

กองทุนเงินให้กู้ยืมเพื่อการศึกษา มีฐานะเป็นนิติบุคคลโดยอยู่ในการกำกับดูแลของกระทรวงการคลัง มีวัตถุประสงค์ให้กู้ยืมเงินแก่นักเรียนหรือนักศึกษาที่ขาดแคลนทุนทรัพย์ เพื่อเป็นค่าเล่าเรียน ค่าใช้จ่ายที่เกี่ยวข้องกับการศึกษา และค่าใช้จ่ายที่จำเป็นในการครองชีพระหว่างศึกษา

ต่อมา พระราชบัญญัติกองทุนเงินให้กู้ยืมเพื่อการศึกษา พ.ศ. 2560 ได้ประกาศในราชกิจจานุเบกษา เมื่อวันที่ 27 มกราคม 2560 และมีผลบังคับใช้เมื่อวันที่ 26 กรกฎาคม 2560 มีผลให้กองทุนเงินให้กู้ยืมเพื่อการศึกษาอยู่ในการกำกับดูแลของรัฐมนตรี และมีฐานะเป็นนิติบุคคลที่ไม่เป็นส่วนราชการตามกฎหมายว่าด้วยระเบียบบริหารราชการแผ่นดินหรือรัฐวิสาหกิจตามกฎหมายว่าด้วยวิธีการงบประมาณและกฎหมายอื่น เนื่องจากกองทุนเงินให้กู้ยืมเพื่อการศึกษาตามพระราชบัญญัติกองทุนเงินให้กู้ยืมเพื่อการศึกษา พ.ศ. 2541 และกองทุนเงินกู้ยืมเพื่อการศึกษาที่ผูกกับรายได้ในอนาคตตามระเบียบกระทรวงการคลัง ว่าด้วยการบริหารกองทุนเพื่อการศึกษา พ.ศ. 2549 ที่ออกตามพระราชบัญญัติเงินคงคลัง พ.ศ. 2491 มีการบริหารจัดการและการดำเนินการที่มีข้อจำกัด และไม่สอดคล้องกับนโยบายการผลิตกำลังคนและการพัฒนาประเทศสมควรบูรณาการการบริหารจัดการและการดำเนินการของกองทุน กยศ. และกองทุน กรอ. ให้เป็นเอกภาพอยู่ภายใต้กฎหมายเดียวกัน และเพิ่มมาตรการในการบริหารจัดการกองทุนให้มีประสิทธิภาพมากยิ่งขึ้น

ปัจจุบัน เนื่องจากพระราชบัญญัติกองทุนเงินให้กู้ยืมเพื่อการศึกษา (ฉบับที่ 2) พ.ศ. 2566 ได้ประกาศในราชกิจจานุเบกษา เล่ม 140 ตอนที่ 20 ก เมื่อวันที่ 19 มีนาคม 2566 และมีผลใช้บังคับตั้งแต่วันที่ 20 มีนาคม 2566 ซึ่งเป็นการแก้ไขเพิ่มเติมพระราชบัญญัติกองทุนเงินให้กู้ยืมเพื่อการศึกษา พ.ศ. 2560 เพื่อขยายโอกาสในการเข้าถึงเงินกู้ยืมเพื่อการศึกษาให้มากขึ้นเพื่อรองรับการเรียนรู้ตลอดชีวิตให้มีระบบการให้ทุนการศึกษา ในสาขาวิชาขาดแคลนที่ต้องได้รับการส่งเสริมเป็นพิเศษ ปรับเปลี่ยนการดำเนินงานของกองทุนเงินให้กู้ยืมเพื่อการศึกษาให้ทำงานเชิงรุก โดยเฉพาะอย่างยิ่งการให้ข้อมูลแก่นักเรียนหรือนักศึกษาก่อนเลือกสาขาวิชาที่จะกู้ยืมเงินเรียน เพื่อให้สอดคล้องกับอาชีพแห่งอนาคต รวมทั้งให้มีความคล่องตัวในการบริหารจัดการเงินกู้ยืมเพื่อการศึกษาอย่างเหมาะสมและเป็นธรรม รวมถึงกำหนดกลไกให้ผู้กู้ยืมเงินสามารถชำระเงินคืนกองทุนตามความสามารถในการหารายได้และสร้างวินัยในการชำระเงินคืนกองทุนเพื่อสร้างความยั่งยืนแก่กองทุน

2.1.2 เป้าหมาย (Goal)

- สนับสนุนส่งเสริมนักเรียน นักศึกษา และประชาชนให้เข้าถึงการศึกษาและการเรียนรู้ตลอดชีวิต

- ยกระดับการให้บริการเพื่อตอบสนองความต้องการและสร้างความพึงพอใจแก่ผู้มีส่วนได้ส่วนเสีย
- สร้างภาพลักษณ์องค์กรที่ดี และความภาคภูมิใจแก่ผู้กู้ยืมที่ได้รับเงินกู้ยืมจากกองทุนฯ รวมทั้งสร้างจิตสำนึกที่ดีในการชำระหนี้คืนเพื่อส่งต่อโอกาสให้แก่เยาวชนรุ่นต่อไป
- สามารถบริหารจัดการกองทุนโดยไม่พึ่งพางบประมาณแผ่นดิน
- เป็น Smart Organization อย่างเต็มรูปแบบ (Smart Services / Smart Operations / Smart Offices / Smart Peoples)

2.1.3 พันธกิจ (Mission)

- สนับสนุนและส่งเสริมการศึกษาด้วยการให้เงินกู้ยืมเพื่อการศึกษาให้นักเรียนหรือนักศึกษาที่ขาดแคลนทุนทรัพย์/ ศึกษาในสาขาวิชาที่เป็นความต้องการหลัก / ศึกษาในสาขาวิชาขาดแคลน หรือสาขาวิชาที่กองทุนมุ่งส่งเสริมเป็นพิเศษ/ เรียนดีเพื่อสร้างความเป็นเลิศ / ลักษณะอื่นตามความจำเป็นและเหมาะสม
- พัฒนาองค์กรในทุกมิติด้วยนวัตกรรมที่ทันสมัยตามหลักธรรมาภิบาล
- พัฒนาบุคลากรกองทุนให้มีศักยภาพและมีความพร้อมรองรับนวัตกรรมและการใช้เทคโนโลยีดิจิทัล



รูปที่ 2.1 กองทุนเงินให้กู้ยืมเพื่อการศึกษา

2.2 วงจรการพัฒนากระบวน (System Development Life : SDLC)

2.2.1 Software Development Life Cycle (SDLC) คืออะไร

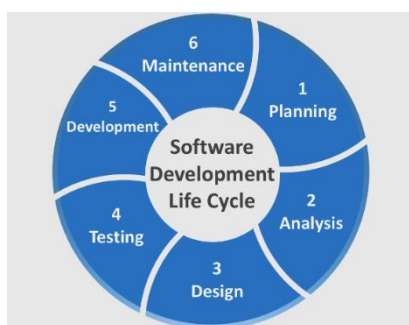
Software development life cycle (SDLC) เป็นกระบวนการที่ใช้ในการพัฒนาซอฟต์แวร์ โดยได้มีการกำหนดขั้นตอนต่างๆ เพื่อสามารถพัฒนาและส่งมอบซอฟต์แวร์ ได้อย่างมีคุณภาพและสมบูรณ์ โดยทั่วไปแล้ว SDLC จะประกอบไปด้วย 6 ขั้นตอนได้แก่ การวางแผน การเก็บรวบรวม Requirement การออกแบบซอฟต์แวร์ การพัฒนาซอฟต์แวร์ การทดสอบซอฟต์แวร์ การบำรุงรักษาซอฟต์แวร์

2.2.2 ขั้นตอนกระบวนการพัฒนาซอฟต์แวร์

- การวางแผน (Planning) เป็นการวางแผนในการพัฒนาซอฟต์แวร์ ใน Phase นี้ Program manager, Project Manager จะวางแผนใน Project ว่าจะใช้ระยะเวลาในการพัฒนาซอฟต์แวร์เท่าไร (Time) มี Feature อะไรบ้างที่จะพัฒนา (Scope) มีจำนวนคนที่จะใช้พัฒนาซอฟต์แวร์กี่คน (Resource) วางแผนว่าใครหรือองค์กรไหนที่จะมีส่วนเกี่ยวข้องกับการพัฒนาซอฟต์แวร์ได้บ้างเช่น หากทำ Application เกี่ยวกับการซื้อขายหุ้นก็ต้องอาศัยกฎ ข้อบังคับต่างๆของ กสท เข้ามาเป็นปัจจัยการออกแบบซอฟต์แวร์ด้วย วางแผนเพื่อเก็บ Requirement, Design Software และการทดสอบซอฟต์แวร์ การมีข้อมูลที่ละเอียดจะช่วยให้การวางแผนงานได้อย่างแม่นยำจะช่วยลดความเสี่ยงของการพัฒนาซอฟต์แวร์และสามารถพัฒนาสำเร็จตามวัตถุประสงค์
- การเก็บรวบรวมและวิเคราะห์ Requirement (Requirement gathering and analysis) ในขั้นตอนนี้ Requirement นั้น Project Manager, Product owner หรือ Business analysis จะเป็นผู้เก็บเก็บรวบรวม Requirement จากลูกค้าหรือผู้มีส่วนเกี่ยวข้อง เพื่อเข้าใจความต้องการจากผู้ใช้งานและการทำงานของระบบ โดยอาจจัดทำอยู่ในรูปแบบ Prototyping เพื่อให้ผู้ใช้งานสามารถเข้าใจการทำงานของระบบได้ง่ายขึ้น หลังจากการรวบรวม Requirement เสร็จแล้วทีมพัฒนาเช่น Developer, QA สามารถช่วยตรวจสอบ Requirement นั้นได้เพื่อใช้ในการ Design Software และประเมินความเสี่ยงและข้อจำกัดของซอฟต์แวร์ที่อาจเกิดขึ้นได้
- การออกแบบซอฟต์แวร์ (Design Software) ขั้นตอนนี้หลักๆแล้ว Developer จะนำ Requirement ที่ได้มาออกแบบ เช่น Design สถาปัตยกรรมที่จะใช้ (Architects) ส่วนต่อประสานผู้ใช้งาน (User Interface) ภาษาที่จะใช้ในการพัฒนาซอฟต์แวร์ Design ฐานข้อมูล Security และ Network เป็นต้น ผลลัพธ์ที่ได้จากขั้นตอนนี้จะได้ System Design Specification ซึ่งสามารถให้ผู้มีส่วนเกี่ยวข้องหรือผู้เชี่ยวชาญได้ Review และสามารถให้คำแนะนำได้
- การพัฒนาซอฟต์แวร์ (Software development) ขั้นตอนนี้เป็นการพัฒนาซอฟต์แวร์โดยทีม Developer โดยจะเริ่มนำ Design และ Requirement ต่างๆที่ได้ ออกแบบในขั้นตอนที่ 2 และ 3 มาเริ่มเขียนโปรแกรม การพัฒนาซอฟต์แวร์นั้นจะนำเครื่องมือต่างๆมาช่วยพัฒนาซอฟต์แวร์ได้ เครื่องมือที่จะนำมาใช้นั้นก็ขึ้นอยู่กับลักษณะของซอฟต์แวร์ เมื่อการพัฒนาซอฟต์แวร์เริ่มขึ้นจะเริ่มแต่ละ Feature ย่อย จากนั้นค่อยๆรวมแต่ละ Feature ย่อยเข้าด้วยกันให้เป็นระบบ ในการพัฒนา

ซอฟต์แวร์นั้นจะพัฒนาได้อย่างรวดเร็วหาก Phase ช่วงการ Design นั้นละเอียดและมีเอกสารที่เข้าใจได้ง่าย

- การทดสอบซอฟต์แวร์ (Software Testing) การทดสอบซอฟต์แวร์เป็นการประกันคุณภาพของซอฟต์แวร์ ค้นหาข้อผิดพลาด ป้องกันการเกิดข้อผิดพลาดของซอฟต์แวร์ รวมไปถึงตรวจสอบว่าซอฟต์แวร์ที่พัฒนานั้นเป็นไปตาม Requirement หรือไม่ ในการทดสอบนั้นจะมีการสร้าง Test case หรือเป็นกรณี que ที่ผู้ใช้งานจะใช้งานระบบและ Expected results คือผลที่คาดหวังจากการทดสอบโดยผลที่คาดหวังนี้จะอิงตาม Requirement ที่ได้จาก Phase ที่ 2 หากซอฟต์แวร์ทำงานถูกต้องเป็นไปตามสิ่งที่คาดหวังผลการทดสอบก็จะผ่านแต่หากไม่เป็นไปตามผลที่คาดหวังผลการทดสอบก็จะไม่ผ่าน ต้องสิ่งให้ Developer ต่อไป ปัจจุบันนั้นมีการทำ Automated test เข้าใจช่วยในการ Run test เพื่อการทำงานที่รวดเร็วมากขึ้น
- การบำรุงรักษาซอฟต์แวร์ (Operations and Maintenance) เมื่อซอฟต์แวร์ผ่านการทดสอบแล้วก็จะมีการ Deployed เพื่อให้ผู้ใช้งานได้ใช้งานระบบจริง ในระหว่างที่ผู้ใช้งานซอฟต์แวร์อยู่นั้นก็อาจจะเกิดปัญหาต่างๆได้เช่น ต้องการ Feature เพิ่มเติม ซอฟต์แวร์ทำงานไม่เป็นไปตามความคาดหวัง ซอฟต์แวร์มี Bug อยู่ หรือการ Deployed ครั้งแรกอาจมีข้อจำกัดอะไรบางอย่าง เมื่อเกิดปัญหาเหล่านี้เกิดขึ้นก็ต้องมีการแก้ไข บำรุงรักษาซอฟต์แวร์เพื่อให้ผู้ใช้งานพึงพอใจมากที่สุด ซึ่งในการแก้ไขก็อาจจะกลับไปเริ่มในขั้นตอนที่ 1 ถึง 6 ใหม่ (New Development Cycle)



รูปที่ 2.2 ขั้นตอนกระบวนการพัฒนาซอฟต์แวร์

2.3 Data Flow Diagram : DFD

2.3.1 Data Flow Diagram คืออะไร


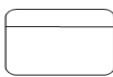
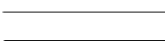



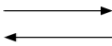
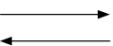
DFD หรือ (Data Flow Diagram) ก็คือแผนภาพกระแสข้อมูลหรือแผนภาพการไหลของข้อมูลเป็นเครื่องมือที่ใช้แสดงการไหลของข้อมูลและการประมวลผลต่างๆ ในระบบ

สัมพันธ์กับแหล่งเก็บข้อมูลที่ใช้ โดยแผนภาพนี้จะเป็นสื่อช่วยให้การวิเคราะห์เป็นไปได้โดยง่าย และมีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบกับโปรแกรมเมอร์หรือ ระหว่างผู้วิเคราะห์ระบบกับผู้ใช้

2.3.2 วัตถุประสงค์ของการสร้างแผนภาพกระแสข้อมูล

- เป็นแผนภาพที่สรุปรวมข้อมูลทั้งหมดที่ได้จากการวิเคราะห์ในลักษณะของรูปแบบที่เป็นโครงสร้าง
- เป็นข้อตกลงร่วมกันระหว่างนักวิเคราะห์ระบบและผู้ใช้งาน
- เป็นแผนภาพที่ใช้ในการพัฒนาต่อในขั้นตอนของการออกแบบระบบ
- เป็นแผนภาพที่ใช้ในการอ้างอิง หรือเพื่อใช้ในการพัฒนาต่อในอนาคต
- ทราบที่มาที่ไปของข้อมูลที่ไหลไปในกระบวนการต่างๆ (Data and Process)

2.3.3 ความหมายของสัญลักษณ์ต่างๆ (ใช้สัญลักษณ์ของ Gane & Sarson)

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

รูปที่ 2.3 ความหมายของสัญลักษณ์ Data Flow Diagram

2.3.4 ข้อดี หรือ ประโยชน์ของ Data Flow Diagram

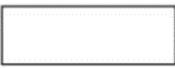


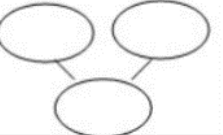
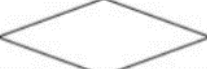





- ใช้ได้อย่างอิสระ โดยไม่ต้องมีเทคนิคอื่นมาช่วย
- สามารถมองเห็นระบบเก่าและระบบใหม่ได้ชัดเจน
- เห็นถึงข้อมูลและขั้นตอนต่าง ๆ ของระบบเป็นแผนภาพการไหลของข้อมูล
- มีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบกับ โปรแกรมเมอร์ หรือระหว่างผู้วิเคราะห์กับผู้ใช้ระบบ

2.4 Entity Relationship Model (ERD) : ER Diagram

2.4.1 ER Diagram คืออะไร

ER Diagram คือ แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพ อธิบายโครงสร้างและความสัมพันธ์ของข้อมูล ประกอบด้วย 2 ส่วน เอนทิตี (Entity) ความสัมพันธ์ (Relationship) ระหว่าง Entity

2.4.2 สัญลักษณ์ที่ใช้ในการเขียน

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

รูปที่ 2.4 สัญลักษณ์ที่ใช้ใน er diagram

2.4.3 สัญลักษณ์ที่ใช้ในการเขียน

Entity (เอนทิตี) เป็นส่วนที่ใช้สำหรับเก็บข้อมูลแต่ละรายการภายในระบบที่เรา กำลังจัดทำอยู่ เช่น บุคคล (พนักงาน, ผู้ป่วย, บุคลากร, นักศึกษา, ลูกค้า) สถานที่ (จังหวัด, อำเภอ, ภาค, ที่อยู่) วัตถุ (รถยนต์, อาคาร, เครื่องจักร, สินค้า) เหตุการณ์ (ประวัติ, การลงทะเบียน, การรักษาโรค, ซื้อ, ขาย)

คือสิ่งที่คงอยู่ สามารถระบุได้ในความจริง เช่น บุคคล เหตุการณ์ สถานที่ที่มักจะอยู่ในรูปของนาม สามารถมีคุณสมบัติ (Property) หรือแอททริบิวต์ (Attribute) ได้

2.4.4 แบ่งประเภทของ Entity ได้ 3 ชนิดดังนี้

- Regular Entity คือ Entity ทั่ว ๆ ไป ที่มี Attribute หนึ่งแยกความแตกต่างของข้อมูลแต่ละ Tuple ได้

เอนทิตี STUDENT

SID	SName	Address	Grade
47035501	ประธาน สมพงษ์	111 หมู่ 5 สามเสน กรุงเทพฯ	3.5
47035502	อำนาจ รักดี	222 หมู่ 2 บางซื่อ กรุงเทพฯ	2.8
47035503	ประนอม รักอ่าน	333 หมู่ 1 หนองจอก กรุงเทพฯ	3.0

Student

รูปที่ 2.5 รูปแบบ Regular Entity

- Weak Entity หรือ Entity อ่อนแอ คือ Entity ที่ต้องอาศัย Attribute จาก Entity อื่นมาช่วยในแยกความแตกต่างของข้อมูลแต่ละ Tuple

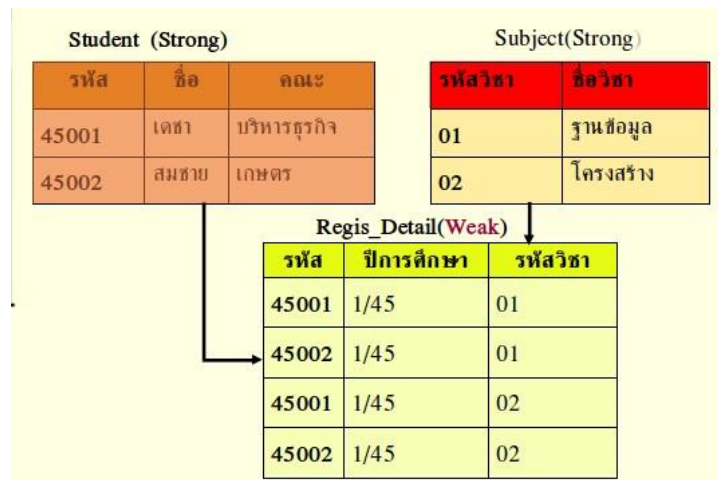
เอนทิตี Time stamp

EmpID	Date	Time_In	Time_Out
47035501	2/7/47	7.30	17.30
47035502	2/7/47	8.00	18.00
47035503	2/7/47	7.45	17.45

Timestamp

รูปที่ 2.6 รูปแบบ weak Entity

- Composite Entity คือ Entity ที่ต้องมี Attribute ที่เป็น Primary key จาก 2 Entity แยกความแตกต่างของข้อมูลแต่ละ Tuple เป็น Entity ที่ถูกสร้างขึ้นเพื่อแปลงความสัมพันธ์แบบ M:M ให้เป็น แบบ 1:M



รูปที่ 2.7 รูปแบบ Composite Entity

2.5 ระบบจัดการฐานข้อมูล (Database Management Systems : DBMS)

2.5.1 Database Management System คืออะไร

DBMS หรือ Database Management System คือ โปรแกรมที่ทำหน้าที่เป็นตัวกลางระหว่างผู้ใช้ (user) กับฐานข้อมูลเพื่อจัดการและควบคุมความถูกต้อง ความซ้ำซ้อน และความสัมพันธ์ระหว่างข้อมูลต่างๆ ภายในฐานข้อมูล ซึ่งต่างไปจากระบบแฟ้มข้อมูลคือ หน้าที่เหล่านี้จะเป็นของโปรแกรมเมอร์ ในการต่อฐานข้อมูลไม่ว่าจะด้วยการใช้คำสั่งในกลุ่ม DML หรือ DDLหรือจะด้วยโปรแกรมต่าง ๆ ทุกคำสั่งที่ใช้กระทำกับฐานข้อมูลจะถูกโปรแกรม DBMS นำไปแปล (Compile) เป็นการกระทำ (Operation) ต่างๆภายใต้คำสั่งนั้นๆ เพื่อนำไปกระทำกับตัวข้อมูลในฐานข้อมูลต่อไปส่วนการทำงานต่าง ๆ ภายในโปรแกรม DBMS ที่ทำหน้าที่ในการแปลคำสั่งไปเป็นการกระทำต่าง ๆ ดังนี้

- Database Manager เป็นส่วนที่ทำหน้าที่กำหนดการกระทำต่าง ๆ ให้กับส่วน File Manager เพื่อไปกระทำกับข้อมูลที่เก็บอยู่ในฐานข้อมูล File Manager เป็นส่วนที่ทำหน้าที่บริหาร และจัดการกับข้อมูลที่เก็บอยู่ในฐานข้อมูลในระดับกายภาพ
- Query Process เป็นส่วนที่ทำหน้าที่แปลงประโยคคำสั่งของ Query Language ให้อยู่ในรูปแบบของคำสั่งที่ Database Manager เข้าใจ
- Data Manipulation Language Precompiler เป็นส่วนที่ทำหน้าที่แปลงประโยคคำสั่งของกลุ่มคำสั่ง DML ให้อยู่ในรูปแบบที่ส่วน Application Programs Object Code จะนำไปเข้ารหัสเพื่อส่งต่อไปยังส่วน Database Manager ในการแปล

ประโยคคำสั่งของกลุ่มคำสั่ง DML ของ Data Manipulation Language Precompiler นี้ จะต้องทำงานร่วมกับส่วน Query Processor

- Data Definition Language Precompiler เป็นส่วนที่ทำหน้าที่แปลประโยคคำสั่งของกลุ่มคำสั่ง DDL ให้อยู่ในรูปแบบของ Meta Data (รายละเอียดที่บอกถึงโครงสร้างต่าง ๆ ของข้อมูล) ที่เก็บอยู่ในส่วน Data Dictionary ของฐานข้อมูล
- Application Programs Object Code เป็นส่วนที่ทำหน้าที่แปลงคำสั่งต่าง ๆ ของโปรแกรมรวมทั้งคำสั่งในกลุ่มคำสั่ง DML ที่ส่งต่อมาจากส่วน Data Manipulation Language Precompiler ให้อยู่ในรูปแบบของ Object Code ที่จะส่งต่อไปให้ Database manager เพื่อกระทำกับข้อมูลในฐานข้อมูล

โปรแกรม DBMS ถูกพัฒนาขึ้นเพื่อแก้ไขปัญหาด้าน Data Independence ที่ไม่มีในระบบแฟ้มข้อมูล ทำให้มีความเป็นอิสระจากทั้งส่วนของฮาร์ดแวร์ และข้อมูลภายในฐานข้อมูลกล่าวคือโปรแกรม DBMS นี้จะมีความทำงานที่ไม่ขึ้นอยู่กับรูปแบบ (Platform) ของตัวฮาร์ดแวร์ ที่นำมาใช้กับระบบฐานข้อมูลรวมทั้งมีรูปแบบในการอ้างถึงข้อมูลที่ไม่ขึ้นอยู่กับโครงสร้างทางกายภาพของข้อมูลด้วยการใช้ Query Language ในการติดต่อกับข้อมูลในฐานข้อมูลแทนคำสั่งภาษาคอมพิวเตอร์ในยุคที่ 3 ส่งผลให้ผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลได้โดยไม่ต้องทราบถึงประเภทหรือขนาดของข้อมูลนั้นหรือสามารถกำหนดลำดับที่ของฟิลด์ ในการกำหนดการแสดงผลได้โดยไม่ต้องคำนึงถึงลำดับที่จริงของฟิลด์ นั้น

2.5.2 หน้าที่ของ DBMS

โปรแกรม DBMS ได้ถูกพัฒนาขึ้นมาเพื่อแก้ปัญหาทางด้าน Data Independence ที่ไม่มีในระบบแฟ้มข้อมูล ดังนั้นจึงมีความเป็นอิสระจากทั้งฮาร์ดแวร์ และตัวข้อมูลภายในฐานข้อมูล กล่าวคือ โปรแกรม DBMS จะมีความทำงานที่ไม่ขึ้นอยู่กับรูปแบบ (Platform) ของตัวฮาร์ดแวร์ ที่นำมาใช้กับระบบฐานข้อมูล รวมทั้งมีรูปแบบในการอ้างถึงข้อมูลที่ไม่ขึ้นอยู่กับโครงสร้างทางกายภาพของข้อมูล ด้วยการ ใช้ Query Language ในการติดต่อกับข้อมูลในฐานข้อมูล หน้าที่ของโปรแกรม DBMS โดยสรุปมีดังนี้

- ทำหน้าที่แปลงคำสั่งที่ใช้จัดการกับข้อมูลภายในฐานข้อมูลให้อยู่ในรูปแบบที่ข้อมูลเข้าใจ
- ทำหน้าที่ในการนำคำสั่งต่างๆ ซึ่งได้รับการแปลแล้วไปสั่งให้ฐานข้อมูลทำงาน เช่น การเรียกใช้ข้อมูล (Retrieve) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) หรือ การเพิ่มข้อมูลเป็นต้น (Add) ฯลฯ

- ทำหน้าที่ป้องกันความเสียหายที่จะเกิดขึ้นกับข้อมูลภายในฐานข้อมูล โดยจะคอยตรวจสอบว่าคำสั่งใดที่สามารถทำงานได้และคำสั่งใดที่ไม่สามารถทำได้
- ทำหน้าที่รักษาความสัมพันธ์ของข้อมูลภายในฐานข้อมูลให้มีความถูกต้องอยู่เสมอ
- ทำหน้าที่เก็บรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูลไว้ใน Data Dictionary ซึ่งรายละเอียดเหล่านี้มักจะถูกเรียกว่า "ข้อมูลของข้อมูล" (Meta Data)
- ทำหน้าที่ควบคุมให้ฐานข้อมูลทำงานได้อย่างถูกต้องและมีประสิทธิภาพ



รูปที่ 2.8 ระบบจัดการฐานข้อมูล DBMS

2.6 MySQL

2.6.1 MySQL คืออะไร

MySQL คือ ระบบจัดการฐานข้อมูล หรือ Database Management System (DBMS) แบบข้อมูลเชิงสัมพันธ์ หรือ Relational Database Management System (RDBMS) ซึ่งเป็นระบบฐานข้อมูลที่จัดเก็บรวบรวมข้อมูลในรูปแบบตาราง โดยมีการแบ่งข้อมูลออกเป็นแถว (Row) และในแต่ละแถวแบ่งออกเป็นคอลัมน์ (Column) เพื่อเชื่อมโยงระหว่างข้อมูลในตารางกับข้อมูลในคอลัมน์ที่กำหนด แทนการเก็บข้อมูลที่แยกออกจากกัน โดยไม่มีความเชื่อมโยงกัน ซึ่งประกอบด้วยข้อมูล (Attribute) ที่มีความสัมพันธ์เชื่อมโยงกัน (Relation) โดยใช้ RDBMS Tools สำหรับการควบคุมและจัดเก็บฐานข้อมูลที่จำเป็น ทำให้นำไปประยุกต์ใช้งานได้ง่าย ช่วยเพิ่มประสิทธิภาพในการทำงานให้มีความยืดหยุ่นและรวดเร็ว

ได้มากยิ่งขึ้น รวมถึงเชื่อมโยงข้อมูล ที่จัดแบ่งกลุ่มข้อมูลแต่ละประเภทได้ตามต้องการ จึงทำให้ MySQL เป็นโปรแกรมระบบจัดฐานข้อมูลที่มีความนิยมสูง

2.6.2 หน้าที่ของ MySQL

MySQL มีหน้าที่จัดเก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่งภาษา Structured Query Language หรือ SQL เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ เป็นภาษามาตรฐานบนระบบฐานข้อมูลเชิงสัมพันธ์และเป็นระบบเปิด (Open System) ที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน และนิยมใช้งานร่วมกับภาษาโปรแกรม PHP รวมถึงภาษาอื่น ๆ ที่สามารถทำงานร่วมกันกับฐานข้อมูล MySQL ได้หลากหลาย เช่น C, C++, Python, Java เป็นต้น อีกทั้ง MySQL ยังได้รับการออกแบบและปรับให้มีความเหมาะสมสำหรับการพัฒนา Website และ Web Application ทำให้สามารถรองรับการทำงานได้ทุกแพลตฟอร์ม รวมถึงการอนุญาตให้ผู้ใช้งานหลายคนสามารถใช้งานพร้อมกันได้ (Multi-user) นอกจากนั้นยังสามารถจัดการและสร้างฐานข้อมูลจำนวนมากรวมถึงประมวลผลหลาย ๆ งานได้พร้อมกัน (Multi-threaded) อย่างสมบูรณ์ จึงทำให้ MySQL เป็นตัวเลือกยอดนิยมสำหรับธุรกิจการพาณิชย์อิเล็กทรอนิกส์ หรือ Electronic Commerce (E-Commerce) และเหมาะสำหรับการนำไปใช้งานสร้างเว็บไซต์ทั่วไป เพราะมีความแม่นยำ ครอบคลุม ช่วยให้เข้าถึงข้อมูลได้อย่างรวดเร็ว อีกทั้งยังมีความน่าเชื่อถือสูง และยังมีโปรแกรมเสริมช่วยจัดฐานข้อมูลที่ใช้กันง่าย เช่น Mysql Admin, phpMyAdmin เป็นต้น



รูปที่ 2.9 MySQL

2.7 Visual Studio Code

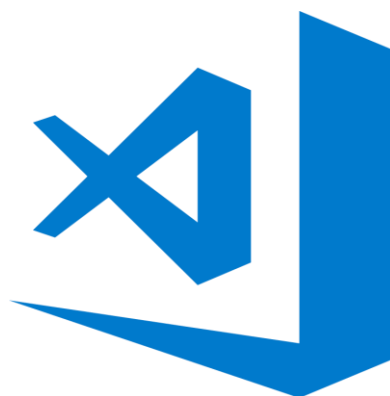
2.7.1 Visual Studio Code คืออะไร

Visual Studio Code หรือ VSCode เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไข และ ปรับแต่งโค้ด จากค่ายไมโครซอฟท์ มีการพัฒนาออกมาในรูปแบบของ OpenSource จึงสามารถ นำมาใช้งานได้แบบฟรี ๆ ที่ต้องการความเป็นมืออาชีพ ซึ่ง Visual Studio Code นั้น

เหมาะสำหรับ นักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้ นำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือส่วนขยายต่าง ๆ ให้เลือกใช้อย่างมากมาย ไม่ว่าจะเป็น

- การ เปิดใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++, C#, Java, Python, PHP หรือ Go
- Themes
- Debugger
- Commands

ความแตกต่างระหว่าง VSCode และ Visual Studio คือ VSCode ได้ทำการตัดในส่วนของ GUI designer ออกไป เหลือแต่เพียงตัว Editor เท่านั้น จึงทำให้ตัวโปรแกรมนั้นค่อนข้างเบากว่า Visual Studio เป็นอย่างมาก VSCode สามารถนำมาใช้งานได้ฟรี รองรับการ ทำงานข้ามแพลตฟอร์ม



รูปที่ 2.10 Visual Studio Code

2.8 ภาษา HTML

2.8.1 HTML คืออะไร

HTML ย่อมาจาก Hyper Text Mark-Up Language เป็นภาษามาตรฐานที่ได้รับ ความนิยม นำมาเขียนเว็บไซต์ในปัจจุบันนี้ เพราะสามารถนำมาช่วยในการตกแต่งรูปร่าง หน้าตาของเว็บไซต์ และ ใส่รูปภาพทำให้แต่ละเว็บไซต์มีความสวยงามแตกต่างกันออกไป อีกทั้ง ภาษา HTML ยังสามารถใช้งาน ร่วมกับภาษาอื่นหรือสคริปต์อื่นได้หลายภาษา หลากหลาย เครื่องมือในการใช้งานไม่ว่าจะเป็นโน้ตบุ๊ก คอมพิวเตอร์แบบตั้งโต๊ะ อุปกรณ์พกพา หลากหลายชนิด ในการพัฒนาเว็บไซต์ด้วยภาษา HTML จะมี เอดิเตอร์ช่วยในการจัดการซอร์

ทโค้ด อาทิเช่น Notepad, Editplus, Wordpad เป็นต้น เมื่อเขียน คำสั่งเสร็จให้บันทึกเป็นไฟล์นามสกุล .HTML และคำสั่งเหล่านั้นสามารถนำมารันกับโปรแกรมเว็บ เบราวเซอร์จะได้เป็นเว็บไซต์เพื่อใช้งานต่อไปได้ (ณัตตยา เอี่ยมคง ,2560)

ภาษา HTML ได้ถูกพัฒนาขึ้นอย่างต่อเนื่องตั้งแต่ HTML Level 1, HTML 2.0, HTML 3.0, HTML 3.2 และ HTML 4.0 ในปัจจุบัน ทาง W3C ได้ผลักดัน รูปแบบของ HTML แบบใหม่ ที่เรียกว่า XHTML ซึ่งเป็นลักษณะของโครงสร้าง XML แบบหนึ่ง ที่มีหลักเกณฑ์ในการกำหนดโครงสร้างของ โปรแกรมที่มีรูปแบบที่มาตรฐานกว่า มาทดแทนใช้ HTML รุ่น 4.01 ที่ใช้กันอยู่ในปัจจุบัน

การสร้างเว็บเพจ โดยใช้ภาษา HTML สามารถทำได้โดยใช้โปรแกรม Text Editor ต่างๆ เช่น Notepad, EditPlus หรือจะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น Microsoft FrontPage, Dream Weaver ซึ่งอำนวยความสะดวกในการสร้างหน้า HTML ในลักษณะ WYSIWYG (What You See Is What You Get)

แต่มีข้อเสียคือ โปรแกรมเหล่านี้มัก generate code ที่เกินความจำเป็นมากเกินไป ทำให้ ไฟล์ HTML มีขนาดใหญ่ และแสดงผลช้า ดังนั้นหากเรามีความเข้าใจภาษา HTML จะเป็นประโยชน์ ให้เราสามารถแก้ไข code ของเว็บเพจได้ตามความต้องการ และยังสามารถนำ script มาแทรก ตัด ต่อ สร้างลูกเล่นสีสันให้กับเว็บเพจของเราได้



2.9 ภาษา CSS

2.9.1 CSS คืออะไร

CSS คือ ภาษาที่ใช้สำหรับตกแต่งเอกสาร HTML/XHTML ให้มีหน้าตา สีสัน ระยะห่าง พื้น หลัง เส้นขอบและอื่นๆ ตามที่ต้องการ CSS ย่อมาจาก Cascading Style Sheets มีลักษณะเป็นภาษา 11 ที่มีรูปแบบในการเขียน Syntax แบบเฉพาะและได้ถูก

กำหนดมาตรฐานโดย W3C เป็นภาษาหนึ่งใน การตกแต่งเว็บไซต์ ได้รับความนิยมอย่างแพร่หลาย

2.9.2 ประโยชน์ของ CSS

- ช่วยให้เนื้อหาภายในเอกสาร HTML มีความเข้าใจได้ง่ายขึ้นและในการ แก้ไขเอกสารก็สามารถทำได้ง่ายกว่าเดิม เพราะการใช้ CSS จะช่วยลดการใช้ภาษา HTML ลงได้ในระดับหนึ่ง และแยกระหว่างเนื้อหากับรูปแบบในการแสดงผลได้อย่างชัดเจน
- ทำให้สามารถดาวน์โหลดไฟล์ได้เร็ว เนื่องจาก code ในเอกสาร HTML ลดลง จึงทำให้ไฟล์มีขนาดเล็กลง
- สามารถกำหนดรูปแบบการแสดงผลจากคำสั่ง style sheet ชุดเดียวกัน ให้ มีการแสดงผลในเอกสารแบบเดียวกันทั้งหน้าหรือในทุกๆ หน้าได้ ช่วยลดเวลาในการปรับปรุง และทำให้ การสร้างเอกสารบนเว็บมีความรวดเร็วยิ่งขึ้น นอกจากนี้ยังสามารถควบคุมการแสดงผล ให้คล้ายหรือ เหมือนกันได้ในหลาย Web Browser
- ช่วยในการกำหนดการแสดงผลในรูปแบบที่มีความเหมาะสมกับสื่อต่างๆ ได้ เป็นอย่างดี
- ทำให้เว็บไซต์มีความเป็นมาตรฐานมากขึ้นและมีความทันสมัย สามารถ รองรับการใช้งานในอนาคตได้ดี (<https://blog.sogoodweb.com/Article/Detail/79237>)



รูปที่ 2.12 ภาษา CSS

2.10 phpMyAdmin

2.12.1 phpMyAdmin คืออะไร

phpMyAdmin คือโปรแกรมที่ถูกพัฒนาโดยใช้ภาษา PHP เพื่อใช้ในการบริหารจัดการฐานข้อมูล Mysql แทนการคีย์คำสั่ง เนื่องจากถ้าเราจะใช้ฐานข้อมูลที่เป็น MySQL บางครั้งจะมีความลำบากและยุ่งยากในการใช้งาน ดังนั้นจึงมีเครื่องมือในการจัดการ

ฐานข้อมูล MySQL ขึ้นมาเพื่อให้สามารถจัดการ ตัวDBMS ที่เป็น MySQL ได้ง่ายและสะดวกยิ่งขึ้น โดย phpMyAdmin ก็ถือเป็นเครื่องมือชนิดหนึ่งในการจัดการนั่นเอง

phpMyAdmin เป็นส่วนต่อประสานที่สร้างโดยภาษาพีเอชพี ซึ่งใช้จัดการฐานข้อมูล MySQL ผ่านเว็บเบราว์เซอร์ โดยสามารถที่จะทำการสร้างฐานข้อมูลใหม่ หรือทำการสร้าง TABLE ใหม่ๆ และยังมี function ที่ใช้สำหรับการทดสอบการ query ข้อมูลด้วยภาษา SQL พร้อมกันนั้น ยังสามารถทำการ insert delete update หรือแม้กระทั่งใช้ คำสั่งต่างๆ เหมือนกับกับการใช้ภาษา SQL ในการสร้างตารางข้อมูล

phpMyAdmin เป็นโปรแกรมประเภท MySQL Client ตัวหนึ่งที่ใช้ในการจัดการข้อมูล MySQL ผ่านweb browser ได้โดยตรง phpMyAdmin ตัวนี้จะทำงานบน Web server เป็น PHP Application ที่ใช้ควบคุมจัดการ MySQL Server

2.12.2 ความสามารถของ phpMyAdmin คือ

- สร้างและลบ Database
- สร้างและจัดการ Table เช่น แทรก record, ลบ record, แก้ไข record, ลบ Table, แก้ไข field
- โหลดเท็กซ์ไฟล์เข้าไปเก็บเป็นข้อมูลในตารางได้
- หาผลสรุป (Query) ด้วยคำสั่ง SQL



รูปที่ 2.13 ตัวอย่างการใช้งานโปรแกรม phpMyAdmin

2.11 ภาษา php

PHP ย่อมาจากคำว่า PHP Hypertext Preprocessor แต่สมัยก่อนย่อมาจากคำว่า Personal Home Page Tools เป็นโปรแกรมภาษาคอมพิวเตอร์ระดับสูง ประเภท Scripting Language ซึ่งภาษาประเภทนี้เก็บคำสั่งต่าง ๆ ในไฟล์ที่เรียกว่า Script และเวลาใช้งานต้องอาศัยตัวแปรชุดคำสั่ง โดย

มีรากฐานโครงสร้างคำสั่งมาจากภาษาซี (C Programming Language) ภาษาจาวาสคริปต์ (JavaScript) และ ภาษาเพิร์ล (Practical Extraction and Report Language)

PHP สามารถใช้งานได้ผ่านอินเทอร์เน็ต มีความสามารถสูง และมีผู้นิยมใช้เป็นจำนวนมาก อีกทั้งยังสามารถดาวน์โหลดมาใช้งานได้ฟรี จากเว็บไซต์ <http://php.net/>

ลักษณะของ PHP แตกต่างจากภาษาสคริปต์แบบอื่น ๆ เพราะ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML สามารถสอดแทรกหรือแก้ไขเนื้อหาได้อัตโนมัติ ดังนั้น PHP จึงเป็นภาษาที่เรียกว่า Server-Side หรือ HTML-Embedded Scripting Language สามารถประมวลผลตามคำสั่งและแสดงผลลัพธ์เป็นเว็บเพจตามที่ต้องการ ถือได้ว่า PHP เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้เราสามารถสร้าง Dynamic Web Pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

PHP เกิดขึ้นมาจากกลุ่มนักพัฒนาที่มีการเปิดเผยโค้ดต้นฉบับ ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็วและแพร่หลายโดยเฉพาะอย่างยิ่ง เมื่อใช้ร่วมกับ Apache Web Server, ระบบปฏิบัติการ เช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลาย ๆ ตัวบนระบบปฏิบัติการต่าง ๆ เช่น Windows 95/98/NT เป็นต้น

PHP เป็นภาษาสคริปต์ (Scripting Language) คำสั่งต่าง ๆ จะเก็บในรูปแบบของข้อความ (Text) อาจเขียนแทรกอยู่ภายในภาษา HTML หรือถูกเขียนอย่างอิสระ แต่ในการใช้งานจริงมักใช้งานร่วมกับภาษา HTML ดังนั้นการเขียนโปรแกรมด้วยภาษา PHP ต้องมีความรู้ด้านภาษา HTML เป็นอย่างดีจึงสามารถเขียนโปรแกรมได้สมบูรณ์แบบ อย่างไรก็ตามเราสามารถใช้อุปกรณ์ประยุกต์มาช่วยอำนวยความสะดวกในการสร้างงานได้เช่นกัน เช่น Macromedia, Dreamweaver หรือโปรแกรมประเภท Editor ต่าง ๆ เช่น EditPlus เป็นต้น ซึ่งโปรแกรมเหล่านี้จะช่วยจำแนกคำสั่งต่าง ๆ ด้วยสีที่แตกต่างกันออกไป เช่น คำสั่ง คำทั่วไป ตัวแปร ฯลฯ เพื่อความสะดวกในการสังเกตและยังมีตัวเลขบอกบรรทัดทำให้สะดวกในการแก้ไขมากขึ้นอีกด้วย

PHP จึงเหมาะสำหรับการจัดทำเว็บไซต์และสามารถประมวลผลออกมาในรูปแบบ HTML โดยเป้าหมายหลักของภาษา PHP คือให้นักพัฒนาเว็บไซต์สามารถเขียนเว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว



รูปที่ 2.14 ภาษา php

