

บทที่ 2

แนวคิด ทฤษฎี เครื่องมือและวรรณกรรมที่เกี่ยวข้อง

ในการศึกษาและจัดทำโครงการงานต้องมีการศึกษาและรวบรวมข้อมูลจากแหล่งต่างๆ ซึ่งข้อมูลเหล่านั้นเป็นสารสนเทศที่มีความสำคัญอย่างยิ่งที่จะทำให้การศึกษาและพัฒนาโครงการประสบความสำเร็จลุล่วงไปด้วยดี ทั้งนี้ผู้จัดทำจึงได้จัดทำกรรวบรวมข้อมูลที่ได้เกี่ยวข้องกับการพัฒนาโครงการ ซึ่งประกอบไปด้วย แนวคิด ทฤษฎีและวรรณกรรมที่เกี่ยวข้อง ดังนี้

2.1 แนวคิดที่เกี่ยวข้อง

แนวคิดในการพัฒนาเว็บแอปพลิเคชัน ระบบการพัฒนาเว็บแอปพลิเคชันเพื่อการจัดการกรณีศึกษา สนามยิงธนูพระราม 2 (The development of web application system for management case study of Rama 2 archery) มีความต้องการในการจัดทำระบบขึ้นเพื่อต้องการจัดเก็บข้อมูลและบริหารจัดการข้อมูลภายในสนามยิงธนู ช่วยในการดำเนินงานในกิจการ มีการพัฒนาเพิ่มขึ้นโดยใช้เทคโนโลยีสารสนเทศเข้ามาช่วยในการบริหารงาน เพื่อจัดเก็บข้อมูลในระบบ แทนการบันทึกข้อมูลแบบเดิม โดยการพัฒนาระบบ ในการจัดทำโครงการครั้งนี้มีความคาดหวังว่า ระบบจะช่วยให้การทำงานมีประสิทธิภาพมากขึ้น เพื่อความสะดวก รวดเร็ว และง่ายต่อการจัดเก็บบันทึกข้อมูลต่างๆ และมีการพัฒนาเทคโนโลยีระบบตัวช่วยเหลือสำหรับลูกค้าในการวิเคราะห์ในการยิงธนู รวมถึงการพัฒนาเว็บแอปพลิเคชันแบบเว็บเรสปอนด์ซีฟ(Responsive web design) เพื่อความสะดวกในการใช้งานบนแพลตฟอร์มต่างๆ

มีผู้ใช้งานระบบการพัฒนาเว็บแอปพลิเคชันเพื่อการจัดการ กรณีศึกษา สนามยิงธนูพระราม 2 คือ

1).ผู้ดูแลระบบ สามารถจัดการข้อมูลผู้ใช้ระบบ ข้อมูลพนักงาน ข้อมูลสมาชิก ข้อมูลครุฝึก ข้อมูลเจ้าของกิจการ สามารถจัดการรายการคอร์สที่สอน สามารถจัดการรายละเอียดโปรโมชั่นบนเว็บไซต์ได้

2).แคชเชียร์ สามารถจัดการคิวลูกค้าที่จอง สามารถยืนยันการชำระเงิน คอร์สเรียน สามารถจัดการรายการซ่อมอุปกรณ์ สามารถออกใบเสร็จชำระค่าบริการได้

3).ครูฝึก สามารถดูสมาชิกที่สมัครเรียน สามารถดูตารางฝึก สามารถกำหนดสถานะของคอร์สเรียน สามารถจัดการคอร์สเรียนได้

4).สมาชิก สามารถดูคิวของการบริการ สามารถจองคิว สามารถดูคอร์สเรียน สมัครคอร์สเรียน สามารถดูโปรโมชั่นอื่นๆ สามารถจัดการสถิติ สามารถดูสภาพอากาศได้

5).บุคคลทั่วไป สามารถดูคิวของการบริการ จองคิว สามารถดูคอร์สเรียน สามารถดูโปรโมชั่นอื่นๆ สามารถดูสภาพอากาศได้

6).เจ้าของสนาม สามารถออกรายงานยอดบริการ รายวัน / รายเดือน ออกรายงานยอดขายคอร์สเรียน สามารถดูรายรับ – รายจ่าย/เดือน สามารถจัดการเงินเดือนพนักงาน สามารถออกสลิปเงินเดือนพนักงาน ดูรายงานยอดบริการและคอร์สเรียนในรูปแบบกราฟ สามารถจัดการรายการเบิกอุปกรณ์การซ่อมได้

7).ช่างซ่อม สามารถดูรายการการแจ้งซ่อม สามารถยืนยันสถานะการซ่อม สามารถจัดการรายการเบิกอุปกรณ์การซ่อมได้

ดังนั้นผู้จัดทำได้ทำการรวบรวมข้อมูลที่มีประโยชน์ มีความถูกต้องและน่าเชื่อถือจากบทความ ทฤษฎี และวรรณกรรมที่เกี่ยวข้อง เช่น ทฤษฎีเกี่ยวกับสนามยิงธนู การเคลื่อนที่ของวัตถุแบบโปรเจกไทล์ การออกกำลังกายจากการยิงธนู

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 ทฤษฎีเกี่ยวกับภาษา JavaScript

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปทีละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจกโตรีเยลเต็ด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และ

ภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server) (SEIBOTTECH'S NEWS, 2563)

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อใช้งานกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript JavaScript สามารถทำให้ การสร้างเว็บเพจมีลูกเล่นต่างๆ 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 มีการพัฒนาเป็นเวอร์ชันใหม่ๆ ออกมาด้วย ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้

ข้อดีและข้อเสียของ Java JavaScript

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

อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์เวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุน โดยเฉพาะเช่นกัน แต่ไม่เป็นที่นิยมนัก)

2.2.2 ทฤษฎีเกี่ยวกับฐานข้อมูล

ฐานข้อมูล หมายถึง ที่เก็บข้อมูลและความสัมพันธ์ระหว่างข้อมูลเหล่านั้น (A collection of data and relationships) โดยปกติแล้ว ในเรื่องของฐานข้อมูลมักจะเกี่ยวข้องกับ logical file มากกว่า physical file โดยเฉพาะการออกแบบฐานข้อมูลจะเป็นการออกแบบในส่วนของ logical file ถ้ากล่าวถึง logical file จะเป็นมุมมองของผู้ใช้หรือ application program แต่ถ้ากล่าวถึง physical file จะเป็นมุมมองของ system หรือ operating system การเกี่ยวข้องกันระหว่าง physical file กับ logical file นั่นก็คือ สามารถใช้ physical file มาสร้าง logical file ได้ สำหรับการเปลี่ยน logical file เป็น physical file นั้น ในระดับไฟล์ธรรมดาจะใช้ Operating system แต่ถ้าเป็นฐานข้อมูลจะใช้ระบบจัดการฐานข้อมูลเป็นตัวเปลี่ยน (map) และนำเสนอโครงสร้างข้อมูลให้กับ application หรือผู้ใช้ เช่น ถ้าเราใช้ฐานข้อมูลแบบ relational model โครงสร้างที่เห็นจะเป็นตาราง (relation) แต่ฐานข้อมูลที่มีโครงสร้างแบบ hierarchical model หรือ network model นั้น application หรือผู้ใช้จะมองเห็นเป็น tree และ link list ตามลำดับ

ระบบฐานข้อมูลจะมีลักษณะคล้ายการนำแฟ้มข้อมูล ที่มีความสัมพันธ์กันมาจัดเก็บไว้ด้วยกัน แต่ลักษณะโครงสร้างการจัดเก็บ รวมทั้งวิธีการใช้งานข้อมูลของฐานข้อมูล จะมีความแตกต่างออกไปจากแฟ้มข้อมูล ซึ่งการใช้งานระบบฐานข้อมูลจะต้องมีโปรแกรมที่ทำหน้าที่ในการบริหารจัดการข้อมูลและเป็นตัวกลางระหว่าง ผู้ใช้กับฐานข้อมูล ที่เรียกว่า “Database Management System (DBMS)” หรือระบบจัดการฐานข้อมูล ซึ่งผู้ใช้จะต้องใช้งานฐานข้อมูล ผ่านทางระบบจัดการฐานข้อมูลนี้เท่านั้น

คุณลักษณะที่ดีของฐานข้อมูล (Good Characteristics of Database System)

1) ลดความซ้ำซ้อนของข้อมูลให้เหลือน้อยที่สุด (Minimum redundancy)

เป็นการทำให้ปัญหาเรื่องข้อมูลไม่ตรงกันลดน้อยลงหรือหมดไป โดยนำข้อมูลทั้งหมดมารวมกันเพื่อตัดหรือลดส่วนที่ซ้ำกันทิ้งไป ให้เหลืออยู่เพียงแห่งเดียว และเป็นผลทำให้สามารถแบ่งข้อมูลกันใช้ได้ระหว่างผู้ใช้หลายๆ คน รวมทั้งการใช้ข้อมูลเดียวกันในเวลาพร้อมๆ กันได้อีกด้วย

2) ความถูกต้องสูงสุด (Maximum Integrity : Correctness)

ในระบบฐานข้อมูลจะมีความถูกต้องของข้อมูลสูงสุด เพราะว่าฐานข้อมูลมี DBMS คอยตรวจสอบกฎเกณฑ์หรือเงื่อนไขต่างๆ (Integrity Rules) ให้ทุกครั้งที่มีการแก้ไขข้อมูลหรือเพิ่มเติมข้อมูลเข้าไปในระบบฐานข้อมูลนั้น โดยกฎเกณฑ์เหล่านี้จะเก็บไว้ในฐานข้อมูลตามแนวคิดของ International Organization for Standard (ISO) แต่ในปัจจุบันมี DBMS บาง product ที่ข้อบังคับเหล่านี้ไม่ได้ผูกติดอยู่กับฐานข้อมูลยังคงเก็บอยู่ในโปรแกรม การเปลี่ยนแปลงกฎเกณฑ์เหล่านี้ทำให้ต้องแก้ไขโปรแกรมตามไปด้วยทุกครั้ง ซึ่งไม่สะดวก เช่นเดียวกับระบบแฟ้มข้อมูลเดิมทำให้เกิดความยุ่งยากในการเขียนโปรแกรม แต่ถ้าย้ายการเก็บข้อบังคับหรือกฎเกณฑ์เหล่านี้มาไว้ที่ฐานข้อมูล ระบบจัดการฐานข้อมูลบางชนิดจะมีฟังก์ชันพิเศษ (trigger) กับ procedure อยู่บน FORM ปัจจุบันจะมีให้เลือกว่าจะไว้บนจอหรือไว้ในกฎเกณฑ์กลาง ซึ่งจะเก็บไว้ที่ฐานข้อมูลเรียกว่า stored procedure ซึ่งถูกควบคุมดูแลโดย DBMS สำหรับ DBMS ชั้นดีส่วนใหญ่จะเป็น compile stored procedure เพราะเก็บกฎเกณฑ์เหล่านี้ไว้ที่ stored procedure ไม่ได้เก็บไว้ในโปรแกรมเหมือนระบบแฟ้มข้อมูลเดิม ดังนั้นเมื่อเงื่อนไขเหล่านี้เปลี่ยนแปลงไปก็จะทำการแก้ไขเพียงแห่งเดียว ทำให้ระบบฐานข้อมูลมีความถูกต้องของข้อมูลมากที่สุด และลดค่าใช้จ่ายในการพัฒนา และบำรุงรักษา

3) มีความเป็นอิสระของข้อมูล (Data Independence)

ถือเป็นคุณลักษณะเด่นของฐานข้อมูลซึ่งไม่มีในระบบไฟล์ธรรมดา เนื่องจากในไฟล์ธรรมดาจะเป็นข้อมูลที่ไม่อิสระ (data dependence) กล่าวคือ ข้อมูลเหล่านี้จะผูกพันอยู่กับวิธีการจัดเก็บและการเรียกใช้ข้อมูลซึ่งในลักษณะการเขียนโปรแกรมเราจำเป็นต้องใส่เทคนิคการจัดเก็บและเรียกใช้ข้อมูลไว้ในโปรแกรม เมื่อมีการเปลี่ยนแปลงวิธีการจัดเก็บทำให้ต้องเปลี่ยนแปลงแก้ไขโปรแกรมตามไปด้วย ดังนั้น ถ้าหากมีการแก้ไขหรือเปลี่ยนแปลงโครงสร้างข้อมูลทั้งในระดับ logical และ physical ย่อมมีผลกระทบต่อโปรแกรม แต่ถ้าข้อมูลเก็บในลักษณะของฐานข้อมูลแล้วปัญหานี้จะหมดไป เพราะฐานข้อมูลมี DBMS คอยดูแลจัดการให้ ทำให้โปรแกรมเหล่านี้เป็นอิสระจากการเปลี่ยนแปลงโครงสร้างข้อมูล

4) มีระบบความปลอดภัยของข้อมูลสูง (High Degree of Data Security)

ฐานข้อมูลจะมีระบบรักษาความปลอดภัยของข้อมูลสูง โดย DBMS จะตรวจสอบรหัสผ่าน (login password) เป็นประเด็นแรก หลังจากผ่านเข้าสู่ระบบได้แล้ว DBMS จะตรวจสอบว่าผู้ใช้นั้น มีสิทธิใช้ข้อมูลได้มากน้อยเพียงใด เช่น จะอนุญาตให้ใช้ได้เฉพาะ in query หรือ update และสามารถทำได้เฉพาะตารางใดหรือแถวใดหรือคอลัมน์ใด เป็นต้น นอกจากนี้ โครงสร้างข้อมูลระดับล่างยังถูกซ่อนไว้ไม่ให้ผู้ใช้มองเห็นว่าอยู่ตรงไหน DBMS จะไม่ยอมให้โปรแกรมใดๆ เข้าถึงข้อมูลได้โดยไม่ผ่าน DBMS

5) การควบคุมจะอยู่ที่ส่วนกลาง (Logically Centralized Control)

แนวความคิดนี้จะนำไปสู่ระบบการปฏิบัติงานที่ดี อย่างน้อยสามารถควบคุมความซ้ำซ้อนและความปลอดภัยของข้อมูลได้ นอกจากนี้ในการควบคุมทุกอย่างให้มาอยู่ที่ส่วนกลางจะนำมาสู่ระบบสารสนเทศเพื่อการบริหาร (MIS) โดยต้องมีการควบคุมดูแลจากศูนย์กลางทั้งการใช้และการสร้างโดยหลักการแล้ว จะไม่ยอมให้โปรแกรมเมอร์สร้างตารางหรือวิวเอง แต่จะให้ผู้บริหารฐานข้อมูลเป็นผู้สร้างให้ เพื่อจะได้ทราบว่าตารางหรือวิวซ้ำหรือไม่ นอกจากนี้ผู้บริหารฐานข้อมูลจะเป็นผู้ให้สิทธิแก่ผู้ใช้วิว ดังนั้น โปรแกรมเมอร์จะต้องติดต่อประสานงานกับผู้บริหารฐานข้อมูลในการจัดทำรายงาน คุณลักษณะนี้จะทำให้มีความคล่องตัวในการใช้งาน ซึ่งเป็นผลมาจากข้อมูลมาอยู่รวมกัน (admin.d, 2563)

2.2.3 ทฤษฎีเกี่ยวกับภาษา C#

C# เป็นภาษาเขียนโปรแกรมแบบ multi-paradigm โดยมีรูปแบบกฎเกณฑ์และข้อบังคับในการเขียนที่เข้มงวด ซึ่งมีคุณสมบัติในการเขียนแบบฟังก์ชัน การเขียนทั่วไป และการเขียนโปรแกรมแบบออบเจ็ค มันถูกพัฒนาโดย Microsoft ภายใต้ .NET Framework โดยในการพัฒนาภาษา C# นี้ มีความตั้งใจให้มันเขียนง่าย ทันสมัย เป็นโปรแกรมเพื่อวัตถุประสงค์ทั่วไปและเป็นแบบออบเจ็ค C# เป็นภาษาเขียนโปรแกรมเพื่อวัตถุประสงค์ทั่วไป การพัฒนานั้นนำทีมโดย Anders Hejlsberg (MarcusCode, 2559)

ในการพัฒนาของภาษา C# นั้นมีความตั้งใจว่าเป็นภาษาที่ง่าย ทันสมัย สนับสนุนการเขียนโปรแกรมเพื่อวัตถุประสงค์ทั่วไปและการเขียนโปรแกรมเชิงวัตถุ การพัฒนาของภาษานั้นมีการสนับสนุนสำหรับหลักการของ Software Engineering เช่น การตรวจสอบประเภทข้อมูลที่

เข้มงวด การตรวจสอบขอบเขตของอาเรย์ หรือการพยายามใช้ตัวแปรที่ไม่ได้กำหนดค่า หรือการกำจัด collection ชะงักโน้มนั้ติ ความแข็งแรง ความทนทาน และคุณภาพของโปรแกรม และนอกจากนี้ C# ยังเป็นที่เข้าใจง่ายกับโปรแกรมเมอร์ผู้่คุ้นเคยกับภาษา C และภาษา C++

ภาษา C# ยังถูกออกแบบมาให้เขียนโปรแกรมแบบ GUI (Graphical user interface) สำหรับทำงานบน Windows Form และนอกจากนี้คุณยังสามารถพัฒนาเว็บแอปพลิเคชันและเว็บเซอร์วิสได้ ภายใต้ ASP.NET web framework ที่เป็น Open source จาก Microsoft

ในระหว่างการพัฒนาของ .NET Framework คลาสและไลบรารีต่างๆ ถูกเขียนขึ้นโดยระบบการจัดการโค้ดสำหรับการคอมไพล์ที่เรียกว่า Simple Managed C (SMC) ในเดือนมกราคม 1999 Anders Hejlsberg ได้ก่อตั้งทีมเพื่อสร้างภาษาใหม่ในเวลานั้น ที่เรียกว่า Cool ซึ่งเป็นคำย่อของ "C-like Object Oriented Language" ในเวลาที่โครงการ .NET ของ Microsoft ถูกเผยแพร่ในเดือนกรกฎาคม 2000 ในการประชุมของกลุ่มนักพัฒนามืออาชีพ ภาษาได้ถูกเปลี่ยนชื่อเป็น C# และคลาสไลบรารีและ ASP.NET ได้ถูกเชื่อมเข้ากับ C#

2.2.4 ทฤษฎีเกี่ยวกับการออกแบบหน้าจอ User Interface

User Interface Design หรือ Human-Computer Interaction คือ การออกแบบส่วนต่อประสานกับผู้ใช้ ระหว่างผู้่ใช้กับคอมพิวเตอร์ ซึ่งมีกระบวนการที่เริ่มจากการรวบรวมข้อมูลที่เกี่ยวข้องตลอดจนภูมิความรู้ของนักจิตวิทยา นักการศึกษา นักออกแบบกราฟิก ช่างเทคนิคผู้เชี่ยวชาญด้านมนุษยวิทยา นักออกแบบสถาปัตยกรรมข้อมูล และนักสังคมศาสตร์ เพื่อมาร่วมกันพัฒนาระบบการออกแบบพัฒนาส่วนต่อประสานให้ใช้งานได้อย่างมีประสิทธิภาพ (อ.วิชิตร เทพประสิทธิ์, 2558)

โดยมีวัตถุประสงค์หลักคือ สามารถใช้งานได้ง่าย ใช้ทักษะส่วนบุคคลน้อย มีการฝึกอบรมการใช้งานน้อย เพิ่มมาตรฐานการออกแบบส่วนต่อประสานในระบบ (U.S Military Standard for Human Engineering Design Criteria, 1999) นอกจากนี้ การออกแบบส่วนต่อประสานที่ดีจะทำให้งานที่สำเร็จออกมาดีใช้งานได้ง่าย เรียนรู้ได้ง่าย เมื่อได้ผลงานออกมาก็จะสามารถแข่งขันกับซอฟต์แวร์อื่นๆ ในตลาดได้ ดังที่ Jacob Nielsen ผู้เชี่ยวชาญในการออกแบบ Web Usability ได้กล่าวว่ “Bad usability equal no customers.” ไม่มีใครอยากใช้งานระบบซอฟต์แวร์ที่ใช้งานยาก เพราะเมื่อใช้งานยาก ก็จะไม่มีคนอยากจะใช้

Universal Usability สิ่งที่ต้องคำนึงถึงการออกแบบส่วนต่อประสาน

- 1) ความหลากหลายของผู้ใช้งานทั้งทางกายภาพและสภาพแวดล้อม
- 2) บุคลิกของผู้ใช้ที่แตกต่างกัน / ความต่างระหว่างบุคคล มนุษย์เราย่อมมีความแตกต่างกัน
- 3) ความแตกต่างของสติปัญญาและความสามารถในการรับรู้
- 4) ความหลากหลายทางเชื้อชาติและวัฒนธรรม
- 5) ผู้ใช้งานที่ไร้ความสามารถหรือพิการ
- 6) การออกแบบสำหรับเด็ก เด็กต้องการการออกแบบที่แตกต่างจากผู้ใหญ่ ต้องมีการเฝ้า
ความสนใจสูง
- 7) การปรับให้เข้ากับซอฟต์แวร์และฮาร์ดแวร์ ที่มีอยู่เพื่อไม่ให้เกิดปัญหาความเข้ากันไม่ได้
ของระบบ

ข้อคำนึงดังกล่าวจะเห็นได้ว่า การออกแบบส่วนต่อประสานควรที่จะคำนึงถึงแทบทุกเรื่อง ไม่ว่าจะเป็นเพศ อายุ เชื้อชาติ ศาสนา ก็นำมาเป็นส่วนประกอบในการพิจารณาได้ทั้งสิ้นเช่น ในบางศาสนามีข้อห้ามสำหรับการแสดงภาพสัตว์ สิ่งของบางชนิด, ผู้ใช้งานที่เป็นเด็กจะนิยมภาพที่มีสีสันฉูดฉาดมากกว่าผู้ใหญ่ สภาพแวดล้อมต่างๆ ในการใช้งานก็เป็นส่วนหนึ่งเช่น เมื่อเราจะออกแบบตู้โฆษณากลางแจ้งมีแสงมาก แต่เราออกแบบให้มีสีที่มีการตัดกัน(Contrast) น้อย จะทำให้ผู้ใช้งานมองเห็นข้อความที่เราสื่อไม่ชัดเจน แม้กระทั่งความแตกต่างเฉพาะบุคคลเช่นบางคนชอบอ่านมากกว่าฟัง บางคนชอบภาพเคลื่อนไหว มากกว่าภาพนิ่ง สิ่งเหล่านี้ล้วนเป็นตัวแปรในการออกแบบส่วนต่อประสานทั้งสิ้น

2.2.5 ทฤษฎีเกี่ยวกับชุดคำสั่ง CSS

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลัพท์

ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C (เกล็ดความรู้, 2560)

CSS กับ HTML นั้นทำหน้าที่คนละอย่างกัน โดย HTML จะทำหน้าที่ในการวางโครงร่างเอกสารอย่างเป็นทางการ ถูกต้อง เข้าใจง่าย ไม่เกี่ยวข้องกับการแสดงผล ส่วน CSS จะทำหน้าที่ในการตกแต่งเอกสารให้สวยงาม เรียกได้ว่า HTML คือส่วน coding ส่วน CSS คือส่วน design

2.2.6 ทฤษฎีเกี่ยวกับชุดคำสั่ง API

API ย่อมาจาก Application Programming Interface คือการพัฒนาโปรแกรมสำหรับเป็นส่วนติดต่อเซอร์วิสของแอปพลิเคชัน หรือโมดูลต่างๆ เพื่อให้คนภายนอกมาเรียกใช้งาน หรือกล่าวอย่างง่าย ๆ ก็คือการเขียนโปรแกรมเพื่อให้บริการสำหรับให้คนอื่นมาเรียกใช้งาน ซึ่งแนวคิดเรื่องการสร้าง API เพื่อการใช้งานก็มีมาอย่างยาวนานแล้ว ยกตัวอย่างเช่นตั้งแต่ที่มีการสร้างระบบปฏิบัติการ (OS) ก็จะมีการติดต่อ API ของไดร์เวอร์อุปกรณ์ฮาร์ดแวร์ต่างๆ ของเครื่องคอมพิวเตอร์ เพื่อเรียกใช้งานอุปกรณ์นั้นๆ ตามคำสั่งของนักพัฒนา (นิตี โชติแก้ว, 2559)

API ได้รับการกล่าวถึงและอ้างอิงอีกครั้งในรูปแบบของเว็บเซอร์วิสเทคโนโลยี (Web Service) ซึ่งก็คือเราสามารถพัฒนาแอปพลิเคชันที่ให้บริการข้อมูลบนโปรโตคอล HTTP โดยใช้หลักการและแนวคิดของ API เพื่อให้คนภายนอกมาเรียกใช้งาน

หลักคิดในการที่จะพัฒนาต้องเข้าใจก่อนว่าเทรนด์การพัฒนาเว็บแอปพลิเคชันในยุคปัจจุบันนั้นเปลี่ยนแปลงไปจากเมื่อก่อนเป็นอย่างมาก เนื่องด้วยการเติบโตอย่างก้าวกระโดดของโมบายดีไวซ์ก็คือโทรศัพท์มือถือแบบสมาร์ทโฟน โดยเฉพาะการที่โทรศัพท์มือถือสามารถเข้าถึงอินเทอร์เน็ตได้ การทำให้เว็บแอปพลิเคชันจะต้องสามารถแสดงผลได้อย่างรวดเร็วและได้ทุกขนาดหน้าจอจึงเป็นเรื่องยากลำบากมากในช่วงแรก แต่ ณ ปัจจุบันปัญหาเหล่านี้ก็ค่อยๆ ลดลงอย่างต่อเนื่องเพราะมีผู้คิดค้นเครื่องมือการพัฒนาเว็บแอปพลิเคชันทั้ง Framework, Library และออกแบบสถาปัตยกรรมที่เหมาะสมกับโมบายแอปพลิเคชันโดยเฉพาะ

แท้จริงแล้วหัวใจหลักของการพัฒนา API คือการใช้แนวคิดด้านสถาปัตยกรรมเชิงเซอร์วิส SOA (Service-Oriented Architecture) เพื่อประยุกต์ใช้ออกแบบที่จะดึงความสามารถศักยภาพให้เว็บแอปพลิเคชันสามารถตอบโจทยปัญหาในยุคที่ไคลเอนต์ (Client) ที่มีจำนวนมหาศาลและหลากหลาย หากวันนี้ถ้ายังคงใช้แนวคิดและวิธีการแบบเดิมๆ เชื่อว่าเว็บแอปพลิเคชันอย่าง Facebook คงยังไม่สามารถรองรับผู้ใช้งานมหาศาลจำนวนกว่าหนึ่งพันห้าร้อยล้านจากคนทั่วโลกได้

SOA จะถูกสร้างอยู่บนชั้นของเซอร์วิสในที่นี้คือ Application Service Layer ซึ่งเป็นที่รวมของฟังก์ชันการทำงานของระบบ และเซอร์วิสเหล่านั้นจะไม่มี การเชื่อมต่อกับส่วนแสดงผลโดยตรง นั่นแสดงว่าในส่วนของ การแสดงผล (Presentation Layer) จะเป็นอะไรก็ได้ ไม่ว่าจะเป็นหน้าเว็บบน PC, iPhone App, Android App หรือ ระบบอื่นๆ ดังรูป ทำให้ระบบมีความยืดหยุ่นและสามารถขยายตัวได้อย่างสูง สามารถรองรับการไหลคของไคลเอนต์ ได้อย่างไม่จำกัดและมีประสิทธิภาพ เช่น การยกเอาชั้นเซอร์วิสไปรันไว้บนการประมวลผลแบบกลุ่มเมฆ (cloud) เพื่อให้เกิดการขยายตัวได้อย่างสูงสุด

2.3 เครื่องมือในการออกแบบและวิเคราะห์ระบบ

2.3.1 Flow chart

ผังงาน (Flowchart) คือ รูปภาพ หรือ สัญลักษณ์ที่ใช้เขียนแทนขั้นตอน คำอธิบาย ข้อความ หรือ คำพูดที่ใช้ในอัลกอริทึม (Algorithm) เพราะการนำเสนอขั้นตอนของงานให้เข้าใจตรงกันระหว่างผู้เกี่ยวข้อง ด้วยคำพูด หรือ ข้อความทำได้ยากกว่า สามารถแบ่งเป็น 2 ประเภทคือ

1) ผังงานระบบ (System Flowchart) คือ ผังงานที่แสดงขั้นตอนการทำงานในระบบอย่างกว้างๆ แต่ไม่เจาะลงในระบบงานย่อย

2) ผังงานโปรแกรม (Program Flowchart) คือ ผังงานที่แสดงถึงขั้นตอนในการทำงานของโปรแกรม ตั้งแต่รับข้อมูล คำนวณ จนถึงแสดงผลลัพธ์

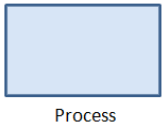
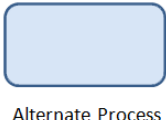
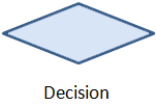
โปรแกรมแบบมีโครงสร้างประกอบด้วยหลักการ 3 อย่าง คือ

1) การทำงานแบบตามลำดับ (Sequence) เป็นรูปแบบการเขียนโปรแกรมที่ง่ายที่สุดคือ เขียนให้ทำงานจากบนลงล่าง เขียนคำสั่งเป็นบรรทัด และทำทีละบรรทัดจากบรรทัดบนสุดลงไป จนถึงบรรทัดล่างสุด สมมติให้มีการทำงาน 3 กระบวนการคือ อ่านข้อมูล คำนวณ และพิมพ์












2) การเลือกกระทำตามเงื่อนไข (Decision) เป็นการตัดสินใจ หรือเลือกเงื่อนไขคือ เขียนโปรแกรมเพื่อนำค่าไปเลือกกระทำ โดยปกติจะมีเหตุการณ์ให้ทำ 2 กระบวนการ คือเงื่อนไขเป็นจริงจะกระทำกระบวนการหนึ่ง และเป็นเท็จจะกระทำอีกกระบวนการหนึ่ง แต่ถ้าซับซ้อนมากขึ้น จะต้องใช้เงื่อนไขหลายชั้น เช่น การตัดเกรดนักศึกษา เป็นต้น ตัวอย่างผังงานนี้ จะแสดงผลการเลือกอย่างง่าย เพื่อกระทำกระบวนการเพียงกระบวนการเดียว

3) การทำซ้ำ (Loop) เป็นการทำกระบวนการหนึ่งหลายครั้ง โดยมีเงื่อนไขในการควบคุม หมายถึงการทำซ้ำเป็นหลักการที่ทำความเข้าใจได้ยากกว่า 2 รูปแบบแรก เพราะการเขียนโปรแกรมแต่ละภาษา จะไม่แสดงภาพอย่างชัดเจนเหมือนการเขียนผังงาน ผู้เขียนโปรแกรมต้องจินตนาการด้วยตนเอง

ตารางที่ 2.1 ความหมายของสัญลักษณ์ Flowchart

สัญลักษณ์	ความหมาย
	การกำหนดค่า หรือ การประมวลผลทั่วไป
	การประมวลของโปรแกรมย่อย (Subroutine)
	การตัดสินใจ การเปรียบเทียบ จะมีผลใน 2 ทิศทาง คือ กรณีผลตรวจสอบเงื่อนไขเป็นเท็จ และเป็นจริง

ตารางที่ 2.1 ความหมายของสัญลักษณ์ Flowchart (ต่อ)

สัญลักษณ์	ความหมาย
 Data	รับ หรือ แสดงข้อมูล โดยไม่ระบุชนิดอุปกรณ์
 Predefined Process	โปรแกรมย่อย หรือ โมดูล เริ่มทำงานหลักจากจุดคำสั่งในโปรแกรมย่อยแล้ว จะกลับมาทำคำสั่งต่อไป
 Document	การแสดงผลเอกสาร หรือ การแสดงผลออกทางเครื่องพิมพ์
 Terminator	การเริ่มต้น หรือ การสิ้นสุด
 Manual Input	การรับข้อมูลเข้าทางแป้นพิมพ์
 Connector	จุดเชื่อมต่อในหน้าเดียวกัน
 Off-page Connector	จุดเชื่อมต่อคนละหน้า
 Direct Access Storage	การจัดเก็บข้อมูลแบบการเข้าถึงโดยตรง
 Display	จอภาพแสดงผล
 Internal Storage	การเก็บข้อมูลภายใน
 Multidocument	การแสดงผลหลายเอกสารพร้อมกัน

2.3.2 แผนภาพบริบท (Context Diagram)

แผนภาพบริบท (Context Diagram) คือ แผนภาพกระแสข้อมูลระดับบนสุด ที่แสดงภาพรวม สูงสุดของระบบ ซึ่งจะแสดงถึงสิ่งแวดล้อมของระบบและองค์ประกอบหลักๆ เท่านั้น โดยที่ จะมี เพียง 1 Process ซึ่งเป็นชื่อของระบบ (0) และไม่มี Data Store ปรากฏอยู่ใน Context Diagram โดยเด็ดขาด จะแสดงภาพรวมการทำงานของระบบที่สัมพันธ์กับสภาพแวดล้อมภายนอกระบบ

2.3.3 แผนภาพกระแสข้อมูล (Data Flow Diagram : DFD)

DFD คือ แผนภาพกระแสข้อมูลที่มีการวิเคราะห์แบบในเชิงโครงสร้าง (Structure) ซึ่งเป็นแผนภาพที่บอกถึงรายละเอียดของระบบ โดยเฉพาะข้อมูล และฟังก์ชันของข้อมูล สิ่งที่ได้จาก DFD ประกอบด้วย

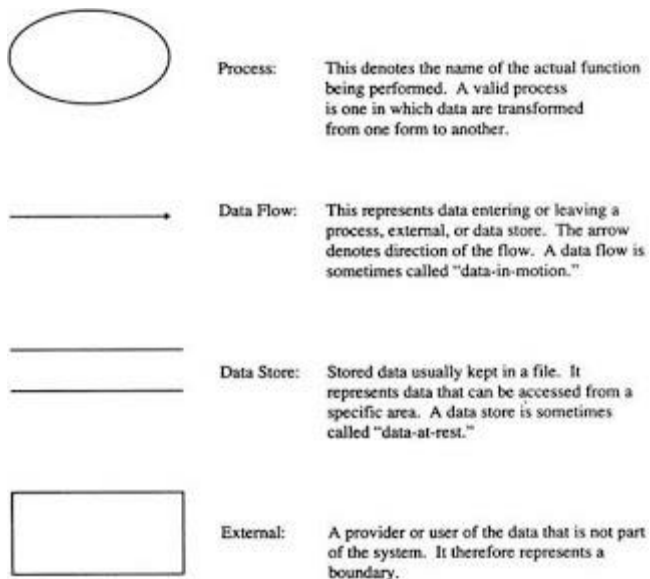
- ข้อมูลมาจากไหน
- ข้อมูลไปที่ใด
- ข้อมูลเก็บที่ใด
- เกิดเหตุการณ์ใดกับข้อมูลบ้าง

2.3.3.1 ขั้นตอนของการวิเคราะห์เพื่อสร้าง DFD

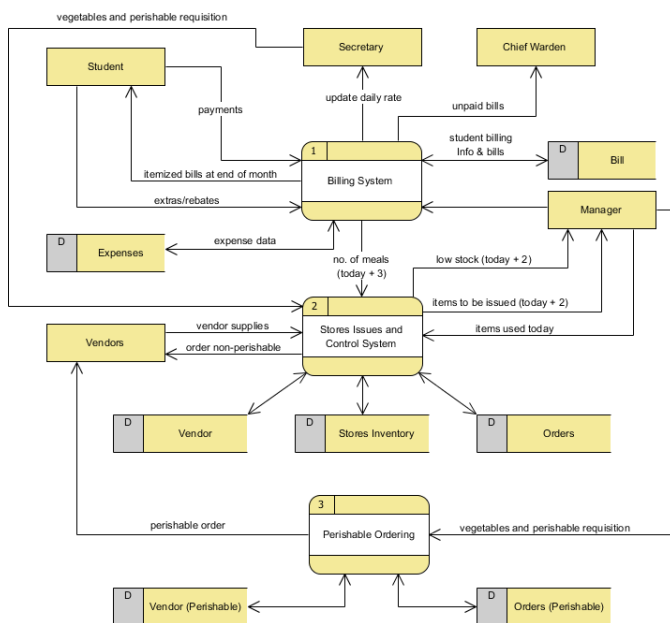
- 1) ศึกษารูปแบบการทำงานในลักษณะ Physical ระบบงานเดิม
- 2) ดำเนินการวิเคราะห์เพื่อให้ได้แบบจำลอง Logical ระบบงานเดิม
- 3) เพิ่มเติมการทำงานใหม่ภายในแบบจำลอง Logical ระบบงานเดิม
- 4) พัฒนาระบบงานใหม่ในรูปแบบของ Physical

2.3.3.2 วัตถุประสงค์ของ DFD

- 1) เป็นแผนภาพสรุปรวมข้อมูลทั้งหมดที่ได้จากการวิเคราะห์
- 2) เป็นข้อตกลงร่วมกันระหว่าง SA และ User
- 3) เป็นแผนภาพที่ใช้ในการพัฒนาต่อในขั้นตอนออกแบบ
- 4) เป็นแผนภาพที่ใช้ในการอ้างอิง หรือเพื่อใช้พัฒนาต่อ
- 5) ทราบที่ไปที่มาของกระบวนการต่างๆ



ภาพที่ 2.1 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูล



ภาพที่ 2.2 DFD Example (ตัวอย่าง ภาพกระแสข้อมูล)

2.3.3.3 กฎเกณฑ์การเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์ของแผนภาพไม่สามารถเชื่อมต่อกันได้โดยตรง ซึ่งต้องมี Flow บอกทิศทางของกระแส (Flow ระบุข้อมูล) และการ Flow ทุกครั้งจะต้องผ่าน Process ก่อนทุกครั้ง

- Process = กิริยา
- Flow = ข้อมูล
- Boundaries, Entity = องค์กร, หน่วยงาน, ผู้ใช้งานระบบ

2.3.3.4 ขั้นตอนการเขียน DFD

- 1) วิเคราะห์ให้ได้ว่าระบบประกอบไปด้วย Boundaries ใดบ้างที่เกี่ยวข้อง
- 2) ดำเนินการออกแบบระบบในระดับหลักการ หรือ Context Diagram
- 3) วิเคราะห์ข้อมูลในระบบว่าควรมีข้อมูลใดบ้าง
- 4) วิเคราะห์กระบวนการหรือ Process ในระบบว่าควรมี Process หลักใด และประกอบไปด้วย Process ย่อยใดบ้าง
- 5) ดำเนินการเขียนแผนภาพกระแสข้อมูลในระดับต่างๆ
- 6) ทำการตรวจสอบ Balancing และปรับแก้ Redraw จนได้แผนภาพที่สมบูรณ์
- 7) อาจใช้ CASE Tools ช่วยในการเขียนแผนภาพ

2.3.3.5 DFD Level 0

จะนำ Context Diagram มาแตกรายละเอียดภายใน ซึ่งจะแสดงถึง Process หลักๆ ผู้เกี่ยวข้อง, ข้อมูลภายใน ที่มีความละเอียดมากขึ้น (Top down Design) ในระดับนี้จะปรากฏทุกๆ ชนิดของ Object DFD จะต้องมีการกำกับหมายเลข Process ด้วยเลข 0

2.3.3.6 DFD Level 1

เป็นแผนภาพ DFD ในระดับย่อยลงมา ที่แสดงรายละเอียด Data Flow และ Process ย่อยลงมาของ DFD Level 0 เพื่อเพิ่มความละเอียดของ กระบวนการมากยิ่งขึ้นแต่ตั้งแต่ Level ที่ 1 ลงไป จะมีแผนภาพนี้ขึ้นตามความจำเป็นเท่านั้น (ซึ่งขึ้นอยู่กับความซับซ้อนของข้อมูล และกิจกรรมที่ต้องการแตกรายละเอียด)

2.3.4 แผนภาพกระแสข้อมูล (Data Flow Diagram: DFD)

ER Diagram คือ แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพการอธิบายโครงสร้างและความสัมพันธ์ของข้อมูล (Relationship) ประกอบด้วย

- เอนทิตี (Entity) เป็นวัตถุ หรือสิ่งของที่เราสนใจในระบบงานนั้นๆ
- แอททริบิว (Attribute) เป็นคุณสมบัติของวัตถุที่เราสนใจ
- ความสัมพันธ์ (Relationship) คือ ความสัมพันธ์ระหว่างเอนทิตี

ER Diagram มีความสำคัญต่อการพัฒนาระบบงานฐานข้อมูล Application ต่างๆ ที่ต้องการการเก็บข้อมูลอย่างมีระบบ มีโครงสร้าง ดังนั้น ER Diagram จึงใช้เพื่อเป็นเอกสารในการสื่อสารระหว่าง นักออกแบบระบบ และนักพัฒนาระบบ เพื่อให้สื่อสารอย่างตรงกัน และเป็นสากลอีกด้วย

2.3.5 แบบของข้อมูล (Data Type)

โครงสร้างฐานข้อมูลของระบบสารสนเทศสำหรับติดตามงานวิจัยเก็บภายใต้โปรแกรมฐานข้อมูลเรียลไทม์ ไฟร์เบส (Firebase) ซึ่งมี API โดยใช้จัดเก็บและชิงค์ข้อมูล ฐานข้อมูลมีลักษณะแบบของข้อมูล (data type) ดังตารางที่ 2.18

2.3.6 แบบจำลองความสัมพันธ์เอนทิตี (E-R Diagram หรือ Entity-relationship model)

แบบจำลองความสัมพันธ์เอนทิตี หรือ อี-อาร์ โมเดล (อังกฤษ: Entity-relationship model, คำย่อ ERM) หรือ อี-อาร์ไดอะแกรม (E-R Diagram) เป็นวิธีที่ช่วยในการออกแบบฐานข้อมูลและได้รับความนิยมอย่างมาก นำเสนอโดย Peter ซึ่งวิธีการนี้อยู่ในระดับ Conceptual level และมีหลักการคล้ายกับ Relational model เพียงแต่ E-R model แสดงในรูปแบบกราฟิก บางระบบจะใช้ E-R model ได้เหมาะสมกว่า แต่บางระบบจะใช้ Relational model ได้เหมาะสมกว่าเป็นต้น ซึ่งแล้วแต่การพิจารณาของผู้ออกแบบว่าจะเลือกใช้แบบใด

E-R Diagram หรือ Entity Relationship Diagram คือแผนผังแสดงความสัมพันธ์ระหว่าง Entity หรือกลุ่มข้อมูล ซึ่งจะแสดงชนิดของความสัมพันธ์ว่าเป็นชนิด หนึ่งต่อหนึ่ง (One to One), หนึ่งต่อหลายสิ่ง (One to Many), หรือ หลายสิ่งต่อหลายสิ่ง (Many to Many)

2.3.6.1) ส่วนประกอบ

1) เอนทิตี (Entity) หมายถึง ชื่อของสิ่งใดสิ่งหนึ่ง เปรียบเสมือนค่านาม อาจได้แก่ คน สถานที่ สิ่งของ การกระทำซึ่งต้องการจัดเก็บข้อมูลไว้ เช่น เอนทิตีของลูกค้าเอนทิตีของพนักงาน เป็นต้น บางเอนทิตีอาจจะไม่มีความหมายเลย หากขาดเอนทิตีอื่นในฐานะข้อมูล เช่น เอนทิตีประวัติ นักศึกษาจะไม่มีความหมาย หากปราศจากเอนทิตีนักศึกษาเพราะจะไม่ทราบว่าเป็นประวัติของ นักศึกษาคนใด

2) แอททริบิวต์ (Attribute) หมายถึง รายละเอียดข้อมูลที่แสดงลักษณะและคุณสมบัติ ของเอนทิตีหนึ่งๆ เช่น เอนทิตีนักศึกษา ประกอบด้วย แอททริบิวต์รหัสนักศึกษาชื่อนักเรียนศึกษา ที่อยู่นักศึกษา เป็นต้น บางเอนทิตีก็ยังประกอบด้วยข้อมูลหลายส่วน กลายแอททริบิวต์ย่อยมา รวมกัน เช่น แอททริบิวต์ที่อยู่นักศึกษา ประกอบด้วย บ้านเลขที่ ถนนตำบล อำเภอ จังหวัด ดังนั้น แอททริบิวต์ที่อยู่นักศึกษาจึงเป็น แอททริบิวต์ผสม (CompositeAttribute) บางแอททริบิวต์ก็อาจจะ ไม่มีค่าของตัวเอง แต่จะสามารถหาค่าได้จากแอททริบิวต์อื่น เช่น แอททริบิวต์อายุ อาจคำนวณ ได้จาก แอททริบิวต์วันเกิด ลักษณะเช่นนี้จึงอาจเรียกแอททริบิวต์อายุว่าเป็น แอททริบิวต์ที่แปรผล ค่ามา (Derived Attribute)

3) ความสัมพันธ์ระหว่างเอนทิตี (Relationship) คือ การสร้างความสัมพันธ์ระหว่าง เอนทิตีในฐานข้อมูลเชิงสัมพันธ์กระทำได้ โดยการกำหนดให้เอนทิตีที่มีความสัมพันธ์กันมีแอททริ บิวต์ที่เหมือนกัน และใช้ค่าของแอททริบิวต์ที่เหมือนกันเป็นตัวระบุข้อมูลในเอนทิตีที่มีความสัมพันธ์ กัน ความสัมพันธ์ระหว่างเอนทิตี แบ่งออกเป็น 3 ประเภท ดังนี้

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-one Relationship) คือ เป็นการแสดง ความสัมพันธ์ของข้อมูลในเอนทิตีหนึ่ง ว่ามีความสัมพันธ์กับข้อมูลของอีกเอนทิตีหนึ่ง ในลักษณะที่ เป็นหนึ่งต่อหนึ่ง เช่น ความสัมพันธ์ของประชาชนกับหมายเลขรหัสประจำตัวประชาชน ซึ่งประมาณ 1 คน จะต้องมียาหมายเลขรหัสประจำตัวประชาชน 1 หมายเลข ไม่ซ้ำกัน

- ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-many Relationship) คือเป็นการแสดง ความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่ง ว่ามีความสัมพันธ์กับข้อมูลหลายข้อมูลกับอีกเอนทิตีหนึ่ง เช่น ความสัมพันธ์ของ แผนกกับพนักงาน ซึ่งแผนกแต่ละแผนกจะประกอบไปด้วยพนักงานที่สังกัด อยู่ในแผนกหลายคน

- ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-many Relationship) เป็นการแสดงความสัมพันธ์ของข้อมูลของ 2 เอนทิตี ในลักษณะแบบกลุ่มต่อกลุ่มเช่น ความสัมพันธ์ของนักศึกษา กับหลักสูตร นักศึกษาหลายคน อาจเรียนอยู่ในหลายหลักสูตรคือ จำนวนเอนทิตีในการมีส่วนร่วม (Participation) ของความสัมพันธ์ระหว่างกัน ซึ่งจำนวนความสัมพันธ์ระหว่างเอนทิตีมีอยู่ คือ ความสัมพันธ์ 1 Entity(Unary relationship) ความสัมพันธ์ 2 Entity (Binary relationship) ความสัมพันธ์ 3 Entity(Ternary relationship)

2.3.6.2) ขั้นตอนการเขียนอีอาร์โมเดล

1) กำหนด Entity type โดยกำหนดมาจากความต้องการของผู้ใช้ระบบว่าจะให้มี Entity สำหรับเก็บข้อมูลอะไรบ้าง เอนทิตี (Entity) อาจเรียกว่า file หรือ tables Strong entity คือเกิดขึ้นด้วยตนเองไม่ขึ้นกับ entity ใด เช่น นักศึกษา หรือ อาจารย์หรือสินค้า เป็นต้น

Weak entity ขึ้นโดยอาศัย entity อื่น เช่น เกรดเฉลี่ย ที่มาจากแฟ้มผลการเรียน หรือ แฟ้มลงทะเบียน หรือ แฟ้มสั่งซื้อ เป็นต้น สิ่งต่างๆ ที่ผู้ใช้งานฐานข้อมูลจะต้องยุ่งเกี่ยวกับ เช่น คน แพนก ประเภท การสั่งซื้อ

2) กำหนดความสัมพันธ์ (Relationship type) ที่เกิดขึ้นระหว่าง entity ในลักษณะของกริยา ดีกรีของความสัมพันธ์ (Degree of relation) มี 4 แบบ

- Unary relationship คือความสัมพันธ์ภายใน entity เดียวกัน เช่น ตำแหน่งงานของพนักงาน แต่ถ้ามีระดับแบบลูกน้อง หัวหน้าจะเรียก Recursive relationship



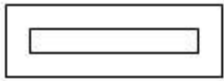
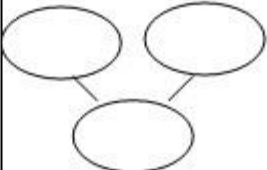
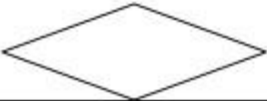





- Binary relationship คือความสัมพันธ์แบบสอง entity

- Ternary relationship คือความสัมพันธ์แบบสาม entity

- Quaternary relationship คือความสัมพันธ์แบบสี่ entity

3) แอททริบิวต์ (Attribute) อาจเรียก field หรือ column คือ สิ่งที่ใช้อธิบายคุณสมบัติของเอนทิตี เช่นคุณสมบัติของคน ก็มี รหัส ชื่อ อายุ เพศ เป็นต้น

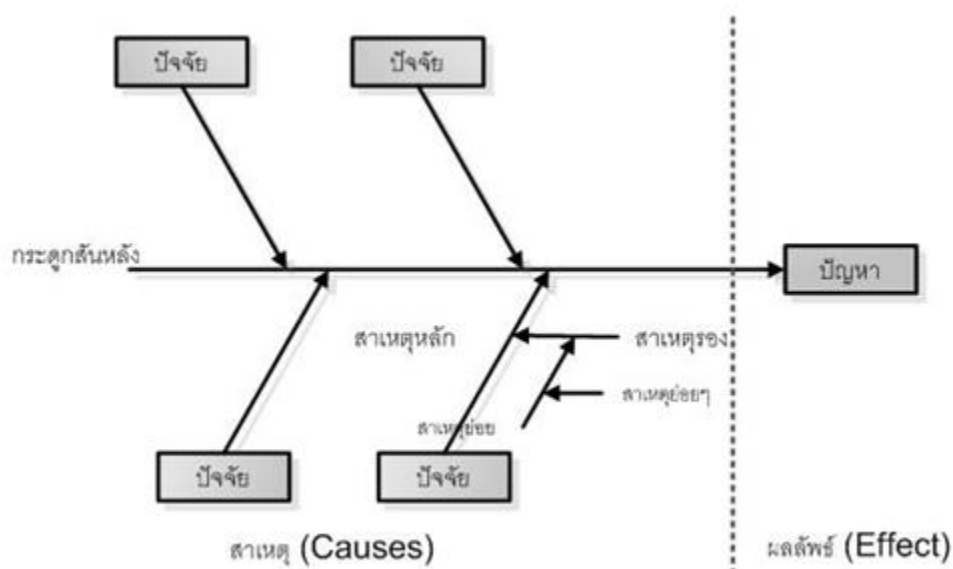
ตารางที่ 2.2 สัญลักษณ์ของ ER-Diagram

สัญลักษณ์	ความหมาย	สัญลักษณ์	ความหมาย
	Entity set		Discriminator key attribute
	Weak entity set		Composite attribute
	Relationship set		Derived attribute
	Identifying relationship set		Key attribute
	Attribute		Multi valued attribute

2.3.7 แผนภูมิแก้งปลา

แผนภูมิแก้งปลาหรือแผนผังสาเหตุและผล (Cause And Effect Diagram) เป็นเครื่องมือทางการบริหารรูปแบบหนึ่งซึ่งช่วยในการวิเคราะห์หาสาเหตุของปัญหาที่เกิดขึ้น จากการดำเนินงานทางธุรกิจถือว่าเป็นเรื่องรวมปกติ ซึ่งอาจประกอบไปด้วยปัญหาเพียงเล็กน้อยจนถึงปัญหาระดับใหญ่ ถึงแม้ว่าจะเป็นปัญหาเพียงเล็กน้อยหรือเป็นปัญหาใหญ่ก็สมควรอย่างยิ่งที่จะต้องได้รับการแก้ไขปัญหาดังกล่าว เนื่องจากปัญหาได้รับการพอกพูนอย่างต่อเนื่องโดยไม่ได้รับการเอาใจใส่ นอกจากจะส่งผลกระทบต่อประสิทธิภาพโดยรวมในด้านการดำเนินงานแล้ว อาจทำให้ธุรกิจได้รับผลกระทบและส่งผลกระทบต่อความเสียหายหรือล่มสลายได้ ในขณะเดียวกันหากธุรกิจใดที่สามารถจัดการกับปัญหาและแก้ไขปัญหาล่วงไปได้ด้วยดีย่อมหมายถึงความสำเร็จในการแก้ไขปัญหาเพื่อให้ธุรกิจสามารถดำรงอยู่และก้าวไปสู่ความสำเร็จตามเป้าหมาย หลักการแก้ไขปัญหที่ดี

นักวิเคราะห์ระบบควรมีการกำหนดหัวข้อของปัญหาและหาสาเหตุของปัญหาให้ได้ก่อน ซึ่งแนวทางหนึ่งที่สามารถใช้ได้เป็นอย่างดีคือการเอามาประยุกต์เขียนแผนภูมิแก้างปลา ซึ่งแผนภูมิแก้างปลาสามารถเรียกได้หลายชื่อ Fishbone Diagram เช่น Cause-and-Effect Diagram หรือ Ishikawa Diagram



ภาพที่ 2.3 รูปแบบการเขียนแผนภูมิแก้างปลา

2.3.8 พจนานุกรมข้อมูล (Data Dictionary)

พจนานุกรมข้อมูล (Data Dictionary) เป็นเครื่องมือที่ช่วยในการจัดเก็บรายละเอียดต่างๆ เกี่ยวกับข้อมูลให้เป็นหมวดหมู่ ทำให้สามารถค้นหารายละเอียดที่ต้องการได้โดยสะดวก ตัวอย่างเช่น ผู้ใช้อาจเก็บข้อมูลเกี่ยวกับรายงานต่าง ๆ ไว้ภายในหมวดรายการชื่อ “Report” เป็นต้น ทั้งนี้วัตถุประสงค์ของการจัดเก็บรายละเอียดเกี่ยวกับข้อมูลให้เป็นหมวดหมู่ในพจนานุกรมข้อมูล เพื่อให้สามารถอธิบายความหมายของข้อมูลต่างๆ แก่ผู้ใช้งานได้อย่างถูกต้องและเป็นมาตรฐานเดียวกันทั้งหน่วยงาน

ในการกำหนดโครงสร้างของฐานข้อมูลระบบจัดการฐานข้อมูล (Data Base Management System : DBMS) ซึ่งเป็นซอฟต์แวร์ที่ทำหน้าที่เป็นสื่อกลางประสานงานระหว่างผู้ใช้กับฐานข้อมูล โดยทำการควบคุม ดูแล และจัดการเรื่องต่างๆ ที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูลตัวอย่างเช่น การจัดเก็บและดูแลรักษาข้อมูล การปรับปรุงข้อมูล และการเรียกใช้ข้อมูล เป็นต้น โดยจะทำการ

เก็บรวบรวมรายละเอียดและคำอธิบายต่างๆ ที่เกี่ยวข้องกับข้อมูลตัวอย่างเช่น ชื่อตาราง (Table) ชื่อขอบเขตข้อมูล (Field) และคีย์ต่างๆ เป็นต้น ไว้ในพจนานุกรมข้อมูลที่มีการสร้างขึ้นมาเป็นส่วนหนึ่งของฐานข้อมูล

พจนานุกรมข้อมูล (Data Dictionary) จึงเป็นสิ่งที่มีความสำคัญอย่างยิ่งต่อการจัดเก็บรายละเอียดของข้อมูลไว้อย่างเป็นระบบ เนื่องจากทุกฐานข้อมูลจะมีการจัดเก็บรายละเอียดต่างๆ เกี่ยวกับข้อมูล (Metadata) ภายในฐานข้อมูล ตัวอย่างเช่น โครงร่างของฐานข้อมูลระดับภายนอก (External Schema) โครงร่างของฐานข้อมูลระดับแนวคิด (Conceptual Schema) และโครงร่างของฐานข้อมูลระดับภายใน (Internal Schema) เป็นต้น ซึ่งส่วนที่ใช้สำหรับจัดเก็บข้อมูลลักษณะดังกล่าว คือ พจนานุกรมข้อมูล หรือเรียกอีกอย่างหนึ่งว่า System Catalog

โครงสร้างฐานข้อมูลโดยใช้ภาษาเอสคิวแอล (SQL) ในการจัดการฐานข้อมูล มีลักษณะแบบของข้อมูล (Data Type) ดังต่อไปนี้

ตารางที่ 2.3 ประเภทข้อมูลชนิดตัวอักษร

ลำดับที่	ชื่อประเภทข้อมูล	รายละเอียด	เนื้อที่เก็บข้อมูล
1	VARCHAR(M)	สำหรับเก็บข้อมูลประเภทตัวอักษร ทุกครั้งที่เลือกชนิดของฟิลด์เป็นประเภทนี้ จะต้องมีการ กำหนดความยาวของข้อมูล ลงไปด้วย ซึ่งสามารถ กำหนด ค่าได้ ตั้งแต่ 1 - 255 ฟิลด์	ขนาดข้อมูลจริง 1byte
2	CHAR(M)	สำหรับเก็บข้อมูลประเภทตัวอักษรแบบที่ ถูก จำกัดความกว้างเอาไว้คือ 255 ตัวอักษร ไม่ สามารถปรับเปลี่ยนได้ เหมือนกับ VARCHAR หาก ทำการสืบค้น โดยเรียงตามลำดับก็จะเรียงข้อมูล	ตามจำนวน อักขรที่ระบุ
3	TINYTEXT	ในกรณีที่ข้อความยาวๆ หรือต้องการที่จะ ค้นหา ข้อความ โดยอาศัยฟิเจอร์ FULL	ขนาดข้อมูลจริง 1byte

ตารางที่ 2.3 ประเภทข้อมูลชนิดตัวอักษร (ต่อ)

		เลือกที่จะไม่เก็บข้อมูลลงในฟิลด์ประเภท VARCHAR ที่มีข้อจำกัด	
4	TEXT	สำหรับเก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่สามารถเก็บได้มากขึ้น โดยสูงสุดคือ 65,535 ตัวอักษร หรือ 64KB เหมาะสำหรับเก็บ ข้อมูลพวก เนื้อหาต่างๆ ที่ยาวๆ	ขนาดข้อมูลจริง 2byte
5	MEDIUMTEXT	เก็บข้อมูลประเภทตัวอักษรเช่นเดียวกับ TINYTEXT แต่เก็บข้อมูลได้ 16,777,215 ตัวอักษร	ขนาดข้อมูลจริง 3byte
6	LONGTEXT	เก็บข้อมูลประเภทตัวอักษรเช่นเดียวกับ TINYTEXT แต่เก็บข้อมูลได้ 4,294,967,295 ตัวอักษร	ขนาดข้อมูลจริง 4byte
7	ENUM	เป็นข้อมูลประเภทระบุค่าที่ต้องการ หรือ ถ้าไม่มีจะให้ค่า null สามารถกำหนดค่าได้ถึง 65,535 ตัวอักษร	ตามจำนวนอักษร ที่ระบุ

ตารางที่ 2.4 ประเภทข้อมูลชนิดจำนวนเต็ม

ลำดับที่	ชื่อประเภทข้อมูล	ค่าตัวเลขแบบมีเครื่องหมาย	ค่าตัวเลขแบบไม่มีเครื่องหมาย	เนื้อที่เก็บข้อมูล
1	TINYINT(M)	-128 ถึง 127	0 ถึง 255	1 byte
2	SMALLINT(M)	-32768 ถึง 32767	0 ถึง 65535	2 byte
3	MEDIUMINT(M)	-8388608 ถึง 8388607	0 ถึง 16777215	3 byte
4	INT(M) หรือ INTEGER(M)	-2147483648 ถึง 2147483647	0 ถึง 4294967295	4 byte

ตารางที่ 2.4 ประเภทข้อมูลชนิดจำนวนเต็ม(ต่อ)

ลำดับที่	ชื่อประเภทข้อมูล	ค่าตัวเลขแบบมีเครื่องหมาย	ค่าตัวเลขแบบไม่มีเครื่องหมาย	เนื้อที่เก็บข้อมูล
5	BIGINT(M)	- 9223372036854775808 ถึง 9223372036854775807	0 ถึง 1844674407370 9551615	8 byte

ตารางที่ 2.5 ประเภทข้อมูลชนิดจำนวนทศนิยม

ลำดับที่	ชื่อประเภทข้อมูล	ค่าตัวเลขแบบมีเครื่องหมาย	ค่าตัวเลขแบบไม่มีเครื่องหมาย	เนื้อที่เก็บข้อมูล
1	FLOAT(M,D)	-3. 402823466E+38 ถึง - 1.175494351E-38	0 และ 1.175494351E-38 ถึง 3.402823466E+38	4 byte
2	DOUBLE(M,D)	-1. 797693134862315 7E +308 ถึง - 2.2250738585072 014E -308	2.2250738585072 014E -308 ถึง 1.79769313486231 57E +308	8 byte

ตารางที่ 2.5 ประเภทข้อมูลชนิดจำนวนทศนิยม (ต่อ)

ลำดับที่	ชื่อประเภทข้อมูล	ค่าตัวเลขแบบมีเครื่องหมาย	ค่าตัวเลขแบบไม่มีเครื่องหมาย	เนื้อที่เก็บข้อมูล
3	DECIMAL (M, D) หรือ NUMERIC(M,D)	เก็บค่าเลขทศนิยมแบบ ระบุจำนวนหลัก M ทุก หลักรวมจุดทศนิยม และ D หลักหลัง ทศนิยม เช่น 123.34 ให้กำหนด เป็น DECIMAL(3,2)	เก็บค่าเลขทศนิยมแบบ ระบุจำนวนหลัก M ทุกหลักรวมจุด ทศนิยม และ D หลัก หลังทศนิยม เช่น 123.34 ให้ กำหนด เป็น DECIMAL(3,2)	ถ้า $d = 0$ ขนาดที่เก็บ คือ $m+1$ byte ถ้า $d > 0$ ขนาดที่เก็บ คือ $m+2$ byte

ตารางที่ 2.6 ประเภทข้อมูลชนิดวันและเวลา

ลำดับที่	ชื่อประเภทข้อมูล	รายละเอียด	เนื้อที่เก็บข้อมูล
1	DATE	สำหรับเก็บข้อมูลประเภทวันที่ โดยเก็บได้จาก 1 มกราคม ค.ศ. 1000 ถึง 31 ธันวาคม ค.ศ. 9999 โดยจะแสดงผลในรูปแบบ YYYY-MMDD	3 byte
2	DATETIME	สำหรับเก็บข้อมูลประเภทวันที่ และเวลา โดยจะ เก็บได้ตั้งแต่ 1 มกราคม ค.ศ. 1000 เวลา 00:00:00 ไปจนถึง 31 ธันวาคม ค.ศ. 9999 เวลา 23:59:59 โดยรูปแบบการแสดงผลจะ เป็น YYYY-MM-DD HH:MM:SS	8 byte

ตารางที่ 2.6 ประเภทข้อมูลชนิดวันและเวลา (ต่อ)

ลำดับที่	ชื่อ ประเภท ข้อมูล	รายละเอียด	เนื้อที่เก็บข้อมูล
3	TIMESTAMP(M)	สำหรับเก็บข้อมูลประเภทวันที่ และเวลา เช่นกัน แต่จะเก็บในรูปแบบของ YYYYMMDDHHMMSS หรือ YMMDDHHMMSS หรือ YYYYMMDD หรือ YYMMDD แล้วแต่ว่าจะระบุค่า M เป็น 14, 12, 8 หรือ 6 ตามลำดับ สามารถเก็บได้ ตั้งแต่วันที่ 1 มกราคม ค.ศ. 1000 ไป จนถึง ประมาณปี ค.ศ. 2037	8 byte
4	TIME	สำหรับเก็บข้อมูลประเภทเวลา มีค่าได้ ตั้งแต่ - 838:59:59 ไป จนถึง 838:59:59 โดยจะ แสดงผล ออกมาใน รูปแบบ HH:MM:SS	3 byte
5	YEAR(2/4)	สำหรับเก็บข้อมูลประเภทปี ในรูปแบบ YYYY หรือ YY แล้วแต่ว่าจะเลือก 2 หรือ 4 (หากไม่ ระบุ จะถือว่าเป็น 4 หลัก)	1 byte

2.4 วรรณกรรมที่เกี่ยวข้อง

2.4.1 การประยุกต์แนวคิดของการออกแบบเว็บไซต์ที่รองรับการใช้งานบนทุกขนาดของ หน้าจออุปกรณ์สำหรับ ระบบติดตามผลการเรียนของนักเรียน

เกียรติศักดิ์ มุขสิกรณ์, (2559) จากการศึกษาพฤติกรรมการเข้าถึงข้อมูลอินเทอร์เน็ตของ ผู้ใช้งาน พบว่าผู้ใช้งานมีความต้องการเข้าถึงข้อมูลอินเทอร์เน็ตอยู่ตลอดเวลา และเข้าถึงได้จากทุก ที่ ตามปรากฏการณ์การใช้งานแบบ

(Bring Your Own Device : BYOD) คือ การที่ผู้ใช้งานนำอุปกรณ์พกพาของตัวเองมาใช้ในที่ทำงาน เพื่อการเข้าถึงข้อมูล โดยแนวทางในการพัฒนาระบบสารสนเทศเพื่อให้ตอบสนองความต้องการของผู้ใช้ ที่สามารถใช้งานผ่านอุปกรณ์โมบาย แท็บเล็ตคอมพิวเตอร์โน้ตบุ๊ก และคอมพิวเตอร์ส่วนบุคคลได้ คือการพัฒนาสารสนเทศให้อยู่ในรูปแบบของเว็บไซต์ และจากการศึกษาของงานวิจัยของ PENG และ ZHOU (2015) พบว่าปัญหาสำคัญในการออกแบบเว็บไซต์ คือการทำให้เว็บไซต์สามารถรองรับประเภทของอุปกรณ์ที่หลากหลาย และทำให้ผู้ใช้งานได้ประสบการณ์ที่ดีจากการใช้งาน ดังนั้นการออกแบบเว็บไซต์จึงได้มีการพัฒนาตามแนวคิด “Responsive Web Design : RWD” เพื่อให้เว็บไซต์สามารถเรียนรู้ถึงการปรับตัวตามขนาดของ 2 อุปกรณ์ โดยการใช้เทคนิคการพัฒนาเว็บไซต์ 3 รูปแบบ คือ (1) การตั้งค่าการแสดงผลให้มีความยืดหยุ่นกับทุกขนาดของหน้าจอ (Fluid grid) (2) การกำหนดขนาดของรูปภาพ (Liquid Image) และ(3) การกำหนดรูปแบบของการแสดงผล (Media Queries) และจากการศึกษาการพัฒนาเว็บไซต์ตามแนวคิด “Responsive Web Design : RWD” พบว่าเป็นการพัฒนาเว็บไซต์แบบใช้โดเมนเดียวและโค้ดชุดเดียว ให้สามารถแสดงผลผ่านอุปกรณ์ได้ทั้งหมด ทั้งสมาร์ตโฟน แท็บเล็ต และคอมพิวเตอร์ส่วนบุคคล และวิธีการออกแบบตามแนวคิด RWD ที่เหมาะสมกับการใช้งานผ่านสมาร์ตโฟน และแท็บเล็ต คือ การออกแบบเว็บไซต์สำหรับอุปกรณ์โมบายเป็นลำดับแรก (mobile first design)

จากการพิจารณาวรรณกรรมและบทความข้างต้น ผู้ศึกษาได้ทำการวางแผนที่จะใช้ประโยชน์จาก RDW (Responsive Web Design) เพื่อมาพัฒนาระบบ Web Application ให้มีความเหมาะสมกับขนาดหน้าจอการแสดงผลของรูปแบบที่ต่างกันของผู้ใช้งานไม่ว่าจะเป็นหน้าจอคอมพิวเตอร์หรือหน้าจออุปกรณ์พกพา

2.4.2 การพัฒนาระบบลงทะเบียนเรียนออนไลน์

ศิริวัฒน์ ภาภิรมย์, (2559) เทคโนโลยีสารสนเทศเปรียบเสมือนเส้นประสาทของธุรกิจและการประยุกต์เทคโนโลยีสารสนเทศในองค์กรจะส่งผลกระทบต่อการทำงานและบุคลากรมากกว่าการเพิ่มประสิทธิภาพหรือการลดขั้นตอนในการทำงาน ระบบเดิมของหน่วยงานก่อนที่จะนำระบบสารสนเทศเข้ามาใช้ การทำงานต่างๆ ล้วนทำด้วยมือ ซึ่งต้องใช้กำลังคนและเวลาในการทำงานค่อนข้างมาก โอกาสเกิดความผิดพลาดในการทำงานมาก ข้อมูลสารสนเทศที่หน่วยงานมี

อยู่ต้องมีคุณภาพถูกต้องและเป็นปัจจุบันอยู่เสมอเพื่อให้ผู้บริหารนำข้อมูลและสารสนเทศเหล่านั้นไปประกอบการตัดสินใจได้ทันต่อเวลาและเหตุการณ์

จากการพิจารณาวรรณกรรมและบทความข้างต้น ผู้ศึกษาได้มองเห็นว่าการลงทะเบียนออนไลน์ในปัจจุบัน ซึ่งเปิดหลักสูตรการสอนในระดับต่างๆ และมีผู้คนจำนวนมาก มีการใช้บุคลากรในการปฏิบัติงานค่อนข้างมาก จึงมีความล่าช้าในงานลงทะเบียนเรียน และเกิดความผิดพลาดค่อนข้างมากจึงมีความสนใจที่จะพัฒนาระบบลงทะเบียนเรียนออนไลน์ มาใช้ให้เป็นระบบอัตโนมัติ โดยทำงานในรูปแบบของโปรแกรมประยุกต์บนเว็บเพื่อเพิ่มความสะดวกรวดเร็ว และความถูกต้องในการลงทะเบียนเรียนและลดภาระงานให้กับบุคลากรของสนามยิงธนูต่อไป

2.4.3 การพัฒนาระบบประเมินความพึงพอใจของผู้เรียนที่ต่อคุณภาพจัดการเรียนการสอนและสิ่งสนับสนุนการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ต

สายสุนีย์ เจริญสุข และอารียา ศรีประเสริฐ (2559) ได้วิจัยเรื่อง การพัฒนาระบบประเมินความพึงพอใจของผู้เรียนที่ต่อคุณภาพจัดการเรียนการสอนและสิ่งสนับสนุนการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ตโดยมีวัตถุประสงค์เพื่อพัฒนาระบบประเมินความพึงพอใจของผู้เรียนที่มีต่อคุณภาพการจัดการเรียนการสอน และสิ่งสนับสนุนการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ตโดยมีกลุ่มตัวอย่างได้มาแบบเจาะจง เป็นผู้บริหารมหาวิทยาลัยจำนวน 12 ท่าน อาจารย์ประจำจำนวน 27 ท่าน และนักศึกษาจำนวน 515 คน เลือกกลุ่มตัวอย่างที่ได้มาแบบเจาะจงตามภาระหน้าที่เกี่ยวข้องกับระบบและเลือกจากการตอบรับที่เข้าร่วมการทดลองใช้ระบบ การวิจัยครั้งนี้ใช้ระยะเวลาในการพัฒนาและทดลองใช้ระบบรวมทั้ง 12 เดือน เครื่องมือและซอฟต์แวร์ที่ใช้ในการพัฒนาประกอบด้วย PHP และระบบการจัดการฐานข้อมูล MySQL รวมทั้ง แบบประเมินผลในงนระบบ โดยระบบที่พัฒนาขึ้นสามารถประเมินผลการประเมินจากนักศึกษา ประมวลผลและแสดงผลการประเมินสำหรับอาจารย์ผู้สอน และผู้บริหารระดับต่างๆ ตามสิทธิ์ในการเข้าถึงรายงานต่างๆ ระบบแบ่งการทำงานเป็นสี่คือ ส่วนของผู้ดูแลระบบเป็นส่วนของการจัดการระบบส่วนของนักศึกษาสำหรับประเมินการเรียนการสอน และประเมินสิ่งสนับสนุนการเรียนรู้โดยระบบทำหน้าที่รวบรวมข้อมูลการประเมินของนักศึกษา ส่วนของอาจารย์และส่วนของผู้บริหารเป็นส่วนแสดงรายงานผลการประเมินผลทางเว็บไซต์ จำแนกตามสิทธิ์การเข้าใช้ระบบได้แก่ คณบดี ผู้อำนวยการศูนย์บริหาร และพัฒนาทรัพยากรบุคคล รองอธิการบดีฝ่ายวิชาการ และ อธิการบดีผลวิจัย การพัฒนาระบบ

ประเมินความพึงพอใจของผู้เรียนที่มีต่อคุณภาพการจัดการเรียนการสอนและสิ่งสนับสนุนการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ตโดยกลุ่มตัวอย่างประกอบด้วยผู้บริหารอาจารย์ และนักศึกษา ทดลองใช้งานระบบประเมินความพึงพอใจของผู้เรียนที่มีต่อคุณภาพการจัดการเรียนการสอนและสิ่งสนับสนุนการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ต พบว่า กลุ่มตัวอย่างที่เข้าถึงการใช้งานระบบทุกคนมีความพึงพอใจต่อการใช้ระบบที่พัฒนาอยู่ในระดับมาก

จากการพิจารณารวบรวมและบทความข้างต้น ผู้ศึกษาได้มองเห็นระบบการประเมินความพึงพอใจในการใช้งานของผู้ใช้ระบบ พบว่าผลลัพธ์จากการเก็บสถิติความพึงพอใจในของผู้เรียนที่มีต่อคุณภาพจัดการเรียนการสอนและสิ่งสนับสนุนการเรียนรู้ผ่านเครือข่ายอินเทอร์เน็ตเป็นผลลัพธ์อย่างหนึ่งในการนำไปพัฒนาระบบสถิติต่างๆ รวมทั้งระบบความพึงพอใจในการบริการ และสรุปผลลัพธ์ออกมาให้ทางกิจการได้เอาไปพัฒนาธุรกิจสนามยิงธนูต่อไปอีกด้วย

2.4.4 ปัจจัยที่มีผลต่อการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬาในการบริหารจัดการงานลานกีฬาด้านยาเสพติดของกรุงเทพมหานคร

วิจักรณ อัครเลิศพานิช (2542) ศึกษาเรื่อง ปัจจัยที่มีผลต่อการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬาในการบริหารจัดการงานลานกีฬาด้านยาเสพติดของกรุงเทพมหานคร กรณีศึกษา: โครงการพัฒนาระบบการจัดการและศักยภาพของลานกีฬาด้านยาเสพติดในกรุงเทพมหานคร มีวัตถุประสงค์เพื่อศึกษาปัจจัยที่มีผลต่อการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬาในการบริหารจัดการงานกีฬาของกรุงเทพมหานคร วิธีการศึกษาเป็นการศึกษาเชิงสำรวจ (survey research)

ปัจจัยที่มีผลต่อการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬา ผลการศึกษา พบว่า ปัจจัยจูงใจประกอบด้วย การทำประโยชน์เพื่อสังคม ความสมัครใจ การทำงานเป็นทีม กลุ่มและเพื่อนใหม่ ความเชื่อถือในตัวผู้นำ การได้รับการชักจูง ความภาคภูมิใจ มีผลต่อการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬา ปัจจัยแวดล้อมประกอบด้วย ระยะเวลาในการเดินทาง ความไม่พอใจต่อสภาพแวดล้อม ความพร้อมของสถานที่และอุปกรณ์ ผู้นำกิจกรรมที่ดี และการจัดการแข่งขันกีฬาระดับสากลมีผลต่อการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬา ส่วนระดับของลักษณะการเข้ามามีส่วนร่วม ผลการศึกษาพบว่า ระดับของการเข้ามามีส่วนร่วมทั้ง 5 ลักษณะ คือ การกำหนดนโยบายและวางแผน การจัดการองค์การ การดำเนินงาน การประสานงาน และการ

ประเมินผลอยู่ในระดับปานกลาง ผลการทดสอบสมมุติฐานพบว่า ความเชื่อถือในตัวผู้นำ เกียรติ และชื่อเสียง ไม่มีความสัมพันธ์กับลักษณะการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬา ส่วนความพร้อมของสถานที่และวัสดุอุปกรณ์ และการจัดการแข่งขันกีฬาระดับสากลมีความสัมพันธ์กับลักษณะการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬา อย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05

ข้อเสนอแนะของการศึกษา จำแนกออกเป็น 3 ด้าน ดังนี้

1). ด้านนโยบาย ควรกำหนดนโยบายที่ชัดเจน และชี้แจงผู้เกี่ยวข้องทุกฝ่ายเพื่อให้เกิดความเข้าใจที่ตรงกัน มีการกำหนดผลตอบแทนการเป็นกรรมการลานกีฬาให้ชัดเจน ควรมีการเลือกตั้งคณะกรรมการลานกีฬา จัดตั้งหน่วยงานเพื่อดูแลลานกีฬาโดยเฉพาะ และกำหนดนโยบายกิจกรรมให้ชัดเจน โดยเน้นเรื่องการมีส่วนร่วมของคณะกรรมการลานกีฬาในกิจกรรมที่เข้าร่วม

2). ด้านบริหารการจัดการ ควรหาผู้สนับสนุนทั้งในด้านวัสดุอุปกรณ์ การปรับปรุงและขยายพื้นที่ลานกีฬา และจัดหาบุคลากรให้เพียงพอ

3). ด้านการดำเนินงาน ควรเสริมกิจกรรมพิเศษที่เน้นการมีส่วนร่วมของสมาชิกและพัฒนากิจกรรมทั้ง 5 ลักษณะให้คณะกรรมการลานกีฬาเข้ามามีส่วนร่วมมากขึ้น ควรมีส่วนร่วมป้องกันปัญหาเสพติดในลานกีฬา ควรส่งเสริมให้ลานกีฬาเป็นสถานที่ที่สามารถผลิตนักกีฬาระดับชาติได้ และควรประชาสัมพันธ์โครงการลานกีฬาผ่านสื่อต่างๆ ให้มากยิ่งขึ้น

จากการพิจารณาวรรณกรรมและบทความข้างต้น ผู้ศึกษาได้ทราบถึงการเข้ามามีส่วนร่วมของคณะกรรมการลานกีฬา ได้อธิบายปัญหาและนโยบายต่างๆ ได้ตระหนักถึงการบริหารจัดการด้านการดำเนินงาน ให้ครอบคลุมในการบริการสถานที่ โดยผู้ศึกษาจะนำเอามาประยุกต์พัฒนาระบบบริหารภายในกิจการสนามยิงธนูอย่างครอบคลุมทุกระบบ มีความชัดเจนในระบบ ใช้งานง่ายมากยิ่งขึ้น และเป็นประโยชน์สูงสุดแก่ผู้ใช้งาน

2.4.5 ปัจจัยที่มีผลต่อพฤติกรรมกรรมการจังหวังพักผ่านระบบออนไลน์ กรณีศึกษาเว็บไซต์อโกต้า

นางสาวนิตยา มหาชานิกะ (2557) ได้ศึกษาเรื่อง “ปัจจัยที่มีผลต่อพฤติกรรมกรรมการจังหวังพักผ่านระบบออนไลน์ กรณีศึกษา เว็บไซต์อโกต้า” ผลการวิจัยพบว่า ลักษณะประชากรที่แตกต่างกันส่งผลให้เกิดพฤติกรรมกรรมการจังหวังโรงแรมออนไลน์ไม่แตกต่างกัน แต่เมื่อนำมาหาความสัมพันธ์แล้วนั้นพบว่าลักษณะประชากรนั้นไม่มีความสัมพันธ์กับพฤติกรรมกรรมการจังหวังโรงแรม

ออนไลน์ในด้านเพศ อายุการศึกษาและอาชีพ ยกเว้นรายได้ที่มีความสัมพันธ์กับพฤติกรรมการจองห้องพักออนไลน์ในด้านการซื้อซ้ำ ส่วนด้านปัจจัยทางเทคโนโลยี โครงสร้างเว็บไซต์ สรุปว่ามีความสัมพันธ์กับพฤติกรรมการจองโรงแรมออนไลน์ทุกๆ องค์ประกอบ ได้แก่ เรื่องโครงสร้างข้อมูล เนื้อหา กิจกรรมกลุ่ม ผู้เยี่ยมชมเว็บ การสื่อสาร การเชื่อมต่อข้อมูลและระบบการขาย ในขนาดที่ ปัจจัยกระตุ้นทางการตลาด พบว่าพฤติกรรมการจองโรงแรมออนไลน์ ทั้งในเรื่องราคาช่องทางการจัดจำหน่าย และการส่งเสริมการขายก็เช่นกัน โดยสิ่งที่ส่งผลมากที่สุดคือเรื่องราคาและมีความสัมพันธ์แบบตรงข้ามยิ่งราคาสูงการซื้อซ้ำ และบอกต่อก็จะน้อยลง

จากการพิจารณาวรรณกรรมและบทความข้างต้น ผู้ศึกษาได้ทราบถึงพฤติกรรมการจองผ่านระบบออนไลน์ ซึ่งมีคนสนใจในการจองผ่านระบบออนไลน์มากขึ้น ผู้ศึกษาจึงจำเป็นต้องคิดพัฒนาและออกแบบระบบให้มีความกระชับสามารถใช้งานได้ง่ายไม่ซับซ้อนเกินไป เพื่อตอบสนองความต้องการของผู้ที่สนใจให้เร็วที่สุดและยังช่วยลดขั้นตอนการทำงานให้กับเจ้าหน้าที่พนักงานได้อีกด้วย