

บทที่ 2

ทฤษฎีและระบบงานที่เกี่ยวข้อง

จากการศึกษาเรื่อง การพัฒนาเว็บไซต์ของคณะสัตวแพทยศาสตร์ มหาวิทยาลัยมหาสารคาม ผู้จัดทำโครงการวิจัยได้ศึกษาทฤษฎีที่เกี่ยวข้องและระบบงานที่เกี่ยวข้อง ดังนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 แบบประเมินเว็บไซต์หน่วยงานภายใน

เนื่องจากสำนักคอมพิวเตอร์ มหาวิทยาลัยมหาสารคาม ได้กำหนดคุณสมบัติของเว็บไซต์หน่วยงานภายใน เพื่อให้เว็บไซต์ของทุกหน่วยงานภายในมหาวิทยาลัยมหาสารคามเป็นไปในแนวทางเดียวกัน ดังนั้นโครงการปริญญาโทจึงพัฒนาเว็บไซต์ของคณะสัตวแพทยศาสตร์ภายใต้ข้อกำหนดตามแบบประเมินเว็บไซต์หน่วยงานภายใน[13]ที่ทางมหาวิทยาลัยได้กำหนดไว้ ดังนี้

ส่วนที่ 1 ข้อกำหนด/แนวปฏิบัติพื้นฐาน

1. การแสดงผล

- มีการแสดงผลอย่างน้อย 2 ภาษา คือ ภาษาไทย ภาษาอังกฤษ

2. URL เว็บไซต์คณะ/หน่วยงาน

- เว็บไซต์คณะ/หน่วยงานอยู่ภายใต้โดเมนเนม (Domain Name) ของมหาวิทยาลัย คือ (.msu.ac.th)

- เว็บไซต์คณะ/หน่วยงานมีการติดตั้ง SSL Certificate (<https://xxx.msu.ac.th>)

3. เกี่ยวกับหน่วยงาน

- ประวัติความเป็นมา
- วิสัยทัศน์ พันธกิจ
- โครงสร้างหน่วยงาน ผู้บริหาร อำนวยการหน้าที่
- ภารกิจ และหน้าที่รับผิดชอบของหน่วยงาน
- ข้อมูลบุคลากรภายในหน่วยงาน ประกอบด้วย ชื่อ-นามสกุล, ตำแหน่ง, อีเมล
- ข้อมูลการบริการของหน่วยงาน (หน่วยงานสนับสนุน) /หลักสูตรที่เปิดสอน (คณะ/หน่วยงานที่มีการเรียนการสอน)
- แผนกลยุทธ์หน่วยงาน (เฉพาะเว็บไซต์ฉบับภาษาไทย)
- แผนปฏิบัติราชการ (เฉพาะเว็บไซต์ฉบับภาษาไทย)

- ข้อมูลการติดต่อ ประกอบด้วย ที่อยู่ เบอร์โทรศัพท์ โทรสาร และแผนที่ตั้งหน่วยงาน
- ที่อยู่ไปรษณีย์อิเล็กทรอนิกส์ (e-Mail Address) ของบุคคลภายในหน่วยงานที่รับผิดชอบข้อมูล เช่น ผู้ดูแลเว็บไซต์(Webmaster) เป็นต้น

4. ข่าวประชาสัมพันธ์

- ข่าวประชาสัมพันธ์ทั่วไป
- ข่าวสารและประกาศของหน่วยงาน เช่น ประกาศรับสมัครงาน การจัดซื้อจัดจ้าง การจัดฝึกอบรม เป็นต้น
- ปฏิทินกิจกรรมของหน่วยงาน

5. e-Form/Online Form

- ส่วนที่ให้บริการบันทึกข้อมูลลงใน e-Form/Online Form ต่าง ๆ ของหน่วยงาน

6. เว็บลิงค์

- ส่วนงานภายใน
- หน่วยงานภายนอก (หน่วยงานราชการ สถานศึกษา รัฐวิสาหกิจ)

7. เอกสารเผยแพร่

- ส่วนที่ให้บริการสำหรับ Download แบบฟอร์มต่าง ๆ ของหน่วยงาน
- ผลงานวิจัย บทความ หรือข้อมูลเผยแพร่ความรู้

8. ผังเว็บไซต์ (Site map)

- ส่วนที่แสดงแผนผังเว็บไซต์ทั้งหมด

9. ช่องทางการติดต่อสื่อสารกับผู้ใช้บริการ

- ช่องทางสอบถามข้อมูล หรือข้อสงสัยมายัง ผู้ให้บริการ และได้รับคำตอบจากผู้ให้บริการ และผู้ให้บริการ สามารถแจ้งข่าวสาร หรือแจ้งเตือนผู้ใช้บริการ ผ่านช่องทางต่าง ๆ เช่น

ช่องทาง ถาม- ตอบ (Q & A)

ช่องทางการติดต่อหน่วยงานในรูปแบบเครือข่ายสังคมออนไลน์ (Social Networking)

ช่องทางการรับเรื่องร้องทุกข์ ร้องเรียน และช่องทางติดตามสถานะเรื่องร้องเรียน

ช่องทางแสดงความคิดเห็น หรือข้อเสนอแนะ เช่น e-mail, Web board เป็นต้น

ช่องทางการแบ่งปัน (Share) ข้อมูลไปยัง Social Networking ต่าง ๆ

10. ระบบการค้นหาข้อมูลภายในเว็บไซต์

- ฟังก์ชันค้นหาข้อมูลภายในเว็บไซต์
- แสดงผลการค้นหา ประกอบด้วย หัวข้อข้อมูลที่ทำการค้นหา พร้อมแสดงผลเป็นข้อความ รายละเอียด หรือเส้นเชื่อม (Link) สำหรับคลิกเพื่อเข้าถึงข้อมูล

11. ส่วนล่างของเว็บไซต์ (Page Footer)

- เมนูหลักในรูปแบบข้อความ
- ข้อมูลติดต่อหน่วยงาน ได้แก่ ชื่อและที่อยู่หมายเลขโทรศัพท์ หมายเลขโทรสาร และที่อยู่ไปรษณีย์อิเล็กทรอนิกส์
- เส้นเชื่อมกลับไปยังหน้าหลักของเว็บไซต์
- คำสงวนลิขสิทธิ์ (Copyright)

ส่วนที่ 2 ข้อกำหนด/แนวปฏิบัติเพิ่มเติม จำนวนรวม 3 ข้อ

1. เครื่องมือสนับสนุนการใช้งาน

มีระบบ Navigation ที่ชัดเจน ง่ายต่อการเข้าใจ

2. เครื่องมือสำหรับเก็บข้อมูลการเยี่ยมชมเว็บไซต์ (Web Analytic)

มีเครื่องมือสำหรับเก็บข้อมูล การเยี่ยมชมเว็บไซต์ของผู้ใช้บริการ เช่น จำนวนครั้ง จำนวนหน้า ความสนใจระยะเวลา เป็นต้น ทั้งนี้ เพื่อให้หน่วยงานมีข้อมูลในการวิเคราะห์พฤติกรรมกรรมการเยี่ยมชมเว็บไซต์ (Web Behavior)

3. เส้นเชื่อม (Link)

- เส้นเชื่อม (Link) มีความพร้อมใช้งาน (ใช้เครื่องมือ

<https://www.deadlinkchecker.com/> ในการตรวจสอบเส้นเชื่อม Link)

4. แบนเนอร์ขอความยินยอมจาก User ในการใช้ข้อมูลที่ได้จากคุกกี้ (Cookie Consent)

- มีการติดตั้งแบนเนอร์ขอความยินยอมจาก User ในการใช้ข้อมูลที่ได้จากคุกกี้

(Cookie Consent) ตัวอย่างโค้ดการติดตั้ง Cookie Consent

ส่วนที่ 3 แนวปฏิบัติสำหรับมหาวิทยาลัย เพื่อการพัฒนาเว็บไซต์ให้สอดคล้องการจัดอันดับมหาวิทยาลัย โดยเว็บโคมเมตริกซ์ และมาตรฐานอื่น ๆ จำนวนรวม 4 ข้อ

1. เว็บไซต์มีความยืดหยุ่นในการแสดงผล

- เว็บไซต์ต้องมีความยืดหยุ่น (Responsive) ในการแสดงผลสามารถรองรับอุปกรณ์หลากหลายรูปแบบ(ใช้เครื่องมือ <https://www.responsinator.com/>)

2. เว็บไซต์ที่มีการเผยแพร่ผลงานวิชาการ

- เว็บไซต์ที่มีการเผยแพร่ผลงานทางวิชาการจะต้องอยู่ภายใต้โดเมนเนมเดียวกันของมหาวิทยาลัย คือ .msu.ac.th

3. กำหนด Meta data ให้เอกสารดิจิทัลไฟล์

- มีการกำหนดเมตาเดตา ให้กับเอกสารดิจิทัลไฟล์ pdf doc docx ppt pptx ตลอดจนไฟล์รูปภาพต่างๆ เพื่อเป็นตัวแทน (Agent) ในการสืบค้นจากเสิร์ชเอนจิน

4. กำหนด Meta data บน Web Page

- กำหนดคำอธิบายในโค้ดหรือการกำหนดเป็นชื่อรูปภาพในคำสั่ง html
- กำหนดไตเติ้ล (title) ให้กำหนดเป็นทั้งภาษาไทยและภาษาอังกฤษ
- กำหนด Meta Language Tag
- กำหนด Meta Language Tag
- กำหนด Meta Keyword Tag
- กำหนด Meta Description Tag
- กำหนด Meta Meta Revisit-After Tag
- กำหนด Meta Robots Tag

2.1.2 RESTful API

RESTful API [2] เป็นอินเทอร์เฟซที่ระบบคอมพิวเตอร์สองระบบใช้เพื่อแลกเปลี่ยนข้อมูลผ่านอินเทอร์เน็ตซึ่งเป็นระบบที่มีมาตรฐานการสื่อสารระหว่างซอฟต์แวร์ที่ปลอดภัย เสถียร และมีประสิทธิภาพ

API (Application Programming Interface) คือส่วนต่อประสานโปรแกรมประยุกต์ กำหนดกฎที่ต้องปฏิบัติตามเพื่อสื่อสารกับระบบซอฟต์แวร์อื่นโดยนักพัฒนาเปิดเผยหรือสร้าง API เพื่อให้แอปพลิเคชันอื่นสามารถสื่อสารกับแอปพลิเคชันของตนได้ทางโปรแกรม

ไคลเอ็นต์ คือผู้ใช้ที่ต้องการเข้าถึงข้อมูลจากเว็บ โดยไคลเอ็นต์อาจเป็นบุคคลหรือระบบซอฟต์แวร์ที่ใช้ API ก็ได้ ตัวอย่างเช่น นักพัฒนาสามารถเขียนโปรแกรมที่เข้าถึงข้อมูลสภาพอากาศจากระบบสภาพอากาศ หรือคุณสามารถเข้าถึงข้อมูลเดียวกันจากเบราว์เซอร์เมื่อคุณเยี่ยมชมเว็บไซต์ รายงานสภาพอากาศได้โดยตรง

ทรัพยากร คือข้อมูลที่แอปพลิเคชันต่างๆ มอบให้แก่ไคลเอ็นต์ โดยทรัพยากรอาจเป็นรูปภาพ วิดีโอ ข้อความ ตัวเลข หรือข้อมูลประเภทใดก็ได้ ทั้งนี้เครื่องคอมพิวเตอร์ที่มอบทรัพยากรให้แก่ไคลเอ็นต์นั้นเรียกอีกอย่างว่าเซิร์ฟเวอร์ องค์กรต่างๆ ใช้ API เพื่อแบ่งปันทรัพยากรและให้บริการเว็บในขณะที่ยังคงดูแลรักษาความปลอดภัย การควบคุม และการรับรองความถูกต้องไปพร้อมกัน นอกจากนี้ API ยังช่วยให้ลูกค้าระบุได้ว่าไคลเอ็นต์ใดสามารถเข้าถึงทรัพยากรภายในที่เฉพาะเจาะจงได้

REST(Representational State Transfer) คือสถาปัตยกรรมซอฟต์แวร์ที่กำหนดเงื่อนไขว่า API ควรทำงานอย่างไร โดยแต่แรกเริ่มนั้น มีการสร้าง REST ขึ้นเพื่อเป็นแนวทางในการจัดการการสื่อสารบนเครือข่ายที่ซับซ้อน นักพัฒนา API สามารถออกแบบ API ได้โดยใช้สถาปัตยกรรมต่างๆ โดย API ที่เป็นไปตามรูปแบบสถาปัตยกรรม REST เรียกว่า REST API บริการเว็บที่ใช้สถาปัตยกรรม REST เรียกว่าบริการเว็บ RESTful คำว่า RESTful API โดยทั่วไปหมายถึง API เว็บแบบ RESTful หลักการบางประการของรูปแบบสถาปัตยกรรม REST มีดังต่อไปนี้

1. อินเทอร์เน็ตเฟิร์มแวร์แบบเดียวกันถือเป็นพื้นฐานในการออกแบบบริการเว็บ RESTful ทุกประเภท ซึ่งระบุว่าเซิร์ฟเวอร์ถ่ายโอนข้อมูลในรูปแบบมาตรฐาน ทรัพยากรที่จัดรูปแบบเรียกว่าการแทนข้อมูลใน REST โดยรูปแบบนี้อาจแตกต่างจากการแทนข้อมูลภายในของทรัพยากรบนแอปพลิเคชันเซิร์ฟเวอร์ โดยอินเทอร์เน็ตเฟิร์มแวร์แบบเดียวกันกำหนดข้อจำกัดทางสถาปัตยกรรมไว้ 4 ประการ ได้แก่

- คำขอควรระบุทรัพยากร ซึ่งสามารถทำเช่นนั้นได้โดยใช้ตัวระบุทรัพยากรรูปแบบเดียวกัน

- โคลเอ็นต์มีข้อมูลเพียงพอในการแทนข้อมูลทรัพยากรเพื่อแก้ไขหรือลบทรัพยากรดังกล่าวหากต้องการ เซิร์ฟเวอร์เป็นไปตามเงื่อนไขนี้โดยการส่งข้อมูลเมตาที่อธิบายทรัพยากรเพิ่มเติม โคลเอ็นต์ได้รับข้อมูลเกี่ยวกับวิธีการประมวลผลการแทนข้อมูลเพิ่มเติม

- เซิร์ฟเวอร์ดำเนินการเช่นนี้ได้โดยการส่งข้อความอธิบายตนเองที่มีข้อมูลเมตาเกี่ยวกับวิธีการที่โคลเอ็นต์สามารถใช้งานข้อมูลดังกล่าวได้ดีที่สุด

- โคลเอ็นต์ได้รับข้อมูลเกี่ยวกับทรัพยากรอื่นๆ ที่เกี่ยวข้องทั้งหมดที่จำเป็นสำหรับการทำงานให้เสร็จสมบูรณ์ เซิร์ฟเวอร์ดำเนินการเช่นนี้ได้โดยการส่งไฮเปอร์ลิงก์ในการแทนข้อมูลเพื่อให้ลูกค้านำสามารถค้นพบทรัพยากรเพิ่มเติมได้แบบไดนามิก

2. ความไร้สถานะ ในสถาปัตยกรรม REST ความไร้สถานะหมายถึงวิธีการสื่อสารที่เซิร์ฟเวอร์ดำเนินการตามคำขอของโคลเอ็นต์ทั้งหมดโดยไม่ขึ้นกับคำขอก่อนหน้าทั้งหมด โดยโคลเอ็นต์สามารถร้องขอทรัพยากรในลำดับใดก็ได้ และทุกคำขอจะไร้สถานะหรือแยกออกจากคำขออื่นๆ ข้อจำกัดในการออกแบบ REST API นี้บ่งบอกว่าเซิร์ฟเวอร์สามารถเข้าใจและดำเนินการตามคำขอได้อย่างสมบูรณ์ทุกครั้ง

3. ระบบที่แบ่งออกเป็นชั้น ในสถาปัตยกรรมระบบที่แบ่งออกเป็นชั้นโคลเอ็นต์สามารถเชื่อมต่อกับตัวกลางอื่นๆ ที่ได้รับอนุญาตระหว่างโคลเอ็นต์และเซิร์ฟเวอร์ได้ และจะยังคงได้รับการตอบสนองจากเซิร์ฟเวอร์ เซิร์ฟเวอร์ยังสามารถส่งต่อคำขอไปยังเซิร์ฟเวอร์อื่นได้อีกด้วย คุณสามารถออกแบบบริการเว็บ RESTful ให้ทำงานบนเซิร์ฟเวอร์หลายตัวที่มีหลายชั้นได้ เช่น ความปลอดภัย แอปพลิเคชัน และตรรกะทางธุรกิจ โดยทำงานร่วมกันเพื่อตอบสนองคำขอของโคลเอ็นต์ โดยโคลเอ็นต์ไม่สามารถมองเห็นชั้นต่างๆ เหล่านี้ได้

4. ความสามารถในการแคช บริการเว็บ RESTful รองรับการแคช ซึ่งเป็นกระบวนการจัดเก็บการตอบสนองบางส่วนบนโคลเอ็นต์หรือตัวกลางเพื่อปรับปรุงเวลาตอบสนองของเซิร์ฟเวอร์ ตัวอย่างเช่น สมมติว่าคุณเยี่ยมชมเว็บไซต์ที่มีรูปภาพส่วนหัวและส่วนท้ายทั่วไปในทุกหน้า ทุกครั้งที่คุณเยี่ยมชมหน้าเว็บไซต์ใหม่ เซิร์ฟเวอร์จะต้องส่งภาพเดิมอีกครั้ง เพื่อหลีกเลี่ยงปัญหานี้ โคลเอ็นต์จะแคชหรือจัดเก็บรูปภาพเหล่านี้หลังจากการตอบสนองครั้งแรก จากนั้นจึงใช้รูปภาพโดยตรงจากแคช ทั้งนี้บริการเว็บ RESTful ควบคุมการแคชโดยใช้การตอบสนอง API ที่ระบุว่าสามารถแคชได้หรือไม่สามารถแคชได้

5. การปรับแต่งโค้ดได้ตามความต้องการ ในรูปแบบสถาปัตยกรรม REST เซิร์ฟเวอร์สามารถขยายหรือปรับแต่งฟังก์ชันการทำงานของไคลเอ็นต์ได้ชั่วคราวโดยการถ่ายโอนโค้ดการเขียนโปรแกรมซอฟต์แวร์ไปยังไคลเอ็นต์ ตัวอย่างเช่น เมื่อคุณกรอกแบบฟอร์มลงทะเบียนบนเว็บไซต์เบราว์เซอร์ของคุณจะเน้นย้ำให้เห็นถึงข้อผิดพลาดของคุณทันที เช่น หมายเลขโทรศัพท์ที่ไม่ถูกต้อง ซึ่งระบบสามารถทำเช่นนั้นได้เนื่องจากโค้ดที่เซิร์ฟเวอร์ส่งไป

ทำงานของ RESTful API

ฟังก์ชันพื้นฐานของ RESTful API จะเหมือนกับการท่องอินเทอร์เน็ต ไคลเอ็นต์จะติดต่อกับเซิร์ฟเวอร์โดยใช้ API เมื่อต้องใช้ทรัพยากร นักพัฒนา API อธิบายวิธีการที่ไคลเอ็นต์ควรใช้ REST API ในเอกสารประกอบ API ของแอปพลิเคชันเซิร์ฟเวอร์ โดยการเรียกใช้ REST API มีขั้นตอนทั่วไปดังนี้

- ไคลเอ็นต์ส่งคำขอไปยังเซิร์ฟเวอร์ ไคลเอ็นต์ปฏิบัติตามเอกสารประกอบ API เพื่อจัดรูปแบบคำขอในลักษณะที่เซิร์ฟเวอร์เข้าใจได้

- เซิร์ฟเวอร์รับรองความถูกต้องของไคลเอ็นต์ และยืนยันว่าไคลเอ็นต์มีสิทธิ์ส่งคำขอดังกล่าว

- เซิร์ฟเวอร์รับคำขอและประมวลผลเป็นการภายใน

- เซิร์ฟเวอร์ส่งคืนการตอบสนองกลับไปยังไคลเอ็นต์ การตอบสนองมีข้อมูลที่บอกให้ลูกค้าทราบว่าคำขอดังกล่าวสำเร็จหรือไม่ การตอบสนองยังรวมถึงข้อมูลใดๆ ที่ไคลเอ็นต์ร้องขออีกด้วยคำขอของไคลเอ็นต์ RESTful API มีอะไรบ้าง

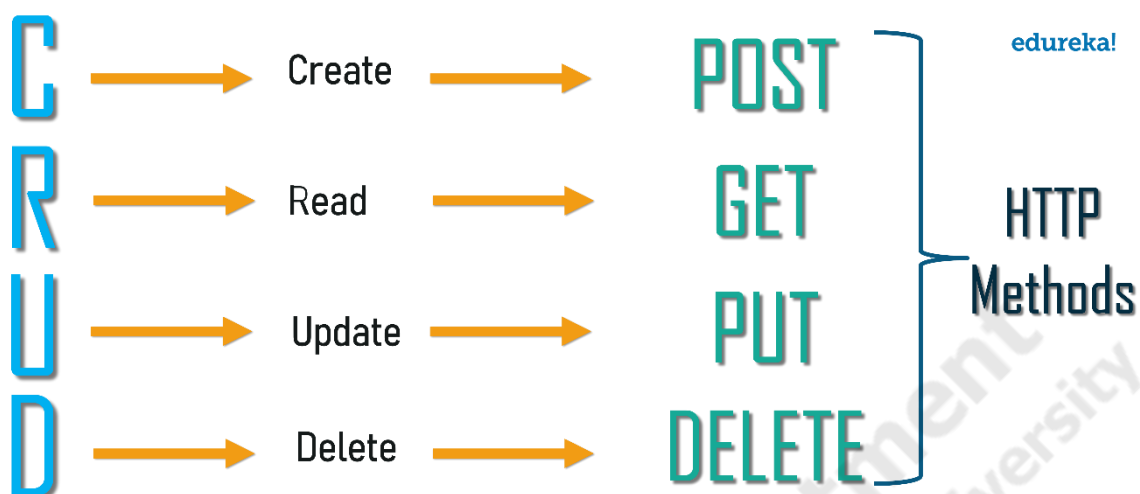
RESTful API กำหนดให้คำขอมีส่วนประกอบหลักดังต่อไปนี้

- ตัวระบุทรัพยากรที่ไม่ซ้ำกัน

เซิร์ฟเวอร์ระบุทรัพยากรแต่ละรายการด้วยตัวระบุทรัพยากรที่ไม่ซ้ำกันสำหรับบริการ REST โดยปกติแล้วเซิร์ฟเวอร์จะดำเนินการระบุทรัพยากรโดยใช้ตัวชี้แหล่งในอินเทอร์เน็ต (Uniform-Resource Locator หรือ URL) โดย URL ระบุเส้นทางไปยังทรัพยากร ทั้งนี้ URL คล้ายคลึงกับที่อยู่เว็บไซต์ที่คุณป้อนลงในเบราว์เซอร์เพื่อเข้าชมหน้าเว็บต่างๆ นอกจากนี้ URL ยังเรียกอีกอย่างว่าตำแหน่งข้อมูลคำขอ และระบุอย่างชัดเจนต่อเซิร์ฟเวอร์ถึงสิ่งที่ลูกค้าต้องการ

- วิธีการ

นักพัฒนามักใช้ RESTful API โดยใช้เกณฑ์วิธีขนส่งข้อความหลายมิติ (Hypertext Transfer Protocol หรือ HTTP) วิธีการ HTTP จะบอกให้เซิร์ฟเวอร์ทราบถึงสิ่งที่ต้องทำกับทรัพยากร โดยวิธีการ HTTP ทั่วไปมี 4 วิธีดังต่อไปนี้



ภาพประกอบที่ 2.1 ภาพอธิบายหลักการของ RESTful API

GET

ไคลเอ็นต์ใช้ GET เพื่อเข้าถึงทรัพยากรที่อยู่ URL ที่ระบุบนเซิร์ฟเวอร์ ซึ่งสามารถแคชค่าขอ GET และส่งพารามิเตอร์ในคำขอ RESTful API เพื่อสั่งให้เซิร์ฟเวอร์กรองข้อมูลก่อนส่ง

POST

ไคลเอ็นต์ใช้ POST เพื่อส่งข้อมูลไปยังเซิร์ฟเวอร์ ซึ่งรวมถึงการแทนข้อมูลพร้อมคำขอ การส่งคำขอเดียวกันหลายครั้งมีผลข้างเคียงเหมือนกับการสร้างทรัพยากรเดียวกันหลายครั้ง

PUT

ไคลเอ็นต์ใช้ PUT เพื่ออัปเดตทรัพยากรที่มีอยู่บนเซิร์ฟเวอร์ การส่งคำขอ PUT เดียวกันหลายครั้งในบริการเว็บ RESTful จะให้ผลลัพธ์เหมือนกัน ซึ่งแตกต่างจาก POST

DELETE

ไคลเอ็นต์ใช้คำขอ DELETE เพื่อลบทรัพยากรออก โดยคำขอสามารถเปลี่ยนสถานะเซิร์ฟเวอร์ได้ อย่างไรก็ตาม หากผู้ใช้ไม่มีการรับรองความถูกต้องที่เหมาะสม คำขอก็จะล้มเหลว ส่วนหัว HTTP

ส่วนหัวของคำขอคือข้อมูลเมตาที่แลกเปลี่ยนระหว่างไคลเอ็นต์และเซิร์ฟเวอร์ ตัวอย่างเช่น ส่วนหัวของคำขอจะระบุรูปแบบของคำขอและการตอบกลับ ให้ข้อมูลเกี่ยวกับสถานะคำขอ คำขอ REST API อาจรวมถึงข้อมูลสำหรับวิธีการ POST, PUT และ HTTP อื่นๆ เพื่อให้ทำงานได้สำเร็จ

ข้อดี RESTful API ต่างๆ ดังต่อไปนี้

- ความสามารถในการปรับขนาด

ระบบที่ใช้ REST API สามารถปรับขนาดได้อย่างมีประสิทธิภาพเนื่องจาก REST ปรับการโต้ตอบระหว่างไคลเอ็นต์กับเซิร์ฟเวอร์ให้เหมาะสม ความไร้สถานะช่วยจัดการโหลดเซิร์ฟเวอร์เนื่องจากเซิร์ฟเวอร์ไม่จำเป็นต้องเก็บข้อมูลคำขอของไคลเอ็นต์ในอดีต การแคชที่มีการจัดการเป็นอย่างดี

ดีบางส่วนหรือทั้งหมดจะช่วยลดการโต้ตอบระหว่างไคลเอนต์กับเซิร์ฟเวอร์บางส่วน คุณสมบัติเหล่านี้ทั้งหมดจะสนับสนุนความสามารถในการปรับขนาดโดยไม่ทำให้เกิดปัญหาคอขวดในการสื่อสารซึ่งลดประสิทธิภาพการทำงาน

- ความยืดหยุ่น

บริการเว็บ RESTful รองรับการแยกไคลเอนต์และเซิร์ฟเวอร์โดยสมบูรณ์ ซึ่งลดความซับซ้อนและแยกส่วนประกอบเซิร์ฟเวอร์ต่างๆ เพื่อให้แต่ละส่วนสามารถพัฒนาได้อย่างอิสระ ทั้งนี้การเปลี่ยนแปลงแพลตฟอร์มหรือเทคโนโลยีที่แอปพลิเคชันเซิร์ฟเวอร์ไม่ส่งผลกระทบต่อแอปพลิเคชันไคลเอนต์ความสามารถในการแบ่งชั้นการทำงานของแอปพลิเคชันช่วยเพิ่มความยืดหยุ่นยิ่งขึ้นไปอีก ตัวอย่างเช่นนักพัฒนาสามารถเปลี่ยนแปลงพื้นฐานข้อมูลได้โดยไม่ต้องเขียนตรรกะของแอปพลิเคชันขึ้นใหม่ความไม่ขึ้นกับระบบใด

REST API ไม่ขึ้นอยู่กับเทคโนโลยีที่ใช้ คุณจึงสามารถเขียนแอปพลิเคชันไคลเอนต์และเซิร์ฟเวอร์ในภาษาการเขียนโปรแกรมต่างๆ ได้โดยไม่กระทบต่อการออกแบบ API นอกจากนี้ คุณยังสามารถเปลี่ยนเทคโนโลยีพื้นฐานในทั้งสองฝั่งได้โดยไม่มีผลกระทบต่อสื่อสารอีกด้วย

2.1.3 JavaScript

JavaScript [3] คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินการไปที่ละคำสั่ง" (interpret) หรือเรียกว่า ออบเจกต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server)

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้ และได้ปรับปรุง Live Script ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript Java Script สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่นต่าง ๆ มากมาย และยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

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

JavaScript ทำอะไรได้บ้าง

1. JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่าย ๆ ได้ โดยไม่ต้องพึ่งภาษาอื่น
2. JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่นเมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็ยังสามารถสั่งให้เปิดหน้าต่างใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น
3. JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่าย ๆ
4. JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์
5. JavaScript สามารถใช้ในการตรวจสอบผู้ใช้ได้เช่น ตรวจสอบว่าผู้ใช้ ใช้ web – browser อะไร
6. JavaScript สร้าง Cookies (เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เอง) ได้

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าคุณจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side script) ดังนั้นจึงต้องใช้นเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ตาม จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุนโดยเฉพาะเช่นกัน แต่ไม่เป็นที่นิยมนัก)

2.1.1 Angular

Angular [4] คือ หนึ่งใน Front-end framework ที่ได้รับความนิยมสูงที่สุดในปัจจุบัน พัฒนาโดย Google เพื่อนำมาใช้ในการสร้างโปรเจกแบบ SPA หรือ Single Page Application แปลตรงตัวก็

คือ application ที่มีเพียง page เดียว โดยที่ client จะติดต่อกับ server ด้วยการเรียก AJAX ไปที่ Restful API ของ server

AngularJS คือ JavaScript Framework รูปแบบหนึ่ง que พัฒนามาจาก Google หน้า que ของมัน คือเป็น engine ที่ใช้ควบคุมใน ส่วน front end ของเว็บได้ดี ส่วนการจัดการ template และการ binding data object data แลมยังมีส่วนที่ สามารถเขียน plugin เพิ่มเติมได้ ส่วน plugin ถือเป็น จุดเด่นที่สุดของ jQuery ที่ประสบความสำเร็จมาก

Angular [5] เป็นส่วนหนึ่งของ MEAN Stack เป็นชุดของเฟรมเวิร์คสำหรับการพัฒนาแอปพลิเคชันที่ได้รับความนิยมอย่างมากในปัจจุบัน MEAN Stack โดย MEAN ย่อมาจาก

M : MongoDB เป็น ฐานข้อมูลแบบ NoSQL เก็บข้อมูลแบบอบเจ็กต์ ไม่ได้เก็บข้อมูลใน ตาราง

E : Express เป็น Backend Framework ใช้สำหรับพัฒนาแอปพลิเคชัน ฝั่งเซิร์ฟเวอร์

A : Angular เป็น Frontend Framework ใช้สำหรับพัฒนาแอปพลิเคชัน ฝั่งไคลเอนต์

N : Node.js เป็น Runtime Environment ที่ประกอบด้วย JavaScript Engine V8 ของ Google Chrome ครอบด้วย C++ เพื่อให้สามารถใช้งาน JavaScript ในที่ใดก็ได้ไม่ จำเป็นต้องใช้งานบน Browser

จุดเด่นของ Angular

1. พัฒนาต่อเนื่องโดยทีมงานของ Google และปัจจุบันพัฒนาถึง Angular8
2. มีโครงสร้างที่ดี แบ่งส่วนประกอบของแอปพลิเคชัน ออกเป็น component ย่อย ๆ ทำให้ง่ายต่อการพัฒนาและแก้ไข
3. รูปแบบที่สร้างเว็บเพจแบบไดนามิก เนื้อหาบนเว็บเพจจึงสามารถเปลี่ยนแปลงตาม เงื่อนไขที่แตกต่างกัน
4. สามารถใช้งานข้ามแพลตฟอร์มได้สามารถใช้ Angular บน windows Mac Linux
5. รองรับการทำงานกับ Browser ขึ้นมาได้ทุกตัว เช่น Microsoft Edge, Google Chrome, Safari, FireFox, Opera

Versions of Angular

1. AngularJS เวอร์ชันแรกของ Angular
2. Angular2 พัฒนามาจาก AngularJS ไม่สามารถใช้กับรุ่นแรกได้
3. Angular4 พัฒนามาจาก Angular2 และ Angular3 Angular3 ถูกยกเลิกไป สามารถใช้ร่วมกับ Angular2 ได้
4. Angular 5 พัฒนามาจาก Angular 4 สามารถใช้ร่วมกับ Angular 2 และ Angular 4
5. Angular 6 พัฒนามาจาก Angular 5 สามารถใช้ร่วมกับ Angular2 Angular 4 และ Angular 5

6. Angular 7 พัฒนามาจาก Angular6

7. Angular 8 ปัจจุบันพัฒนามาจาก Angular 7

2.1.2 PHP

PHP [6] ย่อมาจาก PHP Hypertext Preprocessor แต่เดิมนย่อมาจาก Personal Home Page Tools

PHP คือภาษาคอมพิวเตอร์จำพวก scripting language ภาษาจำพวกนี้คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่า script และเวลาใช้งานต้องอาศัยตัวแปรชุดคำสั่ง ตัวอย่างของภาษาสคริปต์ก็ เช่น JavaScript , Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า server-side หรือ HTML - embedded scripting language นั่นคือในทุกๆ ครั้งก่อนที่เครื่องคอมพิวเตอร์ซึ่งให้บริการเป็น Web server จะส่งหน้าเว็บเพจที่เขียนด้วย PHP ให้เรา มันจะทำการประมวลผลตามคำสั่งที่มีอยู่ให้เสร็จเสียก่อน แล้วจึงค่อยส่งผลลัพธ์ที่ได้ให้เรา ผลลัพธ์ที่ได้นั้นก็คือเว็บเพจที่เราเห็นนั่นเอง ถือได้ว่า PHP เป็นเครื่องมือที่สำคัญชนิดหนึ่งซึ่งช่วยให้เราสามารถสร้าง Dynamic Web pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

PHP เป็นผลงานที่เติบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับ หรือ Open Source ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็ว และแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Web server ระบบปฏิบัติการอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลายๆตัวบนระบบปฏิบัติการอย่างเช่น Windows 95/98/NT เป็นต้น

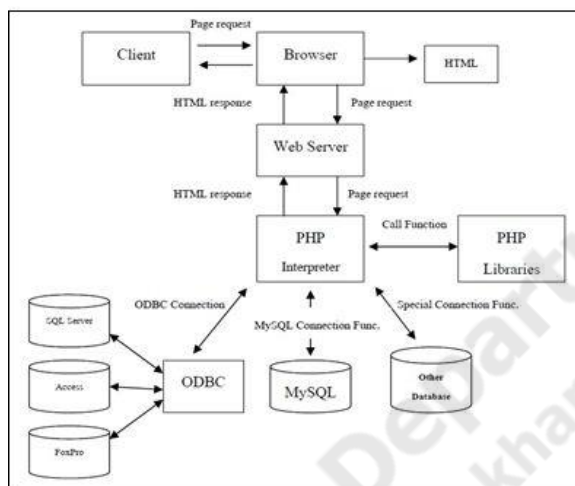
ลักษณะเด่นของ PHP

1. ใช้ได้ฟรี
2. PHP เป็นโปรแกรมวิ่งข้าง Sever ดังนั้นขีดความสามารถไม่จำกัด
3. Conlatfun นั่นคือPHP วิ่งบนเครื่อง UNIX, Linux, Windows ได้หมด
4. เรียนรู้ง่าย เนื่องจาก PHP ผ่งเข้าไปใน HTML และใช้โครงสร้างและไวยากรณ์ภาษาง่าย ๆ
5. เร็วและมีประสิทธิภาพ โดยเฉพาะเมื่อใช้กับ Apach Xerve เพราะไม่ต้องใช้โปรแกรมจากภายนอก
6. ใช้ร่วมกับ XML ได้ทันที
7. ใช้กับระบบแฟ้มข้อมูลได้
8. ใช้กับข้อมูลตัวอักษรได้อย่างมีประสิทธิภาพ

9. ใช้กับโครงสร้างข้อมูล แบบ Scalar, Array, Associative array

10. ใช้กับการประมวลผลภาพได้

หลักการการทำงานของ PHP [7]



ภาพประกอบที่ 2.2 หลักการทำงานของ PHP

1. จากโคลเอนต์จะเรียกไฟล์ PHPscript ผ่านทางโปรแกรมบราวเซอร์ (Internet Explore)
2. บราวเซอร์จะส่งคำร้อง (Request) ไปยังเว็บเซิร์ฟเวอร์ผ่านทางเครือข่ายอินเทอร์เน็ต
3. เมื่อเว็บเซิร์ฟเวอร์รับคำร้องขอจากบราวเซอร์แล้วก็จะนำสคริปต์ PHP ที่เก็บอยู่ในเซิร์ฟเวอร์มาประมวลผลด้วยโปรแกรมแปลภาษา PHP ที่เป็นอินเทอร์พรีเตอร์
4. กรณีที่php script มีการเรียกใช้ข้อมูลก็จะติดต่อกับฐานข้อมูลต่างๆผ่านทาง ODBC Connection ถ้าเป็นฐานข้อมูลกลุ่ม Microsoft SQL Server, Microsoft Access, FoxProหรือใช้ Function Connection ที่มีอยู่ใน PHP Library ในการเชื่อมต่อฐานข้อมูลเพื่อดึงข้อมูลออกมาหลังจากแปลสคริปต์ PHP เสร็จแล้วจะได้รับไฟล์ HTML ใหม่ที่มีแต่แท็กHTMLไปยัง Web Server
5. Web Server ส่งไฟล์ HTML ที่ได้ผ่านการแปลแล้วกลับไปยังบราวเซอร์ที่ร้องขอผ่านทางเครือข่ายอินเทอร์เน็ต
6. บราวเซอร์รับไฟล์ HTML ที่เว็บเซิร์ฟเวอร์ส่งมาให้แปล HTML แสดงผลออกมาทางจอภาพเป็นเว็บเพจโดยใช้ตัวแปลภาษา HTML ที่อยู่ในบราวเซอร์ซึ่งเป็นอินเทอร์พรีเตอร์เช่นเดียวกัน

2.1.3 Slim

Slim [8] เป็น Framework ของ PHP ที่มีน้ำหนักเบาและทรงพลัง สามารถใช้ได้อย่างรวดเร็ว Slim Framework จะวางคู่กับเฟรมเวิร์คขนาดใหญ่ เช่น Laravel, CodeIgniter เป็นต้น นอกจากความเบาแล้ว ยังสามารถสร้าง API ด้วย PHP ได้อย่างรวดเร็ว

Slim Framework สร้างขึ้นโดย Josh Lockhart นักพัฒนาของ newmediacampaigns.com และเขาเป็น "ผู้อยู่เบื้องหลัง" PHP ที่ The Right Way Josh Lockhart กล่าวว่า Slim Framework คือ “เฟรมเวิร์คโคร PHP ที่ช่วยให้นักพัฒนา PHP เขียนเว็บแอปพลิเคชันและ API ได้อย่างรวดเร็ว และง่ายตาย ให้คิดว่ามันเป็นชุดเครื่องมือหลักที่นักพัฒนาสามารถสร้างสิ่งที่น่าอัศจรรย์ได้”

Slim เป็นเฟรมเวิร์คขนาดเล็กที่เน้นความต้องการพื้นฐาน เช่น การสร้างเว็บแอปพลิเคชัน API (รับคำขอ HTTP และส่งคำขอพร้อมกับผลลัพธ์ของการตอบสนอง)

ข้อดีหลักของ Slim framework คือ

1. HTTP Router
2. ตัวเชื่อมระหว่าง APP และ OS (Middleware)
3. ป้องกันการสร้างอะไรบางอย่าง (Dependency Injection)
4. สนับสนุน PSR-7

2.1.4 WYSIWYG

WYSIWYG [9] ย่อมาจาก What You See Is What You Get (วอท ยู ซี อีส วอท ยู เก็ท) แปลว่า คุณเห็นอย่างไรคุณก็ได้รับอย่างนั้น คือ ลักษณะของเอกสารและภาพกราฟฟิกที่คุณเห็นในจอคอมพิวเตอร์ เมื่อพิมพ์ออกมาแล้วลักษณะของเอกสารหรือภาพกราฟฟิกที่คุณได้ก็จะมีเหมือนกับภาพที่ปรากฏในจอคอมพิวเตอร์ ถือเป็นสมรรถนะอย่างหนึ่งของโปรแกรมประเภทประมวลผลคำ หรือโปรแกรมการจัดพิมพ์ ซึ่งสามารถแสดงตัวอย่างเอกสารก่อนการพิมพ์ และสามารถทำให้เอกสารที่พิมพ์ออกมามีลักษณะเหมือนในจอคอมพิวเตอร์ นั่นคือ ถ้ามองเห็นบนจอภาพอย่างไร สิ่งพิมพ์ออกมา ก็จะได้ผลอย่างที่เราเห็นนั้น

ตัวอย่างเช่น ในการสร้างและออกแบบ web site หรือ web page มักจะใช้โปรแกรมประเภท Web Design ซึ่งมีคุณสมบัติในการใช้งานในแบบ WYSIWYG เพื่อให้การทำ web site หรือ web page นั้น ทำมาอย่างไร ก็แสดงผลอย่างนั้น ไม่ผิดเพี้ยนไป อีกทั้งยังง่ายต่อการใช้ เพราะผู้ใช้ไม่จำเป็นต้องมานั่งเขียนโค้ด HTML เอง แบบว่าไม่มีความรู้เรียกการเขียนโปรแกรมก็ใช้ได้ เช่น โปรแกรม Dreamweaver , WYSIWYG Web Builder, FrontPage และ XSitePro เป็นต้น

2.1.5 MySQL

MySQL [10] เป็นระบบการจัดการฐานข้อมูลแบบเชิงสัมพันธ์โอเพนซอร์ส ซึ่งเป็นระบบฐานข้อมูลที่นิยมสำหรับแอปพลิเคชันเว็บ และสามารถทำงานบนแพลตฟอร์มใดก็ได้ MySQL ทำงานเป็นดาต้าเบสเซอร์ฟเวอร์ และอนุญาตให้ผู้ใช้หลายคนจัดการและสร้างฐานข้อมูลจำนวนมาก มันเป็นส่วนประกอบสำคัญใน LAMP (LAMP ย่อมาจาก Linux, Apache, MySQL และ PHP) เมื่อความ

ต้องการใหม่และแตกต่างกันเกิดขึ้นกับอินเทอร์เน็ต MySQL กลายเป็นแพลตฟอร์มทางเลือกสำหรับนักพัฒนาเว็บไซต์และเว็บแอปพลิเคชัน เนื่องจากได้รับการออกแบบมาเพื่อประมวลผลการค้นหาล้าน

SQL (Structured Query Language) [11] เป็นภาษาโปรแกรมสำหรับจัดเก็บและประมวลผลข้อมูลในฐานข้อมูลแบบเชิงสัมพันธ์ ฐานข้อมูลแบบเชิงสัมพันธ์เก็บข้อมูลในรูปแบบตารางที่มีแถว และคอลัมน์ที่เป็นตัวแทนของหมวดข้อมูลที่แตกต่างกันและความสัมพันธ์ต่างๆ ระหว่างค่าข้อมูล สามารถใช้คำสั่ง SQL ในการจัดเก็บ ปรับปรุง ลบ ค้นหา และดึงข้อมูลจากฐานข้อมูล นอกจากนี้ยังสามารถใช้ SQL ในการรักษาและเพิ่มประสิทธิภาพการทำงานของฐานข้อมูล

SQL เป็นภาษามาตรฐานสำหรับการสร้างและการจัดการฐานข้อมูล MySQL เป็นโปรแกรมฐานข้อมูลแบบเชิงสัมพันธ์ที่ใช้คำสั่ง SQL ในขณะที่คำสั่ง SQL จะถูกกำหนดโดยมาตรฐานสากลซอฟต์แวร์ MySQL ผ่านการอัปเดตและการปรับปรุงอย่างต่อเนื่อง

คำสั่ง SQL

แบบสอบถามภาษามีโครงสร้าง (SQL) มีคำสั่งเป็นคำหลักเฉพาะหรือคำสั่ง SQL ที่นักพัฒนาใช้ในการจัดการกับข้อมูลที่เก็บไว้ในฐานข้อมูลแบบเชิงสัมพันธ์ สามารถจัดหมวดหมู่คำสั่ง SQL ดังต่อไปนี้

ภาษานิยามข้อมูล

ภาษานิยามข้อมูล (DDL) หมายถึงคำสั่ง SQL ที่ออกแบบโครงสร้างฐานข้อมูล วิศวกรฐานข้อมูลใช้ DDL เพื่อสร้างและแก้ไขวัตถุฐานข้อมูลตามความต้องการของธุรกิจ

ภาษาแบบสอบถามข้อมูล (DQL) ประกอบด้วยคำสั่งสำหรับการดึงข้อมูลที่เก็บไว้ในฐานข้อมูลแบบเชิงสัมพันธ์ การใช้คำสั่ง SELECT เพื่อกรอง และส่งกลับผลลัพธ์เฉพาะจากตาราง SQL

ภาษาการจัดการข้อมูล (DML) เขียนข้อมูลใหม่หรือปรับเปลี่ยนระเบียบที่มีอยู่ในฐานข้อมูลแบบเชิงสัมพันธ์ ตัวอย่างเช่น แอปพลิเคชันใช้คำสั่ง INSERT เพื่อเก็บบันทึกใหม่ในฐานข้อมูล

ภาษาควบคุมข้อมูล (DCL) เพื่อจัดการหรืออนุญาตการเข้าถึงฐานข้อมูลสำหรับผู้ใช้รายอื่น ตัวอย่างเช่นพวกเขาสามารถใช้คำสั่ง GRANT เพื่ออนุญาตให้บางแอปพลิเคชันจัดการกับหนึ่งตารางหรือมากกว่าได้

ภาษาควบคุมการทำธุรกรรม (TCL) เพื่อเปลี่ยนแปลงฐานข้อมูลโดยอัตโนมัติ ตัวอย่างเช่นฐานข้อมูลใช้คำสั่ง ROLLBACK เพื่อยกเลิกการทำธุรกรรมที่ผิดพลาด

2.1.1 Visual Studio Code

Visual Studio Code หรือ VS Code [12] เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและ

ปรับแต่งโค้ด จากค่ายไมโครซอฟท์ มีการพัฒนาออกมาในรูปแบบของ Open Source จึงสามารถนำมาใช้งานได้แบบฟรี ๆ ที่ต้องการความเป็นมืออาชีพซึ่ง Visual Studio Code นั้น เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้ นำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือส่วนขยายต่าง ๆ ให้เลือกใช้อย่างมากมาย ไม่ว่าจะเป็น

1. การเปิดใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++, C#, Java, Python, PHP หรือ Go
2. Themes
3. Debugger
4. Commands

ความแตกต่างระหว่าง VS Code และ Visual Studio คือ VS Code ได้ทำการตัดในส่วนของ GUI designer ออกเหลือแค่ Editor จึงทำให้ตัวโปรแกรมนั้นค่อนข้างเบากว่า Visual Studio เป็นอย่างมาก VS Code สามารถนำมาใช้งานได้ฟรี รองรับการทำงานข้ามแพลตฟอร์ม

2.1.2 การออกแบบฐานข้อมูลด้วยแผนภาพ E-R (Entity-Relationship Diagrams)

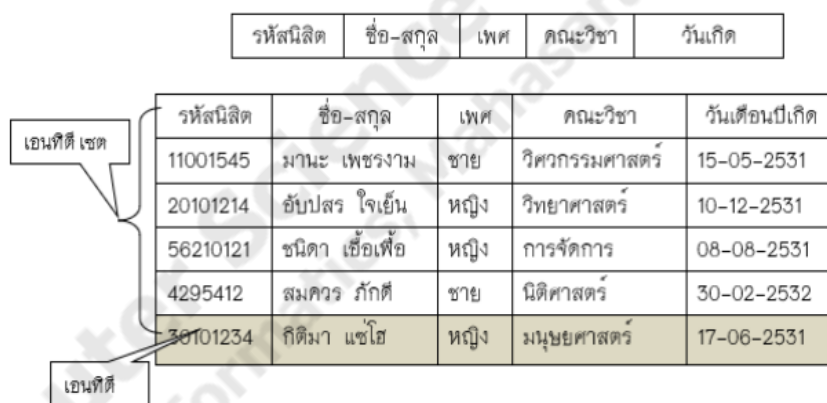
ในปี ค.ศ. 1976 Peter Chen ได้พัฒนาแบบจำลอง E-R [17] (Entity-Relationship Model , ERM) ขึ้นมาใช้เป็นเครื่องมือในการนำเสนอโครงสร้างของฐานข้อมูลใน ระดับ ความคิด (Conceptual Level) ออกมาในลักษณะของแผนภาพ (Diagram) ที่เข้าใจได้ ง่ายในการสื่อความหมายระหว่างนักออกแบบฐานข้อมูล และผู้ใช้ข้อมูลเกี่ยวกับ ความสัมพันธ์ของเอนทิตีกับเอนทิตี และความสัมพันธ์ของเอนทิตีกับแอททริบิวต์ และ นับจากนั้นเป็นต้นมา แผนภาพ E-R ก็ได้มีการนำไปใช้งานอย่างกว้างขวาง มีการ พัฒารูปแบบที่หลากหลาย แต่รูปแบบของแผนภาพ E-R ที่หลากหลายดังกล่าว ล้วนอยู่บนพื้นฐานแนวความคิด

แผนภาพ E-R (E-R Diagram) เป็นแบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพการอธิบายโครงสร้างและความสัมพันธ์ของข้อมูลความสัมพันธ์ของเอนทิตี ที่ช่วยในการออกแบบฐานข้อมูล และได้รับความนิยมเป็นอย่างมากแผนภาพ E-R เป็นแบบจำลองเชิงแนวคิด (Conceptual Data Model) ที่แสดงออกมาในลักษณะของแผนภาพ โดยใช้หลักการจากโมเดลฐานข้อมูลเชิงสัมพันธ์ในการแสดงลักษณะโดยรวมของข้อมูลในระบบช่วยสื่อสารให้เกิดความเข้าใจร่วมกันระหว่างผู้วิเคราะห์และผู้ใช้ได้เป็นอย่างดีแผนภาพที่นิยมใช้ในการนำเสนอโครงสร้างฐานข้อมูลที่นิยมคือแผนภาพ E-R ซึ่งประกอบไปด้วยเอนทิตีแอททริบิวต์ของแต่ละเอนทิตี ความสัมพันธ์ระหว่างเอนทิตี และ ดีกรี ของ ความสัมพันธ์ (Degree Of A Relationship) แผนภาพ E-R มีความสำคัญต่อการพัฒนาระบบงานฐานข้อมูล แอปพลิเคชัน (Applications) ต่าง ๆ ที่

ต้องการการเก็บข้อมูลอย่างมีระบบ แผนภาพ E-R จึงใช้เพื่อ เป็นเอกสารในการสื่อสารระหว่างนักออกแบบระบบ และนักพัฒนาระบบ เพื่อให้ สื่อสารอย่างตรงกัน และเป็นสากล

1. องค์ประกอบของแผนภาพ E-R

1.1 เอนทิตี (Entity) เอนทิตีหมายถึงสิ่งที่มีอยู่จริง จับต้องได้ หรืออาจจะเป็นจินตภาพที่แสดงความเป็นหนึ่งเดียว ซึ่งเมื่อกล่าวถึงแล้วทุกคนเข้าใจตรงกัน แต่โดยทั่วไปแล้วเอนทิตี มักจะ อยู่ในรูปของนาม ซึ่งอาจเป็นสิ่งที่ป็นรูปธรรมคือสามารถมองเห็นได้ด้วยตา หรืออยู่ในรูปของนามธรรมคือไม่สามารถมองเห็นได้ด้วยตาก็ได้ เช่น บุคคล (นิสิต อาจารย์ ลูกค้า พนักงาน คนไข้) สถานที่ (โรงเรียน ห้องเก็บสินค้า คลังสินค้า ร้านค้า) วัตถุ (หนังสือ เครื่องจักรกล เครื่องยนต์ สินค้า วัตถุดิบ) เหตุการณ์ (การยืม การคืน ลงทะเบียน การฝาก-การถอนเงิน) หรือ แนวคิด (วัน วิชา บัญชี หลักสูตร ความชำนาญ) ที่ทำให้เกิดกลุ่มของข้อมูลที่ต้องการจัดเก็บรวมทั้งสามารถบ่งชี้ถึงความเป็น เอกลักษณ์เฉพาะตัวได้ เอนทิตีมีคุณสมบัติได้หลายอย่าง และเอนทิตีเป็นสิ่งที่ผู้ใช้งาน ฐานข้อมูลจะต้องเกี่ยวข้องกับเมื่อมีการออกแบบระบบฐานข้อมูล สัญลักษณ์รูปสี่เหลี่ยมผืนผ้าแทนเอนทิตีหนึ่งเอนทิตีและมีชื่อเอนทิตีกำกับอยู่ ภายใน ชื่อที่ใช้จะต้องเป็น คำนาม และโดยปกติมักจะเขียน ตัวพิมพ์ใหญ่



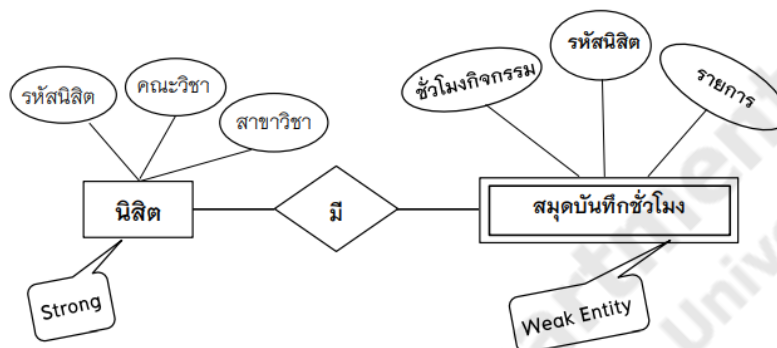
ภาพประกอบที่ 2.3 เอนทิตีและเอนทิตีเซต

เอนทิตีสามารถแบ่งออกได้หลายประเภทด้วยกัน เช่น

1) เอนทิตีปกติ (Strong Entity หรือ Regular Entity) เป็นเอนทิตีที่มี คุณสมบัติเฉพาะ (Identity) ในตัวเอง สามารถเกิดขึ้นได้ด้วยตัวเอง การคงอยู่ของ เอนทิตีจะไม่ขึ้นกับเอนทิตีอื่น เช่น นิสิต หรือ อาจารย์ หรือ สินค้า สัญลักษณ์ที่ใช้คือ รูปสี่เหลี่ยมผืนผ้า

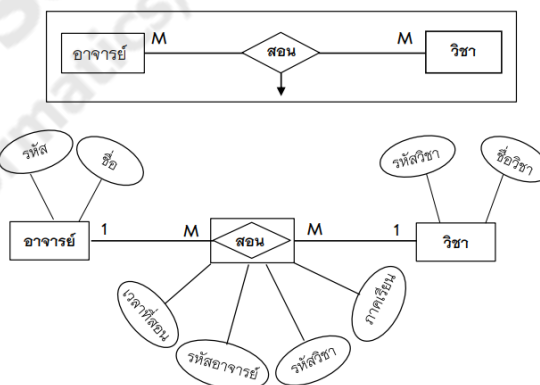
2) เอนทิตีอ่อนแอ (Weak Entity) เป็นเอนทิตีที่ไม่สามารถเกิดขึ้นได้ ตามลำพัง จะขึ้นอยู่กับเอนทิตีอื่น จะมีคีย์หลักจากการสืบทอดเอนทิตีที่อ้างอิงอยู่มา ใช้เป็นคีย์หลักหรือส่วนหนึ่งของคีย์หลัก และจะถูกลบออกไปด้วยเมื่อเอนทิตีหลักถูก ลบ สัญลักษณ์ที่ใช้คือรูปสี่เหลี่ยมผืนผ้าเส้นคู่ เอนทิตีทั้ง 2 ประเภทอธิบายได้ดังนี้ เช่น ในมหาวิทยาลัยแต่ละแห่ง นิสิตแต่ละ คนจะมี รหัสนิสิต ชื่อ-สกุล หมายเลขโทรศัพท์ ที่ไม่ซ้ำกัน ดังนั้นเอนทิตีนิสิตจะ จัดเป็น เอนทิตีปกติ และนิสิตในมหาวิทยาลัยแต่ละคน จะมี

สมุดบันทึกชั่วโมงกิจกรรม นิสิตแต่ละคนอาจจะมีรายการกิจกรรม หรือมีชั่วโมงกิจกรรมที่เหมือนกันหรือไม่ เหมือนกันก็ได้ ดังนั้นถ้าไม่มีเอนทิตินิสิต ก็จะไม่สามารถทราบว่านิสิตคนใดทำ กิจกรรมอะไร จำนวนชั่วโมงกิจกรรมเป็นเท่าใด ดังนั้นเอนทิตินิสิตกับสมุดบันทึกชั่วโมง กิจกรรม จึงจัดเป็นเอนทิตีที่อ่อนแอ เพราะเอนทิตินิสิตจะคงอยู่ได้ต้องอาศัยเอนทิตี นิสิต



ภาพประกอบที่ 2.4 เอนทิตีปกติ (Strong Entity) และเอนทิตีที่อ่อนแอ (Weak Entity)

3) คอมโพสิตเอนทิตี หรือเอนทิตีเชิงความสัมพันธ์ (Composite / Associate Entity) เป็นเอนทิตีที่สร้างขึ้นมาแปลงความสัมพันธ์ของเอนทิตีสองเอนทิตีที่มีความสัมพันธ์แบบ $M : N$ ให้เป็นแบบ $1 : M$ เพื่อให้เข้าใจได้ง่าย โดยการนำเอาคีย์ หลักของเอนทิตีทั้งสองมารวมกันกับแอททริบิวต์อื่นๆ ที่สนใจ สัญลักษณ์ที่ใช้คือ สี่เหลี่ยมผืนผ้าที่ภายในมีรูปสี่เหลี่ยมขนมเปียกปูน เช่น อาจารย์หนึ่งคนสอนได้หลาย วิชา และวิชาหนึ่งรายวิชามีอาจารย์สอนได้หลายคน



ภาพประกอบที่ 2.5 คอมโพสิตเอนทิตี หรือเอนทิตีเชิงความสัมพันธ์

1.2 แอททริบิวต์ หรือ พรอพเพอร์ตี (Property)

เอนทิตีหนึ่งๆ จะประกอบไปด้วยคุณสมบัติ หรือ ลักษณะที่แตกต่างกัน ออกไป ข้อมูลที่แสดงถึงคุณลักษณะของเอนทิตีเรียกว่าแอททริบิวต์ ดังนั้นความหมาย ของแอททริบิวท์ก็คือ คุณสมบัติของเอนทิตีหรือข้อมูลของเอนทิตีที่เราสนใจจะเก็บ หรืออาจจะหมายถึงกลุ่มของค่าความจริงใด ๆ ที่เป็นรายละเอียดของเอนทิตีซึ่งแสดง ลักษณะและคุณสมบัติของเอนทิตี แต่ละเอนทิตีสามารถมีคุณสมบัติต่าง ๆ ได้ มากมาย ขึ้นอยู่กับว่าต้องการจัดเก็บข้อมูลอะไรบ้าง เช่น คุณสมบัติหรือข้อมูลของ นิสิต

ประกอบไปด้วย ชื่อ,ที่อยู่, หมายเลขโทรศัพท์ รหัสสนิสิต, คณะวิชา, สาขาวิชา, วัน เดือนปีเกิด, เพศ, ส่วนสูง, น้ำหนัก ฯลฯ

แอททริบิวท์สามารถแบ่งออกได้หลายประเภทเช่น

1) ซิมเพิลแอททริบิวท์ (Simple Attribute) เป็นแอททริบิวท์แบบ ง่าย ๆ มีเพียงองค์ประกอบเดียวหรือข้อมูลเดียว ภายในแอททริบิวท์ไม่สามารถ แบ่งย่อยได้อีก เช่น ข้อมูล เพศ, อายุ, ส่วนสูง, น้ำหนัก,รหัสสนิสิต, คณะวิชา, สาขาวิชา เป็นต้น จะประกอบด้วยค่าข้อมูลเดียว สัญลักษณ์ที่ใช้แทนแอททริบิวท์ ชนิดนี้ คือวงรีที่มีเส้นเชื่อมต่อไปยังเอนทิตีของแอททริบิวท์ นั้นๆ โดยมีชื่อแอททริบิวท์ อยู่ภายในวงรี

2) คอมโพสิท แอททริบิวท์ (Composite Attribute) หมายถึง แอททริบิวท์ที่มีองค์ประกอบหรือข้อมูล ซึ่งจะตรงกันข้ามกับซิมเพิลแอททริบิวท์ ค่าภายในแอททริบิวท์ สามารถแบ่งย่อยออกมาได้อีก แต่มีอิสระต่อกัน เช่น

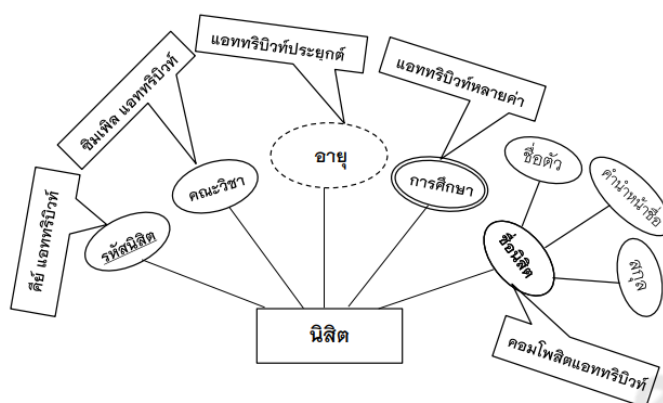
- แอททริบิวท์ “ที่อยู่” จะประกอบด้วย บ้านเลขที่ หมู่ที่ ตำบล อำเภอ และจังหวัด
- แอททริบิวท์ “ชื่อ” จะประกอบด้วย คำนำหน้าชื่อ ชื่อ ตัว และ นามสกุล

3) แอททริบิวท์ ค่าเดียว (Single-value Attribute) คือ แอททริบิวท์ที่มีค่าข้อมูลที่เก็บเพียงค่าเดียว เช่น รหัสสนิสิต ชื่อ ภาควิชา ในเอนทิตี “นิสิต” ซึ่งนิสิต แต่ละคนก็จะมีค่าข้อมูลเหล่านี้เพียงค่าเดียวเท่านั้น หรือเอนทิตี “เพศ” ก็จะมีค่าข้อมูลที่เก็บเพียง 1 ค่า คือ หญิง หรือ ชาย เท่านั้นเช่นกัน

4) แอททริบิวท์ หลายค่า (Multivalued Attribute) คือ แอททริบิวท์ที่สามารถมีได้หลายค่า เช่น คนหนึ่งคนสามารถมี “วุฒิการศึกษา” ได้หลายระดับ เช่น ระดับปริญญาตรี ระดับปริญญาโท และระดับปริญญาเอก หรือ คนหนึ่งคนอาจจะ มี “หมายเลขโทรศัพท์” ได้หลายหมายเลข หรือข้อมูล “โรงเรียนที่เรียนจบ” ตั้งแต่ระดับ อนุบาล ประถมศึกษา มัธยมศึกษา และอุดมศึกษา อาจจะมาจากหลายโรงเรียน เป็น ต้น สัญลักษณ์ของแอททริบิวท์ หลายค่า คือรูปวงรีสองวงซ้อนกัน

5) แอททริบิวท์ ประยุกต์ (Derived Attribute) หมายถึง แอททริบิวท์ที่ไม่ได้เก็บอยู่ในฐานข้อมูล แต่ได้จากการประยุกต์ใช้แอททริบิวท์อื่น เช่น แอททริบิวท์ รายได้รวมของพนักงานขายของ ได้มาจากการรวมเงินเดือน ค่าโบนัส ค่าบำเน็จ (Commission) ของแต่ละเดือน หรือ แอททริบิวท์ อายุ ได้จากการคำนวณจากวันเดือน ปีเกิด สัญลักษณ์ของแอททริบิวท์ประยุกต์ เป็นรูปวงรีแบบเส้นประ

6) คีย์ แอททริบิวท์ (Key Attributes) หรือ คีย์หลัก หรือ กุญแจหลัก เป็นแอททริบิวท์ที่บอกเอกลักษณ์ของเอนทิตีนั้นๆได้ จะมีค่าของข้อมูลที่ไม่ซ้ำกัน เช่น รหัสสนิสิต จัดเป็นคีย์แอททริบิวท์ ใช้ระบุความแตกต่างของแต่ละคนในเอนทิตี สัญลักษณ์ของคีย์หลักคือรูปวงรีภายในมีชื่อแอททริบิวท์ที่มีการขีดเส้นใต้แทนคีย์หลัก



ภาพประกอบที่ 2.6 แสดงแอททริบิวต์ชนิดต่าง ๆ

1.3 ความสัมพันธ์ระหว่างเอนทิตี (Entity Relationship)

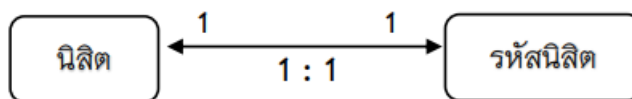
ในปี ค.ศ. 1976 Chen ได้เสนอโมเดลข้อมูลเชิงสัมพันธ์ (Entity Relationship Model) หรือ E-R Model และได้รับการพัฒนาจนสมบูรณ์ในปี ค.ศ. 1979 เพื่อใช้ในการอธิบายฐานข้อมูลในรูปแบบของแผนภาพ เรียกว่า “E-R diagram” ซึ่งจะใช้พื้นฐานของ โมเดลข้อมูลเชิงตรรกะที่ Fleming ที่ได้เสนอไว้ในปี ค.ศ. 1989 เป็นการสร้างโครงร่าง วิวของผู้ใช้ จะแสดงข้อมูลในขอบเขตที่ผู้ออกแบบสนใจ โดยมีสิ่งที่จะต้องกำหนดเป็น พื้นฐานได้แก่ ความสัมพันธ์ระหว่างเอนทิตีและแอททริบิวต์ ในฐานข้อมูลนั้น

ความสัมพันธ์ หรือ Relationship จะหมายถึงความสัมพันธ์ระหว่างเอนทิตี หรือความสัมพันธ์ระหว่างข้อมูล เป็นลักษณะของความสัมพันธ์ระหว่างเอนทิตีหนึ่งกับ ตัวของมันเองหรือกับเอนทิตีอื่น (มากกว่า 2 เอนทิตีก็ได้) ว่ามีความสัมพันธ์ของข้อมูล เป็นแบบใด ความสัมพันธ์ระหว่างเอนทิตีในแผนภาพ E-R จะแทนด้วยรูปสี่เหลี่ยมข้าวหลามตัดที่มีชื่อของความสัมพันธ์อยู่ภายในและมีเส้นเชื่อมระหว่างเอนทิตีและ ความสัมพันธ์ระหว่างเอนทิตี

ความสัมพันธ์ระหว่างข้อมูลที่เราสนใจ ซึ่งพิจารณาได้ 2 แบบ คือ ความสัมพันธ์ แบบจำเป็น หรือแบบบังคับ (Mandatory) หมายถึงความสัมพันธ์ที่ จะต้อง มี หรือ จะต้องเกิดขึ้นอย่างแน่นอน นิยมแทนด้วยสัญลักษณ์เส้นตรงทึบ หรือ เส้น ตรงสองเส้น และความสัมพันธ์แบบไม่จำเป็น (Optional) เป็นความสัมพันธ์ที่ อาจจะมีหรือไม่มีก็ได้ นิยมแทนด้วยสัญลักษณ์เส้นตรงประหรือเส้นตรงหนึ่งเส้น

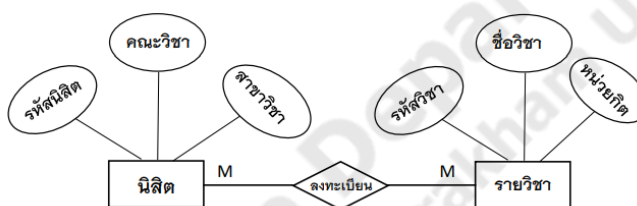
1.3.1 ชนิดความสัมพันธ์ระหว่างข้อมูล

1) ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to One Relationships ; 1 : 1) เป็นการแสดงถึงความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งว่ามีความสัมพันธ์กับ ข้อมูลในอีกเอนทิตีหนึ่งในลักษณะหนึ่งต่อหนึ่ง หมายถึงข้อมูลหนึ่งข้อมูลของเอนทิตีที่ 1 มีความสัมพันธ์กับข้อมูลในเอนทิตีที่ 2 เพียงข้อมูลเดียวเท่านั้น ซึ่งใช้ตัวเลขกำกับที่ เส้นเพื่อระบุความสัมพันธ์ เช่น ความสัมพันธ์ระหว่างข้อมูลนิสิตกับรหัสนิสิต : นิสิต 1 คน จะมีรหัส นิสิตได้เพียง 1 รหัสเท่านั้น ในขณะที่ตัวรหัสนิสิต 1 รหัส จะใช้กับนิสิตได้เพียง 1 คน เช่นกัน



ภาพประกอบที่ 2.7 ความสัมพันธ์ระหว่างข้อมูลนิสิตกับรหัสนิสิต แบบ 1:1

3) ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many to Many Relationships, M : N) เป็นความสัมพันธ์แบบหลายรายการระหว่างสองเอนทิตี เป็น ความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับข้อมูลในอีกเอนทิตีหนึ่ง ใน ลักษณะกลุ่มต่อกลุ่ม เช่น ความสัมพันธ์ระหว่างนิสิตกับรายวิชาที่เปิดสอน : นิสิต 1 คน ลงทะเบียนเรียนได้หลายวิชาและวิชาหนึ่ง ๆ จะมีนิสิตลงทะเบียนได้มากกว่า 1 คน

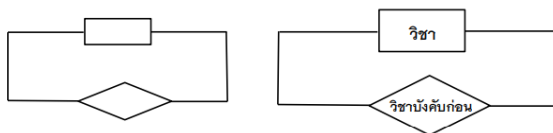


ภาพประกอบที่ 2.8 แสดงความสัมพันธ์แบบกลุ่มต่อกลุ่ม หรือ M : N

1.3.2 ดีกรีของความสัมพันธ์ (Degree of Relationship)

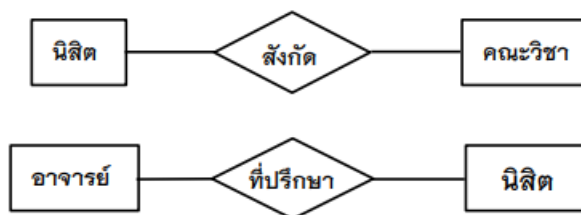
ดีกรีความสัมพันธ์ หรือระดับขั้นของความสัมพันธ์ หมายถึงจำนวนเอนทิตีที่มี ส่วนร่วมกับความสัมพันธ์นั้น ๆ ดีกรีความสัมพันธ์แบ่งออกได้เป็น 3 ประเภท คือ

1) ความสัมพันธ์แบบยูนารี (Unary Relationship) หรือ ความสัมพันธ์แบบรีเคอร์ซีฟ (Recursive Relationship) หรือความสัมพันธ์เอนทิตีเดียว เป็นความสัมพันธ์ระหว่างเอนทิตีชนิดเดียวกัน หรือความสัมพันธ์ในตัวเอง เช่น ความสัมพันธ์ระหว่างเอนทิตี “วิชาที่บังคับเรียนก่อน” และ “วิชา” (วิชาเรียนบางวิชาบังคับให้ต้องเรียนบางวิชาก่อนจึงจะสามารถเรียนวิชาดังกล่าวได้)



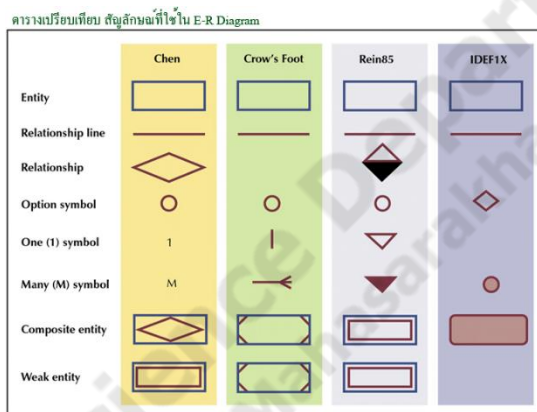
ภาพประกอบที่ 2.9 ความสัมพันธ์แบบยูนารี

2) ความสัมพันธ์แบบไบนารี หรือ ความสัมพันธ์ระหว่างสอง เอนทิตี (Binary Relationship) เป็นความสัมพันธ์เป็นความสัมพันธ์แบบ 2 ทางระหว่าง เอนทิตี 2 เอนทิตี คือ เอนทิตีหนึ่งมีความสัมพันธ์กับเอนทิตีอื่นที่ต่างประเภทกัน เป็น ความสัมพันธ์ที่พบได้บ่อยที่สุด เช่น นิสิต คณะวิชา อาจารย์ นิสิต



ภาพประกอบที่ 2.10 แสดงตัวอย่างความสัมพันธ์แบบไบนารี

3) ความสัมพันธ์ที่มีเอนทิตีมาเกี่ยวข้องตั้งแต่สามเอนทิตีขึ้นไป (N-ary Relationship) เป็นความสัมพันธ์แบบสามทาง หรือเป็นความสัมพันธ์ระหว่าง เอนทิตีต่างประเภทกันตั้งแต่ 3 เอนทิตีขึ้นไป โดยที่ N จะหมายถึงจำนวนเอนทิตีที่มา สัมพันธ์กับความสัมพันธ์หนึ่ง



ภาพประกอบที่ 2.11 สัญลักษณ์ที่ใช้ใน ER Diagram[18]

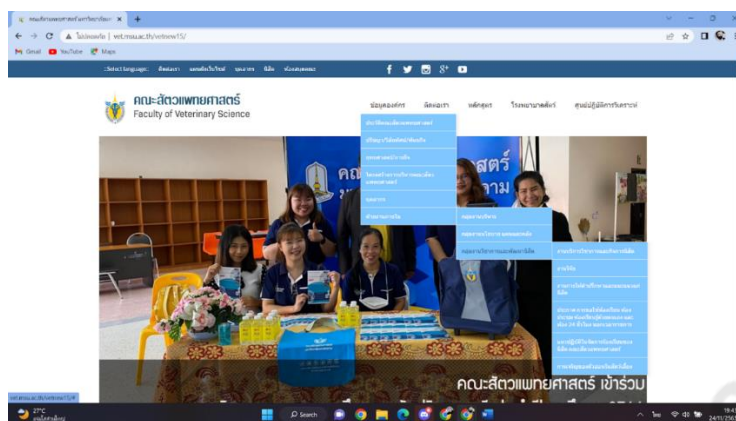
2.2 ระบบงานที่เกี่ยวข้อง

ตัวอย่างเว็บไซต์อื่น

2.2.1 ระบบเว็บไซต์คณะสัตวแพทยศาสตร์ มหาวิทยาลัยมหาสารคาม(เวอร์ชันเก่า)[14] มีรายละเอียดค่อนข้างครบถ้วน

ข้อดี : มีฟังก์ชันการใช้งานค่อนข้างดี

ข้อเสีย : รูปแบบเว็บซับซ้อน ใ้ยาก ตัวหนังสือค่อนข้างเล็ก



ภาพประกอบที่ 2.12 เว็บไซต์คณะสัตวแพทยศาสตร์

2.2.2 ระบบเว็บไซต์คณะศิลปศาสตร์และวิทยาศาสตร์ มหาวิทยาลัยราชภัฏชัยภูมิ[15]

ข้อดี : มีประชาสัมพันธ์ค่อนข้างชัดเจน รูปแบบเรียบง่าย

ข้อเสีย : ตัวหนังสือบางที่ค่อนข้างเล็ก ไม่มีตัวเลือกสำหรับการเปลี่ยนภาษา หัวข้อหมวดหมู่ซ้ำ



ภาพประกอบที่ 2.13 เว็บไซต์คณะศิลปศาสตร์และวิทยาศาสตร์ มหาวิทยาลัยราชภัฏชัยภูมิ

2.2.3 ระบบเว็บไซต์คณะสาธารณสุขศาสตร์ มหาวิทยาลัยราชภัฏนครราชสีมา[16]

ข้อดี : สีสดใส รูปแบเว็บไซต์ไม่ซับซ้อน

ข้อเสีย : ตัวหนังสือบางที่ค่อนข้างใหญ่ ไม่มีตัวเลือกสำหรับการเปลี่ยนภาษา เมนูไม่โดดเด่น



ภาพประกอบที่ 2.14 เว็บไซต์คณะสาธารณสุขศาสตร์ มหาวิทยาลัยราชภัฏนครราชสีมา

2.3 ตารางเปรียบเทียบ

ตารางที่ 2.1 ตารางเปรียบเทียบเว็บไซต์

การบริการ	ระบบเว็บไซต์คณะสัตวแพทยศาสตร์ มหาวิทยาลัยมหาสารคาม(เวอร์ชันเก่า)	ระบบเว็บไซต์คณะศิลปศาสตร์ และวิทยาศาสตร์ มหาวิทยาลัยราชภัฏชัยภูมิ	ระบบเว็บไซต์คณะสาธารณสุขศาสตร์ มหาวิทยาลัยราชภัฏนครราชสีมา	ระบบเว็บไซต์คณะสัตวแพทยศาสตร์ มหาวิทยาลัยมหาสารคาม(เวอร์ชันใหม่)
1.มีการแสดงผลอย่างน้อย 2 ภาษา	✓			
2.มีข้อมูลเกี่ยวกับหน่วยงาน	✓	✓	✓	✓
3.มีข่าวประชาสัมพันธ์	✓	✓	✓	✓
4.มีเอกสารออนไลน์	✓	✓		✓

ตารางที่ 2.1 ตารางเปรียบเทียบเว็บไซต์ (ต่อ)

การบริการ	ระบบเว็บไซต์คณะสัตวแพทยศาสตร์ มหาวิทยาลัยมหาสารคาม(เวอร์ชันเก่า)	ระบบเว็บไซต์คณะศิลปศาสตร์ และวิทยาศาสตร์ มหาวิทยาลัยราชภัฏชัยภูมิ	ระบบเว็บไซต์คณะสาธารณสุขศาสตร์ มหาวิทยาลัยราชภัฏนครราชสีมา	ระบบเว็บไซต์คณะสัตวแพทยศาสตร์ มหาวิทยาลัยมหาสารคาม(เวอร์ชันใหม่)
5.มีเว็บลิงค์หน่วยงานภายใน	✓	✓		✓
6.มีเว็บลิงค์หน่วยงานภายนอก	✓			✓
7.มีเอกสารให้ความรู้เผยแพร่				✓
8.มีผังเว็บไซต์	✓			✓
9.มีช่องทางการติดต่อ	✓	✓	✓	✓
10.มีการค้นหาภายในเว็บไซต์	✓	✓		✓
11.มีรูปแบบที่ทันสมัย	✓			✓