

2.1 ทฤษฎีทางด้านระบบฐานข้อมูล

2.1.1 ความหมายของระบบฐานข้อมูล

ฐานข้อมูลประกอบไปด้วยรายละเอียดของข้อมูลที่เกี่ยวข้องกัน ซึ่งถูกนำมาใช้ในงานด้านต่างๆ เช่น ด้านทะเบียนนักศึกษา จะมีฐานข้อมูลเกี่ยวกับนักศึกษา ข้อมูลเกี่ยวกับที่เกี่ยวกับรายวิชาที่ลงทะเบียน ข้อมูลอาจารย์ผู้สอน ข้อมูลเกี่ยวกับเกรดของนักศึกษา ข้อมูลการให้สินเชื่อ หรือข้อมูลเกี่ยวกับร้านค้าจะมีข้อมูลเกี่ยวกับ เจ้าหน้าที่การค้า ลูกหนี้การค้า ข้อมูลเกี่ยวกับสินค้าคงคลัง เป็นต้น ข้อมูลเหล่านี้จะถูกเก็บอย่างมีระบบเพื่อประโยชน์ในการจัดการและเรียกใช้ข้อมูลได้อย่างมีประสิทธิภาพ

องค์ประกอบของระบบฐานข้อมูล

ระบบฐานข้อมูลส่วนใหญ่เป็นระบบที่มีการนำระบบคอมพิวเตอร์เข้ามาช่วยในการจัดเก็บ โดยมีโปรแกรม Software ช่วยในการจัดเก็บข้อมูลเหล่านี้เพื่อให้ได้ข้อมูลตามที่ต้องการ

ฮาร์ดแวร์ (Hardware)

ในระบบฐานข้อมูลที่มีประสิทธิภาพควรมีฮาร์ดแวร์ต่างๆ ที่พร้อมจะอำนวยความสะดวกในการบริหารข้อมูลได้อย่างมีประสิทธิภาพไม่ว่าจะเป็นความเร็วของหน่วยประมวลผลกลาง ขนาดของหน่วยความจำหลัก อุปกรณ์นำเข้าและออกข้อมูล รายงานหน่วยความจำสำรองที่จะรองรับการประมวลผลข้อมูลในระบบได้อย่างมีประสิทธิภาพ

ซอฟต์แวร์ (Software)

ในการประมวลผลข้อมูลอาจจะใช้ซอฟต์แวร์หรือโปรแกรมที่แตกต่างกัน ทั้งนี้ขึ้นอยู่กับคอมพิวเตอร์ที่นำมาใช้ว่าเป็นแบบใด โปรแกรมจะทำหน้าที่ดูแลการสร้าง การเรียกใช้ข้อมูล การจัดทำรายงาน การปรับเปลี่ยน แก้ไข โครงสร้างในการควบคุม หรืออาจกล่าวได้อีกอย่างหนึ่งว่า ระบบจัดการฐานข้อมูล (Database Management System : DBMS) คือโปรแกรมที่ทำหน้าที่ในการจัดการฐานข้อมูล โดยเป็นสื่อกลางระหว่างผู้ใช้และโปรแกรมประยุกต์ต่างๆ ที่มีอยู่ในฐานข้อมูล ตัวอย่างของโปรแกรมระบบการจัดการฐานข้อมูล ได้แก่ DBASE IV, EXCEL, ACCESS, INFORMIX, ORACLE เป็นต้น

ข้อมูล (Data)

ฐานข้อมูลเป็นการจัดเก็บรวบรวมข้อมูลให้เป็นศูนย์กลางข้อมูลอย่างมีระบบ ซึ่งข้อมูลเหล่านี้สามารถเรียกใช้ร่วมกันได้ ผู้ใช้ข้อมูลในระบบฐานข้อมูล จะมองภาพข้อมูลในลักษณะที่แตกต่างกันเช่น ผู้ใช้บางคนมองภาพของข้อมูลที่จัดเก็บไว้ในสื่อข้อมูล ผู้ใช้ บางคนมองภาพข้อมูลจากการใช้งาน

เขตข้อมูล (field)

ขนาดความยาวของเขตข้อมูล (field) ที่สามารถกำหนดได้ เช่น เขตข้อมูล "ชื่อ" อาจใช้ความยาวถึงขนาด 30 ตัวอักษร ส่วนเขตข้อมูล "อายุ" อาจใช้เพียง 2 ตัวอักษร เป็นต้น แฟ้มข้อมูล (file) แต่ละแฟ้มจะต้องกำหนดเขตข้อมูลที่ใช้ชื่อเดียวกันให้ยาวเท่ากัน เช่น ถ้าเป็นเขตข้อมูล "ชื่อ" ไม่ว่าจะชื่อสั้นหรือยาวเพียงใด ก็ต้องใช้เนื้อที่ 30 ตัว อักษรหมดทั้งแฟ้ม ชื่อเขตข้อมูล (Field Name) ตั้งชื่อฟิลด์ได้ยาว ไม่เกิน 64 ตัวอักษร มีช่องว่าง เว้นวรรคได้ แต่ยกเว้นเครื่องหมายต่อไปนี้ คือ . ! [] สามารถตั้งชื่อเป็นภาษาไทย / ภาษาอังกฤษไทย

ชนิด	คำอธิบาย
CHAR(M)	จะเอาไว้เก็บข้อมูลที่เป็น string (สายอักษร) โดยที่ขนาดของการเก็บมีความคงที่ first name CHAR(25);
VARCHAR(M)	ข้อมูลที่เป็น string (สายอักษร) โดยที่ขนาดของการเก็บมีความคงที่ first name CHAR(25);
INT(M)	INT เก็บค่าจำนวนเต็ม มีค่าตั้งแต่ -2147483648 ถึง +2147483647

ตารางที่ 2.1 ตารางตัวอย่างชนิดของข้อมูล (Data type)

ขนาดหรือความยาวของเขตข้อมูล (Field size)

ในกรณีที่มีการกำหนด Field ไต ๆ เป็นแบบตัวเลข (Number) จะต้องกำหนด Field Size ของตัวเลข ให้ตรงตามที่ต้องการด้วย ซึ่งแบบของ Field Size แต่ละชนิดจะแตกต่างกันที่ขอบเขตตัวเลขที่สามารถเก็บได้ ในแต่ละ Field

ชนิด	คำอธิบาย
Byte	ข้อมูลตัวเลขจำนวนเต็มมีค่าระหว่าง 0 ถึง 255
Integer	ข้อมูลตัวเลขจำนวนเต็มมีค่าระหว่าง -32,768 ถึง 32,767
Long Integer	ข้อมูลตัวเลขจำนวนเต็มมีค่าระหว่าง -2,147,483,648 ถึง 2,147,483,647
Single	ข้อมูลตัวเลขทศนิยมตั้งแต่ -3.4×10^{38} ถึง 3.4×10^{38}
Double	ข้อมูลตัวเลขทศนิยมตั้งแต่ -1.797×10^{308} ถึง 1.797×10^{308}
Replication ID	ข้อมูลสำหรับ Replication
Decimal	ข้อมูลตัวเลขเป็นทศนิยม

ตารางที่ 2.2 ตารางตัวอย่างชนิดขนาดเขตข้อมูล

ระเบียบ (Record)

ระเบียบ เป็นที่รู้จักแพร่หลายทั่วไปในชีวิตปัจจุบัน ตัวอย่าง เช่น ข้อมูลของนักศึกษาที่เก็บไว้ที่ หน่วยทะเบียนของมหาวิทยาลัย หรือ ข้อมูล เกี่ยวกับหนังสือในห้องสมุด ระเบียบดังกล่าวบันทึกไว้ในแฟ้มเอกสาร หรือในบัตรดรรชนี หรือไว้ในระบบคอมพิวเตอร์ ระเบียบ เป็นประโยชน์มากในการเขียนโปรแกรม เมื่อกล่าวถึงระเบียบในชีวิตประจำวันรวมความถึงการเก็บข้อมูลในหน่วยความจำ นอกจากนี้อาจคิดว่าระเบียบเป็นการเก็บ ข่าวสารที่เกี่ยวข้องกับวัตถุสิ่งเดียว ซึ่งเราสามารถอ้างถึงข้อมูลทั้งข้อมูล หรืออ้างถึงเฉพาะบางส่วน โดยการเรียกชื่อของข้อมูลสำหรับหน่วยย่อยของระเบียบ เรียกว่า เขต (field)

ตัวอย่างของระเบียบศึกษาได้จากทะเบียนนักศึกษา ซึ่งประกอบด้วย 5 เขต ดังนี้ คือ

- 1) ชื่อ (first name)
- 2) นามสกุล (surname)
- 3) วัน เดือน ปีเกิด (date of birth)
- 4) เพศ (sex)
- 5) สถานะสมรส (marital status)

ประโยชน์ของระเบียบ คือ สามารถเรียกระเบียนที่เกี่ยวข้องกับข้อมูลหลายตัวได้ เช่น หน่วยทะเบียนต้องการรายละเอียด ของบุคคลหลายคน เช่น 'my record' หรือ 'David's record' ระเบียบแต่ละอันประกอบด้วยเขตชนิดเดียวกัน ดังนั้นจึงสามารถกล่าวถึง 'my surname' หรือ 'David's date of birth' ในแต่ละกรณีเราจะละข้อความแสดงความเป็นเจ้าของ ('my' หรือ 'David's') ซึ่งบ่งถึงระเบียบที่กำลังกล่าวถึง โดยใช้ 'surname' หรือ 'date of birth' เพื่อระบุถึงเขตของ

ระเบียบที่กำลังกล่าวถึง ในทำนองเดียวกัน เราสามารถกล่าวถึงวันที่ที่ต่างกัน เช่น 'today' หรือ 'the date of the next eclipse' และเกี่ยวกับเขตของวันที่นั้น เช่น 'the year of the next eclipse' สิ่งที่สำคัญอีกประการหนึ่ง คือ ความแตกต่างระหว่างระเบียบและแถวลำดับ ข้อแรก คือ เขตของระเบียบ ประกอบด้วยข้อมูลต่างชนิด แต่แถวลำดับต้องมีข้อมูลภายในที่เป็นชนิดเดียวกัน ข้อที่สอง การอ้างถึงเขตของระเบียบกระทำโดยเรียกชื่อเขต ขณะที่การอ้างถึงข้อมูลในแถวลำดับ กระทำโดยเรียกตัวดรรชนี (index) ซึ่งอาจจะเป็นตัวแปร ดังนั้น จึงเป็นไปได้ที่จะกล่าวถึงข้อมูล ตัวที่ ของแถวลำดับ ในทางตรงกันข้ามการกล่าวถึงเขตที่ ของระเบียบจะไม่มีผลถ้าผู้ใช้ไม่รู้ว่า เขตนั้นเป็นชนิดอะไร

การกำหนดระเบียบ

การกำหนดชนิดของระเบียบในปาสกาล ทำได้โดยการเขียนชื่อตัวแปรและชนิดของเขต อยู่ระหว่าง record และ end โดยที่ชนิดของเขตจะเป็นแบบใดก็ได้ มีการใช้เครื่องหมาย ';' เป็นตัวแบ่งเขต สำหรับการอ้างถึงเขตในระเบียบทำโดยใช้ตัวบอกเขต (field designator)

แฟ้มข้อมูล (File)

แฟ้มข้อมูล จะเป็นส่วนที่รวบรวมเอาระเบียบข้อมูลที่เป็นเรื่องเดียวกัน มารวมเข้าไว้ด้วยกัน เช่น แฟ้มข้อมูลประวัตินักศึกษา ก็จะรวบรวมเอาระเบียบข้อมูลเกี่ยวกับประวัติของนักศึกษา แต่ละคนที่มีอยู่ทั้งหมดมารวมเข้าไว้ด้วยกัน

ประเภทของแฟ้มข้อมูล

ที่จะนำมาใช้ในการประมวลผลด้วยเครื่องคอมพิวเตอร์ อาจมีเพียงแฟ้มข้อมูลเดียวหรือหลายแฟ้มข้อมูลก็ได้ และแต่ละข้อมูล สามารถนำมาแก้ไขตัดต่อและเรียงลำดับของข้อมูลใหม่ได้ในเวลาต่างกัน ซึ่งประเภทของแฟ้มข้อมูลสามารถแบ่งได้เป็น 2 ประเภทหลักๆ คือ

1) แฟ้มข้อมูลหลัก (Master File) หรือเรียกอีกชื่อหนึ่งว่า แฟ้มข้อมูลถาวร (Permanent File) แฟ้มข้อมูลประเภทนี้จะถูกเก็บไว้อย่างถาวร เมื่อนำมาประมวลผลแล้วจะเก็บไว้ใช้งานเป็นเวลานาน เพื่อที่จะนำมาใช้อ้างอิงภายหลังได้ และถูกนำมาใช้เสมอๆ ดังนั้น ข้อมูลต้องเป็นข้อมูลที่ทันสมัยและที่สำคัญคือ ต้องตรงกับความเป็นจริง เช่น แฟ้มข้อมูลหลักของสินค้าคงคลัง (Inventory file)

2) แฟ้มข้อมูลรายการ (Transaction File) หรือเรียกอีกชื่อหนึ่งว่า แฟ้มข้อมูลชั่วคราว (Temporary File) แฟ้มข้อมูลประเภทนี้มีลักษณะตรงกันข้ามกับแฟ้มข้อมูลหลัก เพราะเป็นแฟ้มข้อมูลที่รวบรวมข้อมูลบางส่วนที่มีความสัมพันธ์กับแฟ้มข้อมูลหลัก ซึ่งแฟ้มข้อมูลรายการจะเป็น

แฟ้มข้อมูลที่สามารถแก้ไขแฟ้มข้อมูลหลักบางรายการได้ เช่น การเพิ่มข้อมูล, การลบข้อมูลบางระเบียบ หรือจะเป็นการแก้ไขข้อมูลในเขตข้อมูลของระเบียบต่างๆ เป็นต้น

เอนทิตี (Entity)

ชื่อของสิ่งหนึ่งสิ่งใด เปรียบเสมือนคำนาม ได้แก่ บุคคล สถานที่ สิ่งของ เช่น นักศึกษา อาจารย์ ภาควิชา ฯลฯ ที่ต้องการจัดเก็บข้อมูลนั้นไว้ สำหรับบาง Entity ในฐานข้อมูลจะไม่มี ความหมายหากไม่มี Entity อื่นในฐานข้อมูล เอนทิตีประเภทนี้เรียกว่า Entity ชนิดอ่อนแอ (Weak Entity) นอกจากนี้ ยังมีการแบ่งเอนทิตีออกเป็น Supertype และ Subtype โดยที่ Subtype ประกอบด้วยข้อมูลเฉพาะนอกเหนือจากที่มีอยู่ใน Supertype ดังนั้น เมื่อมีข้อมูลของ Subtype ก็จะต้องมีข้อมูลของ Supertype อยู่เช่นกัน กล่าวอีกนัยหนึ่ง Entity(A) เป็น Subtype ของ Entity(B) และ Entity(B) เป็น Supertype ของ Entity(A) ก็ต่อเมื่อ Subtype A ประกอบด้วยข้อมูลทุกอย่างที่มีใน Supertype B และมีข้อมูลเฉพาะของตัวเองเพิ่มเติม และเมื่อมีข้อมูลของ Subtype จะต้อง มีข้อมูลของ Supertype

แอททริบิวต์ (Attribute)

รายละเอียดของข้อมูล (ในแต่ละคอลัมน์ (Column) หรือฟิลด์ (Field)) ซึ่งเป็นข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity เช่น Entity ของนักศึกษา ได้แก่ ชื่อ อายุ ฯลฯ บางแอททริบิวต์ ประกอบด้วยข้อมูลหลายส่วนมารวมกัน ซึ่งอาจแยกเป็นชื่อแอททริบิวต์ย่อยๆ ได้อีก แอททริบิวต์ที่มีคุณสมบัติเช่นนี้ เรียกว่า แอททริบิวต์ผสม (Composite Attribute) นอกจากนี้ แอททริบิวต์บาง แอททริบิวต์อาจจะไม่มีค่าของตัวเอง แต่สามารถหาค่าได้จากแอททริบิวต์อื่นๆ แอททริบิวต์ที่มีคุณสมบัติอย่างนี้เรียกว่า แอททริบิวต์ที่ถูกแปลค่ามา (Derived Attribute) กล่าวอีกนัยหนึ่ง Entity(A) เป็น Subtype ของ Entity(B) และ Entity(B) เป็น Supertype ของ Entity(A) ก็ต่อเมื่อ Subtype A ประกอบด้วยข้อมูลทุกอย่างที่มีใน Supertype B และมีข้อมูลเฉพาะของตัวเองเพิ่มเติม และเมื่อมีข้อมูลของ Subtype จะต้อง มีข้อมูลของ Supertype

รีเลชัน (Relation)

รีเลชัน (Relation) มีคำเรียกทั่วไปว่า ตารางข้อมูล (Table) เนื่องจากเป็นคำนามที่แทน ข้อมูลของเรื่องใดเรื่องหนึ่ง จึงใช้คำว่า รีเลชัน แทนความหมายของตารางในระบบฐานข้อมูลเชิงสัมพันธ์

ชนิด	คำอธิบาย
Tuple	ค่าของข้อมูลที่อยู่ในแต่ละแถว (Row) หรือที่เรียกกันว่า เรคอร์ด (Record)
Cardinality	จำนวนแถวของข้อมูลในตาราง
Attribute	รายละเอียดของข้อมูลในแต่ละคอลัมน์ (Column) หรือฟิลด์ (Field)
Primary Key	แอททริบิวต์ที่มีค่าของข้อมูลเป็นเอกลักษณ์หรือเฉพาะเจาะจง หรือ เป็นค่าที่ไม่ซ้ำกันในแต่ละทูเพิล (Tuple)
Domain	ขอบเขตของค่าของข้อมูล

ตารางที่ 2.3 ตารางชนิดของรีเลชัน

คีย์หลัก (Primary Key)

เป็นแอททริบิวต์ที่มีคุณสมบัติของข้อมูลที่เป็นค่าเอกลักษณ์หรือมีค่าที่ไม่ซ้ำซ้อนกัน คุณสมบัติดังกล่าวจะสามารถระบุข้อมูลนั้นเป็นข้อมูลของ ทูเพิล/ เรคอร์ดใด แอททริบิวต์ที่มีคุณสมบัติเป็นคีย์หลักอาจประกอบด้วยหลายแอททริบิวต์/ คอลัมน์/ ฟิลด์รวมกัน เพื่อที่จะกำหนดค่าที่เป็นเอกลักษณ์ได้ คีย์หลักที่ประกอบด้วยหลายแอททริบิวต์นี้เรียกว่า คีย์ผสม (Composite Key) นั่นคือเมื่อแอททริบิวต์แต่ละตัวประกอบกันจึงจะให้ค่าที่เป็นเอกลักษณ์หรือไม่ซ้ำซ้อนกันได้

2.1.2 การจัดการฐานข้อมูล

การจัดการฐานข้อมูล(Database Management) คือ การบริหารแหล่งข้อมูลที่ถูกเก็บรวบรวมไว้ที่ศูนย์กลาง เพื่อตอบสนองต่อการใช้ของโปรแกรมประยุกต์อย่างมีประสิทธิภาพและลดการซ้ำซ้อนของข้อมูล รวมทั้งความขัดแย้งของข้อมูลที่เกิดขึ้นภายในองค์กร ในอดีตการเก็บข้อมูลมักจะเป็นอิสระต่อกันไม่มีการเชื่อมโยงของข้อมูลเกิดการ สิ้นเปลืองพื้นที่ในการเก็บข้อมูล

2.1.3 ส่วนประกอบระบบการจัดการฐานข้อมูล

2.1.3.1 ภาษาคำนิยามของข้อมูล (Data Definition Language: DDL)

ในส่วนนี้จะกล่าวถึงส่วนประกอบของระบบการจัดการฐาน ข้อมูลว่าข้อมูลแต่ละส่วนประกอบด้วยอะไรบ้าง (Data element) ในฐานข้อมูลซึ่งเป็นภาษาทางการที่นักเขียนโปรแกรมใช้ในการ สร้างเนื้อหาข้อมูลและโครงสร้างข้อมูลก่อนที่ข้อมูลดังกล่าวจะถูกแปลงเป็น แบบฟอร์มที่ ต้องการของโปรแกรมประยุกต์หรือในส่วนของ DDL จะประกอบด้วยคำสั่งที่ใช้ในการกำหนด

โครงสร้างข้อมูลว่ามีคอลัมน์อะไร แต่ละคอลัมน์เก็บข้อมูลประเภทใด รวมถึงการเพิ่มคอลัมน์ การกำหนดดัชนี เป็นต้น

2.1.3.2 ภาษาการจัดการฐานข้อมูล (Data Manipulation Language: DML)

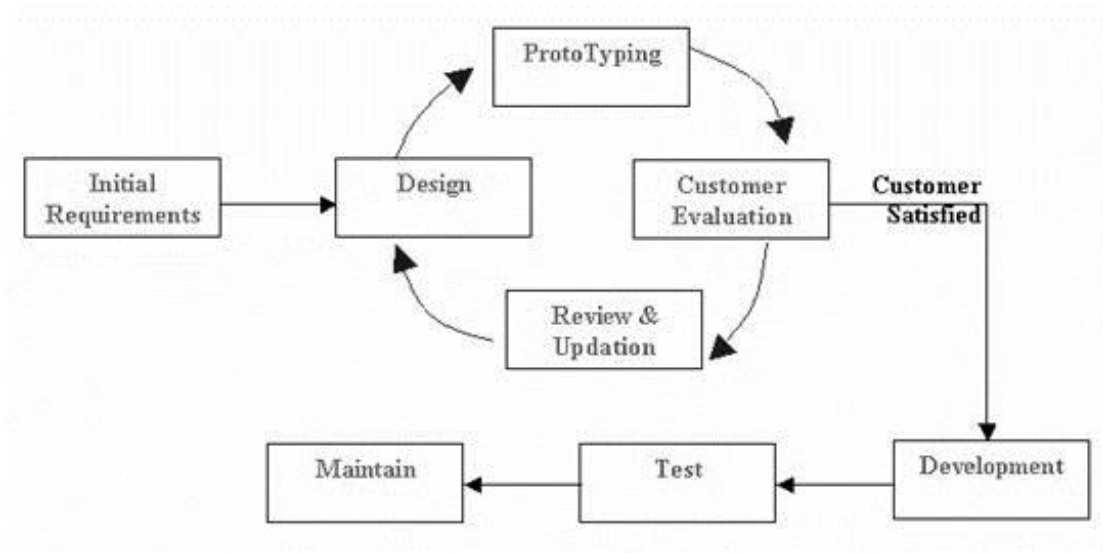
เป็นภาษาเฉพาะที่ใช้ในการจัดการระบบฐานข้อมูล ซึ่งอาจจะเป็นการเชื่อมโปรแกรมภาษาในยุคที่สามและยุคที่สี่เข้าด้วยกันเพื่อจัดการข้อมูลในฐานข้อมูล ภาษานี้มักจะประกอบด้วยคำสั่งที่อนุญาตให้ผู้ใช้สามารถสร้างโปรแกรมพิเศษขึ้นมา รวมถึงข้อมูลต่างๆ ในปัจจุบันที่นิยมใช้ ได้แก่ ภาษา SQL (Structure Query Language) แต่ถ้าหากเป็นเครื่องคอมพิวเตอร์ขนาดใหญ่ DBMS มักจะสร้างด้วยภาษาโคบอล (COBOL language) ภาษาฟอร์แทรน (FORTRAN) และภาษาอื่นในยุคที่สาม

2.1.3.3 พจนานุกรมข้อมูล (Data Dictionary)

เป็นเครื่องมือสำหรับการเก็บและการจัดข้อมูลสำหรับการบำรุงรักษาในฐานข้อมูล โดยพจนานุกรมจะมีการกำหนดชื่อของสิ่งต่างๆ (Entity) และระบุไว้ในโปรแกรมฐานข้อมูล เช่น ชื่อของฟิลด์ ชื่อของโปรแกรมที่ใช้รายละเอียดของข้อมูล ผู้มีสิทธิ์ใช้และผู้ที่ได้รับผิดชอบ แสดงส่วนประกอบของระบบการจัดการฐานข้อมูล

2.2 ทฤษฎีกระบวนการผลิตซอฟต์แวร์

Prototype Model หรือ โมเดลที่ใช้ในการพัฒนา



ภาพที่ 2.1 ขั้นตอนการสร้างซอฟต์แวร์โดย Prototype Model Process

เป็นการทำต้นแบบ (Prototype) ของซอฟต์แวร์ออกมาก่อนซึ่งต้นแบบนี้จะมีรูปร่างหน้าตาที่เหมือนจริง ทำให้ผู้คิดค้นสามารถรู้ได้ทันทีว่าตรงกับความต้องการหรือไม่ แม้ว่ากลไกการทำงานข้างในต้นแบบจะยังไม่มี แต่การทำต้นแบบเสร็จเร็วก็จะช่วยให้ขั้นตอนการพัฒนาซอฟต์แวร์เสร็จเร็วด้วย นอกจากนี้การแก้ไขต้นแบบก็ทำได้ง่าย เพราะไม่มีอะไรซับซ้อนข้างในแทนที่จะไปทำการแก้ไขที่ระบบจริง ซึ่งแบบจำลองต้นแบบ (Prototype Model) สามารถแบ่งได้เป็น 2 ประเภท ขึ้นอยู่กับการนำเอาแบบจำลองต้นแบบ (Prototype Model) ไปใช้งาน

- 1) Throw Away Prototype การสร้างตัวต้นแบบ แบบใช้งานเสร็จแล้วทิ้ง สร้างมาเพื่อให้เห็นรูปแบบของ interface ตัวโปรแกรมต่างๆ ส่วนมากเป็นการเขียนในกระดาษ วาดกระดาน
- 2) Evolutionary Prototype การสร้างตัวต้นแบบที่สามารถใช้งานได้จริงๆ อาจจะค่อยๆ ทำในที่ละส่วนปรับปรุงแก้ไขโปรแกรมจนกระทั่งเป็นโปรแกรมจริงที่สามารถใช้งานได้ตามความต้องการ

Prototype Model มีขั้นตอนดังนี้

- 1) Initial Requirement เริ่มเก็บความต้องการจากผู้ใช้งาน หลังจากนั้นจะเริ่มวงจรการพัฒนาตัวต้นแบบเริ่มต้นจากการออกแบบ
- 2) Design เป็นการออกแบบ Module, Interface หลังจากนั้นนำตัวต้นแบบมาให้ผู้ใช้งานได้ทำงานลองใช้งานหรือให้ผู้ใช้งานได้เห็นได้ประเมินการใช้งาน
- 3) Customer Evaluation ถ้าหากผู้ใช้งานยอมรับหรือว่าตัวต้นแบบในการพัฒนาเป็นไปตามความต้องการของผู้ใช้งาน ทีมพัฒนายก็จะดำเนินการพัฒนาระบบต่อซึ่งขั้นตอนต่อไปก็จะเหมือนกับ Waterfall model แต่หากผู้ใช้งานต้องการให้แก้ไขงานเพิ่มเติมหรือไม่ตรงความต้องการ
- 4) Review and Update เป็นการนำความเห็นของผู้ใช้มาปรับปรุงต้นแบบให้มีประสิทธิภาพมากขึ้น ขั้นตอนนี้จะเกิดซ้ำไปซ้ำมาจนกระทั่งผู้ใช้เกิดความพอใจ แล้วจึงจะสามารถนำต้นแบบไปใช้งานได้

ข้อดีของ Prototype Model

- เรียนรู้จากตัวซอฟต์แวร์จริง
- ช่วยสื่อสารระหว่างผู้ใช้กับผู้พัฒนาระบบ เพื่อให้ทราบถึงความก้าวหน้าของระบบ
- ลูกค้าเข้าใจความต้องการที่แท้จริง และระบุความต้องการได้ชัดเจนขึ้น

ข้อเสียของ Prototype Model

- การทำต้นแบบจะต้องมีบุคลากรที่มีความสามารถในด้านการพัฒนาสูง เพื่อพัฒนาต้นแบบได้เร็ว
- การเปลี่ยนแปลงหรือการเพิ่มความต้องการ เพื่อสร้างต้นแบบหลาย Version ทำให้ได้โครงสร้างซอฟต์แวร์ที่ไม่ดี
- การผลิตเอกสารสำหรับต้นแบบในแต่ละ Version เป็นการสิ้นเปลืองงบประมาณ

2.3 ทฤษฎีหรือเทคโนโลยีและเทคโนโลยีที่ใช้ในการพัฒนา

2.3.1 ภาพรวมเทคโนโลยีต่าง ๆ

2.3.1.1 การเขียนโปรแกรมเชิงวัตถุ (Object-oriented programming : OOP)

OOP (Object Oriented Programming) เป็นวิธีการเขียนโปรแกรม โดยอาศัยแนวคิดของวัตถุซึ่งหนึ่งมีความสามารถในการปกป้องข้อมูล และการสืบทอดคุณสมบัติซึ่งทำให้แนวโน้มของ OOP ได้รับการยอมรับและพัฒนามาใช้ในระบบต่างๆ มากมาย เช่น ระบบปฏิบัติการ วินโดวส์ เป็นต้น ในที่นี้จะกล่าวถึงความเป็นมาและความหมายของแนวคิดการเขียนโปรแกรมเชิงวัตถุข้อควรรู้เกี่ยวกับหลักการเขียนโปรแกรมเชิงวัตถุตลอดจนความสำคัญของการเขียนและออกแบบระบบงานก่อนเขียนโปรแกรม รวมถึงประโยชน์ของการเขียนโปรแกรมเชิงวัตถุซึ่งจะทำให้ผู้เขียนได้เข้าใจถึงหลักการเบื้องต้นของแนวความคิดเชิงวัตถุได้

ความเป็นมาของแนวคิดแบบ OOP

แนวความคิดดั้งเดิมของการเขียนโปรแกรมคือ การแก้ปัญหาโดยใช้คอมพิวเตอร์เป็นเครื่องมือ คล้ายกับการใช้เครื่องคิดเลขในการแก้ปัญหาทางคณิตศาสตร์

แนวความคิดแบบใหม่ที่ใช้ในการเขียนโปรแกรมก็คือ การเน้นถึงปัญหาและองค์ประกอบของปัญหา (เพื่อแก้ปัญหา) การเน้นที่ปัญหาและองค์ประกอบของปัญหา (Problem Space) จะคล้ายกับแก้ไขปัญหาและชีวิตความเป็นอยู่ของมนุษย์ที่จะต้องมี คน สัตว์ สิ่งของ เพื่อแก้ปัญหา (มีหน้าที่แก้ปัญหา) มากกว่าจะมองที่วิธีการแก้ปัญหานั้นๆ หรือขั้นตอนในการแก้ปัญหา (Solution Space) ซึ่งเป็นวิธีการเขียนโปรแกรมแบบเก่านั้นเอง

อาลันเคย์ (Alan Kay) ได้เสนอกฎ 5 ข้อ ที่เป็นแนวทางของภาษาคอมพิวเตอร์เชิงวัตถุ หรือที่เรียกว่า Object-Oriented Programming (OOP) ไว้ดังนี้

- 1) ทุกๆ สิ่งเป็นวัตถุ (Everything is an Object)
- 2) โปรแกรมก็คือ กลุ่มของวัตถุที่ส่งข่าวสารบอกกันและกันให้ทำงาน (A Program is a Bunch of Object Telling Each Other What to do by Sending Messages)
- 3) ในวัตถุแต่ละวัตถุจะต้องมีหน่วยความจำและประกอบไปด้วยวัตถุอื่นๆ (Each Object has Its Own Memory Made Up of Other Objects)
- 4) วัตถุทุกชนิดจะต้องจัดอยู่ในประเภทใดประเภทหนึ่ง (Every Object has a Type)
- 5) วัตถุที่จัดอยู่ในประเภทเดียวกันย่อมได้รับข่าวสารเหมือนกัน (All Object of a Particular Type Can Receive the Same Messages)

แนวคิดแบบ OOP

OOP ก็คือ “ธรรมชาติของวัตถุ” หมายความว่า OOP จะมองสิ่งแต่ละสิ่งถือเป็น “วัตถุชิ้นหนึ่ง” (Object) มันจะมีสีแดงหรือสีเขียว ยาวหรือสั้นมันก็คือวัตถุชิ้นหนึ่งเหมือนกัน และเราสามารถกำหนดประเภทหรือคลาสให้กับวัตถุเหล่านั้นได้ นอกจากนี้เมื่อ OOP มองทุกสิ่งถือเป็นวัตถุชิ้นหนึ่งแล้วยังสามารถคิดต่อไปอีกว่า “วัตถุแต่ละอย่างนั้น ต่างก็มีลักษณะและวิธีการใช้งานเป็นของตัวเอง” หมายความว่า วัตถุแต่ละชนิดหรือแต่ละชิ้นต่างก็มีรูปร่าง ลักษณะ และการใช้งาน (การกระทำ) ที่แตกต่างกันออกไปเราจะเรียกคุณลักษณะของวัตถุว่า แอททริบิวต์(Attribute) และจะเรียกวิธีการใช้งานวัตถุว่า Method ตัวอย่างเช่น “ดินสอเป็นวัตถุที่มีลักษณะเรียวยาวภายในเป็นไส้ถ่านใช้สำหรับเขียน การใช้ดินสอทำได้โดยใช้มือจับและเขียนลงบนวัสดุรองรับ” จากข้อความข้างต้นสามารถจับใจความได้ว่าคุณลักษณะของวัตถุ (Attribute) ก็คือ “ยาวเรียวยาวภายในเป็นไส้ถ่าน” ส่วนการใช้งาน Method คือ “ใช้มือจับและเขียนลงบนวัสดุรองรับ” จะเห็นได้ว่าแนวคิดของ OOP นั้นจะมีลักษณะที่คล้ายกับธรรมชาติของสิ่งหนึ่งซึ่งสามารถแบ่งแยกสิ่งต่างๆ ออกเป็นแต่ละประเภทได้ ถ้านำเอาแนวคิดของ OOP มาใช้ในการเขียนโปรแกรมและการจัดการข้อมูล จะพบว่าโปรแกรมหรือฟังก์ชันจะมีความเป็นอิสระแก่กันอย่างเห็นได้ชัด ก็คือ โปรแกรมหรือฟังก์ชันแต่ละตัวถึงแม้จะมาจากที่เดียวกันแต่สามารถทำงานในคนละหน้าที่ เก็บข้อมูลคนละค่าได้ โดยจะไม่มายุ่งเกี่ยวกันแต่อย่างใด

หลักการเขียนโปรแกรมเชิงวัตถุ

เนื่องจากหลักการเขียนโปรแกรมเชิงวัตถุเป็นแนวคิดแบบใหม่นั้น การทำงานหลายๆ ส่วนของการเขียนโปรแกรมแบบนี้อาจจะยังไม่เป็นที่คุ้นเคยมากนักจึงจำเป็นที่ผู้ศึกษาต้องทำความเข้าใจการทำงานแนวคิดนี้ได้ดียิ่งขึ้น ข้อควรรู้เบื้องต้นเกี่ยวกับ OOP มีดังนี้

- การเชื่อมต่อ (Interface)

อินเทอร์เฟซ (Interface) คือ การเชื่อมต่อถ้าเป็นการเชื่อมต่อระหว่างผู้ใช้กับคอมพิวเตอร์ จะเรียกการเชื่อมต่อนั้นว่ายูสเซอร์อินเทอร์เฟซ (User Interface) แต่ในการเขียนโปรแกรมเชิงวัตถุ การเชื่อมต่อยังรวมไปถึงวัตถุ (Object) เพราะในวัตถุจะต้องมีอินเทอร์เฟซ อันเป็นส่วนที่วัตถุนั้นๆ จะให้บริการหรือเป็นส่วนที่บอกว่าวัตถุนั้นๆ สามารถทำอะไรได้บ้างซึ่งบางครั้งเรียกว่า เมธอด

ข้อดีของการมีอินเทอร์เฟซคือ การเปลี่ยนแปลงที่เกิดขึ้นภายในวัตถุจะไม่กระทบต่ออินเทอร์เฟซ ดังนั้น ภายในวัตถุผู้เขียนคำสั่งสามารถดัดแปลง แก้ไข หรือเพิ่มเติมได้ตลอดเวลา นอกจากนี้ ภายในวัตถุยังสามารถเก็บค่าต่างๆ ได้อีกด้วย

- การซ่อนรายละเอียด (Encapsulation)

ส่วนประกอบของวัตถุตามแนวความคิดการเขียนโปรแกรมเชิงวัตถุจะต้องประกอบไปด้วยสองส่วนเป็นอย่างน้อยคือ ส่วนของคุณสมบัติใช้เก็บข้อมูลรายละเอียด สถานะ โดยใช้ตัวแปรเก็บค่าต่างๆ ไว้ และส่วนของเมธอดที่เป็นตัวเชื่อมการทำงานของวัตถุนั้นๆ โดยผู้ใช้จะไม่สามารถติดต่อใช้งานกับตัวแปรที่อยู่ข้างในได้ในภาษา C++ จะใช้คำ Public, Private และ Protected เข้ามาช่วยกำหนดขอบเขตการใช้

- วัตถุมาใช้ใหม่ (Reuse the Object)

จุดประสงค์ใหญ่ของการเขียนโปรแกรมเชิงวัตถุคือ การนำส่วนต่างๆ ของวัตถุที่สร้างขึ้นกลับมาใช้ใหม่หรือที่เรียกในภาษาอังกฤษว่า “Reuse” เมื่อผู้เขียนโปรแกรมสร้างวัตถุมีจำนวนมากพอก็สามารถนำวัตถุที่สร้างขึ้นมาประกอบเป็นวัตถุใหม่ หรือที่เรียกว่าคอมโพสิตชัน “Composition” นอกจากวิธีการคอมโพสิตแล้วผู้ใช้อย่างยังสามารถ Reuse ส่วนของวัตถุโดยใช้การสืบทอดคุณสมบัติ (Inheritance) จากคลาสลักษณะเช่นนี้คือ เป็นการนำส่วนของวัตถุทั้งหมดมาใช้ซึ่งปกติแล้ววัตถุที่นำมาใช้ในลักษณะนี้จะมีขนาดใหญ่ ถ้าเป็นการคอมโพสิตจะประกอบขึ้นจากส่วนของวัตถุที่มีขนาดเล็กกว่าอย่างไรก็ตามขนาดของวัตถุได้เป็นตัวกำหนดที่แน่นอนตายตัวเสมอไป

- การพ้องรูป (Polymorphism)

รากฐานของการพ้องรูป (Polymorphism) คือ การถ่ายทอดคุณสมบัติ เพราะถ้าไม่มีการถ่ายทอดคุณสมบัติก็จะไม่เกิดสภาวะการพ้องรูป การถ่ายทอดคุณสมบัติเป็นเครื่องมือยืนยันได้ว่าคลาสลูกที่เกิดจากคลาสมแม่เดียวกันย่อมมีคุณสมบัติเหมือนกัน

ประโยชน์ของการโปรแกรมแบบ OOP

เทคโนโลยีของออบเจกต์โอเรียนเตดได้นำประโยชน์ของการพัฒนาซอฟต์แวร์มาให้งดังนี้

1) ความสามารถในการเรียกใช้ได้หลายครั้งออบเจกต์ได้ถูกออกแบบตามหลักการที่ว่าสามารถเรียกใช้งานได้หลายๆ ครั้ง ในหลักการนี้ทำให้ Application ของ OOP ตัวแรกอาจจะทำได้ยาก แต่ว่าโปรแกรมแอปพลิเคชันที่เขียนภายหลังจะสร้างง่ายเพราะสามารถเรียกใช้ออบเจกต์ที่ถูกสร้างไว้ตั้งแต่โครงงานแรกได้

2) ความเชื่อถือได้ โปรแกรมแอปพลิเคชันของ OOP จะมีความเชื่อถือได้สูงเพราะจะรวมเอาส่วนย่อยที่ทดสอบจนได้มาตรฐานแล้วมารวมเข้าไว้ด้วยกัน รหัส (Code) ที่เขียนขึ้นมาใหม่ในแต่ละแอปพลิเคชันจะมีไม่มากนัก เนื่องจากรหัสส่วนใหญ่จะถูกดึงมาจากไลบรารีที่มีความเชื่อถือได้สูงอยู่แล้ว และในการโปรแกรมภาษา C++ ยังมีคุณสมบัติประโยชน์อื่นอีก

3) ความต่อเนื่องกัน การพัฒนาซอฟต์แวร์แบบ OOP ใน C++ จะเปลี่ยนไปตามฝีมือและจำนวนนักเขียนโปรแกรมภาษา C นักโปรแกรมภาษา C ที่ชำนาญสามารถเรียนรู้หลักการของ OOP ได้ภายในเวลาไม่นาน และสามารถเข้าใจเนื้อหาได้ไม่ยากอีกทั้งสามารถแปลงโปรแกรมแอปพลิเคชันของ C เป็น C++ ได้

2.3.1.2 พีเอชพี (Personal Home Page : PHP)

พีเอชพี (PHP) คือ ภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ โดยลิขสิทธิ์อยู่ในลักษณะโอเพนซอร์ส ภาษาพีเอชพีใช้สำหรับจัดทำเว็บไซต์ และแสดงผลออกมาในรูปแบบ HTML โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล ซึ่ง ภาษาพีเอชพี นั้นง่ายต่อการเรียนรู้ ซึ่งเป้าหมายหลักของภาษานี้ คือให้นักพัฒนาเว็บไซต์สามารถเขียน เว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว ภาษาพีเอชพี ในชื่อภาษาอังกฤษว่า PHP ซึ่งใช้เป็นคำย่อแบบกล่าวซ้ำ จาก คำว่า PHP Hypertext Preprocessor หรือ ชื่อ เต็ม Personal Home Page พีเอชพีรุ่นล่าสุดคือ PHP 5.4.0 ส่วนรุ่นพัฒนาคือ PHP 6.0.0

ประวัติความเป็นมาของ PHP

PHP นั้นถูกคิดค้นขึ้นมาในปี ค.ศ.1994 โดยนาย Rasmus Lerdorf แต่เป็นเวอร์ชันที่ไม่เป็นทางการหรือรุ่นทดลองนั่นเอง ซึ่งเวอร์ชันนี้ได้มีการทดสอบกับเครื่องของเขาเอง โดยใช้ตรวจสอบติดตามเก็บสถิติข้อมูล ผู้ที่เข้าเยี่ยมชมประวัติส่วนตัวบนเว็บเพจของเขาเท่านั้น ต่อมา PHP เวอร์ชันแรกได้ถูกพัฒนาและเผยแพร่ให้กับผู้อื่นที่ต้องการใช้ศึกษา ในปี ค.ศ.1995 ซึ่งถูกเรียกว่า "Hypertext Preprocessor" ซึ่งเป็นที่มาของคำว่า PHP นั่นเอง ซึ่งในระยเวลานั้น PHP ยังไม่มีความสามารถอะไรที่โดดเด่นมากมาย จนกระทั่งเมื่อประมาณกลางปี ค.ศ.1995 นาย Rasmus ได้คิดค้นและพัฒนาให้ PHP/PI หรือ PHP เวอร์ชัน 2 ให้มีความสามารถจัดการเกี่ยวกับแบบฟอร์มข้อมูลที่ถูกสร้างมาจากภาษา HTML และสนับสนุนการติดต่อกับโปรแกรมจัดการฐานข้อมูล MYSQL จึงทำให้ PHP เริ่มถูกใช้มากขึ้นอย่างรวดเร็ว และเริ่มมีผู้สนับสนุนการใช้งาน PHP มากขึ้น โดยในปลายปี ค.ศ.1996 PHP ถูกนำไปใช้ประมาณ 15,000 เว็บทั่วโลก และเพิ่มจำนวนขึ้นเรื่อยๆ นอกจากนี้ในราวกลางปี ค.ศ.1997 PHP ได้มีการเปลี่ยนแปลงและพัฒนาจากเจ้าของเดิมคือ นาย Rasmus ที่พัฒนาอยู่เพียงผู้เดียว มาเป็นทีมงาน โดยมีนาย Zeev Suraski และนาย Adni Gutmans ทำการวิเคราะห์พื้นฐานของ PHP/FI และได้้นำโค้ดมาพัฒนาให้เป็น PHP เวอร์ชัน 3 ซึ่งมีความสามารถที่มีความสมบูรณ์มากขึ้น ในราวกลางปี ค.ศ.1999 PHP เวอร์ชัน 3 ได้ถูกพัฒนาจนสามารถทำงานร่วมกับ C2's StrongHold Web Server และ Red Hat Linux ได้

คุณสมบัติของภาษา PHP

การแสดงผลของพีเอชพี จะปรากฏในลักษณะ HTML ซึ่งจะไม่แสดงคำสั่งที่ผู้ใช้เขียน ซึ่งเป็นลักษณะเด่นที่พีเอชพีแตกต่างจากภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ เช่น ภาษาจาวาสคริปต์ ที่ผู้ชมเว็บไซต์สามารถอ่าน ดูและคัดลอกคำสั่งไปใช้เองได้ นอกจากนี้พีเอชพียังเป็นภาษาที่เรียนรู้และเริ่มต้นได้ไม่ยาก โดยมีเครื่องมือช่วยเหลือ และคู่มือที่สามารถหาอ่านได้ฟรีบนอินเทอร์เน็ต ความสามารถในการประมวลผลหลักของพีเอชพี ได้แก่ การสร้างเนื้อหาอัตโนมัติจัดการคำสั่ง การอ่านข้อมูลจากผู้ใช้และประมวลผล การอ่านข้อมูลจากดาต้าเบส ความสามารถจัดการกับคุกกี้ ซึ่งทำงานเช่นเดียวกับโปรแกรมในลักษณะ CGI คุณสมบัติอื่น เช่น การประมวลผลตามบรรทัดคำสั่ง (Command Line Scripting) ทำให้ผู้เขียนโปรแกรมสร้างสคริปต์พีเอชพี ทำงานผ่านพีเอชพี พาร์เซอร์ (PHP Parser) โดยไม่ต้องผ่านเชิร์ฟเวอร์หรือเบราร์เซอร์ ซึ่งมีลักษณะเหมือนกับ Cron (ในยูนิกซ์หรือลินุกซ์) หรือ Task Scheduler (ในวินโดวส์) สคริปต์เหล่านี้สามารถนำไปใช้ในแบบ Simple Text Processing Tasks ได้

การแสดงผลของพีเอชพี ถึงแม้ว่าจุดประสงค์หลักใช้ในการแสดงผล HTML แต่ยังสามารถสร้าง XHTML หรือ XML ได้ นอกจากนี้สามารถทำงานร่วมกับคำสั่งเสริมต่างๆ ซึ่งสามารถแสดงผลข้อมูลหลัก PDF แฟลช (โดยใช้ Libswf และ Ming) พีเอชพีมีความสามารถอย่างมากในการทำงานเป็นประมวลผลข้อความ จาก POSIX Extended หรือ รูปแบบ Perl ทั่วไป เพื่อแปลงเป็นเอกสาร XML ในการแปลงและเข้าสู่เอกสาร XML เรารองรับมาตรฐาน SAX และ DOM สามารถใช้รูปแบบ XSLT ของเราเพื่อแปลงเอกสาร XML เมื่อใช้พีเอชพีในการทำอีคอมเมิร์ซ สามารถทำงานร่วมกับโปรแกรมอื่น เช่น Cybercash payment, CyberMUT, VeriSign Payflow Pro และ CCVS Functions เพื่อใช้ในการสร้างโปรแกรมทำธุรกรรมทางการเงิน

การรองรับภาษา PHP

คำสั่งของพีเอชพี สามารถสร้างผ่านทางโปรแกรมแก้ไขข้อความทั่วไป เช่น โน้ตแพด หรือ Vi ซึ่งทำให้การทำงานพีเอชพี สามารถทำงานได้ในระบบปฏิบัติการหลักเกือบทั้งหมด โดยเมื่อเขียนคำสั่งแล้วนำมาประมวลผล Apache, Microsoft Internet Information Services (IIS), Personal Web Server, Netscape และ iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, และอื่นๆ อีกมากมาย สำหรับส่วนหลักของ PHP ยังมี Module ในการรองรับ CGI มาตรฐาน ซึ่ง PHP สามารถทำงานเป็นตัวประมวลผล CGI ด้วย และด้วย PHP มีอิสรภาพในการเลือก ระบบปฏิบัติการ และ เว็บเซิร์ฟเวอร์ นอกจากนี้คุณยังสามารถใช้สร้างโปรแกรมโครงสร้าง สร้างโปรแกรมเชิงวัตถุ (OOP) หรือสร้างโปรแกรมที่รวมทั้งสองอย่างเข้าด้วยกัน แม้ว่า

ความสามารถของคำสั่ง OOP มาตรฐานในเวอร์ชันนี้ยังไม่สมบูรณ์ แต่ตัวไลบรารีทั้งหลายของโปรแกรม และตัวโปรแกรมประยุกต์ (รวมถึง PEAR Library) ได้ถูกเขียนขึ้นโดยใช้รูปแบบการเขียนแบบ OOP เท่านั้น

พีเอชพีสามารถทำงานร่วมกับฐานข้อมูลได้หลายชนิด ซึ่งฐานข้อมูลส่วนหนึ่งที่รองรับได้แก่ ออราเคิล, dBase, PostgreSQL, IBM, DB2, MySQL, Informix, ODBC โครงสร้างของฐานข้อมูลแบบ DBX ซึ่งทำให้พีเอชพีใช้กับฐานข้อมูลอะไรก็ได้ที่รองรับรูปแบบนี้ และ PHP ยังรองรับ ODBC (Open Database Connection) ซึ่งเป็นมาตรฐานการเชื่อมต่อฐานข้อมูลที่ใช้กันแพร่หลายอีกด้วย คุณสามารถเชื่อมต่อกับฐานข้อมูลต่างๆ ที่รองรับมาตรฐานโลกนี้ได้ พีเอชพียังสามารถรองรับการสื่อสารกับการบริการในโพรโทคอลต่างๆ เช่น LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM(บนวินโดวส์) และอื่นๆ อีกมากมาย คุณสามารถเปิด Socket บนเครือข่ายโดยตรงและตอบโต้โดยใช้ โพรโทคอลใดๆ ก็ได้ PHP มีการรองรับสำหรับการแลกเปลี่ยนข้อมูลแบบ WDDX Complex กับ Web Programming อื่นๆ ทั่วไปได้พูดถึงในส่วน Interconnection, พีเอชพีมีการรองรับสำหรับ Java objects ให้เปลี่ยนมันเป็น PHP Object แล้วใช้งาน คุณยังสามารถใช้ รูปแบบ CORBA เพื่อเข้าสู่ Remote Object ได้

2.3.1.3 ซีเอสเอส (Cascading Style Sheet : CSS)

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสารไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลล์พ์ของเอกสาร 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

ประโยชน์ของ CSS

1. CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความ รวมทั้งสี รูปแบบของข้อความที่กล่าวมาแล้ว
2. CSS นั้นกำหนดที่ต้นของไฟล์ html หรือตำแหน่งอื่น ๆ ก็ได้ และสามารถมีผลกับเอกสารทั้งหมด หมายถึงกำหนด ครั้งเดียวจุดเดียวก็มีผลกับการแสดงผลทั้งหมด ทำให้เวลาแก้ไขหรือปรับปรุงทำได้สะดวก ไม่ต้องไล่ตามแก้ tag ต่างๆ ทั่วทั้งเอกสาร
3. CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้ร่วมกับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด

2.3.1.4 เจควียรี (jQuery)

jQuery นั้นเป็น JavaScript Library ที่บรรจุเอา Function และคำสั่งต่างๆ ที่จะทำให้ นักเขียนโปรแกรม ไม่ต้องมาเขียนเองใหม่ทั้งหมดตั้งแต่ต้น สามารถที่จะเขียน Ajax ได้แบบง่ายๆ เพียง คำสั่งไม่กี่บรรทัด หรือจะเขียน JavaScript เพื่อดักเหตุการณ์ต่างๆ ที่ต้องการ เช่น การ click, rollover, mouse ซึ่ง jQuery เปิดตัวครั้งแรกในงานบาร์แคมป์นิวยอร์ก โดย จอห์น เรซิก (John Resig) เมื่อ 14 มกราคม พ.ศ. 2549 ตัวโค้ดของเจควียรีมีลิขสิทธิ์และสัญญาอนุญาตแบบโอเพ่นซอร์ส โดยใช้สัญญาอนุญาตของ GFDL และ MIT License

ความสามารถของ jQuery

- ใช้งาน DOM element โดยการเขียนไม่จำเป็นต้องยึดติดกับ Browser
- จัดการกับ Event เพิ่ม Event ลงในหน้าเว็บตามแต่ใจต้องการ
- การจัดการ CSS สามารถเพิ่ม ลบ class ,id ของ CSS
- Effects and animations ลูกเล่นบนหน้าเว็บ
- ช่วยให้เขียน Ajax ได้ง่ายขึ้นประหยัดเวลาในการเขียน Ajax และทำให้โค้ดสั้นลง
- สามารถสืบทอดคุณสมบัติของ jQuery ไปใช้งานได้

กลุ่มคนที่จะนำ jQuery ไปใช้งาน

- Web Programmer
- Web design ใช้งานกับ html, CSS ได้เป็นอย่างดี รวมทั้ง xml ด้วย

2.3.1.5 จาวาสคริปต์ (JavaScript)

JavaScript เป็นภาษาโปรแกรม (programming language) ประเภทหนึ่งที่ถูกเรียกว่า "สคริปต์" (script) ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินการไปทีละคำสั่ง" (interpret) ภาษานี้เดิมมีชื่อว่า LiveScript ได้รับการพัฒนาขึ้นโดย Netscape ด้วยวัตถุประสงค์ เพื่อที่จะช่วยให้เว็บเพจสามารถแสดงเนื้อหา ที่มีการเปลี่ยนแปลงไปได้ ตามเงื่อนไขหรือสภาพแวดล้อมต่างๆกัน หรือสามารถโต้ตอบกับผู้ชมได้มากขึ้น ทั้งนี้เพราะภาษา HTML แต่เดิมนั้นเหมาะสำหรับใช้แสดงเอกสารที่มีเนื้อหาคงที่แน่นอน และไม่มีลูกเล่นอะไรมากมายนัก

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA ซึ่งเราจะพบว่าปัจจุบันจะหาเว็บเพจที่ไม่ใช้ JavaScript เลยนั้น ได้ยากเต็มที

การทำงานของ JavaScript จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ตามสิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชันใหม่ๆออกมาด้วย (ปัจจุบันคือรุ่น 1.5) ดังนั้น ถ้านำคำสั่งของเวอร์ชันใหม่ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุนก็อาจจะทำให้เกิดข้อผิดพลาดได้

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าคุณจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านั้นเท่านั้น อย่างไรก็ตามดีจากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัดคือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชมเพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้จึงยังคงต้องอาศัยภาษา server-side script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์ก็มีซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุนโดยเฉพาะเช่นกัน)

การทำงานของ JavaScript จะมีประสิทธิภาพมาก ถ้าสามารถดัดแปลงคุณสมบัติขององค์ประกอบต่างๆ บนเว็บเพจ (เช่น สี หรือรูปแบบของข้อความ) และสามารถรับรู้เหตุการณ์ที่ผู้ชมเว็บเพจโต้ตอบกับองค์ประกอบเหล่านั้น (เช่น การคลิก หรือเลื่อนเมาส์ไปวาง) ได้ ดังนั้นจากภาษา HTML เดิม ที่มีลักษณะสถิต (static) ใน HTML เวอร์ชันใหม่ๆ จึงได้มีการพัฒนาให้มีคุณสมบัติบางอย่างเพิ่มขึ้น และมีลักษณะเป็นอ็อบเจ็ค "object" มากขึ้น การทำงานร่วมกันระหว่างคุณสมบัติ

ใหม่ของ HTML ร่วมกับ JavaScript นี้เอง ทำให้เกิดเป็นสิ่งที่เรียกว่า Dynamic HTML คือภาษา HTML ที่สามารถใช้สร้างเว็บเพจที่มีลักษณะพลวัต (dynamic) ได้นั่นเอง

นอกจากนี้ อีกองค์ประกอบหนึ่งที่เกี่ยวข้องก็คือ Cascading Style Sheet (CSS) ซึ่งเป็นภาษาที่ช่วยให้เราควบคุมรูปแบบขององค์ประกอบต่างๆ บนเว็บเพจได้อย่างมีประสิทธิภาพมากกว่าคำสั่งหรือแท็ก (tag) ปกติของ HTML เนื่องจาก JavaScript สามารถดัดแปลงคุณสมบัติของ CSS ได้เช่นกัน ดังนั้นมันจึงช่วยให้เราควบคุมเว็บเพจได้อย่างมีประสิทธิภาพมากยิ่งขึ้นไปอีก

ความสามารถของ JavaScript

- 1) JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่ายๆ ได้ โดยไม่ต้องพึ่งภาษาอื่น เช่น PHP เน้นว่าแบบง่ายๆ โดยส่วนใหญ่จะเป็นรูปแบบของการแสดงผลมากกว่า
- 2) JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่นเมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าต่างใหม่ได้ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น นี่คือข้อดีของ JavaScript เลยก็ว่าได้ที่ทำให้เว็บไซต์ดังๆทั้งหลายเช่น Google Map ต่างหันมาใช้
- 3) JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้นั่นเอง สังเกตจากเมนูต่างๆใน Hellomyweb.com สามารถเลื่อนขึ้นลงได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่ายๆนั่นเอง
- 4) JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อกรอกข้อมูลบางเว็บไซต์ เช่น Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างฟ้องขึ้นมาว่าเรากรอกผิด หรือลิ้มกรอกอะไรบางอย่าง ส่วนใหญ่เกือบทั้งหมดใช้ JavaScript ตรวจสอบ
- 5) JavaScript สามารถใช้ตรวจสอบผู้ใช้ได้เช่น ตรวจสอบว่าผู้ใช้ Web browser อะไร
- 6) JavaScript สร้าง Cookies (เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เอง) ได้

การใช้งาน JavaScript

เราจะใช้ JavaScript เมื่อต้องการทำให้เว็บเพจแสดงผลแบบเคลื่อนไหวได้มีชีวิตชีวา ไม่แข็งทื่อ หรือใช้ในการสั่งให้เบราว์เซอร์ทำงานตามที่เราต้องการ โดยงานนั้นอาจเกินความสามารถของภาษา HTML แต่ไม่ใช่งานที่ต้องติดต่อกับ Server นั่นเพราะ JavaScript เป็นภาษาที่ใช้ในการเขียนโปรแกรมทางฝั่ง client โดยมีเบราว์เซอร์เป็นตัวแปลภาษานั้นเอง ตัวอย่างของงานที่จะต้องใช้ภาษา JavaScript เช่น

- 1) สร้างเมนูที่สามารถตอบสนองต่อการเอา mouse ไปชี้ได้
- 2) สร้างเครื่องคิดเลข ปฏิทิน เกม ในเว็บเพจ
- 3) เปิด,ปิด,เคลื่อนย้ายตำแหน่งของ window
- 4) ทำตัวอักษรวิ่งแบบต่างๆ
- 5) ตรวจสอบความถูกต้องของแบบฟอร์มที่ผู้ใช้กรอกก่อนส่งข้อมูลไปยัง Server
- 6) อื่นๆ อีกมากมายที่ไม่ต้องติดต่อกับ Server

2.3.1.6 เอสคิวแอล (Structured Query Language : SQL)

SQL เป็นภาษาที่ใช้ในการเรียกใช้ฐานข้อมูล ซึ่งย่อมาจาก "structured query language" โดยที่ SQL เป็นภาษาที่มีมาตรฐานและเป็นระบบเปิด (open system) หมายถึงเราสามารถใส่คำสั่ง SQL กับฐานข้อมูลชนิดใดก็ได้ และคำสั่งงานเดียวกันเมื่อสั่งงานผ่านระบบฐานข้อมูลที่แตกต่างกันจะได้ผลลัพธ์เหมือนกันทำให้เราสามารถเลือกใช้ฐานข้อมูลชนิดใดก็ได้โดยไม่ติดขัดกับฐานข้อมูลใดฐานข้อมูลหนึ่ง การทำงานของโปรแกรมฐานข้อมูลบน server เมื่อเราใช้ SQL โปรแกรมจะทำงานได้เร็วกว่าการใช้ table ความสามารถของภาษา SQL ยังมีมากกว่าการจัดการ table เราสามารถใช้ SQL ทำงานอื่นๆ ที่นอกเหนือจากความสามารถของ Delphi เช่นการใช้ stored procedure เพื่อให้การทำงานกับฐานข้อมูลเกิดขึ้นภายใน server เท่านั้น ไม่ต้องส่งข้อมูลกลับไปมา กับเครื่อง client ซึ่งจะมีผลให้การทำงานเร็วขึ้น

ภาษา SQL จะมีคำสั่งหลักๆ อยู่ 2 ประเภท คือ

- 1) DDL - Data Definition Language ใช้ในการจัดการโครงสร้างของ table, view และ index ได้แก่ คำสั่ง create, alter เป็นต้น
- 2) DML - Data Manipulation Language ใช้ในการปรับปรุงข้อมูลใน table เช่น
 - select ใช้ในการอ่านข้อมูลจาก table ตามเงื่อนไขที่ต้องการ
 - insert ใช้ในการแทรกแถวข้อมูลลงไปใน table
 - update ใช้ในการปรับปรุงข้อมูลใน table
 - delete ใช้ในการลบข้อมูลใน table

รูปแบบการใช้คำสั่ง SQL สามารถใช้ได้เป็น 2 รูปแบบ ดังนี้ คือ

- 1) คำสั่ง SQL ที่ใช้เรียกดูข้อมูลได้ทันที (Interactive SQL) เป็นการเรียกใช้คำสั่ง SQL สั่งงานบนจอภาพ เพื่อเรียกดูข้อมูลในขณะที่ทำงานได้ทันที เช่น `SELECT CITY FROM SUPPLIER WHERE SNO = ?SE?;`
- 2) คำสั่ง SQL ที่ใช้เขียนร่วมกับโปรแกรมอื่นๆ (Embedded SQL) เป็นคำสั่ง SQL ที่ใช้ร่วมกับคำสั่งของโปรแกรมภาษาต่างๆ เช่น PL/1 PASCAL ฯลฯ หรือแม้แต่ว่ากับคำสั่งในโปรแกรมที่ระบบจัดการฐานข้อมูลนั้นมีใช้เฉพาะ เช่น ORACLE มี PL/SQL (Procedural Language /SQL) ที่สามารถเขียนโปรแกรมและนำคำสั่ง SQL มาเขียนร่วมด้วย เป็นต้น

Data Definition Language (DDL)

เป็นภาษาที่ใช้นิยามโครงสร้างข้อมูล เพื่อเปลี่ยนแปลง หรือยกเลิกโครงสร้างฐานข้อมูลตามที่ ออกแบบไว้โครงสร้างดังกล่าวคือ สคีม่า (Schema) นั่นเอง

ภาษา DDL ประกอบด้วย 3 คำสั่ง คือ

- 1) คำสั่งการสร้าง (Create) ได้แก่ การสร้างตาราง
`CREATE TABLE <Table name>`
- 2) คำสั่งเปลี่ยนแปลงโครงสร้าง
`ALTER TABLE < ชื่อตารางที่ตั้งขึ้น >`
<คำสั่งการเปลี่ยนแปลง> (<ชื่อคอลัมน์ ประเภทข้อมูล>);
- 3) คำสั่งยกเลิก (Drop) ต่างๆ
การลบโครงสร้างตาราง `DROP TABLE < ชื่อตารางที่ตั้งขึ้น >`

คำสั่ง	ความหมาย
CREATE TABLE	นิยามโครงสร้างข้อมูลในรูปตารางบนฐานข้อมูล
DROP TABLE	ลบโครงสร้างตารางข้อมูลออกจากระบบ
ALTER TABLE	แก้ไขปรับปรุงโครงสร้างตาราง

ตารางที่ 2.4 ตารางคำสั่งยกเลิก (Drop)

ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language-DML)

หลังจากที่เราสร้างโครงสร้างฐานข้อมูลขึ้นแล้วคำสั่งต่อไปในการป้อนข้อมูลลงในฐานข้อมูล และเปลี่ยนแปลงข้อมูลในฐานข้อมูล โดยใช้ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language-DML) ใช้จัดการข้อมูลภายในตารางภายในฐานข้อมูล และภาษาแก้ไขเปลี่ยนแปลงตาราง แบ่งออกเป็น 4 Statement คือ

- 1) Insert Statement : การเพิ่มเติมข้อมูลลงในตาราง (Table) จากฐานข้อมูล
- 2) Select Statement : การเรียกหา (Retrieve) ข้อมูลจากฐานข้อมูล
- 3) Delete Statement: การลบข้อมูลออกจาก ตาราง (Table) จากฐานข้อมูล
- 4) Update Statement: การเปลี่ยนแปลงข้อมูลลงใน ตาราง (Table) จากฐานข้อมูล

คำสั่ง	ความหมาย
SELECT	เรียกค้นข้อมูลในตาราง
INSERT	เพิ่มแถวข้อมูลลงในตาราง
DELETE	ลบแถวข้อมูล
UPDATE	ปรับปรุงแถวข้อมูลในตาราง

ตารางที่ 2.5 ตารางภาษาสำหรับการจัดการข้อมูล

ภาษาควบคุม (Control Language)

ใช้เป็นภาษาที่ใช้ควบคุมระบบรักษาความปลอดภัยของฐานข้อมูล ประกอบด้วย คำสั่ง 2 คำสั่งคือ

- 1) คำสั่ง GRANT เป็นคำสั่งที่ใช้กำหนดสิทธิให้กับผู้ใช้แต่ละคนให้มีสิทธิกระทำการใดกับข้อมูล
 - 2) คำสั่ง REVOKE เป็นคำสั่งให้มีการยกเลิกสิทธินั้นหลังจากที่ได้ GRANT แล้ว
- ค่าบูลีน (Boolean Type Data)

ชุดคำสั่ง GROUP BY, ORDER BY และ HAVING

เนื่องจากข้อมูลที่น่าสนใจมักจะเป็นผลสรุป หรือข้อมูลที่ใช้ในการตัดสินใจภาษา SQL จึงได้ออกแบบให้มีชุดคำสั่งพิเศษที่ทำหน้าที่แบ่งออกเป็นกลุ่มๆ ได้แก่

1) ชุดคำสั่ง GROUP BY

GROUP BY เป็นคำสั่งให้มีการจัดกลุ่มแถวข้อมูลตามคอลัมน์ โดยข้อมูลที่เหมือนกันจะถูกจัดให้อยู่ในกลุ่มเดียวกัน

2) ชุดคำสั่ง ORDER BY

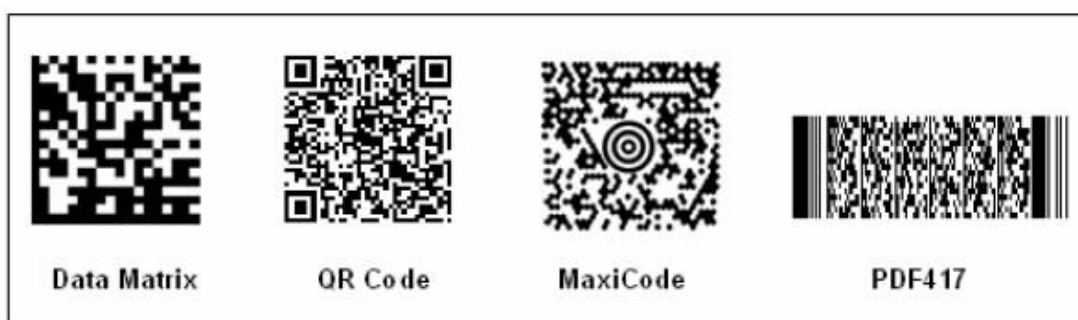
ORDER BY เป็นคำสั่งให้มีการเรียงลำดับข้อมูลในแถวข้อมูลตามคอลัมน์ พิจารณิตเชิงสัมพันธ์และแคลคูลัสเชิงสัมพันธ์ โดยข้อมูลที่เหมือนกันจะถูกจัดเรียงจากน้อยไปมาก ถ้าไม่การระบุแต่ถ้าระบุว่าเป็น DESC จะเรียงจากมากไปน้อย ดังนั้นจากตัวอย่างต่อไปนี้

3) ชุดคำสั่ง HAVING

HAVING เหมือนกับคำสั่ง WHERE ที่ต้องตามด้วยเงื่อนไข แต่ HAVING จะใช้ในกรณีที่มีการจัดกลุ่มหรือการใช้ GROUP BY

2.3.1.7 บาร์โค้ด 2 มิติ (2 Dimension Barcode)

บาร์โค้ด 2 มิติเป็นเทคโนโลยีที่พัฒนาเพิ่มเติมจากบาร์โค้ด 1 มิติ โดยออกแบบให้บรรจุได้ทั้งในแนวดิ่งและแนวนอน ทำให้สามารถบรรจุข้อมูลมากได้ประมาณ 4,000 ตัวอักษรหรือประมาณ 200 เท่าของบาร์โค้ด 1 มิติในพื้นที่เท่ากันหรือเล็กกว่า ข้อมูลที่บรรจุสามารถใช้ภาษาอื่นนอกจากภาษาอังกฤษได้ เช่น ภาษาญี่ปุ่น จีน หรือเกาหลี เป็นต้นและบาร์โค้ด 2 มิติสามารถถอดรหัสได้แม้ภาพบาร์โค้ดบางส่วนมีการเสียหาย อุปกรณ์ที่ใช้อ่านและถอดรหัสบาร์โค้ด 2 มิติมีตั้งแต่เครื่องอ่านแบบซีซีดีหรือเครื่องอ่านแบบเลเซอร์เหมือนกับของบาร์โค้ด 1 มิติจนถึงโทรศัพท์มือถือแบบมีกล้องถ่ายรูปในตัวซึ่งติดตั้งโปรแกรมถอดรหัสไว้ ในส่วนลักษณะของบาร์โค้ด 2 มิติมีอยู่อย่างมากมายตามชนิดของบาร์โค้ด เช่น วงกลม สี่เหลี่ยมจัตุรัส หรือสี่เหลี่ยมผืนผ้าคล้ายกับบาร์โค้ด 1 มิติ เป็นต้น ตัวอย่างบาร์โค้ด 2 มิติ ได้แก่ PD417, MaxiCode, Data Matrix, และ QR Code

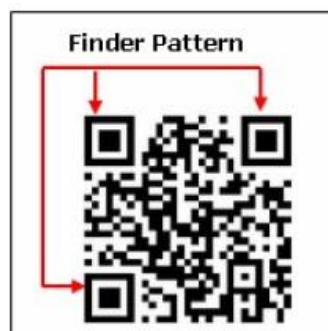


ภาพที่ 2.2 ภาพบาร์โค้ด 2 มิติ (2 Dimension Barcode)

บาร์โค้ด 2 มิติ		PDF417	MaxiCode	Data Matrix	QR Code
ผู้พัฒนา (ประเทศ)		Symbol Technologies (สหรัฐอเมริกา)	Onipolar (สหรัฐอเมริกา)	RVSI Acuity CiMatrix	DENSO
ประเภทบาร์โค้ด		แบบสแต็ก	แบบเมตริกซ์	แบบเมตริกซ์	แบบเมตริกซ์
ขนาด ความ จุ ข้อมูล	ตัวเลข	2, 710	138	3,116	7,089
	ตัวอักษร	1,850	93	2,355	4,296
	เลขฐานสอง	1,1018	-	1,556	2,953
	ตัวอักษร ญี่ปุ่น	554	-	778	1,817
ลักษณะที่สำคัญ		- บรรจุข้อมูลได้มาก	- มีความเร็วในการอ่านสูง	- บาร์โค้ดมีขนาดเล็ก	- บาร์โค้ดมีขนาดเล็ก - มีความเร็วในการอ่านสูง - บรรจุข้อมูลได้มาก
มาตรฐานที่ได้รับ		-ISO/IEC 15438 - AIM USS- PDF417	-ISO/IEC 16023 - ANSI/AIMBC10- ISS-MaxiCode	- SIO/IEC 16022 ANSI/AIM BC11-ISS-Data Maxtix	- SIO/IEC 18004 - JIS X 0510 JEUDA-55 - AIM ITS/97/001 ISS-QR Code

ตารางที่ 2.6 แสดงการเปรียบเทียบบาร์โค้ด 2 มิติชนิดต่างๆ

บาร์โค้ดแบบ QR Code (Quick Response Code)



ภาพที่ 2.3 ลักษณะของบาร์โค้ดแบบ QR Code (Quick Response Code)

QR Code เป็นบาร์โค้ด 2 มิติแบบเมตริกซ์ที่ถูกพัฒนาขึ้นโดยบริษัท Nippon Denso ประเทศญี่ปุ่นในปี 2537 สอดคล้องกับมาตรฐาน ISO/IEC 18004, JIS X 0510, JEIDA-55 และ AIM ITS/97/001 ISS-QR Code ลักษณะของบาร์โค้ดเป็นรูปสี่เหลี่ยมจัตุรัส มีโมดูลข้อมูล 21x21 ถึง 177x177 โมดูล สามารถบรรจุข้อมูลได้มากที่สุด 7,089 ตัวเลขหรือ 4,296 ตัวอักษร ข้อมูลเลขฐานสอง 2,953 ไบต์ และตัวอักษรญี่ปุ่น 1,817 ตัวอักษร รูปแบบค้นหาของ QR Code อยู่ที่มุมทั้งสามของบาร์โค้ด คือ มุมซ้ายบน มุมซ้ายล่าง และมุมขวาบน QR Code ส่วนใหญ่ใช้ในงานที่ต้องการบรรจุข้อมูลจำนวนมากลงในบาร์โค้ดและต้องการอ่านข้อมูลจากบาร์โค้ดอย่างรวดเร็ว

การนำเทคโนโลยีบาร์โค้ด 2 มิติมาใช้งาน

ตัวอย่างการใช้งานบาร์โค้ดแบบ QR Code

- ด้านอุตสาหกรรมยานยนต์ มีการติดบาร์โค้ดบนชิ้นส่วนอะไหล่ยนต์ต่างๆ เพื่อเก็บข้อมูลของอะไหล่ชิ้นนั้น เช่น ชื่อรุ่น รหัสอะไหล่ และประเภทของอะไหล่ เป็นต้น
- ด้านกระบวนการผลิตสินค้ามีการติดบาร์โค้ด 2 มิติบนแผงวงจรอิเล็กทรอนิกส์ เก็บข้อมูลรายละเอียดต่างๆ ของแผงวงจรนั้น



ภาพที่ 2.4 แสดงการนำไปใช้ในกระบวนการผลิตสินค้า ในประเทศญี่ปุ่น

- ด้านการจัดสต็อกสินค้า เพื่อตรวจสอบชื่อรุ่นของสินค้า รหัสสินค้า และจำนวนของสินค้าคงเหลือ



ภาพที่ 2.5 แสดงการนำไปใช้ด้านการจัดการสต็อกสินค้า ในประเทศญี่ปุ่น

2.3.2 โปรแกรมประยุกต์ที่ใช้ในการพัฒนา

2.3.2.1 เว็บเบราว์เซอร์ (Web Browser)

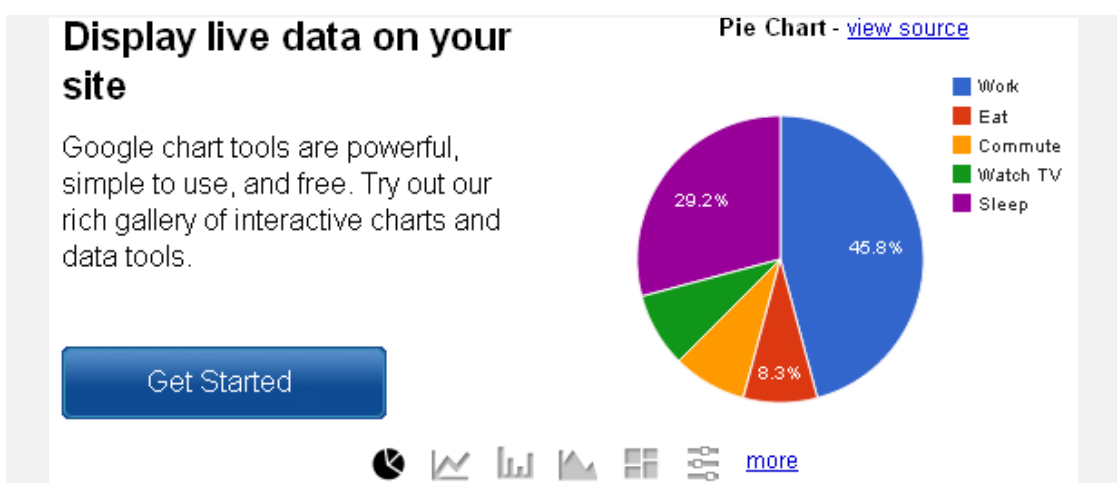
เว็บเบราว์เซอร์ (web browser) เบราว์เซอร์ หรือ โปรแกรมดูเว็บ คือ โปรแกรมคอมพิวเตอร์ที่ผู้ใช้สามารถดูข้อมูลและโต้ตอบกับข้อมูลสารสนเทศที่จัดเก็บในหน้าเว็บที่ สร้างด้วยภาษาเฉพาะ เช่น ภาษาเอชทีเอ็มแอล (html) ที่จัดเก็บไว้ที่ระบบบริการเว็บหรือเว็บเซิร์ฟเวอร์ หรือระบบคลังข้อมูลอื่นๆ โดยโปรแกรมค้นดูเว็บเปรียบเสมือนเครื่องมือในการติดต่อกับเครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่เรียกว่าเวิลด์ไวด์เว็บ

ประโยชน์ของเว็บเบราว์เซอร์

สามารถดูเอกสารภายในเว็บเซิร์ฟเวอร์ได้ อย่างสวยงามมีการแสดงข้อมูลในรูปของข้อความ ภาพ และระบบมัลติมีเดียต่างๆ ทำให้การดูเอกสารบนเว็บมีความน่าสนใจมากขึ้น ส่งผลให้อินเทอร์เน็ตได้รับความนิยมเป็นอย่างมากเช่นในปัจจุบัน ปัจจุบัน web browser ส่วนใหญ่จะรองรับ html 5 และ อ่าน CSS เพื่อความสวยงามของหน้า web page

2.3.2.2 การสร้างบาร์โค้ด 2 มิติ(Google Chart Tools)

Google Chart Tools ตัวช่วยที่ดีในการสร้างกราฟหรือชาร์ตรูปแบบต่างๆ ซึ่งก็มีให้เลือกอยู่มากมาย เช่น Line charts , Bar charts, Pie charts, Maps, Google-o-meters, QR codes และรูปแบบอื่น ๆ



ภาพที่ 2.6 แสดงตัวอย่างหน้าเว็บ Google Chart Tools

2.4 งานวิจัยหรือผลงานที่เกี่ยวข้อง

2.4.1 คู่มือการระเบียบสำนักนายกรัฐมนตรีว่าด้วยการพัสดุ พ.ศ. ๒๕๓๕ และที่แก้ไขเพิ่มเติม

คำจำกัดความ

พัสดุ หมายถึง วัสดุ ครุภัณฑ์ ที่ดินและสิ่งก่อสร้าง ที่กำหนดไว้ในหนังสือการจำแนกประเภทรายจ่ายตามงบประมาณของสำนักงบประมาณ

วัสดุ หมายถึง สินทรัพย์ที่หน่วยงานมีไว้เพื่อใช้ในการดำเนินงานตามปกติ โดยทั่วไปมีมูลค่าไม่สูงและไม่มีลักษณะคงทนถาวร ได้แก่ วัสดุสำนักงาน เป็นต้น หน่วยงานจะรับรู้วัสดุคงเหลือในราคาทุน ซึ่งโดยหลักการหน่วยงานอาจรับรู้วัสดุเป็นสินทรัพย์หรือค่าใช้จ่ายก่อน แล้วจึงปรับปรุงสิ้นปีก็ได้

ครุภัณฑ์ หมายถึง สินทรัพย์ที่หน่วยงานมีไว้เพื่อใช้ในการดำเนินงานมีลักษณะคงทน และมีอายุการใช้งานเกินกว่า 1 ปี และมีมูลค่าตั้งแต่ 5,000 บาท ขึ้นไป

ครุภัณฑ์มูลค่าต่ำกว่าเกณฑ์ หมายถึง วัสดุที่มีลักษณะคงทนถาวร และมีอายุการใช้งานเกินกว่า 1 ปีมีราคาต่อหน่วยหรือต่อชุดไม่เกิน 5,000 บาท

หัวหน้าเจ้าหน้าที่พัสดุ หมายถึง หัวหน้าหน่วยงานระดับกองหรือที่มีฐานะเทียบกอง ซึ่งปฏิบัติงานในสายงานที่เกี่ยวข้องกับการพัสดุตามท้องที่การบริหารงานบุคคลกำหนด หรือข้าราชการอื่นซึ่งได้รับแต่งตั้งจากหัวหน้าส่วนราชการให้เป็นหัวหน้าเจ้าหน้าที่พัสดุ แล้วแต่กรณี

หัวหน้าหน่วยพัสดุ มีหน้าที่ควบคุมและสั่งจ่ายพัสดุตามระเบียบสำนักนายกรัฐมนตรีว่าด้วยการพัสดุ พ.ศ. 2535 และที่แก้ไขเพิ่มเติม ข้อ 153 วรรคสาม

เจ้าหน้าที่พัสดุ หมายถึง เจ้าหน้าที่ซึ่งดำรงตำแหน่งที่มีหน้าที่เกี่ยวกับการพัสดุ หรือผู้ได้รับแต่งตั้ง

จากหัวหน้าส่วนราชการให้มีหน้าที่หรือปฏิบัติงานเกี่ยวกับการพัสดุ

การควบคุม หมายถึง การลงบัญชีวัสดุหรือทะเบียนคุมทรัพย์สิน การเก็บรักษาพัสดุ การเบิกพัสดุ การจ่ายพัสดุ

การลงบัญชีหรือทะเบียนพัสดุ หมายถึง การบันทึกรายละเอียดการรับหรือจ่ายในบัญชีหรือทะเบียนโดยจำแนกแต่ละประเภท แต่ละรายการของพัสดุ

การเก็บรักษาพัสดุ หมายถึง การจัดเก็บรักษาพัสดุให้เป็นระเบียบเรียบร้อย ถูกต้อง ครบถ้วน ตรงตามบัญชีหรือทะเบียน

การตรวจสอบพัสดุประจำปี หมายถึง การตรวจสอบการรับ-จ่ายพัสดุ งวดตั้งแต่วันที่ 1 ตุลาคม ของปีก่อน จนถึงวันที่ 30 กันยายน ของปีปัจจุบัน และการตรวจนับพัสดุคงเหลือ ณ วันที่ 30 กันยายน ของปีปัจจุบัน

หน้าที่ความรับผิดชอบ

- หัวหน้าหน่วยพัสดุ มีหน้าที่ควบคุมและสั่งจ่ายพัสดุ
- เจ้าหน้าที่พัสดุ มีหน้าที่ลงบัญชีหรือทะเบียนเพื่อควบคุมพัสดุ เมื่อได้รับมอบพัสดุ

ข้อควรทราบเกี่ยวกับทะเบียนคุมทรัพย์สิน

1. ครุภัณฑ์มูลค่าต่ำกว่าเกณฑ์ ให้บันทึกการควบคุมในทะเบียนคุมทรัพย์สิน แต่ไม่คิดค่าเสื่อมราคาตามหนังสือกรมบัญชีกลางที่ กค 04103 /ว48 ลงวันที่ 13 กันยายน 2549
2. เมื่อได้รับครุภัณฑ์และกำหนดรหัสครุภัณฑ์แล้ว ให้เขียนรหัสครุภัณฑ์ที่ตัวครุภัณฑ์ด้วย เพื่อสามารถควบคุมและตรวจสอบได้
3. ครุภัณฑ์ที่มีมูลค่าเกินกว่า 5,000 บาท ให้เขียนหมายเลขสินทรัพย์จากระบบ GFMS ที่ตัวครุภัณฑ์ด้วย เพื่อสามารถควบคุมและตรวจสอบได้
4. ทรัพย์สินที่ได้รับจากการบริจาคจะต้องดำเนินการตามระเบียบกระทรวงการคลังว่าด้วยการรับเงินและทรัพย์สินที่มีผู้บริจาคให้ทางราชการ พ.ศ. 2526 ด้วย

การตรวจสอบพัสดุประจำปี

ข้อ 155 ก่อนสิ้นเดือนกันยายนทุกปี ให้หัวหน้าส่วนราชการหรือหัวหน้าหน่วยงานซึ่งมีพัสดุไว้ จ่ายตามข้อ 153 แล้วแต่กรณี แต่งตั้งเจ้าหน้าที่ในส่วนราชการหรือหน่วยงานนั้น ซึ่งมีใช้เจ้าหน้าที่พัสดุคน หนึ่งหรือหลายคนตามความจำเป็น เพื่อตรวจสอบการรับจ่ายพัสดุดังตั้งแต่วันที่ 1 ตุลาคมปีก่อน จนถึง วันที่ 30 กันยายนปีปัจจุบัน และตรวจนับพัสดุประเภทที่คงเหลืออยู่เพียงวันสิ้นงวดนั้น

ในการตรวจสอบตามวรรคหนึ่ง ให้เริ่มดำเนินการตรวจสอบพัสดุในวันเปิดทำการวันแรกของเดือน ตุลาคมเป็นต้นไป ว่าการรับจ่ายถูกต้องหรือไม่ พาสุดคงเหลือมีตัวอยู่ตรงตามบัญชีหรือทะเบียนหรือไม่ มี พาสุดใดชำรุด เสื่อมคุณภาพ หรือสูญไป เพราะเหตุใด หรือพัสดุดีไม่จำเป็นต้องใช้ในราชการต่อไป แล้ว ให้เสนอรายงานผลการตรวจสอบดังกล่าวต่อผู้แต่งตั้งภายใน 30 วันทำการ นับแต่วันเริ่มดำเนินการ ตรวจสอบพัสดุนั้น

เมื่อผู้แต่งตั้งได้รับรายงานจากเจ้าหน้าที่ผู้ตรวจสอบแล้ว ให้ส่งรายงานเสนอตามลำดับชั้น จนถึง หัวหน้าส่วนราชการ 1 ชุด และส่งสำเนารายงานไปยังสำนักงานตรวจเงินแผ่นดินหรือสำนักงานตรวจเงิน แผ่นดินภูมิภาค แล้วแต่กรณี 1 ชุด สำหรับหน่วยงานในราชการบริหารส่วนภูมิภาค ให้ส่งสำเนารายงานไป ยังส่วนราชการต้นสังกัดอีก 1 ชุด ด้วย

ข้อ 156 เมื่อหัวหน้าส่วนราชการได้รับรายงานดังกล่าวตามข้อ 155 และปรากฏว่ามีพัสดุชำรุด เสื่อมสภาพ หรือสูญไป หรือไม่จำเป็นต้องใช้ในราชการต่อไป ก็ให้แต่งตั้งคณะกรรมการสอบหา

ข้อเท็จจริงขึ้นคณะหนึ่ง โดยให้นำความในข้อ 35 และข้อ 36 มาใช้บังคับโดยอนุโลม เว้นแต่กรณีที่เห็น ได้อย่างชัดเจนว่า เป็นการเสื่อมสภาพเนื่องมาจากการใช้งานตามปกติ หรือสูญเสียไปตามธรรมชาติ ให้หัวหน้า ส่วนราชการพิจารณาสั่งการให้ดำเนินการจำหน่ายต่อไปได้

ถ้าผลการพิจารณาปรากฏว่า จะต้องหาตัวผู้รับผิดชอบ ให้หัวหน้าส่วนราชการดำเนินการตามกฎหมายและระเบียบของทางราชการที่เกี่ยวข้องต่อไป

2.4.2 พิพุทธ กาญจนพันธุ์. (2551). ระบบควบคุมครุภัณฑ์ของคณะวิทยาการจัดการ มหาวิทยาลัยราชภัฏลำปางผ่านอินเทอร์เน็ต. ปรินญาวิทยาศาสตรมหาบัณฑิต สาขาวิชา วิทยาการคอมพิวเตอร์. เชียงใหม่: มหาวิทยาลัยเชียงใหม่.

การศึกษาพัฒนาระบบควบคุมครุภัณฑ์ของคณะวิทยาการจัดการ มหาวิทยาลัยราชภัฏลำปาง ใช้มายเอสคิวแอลเป็นเครื่องมือในการจัดการฐานข้อมูล และใช้ภาษาพีเอชพีในการพัฒนาติดต่อกับผู้ใช้งานทางอินเทอร์เน็ต ระบบประกอบด้วย 8 ระบบย่อยได้แก่ ส่วนของการจัดการข้อมูลครุภัณฑ์ ส่วนของการจัดการข้อมูลการยืมคืนครุภัณฑ์ ส่วนของการจำหน่ายครุภัณฑ์ ส่วนของการจัดการข้อมูลซ่อมแซมครุภัณฑ์ ส่วนของดูสรุปข้อมูล ส่วนของการจัดการข้อมูลผู้ใช้งาน ส่วนของการจัดการข้อมูลระบบ และส่วนของการอนุมัติการยืมครุภัณฑ์ การประเมินผลระบบได้ใช้แบบสอบถามกับผู้ใช้งานวน 12 คน หัวหน้าพัสดุจำนวน 1 คน เจ้าหน้าที่พัสดุจำนวน 1 คน และบุคลากรจำนวน 10 คน ผลการประเมินพบว่า ความสะดวกในการใช้งาน ความสามารถในการจัดการข้อมูล ความถูกต้องในการทำงาน และความรวดเร็วในการประมวลผลอยู่ในเกณฑ์ดี ดังนั้นแสดงว่าโดยรวมของระบบมีประสิทธิภาพอยู่ในเกณฑ์ดี

2.4.3 วัชรพล สงวนเต็ง. (2551). การพัฒนาระบบเพื่อการจัดการวัสดุและครุภัณฑ์สำหรับ สาขาวิชาสื่อศิลปะ และการออกแบบสื่อ มหาวิทยาลัยเชียงใหม่. วิทยานิพนธ์ปรินญาวิทยา ศาสตรมหาบัณฑิต สาขาวิชา เทคโนโลยีสารสนเทศและการจัดการ. เชียงใหม่: มหาวิทยาลัยเชียงใหม่.

การศึกษาการพัฒนาระบบเพื่อการจัดการวัสดุ และครุภัณฑ์สำหรับสาขาวิชาสื่อศิลปะ และการออกแบบสื่อ มหาวิทยาลัยเชียงใหม่ มีวัตถุประสงค์ เพื่อการพัฒนาระบบเพื่อการจัดการวัสดุ และครุภัณฑ์สำหรับสาขาวิชาสื่อศิลปะ และการออกแบบสื่อมหาวิทยาลัยเชียงใหม่ระบบนี้ถูกพัฒนาขึ้นจากการประเมินแบบสอบถามจากกลุ่มผู้ใช้งาน จำนวน 30 คน โดยแบ่งเป็น ผู้บริหาร 1 คน

อาจารย์ 4 คน นักศึกษา 16 คน เจ้าหน้าที่ 3 คน ผู้ใช้งานทั่วไป 5 คน และผู้ดูแลระบบ 1 คน ใช้ข้อมูลหนังสือ มัลติมีเดีย และครุภัณฑ์ในการประเมิน ข้อมูลละ 100 ชุด ข้อมูลนักศึกษา 50 ชุด ข้อมูลอาจารย์และบุคคลทั่วไป 20 ชุด พบว่าผู้ใช้งานมีระดับความพึงพอใจต่อระบบมาก สามารถช่วยลดขั้นตอนการทำงานที่เคยปฏิบัติอยู่เป็นประจำ ระบบมีความรวดเร็วในการแสดงผลและความถูกต้องในการประมวลผล อีกทั้งยังช่วยอำนวยความสะดวกในการดำเนินงาน ผู้ค้นคว้าเห็นว่าระบบนี้น่าจะเป็นประโยชน์ต่องานการบริหารระบบเพื่อการจัดการวัสดุ และครุภัณฑ์สำหรับสาขาวิชาสื่อ ศิลปะ และการออกแบบสื่อ มหาวิทยาลัยเชียงใหม่

2.4.4 สุวภี เรือนคา. (2552). การพัฒนาระบบสารสนเทศสำหรับครุภัณฑ์คอมพิวเตอร์ในโรงเรียนลำปาง พานิชัยการและเทคโนโลยี. ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์. เชียงใหม่: มหาวิทยาลัยเชียงใหม่.

การศึกษาการพัฒนาระบบสารสนเทศสำหรับครุภัณฑ์คอมพิวเตอร์ โรงเรียนลำปาง พานิชัยการและเทคโนโลยี มีวัตถุประสงค์เพื่อพัฒนาระบบสารสนเทศ สำหรับครุภัณฑ์คอมพิวเตอร์ ของโรงเรียนลำปางพานิชัยการและเทคโนโลยี โดยจากการศึกษาจากระบบงานเดิม พบว่าปัญหาที่เกิดขึ้นเกิดจากการประสานงานระหว่างช่างเทคนิค อาจารย์ผู้ควบคุมห้องปฏิบัติการ อาจารย์ผู้ควบคุมสต็อกอุปกรณ์ของโรงเรียนในการแจ้งซ่อมครุภัณฑ์ การซ่อมบำรุง และการเบิกอุปกรณ์คอมพิวเตอร์ที่ล่าช้า ผู้ศึกษาจึงได้ทำการพัฒนาระบบงานใหม่ในรูปแบบของเว็บเซอร์วิส (Web Service) เพื่อช่วยในการประสานงานระหว่างผู้ที่เกี่ยวข้องในระบบอย่างรวดเร็ว ถูกต้องและมีประสิทธิภาพ