

บทที่ 2

ทฤษฎีและระบบงานที่เกี่ยวข้อง

จากการศึกษาเรื่อง ระบบบริหารจัดการฐานข้อมูลแมลงศัตรูพืช ผู้จัดทำโครงการวิจัยได้ศึกษาทฤษฎีที่เกี่ยวข้องและระบบงานที่เกี่ยวข้อง ดังนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 แมลงศัตรูพืช

แมลงศัตรูพืชหมายถึงสัตว์ที่มีลำตัวเป็นปล้อง (arthropods) จัดอยู่ในชั้น (class) Insecta ประกอบด้วยสัตว์ประมาณ 26 อันดับ (order) ซึ่งสัตว์เหล่านี้ได้ก่อความเสียหายแก่พืชเพาะปลูกแมลงเป็นสัตว์ไม่มีกระดูกสันหลัง ลำตัวแบ่งออกเป็นสามส่วน คือ ส่วนศีรษะ (head) อก (thorax) และท้อง (abdomen) ซึ่งบนส่วนอกมี 3 ปล้อง ซึ่งแต่ละปล้องมีขา 1 คู่ ส่วนท้องมี 8-11 ปล้อง แมลงมีผนังหุ้มลำตัวแข็ง (exoskeleton) ดังนั้นการเจริญเติบโตของแมลงจึงต้องอาศัยการลอกคราบ (molting) การจำแนกชนิดของแมลงที่ถูกต้องจะแบ่งตามหลักการอนุกรมวิธานโดยนักกีฏวิทยา (entomologist) แต่ในที่นี้จะขอแบ่งชนิดของแมลงศัตรูพืชออกตามลักษณะของการทำลายดังนี้แมลงจำพวกกัดกินใบ (leaf feeder) ได้แก่ หนอนผีเสื้อ ตั๊กแตน ตัวงักแข็ง , แมลงจำพวกดูดกินน้ำเลี้ยง (juice sucker) ได้แก่ เพลี้ยอ่อน เพลี้ยกระโดดเพลี้ยจักจั่น และมวนต่างๆ , แมลงจำพวกหนอนซอนใบ (leaf minor) ได้แก่ หนอนผีเสื้อ หนอนแมลงวันบางชนิด , แมลงจำพวกหนอนเจาะลำต้น (stem borer) ได้แก่ หนอนด้วง หนอนผีเสื้อ, แมลงจำพวกกัดกินราก (root feeder) ได้แก่ ตัวงัดดี จิ้งหรีด แมลงกระซอน ตัวงัดดิน ตัวงัดงวง , แมลงจำพวกที่ทำให้เกิดปุ่มปม (gall maker) ได้แก่ ต่อ แตน และเพลี้ย

ศัตรูพืชแต่ละชนิดสร้างความเสียหายให้แก่พืชที่ปลูกมากน้อยแตกต่างกันแล้วแต่ชนิดของศัตรูพืชและชนิดของพืชที่ปลูกปริมาณระดับความรุนแรงของการทำลายแสดงในตารางที่ 1 ซึ่งแสดงปริมาณของความเสียหายของพืช 6 ชนิด ที่ถูกทำลายโดยศัตรูพืช 3 ชนิด คือ โรคพืช แมลงศัตรูพืช และวัชพืช โดยสรุปแล้วประมาณ 30 เปอร์เซ็นต์ของผลผลิตพืชถูกทำลายโดยศัตรูพืชเหล่านี้

ตารางที่ 2.1 ประมาณความเสียหายของพืช

ปลูก	พืช	ผลผลิตที่คาดว่าจะได้รับ	ความเสียหายเนื่องจาก			ผลผลิตที่ได้รับจริง
			โรคพืช	แมลงศัตรู	วัชพืช	
	ข้าวสาลี	351.1	33.3 (39)	17.8 (21)	34.5 (40)	265.5
	ข้าว	458.8	39.4 (19)	120.7 (58)	46.7 (23)	252.0
	ข้าวโพด	339.5	32.7 (27)	44.0 (36)	44.3 (37)	218.5
	มันฝรั่ง	400.0	88.9 (69)	23.8 (18)	17.5 (13)	270.8
	ผัก	279.9	31.1 (40)	23.4 (30)	23.7 (30)	201.7
	ผลไม้	197.0	32.6 (59)	11.3 (20)	11.4 (21)	141.7
	รวม		258.0 (38)	241.0 (36)	178.1 (26)	

2.1.2 Mobile Application

Mobile Application ประกอบขึ้นด้วยคำ สองคำ คือ Mobile กับ Application มีความหมายดังนี้ Mobile คืออุปกรณ์สื่อสารที่ใช้ในการพกพาซึ่งนอกจากจะใช้งานได้ตามพื้นฐานของโทรศัพท์แล้ว ยังทำงานได้เหมือนกับเครื่องคอมพิวเตอร์เนื่องจากเป็น อุปกรณ์ที่พกพาได้จึงมีคุณสมบัติเด่น คือขนาดเล็กน้ำหนักเบาใช้พลังงานค่อนข้างน้อย ปัจจุบันนำมาใช้ทำหน้าที่ได้หลายอย่างในการติดต่อแลกเปลี่ยนข่าวสารบนคอมพิวเตอร์สำหรับ Application หมายถึง ซอฟต์แวร์ที่ใช้เพื่อช่วยการทำงานของผู้ใช้ (User) โดย Application จะต้องมีสิ่งที่เรียกว่า ส่วนติดต่อกับ ผู้ใช้ (User Interface หรือ UI) เพื่อเป็นตัวกลางการใช้งานต่าง ๆ Mobile Application เป็นการพัฒนาโปรแกรมประยุกต์สำหรับอุปกรณ์เคลื่อนที่ เช่น โทรศัพท์มือถือแท็บเล็ตโดยโปรแกรมจะช่วยตอบสนองความต้องการของผู้บริโภคอีกทั้งยังสนับสนุนให้ผู้ใช้โทรศัพท์ได้ใช้ง่ายยิ่งขึ้นในปัจจุบันโทรศัพท์มือถือ หรือ สมาร์ทโฟน มีหลายระบบปฏิบัติการที่พัฒนาออกมาให้ผู้บริโภคใช้ส่วนที่มีคนใช้และเป็นที่ยอมรับมากก็คือ ios และ Android

จึงทำให้เกิดการเขียนหรือพัฒนา 5 Application ลงบนสมาร์ตโฟนเป็นอย่างมากอย่างเช่น แอปที่ เกมส์ โปรแกรมคุยต่างๆ และหลายธุรกิจก็เข้าไปเน้นในการพัฒนา Mobile Application เพื่อเพิ่มช่องทางการสื่อสารกับลูกค้ามากขึ้น Mobile Application เหมาะสำหรับธุรกิจและองค์กรต่างๆ ในการเข้าถึงกลุ่มคนรวมถึงขยายการให้บริการผ่านมือถือสะดวกง่ายทุกที่ทุกเวลา

2.1.3 Flutter

Flutter เป็น Framework ที่สามารถพัฒนา native app แบบ cross-platform ได้ การรวบรวมข้อมูล Native ให้ตัวประมวลผล ARM นั้นช่วยให้การพัฒนาง่ายขึ้นมีการ render ที่ง่ายขึ้นพวกชุดเครื่องมือ, widgets, และ features ก็มีส่วนช่วยให้ process นั้นง่ายขึ้นนอกจากการเป็น Framework แบบ open source และยังเป็นแบบ open source Developer Flutter สามารถจัดการไฟล์ที่ไม่เป็นระเบียบได้ในโหมด Development การรวบรวม Jit ช่วยให้การพัฒนาและการแก้ bug นั้นเร็วขึ้นนอกจากนั้น Framework ตัวนี้ยังรองรับ plugins หลากๆ อย่างเช่น Firebase หรือ SQLite Pub.dev จะช่วยหา plugins ที่คุณต้องการ Flutter ยังมีความคล้ายคลึงกับองค์ประกอบของภาษาในการเขียนโปรแกรมมาตรฐานอีกด้วย

ข้อดีของ flutter HotReload Feature นั้นนอกจากจะช่วยให้ Developer ทำงานเร็วขึ้นถ้าต้องการจะเปลี่ยนแปลงบางอย่างใน Data ในการสร้าง app interface สามารถเปลี่ยนแปลงภายใน code ขณะที่ Software กำลังดำเนินการอยู่การเปลี่ยนแปลงทุกอย่างก็จะปรากฏให้เห็นในแอปเลย การพัฒนาที่รวดเร็วนอกจากจากปุ่ม Hot reload แล้วยังมี widgets, ตัวควบคุมและ libraries จำนวนมากที่คุณสามารถปรับแต่งได้ตามต้องการและกระบวนการการพัฒนา UI ยังมีความรวดเร็วมากๆ Flutter ทำให้เราสามารถสร้าง UI ที่ใช้ได้ทั้งใน iOS และ Android ด้วยความช่วยเหลือของหลายๆ เครื่องมืออย่างเช่น IntelliJ, Android Studio และ Visual Studio Code Open source และฟรี

ข้อเสียของ flutter ไม่ค่อยมีคนนิยมใช้ Dart ไม่เหมือนกับพวก Java หรือ Kotlin ซึ่งเป็นภาษาส่วนใหญ่ที่คนนิยมใช้พัฒนาแอปกัน Dart ยังไม่มีความนิยมเท่าไรนั่นคือเหตุผลที่ Developers ส่วนใหญ่ภาษาใหม่ๆ เพราะมีคนที่ใช้ Java เป็นประจำอยู่แล้วและ UI Framework ที่มีอยู่หลากหลายก็เพียงพอ อย่างไรก็ตาม Dart นั้นมีข้อดีมากกว่า Java ในการพัฒนาแอปแต่ก็ยังน้อยกว่า Kotlin ในขณะเดียวกันฟังก์ชัน ของ Flutter เช่น Hot Reload จะไม่ทำงานถ้าหากไม่มี Dart และไม่รองรับอุปกรณ์ทั้งหมดมี libraries ที่จำกัด เนื่องจาก Flutter เป็น Framework ที่ยังใหม่อยู่จึงยังไม่ค่อยมี libraries ที่เป็นประโยชน์อย่างเช่นภาษาอื่น ๆ นั้นจึงเป็นเหตุผลว่าทำไม Developers จำเป็นต้องสร้าง libraries ส่วนใหญ่ด้วยตัวเอง มันก็เลยต้องเสียเวลาไปมากกว่าเดิม

2.1.4 MySQL

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูลที่พัฒนาโดยบริษัทมายเอสคิวแอลเอบี (MySQL AB) มีหน้าที่เก็บข้อมูลอย่างเป็นระบบรองรับคำสั่งเอสคิวแอล (SQL = Structured Query Language) เป็นเครื่องมือสำหรับเก็บข้อมูลที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการเพื่อให้ได้ระบบงานที่รองรับความต้องการของผู้ใช้เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษาพีเอชพี ภาษาเอเอสพีดอทเน็ต หรือภาษาเจเอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษาวิซวลเบสิกดอทเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนทเซอร์ที่ถูกนำไปใช้งานมากที่สุด MySQL เป็นที่นิยมใช้กันมากสำหรับฐานข้อมูลสำหรับเว็บไซต์เช่น วิกิพีเดียและ phpBB และนิยมใช้งานร่วมกับภาษาโปรแกรม PHP ซึ่งมักจะได้ชื่อว่าเป็นคู่จะเห็นได้จากคู่มือคอมพิวเตอร์ต่างๆ ที่จะสอนการใช้งาน MySQL และ PHP ควบคู่กันไป นอกจากนี้หลายภาษาโปรแกรมที่สามารถทำงานร่วมกับฐานข้อมูล MySQL ซึ่งรวมถึง ภาษาซี ซีพลัสพลัส ปาสคาล ซีชาร์ป ภาษาจาวา ภาษาเพิร์ล พีเอชพีไพทอน รูบีและภาษาอื่น ใช้งานผ่าน API สำหรับโปรแกรมที่ติดต่อผ่าน ODBC หรือส่วนเชื่อมต่อกับภาษาอื่น เช่น เอเอสพีสามารถเรียกใช้MySQL ผ่านทาง MyODBC, ADO, ADO.NET เป็นต้น

ข้อดีของ MySQL เพิ่ม performance ให้กับแอปพลิเคชัน เมื่อถูกสร้างขึ้นจะถูกคอมไพล์และเก็บไว้ใน database แต่ mysql นั้นเขียน stored procedure ในรูปแบบที่แตกต่างไปเล็กน้อย คือ มันจะถูก compile on demand เท่านั้น และหลังจาก compile เสร็จจะเก็บไว้ใน cache ซึ่งจะเก็บไว้ใช้ใน 1 connection ซึ่งหมายความว่ามันจะมีประโยชน์ และเพิ่ม speed ได้ถ้ามีการเรียกใช้หลายๆ ครั้ง ใน 1 connection หากไม่แล้วก็จะทำงานเหมือนกับ query ธรรมดา(ควรพิจารณาดูว่าจะเขียนหรือไม่ โดยใช้หลักการข้อนี้, ช่วย ลด traffic ระหว่าง application และ database server, ช่วยในการใช้ซ้ำ และ ซ่อนเร้น

การทำงานในกรณีที่น่าไปใช้กับ application อื่น เนื่องจากแสดงเป็นรูปแบบ interface ของ database เท่านั้นซึ่งเป็นการลดงานของ developer, ความปลอดภัยของ store procedure จำกัดโดย db admin ซึ่งสามารถกำหนด permission ต่างๆ ในการเรียกใช้ stored procedure ได้

ข้อเสียของ MySQL ถ้าใช้ stored procedure เป็นจำนวนมากมันก็จะกินทรัพยากรมากตามไปด้วยนอกจากนี้ถ้ามีการเรียกใช้ logical operation จำนวนมากๆ ใน store procedure(ข้อนี้ต้องระวังให้ดี) การใช้งาน CPU ก็จะมีเพิ่มขึ้นอีกเนื่องจาก database server นั้นไม่ได้ถูกออกแบบมาให้ทำงานกับ logical operation, โครงสร้างของ store procedure นั้นยากต่อการพัฒนาในกรณีการทำงานในรูปแบบซับซ้อนมากๆ, MySQL ไม่มีเครื่องมือสำหรับ debug store procedure, ยากที่จะ

พัฒนาและดูแล stored procedure ซึ่งจะต้องใช้ skill ที่สูงพอสมควรและนักพัฒนาส่วนใหญ่ไม่ค่อยใส่ใจเรื่องนี้ซึ่งจะนำไปสู่ปัญหาทั้งด้านการพัฒนาและดูแลระบบ

2.1.5 Apache

Apache ได้รับความนิยมขึ้นเรื่อย ๆ จนปัจจุบันได้รับความนิยมเป็นอันดับหนึ่งมีผู้ใช้งานอยู่ประมาณ 65% ของเว็บเซิร์ฟเวอร์ความสามารถของ Apache การที่อาปาเซ่เป็นซอฟต์แวร์ที่อยู่ในลักษณะของโอเพ่นซอร์สที่เปิดให้บุคคลทั่วไปสามารถเข้ามาร่วมพัฒนาส่วนต่างๆ ของอาปาเซ่ได้ซึ่งทำให้เกิดเป็นโมดูลที่เกิดประโยชน์มากมายเช่น mod_perl, mod_python หรือ mod_php ซึ่งเป็นโมดูลที่ทำให้อาปาเซ่สามารถใช้ประโยชน์และทำงานร่วมกับภาษาอื่นได้แทนที่จะเป็นเพียงเซิร์ฟเวอร์ที่ให้บริการเพียงแค่ HTML อย่างเดียวนอกจากนี้อาปาเซ่เองยังมีความสามารถอื่นๆ ด้วยเช่น การยืนยันตัวบุคคล(mod_auth,mod_access, mod_digest) หรือเพิ่มความปลอดภัยในการสื่อสารผ่านโปรโตคอล https(mod_ssl) นอกจากนี้ก็ยังมีโมดูลอื่นๆ ที่ได้รับความนิยมใช้เช่น mod_vhost ทำให้สามารถสร้างโฮสต์เสมือน www.sample.com, wiki.sample.com, mail.sample.com หรือ www.ilovewiki.org ภายในเครื่องเดียวกันได้หรือ mod_rewrite เป็นเครื่องมือที่จะช่วยให้ url ของเว็บนั้นอ่านง่ายขึ้น

ข้อดีของ เว็บเซิร์ฟเวอร์ Apache โปรแกรมโอเพ่นซอร์สทุกโปรแกรมเผยแพร่ภายใต้เงื่อนไขสิทธิ์การใช้งานที่ทำให้ซอร์สโค้ดใช้งานได้อย่างอิสระ สามารถดาวน์โหลดโมดูลจากเซิร์ฟเวอร์ Apache (และเรียกใช้บน Windows) ได้โดยไม่ต้องเสียค่าธรรมเนียมการออกใบอนุญาต

ข้อเสียของ เว็บเซิร์ฟเวอร์ Apache ด้วยความสามารถของ Apache ในการให้ผู้ใช้เลือกและปิดใช้งานบริการที่ผู้ใช้ต้องการอาจทำให้เกิดภัยคุกคามร้ายแรงหากไม่ได้รับการป้องกัน

2.1.6 Visual Studio Code

Visual Studio Code หรือ VSCode เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ดจากค่ายไมโครซอฟท์มีการพัฒนาออกมาในรูปแบบของ OpenSource ซึ่ง Visual Studio Code นั้นเหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์มรองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้นำมาใช้งานได้ง่ายไม่ซับซ้อนมีเครื่องมือส่วนขยายต่างๆ ให้เลือกใช้อย่างมากมายไม่ว่าจะเป็น 1.การเปิดใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++, C#, Java, Python, PHP หรือ Go 2.Themes 3.Debugger 4.Commands เป็นต้น

2.1.7 การออกแบบฐานข้อมูล(Database Design)

หลังจากได้เป้าหมายของงานชัดเจนแล้วว่าในระบบใหม่จะต้องทำอะไรบ้างมีการออกรายงานอะไรและใช้ข้อมูลใดบ้างในระบบจากนั้นก็จะมาเริ่มทำการออกแบบฐานข้อมูลได้แก่การวิเคราะห์หาเอนทิตีหรือรีเลชันการวิเคราะห์หาแอททริบิวต์และคีย์ของเอนทิตีหรือรีเลชันรวมถึงการกำหนดความสัมพันธ์ระหว่างเอนทิตีหรือรีเลชันผังระบบ (Context Diagram) แสดงลักษณะของระบบงาน โดยจะเป็นมุมมองในระดับสูงซึ่งจะไม่แสดงถึงสัญลักษณ์ในการเก็บข้อมูลเพราะจะเป็นการเขียนถึงภายในระบบเชื่อมโยงกับสิ่งที่อยู่นอกระบบรวมทั้งเป็นตัวกำหนดขอบเขตความต้องการของระบบสามารถทำให้เห็นภาพรวมของระบบได้อย่างง่ายดายตายค่าคือผังการไหลของระบบระดับ 1,2,...,N (Data Flow Diagram: DFD Level 1,2,...,N) อธิบายผังการไหลของข้อมูลได้ตั้งนี้การออกแบบฐานข้อมูล(Database Design) หลังจากได้เป้าหมายของงานชัดเจนแล้วว่าในระบบใหม่จะต้องทำอะไรบ้างมีการออกรายงานอะไรและใช้ข้อมูลใดบ้างในระบบจากนั้นก็จะมาเริ่มทำการออกแบบฐานข้อมูลได้แก่การวิเคราะห์หาเอนทิตีหรือรีเลชันการวิเคราะห์หาแอททริบิวต์และคีย์ของเอนทิตีหรือรีเลชันรวมถึงการกำหนดความสัมพันธ์ระหว่างเอนทิตีหรือรีเลชันผังระบบ (Context Diagram) แสดงลักษณะของระบบงานโดยจะเป็นมุมมองในระดับสูงซึ่งจะไม่แสดงถึงสัญลักษณ์ในการเก็บข้อมูลเพราะจะเป็นการเขียนถึงภายในระบบเชื่อมโยงกับสิ่งที่อยู่นอกระบบรวมทั้งเป็นตัวกำหนดขอบเขตความต้องการของระบบสามารถทำให้เห็นภาพรวมของระบบได้อย่างง่ายดายตายค่าคือผังการไหลของระบบระดับ 1,2,...,N (Data Flow Diagram: DFD Level 1,2,...,N) อธิบายผังการไหลของข้อมูลได้ตั้งนี้

2.1.8 แผนภาพกระแสข้อมูล (Data Flow Diagram)

แผนภาพกระแสข้อมูล (Data Flow Diagram: DFD) หรือเรียกอีกอย่างหนึ่งว่า แผนภาพการไหลของข้อมูลเป็นเครื่องมือที่ใช้เพื่อแสดงการไหลของข้อมูลและการประมวลผลต่างๆ ในระบบสัมพันธ์กับแหล่งเก็บข้อมูลที่ใช้โดยแผนภาพนี้จะป็นสื่อที่ช่วยให้การวิเคราะห์เป็นไปได้อย่างง่ายและมีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบเองหรือระหว่างผู้วิเคราะห์ระบบกับโปรแกรมเมอร์หรือระหว่างผู้วิเคราะห์ระบบกับผู้ใช้ระบบ (รัชณี กัลยาวิณัย และ อัจฉรา ธารอุไรกุลม 2542)

ประโยชน์ที่ได้จากการใช้แผนภาพกระแสข้อมูลมีดังนี้


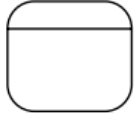






- 1.มีความอิสระในการทำงานโดยไม่ต้องมีเทคนิคอื่นมาช่วยเนื่องจากสามารถใช้สัญลักษณ์ต่างๆ แทนสิ่งที่วิเคราะห์มา
- 2.เป็นสื่อที่ง่ายต่อการแสดงความสัมพันธ์ระหว่างระบบใหญ่และระบบย่อยซึ่งจะทำให้เข้าใจความสัมพันธ์ต่างๆ ได้ดี

3. เป็นสื่อที่ช่วยในการวิเคราะห์ระบบให้เป็นไปได้ง่ายและมีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบเองหรือระหว่างผู้วิเคราะห์ระบบกับโปรแกรมเมอร์หรือระหว่างผู้วิเคราะห์ระบบกับผู้ใช้ระบบ
4. ช่วยในการวิเคราะห์ระบบให้สะดวกโดยสามารถเห็นข้อมูลและขั้นตอนต่างๆเป็นแผนภาพ

2.1.8.1 สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูลนั้น ประกอบด้วย 4 สัญลักษณ์ ที่แสดงถึงการประมวลผลการไหลของข้อมูลส่วนใหญ่ที่ใช้เก็บข้อมูลและสิ่งที่ยอยู่นอกระบบ โดยได้มีการศึกษาคิดค้นพัฒนาวิธีการอยู่หลายแบบแต่ที่เป็นมาตรฐานมี 2 กลุ่ม คือกลุ่มที่คิดค้นโดย Gane and Sarson (1979) และกลุ่มของ DeMarco and Yourdon (SeMarco, 1979) ถึงแม้สัญลักษณ์บางอย่างของสององค์กรนี้จะต่างกันแต่องค์ประกอบของแผนภาพและหลักการเขียนแผนภาพไม่ได้แตกต่างกัน

ตารางที่ 2.2 สัญลักษณ์ต่างๆ ที่ใช้ในการเขียนแผนภาพการไหลของข้อมูล

ชื่อสัญลักษณ์	DeMarco & Yourdon symbols	Gane & Sarson symbols
การประมวลผล (Process)		
แหล่งเก็บข้อมูล (Data Store)		
กระแสข้อมูล (Data Flow)		
สิ่งที่ยูภายนอก (External Entity)		

เนื่องจากสัญลักษณ์การเขียนแผนภาพกระแสข้อมูลประกอบด้วยสัญลักษณ์ต่างๆซึ่งมีความหมายในตัวเองดังนั้นการเขียนแผนภาพกระแสข้อมูลจึงต้องมีกฎเกณฑ์ต่างๆ เพื่อแสดงถึงความถูกต้องในการเขียนแผนภาพ ดังนี้

2.1.8.2 สัญลักษณ์การประมวลผล (Process Symbol) การประมวลผล (Process)

เป็นการเปลี่ยนแปลงข้อมูลจากรูปแบบหนึ่ง (Input) ไปเป็นอีกรูปแบบหนึ่ง (Output) เช่น การคำนวณรายได้สุทธิของลูกค้ารายวัน จะต้องประกอบด้วยข้อมูลนำเข้าที่เป็น “อัตราค่าจ้างต่อชั่วโมง” และ “จำนวนชั่วโมงการทำงาน” เมื่อผ่านการประมวลผลแล้วจะได้ “รายได้สุทธิ”

ตัวอย่างการประมวลผล ได้แก่

- คำนวณค่าคอมมิชชัน
- ตรวจสอบใบสั่งซื้อ
- ลงทะเบียน เป็นต้น

การใช้สัญลักษณ์การประมวลผล

1. ต้องใช้สัญลักษณ์การประมวลผล (Process) คู่กับสัญลักษณ์กระแสข้อมูล (Data Flow) เสมอ โดยที่ถ้าลูกศรชี้เข้าหมายถึงเป็นข้อมูลนำเข้าถ้าลูกศรชี้ออกหมายถึงเป็นข้อมูลออกจากการประมวลผล ซึ่ง 1 Process สามารถมีข้อมูลนำเข้ามากกว่า 1 เส้น หรือข้อมูลออกมากกว่า 1 เส้นได้

2. การตั้งชื่อของ Process ควรเป็นวลีเดียวที่อธิบายการทำงานทั้งหมดได้และควรอธิบายการทำงานอย่างใดอย่างหนึ่งโดยเฉพาะมากกว่าที่จะอธิบายการทำงานอย่างกว้างๆ เช่น หากแสดงถึงการประมวลผล “ตรวจสอบรายการ” ควรจะระบุว่าเป็น “การตรวจสอบรายการถอนเงิน” หรือ “ตรวจสอบรายการค่าใช้จ่ายรายสัปดาห์” เป็นต้น

3. แต่ละ Process จะมีแต่ข้อมูลเข้าอย่างเดียว หรือออกอย่างเดียวไม่ได้

2.1.8.3 สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol)

แหล่งที่เก็บข้อมูล(DataStore)เป็นส่วนที่ใช้แทนชื่อเพิ่มข้อมูลที่เก็บข้อมูลเพราะมีการประมวลผลหลายแบบที่จะต้องมีการเก็บข้อมูลไว้เพื่อที่จะได้นำไปใช้ภายหลังซึ่งแหล่งเก็บข้อมูลจะต้องมีทั้งข้อมูลเข้าและข้อมูลออก โดยข้อมูลที่ออกจากแหล่งเก็บข้อมูลจะอยู่ในลักษณะที่ถูกอ่านขึ้นมา ส่วนข้อมูลที่ไหลเข้าสู่แหล่งเก็บข้อมูลจะอยู่ในรูปของการบันทึกการเพิ่ม-ลบ แก้ว

ตัวอย่างแหล่งเก็บข้อมูล ได้แก่

- เพิ่มคนไข้
- เพิ่มพนักงาน เป็นต้น

การใช้สัญลักษณ์แหล่งเก็บข้อมูล

1. ต้องใช้สัญลักษณ์แหล่งเก็บข้อมูล (Data Store) คู่กับสัญลักษณ์กระแสข้อมูล (Data Flow) เสมอโดยที่ถ้าลูกศรชี้เข้าหมายถึงเป็นข้อมูลนำเข้าไปเก็บยังแหล่งเก็บถ้าลูกศรชี้ออกหมายถึงอ่านข้อมูลจากแหล่งเก็บข้อมูลไปใช้ในการประมวลผล

2. Data Store ต้องเชื่อมต่อการประมวลผล (Process) เสมอโดยเชื่อมผ่านกระแสข้อมูล (Data Flow)

3. เนื่องจาก Data Store ใช้แทนสิ่งที่เก็บข้อมูลเกี่ยวกับคน, สถานที่, หรือสิ่งของดังนั้นควรเขียนชื่อกำกับ ด้วยคำนาม เช่น “เพิ่มข้อมูลสินค้า”, “เพิ่มเวลาทำงานของพนักงาน” เป็นต้น

4. ใช้อักษรย่อ D1, D2 เป็นต้น เขียนด้านซ้ายมือของสัญลักษณ์เพื่อแสดงว่าเป็นแหล่งเก็บข้อมูล อันที่เท่าใด สามารถเขียนซ้ำในระดับต่างๆของแผนภาพกระแสข้อมูลได้

5.DataStoreใช้แทนสิ่งที่เป็นที่เก็บข้อมูลซึ่งอาจเป็นการทำด้วยมือหรือเก็บในรูปแบบคอมพิวเตอร์คือ แฟ้มข้อมูลหรือฐานข้อมูลก็ได้

2.1.8.3 สัญลักษณ์สิ่งที่อยู่ภายนอก (External Entity Symbol)

สิ่งที่อยู่ภายนอก (External Entity) เป็นส่วนที่ใช้แทนคนแผนกภายในองค์กรและแผนกภายนอก องค์กรหรือระบบสารสนเทศอื่นที่เป็นส่วนที่จะให้ข้อมูลหรือรับข้อมูลสิ่งที่อยู่นอกระบบนี้ใช้แสดงถึงขอบเขตของระบบสารสนเทศและแสดงถึงวาระบบที่ศึกษาอยู่นี้จะติดต่อกับสิ่งที่อยู่ภายนอกด้วยวิธีใด (นำข้อมูลเข้ามาหรือได้ข้อมูลออกไป)

ตัวอย่างสิ่งที่อยู่ภายนอกได้แก่

- นักศึกษา
- สินค้าคงคลัง เป็นต้น
- สมาชิก เป็นต้น

การใช้สัญลักษณ์สิ่งที่อยู่ภายนอก

1.ใช้สัญลักษณ์สิ่งที่อยู่ภายนอกคู่กับสัญลักษณ์กระแสข้อมูลเสมอโดยที่ถ้าลูกศรชี้เข้าหมายถึงเป็นการนำข้อมูลจากหน่วยงานภายนอกเข้าสู่ระบบถ้าลูกศรชี้ออกหมายถึงส่งข้อมูลจากระบบไปให้หน่วยงานภายนอก

2.1.9 แผนภาพ ER Diagram



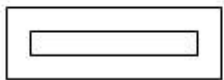
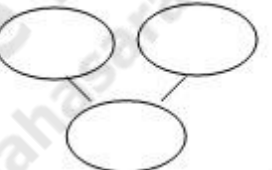
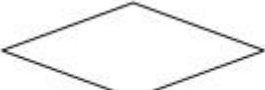
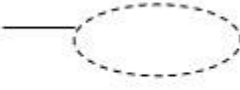




กิตติและพินดา(2546:200)ได้ให้ความหมายและข้อมูลที่เกี่ยวข้องกับแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลไว้ว่าการสร้างแผนภาพจำลองข้อมูลและกระบวนการดำเนินงานของระบบนั้นมีบทบาทสำคัญในการพัฒนาระบบเนื่องจากสามารถแสดงโครงสร้างของข้อมูลและการทำงานภายในระบบชัดเจนซึ่งจะช่วยให้ทั้งนักวิเคราะห์ระบบและผู้ใช้งานเกิดความเข้าใจในการทำงานของระบบอย่างถูกต้อง แบบจำลองข้อมูลที่สร้างขึ้นในขั้นตอนการวิเคราะห์ความต้องการของระบบนี้ยังเรียกว่าเป็น “การออกแบบฐานข้อมูลในระดับแนวความคิด (Conceptual Database Design)” ของขั้นตอนการออกแบบ (Design Phase) ในกิจกรรมการออกแบบฐานข้อมูลซึ่งจะนำ Conceptual Data Model ที่ได้จากกิจกรรมย่อยนี้ไปทำการปรับปรุงและออกแบบฐานข้อมูลในระดับ Logical และ Physical ต่อไป ในที่นี้เพื่อความสะดวกจะเรียก Conceptual Data Model ว่า “Data Model”

แบบจำลองข้อมูล (Data Model) หมายถึง การจำลองข้อมูลที่เกิดขึ้นทั้งหมดในระบบพร้อมทั้งจำลองความสัมพันธ์ระหว่างข้อมูลที่เกิดขึ้นนั้นโดยใช้แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (Entity Relationship Diagram : ER Diagram)

แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลหมายถึงแผนภาพที่ใช้เป็นเครื่องมือสำหรับจำลองข้อมูล ซึ่งประกอบไปด้วย Entity (แทนกลุ่มของข้อมูลที่เป็นเรื่องเดียวกันหรือเกี่ยวข้องกัน) และความสัมพันธ์ระหว่างข้อมูล(Relationship)ที่เกิดขึ้นทั้งหมดในระบบ

2.1.9.1 สัญลักษณ์ที่ใช้ใน ER Diagram

ตารางที่ 2.3 ตารางสัญลักษณ์ที่ใช้ใน 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

2.1.9.2 องค์ประกอบของ ER Diagram

การสร้างแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลตามที่กล่าวไว้นั้นใช้พื้นฐานหลัก 3 ประการ ดังนี้

2.1.9.2.1 Entity คือ บุคคล สถานที่วัตถุและรวมถึงเหตุการณ์ที่ทำให้เกิดกลุ่มของข้อมูลที่ต้องการจัดเก็บ ซึ่งสามารถบ่งชี้ถึงความเป็นเอกลักษณ์เฉพาะตัวได้ (Uniquely Identifiable)

2.1.9.2.2 Relationships คือความสัมพันธ์ระหว่างเอนิตตี้ (Entity) โดยจะเป็นไปตามชนิดของแต่ละความสัมพันธ์อาจกล่าวได้อีกนัยหนึ่งว่าความสัมพันธ์ของแต่ละเอนิตตี้นี้จะเกิดขึ้นตาม

ธรรมชาติในกระบวนการทางธุรกิจซึ่งความสัมพันธ์จะนำเสนอด้วยเหตุการณ์เชื่อมโยงระหว่างเอนทิตี เช่น พนักงานมีความสัมพันธ์กับแผนกที่ต้นสังกัดอยู่ เป็นต้น สำหรับข้อกำหนดในความสัมพันธ์ (Constraints) จะเป็นกฎเกณฑ์ที่ใช้บังคับเงื่อนไขเพื่อให้การจัดเก็บข้อมูลในฐานข้อมูลเป็นไปอย่างเหมาะสมและถูกต้อง โดยข้อกำหนดความสัมพันธ์จะเป็นเงื่อนไขที่ใช้บังคับส่วนต่างๆ ในแบบจำลองซึ่งโปรแกรมจะต้องรักษาให้ถูกต้องตามความเป็นจริงเสมอแสดง Cardinality Constraints โดยความหมายของความสัมพันธ์แต่ละแบบได้อธิบายไว้ดังนี้

(1) ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-One Relationship) เป็นความสัมพันธ์ระหว่างเอนทิตีหนึ่งไปมีความสัมพันธ์กับอีกเอนทิตีหนึ่งเพียงหนึ่งรายการเท่านั้น

(2) ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-Many Relationship) เป็นความสัมพันธ์ระหว่างเอนทิตีหนึ่งไปมีความสัมพันธ์กับอีกเอนทิตีหนึ่งมากกว่าหนึ่งรายการ

(3) ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-Many Relationship) เป็นความสัมพันธ์หลายรายการระหว่างเอนทิตีทั้งสอง

2.1.9.2.3 Attributes คือ คุณสมบัติของเอนทิตี เช่น เอนทิตีลูกค้าประกอบด้วยแอตทริบิวต์ (Attributes) รหัสลูกค้า, ชื่อ, นามสกุล, เพศ, ที่อยู่, โทรศัพท์และวันเกิด

2.2 ระบบงานที่เกี่ยวข้อง

2.2.1 แอปพลิเคชันแมลงมัน

แมลงมันรบกวนกัดกินผัก ผลไม้ แอปพลิเคชันแมลงมันเอาอยู่รวบรวมสารเคมี ตัวยาในการป้องกัน และกำจัดแมลงศัตรูพืช และผลไม้พร้อมทั้งรายละเอียดตัวสารเคมี ยี่ห้อ อัตราการผสมยาฆ่าแมลง วิธีการพ่นยาฆ่าแมลง

- การเลือกยาฆ่าแมลง
- เนื้อหาพิเศษการเลือกซื้อยาฆ่าแมลงแต่ละยี่ห้อ
- ความรู้ในการเลือกซื้อยาฆ่าแมลงที่ถูกต้องและเหมาะสมกับแมลงแต่ละชนิด



ภาพประกอบที่ 2.1 ตัวอย่างแอปพลิเคชันแมลงมัน

2.2.2 แอปพลิเคชันมูมหมอฟีช

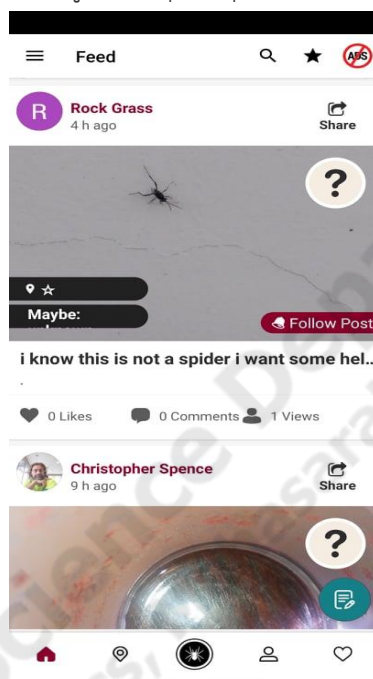
แอปมูมหมอฟีช เป็นแอปที่ให้ความรู้เรื่องของพืช ด้วยตนเองพร้อมข้อมูลประกอบให้รู้จักในเรื่องขั้นตอนการบำรุงดูแล ชนิดและการป้องกันกำจัดวัชพืช อาการและการป้องกันกำจัดโรคพืช



ภาพประกอบที่ 2.2 ตัวอย่างแอปพลิเคชันมูมหมอฟีช

2.2.3 แอปพลิเคชันตัวระบุแมงมุม

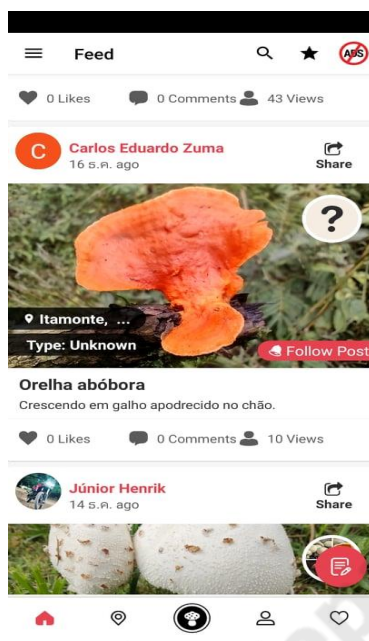
ตัวระบุแมงมุม เป็นแอปที่ให้ความรู้เรื่องของแมงมุม ด้วยตนเองพร้อมข้อมูลประกอบให้รู้จักในเรื่องระบุแมงมุมทันทีด้วยตัวระบุแมงมุม แมงมุมหรือผีเสื้อและแมงมุมอื่น ๆ ฐานข้อมูลคุณภาพสูงดูแลโดยนักวิทยาศาสตร์และพนักงานทั่วโลก การตรวจสอบแมงมุมติดกับดักระบุทุกที่ทุกเวลาโดยใช้ตัวระบุแมงมุม หนังสือรายวันของแมงมุมที่ติดอยู่ในแมงมุมระบุ



ภาพประกอบที่ 2.3 ตัวอย่างแอปพลิเคชันตัวระบุแมงมุม

2.2.4 แอปพลิเคชันระบุเห็ด

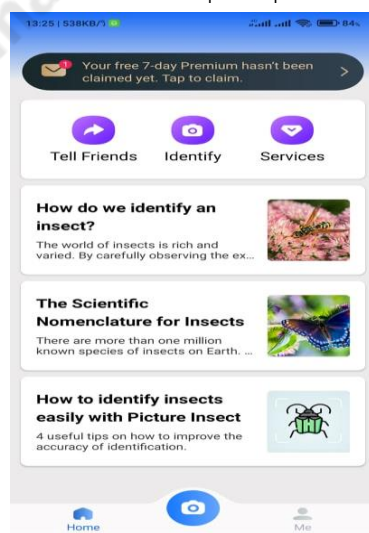
ตัวระบุแมงมุม เป็นแอปที่ให้ความรู้เรื่องของเห็ด ด้วยตนเองพร้อมข้อมูลประกอบให้รู้จักในเรื่องระบุเห็ดด้วยตัวระบุเห็ดทันที ฐานข้อมูลคุณภาพสูงดูแลโดยนักวิทยาศาสตร์และพนักงานทั่วโลก



ภาพประกอบที่ 2.4 ตัวอย่างแอปพลิเคชันตัวระบุเห็ด

2.2.5 แอปพลิเคชัน Picture Insect Bug Identifier

แอปพลิเคชัน Picture Insect Bug Identifier ให้ข้อมูลเชิงลึกเกี่ยวกับแมลง เรียนรู้เกี่ยวกับโครงสร้าง ลักษณะ วิวัฒนาการ สายพันธุ์ที่อาจสับสน ลักษณะ และอื่นๆ ของแมลง ระบุแมลงมากกว่า 4,000 ชนิดทันที อัตราความแม่นยำในการระบุ 95% แหล่งเรียนรู้เกี่ยวกับแมลงมากมาย Interface ที่ใช้งานง่ายพร้อมคำแนะนำที่เป็นมิตร ติดตามสายพันธุ์ที่ระบุในคอลเล็กชันส่วนตัวของคุณ



ภาพประกอบที่ 2.5 ตัวอย่างแอปพลิเคชัน Picture Insect Bug Identifier

2.2.6 ตารางเปรียบเทียบแอปพลิเคชัน

ตารางที่ 2.4 ตารางเปรียบเทียบแอปพลิเคชัน

ฟังก์ชันการทำงาน	pests	แมลงมัน	มูมทอมพีช	แอปพลิเคชัน ตัวระบุเห็ด	แอปพลิเคชัน Picture Insect Bug Identifier
ฟังก์ชันการเข้าสู่ระบบ	/		/	/	/
ฟังก์ชันการลงทะเบียน	/		/	/	/
ฟังก์ชันการจัดการข้อมูลสมาชิก	/		/	/	/
ฟังก์ชันการจัดการข้อมูลแมลง	/	/	/		
ฟังก์ชันการค้นหาข้อมูลแมลง	/	/		/	/
ฟังก์ชันการแยกหมวดหมู่ข้อมูลแมลง	/	/	/		
ฟังก์ชันรายงานข้อมูลแมลง ต่างๆ ของสมาชิก	/			/	/
ฟังก์ชันแสดงข้อมูลตำแหน่งที่พบแมลงศัตรูพืชลงในแผนที่	/				