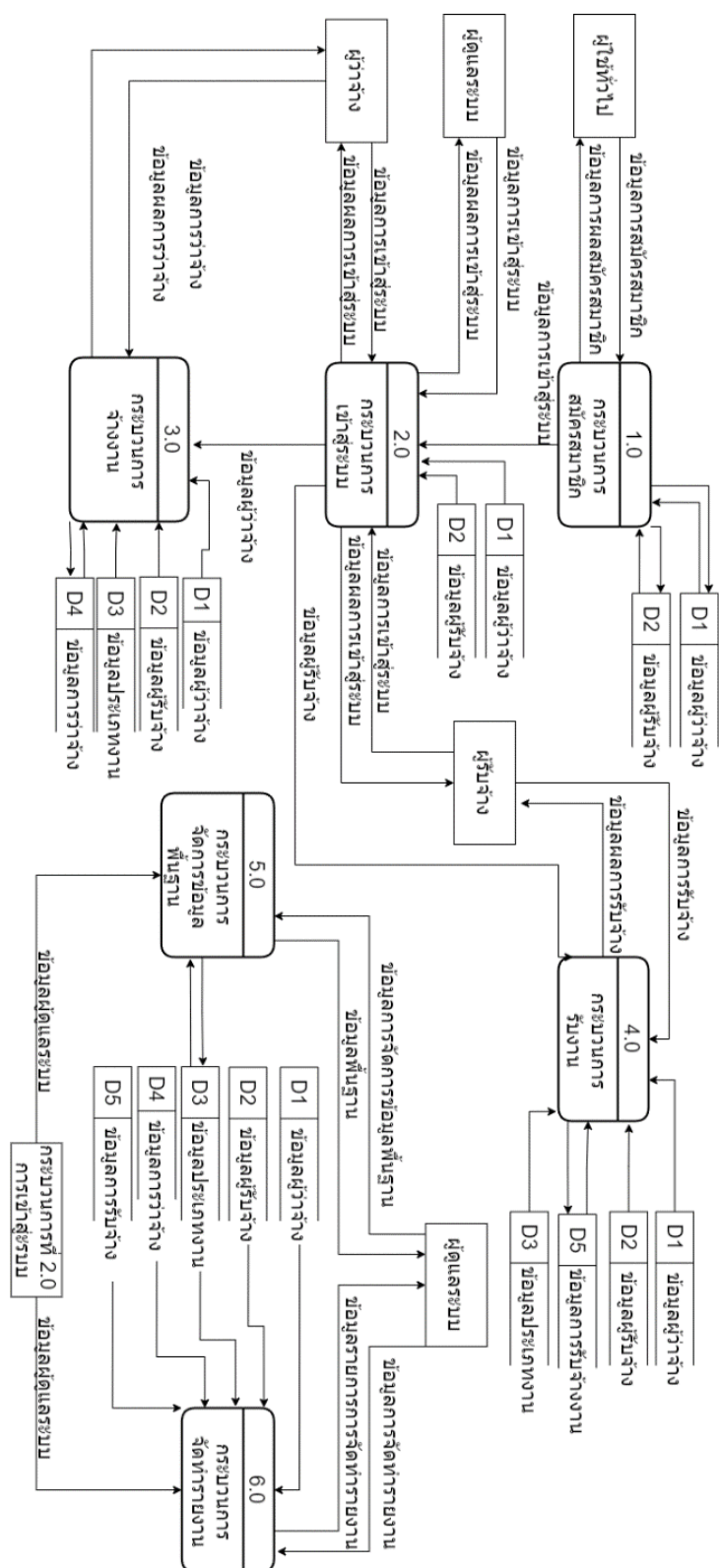
3.1.4.1 การเข้าสู่ระบบ

3.1.4.4 การจัดการข้อมูลพื้นฐาน

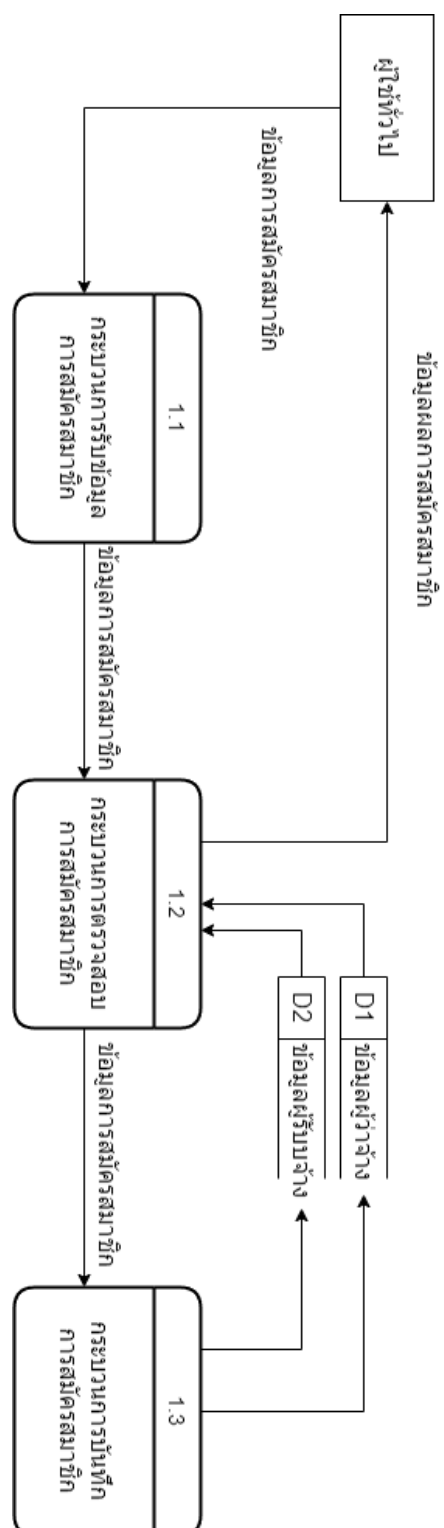
3.1.4.5 การจัดทำรายงาน



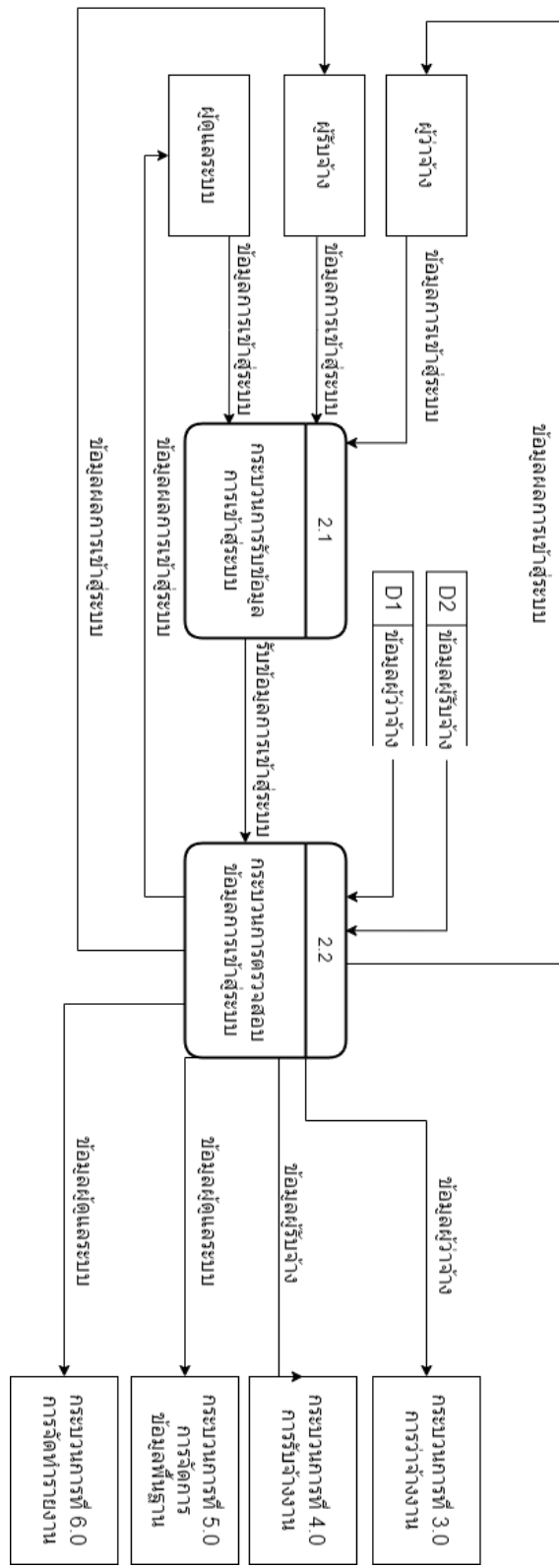
ภาพที่ 3-2 แสดงผังการไหลข้อมูลระดับที่ 0 การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API



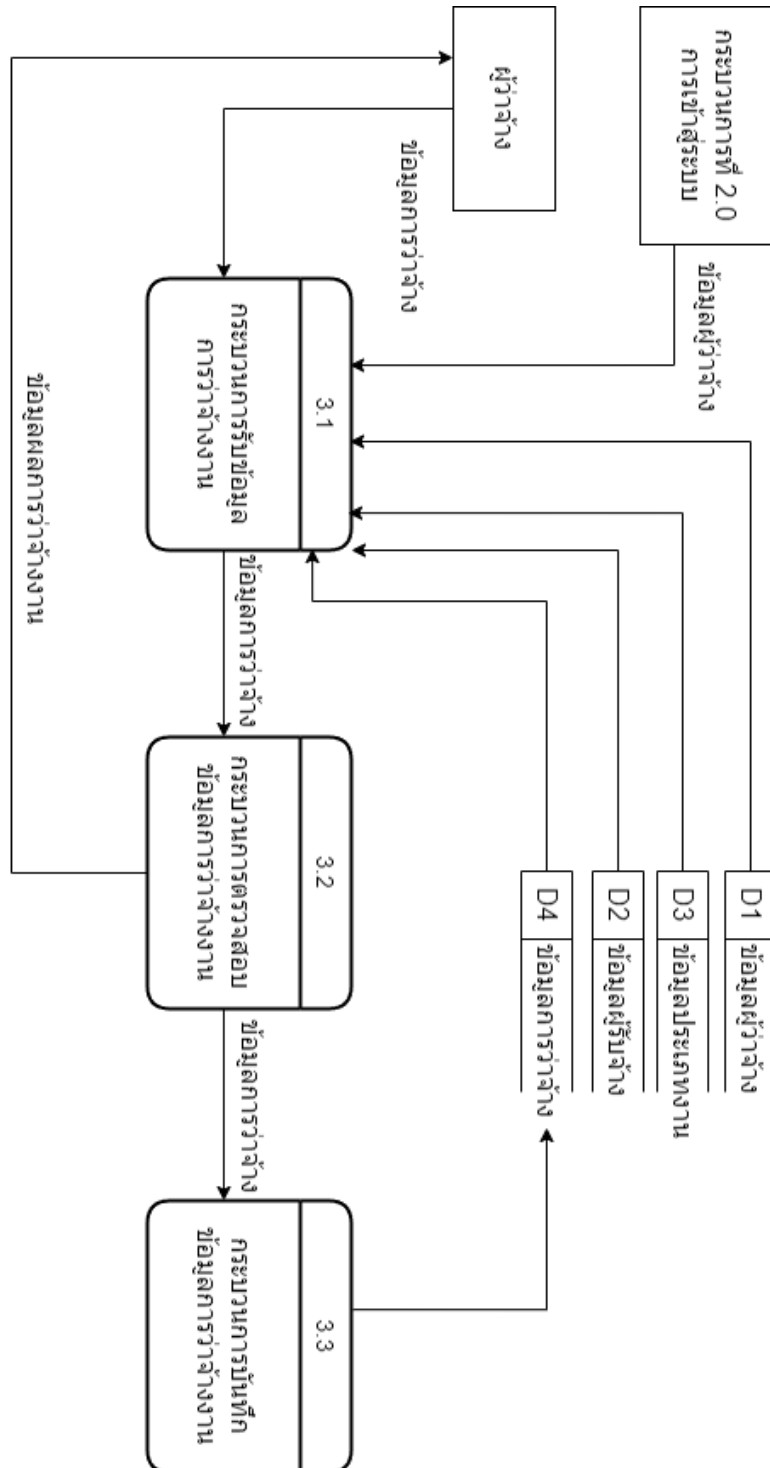
ภาพที่ 3-3 แสดงผังการไหลข้อมูลระดับที่ 1 การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API



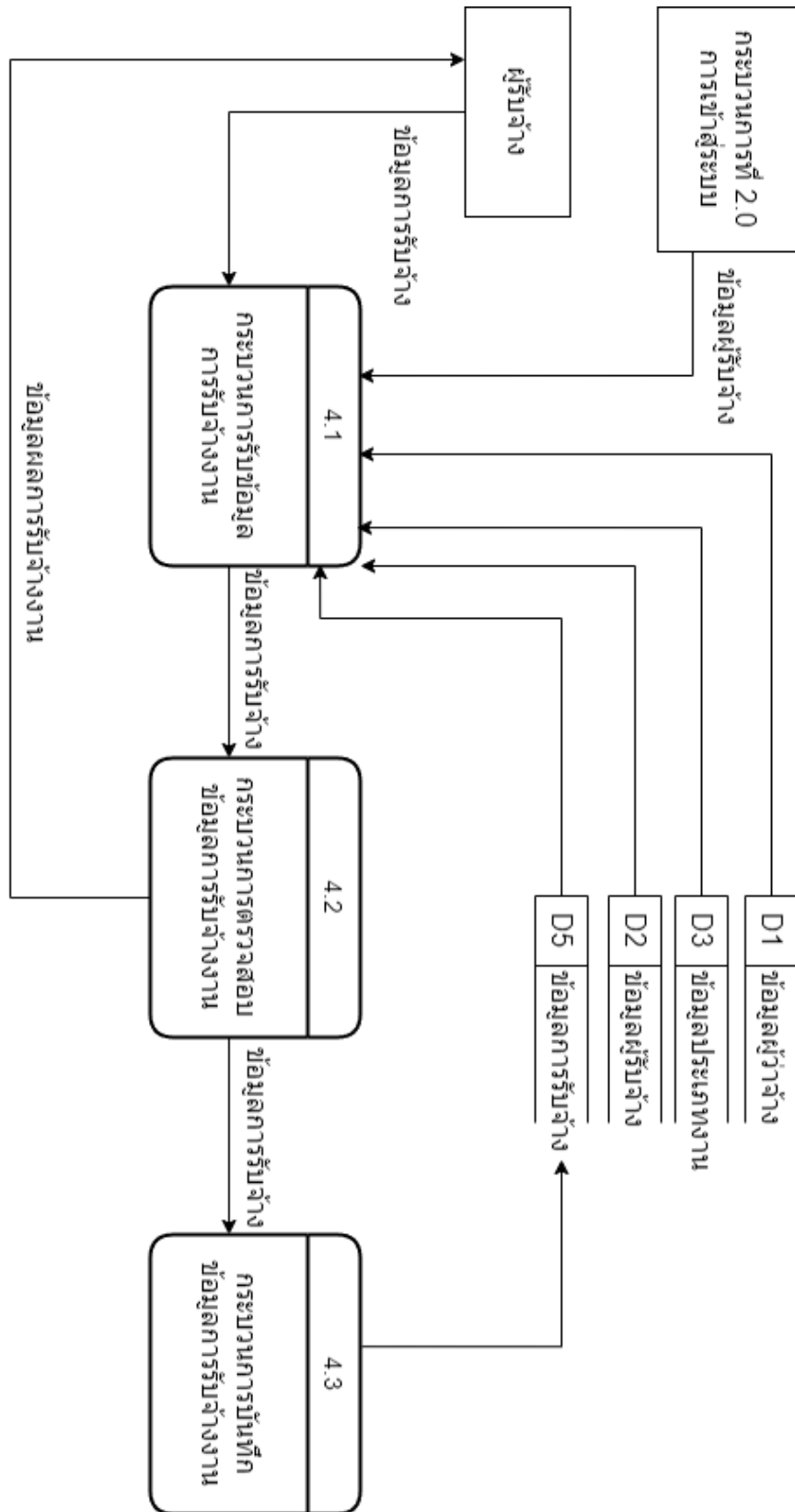
ภาพที่ 3-4 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 1 การสมัครสมาชิก



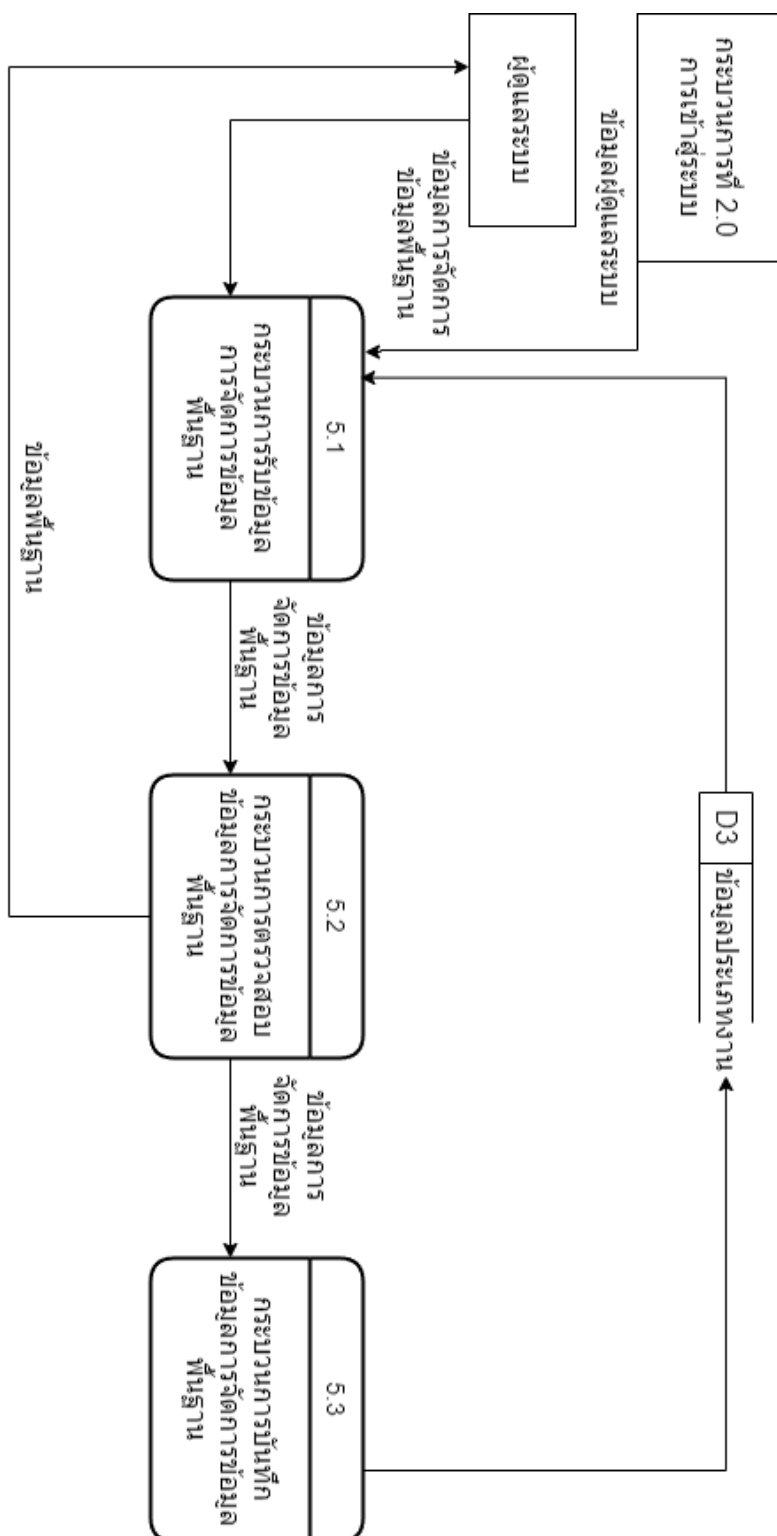
ภาพที่ 3-5 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 2 การเข้าสู่ระบบ



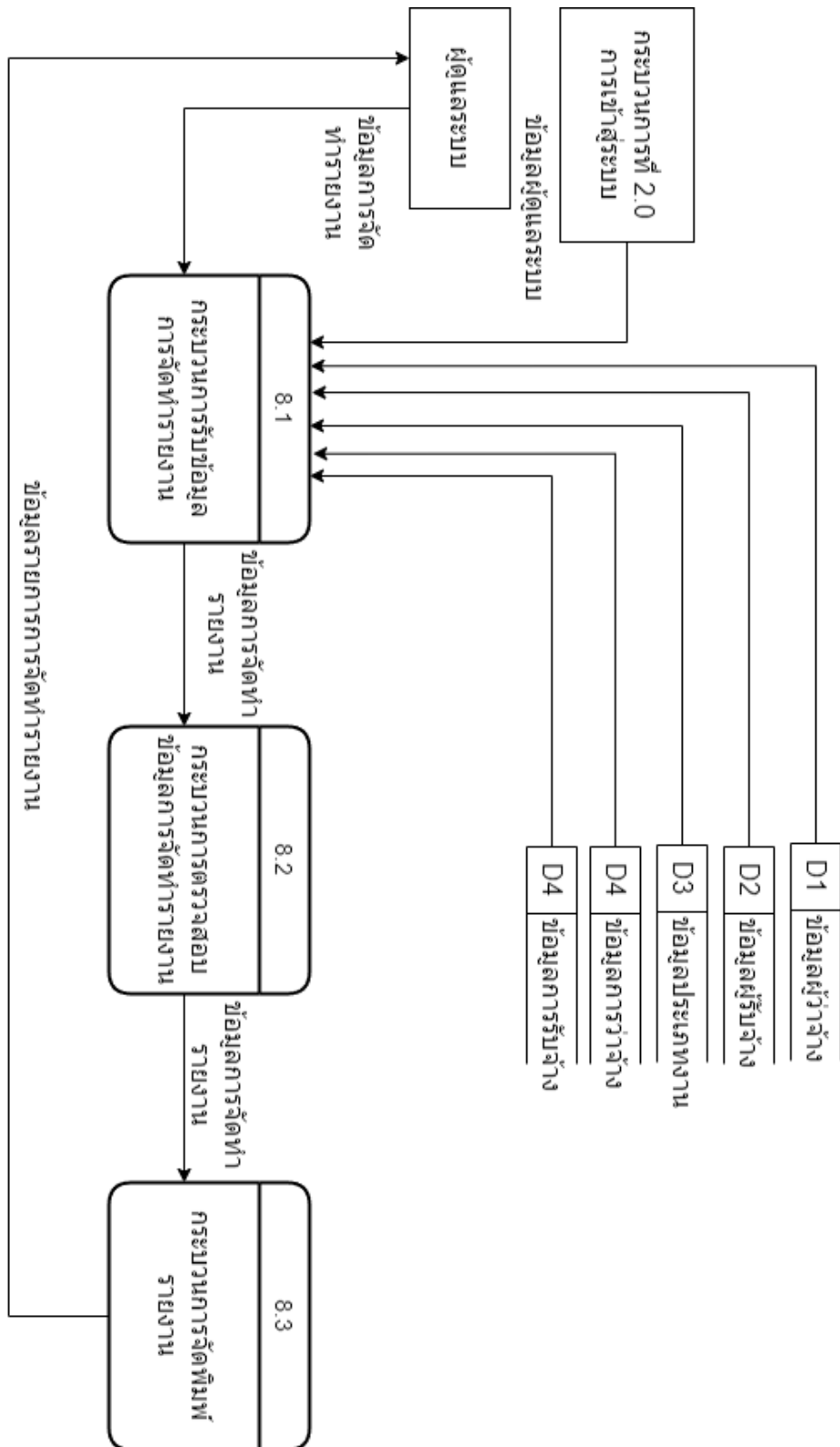
ภาพที่ 3-6 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 3 การว่าจ้างงาน



ภาพที่ 3-7 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 4 การรับจ้างงาน



ภาพที่ 3-8 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 5 การจัดการข้อมูลพื้นฐาน



ภาพที่ 3-9 แสดงผังการไหลข้อมูลระดับที่ 2 กระบวนการที่ 6 การจัดทำรายงาน

การเขียนอธิบายการประมวลผล (Process Description) เพื่อเป็นการอธิบายกระบวนการที่มาจาก DFD ระดับ 2 ทุกๆ กระบวนการ ซึ่งเป็นการเขียนในรูปแบบการเขียนบรรยายเป็นตัวอักษรเพื่อให้มีความเข้าใจในการทำงานของแต่ละกระบวนการมากยิ่งขึ้น แสดงดังตารางที่

ตารางที่ 3-1 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 1.0 การสมัครสมาชิก

Process Description	
System	การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
DFD Number	1.0
Process Name	กระบวนการสมัครสมาชิก
External Entity	ผู้ใช้งาน
Input Data Flow	ข้อมูลการสมัครสมาชิก
Data Store Used	D1 ข้อมูลผู้ว่าจ้าง D2 ข้อมูลผู้รับจ้าง D3 ข้อมูลประเภทงาน
Description	<p>1.กระบวนการที่ 1.0 (กระบวนการสมัครสมาชิก) รับข้อมูลการสมัครสมาชิก</p> <p>2.กระบวนการที่ 1.0 (กระบวนการสมัครสมาชิก) จะทำการตรวจสอบข้อมูลการสมัครสมาชิกเมื่อตรวจสอบแล้วไม่ถูกต้อง จะทำการแจ้งไปยังผู้ใช้งาน เพื่อให้ทำการแก้ไขได้ถูกต้อง เมื่อตรวจสอบแล้วว่าข้อมูลถูกต้อง กระบวนการที่ 1.0 (กระบวนการสมัครสมาชิก) จะทำการบันทึกข้อมูลลงใน ข้อมูลผู้ว่าจ้าง D1</p> <p>3.กระบวนการที่ 1.0 (กระบวนการสมัครสมาชิก) ส่งข้อมูลผลไปยังผู้ใช้งานเมื่อสมัครสมาชิกสำเร็จ</p>

ตารางที่ 3-2 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 2.0 การเข้าสู่ระบบ

Process Description	
System	การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
DFD Number	2.0
Process Name	กระบวนการเข้าสู่ระบบ
External Entity	ผู้ว่าจ้าง,ผู้รับจ้าง,ผู้ดูแลระบบ
Input Data Flow	ข้อมูลการเข้าสู่ระบบ
Data Store Used	D1 ข้อมูลผู้ว่าจ้าง D2 ข้อมูลผู้รับจ้าง D3 ข้อมูลประเภทงาน
Description	<p>กรณีผู้ว่าจ้าง</p> <ol style="list-style-type: none"> 1.กระบวนการที่ 2.0 (กระบวนการเข้าสู่ระบบ) รับข้อมูลการเข้าสู่ระบบ 2.กระบวนการที่ 2.0 (กระบวนการเข้าสู่ระบบ) จะทำการตรวจสอบข้อมูลการเข้าสู่ระบบ เมื่อตรวจสอบแล้วไม่ถูกต้อง จะทำการแจ้งไปยังผู้ใช้งาน เพื่อให้ทำการแก้ไขได้ถูกต้อง เมื่อตรวจสอบแล้วถูกต้องจะทำการเข้าสู่ระบบ 3.กระบวนการที่ 2.0 (กระบวนการเข้าสู่ระบบ) ส่งข้อมูลผลไปยังผู้ใช้งานเมื่อเข้าสู่ระบบสำเร็จ

ตารางที่ 3-3 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 3.0 การว่าจ้างงาน

Process Description	
System	การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
DFD Number	3.0
Process Name	กระบวนการว่าจ้างงาน
External Entity	ผู้ว่าจ้าง
Input Data Flow	ข้อมูลการลงทะเบียนการว่าจ้างงาน
Data Store Used	D1 ข้อมูลผู้ว่าจ้าง D2 ข้อมูลผู้รับจ้าง D3 ข้อมูลประเภทงาน D4 ข้อมูลการว่าจ้าง
Description	<ol style="list-style-type: none"> 1.กระบวนการที่ 3.0 (กระบวนการว่าจ้างงาน) รับข้อมูลการว่าจ้างงาน 2.กระบวนการที่ 3.0 (กระบวนการว่าจ้างงาน) จะทำการตรวจสอบข้อมูลการว่าจ้างงาน เมื่อตรวจสอบแล้วไม่ถูกต้อง จะทำการแจ้งไปยังผู้ใช้งาน เพื่อให้ทำการแก้ไขได้ถูกต้อง เมื่อตรวจสอบแล้วถูกต้องจะทำการบันทึกข้อมูลลง ข้อมูลการว่าจ้าง D4 3.กระบวนการที่ 3.0 (กระบวนการว่าจ้างงาน) ส่งข้อมูลผลไปยังผู้ใช้งานเมื่อลงทะเบียนการว่าจ้างงานสำเร็จ

ตารางที่ 3-4 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 4.0 การรับจ้างงาน

Process Description	
System	การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
DFD Number	4.0
Process Name	กระบวนการรับจ้างงาน
External Entity	ผู้รับจ้าง
Input Data Flow	ข้อมูลการรับจ้างงาน
Data Store Used	D1 ข้อมูลผู้ว่าจ้าง D2 ข้อมูลผู้รับจ้าง D3 ข้อมูลประเภทงาน D5 ข้อมูลการรับจ้าง
Description	<p>1.กระบวนการที่ 4.0 (กระบวนการรับจ้างงาน) รับข้อมูลการว่าจ้างงาน</p> <p>2.กระบวนการที่ 4.0 (กระบวนการรับจ้างงาน) จะทำการตรวจสอบข้อมูลการรับจ้างงาน เมื่อตรวจสอบแล้วไม่ถูกต้อง จะทำการแจ้งไปยังผู้ใช้งาน เพื่อให้ทำการแก้ไขได้ถูกต้อง เมื่อตรวจสอบแล้วถูกต้องจะทำการบันทึกข้อมูลลง ข้อมูลการรับจ้าง D</p> <p>3.กระบวนการที่ 4.0 (กระบวนการรับจ้างงาน) ส่งข้อมูลผลไปยังผู้ใช้งานเมื่อลงทะเบียนการรับจ้างงานสำเร็จ</p>

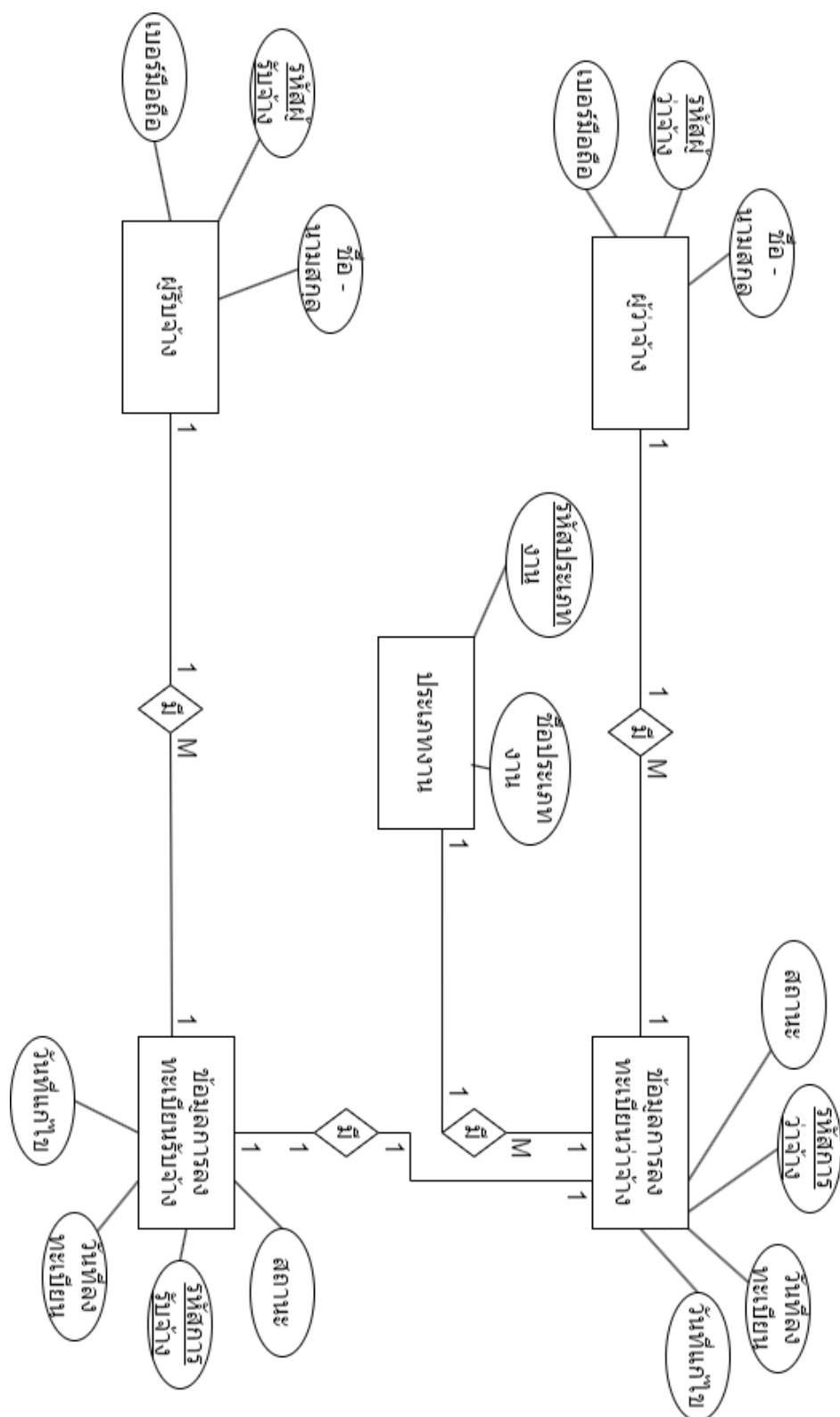
ตารางที่ 3-5 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 5.0 การจัดการข้อมูลพื้นฐาน

Process Description	
System	การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
DFD Number	5.0
Process Name	กระบวนการจัดการข้อมูลพื้นฐาน
External Entity	ผู้ดูแลระบบ
Input Data Flow	ข้อมูลการจัดการข้อมูลพื้นฐาน
Data Store Used	D3 ข้อมูลประเภทงาน
Description	<p>1.กระบวนการที่ 5.0 (กระบวนการจัดการข้อมูลพื้นฐาน) รับข้อมูลผู้ดูแลระบบ</p> <p>2.กระบวนการที่ 5.0 (กระบวนการจัดการข้อมูลพื้นฐาน) รับข้อมูลประเภทงาน(D3)</p> <p>3.กระบวนการที่ 5.0 (กระบวนการจัดการข้อมูลพื้นฐาน) จะทำการจัดการข้อมูลพื้นฐาน การแก้ไขเมื่อตรวจสอบแล้วว่าพบข้อมูลที่ไม่ถูกต้อง จะทำการแจ้งไปยังผู้ดูแลระบบ เพื่อให้ทำการแก้ไขให้ถูกต้อง จะทำการบันทึกลงในข้อมูลประเภทงาน(D3)</p>

	4.กระบวนการที่ 5.0 (กระบวนการจัดการข้อมูลพื้นฐาน) ส่งข้อมูลรายการจัดทำข้อมูลพื้นฐานไปยังผู้ดูแลระบบหลังจากบันทึกเรียบร้อยแล้ว
--	---

ตารางที่ 3-6 แสดงคำอธิบายการประมวลผลของกระบวนการที่ 6.0 การจัดทำรายงาน

Process Description	
System	การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API
DFD Number	6.0
Process Name	กระบวนการจัดทำรายงาน
External Entity	ผู้ดูแลระบบ
Input Data Flow	ข้อมูลการจัดทำรายงาน
Output Data Flow	ข้อมูลรายงาน
Data Store Used	D1 ข้อมูลผู้ว่าจ้าง D2 ข้อมูลผู้รับจ้าง D3 ข้อมูลประเภทงาน D4 ข้อมูลการว่าจ้าง D5 ข้อมูลการรับจ้าง
Description	<p>1.กระบวนการที่ 6.0 (กระบวนการจัดทำรายงาน) รับข้อมูลผู้ดูแลระบบ</p> <p>2.กระบวนการที่ 6.0 (กระบวนการจัดทำรายงาน) รับข้อมูลประเภทงาน(D3)</p> <p>3.กระบวนการที่ 6.0 (กระบวนการจัดทำรายงาน) จะทำการตรวจสอบข้อมูลการจัดทำรายงาน เมื่อตรวจสอบแล้วว่าพบข้อมูลที่ไม่ถูกต้อง จะทำการแจ้งไปยังผู้ดูแลระบบ เพื่อให้ทำการแก้ไขให้ถูกต้อง เมื่อตรวจสอบแล้วพบว่าข้อมูลถูกต้อง กระบวนการที่ 6.0 (กระบวนการจัดทำรายงาน) ส่งข้อมูลไปยังผู้ดูแลระบบหลังจากบันทึกเรียบร้อยแล้ว</p> <p>4.กระบวนการที่ 6.0 (กระบวนการจัดทำรายงาน) ส่งข้อมูลรายการไปยังผู้ดูแลระบบหลังจากบันทึกเรียบร้อยแล้ว</p>



ภาพที่ 3-10 แสดงแผนผังความสัมพันธ์ ER-Diagram การพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST

3.4.2.5 โครงสร้างข้อมูลหรือนิยามข้อมูล (Schema) เป็นการแจกแจงรายละเอียด เพื่อแสดงให้เห็นถึง ส่วนประกอบภายในข้อมูลแผนภาพความสัมพันธ์ระหว่างข้อมูล (ER-Diagram) โครงสร้างข้อมูลหรือนิยามข้อมูล (Schema)

- 1.ผู้ว่าจ้าง (รหัสผู้ว่าจ้าง,ชื่อ-นามสกุล,เบอร์มือถือ)
- 2.ผู้รับจ้าง (รหัสผู้รับจ้าง,ชื่อ-นามสกุล,เบอร์มือถือ)
- 3.ข้อมูลการลงทะเบียนว่าจ้างงาน (รหัสการว่าจ้าง,สถานะ,วันที่ลงทะเบียน,วันที่แก้ไข)
- 4.ข้อมูลการลงทะเบียนรับจ้างงาน (รหัสการรับจ้าง,สถานะ,วันที่ลงทะเบียน,วันที่แก้ไข)
- 5.ประเภทงาน (รหัสประเภทงาน,ชื่อประเภทงาน)

พจนานุกรมข้อมูล (Data Dictionary) เป็นการนำโครงสร้างข้อมูลหรือนิยามข้อมูล (Schema)

มาทำเป็นตารางฐานข้อมูล เพื่อแจกแจงรายละเอียดว่า แต่ละ Entity มีชนิดของข้อมูลและขนาดของข้อมูลเท่าไร และมีสถานะเป็นอะไร แสดงดังตารางที่ 3-10 ถึง 3-17

ตารางที่ 3-7 แสดงพจนานุกรมข้อมูลผู้ว่าจ้าง (employer)

ลำดับ	ชื่อฟิลด์	ความหมาย	ชนิดข้อมูล	ขนาด	คีย์
1	Id_employer	รหัสผู้ว่าจ้าง	Int	10	PK
2	name_employer	ชื่อ-นามสกุล	Varchar	50	-
3	phonenummer_employer	เบอร์มือถือ	Int	10	-

ตารางที่ 3-8 แสดงพจนานุกรมข้อมูลผู้รับจ้าง (contractor)

ลำดับ	ชื่อฟิลด์	ความหมาย	ชนิดข้อมูล	ขนาด	คีย์
1	Id_contractor	รหัสผู้รับจ้าง	Int	10	PK
2	name_contractor	ชื่อ-นามสกุล	Varchar	50	-

ตารางที่ 3-9 แสดงพจนานุกรมข้อมูลการลงทะเบียนว่าจ้างงาน (Register employer)

ลำดับ	ชื่อฟิลด์	ความหมาย	ชนิดข้อมูล	ขนาด	คีย์
1	Id_Regemp	รหัสการลงทะเบียน	Int	10	PK
2	status_Regemp	สถานะ	Varchar	10	-
3	creatAt_Regemp	วันที่ลงทะเบียน	Int	20	-
4.	updateAt_Regemp	วันที่แก้ไข	Int	20	

ตารางที่ 3-10 แสดงพจนานุกรมข้อมูลการลงทะเบียนว่าจ้างงาน (Register contractor)

ลำดับ	ชื่อฟิลด์	ความหมาย	ชนิดข้อมูล	ขนาด	คีย์
1	Id_Regcon	รหัสการลงทะเบียน	Int	10	PK
2	status_Regcon	สถานะ	Varchar	10	-
3	creatAt_Regcon	วันที่ลงทะเบียน	Int	20	-
4.	updateAt_Regcon	วันที่แก้ไข	Int	20	

บทที่ 4

ผลการดำเนินการ

4.1 ผลการดำเนินงาน

จากที่ได้ปฏิบัติงานสหกิจศึกษา ที่บริษัท Kn Group (ฝ่ายโปรแกรมเมอร์ Backend Developer) เป็นระยะเวลาการปฏิบัติงานสหกิจศึกษา วันที่ 20 มิถุนายน – 7 ตุลาคม 2565 โดยพนักงานที่ปรึกษา ได้มอบหมายให้ข้าพเจ้าปฏิบัติหน้าที่ โปรแกรมเมอร์งานที่ได้รับมอบหมายเป็นงานพัฒนา API สำหรับแอปพลิเคชัน MeJob และได้ปรึกษานักงานที่ปรึกษา เพื่อการพัฒนา API ทำให้ได้รับประสบการณ์ใหม่ๆ ในการทำงานมีความรับผิดชอบต่องานที่ได้รับมอบหมายและรู้ระบบการทำงานขององค์กรได้รู้จักการทำงานเป็นทีมโดยการทำงานเป็นทีมนั้น เราต้องรู้จักรับฟังความคิดเห็นของคนในทีมแล้ว นำมาปรับใช้และต้องหาความรู้ใหม่ๆ มาสร้างสรรค์งานเพื่อให้งานออกมามีคุณภาพมากขึ้น จากการปฏิบัติงานสหกิจศึกษาได้ใช้ความรู้ในการทำกราฟิกดีไซน์ ได้ฝึกทักษะการเขียนเว็บเพจโดยใช้โปรแกรม Visual Studio Code ภาษา Javascript และ Nodejs นำมาต่อยอดการพัฒนาความสามารถของแอปพลิเคชันทำให้มีประสิทธิภาพมากขึ้น โดยใช้ความรู้จากประสบการณ์ที่ได้ศึกษามา นำมาใช้ปฏิบัติงานจริง ซึ่งในการปฏิบัติงานนั้นอาจจะมีบางครั้งที่การออกแบบงานไม่ตรงกับความต้องการของลูกค้า จึงทำให้เราต้องนำชิ้นงานมาแก้ไขและปรับปรุงเพื่อพัฒนาทักษะในการออกแบบต่อไป นอกจากนี้ยังได้เรียนรู้ การอยู่ร่วมกันภายในองค์กร การรู้จักปรับปรุงและพัฒนาตนเอง ให้สามารถ อยู่ร่วมและทำงานกับผู้อื่นได้ เรียนรู้การมีมนุษยสัมพันธ์ที่ดี และมีความจำเป็นที่ต้องประสานงานกับคน ในองค์กร เพื่อให้งานสำเร็จและบรรลุผล สามารถคิดและตัดสินใจตามหลักเหตุผล ได้อย่างรอบคอบ

4.1.1 ผลการพัฒนาแอปพลิเคชันรับสมัครงาน MeJob ด้วย REST API

1. เรียกดูข้อมูลประเภทงานทั้งหมด.

Input

GET	▼	https://us-central1-myjob-c70d7.cloudfunctions.net/ijob/JobCategory/getAll	Send	▼
-----	---	--	------	---

ภาพที่ 4-1 การเรียกใช้ API แสดงข้อมูลประเภทงานทั้งหมด

output

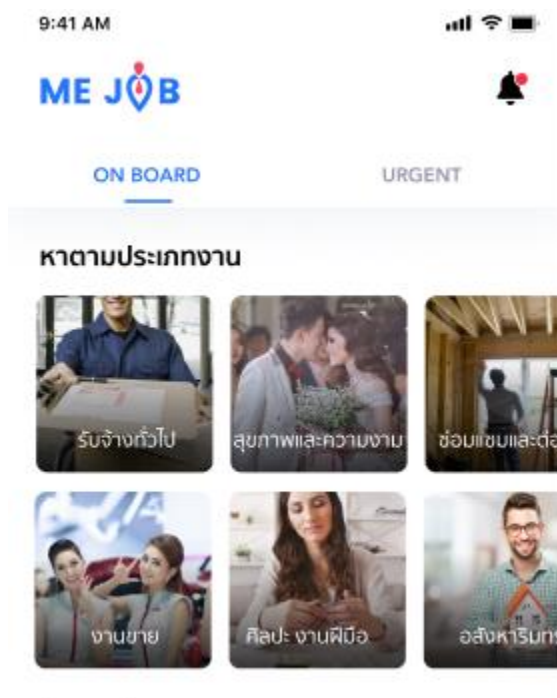
```

3      "data": [
4          {
5              "id": 1659285447392,
6              "wordSearch": [
7                  "ร",
8                  "ริ",
9                  "รับ",
10                 "รับจ",
11                 "รับจ้",
12                 "รับจ่า",
13                 "รับจ้าง",
14                 "รับจ้างท",
15                 "รับจ้างห้",
16                 "รับจ้างห้",
17                 "รับจ้างทั่ว",
18                 "รับจ้างทั่วไป",
19                 "รับจ้างทั่วไป"
20             ],
21             "updateAt": {
22                 "_seconds": 1660476349,
23                 "_nanoseconds": 331000000
24             },
25             "title": "รับจ้างทั่วไป",
26             "Description": "",
27             "createAt": {
28                 "_seconds": 1659285447,
29                 "_nanoseconds": 392000000

```

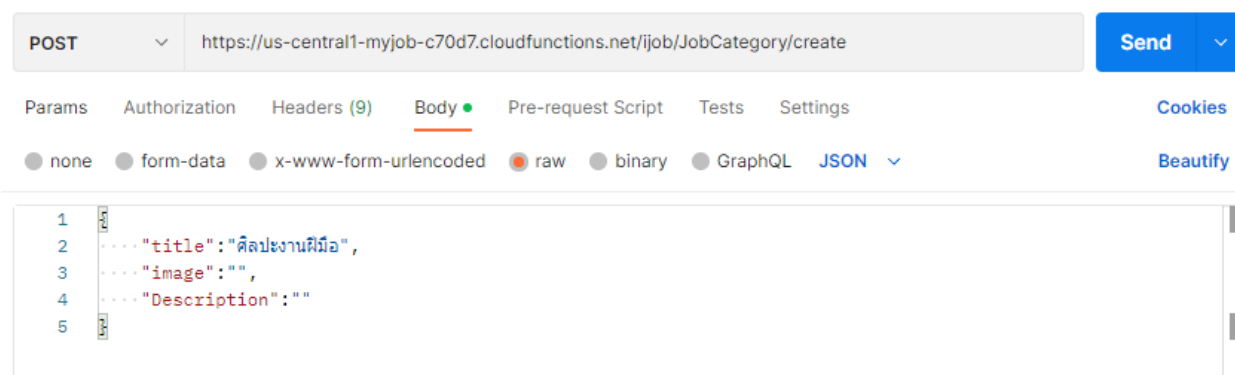
ภาพที่ 4-2 ผลลัพธ์การเรียกใช้ API แสดงข้อมูลประเภทงานทั้งหมด

Front-end



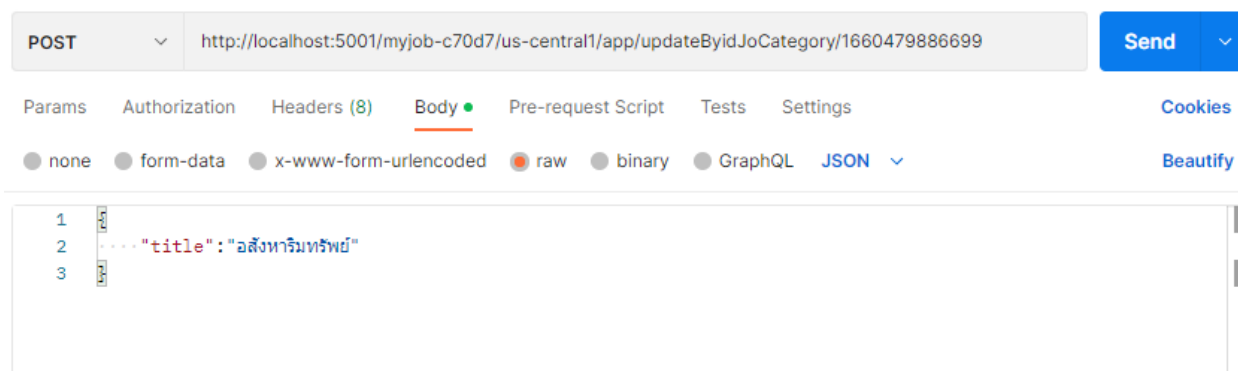
ภาพที่ 4-3 หน้าแสดงผลการเรียกใช้ API แสดงข้อมูลประเภทงานทั้งหมด

2.API การสร้างข้อมูลประเภทงาน ต้องใส่ข้อมูลที่จะบันทึกลงไปด้วย



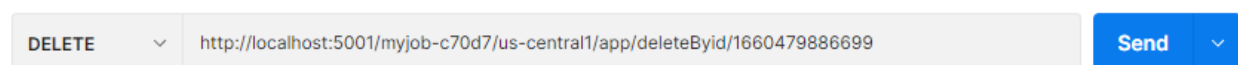
ภาพที่ 4-4 การเรียกใช้ API สร้างข้อมูลประเภทงาน

3.API การแก้ไขข้อมูล ต้องแนบไอดีและข้อมูลที่แก้ไขไปด้วย



ภาพที่ 4-5 การเรียกใช้ API แก้ไขข้อมูลประเภทงาน

4.API การลบข้อมูล ต้องแนบไอดีที่จะลบไปด้วย



ภาพที่ 4-6 การเรียกใช้ API ลบข้อมูลประเภทงาน

4.2 ประโยชน์ที่ได้รับจากการปฏิบัติงานสหกิจศึกษา

4.2.1 ประโยชน์ที่ได้รับต่อตนเอง

4.2.1.1 ได้รับความรู้และประสบการณ์จากการปฏิบัติงานสหกิจในองค์กรที่ได้มา ไปประยุกต์ใช้ในการประกอบอาชีพในอนาคต

4.2.1.2 ได้เรียนรู้และพัฒนาตนเอง

4.2.1.3 ทำให้ตนเองเป็นคนที่มีความรับผิดชอบต่องานที่ได้รับมอบหมาย

4.2.1.4 ได้ฝึกฝนตนเองเป็นบุคคลที่มีทัศนคติในการปฏิบัติงานที่ดี

4.2.1.5 ได้เรียนรู้ถึงการทำงานภายในองค์กร และการทำงานร่วมกันเป็นทีมหรือหมู่คณะ

4.2.1.6 ได้เรียนรู้วิธีการปรับตัวให้เข้ากับองค์กร และบุคลากรภายในองค์กร

4.2.1.7 ได้ฝึกฝนตนเองให้เป็นผู้ที่มีระเบียบวินัย ตรงต่อเวลา ซึ่งเป็นสิ่งสำคัญมาก

4.2.1.8 ได้ฝึกฝนตนเอง ในการปฏิบัติงาน และมีความมั่นใจในการปฏิบัติงาน

4.2.1.9 ได้ฝึกฝนความอดทนของตนเองในการรับแรงกดดันต่าง ๆ

4.2.2 ประโยชน์ที่ได้รับต่อสถานประกอบการ

4.2.2.1 มีคู่มือการใช้งาน เป็นเครื่องมือในเรียนรู้การใช้งานระบบบริหารจัดการงานสารบรรณ และบันทึกภายใน นอกเหนือจากการฝึกอบรมตามแผน

4.2.2.2 ผู้ใช้งานสามารถเรียนรู้ได้ด้วยตนเอง ลดเวลาในการศึกษาเรียนรู้และลดเวลาในการสอนงาน

4.2.2.3 สถานประกอบการมีภาพลักษณ์ที่ดี ในด้านการส่งเสริม และให้ความร่วมมือ สนับสนุนโครงการสหกิจศึกษา

4.2.3 ประโยชน์ที่ได้รับต่อมหาวิทยาลัย

4.2.3.1 เป็นการประชาสัมพันธ์ให้สถานประกอบการและบุคคลภายนอกได้รู้จักและ ยอมรับมหาวิทยาลัยมากยิ่งขึ้น เพื่อการยกระดับคุณภาพของนักศึกษา

4.2.3.2 สร้างภาพลักษณ์ที่ดีของมหาวิทยาลัย เพื่อเป็นจุดเริ่มต้นของการรับนักศึกษาเข้าสู่ โครงการสหกิจศึกษาในรุ่นต่อไป

4.2.3.3 เป็นแนวทางเลือกในการเข้าร่วมโครงการสหกิจศึกษาในรุ่นต่อไป และสามารถนำ

ประสบการณ์ต่างๆ ในการปฏิบัติงานสหกิจศึกษาไปถ่ายทอดสู่รุ่นน้องในมหาวิทยาลัยต่อไป

4.2.3.4 เกิดความร่วมมือที่ดีระหว่างมหาวิทยาลัยกับสถานประกอบการ เพื่อการประสานงานที่ราบรื่นในอนาคต

บทที่ 5

สรุปผลการดำเนินงาน และข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

5.1.1 สรุปผล

การปฏิบัติงานที่ บริษัท Kn Group เป็นระยะเวลาตั้งแต่ 20 มิถุนายน – 7 ตุลาคม 2565

งานที่ได้รับมอบหมายในส่วนของการพัฒนา API สำหรับแอปพลิเคชัน Mejob ทำให้ได้รับประสบการณ์ใหม่ๆ ในการทำงาน

มีความรับผิดชอบต่องานที่ได้รับ มอบหมายและได้รู้ระบบการทำงานขององค์กร ได้รู้จักการทำงานเป็นทีม โดยการทำงานเป็นทีมนี้ เราต้องรู้จักรับฟังความเห็นของคนในทีมแล้ว นำมาปรับใช้และต้องหาความรู้ใหม่ๆ มาสร้างสรรค์งานเพื่อให้งานออกมามีคุณภาพมากขึ้น ในการปฏิบัติงานสหกิจศึกษาได้ใช้ความรู้ในการออกแบบ ได้ฝึกทักษะการพัฒนาซอฟต์แวร์โดยใช้โปรแกรม Visual Studio Code ได้อย่างมีประสิทธิภาพมากขึ้น โดยใช้ความรู้จากประสบการณ์ที่ได้ศึกษา มานำมาใช้ปฏิบัติงานจริง ซึ่งในการปฏิบัติงานนั้นอาจจะมีบางครั้งที่การออกแบบงานไม่ตรงกับความต้องการของลูกค้า จึงทำให้เราต้องนำชิ้นงานมาแก้ไขและปรับปรุงเพื่อพัฒนาทักษะในการออกแบบต่อไป

5.1.2 ปัญหา อุปสรรค

จากการปฏิบัติงานสหกิจศึกษา ภายใต้การดูแลของพนักงานที่ปรึกษา ที่บริษัท Kn Group เป็นระยะเวลาปฏิบัติงานตั้งแต่ วันที่ 20 มิถุนายน – 7 ตุลาคม 2565 ขณะฝึก ปฏิบัติงาน ณ สถานประกอบการได้มีปัญหาดังต่อไปนี้เกิดขึ้นมายมาย ปัญหาเหล่านี้ทำให้ได้รู้จักฝึก กระบวนการคิด วิเคราะห์ และการแก้ไขปัญหาต่างๆ สรุปปัญหา อุปสรรค ดังนี้

5.1.2.1 การปฏิบัติงานในช่วงแรก ยังไม่เข้าใจในขั้นตอนการทำงานของงาน

5.1.2.2 การสื่อสารคลาดเคลื่อนเนื่องจากการทำงานเป็นแบบ (Work from Home)

5.1.3 แนวทางแก้ไข

5.1.3.1 ต้องกล้าที่จะสอบถาม ปรึกษา พี่เลี้ยงที่สอนงาน และพนักงานที่ปรึกษา เพื่อที่จะได้เรียนรู้การทำงานได้อย่างเต็มที่

5.1.3.2 ปรับทัศนคติของตนเอง และสอบถามพี่เลี้ยงที่สอนงาน และพนักงานที่ปรึกษา ก่อนการปฏิบัติงาน ไม่เข้าใจต้องถาม เพื่อลดความผิดพลาดในการทำงาน

5.1.3.3 ศึกษา/หาความรู้เพิ่มเติม เพื่อพัฒนาตนเอง และเพิ่มศักยภาพในการทำงาน

5.1.3.4 มีความขยันและตั้งใจจริงในการปฏิบัติงาน มุ่งเน้นความสำเร็จของงาน

5.2 ข้อเสนอแนะ

ข้อเสนอแนะในการจัดทำโครงการสหกิจศึกษา เพื่อเป็นประโยชน์แก่มหาวิทยาลัยและ นักศึกษา ที่จะฝึกปฏิบัติงานสหกิจศึกษาในรุ่นต่อไป ดังนี้

5.2.1 พัฒนาและเตรียมความพร้อมก่อนการออกปฏิบัติงานสหกิจศึกษา

5.2.2 เตรียมความพร้อมก่อนการออกปฏิบัติงานสหกิจศึกษา โดยการปรับการเรียนการสอนที่ เน้นการปฏิบัติเพื่อให้เกิดประสบการณ์ที่สามารถนำไปใช้ในการปฏิบัติงานสหกิจศึกษาได้มากขึ้น

5.2.3 นักศึกษาต้องหมั่นศึกษาหาความรู้เพิ่มเติมในการทำงาน ขอคำปรึกษาจากผู้ที่มีความรู้และฝึกการปฏิบัติจริง ๆ ด้วยตนเอง