

บทที่ 2

แนวคิด ทฤษฎี เครื่องมือที่ใช้พัฒนา และวรรณกรรมที่เกี่ยวข้อง

การพัฒนาเว็บไซต์แอปพลิเคชันสำหรับการบริหารจัดการงานวิ่ง จำเป็นต้องรวบรวมข้อมูลจากแหล่งข้อมูลต่างๆ ซึ่งข้อมูลเหล่านั้นเป็นสารสนเทศที่สำคัญ ดังนั้น ผู้จัดทำจึงได้ทำการศึกษาค้นคว้าและรวบรวมข้อมูลต่างๆ โดยอาศัยฐานแนวคิด ทฤษฎีและวรรณกรรมที่เกี่ยวข้อง เพื่อใช้ในการพัฒนา

2.1 แนวคิดที่เกี่ยวข้อง

เนื่องด้วยนโยบายของทางมหาวิทยาลัยราชภัฏวชิรเวศน์นครราชสีมา ฯ ที่มีการจัดงานวิ่งประจำปี อยู่เสมอ และมีผู้เข้าร่วมมากมาย ทำให้การบริหารจัดการบุคคลที่เข้าร่วมงานวิ่งประจำปี มีความยุ่งยาก ทั้งการรวบรวมรายชื่อ ทั้งการตรวจสอบการชำระเงิน เพราะไม่มีระบบในการจัดการที่เป็นระบบสารสนเทศแบบเต็มรูปแบบ และการที่จะสมัครงานวิ่งจะต้องติดต่อเจ้าหน้าที่ของทางมหาวิทยาลัยฯ โดยตรง ทำให้เกิดความยุ่งยากต่อการติดต่อ และอาจเกิดข้อผิดพลาดในการสื่อสารได้ ส่วนการออกเกียรติบัตรที่เป็นกระดาษต้องออกให้ทุกคนที่เข้าร่วมงานวิ่ง ทำให้สิ้นเปลืองกระดาษ และเนื่องจากปัญหาที่เกิดขึ้น ทางมหาวิทยาลัยฯ เองก็ได้แก้ปัญหาเหล่านี้โดยการพัฒนาระบบสารสนเทศขึ้นมาจัดการ และเนื่องด้วยคณะบริหารธุรกิจ และศิลปศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี ได้มีนโยบายจัดงานวิ่งประจำปี เช่นเดียวกัน และยังไม่มียุคสารสนเทศเข้ามาจัดการ ดังนั้นคณะผู้จัดทำโครงการจึงมีความต้องการในการพัฒนาระบบเพื่อตอบสนองความต้องการของผู้ใช้งานในหลากหลายแพลตฟอร์ม โดยระบบดังกล่าว สามารถบริหารจัดการเกี่ยวกับการสร้างกิจกรรมงานวิ่งได้ผ่านทางเว็บไซต์ เพื่อความสะดวกสบายต่อผู้จัดงาน อีกทั้งผู้ใช้งานก็สามารถที่จะสมัครสมาชิก และสมัครเข้าร่วมกิจกรรมงานวิ่งผ่านทางเว็บไซต์ได้ ทำให้เกิดความสะดวกรวดเร็วต่อผู้ใช้ และการที่มีระบบเข้ามาจัดการเก็บข้อมูลทำให้ ช่วยลดความซ้ำซ้อนของข้อมูลผู้ใช้ ที่ต้องกรอกซ้ำๆ ทุกครั้งเวลาจัดกิจกรรมการวิ่ง ฉะนั้นคณะผู้จัดทำโครงการจึงนำเทคโนโลยีสารสนเทศเข้ามาช่วยแก้ไขและพัฒนาเว็บไซต์สำหรับจัดการงานวิ่งภายในคณะ เพื่อให้ขั้นตอนการทำงานง่ายขึ้น และเข้าถึงได้ง่าย อีกทั้งยังช่วยประหยัดค่าใช้จ่าย จากการเช่าระบบสารสนเทศภายนอก

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 ทฤษฎีวงจรการวิเคราะห์ระบบงาน

ในการพัฒนาระบบนั้น ได้มีการกำหนดให้เป็นไปในทิศทางเดียวกัน และกำหนดขั้นตอนที่เป็นแนวทางในนักวิเคราะห์ระบบปฏิบัติงานได้โดยมีข้อบกพร่องน้อยที่สุด เพราะงานการวิเคราะห์ระบบในปัจจุบันมีความซับซ้อนของงานมากกว่าสมัยก่อน นักวิเคราะห์ระบบจึงต้องการมาตรฐานในการพัฒนาระบบดังกล่าว จึงได้มีการคิดค้นวงจรการพัฒนาระบบงานขึ้น เพื่อตอบสนองความต้องการของนักวิเคราะห์ระบบ (System Analysis : SA)

2.2.1.1 วงจรการพัฒนาระบบงานสำหรับระบบงานทั่วไป

วงจรการพัฒนาระบบงาน (System Development Life Cycle : SDLC) สำหรับระบบทั่วไปที่ได้มีการคิดค้นขึ้นมา สามารถแบ่งออกเป็นลำดับขั้นได้ 4 ขั้นตอน คือ

1. การวิเคราะห์ระบบงาน เป็นขั้นตอนของการศึกษาระบบงานเดิมที่ใช้ในปัจจุบัน (Current System) ปัญหาที่เกิดจากระบบงานเดิม ตลอดจนการศึกษาดังกล่าวถึงความต้องการของธุรกิจ (Business Needs and Requirements) พร้อมกับการประเมินเหตุการณ์ต่าง ๆ เพื่อหาทางเลือกที่เหมาะสมมาแก้ปัญหา
2. การออกแบบและวางระบบงาน เป็นขั้นตอนหลังจากการวิเคราะห์ระบบงานซึ่งเป็นขั้นตอนที่นักวิเคราะห์ระบบจะต้องวางโครงสร้างของระบบงาน ในรูปลักษณะทั่ว ๆ ไปและในรูปลักษณะเฉพาะโดยมีการแจกแจงรายละเอียดที่แน่ชัดของแต่ละงาน หรือระบบงานย่อยของระบบที่ได้ออกแบบขึ้นจะถูกส่งต่อไปให้กับโปรแกรมเมอร์เพื่อจะได้ทำการเขียนโปรแกรมให้เป็นระบบที่ปฏิบัติงานได้จริงในขั้นตอนต่อไป
3. การนำระบบเข้าสู่ธุรกิจหรือผู้ใช้เป็นขั้นตอนที่นำเอาระบบงานมาติดตั้ง (Install) ให้กับผู้ใช้ และเพื่อให้แน่ใจว่าระบบงานสามารถปฏิบัติงานได้ตรงตามวัตถุประสงค์ระบบงานจะต้องถูกทำการตรวจสอบมาอย่างดีพร้อมกับการฝึกอบรม (Education and Training) ให้ผู้ใช้ระบบสามารถใช้งานได้อย่างมีประสิทธิภาพและถูกต้อง
4. การดำเนินการสนับสนุนภายหลังการติดตั้งระบบงาน เป็นขั้นตอนที่ระบบงานใหม่ได้ถูกนำมาติดตั้งแล้วผู้ใช้ระบบอาจจะยังไม่คุ้นเคยกับการทำงานในระบบใหม่นักวิเคราะห์ระบบควรจะให้คำแนะนำอย่างต่อเนื่องเพื่อช่วยเหลือผู้ใช้ระบบในการปฏิบัติงานทั้งนี้รวมถึงความต้องการต่าง ๆ ที่อาจจะเกิดขึ้นหรือเปลี่ยนแปลงไปภายหลังจากระบบได้ถูกติดตั้ง ซึ่งมักจะ

เกี่ยวข้องกับกำรบำรุงรักษาระบบงาน (System Maintenance) และกำรปรับปรุงระบบงาน (System Improvement)

เมื่อควมต้องการเปลี่ยนแปลงไป และระบบงานที่กำล้งปฏิบัติอยู่เป็นประจำ จำเป็นต้องปรับปรุงใหม่นักวิเคราะห์ระบบจะต้องกลับไปเริ่มต้นที่ขั้นที่ 1 ใหม่ และจะเป็นเช่นนี้เรื่อย ๆ ไป ถ้าเกิดควมเปลี่ยนแปลงของระบบ

2.2.1.2 วงจรการพัฒนาระบบงานสำหรับระบบสารสนเทศ

วงจรการพัฒนาระบบงาน (System Development Life Cycle : SDLC) ของระบบสารสนเทศ ได้มีการคิดค้นขึ้นมาโดยมีขั้นตอนที่แตกต่างไปจากวงจรการพัฒนาระบบงานสำหรับระบบงานทั่วไป ตรงที่มีขั้นตอนในการพัฒนาระบบงานที่ละเอียดกว่าถึง 7 ขั้นตอน ซึ่งนักวิเคราะห์ระบบต้องทำความเข้าใจว่าในแต่ละขั้นตอนว่าทำอะไรและทำสมารถแบ่งออกเป็นลำดับขั้นตอนดังนี้ คือ

1. ค้นหาปัญหา โอกาสและเป้าหมาย (Identifying Problems, Opportunity and Objective)

ระบบสารสนเทศจะเกิดขึ้นได้ก็ต่อเมื่อผู้บริหารหรือผู้ใช้ตระหนักว่าต้องการระบบสารสนเทศ หรือต้องแก้ไขระบบเดิม โดยมีขั้นตอนดังนี้

1.1 นักวิเคราะห์และออกแบบระบบ ต้องศึกษาระบบโดยละเอียด เพื่อให้เข้าใจถึงปัญหาที่เกิดขึ้นในองค์กรตัวอย่างปัญหา เช่น

- บริษัท ก เปิดสาขาเพิ่มมากขึ้น ระบบเดิมไม่ได้ครอบคลุมถึงการขยายตัวของบริษัท
- บริษัท ข เก็บข้อมูลผู้ขายได้เพียง 1,000 ราย แต่ปัจจุบันระบบนี้มีข้อมูลผู้ขาย 900 ราย และในอนาคตจะมีเกิน 1,000 ราย

- ระบบสารสนเทศในองค์กรหลาย ๆ แห่งในปัจจุบัน ที่ใช้มานานแล้วและใช้เพื่อติดตามเรื่องการเงินเท่านั้น ไม่ได้มีจุดประสงค์เพื่อให้เป็นสารสนเทศเพื่อตัดสินใจ

1.2 พยายามหาโอกาสในการปรับปรุงวิธีการทำงานโดยการใช้ระบบคอมพิวเตอร์

1.3 นักวิเคราะห์และออกแบบระบบ ต้องมองเป้าหมายให้ชัดเจน เพื่อจะได้รู้ทิศทางของการทำระบบให้เป็นไปตามเป้าหมายที่วางไว้ เช่น ต้องการแข่งขันกับคู่แข่งในเรื่องการลดต้นทุนในการผลิตสินค้า โดยการลดจำนวนการสต็อกวัตถุดิบ ดังนั้น นักวิเคราะห์และออกแบบระบบ จะเห็นถึงปัญหา โอกาส และเป้าหมายในการนำระบบคอมพิวเตอร์เข้าไปใช้ในการเก็บข้อมูลสต็อกวัตถุดิบ และประมวลผลการสั่งวัตถุดิบ เป็นต้น

2. ศึกษาความเป็นไปได้ (Feasibility Study)

2.1 กำหนดว่าปัญหาคืออะไร และตัดสินใจว่าจะพัฒนาระบบสารสนเทศใหม่หรือการแก้ไขระบบสารสนเทศเดิมมีความเป็นไปได้หรือไม่ โดยเสียค่าใช้จ่ายและเวลาน้อยที่สุด

2.2 นักวิเคราะห์และออกแบบระบบ ต้องกำหนดให้ได้ว่าการแก้ปัญหานั้น

2.2.1 มีความเป็นไปได้ทางเทคนิคหรือไม่ เช่น จำนวนเครื่องคอมพิวเตอร์ที่มีอยู่เพียงพอหรือไม่ ซอฟต์แวร์แก้ไขได้หรือไม่

2.2.2 มีความเป็นไปได้ทางบุคลากรหรือไม่ เช่น มีบุคคลที่เหมาะสมที่จะพัฒนาและติดตั้งระบบหรือไม่ ผู้ใช้มีความคิดเห็นอย่างไรกับการเปลี่ยนแปลง

2.2.3 มีความเป็นไปได้ทางเศรษฐศาสตร์หรือไม่ เช่น มีเงินลงทุนหรือไม่ค่าใช้จ่ายในการวิเคราะห์และออกแบบค่าใช้จ่ายในด้านเวลาที่ต้องใช้ในการพัฒนาระบบ

ดังนั้นในการศึกษาความเป็นไปได้นั้นสามารถสรุปได้ดังต่อไปนี้ คือ

- หน้าที่ : กำหนดปัญหาและศึกษาความเป็นไปได้หรือไม่ที่จะเปลี่ยนแปลงระบบ
- ผลลัพธ์ : รายงานความเป็นไปได้
- เครื่องมือ : เก็บรวบรวมข้อมูลของระบบและคาดคะเนความต้องการของระบบ
- บุคลากรและหน้าที่รับผิดชอบ
- นักวิเคราะห์และออกแบบระบบต้องเก็บรวบรวมข้อมูลทั้งหมดที่จำเป็น
- นักวิเคราะห์และออกแบบระบบ ต้องคาดคะเนความต้องการของระบบและแนว

ทางแก้ไขปัญหา

- นักวิเคราะห์และออกแบบระบบ ก หนดความต้องการที่แน่ชัด เพื่อใช้ในการวิเคราะห์ระบบ โดยที่ผู้บริหารจะตัดสินใจว่าจะดำเนินโครงการต่อไปหรือไม่หรือยกเลิกโครงการ

3. วิเคราะห์ความต้องการของระบบ (Analyzing System Needs)

3.1 เริ่มตั้งแต่ศึกษาการทำงานของธุรกิจเดิม ว่าทำงานอย่างไร

3.2 กำหนดความต้องการของระบบใหม่

3.3 เครื่องมือ : Data Dictionary, DFD, Process Specification, Data Model, Prototype

3.4 บุคลากรและหน้าที่ : ผู้ใช้ต้องให้ความร่วมมือ

3.5 นักวิเคราะห์และออกแบบระบบศึกษาเอกสารที่มีอยู่ และศึกษาระบบเดิมเพื่อให้เข้าใจขั้นตอนการทำงานของระบบ

3.6 นักวิเคราะห์และออกแบบระบบเตรียมรายงานความต้องการของระบบใหม่

3.7 นักวิเคราะห์และออกแบบระบบเขียนแผนภาพการทำงาน (DFD) ของระบบเดิมและระบบใหม่

3.8 นักวิเคราะห์และออกแบบระบบสร้าง Prototype ขึ้นมาก่อน

4. การออกแบบระบบ (Designing the Recommended System)

4.1 ออกแบบระบบใหม่เพื่อให้สอดคล้องกับความต้องการของผู้ใช้และผู้บริหาร

4.2 บุคลากรหน้าที่ :

4.2.1 นักวิเคราะห์และออกแบบระบบตัดสินใจเลือกฮาร์ดแวร์และซอฟต์แวร์

4.2.2 นักวิเคราะห์และออกแบบระบบออกแบบข้อมูล เข้ารายงานการแสดงผลบนหน้าจอ ออกแบบฐานข้อมูล

4.2.3 นักวิเคราะห์และออกแบบระบบกำหนดจำนวนบุคลากรในระบบ

5. พัฒนาซอฟต์แวร์และจัดทำเอกสาร (Developing and Documenting Software)

5.1 เขียนโปรแกรม จัดทำคู่มือการใช้โปรแกรม และฝึกอบรมผู้ใช้ที่เกี่ยวข้องในระบบ

5.2 บุคลากรและหน้าที่

5.2.1 นักวิเคราะห์และออกแบบระบบเตรียมสถานที่และการติดตั้งเครื่องคอมพิวเตอร์

5.2.2 นักวิเคราะห์และออกแบบระบบวางแผนและดูแลการเขียนโปรแกรม

5.2.3 โปรแกรมเมอร์เขียนโปรแกรม

5.2.4 นักวิเคราะห์และออกแบบระบบดูแลการเขียนคู่มือการใช้โปรแกรมและการฝึกอบรม

6. ทดสอบและบำรุงรักษาระบบ (Testing and Maintaining the System)

6.1 นักวิเคราะห์และออกแบบระบบและทีมงานทดสอบโปรแกรม

6.2 ผู้ใช้ตรวจสอบว่าโปรแกรมทำงานตามที่ต้องการ

6.3 ถ้าเกิดข้อผิดพลาดของโปรแกรม ให้ปรับปรุงแก้ไข

6.4 เมื่อทดสอบโปรแกรมแล้ว โปรแกรมไม่เป็นไปตามความต้องการ อาจต้องแก้ไขปรับปรุงใหม่

6.5 การบำรุงรักษา ส่วนใหญ่เป็นการแก้ไขโปรแกรมหลังจากใช้งานแล้ว เนื่องจาก

- มีปัญหาในโปรแกรม (Bug)

- ธุรกิจเกิดการเปลี่ยนแปลงไป เช่น ธุรกิจขยายตัว ธุรกิจสร้างสินค้าตัวใหม่ ความ

ต้องการของระบบก็เพิ่มขึ้น รายงานเพิ่มขึ้น

- การเปลี่ยนแปลงทางฮาร์ดแวร์และซอฟต์แวร์
- ความต้องการผู้ใช้งานเพิ่มขึ้น 40-60 % ของค่าใช้จ่ายในการพัฒนาระบบจะใช้ในการบำรุงรักษาระบบ

7. ดำเนินงานและประเมินผล (Implementing and evaluating the System)

7.1 ติดตั้งระบบให้พร้อม

7.2 นำระบบใหม่มาใช้แทนระบบเดิม

7.3 ใช้ระบบใหม่ควบคู่กับระบบเดิมสักกระยะหนึ่ง แล้วดูผลลัพธ์ว่าตรงกันหรือไม่ถ้าใช้งานดี ก็เลิกใช้ระบบเดิมและใช้ระบบใหม่

7.4 นักวิเคราะห์และออกแบบระบบทำการประเมินผล เพื่อให้ทราบถึงความพอใจของผู้ใช้ระบบ หรือสิ่งที่ต้องแก้ไขปรับปรุง หรือปัญหาที่พบ

2.2.2 ทฤษฎีการพัฒนาเว็บไซต์ที่รองรับกับการใช้งานบนทุกหน้าจออุปกรณ์ (Responsive Web Design: RWD)

เรซปอนด์ซีฟ เว็บไซต์ (Responsive Web Design) คือ การออกแบบเว็บเพจ หรือเว็บไซต์ด้วยแนวคิดใหม่ ที่จะทำให้การแสดงผลของเว็บเพจบนอุปกรณ์ที่แตกต่างกันได้อย่างเหมาะสม โดยใช้ที่อยู่เว็บ เดียวกัน (One Web Address) และโค้ดชุดเดียวกัน (One Code) ไม่ว่าจะแสดงผลบนอุปกรณ์โมบาย แท็บเล็ต หรือคอมพิวเตอร์ส่วนบุคคล

2.2.2.1 ขนาดของหน้าจอที่มีความหลากหลาย

ในปัจจุบันหน้าจอโดยทั่วไปของสมาร์ทโฟนและแท็บเล็ตจะมีขนาดอยู่ระหว่าง 3.5-10 นิ้วและความละเอียดหน้าจอ 320*480 , 480*800, 960*640 และอื่นๆขณะที่ขนาดหน้าจอโดยทั่วไปของคอมพิวเตอร์พีซีมีขนาด 11 นิ้ว และมีความละเอียดหน้าจออยู่ที่ 1024*768 หรือมากกว่า

2.2.2.2 การปรับรูปแบบของหน้าเว็บไซต์

เว็บไซต์ที่ดีควรมีการปรับเปลี่ยนการแสดงผลให้สามารถรองรับกับขนาดของหน้าจอที่แตกต่างกันของแต่ละอุปกรณ์โดยไม่ทำการออกแบบเฉพาะเจาะจงให้กับอุปกรณ์ใด หรือขนาดใด แต่เป็นการออกแบบเพื่อให้มีการเรียนรู้ที่จะปรับเปลี่ยนการแสดงผลได้ด้วยตนเอง ใหญ่เข้ากับ การวางแนวและขนาดของหน้าจอ

2.2.2.3 การใช้งานหน้าเว็บ

สำหรับคอมพิวเตอร์ที่ใช้งานหน้าเว็บจะทำผ่านอุปกรณ์เมาส์และคีย์บอร์ด ที่มีเพียงจุดสัมผัสเล็กๆ บนหน้าจอแต่ในสมาร์ตโฟนและแท็บเล็ตใช้งานหน้าเว็บผ่านการสัมผัสจากนิ้วมือที่มีขนาดใหญ่กว่า

2.3 เครื่องมือที่ใช้ในการพัฒนาระบบที่เกี่ยวข้อง

2.3.1 ภาษา Ruby

Ruby คือภาษาเขียนโปรแกรมแบบไดนามิกส์และสามารถปรับแต่งโครงสร้างของมันเองได้ในขณะที่โปรแกรมทำงาน นอกจากนี้ มันยังเป็นภาษาสำหรับเขียนโปรแกรมเชิงวัตถุมาตั้งแต่กำเนิด (Object-oriented) และเป็นภาษาสำหรับการเขียนโปรแกรมเพื่อวัตถุประสงค์ทั่วไป มันถูกออกแบบและพัฒนาขึ้นในช่วงกลางของ 1990s โดย Yukihiro Matsumoto ในประเทศญี่ปุ่น

ตามที่ Matsumoto ได้บอกไว้ว่าภาษา Ruby นั้นได้รับอิทธิพลมาจากภาษา Perl, Smalltalk, Eiffel, Ada และ Lisp มันเป็นภาษาที่สนับสนุนการเขียนโปรแกรมหลายรูปแบบ ซึ่งประกอบไปด้วยการเขียนโปรแกรมแบบฟังก์ชัน (Functional) การเขียนโปรแกรมเชิงวัตถุและเป็น Imperative นอกจากนี้ ภาษา Ruby ยังมีประเภทข้อมูลเป็นแบบไดนามิกส์และมีระบบการจัดการหน่วยความจำแบบอัตโนมัติ

2.3.2 Ruby On Rails

Ruby On Rails หรือ ROR (ต่อไปจะเรียกสั้น ๆ ว่า Rails แล้วกัน) นั้นเป็น open-source web application framework ที่ถูกพัฒนาด้วยภาษา Ruby โดยที่ Rails นั้นจะมีโครงสร้างการจัดการแบบ Model View Controller ซึ่งมีข้อดีคือ

- มีการออกแบบที่ชัดเจน มองภาพรวมได้ง่าย
- ปรับปรุง/แก้ไข ระบบได้โดยง่าย เพราะแต่ละส่วนแยกกันอย่างชัดเจน
- maintain ระบบได้ง่าย เพราะมีโครงสร้างชัดเจน จึงทำให้ยืดหยุ่นต่อการเพิ่มเติม รวมไปถึงการนำส่วนต่าง ๆ ไปใช้ซ้ำได้ด้วย

อีกทั้ง Rails ยังถูกพัฒนาขึ้นมาเพื่อให้แอปพลิเคชันถูกพัฒนาด้วยโค้ดที่น้อยกว่า framework อื่นอย่างมาก และมี configuration ที่น้อยที่สุด เพื่อให้ง่ายต่อการนำไปประยุกต์ใช้

2.3.3 Tailwind CSS

Tailwind CSS คือ CSS Utility Framework ที่ช่วยให้นักพัฒนาสร้าง UI ที่สำคัญได้ด้วยตัวเองอย่างรวดเร็ว และยังสามารถปรับแต่งในรายละเอียดปลีกย่อยได้ง่าย เนื่องจากมาพร้อม กับ Class สำเร็จรูปสุดอเนกประสงค์ที่ใช้งานได้ทันทีในกรณีที่ต้องการเปลี่ยน UI หลักของเฟรมเวิร์ก เช่น สี ขนาด การจัดวาง หรือปุ่มต่างๆ นั้นทำให้นักพัฒนาแทบไม่ต้องเข้าไปแก้ไขที่ไฟล์ CSS หลักเลย

จะว่าไป Tailwind CSS ก็เหมือนเวลาที่เร เล่นบล็อ กของเล่นหรือตัวต่อตอนเด็ก ๆ เพราะชิ้นส่วนต่างๆ มีขนาด รูปร่าง และสี สันที่แตกต่างกัน ซึ่งคุณสามารถประกอบร่างอะไรก็ได้ตามใจชอบตามความต้องการหรือจินตนาการของคุณ

2.3.4 Hotwire

Ruby on Rails ออกเวอร์ชันใหญ่ 7.0 ซึ่งมีการเปลี่ยนแปลงสำคัญ 2 เรื่องคือ ไม่ต้องผูกกับโลกของ Node.js อีกแล้ว และเปลี่ยนระบบ front-end มาเป็นตัวใหม่คือ Hotwire

David Heinemeier Hansson หรือ DHH ผู้สร้าง Rails อธิบายว่าโครงสร้างของภาษา JavaScript ในอดีตเป็นปัญหาของโปรแกรมเมอร์มาก ทางออกจึงเป็นการเขียนภาษาอื่นแล้วแปลงเป็น JavaScript (เรียกว่า transpiling เช่น CoffeeScript หรือ Babel) ที่ช่วยให้เขียนได้ดั่งง่ายขึ้น แต่ก็ต้องแลกมาด้วยความซับซ้อนที่เพิ่มขึ้น

ส่วนฝั่ง front-end เปลี่ยนมาใช้ Hotwire ซึ่งพัฒนาโดยบริษัท Basecamp เช่นกัน แนวคิดของ Hotwire คือส่งข้อมูลไปยังเบราว์เซอร์เป็น HTML แทน JSON (HTML-over-the-wire) เพื่อให้การเรนเดอร์เทมเพลตอยู่ที่ฝั่งเซิร์ฟเวอร์ เพจโหลดเร็วขึ้น และลดความซับซ้อนลง ถูกนำมาใช้แทนระบบ front-end เดิมของ Rails ที่เป็น Turbolinks และ Rails UJS

2.3.5 PostgreSQL

PostgreSQL (โพสท์เกรสคิวแอล) คือ PostgreSQL เรียกได้ว่าเป็นระบบการจัดการฐานข้อมูลเชิงวัตถุ-สัมพันธ์ (object-relational) แบบ ORDBMS โดยสามารถใช้รูปแบบคำสั่งของภาษา SQL ได้เกือบทั้งหมด นอกจากนี้ยังเป็นระบบฐานข้อมูลที่ทันสมัยที่สุดของ OpenSource ที่สามารถนำไปใช้งานได้โดยไม่มีค่าใช้จ่ายใด ได้มีการพัฒนามาจาก POSTGRES 4.2 โดยมหาวิทยาลัยแคลิฟอร์เนีย (Berkeley Computer Science department, University of California.)

PostgreSQL สามารถทำงานบนระบบปฏิบัติการได้ทั้ง Linux, UNIX (AIX, BSD, HP-UX, SGI Irix, Mac OS X, Solaris, Tru64) และ Windows

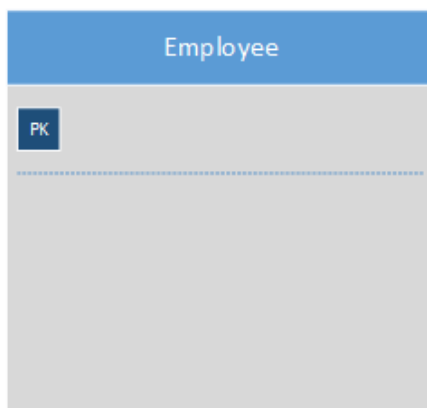
2.3.6 Heroku

Heroku คือ Cloud Platform ที่ให้เราสร้าง App ได้อย่างรวดเร็ว โดยไม่ต้องกังวลเรื่องการจัดการ Infrastructure ต่าง ๆ ทำให้เราสามารถโฟกัสเฉพาะการ Build, Deploy, Monitor App รวมถึง Scale App ในกรณีที่ผู้ใช้เพิ่มมากขึ้น

Heroku คือ Platform as a Service (Paas) ที่ Based-on ระบบ Managed Container System คล้ายกับ Docker เป็นแกนกลาง เรียกว่า Dyno รายล้อมด้วย Data Service ที่หลากหลาย และบริการอื่น ๆ อีกมากมาย ที่ใช้ในการให้บริการ App สมัยใหม่

2.3.7 Entity Relationship Diagram

1. **Entity** ใช้ในการบอกประเภทของข้อมูลที่เกิดขึ้นในลักษณะของตาราง โดยใช้คำนามในการบอกรายละเอียดของข้อมูล เช่น Entity Employee ก็จะบอกว่าเป็น Entity ที่ใช้ในการเก็บข้อมูลที่เกี่ยวข้องกับพนักงาน



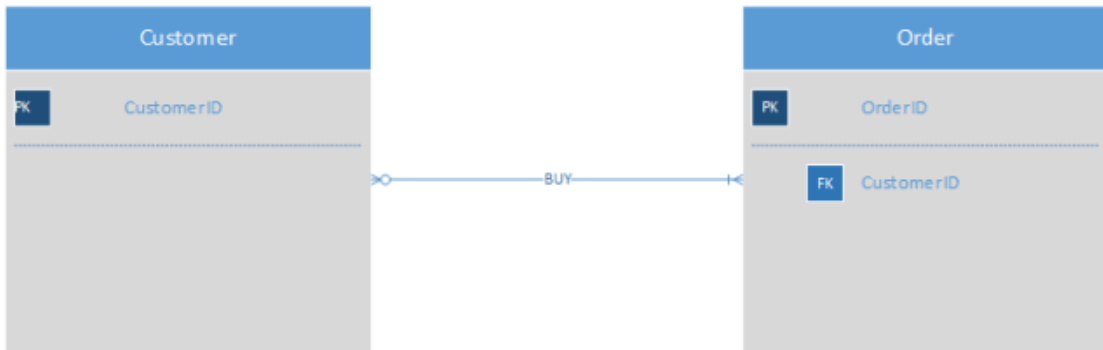
รูปภาพที่ 2.1 Entity

2. **Attribute** ใช้ในการอธิบายคุณลักษณะของแต่ละ Entity โดยใช้คำนามในการบอกถึงคุณลักษณะต่าง ๆ ของ Entity เช่น Entity Employee จะประกอบไปด้วย Attribute: EmployeeID, LastName, FirstName, Title ฯลฯ



รูปภาพที่ 2.2 Attribute

3. Relationship ใช้ในการอธิบายความสัมพันธ์ระหว่าง 2 Entity โดยใช้คำกริยาในการสื่อถึงความสัมพันธ์ระหว่าง 2 Entity เช่น Relationship Buy บอกความสัมพันธ์ระหว่าง Entity Customer และ Entity Order



รูปภาพที่ 2.3 Relationship

4. Cardinality จะประกอบไปด้วยค่า 2 ตัว คือตัวแรก Maximum หรือ Multiplicity (One or Many) และตัวที่สอง Minimum (Zero or One) หรือจะเรียกว่า (Optional or Mandatory)

- Zero or Many



รูปภาพที่ 2.4 Zero or Many

- One or Many



รูปภาพที่ 2.5 One or Many

- One or One



รูปภาพที่ 2.6 One or One

- Zero or One



รูปภาพที่ 2.7 Zero or One

2.3.8 พจนานุกรมข้อมูล (Data Dictionary)

แสดงรายละเอียดของข้อมูลที่จะจัดเก็บไว้ในแฟ้มข้อมูลซึ่งจะต้องสอดคล้องกับพจนานุกรมข้อมูลของกระแสข้อมูลที่ได้กำหนดไว้แล้วก่อนหน้า และสอดคล้องกับแหล่งจัดเก็บข้อมูลที่ปรากฏอยู่ในแผนภาพกระแสข้อมูล









ตารางที่ 2.1 พจนานุกรมข้อมูล

ชื่อตาราง	user		ประเภท ตาราง	master	
คำอธิบาย	เก็บข้อมูลชื่อผู้ใช้ รหัสผ่าน และระดับของผู้ใช้				
คีย์หลัก	user_id		คีย์นอก	inuser, exuser	
เขตข้อมูล	ชนิด	ขนาด	ตัวอย่าง	ค่าที่เป็นไปได้	สัมพันธ์กับ ตาราง
user_id	char	4	0001	0000-9999	
inuser_id	char	3	001	000-999	inuser
exuser_id	char	5	00000	00000-99999	exuser
username	varchar	16	admin		
password	varchar	16	1234		
level	char	1		1 = customer 2 = product 3 = sale 4 = admin	

2.3.9 Data Flow Diagram: DFD

2.3.9.1 สัญลักษณ์ DFD

ตารางที่ 2.2 สัญลักษณ์ DFD

DeMarco & Yourdon	Gane & Sarson	ความหมาย
		Process : ขั้นตอนการทำงานภายในระบบ
		Data Store : แหล่งข้อมูลสามารถเป็นได้ทั้งไฟล์ข้อมูลและฐานข้อมูล (File or Database)
		External Agent : บัญชีหรือสภาพแวดล้อมที่มีผลกระทบต่อระบบ
		Data Flow : เส้นทางไหลของข้อมูล แสดงทิศทางของข้อมูลจากขั้นตอนการทำงานหนึ่งไปยังอีกขั้นตอนหนึ่ง

2.3.9.2 คำอธิบายกระบวนการ (process description)

เป็นการอธิบายถึงรายละเอียดของวิธีการทำงานในกระบวนการโดยอ้างอิงจากแผนภาพกระแสข้อมูล ทำให้เกิดความชัดเจนในรายละเอียดของแต่ละกระบวนการ เพื่อการสื่อสารที่ตรงกันระหว่างนักวิเคราะห์ระบบ ผู้ใช้ ผู้เกี่ยวข้องกัระบบ และโปรแกรมเมอร์

ตารางที่ 2.3 process description

Process Description	
System	ระบบจัดการสินค้า ร้านไอเอสซีพหลาย
DFD Number	2
Process Name	จัดการข้อมูลผู้ใช้ระบบ
Input Data Flow	ข้อมูลการจัดการซื้อผู้ใช้และรหัสผ่าน
Output Data Flow	ข้อมูลผู้ใช้ระบบ, ข้อมูลลูกค้า
Data Store Used	ข้อมูลผู้ใช้ระบบ, ข้อมูลลูกค้า, ข้อมูลพนักงาน
Description	<p>เป็นกระบวนการสำหรับเจ้าของร้านซึ่งได้สิทธิเป็นผู้ดูแลระบบ สามารถใช้ซื้อผู้ใช้และรหัสผ่านของตน เข้ามาจัดการข้อมูลผู้ใช้ในระบบงาน ผู้ใช้จะแบ่งเป็น 2 ประเภท ได้แก่</p> <ol style="list-style-type: none"> 1) ผู้ใช้ภายใน ได้แก่ เจ้าของร้าน พนักงานขาย พนักงานคลังสินค้า และ พนักงานรับส่งสินค้า โดยผู้ดูแลระบบจะสามารถกำหนดชื่อผู้ใช้ และรหัสผ่านให้แก่ผู้ใช้ภายในที่มีข้อมูลอยู่ในฐานข้อมูลได้ 2) ผู้ใช้ภายนอก ได้แก่ ลูกค้า ที่สมัครเป็นสมาชิกผ่านหน้าเว็บไซต์ ผู้ดูแลระบบสามารถแก้ไขชื่อผู้ใช้ และแก้ไขรหัสผ่านได้ ในกรณีที่ลูกค้าอาจลืมชื่อผู้ใช้และรหัสผ่าน และสามารถเพิ่ม ลบ แก้ไข ข้อมูลลูกค้าได้ในกรณีที่มีความจำเป็น

2.4 วรรณกรรมที่เกี่ยวข้อง

2.4.1 ระบบบริหารจัดการโครงการจิตอาสา กรณีศึกษา : สโมสรนักศึกษาคณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏบุรีรัมย์

วรินทร์พิพัชร วัชรพงษ์เกษม และ สมฤทธิ์ แก้วอินทนน (2560)

การวิจัยครั้งนี้มีวัตถุประสงค์ 1) เพื่อพัฒนาระบบบริหารจัดการโครงการจิตอาสา กรณีศึกษา : สโมสรนักศึกษาคณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏบุรีรัมย์ 2) เพื่อประเมินความพึงพอใจของนักศึกษาที่มีต่อระบบบริหารจัดการโครงการจิตอาสา สโมสรนักศึกษาคณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏบุรีรัมย์ ซึ่งมีวิธีดำเนินการวิจัย 7 ขั้นตอน ดังนี้ 1) ศึกษาความเป็นไปได้ และกำหนดปัญหาของระบบ 2) ออกแบบระบบ โดยทำการออกแบบแผนภาพการทำงานของระบบงานใหม่ แผนภาพกระแสข้อมูล (Data Flow Diagram) การออกแบบระบบฐานข้อมูลความสัมพันธ์ของข้อมูลในรูปแบบ IDEF1X และการออกแบบหน้าจอแสดงผลข้อมูล 3) พัฒนาระบบโดยใช้ภาษา HTML5 & CSS3 JavaScript และPHP โดยใช้ฐานข้อมูล MySQL 4) นำระบบไปทดสอบการใช้งานจริง ระยะเวลา 1 ภาคการศึกษา 5) ประเมินความพึงพอใจของนักศึกษาที่มีต่อระบบระบบบริหารจัดการโครงการจิตอาสา จำนวน 308 คน โดยใช้แบบประเมินออนไลน์ 6) เก็บรวบรวมข้อมูล วิเคราะห์โดยการหาค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน และสรุปผลความพึงพอใจของนักศึกษาที่มีต่อระบบระบบบริหารจัดการโครงการจิตอาสา 7) จัดทำคู่มือและจัดการฝึกอบรมการใช้งานผลการวิจัยพบว่า 1) ระบบบริหารจัดการโครงการจิตอาสา พบว่า ระบบจะแบ่งผู้ใช้งานเป็น 2 ส่วน ซึ่งจะมีหน้าจอกการใช้งานที่แตกต่างกัน โดยระบบหน้าเว็บสำหรับผู้ใช้งานทั่วไป จะมีส่วน Login เพื่อตรวจสอบข้อมูลต่าง ๆ รวมถึงข้อมูลข่าวสารและประชาสัมพันธ์ได้ ซึ่งระบบส่วนที่ 2 จะเป็นส่วนหน้าของผู้ดูแลระบบ จะมีส่วนของการบริหารจัดการจะมีทั้งหมด 9 ระบบย่อย ดังนี้ระบบจัดการข้อมูลภาคเรียนและปีการศึกษา ระบบจัดการข้อมูลโครงการจิตอาสา ระบบจัดการข้อมูลกิจกรรมโครงการจิตอาสา ระบบจัดการข้อมูลสถานที่ในการดำเนินกิจกรรม ระบบสร้างรหัสบาร์โค้ด ระบบจัดการสมาชิก ระบบบันทึกข้อมูลการปฏิบัติกิจกรรมจิตอาสา ระบบจัดการเอกสารโครงการ และระบบรายงาน และ 2) ผลการสรุปความพึงพอใจของนักศึกษาที่มีต่อระบบบริหารจัดการโครงการจิตอาสา พบว่า โดยรวมมีค่าเฉลี่ยที่ 4.73 ซึ่งอยู่ในระดับที่มีความพึงพอใจมาก

2.4.2 ระบบการจัดการร้านเฟอร์นิเจอร์ (กรณีศึกษา ร้าน บีบี เฟอร์นิเจอร์)

นายกฤตวิช จงกลฐากร (2561)

ปัจจุบันเทคโนโลยีสารสนเทศได้เข้ามามีบทบาทขึ้นอย่างมาก ทั้งในชีวิตประจำวันการทำงาน และการดำเนินงานขององค์กรต่าง ๆ เทคโนโลยีสารสนเทศมีความสำคัญต่อประสิทธิภาพขององค์กรหรือธุรกิจโดยเฉพาะอย่างยิ่ง ในปัจจุบันโลกมีการเปลี่ยนแปลงตลอดเวลาที่มีการแข่งขันทางธุรกิจสูง องค์กรที่มีการบริการงานที่มีประสิทธิภาพ สามารถเข้าถึงข้อมูลได้อย่างรวดเร็ว ย่อมทำให้องค์กรสามารถดำเนินธุรกิจต่อไปได้ผู้จัดทำจึงได้มีการนำเทคโนโลยีสารสนเทศมาใช้ในธุรกิจเฟอร์นิเจอร์ ในการสร้างระบบการจัดการให้ร้านเฟอร์นิเจอร์ กรณีศึกษา ร้าน บีบี เฟอร์นิเจอร์ ซึ่งปัจจุบันมีผู้ใช้บริการจำนวนมาก และมีระบบการจัดการบริหารงานที่ไม่มีประสิทธิภาพเพียงพอซึ่งจากเดิมการบริหารจัดการร้าน บีบี เฟอร์นิเจอร์ได้จัดเก็บข้อมูลด้วยกระดาษเป็นส่วนใหญ่ ทำให้เกิดความยุ่งยากในการดำเนินธุรกิจ เช่น การบันทึกและการจัดเก็บข้อมูลของกลุ่มลูกค้าที่เข้ามาใช้ซื้อสินค้าการจัดเก็บข้อมูลการขายทำให้เกิดความล่าช้าในการเรียกดูข้อมูล ความเสียหายต่อข้อมูล และยังส่งผลกระทบต่อการทำสรุปยอดขายในแต่ละเดือน จึงจำเป็นต้องอาศัยเทคโนโลยีสารสนเทศเข้ามาช่วยในการบริหาร จัดการธุรกิจ เพื่อเพิ่มประสิทธิภาพในการบริหารงานสามารถเข้าถึงข้อมูลได้อย่างสะดวก และรวดเร็ว ไม่เกิดความผิดพลาด ไม่เสี่ยงต่อการสูญหาย และการเสียหายของข้อมูล

2.4.3 KTn develop

ปัจจุบัน อุปกรณ์ไอทีได้มีบทบาทกับชีวิตความเป็นอยู่ของเราเป็นอย่างมากไม่ว่าจะเป็นมือถือ, แท็บเล็ต, ทีวี, คอมพิวเตอร์ส่วนบุคคล ซึ่งอุปกรณ์เหล่านี้จะมีหน้าจอกการแสดงผลที่แตกต่างกันออกไป ด้วยวิถีชีวิตของคนในสมัยปัจจุบันที่หันมาใช้มือถือกันมากกว่าใช้อุปกรณ์ IT ชนิดอื่น ๆ จึงทำให้มือถือ อีกอุปกรณ์หนึ่งที่เหมาะสมจะทำการตลาด หรือธุรกิจผ่านเว็บไซต์หรือสื่อโซเชียลมีเดียต่าง ๆ ดังนั้นการที่เราจะพัฒนาเว็บไซต์ ให้สามารถรองรับมือถือ และอุปกรณ์IT เหล่านี้จะเป็นเรื่องที่สำคัญอย่างมาก

DesignResponsive Web Design หมายถึง การออกแบบเว็บไซต์ด้วยแนวคิดใหม่ ที่จะสร้างและออกแบบเว็บไซต์ให้เว็บไซต์ สามารถแสดงผลได้อย่างเหมาะสม บนมือถือ, แท็บเล็ต, ทีวี, คอมพิวเตอร์ส่วนบุคคลที่แตกต่างกัน โดยใช้ โค้ดร่วมกัน URL เดียวกัน หน้าจอกการแสดงผลที่แตกต่างกันออกไป

2.4.4 Affde (2564)

Ruby on rails เป็นเฟรมเวิร์คที่คุณควรพิจารณาเป็นอันดับต้น ๆ ในการเรียน คุณอาจเป็นนักพัฒนามือใหม่หรือนักพัฒนาที่มีประสบการณ์มานานแล้ว แต่นั่นก็ไม่นับอุปสรรคต่อการเรียนรู้. Ruby on Rails มีความหลากหลายและสามารถนำไปต่อยอดได้สำหรับโปรเจกต์ขนาดเล็ก หรือขนาดกลางได้เป็นอย่างดี.

เนื่องจาก Ruby on Rails เป็นซอฟต์แวร์โอเพนซอร์ส จึงอนุญาตให้ผู้ใช้แก้ไขซอร์สโค้ดได้มากเท่าที่ต้องการ และยังมีคุณสมบัติที่หลากหลายที่ทำให้เป็นที่นิยมในหมู่นักพัฒนาเช่น คำนึงถึงตัวพิมพ์เล็กและตัวพิมพ์ใหญ่, รองรับการเขียนคอมเมนต์, Ruby on Rails มีรูปแบบภาษาฟรี ซึ่งช่วยให้นักพัฒนาสามารถเริ่มเขียนโค้ดบนบรรทัดหรือคอลัมน์ใดก็ได้ เป็นต้น.

ทำไมคุณควรเลือกใช้ Ruby on Rails ในฐานะนักพัฒนา คุณต้องใช้สิ่งที่มีประสิทธิภาพและไม่ให้คุณเครียด Ruby on Rails มีประโยชน์มากที่ทำให้เป็นเฟรมเวิร์คการพัฒนาระดับแนวหน้า โดยมีเหตุผลคือ Code เขียนง่าย ไม่ซับซ้อน เมื่อเทียบกับ java, ถ้า Code ล้น ดังนั้นโอกาสในการเกิด Bug จึงน้อย, เป็น Open source, มี Library ให้นำไปใช้เป็นจำนวนมาก, ง่ายต่อการนำไปขยายและพัฒนาต่อ (extend) และเป็น pure Object Oriented ส่วนข้อเสียของ Ruby On Rails คือ ผู้พัฒนาต้องการเรียนรู้ตามโครงสร้างของ Rails และ Ruby มี Library ให้เรียกใช้น้อยกว่าภาษาอื่น

2.4.5 โครงการระบบบริหารจัดการร้านจำหน่ายเครื่องใช้ไฟฟ้า

นายเศรษฐพล ศาสนะ (2560)

มีวัตถุประสงค์เพื่อวิเคราะห์และออกแบบระบบงานและสร้างระบบบริหารจัดการร้านจำหน่ายเครื่องใช้ไฟฟ้า กรณีศึกษา : ร้านมิคแอมป์การไฟฟ้า พัฒนาระบบโดยใช้ภาษา PHP เป็นหลักและใช้ระบบการจัดการฐานข้อมูล MySQL เป็นตัวจัดเก็บข้อมูลต่าง ๆ ของระบบ ให้มีประสิทธิภาพมากยิ่งขึ้นจากการพัฒนาระบบบริหารจัดการร้านจำหน่ายเครื่องใช้ไฟฟ้า ทำให้ได้ระบบบริหารจัดการร้านจำหน่ายเครื่องใช้ไฟฟ้าที่สามารถจัดเก็บข้อมูลที่เกี่ยวข้องกับระบบงาน ได้แก่ ข้อมูลพนักงาน ข้อมูลสมาชิก ข้อมูลประเภทสมาชิก ข้อมูลสินค้า ข้อมูลการขาย ข้อมูลการจอง ข้อมูลการเคลม ข้อมูลการย้ายสินค้า ข้อมูลการรับสินค้า ข้อมูลสินค้าคงคลัง ข้อมูลสินค้าวิกฤต ข้อมูลคลังสินค้า พร้อมทั้งออกแบบระบบรายงานต่าง ๆ ได้สะดวกรวดเร็วต่อการใช้งาน ลดความซ้ำซ้อนของข้อมูลและเพิ่มประสิทธิภาพของการทำงานให้ถูกต้อง ผลของการพัฒนาระบบบริหารจัดการร้านจำหน่ายเครื่องใช้ไฟฟ้า ระบบสามารถจัดเก็บข้อมูลต่าง ๆ ได้อย่างรวดเร็วและถูกต้อง ตอบสนองต่อการใช้งานของผู้ใช้ระบบ ทำให้ข้อมูลต่าง ๆ มีประสิทธิภาพและง่ายต่อการจัดเก็บ