

## บทที่ 13

# การวิเคราะห์และออกแบบระบบเชิงวัตถุด้วย UML

การวิเคราะห์และออกแบบระบบเชิงวัตถุเป็นแนวคิดที่พยายามจัดระบบกระบวนการพัฒนาระบบงานให้มีระเบียบ และสามารถนำโปรแกรมที่เคยเขียนมาก่อนให้สามารถนำกลับมาใช้งานใหม่ การวิเคราะห์ระบบเชิงวัตถุเป็นการอธิบายระบบสารสนเทศว่าประกอบด้วยสิ่งต่าง ๆ ที่เรียกว่า วัตถุ (Object) ทั้งที่จับต้องได้และจับต้องไม่ได้ ผลลัพธ์สุดท้ายของการวิเคราะห์ระบบเชิงวัตถุก็คือ แบบจำลองเชิงวัตถุ (Object Model) ที่นำเสนอระบบสารสนเทศในลักษณะเชิงวัตถุ จากนั้นในระหว่างขั้นตอนของการพัฒนาระบบในวงจรการพัฒนา (SDLC) สามารถแปลงเป็นการออกแบบเชิงวัตถุได้โดยตรงโดยใช้โปรแกรมเชิงวัตถุในการพัฒนา เช่น C++ Java เป็นต้น

### 13.1 แนวคิดเชิงวัตถุ

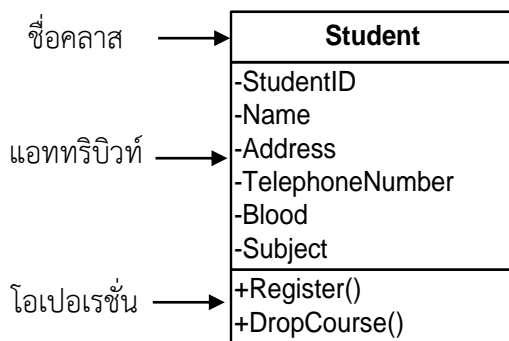
แนวคิดเชิงวัตถุมุ่งเน้นสิ่งต่าง ๆ ที่ใกล้เคียงกับโลกแห่งความจริงในลักษณะรูปธรรมโดยมองระบบเป็นกลุ่มของวัตถุที่มีปฏิสัมพันธ์กันด้วยการนำข้อมูลและฟังก์ชันการทำงานรวมเข้าด้วยกันเป็นวัตถุ ทำให้ข้อมูลที่เป็นวัตถุนั้นสามารถอธิบายคุณสมบัติรวมทั้งฟังก์ชันการทำงานในตัวเองได้

#### 1. วัตถุหรืออ็อบเจกต์ (Object)

คือ ทุกสิ่งที่เราสนใจในเหตุการณ์ใดเหตุการณ์หนึ่งทั้งที่จับต้องได้และจับต้องไม่ได้ เช่น คน สถานที่ เหตุการณ์ หรือรายการต่าง ๆ เป็นต้น วัตถุจะประกอบด้วยคุณสมบัติ กิจกรรม การกระทำ วิธีการ และมีความสัมพันธ์กันระหว่างวัตถุภายในระบบ

#### 2. คลาส (Class)

คือ กลุ่มของวัตถุที่มีโครงสร้างและพฤติกรรมที่เหมือนกัน หรืออาจเปรียบได้ว่า คลาสเปรียบเสมือนแม่พิมพ์ที่ใช้เพื่อสร้างวัตถุ เช่น นิยามคลาสของรถยนต์ว่ามี 4 ล้อ ใช้น้ำมันเป็นเชื้อเพลิง ฉะนั้นจึงสามารถจัดรถยนต์ฮอนด้า รถยนต์โตโยต้า รถยนต์ฟอร์ด เป็นอ็อบเจกต์ของคลาสรถยนต์ได้ แต่ไม่สามารถจัดรถจักรยานยนต์เป็นอ็อบเจกต์ในคลาสรถยนต์ได้เพราะมีคุณสมบัติไม่ตรงกับนิยามที่วางไว้ เป็นต้น คลาสประกอบด้วย ชื่อของคลาส แอททริบิวต์ และโอเปอเรชัน ในภาพที่ 13.1 แสดงคลาส Student ที่มี 6 แอททริบิวต์ และ 2 โอเปอเรชัน



ภาพที่ 13.1 คลาสนักศึกษา (Student)

#### 3. แอททริบิวต์ (Attributes)

คือ คุณสมบัติที่ใช้บรรยายคลาสหรืออ็อบเจกต์โดยคุณสมบัติเหล่านี้จะอยู่ในขอบเขตที่เราสนใจเท่านั้น เช่น จากอ็อบเจกต์รถยนต์ฮอนด้าในคลาสรถยนต์ คุณสมบัติที่สามารถอธิบายคลาสรถยนต์ได้คือ สี ขนาดเครื่องยนต์ รุ่น ความเร็ว เป็นต้น เช่น คลาส Student มีแอททริบิวต์ที่บรรยายคุณสมบัติได้ คือ รหัส

นักศึกษา (StudentID) ชื่อ-สกุล (Name) ที่อยู่ (Address) เบอร์โทรศัพท์ (TelephoneNumber) กรุ๊ปเลือด (Blood) และสาขาวิชา (Subject)

#### 4. การดำเนินการหรือโอเปอเรชัน (Operation) หรือ เมธอด (Method)

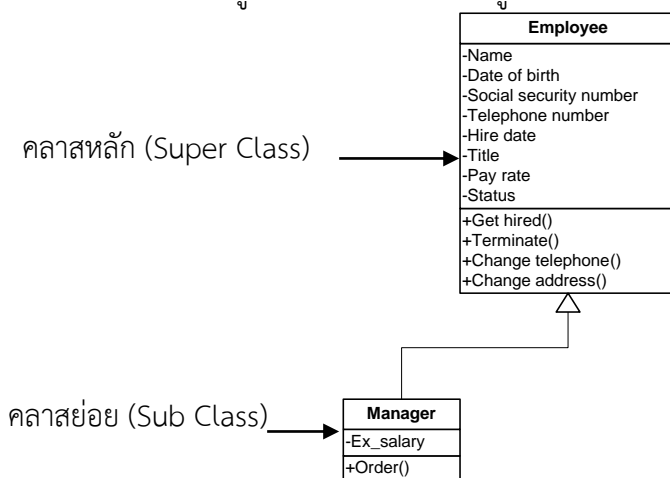
ในอ็อบเจกต์แต่ละอ็อบเจกต์ต้องมีความสามารถในการดำเนินการ หมายถึง การกระทำที่อ็อบเจกต์สามารถกระทำได้หรือสามารถถูกร้องขอให้กระทำได้ ความสามารถในการดำเนินการบางอย่างถูกแสดงออกมาให้เห็นเป็นพฤติกรรมได้ต้องเกิดจากการสื่อสารหรือปฏิสัมพันธ์กันระหว่างอ็อบเจกต์ ในทางการเขียนโปรแกรม เมธอด (Method) เป็นชุดคำสั่งที่เขียนขึ้นมาอย่างเป็นลำดับขั้นตอนเพื่อให้มีการดำเนินการขั้นถัดไป คำว่า “Operation” และ “Method” ในความหมายเดียวกัน เช่น คลาสรถยนต์มีโอเปอเรชัน วิ่ง เลี้ยว สตาร์ท เครื่องยนต์ ดับเครื่องยนต์ เป็นต้น คลาส Student มีความสามารถในการดำเนินการได้ คือ ลงทะเบียน (Register()) ยกเลิกรายวิชา (DropCourse())

#### 5. การสืบทอดคุณสมบัติ (Inheritance)

เป็นวิธีการในการสร้างคลาสใหม่จากคลาสเดิมที่มีอยู่ การสืบทอดคุณสมบัติถือเป็นเรื่องที่สำคัญสำหรับเทคโนโลยีเชิงวัตถุ เนื่องจากการสืบทอดคุณสมบัตินี้ทำให้เกิดข้อดี คือ ทำให้ได้ระบบที่มีโครงสร้างที่เป็นระบบ ปรับเปลี่ยนได้ง่าย และทำให้ลดเวลา ค่าใช้จ่ายในการพัฒนาระบบ การสืบทอดคุณสมบัติของคลาสในระบบทำให้เกิดคลาสย่อย (Subclass) คลาสย่อยจะรับคุณสมบัติของคลาสที่ให้กำเนิด ซึ่งเรียกว่า คลาสหลัก (Super Class) เช่น คลาส Employee เป็นคลาสหลักที่มีแอททริบิวต์ 8 แอททริบิวต์ที่อธิบายคุณสมบัติของคลาส Employee และมีโอเปอเรชัน 4 โอเปอเรชัน จากคลาส Employee สามารถสืบทอดคุณสมบัติของคลาสให้เกิดเป็นคลาสย่อยคือ คลาส Manager ที่มีคุณสมบัติทุกอย่างเหมือนคลาสหลัก แต่มีคุณสมบัติเพิ่มขึ้นมาคือมีแอททริบิวต์เงินประจำตำแหน่ง (Ex\_salary) และโอเปอเรชันออกคำสั่ง (Order) ซึ่งเป็นลักษณะเฉพาะของคลาส Manager ดังแสดงในภาพที่ 13.2

#### 6. โพลิมอร์ฟิซึม (Polymorphism)

หมายถึง คลาสต่างกันสามารถตอบสนองต่อการดำเนินการชื่อเดียวกัน โดยอาจให้พฤติกรรมหรือวิธีการกระทำต่อการดำเนินการนั้นต่างกันได้ เช่น Draw() คือ ฟังก์ชันการวาดซึ่งมีความสามารถในการวาดรูปได้หลายลักษณะมีได้เฉพาะเจาะจงเช่น การวาดรูปวงกลม วงรี สามเหลี่ยม สี่เหลี่ยม เป็นต้น ซึ่งหากต้องการให้มีการตอบสนองรูปวาดต่าง ๆ ผู้ใช้สามารถติดต่อได้ด้วยการใช้ฟังก์ชัน Draw() เพียงฟังก์ชันเดียวส่วนจะมีการตอบสนองการวาดในลักษณะใดนั้นขึ้นอยู่กับรายละเอียดของคำสั่ง ดังนั้นผู้ใช้งานไม่จำเป็นต้องจดจำฟังก์ชันในการวาดมากมายซึ่งแตกต่างจากการเขียนโปรแกรมแบบเดิมที่จำเป็นต้องมีฟังก์ชันการวาดต่าง ๆ ในการวาดรูปเหล่านั้น เช่น ฟังก์ชันวาดรูปสี่เหลี่ยม ฟังก์ชันวาดรูปสามเหลี่ยม ฟังก์ชันวาดรูปวงกลม เป็นต้น



ภาพที่ 13.2 ตัวอย่างการสืบทอดคุณสมบัติ

## 7. เอ็นแคปซูลชัน (Encapsulation)

หมายถึง กระบวนการซ่อนรายละเอียดของคุณลักษณะต่าง ๆ และรายละเอียดการทำงานของคลาสไว้ภายในโดยการที่สิ่งที่อยู่ภายนอกคลาสจะติดต่อกับคลาสได้ต้องติดต่อผ่านทางช่องทางที่คลาสเตรียมไว้ให้เท่านั้น หลักการนี้ทำให้เกิดการมองคลาสนั้น 2 ลักษณะคือ การมองคลาสดังกล่าวจากภายใน และมองคลาสดังกล่าวจากภายนอก ถ้ามองคลาสดังกล่าวจากภายในตัวคลาสนั้นจะเห็นรายละเอียดทั้งหมดของคลาสนั้น แต่ถ้ามองคลาสดังกล่าวจากภายนอกจะเห็นเฉพาะสิ่งที่คลาสนั้นเปิดเผยให้เห็นผ่านทางที่กำหนดเท่านั้น การที่สิ่งที่อยู่ภายนอกคลาสนั้นไม่สามารถเห็นรายละเอียดของคลาสดังกล่าวจากภายนอกได้นั้นเรียกว่า การซ่อนข้อมูล (Information Hiding) ซึ่งหมายถึง การซ่อนรายละเอียดของแอตทริบิวต์และการดำเนินการของคลาสดังกล่าวจากภายนอก ระดับในการเข้าถึงแอตทริบิวต์และโอเปอเรชันของคลาสนั้น มี 3 ระดับ คือ ไพรเวต (Private) โพรเทกต์เท็ด (Protected) และพับลิก (Public)

7.1 ไพรเวต แอตทริบิวต์และ/หรือการดำเนินการของคลาสนั้นที่ถูกกำหนดให้เป็นไพรเวต ไม่ถูกเปิดเผยแก่ภายนอกและไม่สามารถเข้าถึงได้โดยตรงจากภายนอก แต่สามารถเข้าถึงได้จากภายในตัวคลาสนั้นเอง สัญลักษณ์ที่ใช้ระบุในแผนภาพว่าแอตทริบิวต์ใดหรือการดำเนินการใดเป็นไพรเวตคือ เครื่องหมายลบ (-)

7.2 โพรเทกต์เท็ด แอตทริบิวต์และ/หรือการดำเนินการของคลาสนั้นที่ถูกกำหนดให้เป็นโพรเทกต์เท็ด จะไม่ถูกเปิดเผยแก่ภายนอกและไม่สามารถเข้าถึงได้โดยตรงจากภายนอก แต่สามารถเข้าถึงได้จากภายในตัวคลาสนั้นเอง โพรเทกต์เท็ด แอตทริบิวต์และโพรเทกต์เท็ด โอเปอเรชันจะถูกถ่ายทอดไปให้กับคลาสนั้นย่อยและสามารถเข้าถึงได้จากภายในชั้นคลาสนั้น สัญลักษณ์ที่ใช้ในแผนภาพคือ เครื่องหมายชาร์ป (#)

7.3 พับลิก แอตทริบิวต์และ/หรือการดำเนินการของคลาสนั้นที่ถูกกำหนดให้เป็นพับลิกจะถูกเปิดเผยและถูกเข้าถึงได้โดยตรงจากภายนอกไม่มีการปกปิดใด ๆ ทั้งสิ้น รวมทั้งยังสามารถถ่ายทอดไปยังคลาสนั้นย่อยได้ด้วย แอตทริบิวต์และการดำเนินการที่ถูกกำหนดให้เป็นพับลิกสัญลักษณ์ที่ใช้ในแผนภาพคือ เครื่องหมายบวก (+)

ประโยชน์ของการห่อหุ้มแอตทริบิวต์และการดำเนินการ คือ สามารถป้องกันความเสียหายของแอตทริบิวต์และการดำเนินการจากการเข้าถึงของอ็อบเจกต์อื่นโดยไม่ได้รับอนุญาต เนื่องจากถ้ามีการอนุญาตให้อ็อบเจกต์อื่นสามารถเข้าถึงส่วนของโปรแกรมทั้งหมดจะส่งผลให้แอตทริบิวต์และเมธอดนั้นถูกเรียกไปใช้งานอย่างผิด ๆ ได้ง่ายทำให้ค่าของแอตทริบิวต์เปลี่ยนแปลงไปซึ่งอาจทำให้โปรแกรมทำงานผิดพลาดได้

## 13.2 แอ็บสเตรคชัน (Abstraction)

แอ็บสเตรคชันเป็นการมองสิ่งต่าง ๆ แล้วใส่ความคิดรวบยอด (Concept) ให้กับสิ่งที่มองนั้นว่ามีคุณลักษณะที่สำคัญอย่างไรหรือกล่าวได้ว่าแอ็บสเตรคชัน คือ กระบวนการในการสร้างแนวคิดของคลาสดังกล่าวจากกลุ่มของอ็อบเจกต์ที่สนใจ ดังนั้นการมองอ็อบเจกต์หนึ่งชนิดของคนหลายคนจะมีมุมมองที่แตกต่างกันขึ้นอยู่กับความสนใจในสิ่งนั้น แอ็บสเตรคชันเป็นส่วนหนึ่งที่จะช่วยในการวิเคราะห์ถึงปัญหาของระบบงานที่ต้องการพัฒนาซึ่งมีกระบวนการ 4 ประเภท คือ Classification Abstraction Association Abstraction Aggregation Abstraction และ Generalization Abstraction

### 1. Classification Abstraction

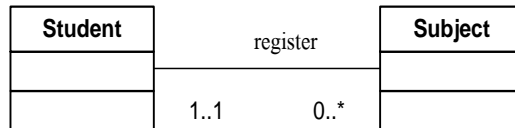
คือ กระบวนการในการให้แนวคิดกับอ็อบเจกต์ที่สนใจ เพื่อก่อให้เกิดแนวคิดของคลาสนั้น หัวใจสำคัญของ Classification Abstraction คือ แนวคิดรวบยอดที่ให้แก่อ็อบเจกต์ การให้แนวความคิดคือ การให้ขอบเขตแก่อ็อบเจกต์ว่าต้องมีคุณลักษณะอะไรบ้าง

### 2. Association Abstraction

Association หมายถึง ความสัมพันธ์ระหว่างคลาสนั้น หรืออ็อบเจกต์ที่อยู่ในระดับเดียวกันคือ คลาสนั้นทั้งสองมีความสำคัญเท่าเทียมกันไม่มีคลาสนั้นใดเป็นองค์ประกอบของคลาสนั้นใด เช่น ลูกค้าจัดทำไปส่งซื้อสินค้า นักศึกษาลงทะเบียนวิชาเรียน สินค้าอยู่ในคลังสินค้า เป็นต้น และกระบวนการในการหาความสัมพันธ์ระหว่าง

คลาสที่สนใจในลักษณะที่คลาสทั้งสองมีความเกี่ยวข้องกันในระดับเดียวกัน เรียกว่า Association Abstraction คลาสที่มีความสัมพันธ์กันถูกเชื่อมความสัมพันธ์ด้วยชื่อความสัมพันธ์ (Association Name) เช่น ลูกค้าจัดทำใบสั่งซื้อ คลาสลูกค้ากับคลาสใบสั่งซื้อถูกเชื่อมความสัมพันธ์ด้วย Association ที่ชื่อว่าจัดทำ เป็นต้น ดังนั้นการเขียนสัญลักษณ์แทนความสัมพันธ์ระหว่างคลาสจึงควรระบุชื่อ Association ไว้ด้วยโดยอาจใช้ลูกศรแสดงให้เห็นทิศทางของความสัมพันธ์

**ตัวอย่าง** แสดง Association Abstraction “นักศึกษาลงทะเบียนวิชาเรียน”

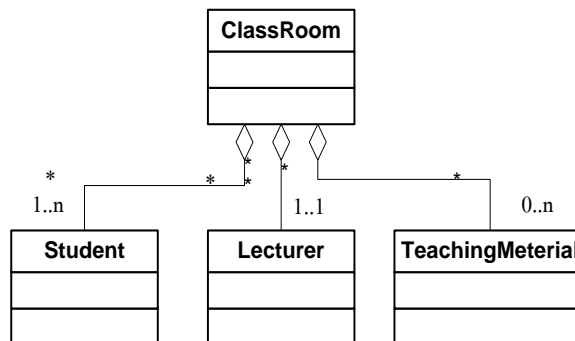


ภาพที่ 13.3 Association Abstraction

### 3. Aggregation Abstraction

เป็นความสัมพันธ์อีกชนิดหนึ่ง โดยที่ Aggregation หมายถึง ความสัมพันธ์ระหว่างคลาสหรืออ็อบเจกต์แบบต่างระดับกัน คือ คลาสหนึ่งมีความสัมพันธ์แบบเป็นองค์ประกอบของอีกคลาสหนึ่งและกระบวนการในการหาความสัมพันธ์ระหว่างคลาสในลักษณะนี้เรียกว่า Aggregation Abstraction สัญลักษณ์ความสัมพันธ์แบบ Aggregation ใช้เส้นตรงหัวข้าวหลามตัดโปร่งเชื่อมทั้งสองคลาสหันหัวข้าวหลามตัดโปร่งไปทางด้านคลาสที่เป็นคลาสหลัก

**ตัวอย่าง** ห้องเรียน (Class) เกิดจากการรวมกันของนักศึกษา (Student) อย่างน้อยหนึ่งคน อาจารย์ผู้สอน (Lecturer) หนึ่งคน และอุปกรณ์การสอน (TeachingMaterial) ซึ่งอาจจะมีหรือไม่มีเลยก็ได้

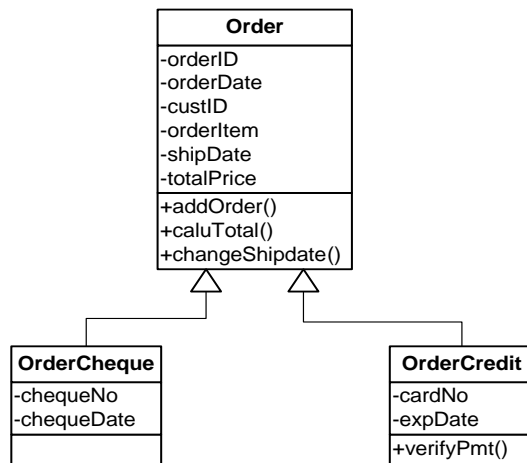


ภาพที่ 13.4 ความสัมพันธ์แบบ Aggregation

### 4. Generalization Abstraction

Generalization หมายถึง ความสัมพันธ์แบบต่างระดับระหว่างคลาสหลัก (Superclass) กับคลาสร้อย (Subclass) โดยที่คลาสร้อยจะสืบทอดคุณลักษณะทั้งแอททริบิวต์และการดำเนินการที่สำคัญของคลาสหลักนั้นมาด้วยทำให้คลาสร้อยมีแอททริบิวต์และการดำเนินการบางอย่างเหมือน กับคลาสหลัก ในขณะเดียวกันคลาสร้อยก็สามารถสร้างแอททริบิวต์และการดำเนินการเพิ่มเติมได้ด้วย

**ตัวอย่าง** ในการสั่งซื้อสินค้า ลูกค้าสามารถเลือกได้ว่าชำระเงินด้วยวิธีใดระหว่างการชำระด้วยเช็ค และชำระด้วยบัตรเครดิต ไม่ว่าลูกค้าจะเลือกชำระเงินด้วยวิธีใดก็จัดเป็นใบสั่งซื้อเหมือนกัน จะแตกต่างกันเพียงรายละเอียดของการชำระเงินเท่านั้น ดังนั้นคลาสใบสั่งซื้อสินค้าจึงสามารถถ่ายทอดคุณลักษณะไปยังคลาสใหม่คือ คลาสใบสั่งซื้อด้วยเช็ค (OrderCheque) และใบสั่งซื้อด้วยบัตรเครดิต (OrderCredit) โดยที่ทั้งสองคลาสใหม่มีแอททริบิวต์และการดำเนินการเพิ่มเติมตามสมควร



ภาพที่ 13.5 แผนภาพแสดง Generalization Abstraction

### 13.3 ภาษายูเอ็มแอล

ยูเอ็มแอล (Unified Modeling Language : UML) เป็นภาษาสัญลักษณ์รูปภาพมาตรฐานที่ใช้เพื่อถ่ายทอดความคิดที่มีต่อระบบให้ออกมาเป็นแผนภาพซึ่งประกอบไปด้วยรูปภาพหรือสัญลักษณ์ตามกฎในการสร้างแผนภาพซึ่งเรียกได้ว่ายูเอ็มแอลเป็นภาษาสำหรับการสร้างแบบจำลองเชิงวัตถุ

#### 13.3.1 ข้อดีของยูเอ็มแอล

ในการใช้ยูเอ็มแอลเป็นภาษามาตรฐานในการวิเคราะห์และออกแบบระบบเชิงวัตถุ จะมีข้อดีหลายประการดังนี้

1. เป็นภาษารูปภาพมาตรฐาน หรือภาษาสากลที่ใช้ในการพัฒนาซอฟต์แวร์เชิงวัตถุและสามารถใช้ในการเปลี่ยนแบบจำลองได้อย่างสื่อความหมายรวมถึงการจัดสร้างเอกสารการวิเคราะห์ออกแบบระบบ การประยุกต์ใช้ยูเอ็มแอลจะทำให้ผู้ร่วมงานมีความเข้าใจและสามารถแลกเปลี่ยนผลของการวิเคราะห์และออกแบบระบบในขั้นตอนต่าง ๆ ได้อย่างรวดเร็วและตรงกัน
2. สามารถนำเสนอและสนับสนุนหลักการเชิงวัตถุได้อย่างครบถ้วน ชัดเจน ทำให้นักพัฒนาระบบสามารถทำความเข้าใจปัญหาและค้นพบวิธีการแก้ไขได้อย่างรวดเร็วและง่ายยิ่งขึ้น
3. ไม่ผูกติดกับภาษาโปรแกรมภาษาใดภาษาหนึ่ง คือสามารถถูกแปลงเป็นระบบจริง ด้วยภาษาเชิงวัตถุใดก็ได้
4. เป็นภาษาที่ง่ายต่อการทำความเข้าใจ
5. สามารถถูกแปลงเป็นภาษาที่ใช้ในการสร้างระบบขึ้นจริงได้อย่างอัตโนมัติทำให้ลดเวลาและค่าใช้จ่ายในการพัฒนาระบบ
6. สนับสนุนการขยายปรับปรุงระบบ
7. ในการพัฒนาสิ่งต่าง ๆ จะถูกบันทึกความคิดของนักพัฒนาในลักษณะของเอกสารที่พร้อมจะนำมาทำความเข้าใจได้อย่างรวดเร็ว

#### 13.3.2 องค์ประกอบของยูเอ็มแอล

องค์ประกอบของภาษายูเอ็มแอลมี 3 ส่วน คือ

1. สัญลักษณ์ทั่วไป (Things) คือสัญลักษณ์พื้นฐานที่ถูกใช้งานในการสร้างไดอะแกรมแบ่งเป็นหมวดย่อยได้ดังนี้

1) หมวดโครงสร้าง (Structural) ได้แก่ ยูเนส คลาส อินเทอร์เฟซ คอมโพเนนต์ คอลเลบอเรชัน และโหนด

2) หมวดพฤติกรรม (Behavioral) คือส่วนที่เป็นไดนามิกของยูเอ็มแอล ซึ่งได้แก่ อินเตอร์แอกชัน สเตตแมชชีน

3) หมวดการจัดกลุ่ม (Grouping) เพื่อใช้ในการรวบรวมองค์ประกอบต่าง ๆ ในโมเดลให้เหมาะสม ได้แก่ แพ็กเกจ

4) หมวดคำอธิบายประกอบ (Annotational) ได้แก่ โน้ต (Note)

2. ความสัมพันธ์ (Relationship) มี 3 ชนิด คือ

1) ความสัมพันธ์แบบพึ่งพา (Dependency Relationship)

2) ความสัมพันธ์แบบเกี่ยวพัน (Association Relationship)

3) ความสัมพันธ์แบบเจเนอรัลไลเซชัน (Generalization Relationship)

3. ไดอะแกรมต่าง ๆ (Diagram)

ยูเอ็มแอลประกอบด้วย 8 ไดอะแกรม แต่ละไดอะแกรมเปรียบเสมือนมุมมองในด้านต่าง ๆ ของระบบที่กำลังพัฒนาซึ่งช่วยให้การวิเคราะห์และออกแบบระบบเป็นไปได้อย่างมีประสิทธิภาพและง่ายยิ่งขึ้น

1) ยูสเคสไดอะแกรม (Use Case Diagram) ใช้ในการจำลองฟังก์ชันการทำงานของระบบ

2) คลาสไดอะแกรม (Class Diagram) ใช้ในการจำลองคลาสต่าง ๆ ที่จำเป็นในระบบ

3) แอกทิวิตีไดอะแกรม (Activity Diagram) มีหลักการเช่นเดียวกับโฟลว์ชาร์ต

4) สเตตชาร์ตไดอะแกรม (Statechart Diagram) ใช้สำหรับแสดงถึงสถานะของอ็อบเจกต์ในระหว่างการทำงาน

5) คอลลาบอเรชันไดอะแกรม (Collaboration Diagram) ใช้แสดงการทำงานร่วมกันของอ็อบเจกต์ในระบบ

6) ซีควเอนซ์ไดอะแกรม (Sequence Diagram) ใช้ในการจำลองกิจกรรมต่าง ๆ ที่เกิดขึ้นกับอ็อบเจกต์ในระบบ

7) คอมโพเนนต์ไดอะแกรม (Component Diagram) ใช้สำหรับสร้างโมเดลของคอมโพเนนต์ในระบบ

8) ดีพลอยเมนต์ไดอะแกรม (Deployment Diagram) ใช้แสดงการติดตั้งใช้งานส่วนประกอบต่าง ๆ ของระบบ

## 13.4 ยูสเคสไดอะแกรม

เมื่อเริ่มต้นการพัฒนาระบบทุกครั้งผู้พัฒนาจะต้องเริ่มต้นที่ขั้นตอนแรก คือการค้นหาและเก็บรวบรวมข้อมูล ความสามารถของระบบที่ผู้ใช้งานต้องการซึ่งถือว่าสำคัญมากในการพัฒนาระบบในมาตรฐานยูเอ็มแอลจะใช้ยูสเคสไดอะแกรมเพื่อเป็นเทคนิคในการจำลองความต้องการของผู้ใช้รวมถึงแสดงความสามารถของระบบ ยูสเคสไดอะแกรมเป็นไดอะแกรมมาตรฐานที่สามารถทำความเข้าใจได้ง่าย

### 13.4.1 ส่วนประกอบสำคัญในยูสเคสไดอะแกรม

ส่วนประกอบที่สำคัญของยูสเคสไดอะแกรมมี 3 ส่วนคือ ยูสเคส (Use Case) แอ็กเตอร์ (Actor) เส้นแสดงความสัมพันธ์ (Relationship) ในการสร้างยูสเคสไดอะแกรมสิ่งสำคัญคือการค้นหาว่าระบบทำอะไรได้บ้าง โดยไม่สนใจว่าจะทำงานอย่างไรหรือใช้เทคนิคการสร้างอย่างไร

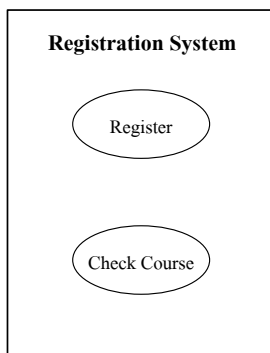
1. ยูสเคส คือ ความสามารถหรือฟังก์ชันที่ระบบจะต้องทำได้จึงจะถือว่าระบบไม่มีข้อผิดพลาด ซึ่งมีคุณสมบัติดังนี้

1) ต้องถูกกระทำโดยแอ็กเตอร์และแอ็กเตอร์เป็นผู้ติดต่อกับระบบตามยูสเคสที่กำหนด

2) ยูสเคสรับข้อมูลจากแอ็กเตอร์และส่งข้อมูลให้แอ็กเตอร์ นั่นคือแอ็กเตอร์ กระทบกับ ยูสเคสโดยการส่งข้อมูลเข้าสู่ระบบตามยูสเคสหรือรับค่าที่ระบบส่งกลับให้

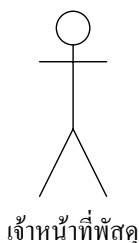
3) ยูสเคสถือว่าการรวบรวมคุณลักษณะความต้องการในระบบอย่างสมบูรณ์ เปรียบเสมือนเป็นการสรุปความต้องการของลูกค้าออกเป็นข้อ ๆ

ยูสเคสในยูเอ็มแอลแทนด้วยรูปวงรี มีชื่อยูสเคสอยู่ข้างใน และทุกยูสเคสจะอยู่ภายใต้ กรอบสี่เหลี่ยมซึ่งหมายถึงระบบ



ภาพที่ 13.6 สัญลักษณ์ของยูสเคส

2. แอ็กเตอร์ คือ ผู้ที่กระทบกับยูสเคสหรือใช้งานยูสเคสอาจเป็นคนหรือไม่ก็ได้ซึ่งจะเป็นผู้ที่ ส่งข้อมูลหรือรับข้อมูล หรือแลกเปลี่ยนข้อมูลข่าวสารกับระบบที่กำลังพัฒนา ชื่อของแต่ละแอ็กเตอร์ไม่ใช่เป็น ชื่อเฉพาะเจาะจงจะไม่บอกว่าผู้ที่ทำอะไรเป็นใครแต่จะบอกถึงประเภทของแอ็กเตอร์หรือบทบาทและ หน้าที่ต่อระบบ ในการหาแอ็กเตอร์ของระบบจะดูที่ใครเป็นผู้ใช้ระบบในฟังก์ชันที่สำคัญ ใครเป็นผู้ดูแลระบบ อุปกรณ์ฮาร์ดแวร์ใดบ้างที่กระทบการเชื่อม ต่อกับระบบ และระบบข้างนอกใดบ้างที่เชื่อมต่อบริการส่งข้อมูลกับ ระบบที่เราสร้าง แอ็กเตอร์ในยูเอ็มแอลแทนด้วยรูปคน (Stick Man)



ภาพที่ 13.7 สัญลักษณ์แอ็กเตอร์เจ้าหน้าที่พัสดุ

3. เส้นแสดงความสัมพันธ์ เป็นการเชื่อมโยงระหว่างยูสเคสและแอ็กเตอร์ เป็นเส้นที่แสดง ความสัมพันธ์ระหว่างยูสเคส มี 2 ชนิด คือ

1) ความสัมพันธ์แบบขยาย (Extend Relationship) ยูสเคสหนึ่งอาจถูกช่วยเหลือโดยการทำงานจากยูสเคสอื่น เช่น ยูสเคสการใส่รหัสอาจถูกช่วยเหลือโดยยูสเคสคำอธิบายการใส่รหัส สัญลักษณ์ใน ยูเอ็มแอลคือลูกศรเส้นประที่ชี้จากยูสเคสแรกไปยังยูสเคสที่ถูกช่วยเหลือหรือถูกขยาย โดยมีคำว่า “extend” อยู่ในเครื่องหมายสเตอริโอไทป์ (stereotype) <<extend>> อยู่ที่กึ่งกลางลูกศร

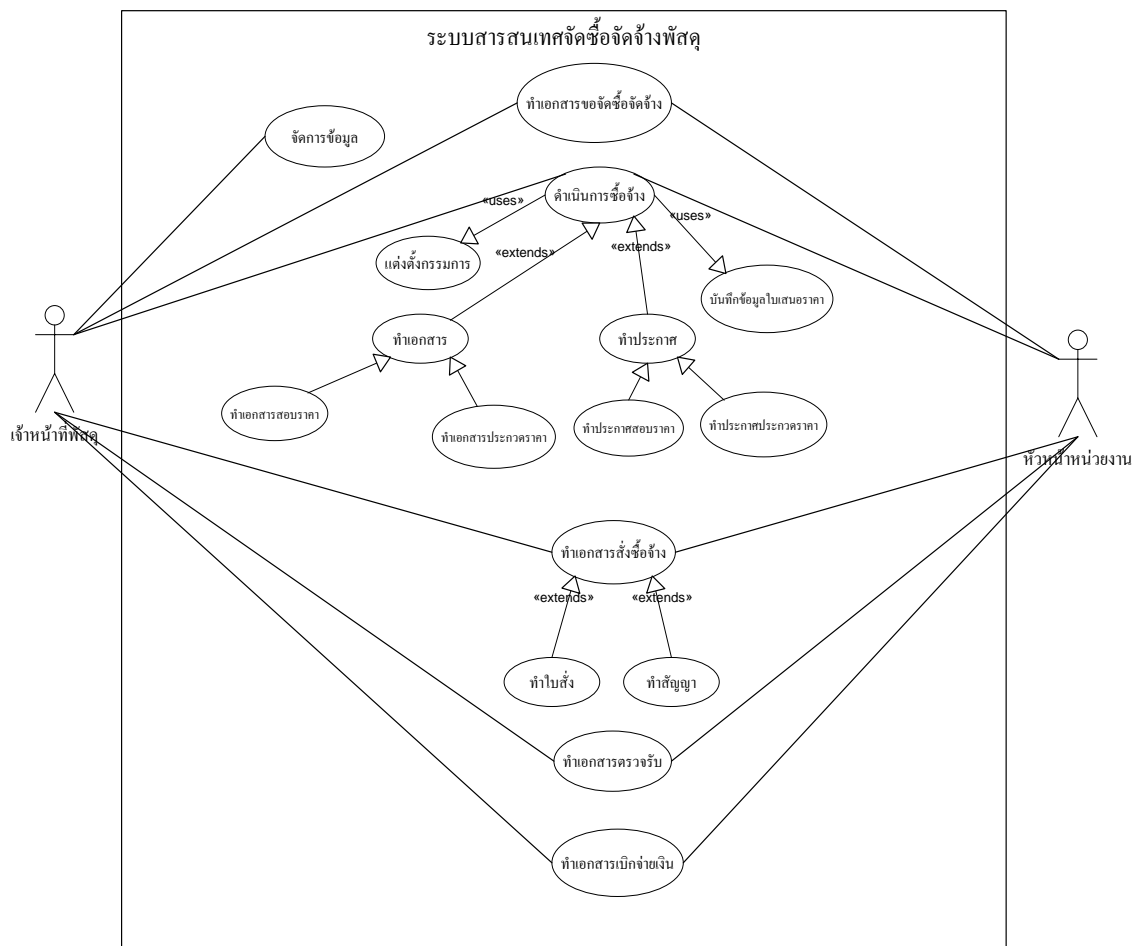
2) ความสัมพันธ์แบบรวม (Include Relationship) ยูสเคสหนึ่งอาจจำเป็นต้องอาศัยการทำงานของยูสเคสอื่น สำหรับยูสเคสที่ถูกเรียกใช้โดยยูสเคสอื่น สัญลักษณ์ในยูเอ็มแอลของความสัมพันธ์ คือ ลูกศรเส้นประชี้ไปยังยูสเคสที่ถูกเรียกใช้หรือถูกรวมไว้ โดยมีคำว่า “Uses” อยู่ในเครื่องหมายสเตอริโอไทป์ (stereotype) <<uses>> อยู่ที่กึ่งกลางลูกศร

**ตัวอย่าง** ระบบสารสนเทศจัดซื้อจัดจ้างพัสดุมีวิธีการทำงาน คือ เมื่อเจ้าหน้าที่พัสดุได้รับเอกสารขอซื้อจ้างจากหน่วยงาน จะจัดทำเอกสารซื้อจ้างซึ่งมีประเภทพัสดุ 2 กลุ่ม คือ วัสดุ และครุภัณฑ์ ในการทำเอกสารจัดซื้อจ้างนั้นเจ้าหน้าที่พัสดุต้องพิจารณาว่าการจัดซื้อจัดจ้างพัสดุนั้นเพื่อนำมาใช้ในแผนงานใดใช้งบประมาณจากหมวดใด จากนั้นเจ้าหน้าที่พัสดุจะพิจารณาว่าใช้วิธีการใดในการจัดซื้อจ้าง โดยแบ่งเป็น 5 วิธี คือ

- การซื้อจ้างวิธีตกลงราคา วงเงินน้อยกว่า 100,000 บาท
- การซื้อจ้างวิธีสอบราคา วงเงินน้อยกว่า 100,000 บาท แต่ไม่เกิน 2,000,000 บาท
- การซื้อจ้างวิธีประกวดราคา วงเงินน้อยกว่า 2,000,000 บาท
- การซื้อจ้างวิธีพิเศษ วงเงินน้อยกว่า 100,000 บาทโดยมีเงื่อนไขพิเศษ เช่น เป็นพัสดุที่ต้องซื้อเร่งด่วนหรือพัสดุที่ใช้ในราชการลับ เป็นต้น

– การซื้อจ้างวิธีกรณีพิเศษ ไม่มีการกำหนดวงเงิน แต่ต้องเป็นการซื้อหรือจ้างจากหน่วยงานราชการ รัฐวิสาหกิจ

ซึ่งสามารถนำเสนอกระบวนการทำงานต่าง ๆ โดยใช้ยูสเคสไดอะแกรม ดังภาพที่ 13.8



ภาพที่ 13.8 แสดงยูสเคสไดอะแกรมระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ

#### 13.4.2 ประโยชน์ของยูสเคสไดอะแกรม

ยูสเคสไดอะแกรมจะจำลองการทำงานต่าง ๆ ของระบบ ซึ่งจะช่วยให้สามารถมองระบบได้อย่างชัดเจนขึ้น ยูสเคสไดอะแกรมมีประโยชน์สรุปได้ดังนี้

1. เพื่อให้ผู้พัฒนาทราบถึงความสามารถของระบบว่าต้องทำอะไรได้บ้าง
2. เพื่อทราบถึงผู้ใช้งานในแต่ละส่วนของระบบ



3. ทำให้การติดต่อสื่อสารระหว่างผู้พัฒนากับลูกค้าหรือระหว่างผู้พัฒนาด้วยกันทำได้ง่าย

4. ใช้ในการทดสอบระบบซอฟต์แวร์ ว่าทำงานได้ครบถ้วนตามความต้องการหรือไม่  
เนื่องจากนักพัฒนาส่วนใหญ่มักไม่มีแนวทางหรือขั้นตอนในการทดสอบอย่างเป็นระบบระเบียบชัดเจนไม่รู้ว่า  
จะต้องเริ่มทดสอบส่วนใดก่อน

### 13.5 คลาสไดอะแกรม

คลาสไดอะแกรมเป็นไดอะแกรมที่แสดงการใช้งานคลาส อ็อบเจกต์ และมีการสร้างความสัมพันธ์  
ระหว่างคลาสหรืออ็อบเจกต์เหล่านั้น เช่น การสืบทอดคุณสมบัติของคลาส เป็นต้น การหาคลาสของอ็อบเจกต์  
ได้ต้องสามารถจัดหมวดหมู่ของอ็อบเจกต์ได้ การหาคลาสจากอ็อบเจกต์ควรให้อยู่ในระบบที่กำลังสร้างเช่น  
ระบบจัดซื้อสามารถหาคลาสของระบบได้คือ คลาสลูกค้า คลาสใบสั่งซื้อ คลาสใบเสนอราคา คลาส  
ใบเสร็จรับเงิน เป็นต้น

#### 13.5.1 การสร้างคลาสไดอะแกรม

วัตถุประสงค์ของการสร้างคลาสไดอะแกรมเพื่อแสดงถึงโครงสร้างของระบบที่ประกอบด้วย  
คลาส และความสัมพันธ์ระหว่างคลาส คลาสไดอะแกรมถือว่าเป็นไดอะแกรมที่มีความสำคัญมากเพราะถูกใช้  
เป็นไดอะแกรมหลักในการสร้างไดอะแกรมอื่นอีกหลายประเภทสิ่งสำคัญในการสร้างคลาสไดอะแกรม คือ การ  
ค้นหาแนวคิดต่าง ๆ ที่อยู่ในขอบข่ายของระบบที่กำลังสนใจ มีวิธีการในการค้นหาคลาสดังนี้

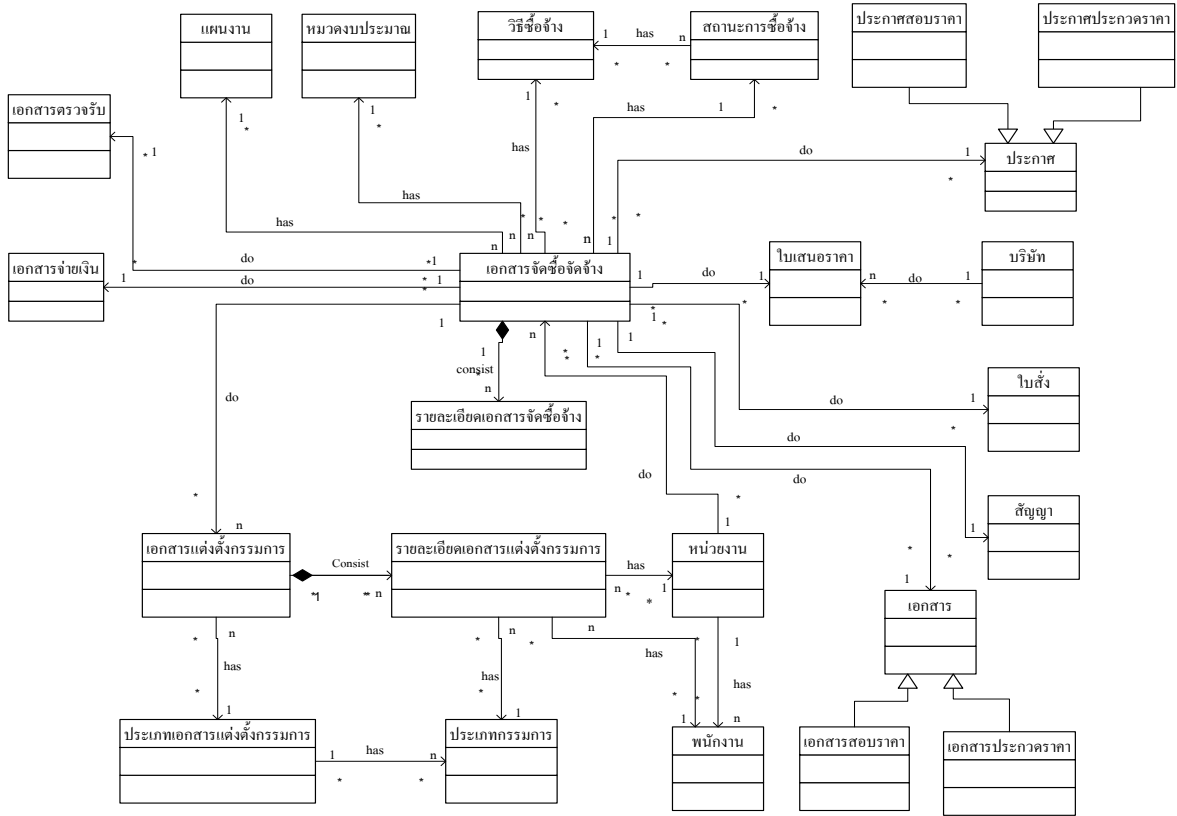
1. คำนามที่ปรากฏอยู่ในคำบรรยายยูสเคสจะถูกสร้างเป็นคลาส เช่น คลาสรถยนต์ คลาสวิชา  
เรียน คลาสหนังสือ คลาสสินค้า เป็นต้น
2. คำวิเศษณ์ที่ปรากฏอยู่ในคำบรรยายยูสเคสจะถูกสร้างเป็นแอททริบิวต์ เช่น สีรถ รุ่นรถ  
ยี่ห้อรถ เป็นต้น
3. คำกริยาที่ปรากฏอยู่ในคำบรรยายยูสเคสจะถูกสร้างเป็นโอเปอเรชัน เช่น สตาร์ทรถ เบรก  
ลงทะเบียน ยกเลิกรายวิชา เป็นต้น

#### 13.5.2 สัญลักษณ์

ตามมาตรฐานยูเอ็มแอล คลาสไดอะแกรมประกอบไปด้วยสัญลักษณ์ของคลาสและเส้นแสดง  
ความสัมพันธ์ สัญลักษณ์คลาสประกอบด้วย 3 ส่วนคือ ชื่อคลาส (Class Name) แอททริบิวต์และโอเปอเรชัน  
ความสัมพันธ์ระหว่างคลาส ซึ่งความสัมพันธ์เหล่านี้สามารถเป็นได้ 3 รูปแบบ ดังนี้

1. ความสัมพันธ์แบบพึ่งพิง (Dependency) ความสัมพันธ์แบบนี้เกิดขึ้นเมื่อการเปลี่ยนแปลง  
ที่เกิดขึ้นกับคลาสที่ถูกพึ่งพิง (Independent Class) ส่งผลต่อคลาสที่พึ่งพิง (Dependent Class) คลาส  
ดังกล่าว การจำลองความสัมพันธ์แบบนี้สามารถทำได้โดยวาดเส้นตรงแบบประที่มีหัวลูกศรเป็นเส้นโปร่งชี้จาก  
คลาสรองที่พึ่งพิงไปยังคลาสหลักที่ถูกพึ่งพิง
2. ความสัมพันธ์แบบถ่ายทอด (Generalization) คือความสัมพันธ์ระหว่างคลาสหลักและ  
คลาสรองนั่นเอง การจำลองความสัมพันธ์แบบนี้สามารถทำได้โดยวาดเส้นตรงทึบที่มีหัวลูกศรเป็นสี่เหลี่ยมโปร่ง  
ชี้จากคลาสรองไปยังคลาสหลัก
3. ความสัมพันธ์แบบเชื่อมโยง (Association) เป็นความสัมพันธ์อีกชนิดหนึ่งระหว่างคลาส

**ตัวอย่าง** จากระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ สามารถค้นหาคลาสของระบบ และแสดงความสัมพันธ์  
ระหว่างคลาสต่าง ๆ ได้ดังภาพที่ 13.9

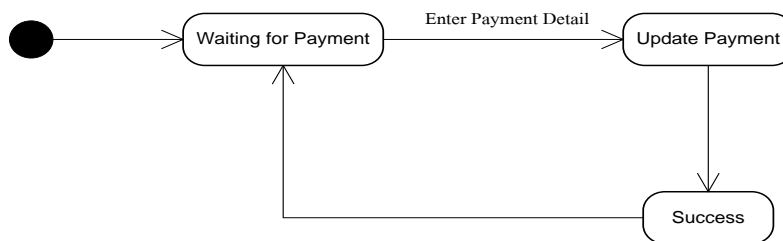


ภาพที่ 13.9 คลาสไดอะแกรมระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ

### 13.6 สเตตชาร์ตไดอะแกรม

สเตตชาร์ตไดอะแกรมบอกถึงพฤติกรรมของคลาสต่าง ๆ ในระบบว่ามีสถานะอะไรบ้างจะเปลี่ยนสถานะเมื่อเกิดเหตุการณ์อะไร สเตตชาร์ตไดอะแกรมของแต่ละคลาสประกอบไปด้วยสถานะที่สามารถเกิดขึ้นได้ เช่น คนอยู่ในสถานะกำลังเดิน รถอยู่ในสถานะกำลังวิ่ง เป็นต้น เมื่อเวลาผ่านไปหรือมีเหตุการณ์บางอย่างเกิดขึ้นย่อมทำให้เกิดการเปลี่ยนสถานะหรือเปลี่ยนพฤติกรรมได้ สเตตชาร์ตไดอะแกรมในยูเอ็มแอลมีจุดเริ่มต้นสถานะและจุดสิ้นสุดสถานะ โดยจุดเริ่มต้นมีสัญลักษณ์เป็นรูปวงกลมทึบและจุดสิ้นสุดสถานะเป็นรูปวงกลมโปร่งล้อมรอบวงกลมทึบข้างใน ส่วนสถานะในไดอะแกรมถูกแสดงเป็นรูปสี่เหลี่ยมหัวมนรูปร่างเหมือนแคปซูลและเชื่อมกันด้วยเส้นลูกศรชี้จากสถานะหนึ่งไปยังอีกสถานะหนึ่งสามารถเขียนคำอธิบายเหตุการณ์ที่ทำให้เปลี่ยนสถานะตรงเส้นลูกศรได้ ดังตัวอย่างภาพที่ 13.10

**ตัวอย่าง** สถานะของการรอชำระเงิน คือเมื่อมีการชำระเงินและป้อนรายละเอียดต่าง ๆ สถานะก็จะเปลี่ยนจากการรอเป็นปรับปรุงข้อมูลและเสร็จสมบูรณ์

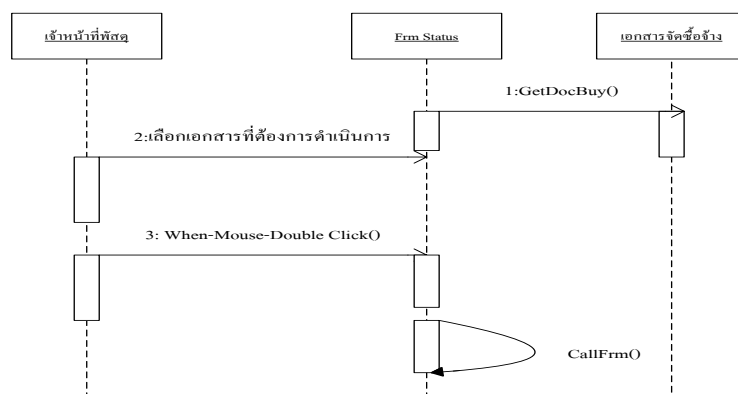


ภาพที่ 13.10 สเตตชาร์ตไดอะแกรม

### 13.6 ซีควเอนไดอะแกรม

ซีควเอนไดอะแกรมบ่งบอกถึงในยูสเคสหนึ่งวัตถุแต่ละตัวจะติดต่อสื่อสารกันอย่างไร มีขั้นตอนการทำงานอย่างไร โดยเน้นไปที่แกนเวลาเป็นสำคัญถ้าเวลาเปลี่ยนขั้นตอนการทำงานจะเปลี่ยนโดยมีแอ็กเตอร์เป็นผู้กระทำเริ่มต้น ในยูเอ็มแอลซีควเอนไดอะแกรมมีแกนสมมติ 2 แกนคือ แกนนอนและแกนตั้ง แกนนอนแสดงขั้นตอนการทำงานและการส่งข้อความของแต่ละวัตถุว่าต้องทำอะไรเมื่อใด แกนตั้งเป็นแกนเวลาโดยแกนนอนและแกนตั้งต้องสัมพันธ์กัน ในซีควเอนไดอะแกรมมีสัญลักษณ์ของวัตถุหรือคลาสแทนรูปสี่เหลี่ยมเรียงกันตามแนวนอน ภายในบรรจุชื่ออ็อบเจ็กต์ตามด้วยเครื่องหมายทวิภาค ( : ) และชื่อคลาส เส้นประที่อยู่แนวแกนเวลาแสดงถึงชีวิตของวัตถุ สี่เหลี่ยมแนวตั้งที่อยู่ตำแหน่งเดียวกับวัตถุหรือคลาสเรียกว่า แอ็กทิเวชัน (Activation) ซึ่งใช้แสดงช่วงเวลาวัตถุกำลังปฏิบัติงาน และเส้นที่ทำหน้าที่ส่งข้อมูลระหว่างวัตถุ

**ตัวอย่าง** จากตัวอย่างระบบสารสนเทศจัดซื้อจัดจ้างพัสดุการทำงานของยูสเคสการดำเนินการซื้อจ้างซึ่งมีขั้นตอนการทำงานคือ เจ้าหน้าที่พัสดุดับเบิ้ลคลิกเมาส์ที่เอกสารขอซื้อจ้างพัสดุที่ต้องการดำเนินการก็จะไปที่หน้าจอเอกสารนั้น แสดงเป็นซีควเอนไดอะแกรมได้ดังภาพที่ 13.11

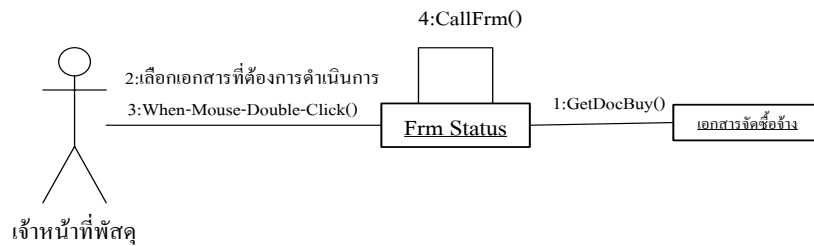


ภาพที่ 13.11 ซีควเอนไดอะแกรมการดำเนินการซื้อจ้าง

### 13.7 คอลลาบอเรชันไดอะแกรม

มีหน้าที่เดียวกันกับซีควเอนไดอะแกรมแต่ไม่แสดงถึงแกนเวลาอย่างชัดเจนยกเว้นการโต้ตอบกันระหว่างอ็อบเจ็กต์สัญลักษณ์ที่ใช้ประกอบด้วย วัตถุ หรือคลาสแทนด้วยรูปสี่เหลี่ยมคล้ายซีควเอนไดอะแกรมมีรูปแบบคือ ชื่ออ็อบเจ็กต์/บทบาท : ชื่อคลาสและขีดเส้นใต้เพื่อแสดงว่าเป็นอินสแตนซ์ แต่ไม่จำเป็นต้องเรียงตามแนวนอนเหมือนในซีควเอนไดอะแกรมมีเส้นเชื่อมกันระหว่างวัตถุ เรียกว่า ลิงก์ (Link) ซึ่งแต่ละลิงก์มีคำอธิบายแสดงขั้นตอนการทำงานตามทิศทางลูกศรโดยมีตัวเลขลำดับกำกับไว้เพื่อบอกว่าขั้นตอนใดทำก่อนทำหลังซึ่งแทนแกนเวลาตามด้วยเครื่องหมายทวิภาคและเมสเสจ ในส่วนของลำดับย่อมนั้นคอลลาบอเรชันไดอะแกรมจะใช้ตัวเลขและเติมจุดย่อแล้วใส่ตัวเลขต่อท้ายเหมือนทศนิยมเพื่อให้รู้ว่าขั้นตอนนี้เป็นการทำงานย่อยของเลขลำดับใด คอลลาบอเรชันไดอะแกรมใช้ในการออกแบบกระบวนการทำงานที่แสดงถึงลำดับของการโต้ตอบกันระหว่างอ็อบเจ็กต์ นั่นคือ แสดงถึงกลุ่มของอ็อบเจ็กต์ที่ทำงานร่วมกันสอดคล้องกับความหมายของชื่อไดอะแกรม ลูกศรที่ชี้จะชี้ไปในทิศทางเดียวไม่มีการชี้ย้อนกลับในเส้นเดียวกัน ตัวเลขที่กำกับข้างหน้าเป็นการบอกลำดับขั้นการทำงานว่าใครส่งก่อนหรือหลังและยังบอกว่ากระบวนการใดที่มีการจัดลำดับเป็นอนุกรมคือต้องทำขั้นตอนนี้เสร็จก่อนจึงสามารถทำขั้นต่อไปได้ตัวเลขที่กำกับข้างหน้าถูกแบ่งย่อยเป็นทศนิยมแต่ถ้ากระบวนการใดสามารถทำพร้อมกันได้ก็ไม่ต้องแยกย่อยเป็นทศนิยมเพิ่มขึ้นอีกให้อยู่ในระดับเดียวกันได้ ในส่วนของการวนซ้ำแสดงด้วยเครื่องหมาย \*[ ] และในส่วนของเงื่อนไขแสดงภายในเครื่องหมายวงเล็บกำกับ

**ตัวอย่าง** จากตัวอย่างระบบสารสนเทศจัดซื้อจัดจ้างพัสดุการทำงานของยูสเคสการดำเนินการซื้อจ้างซึ่งมีขั้นตอนการทำงานคือ เจ้าหน้าที่พัสดุดับเบิ้ลคลิกเมาส์ที่เอกสารขอซื้อจ้างพัสดุที่ต้องการดำเนินการก็จะไปที่หน้าจอเอกสารนั้น สามารถแสดงคอลลาบเลชั่นไดอะแกรมทำเอกสารดำเนินการซื้อจ้าง ดังภาพที่ 13.12

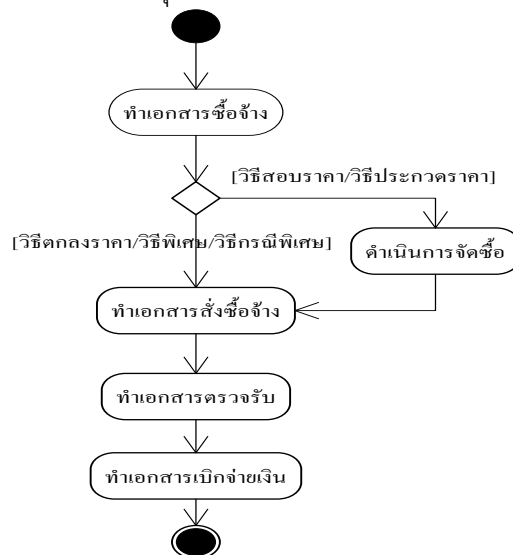


ภาพที่ 13.12 คอลลาบเลชั่นไดอะแกรมการดำเนินการซื้อจ้าง

### 13.8 แอ็กทिवิตีไดอะแกรม

แอ็กทिवิตีไดอะแกรมแสดงขั้นตอนการทำงานของยูสเคสเช่นเดียวกับซีเคเวนไดอะแกรมและคอลลาบเลชันไดอะแกรมแต่นั่นที่งานย่อยของวัตถุ ส่วนแอ็กทिवิตีไดอะแกรมจะเปลี่ยนสถานะได้โดยไม่ต้องมีเหตุการณ์ที่กำหนดไว้ในไดอะแกรมมาก่อนแต่เปลี่ยนสถานะเองตามกระบวนการทำงานคล้ายกับผังงานโปรแกรม สัญลักษณ์ในแอ็กทिवิตีไดอะแกรม มีแอ็กทिवิตีคือกิจกรรมที่แสดงด้วยสี่เหลี่ยมมนเหมือนแคปซูล เชื่อมโยงกันด้วยลูกศรเพื่อแสดงลำดับการทำงาน และมีเส้นทึบหนาในแนวอนใช้วาดในกรณีที่ต้องรอแอ็กทिवิตีอื่นเสร็จหมดก่อนจึงทำแอ็กทिवิตีถัดไปได้ โดยมีเส้นลูกศรเข้ามารวมกันที่จุดเดียวสวิมแลนด์ (Swimlanes) เป็นการแบ่งกลุ่มแอ็กทिवิตีเป็นเลนเหมือนสระว่ายน้ำโดยแบ่งเป็นช่องในแนวดิ่งและกำหนดแต่ละช่องด้วยชื่อของอ็อบเจกต์ไว้แถวบนสุด

**ตัวอย่าง** จากระบบสารสนเทศจัดซื้อจัดจ้างพัสดุแสดงภาพรวมของระบบได้โดยใช้แอ็กทिवิตีไดอะแกรม



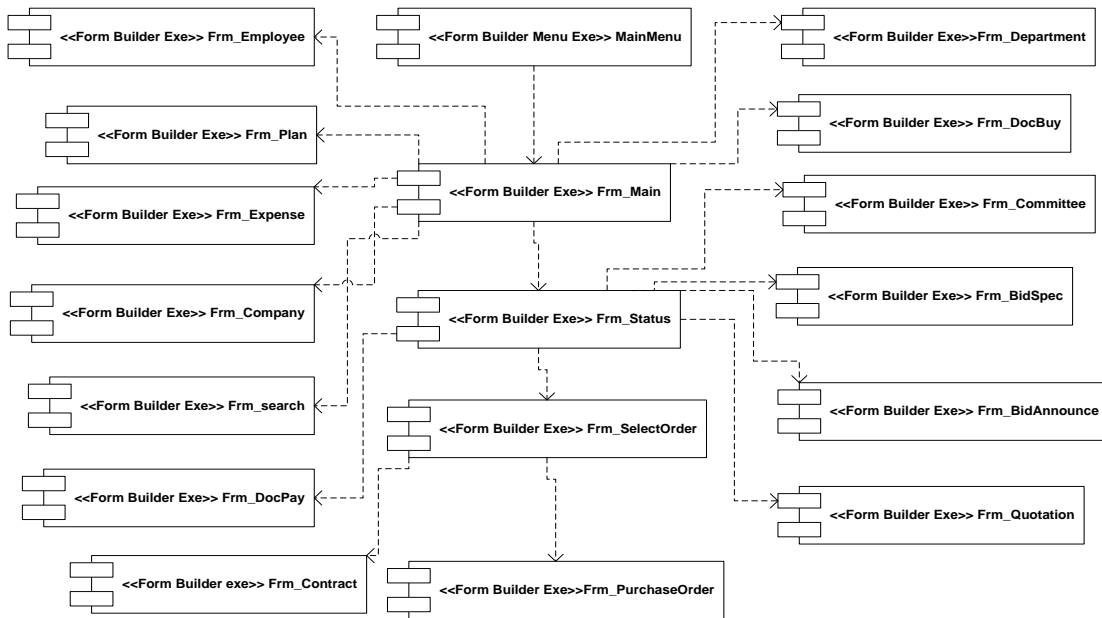
ภาพที่ 13.13 แอ็กทिवิตีไดอะแกรมภาพรวมระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ

### 13.9 คอมโพเนนต์ไดอะแกรม

แสดงความสัมพันธ์ที่เชื่อมต่อกันระหว่างซอฟต์แวร์คอมโพเนนต์ในระบบว่าประกอบด้วยไฟล์อะไรบ้าง ซึ่งอาจเป็นไฟล์ซอร์สโค้ด (Source Code) ไฟล์ไบนารี (Binary Code) และไฟล์เอ็กซิคิวต์ (Executable Code) การตั้งชื่อของคอมโพเนนต์ในคอมโพเนนต์ไดอะแกรมจะใช้ชื่อของคลาสจากคลาสไดอะแกรมไม่ใช่ชื่อ

ของอินสแตนซ์ (Instance) สัญลักษณ์ของคอมโพเนนต์ในคอมโพเนนต์ไดอะแกรมถูกแสดงเป็นสี่เหลี่ยม ประกอบด้วยสี่เหลี่ยมเล็ก 2 รูปติดอยู่ที่ขอบด้านซ้าย และอาจเชื่อมต่อกันด้วยเส้นแสดงความสัมพันธ์

**ตัวอย่าง** จากระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ สามารถสร้างคอมโพเนนต์ไดอะแกรมที่แสดงองค์ประกอบของระบบซอฟต์แวร์ที่ประกอบด้วยไฟล์ที่สร้างโดย Oracle Developer ซึ่งมีส่วนขยายของไฟล์ คือ .fmx และ .mmx โดยไฟล์ทั้งหมดทำงานอยู่บนไฟล์เมนูดังนี้



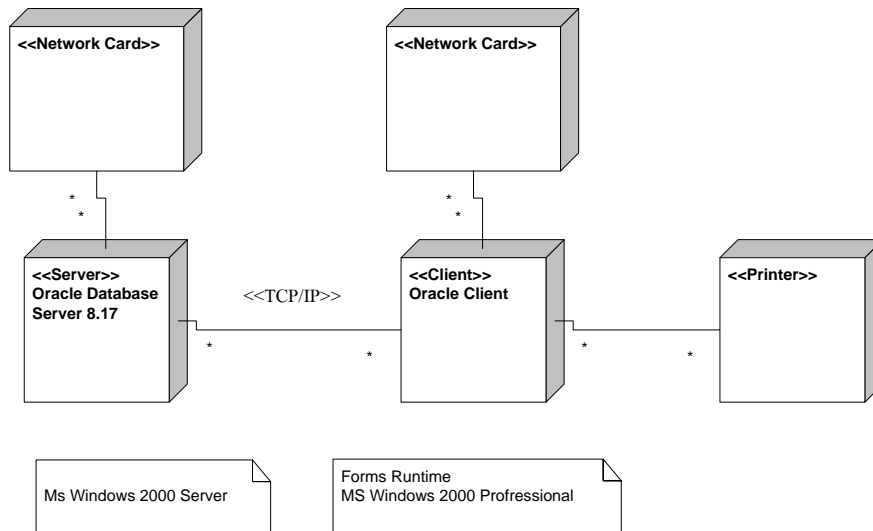
ภาพที่ 13.14 คอมโพเนนต์ไดอะแกรมระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ

### 13.10 ดีพลอยเมนต์ไดอะแกรม

แสดงการเชื่อมต่อของอุปกรณ์ฮาร์ดแวร์ในระบบและมักใช้ร่วมกับคอมโพเนนต์ไดอะแกรมโดยข้างใน ฮาร์ดแวร์อาจประกอบไปด้วยซอฟต์แวร์คอมโพเนนต์ ดีพลอยเมนต์ไดอะแกรมแสดงอยู่ในรูปอินสแตนซ์ และแสดงในช่วงเวลาของการรันหรือระหว่างการทำงานเอ็กซ์ิคิวต์ ดังนั้นไฟล์คอมโพเนนต์ของระบบที่ไม่ได้ใช้สำหรับรันจะไม่ปรากฏในไดอะแกรมนี้แต่มีในคอมโพเนนต์ของไฟล์ที่ใช้ทำงานจริงเท่านั้น

สัญลักษณ์ของดีพลอยเมนต์ไดอะแกรมเป็นการเชื่อมกันระหว่างโหนดซึ่งคือฮาร์ดแวร์ก็จะบรรจุอินสแตนซ์ของซอฟต์แวร์คอมโพเนนต์ที่ถูกแสดงด้วยสัญลักษณ์ของคอมโพเนนต์ไว้ข้างใน แต่ละคอมโพเนนต์เชื่อมต่อกันโดยใช้ความสัมพันธ์แบบพึ่งพิงโดยชี้จากคอมโพเนนต์ที่ขอใช้บริการไปยังคอมโพเนนต์อื่นเหมือนกับคอมโพเนนต์ไดอะแกรมโดยบนลูกศรอาจมีคำกำกับอยู่ภายใต้เครื่องหมายสเตอร์ไอโอบีเพื่อสร้างความกระจ่างยิ่งขึ้นหากจำเป็น สัญลักษณ์ของโหนดถูกแสดงด้วยรูปลูกบาศก์ 3 มิติ ภายในบรรจุชื่อที่แสดงถึงประเภทของโหนดถ้าเป็นโหนดอินสแตนซ์จะมีทั้งชื่อจริงและประเภทของโหนดและต้องขีดเส้นใต้ที่ชื่อของโหนดเพื่อแสดงว่าเป็นอินสแตนซ์หรืออ็อบเจกต์ ชื่อของโหนดเป็นชื่อเฉพาะ ส่วนประเภทของโหนดเป็นการบ่งบอกว่าโหนดนั้นคืออุปกรณ์ชนิดอะไร โหนดอาจถูกเชื่อมต่อกับโหนดอื่นได้ ซึ่งแสดงว่าอุปกรณ์แต่ละตัวมีการติดต่อสื่อสารกันอย่างไรผ่านระบบเครือข่าย อาจมีคำกำกับภายใต้สัญลักษณ์สเตอร์ไอโอบีเพื่อบอกว่าทั้ง 2 โหนดนี้เชื่อมต่อกันด้วยการสื่อสารวิธีใดซึ่งก็คือชนิดของช่องสัญญาณของเครือข่ายหรือโปรโตคอล เช่น <<TCP/IP>> เป็นต้น

ตัวอย่าง จากระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ สามารถสร้างดีพลอยเมนต์ไดอะแกรมที่แสดงการออกแบบสถาปัตยกรรมทางกายภาพของระบบที่เป็นไคลเอนต์/เซิร์ฟเวอร์ ดังนี้



ภาพที่ 13.15 ดีพลอยเมนต์ไดอะแกรมระบบสารสนเทศจัดซื้อจัดจ้างพัสดุ

### 13.11 บทสรุป

การวิเคราะห์และออกแบบระบบเชิงวัตถุเป็นแนวคิดที่พยายามพัฒนาระบบโดยนำโปรแกรมเดิมที่มีอยู่กลับมาใช้งานใหม่ การวิเคราะห์ระบบเชิงวัตถุอาศัยแนวคิดเชิงวัตถุที่ประกอบด้วย วัตถุ คลาส แอททริบิวต์ โอเปอเรชัน/เมธอด การสืบทอดคุณสมบัติ โพลิมอร์ฟิซึม และเอ็นแคปซูลชัน ในการมองสิ่งต่าง ๆ ภายในระบบนักวิเคราะห์แต่ละคนมีวิธีการมองที่ไม่เหมือนกัน ดังนั้นจึงต้องมีกระบวนการสร้างแนวความคิดของคลาสจากกลุ่มของอ็อบเจกต์ ซึ่งเรียกว่า แอ็บสเตรคชัน (Abstraction) มีกระบวนการที่สำคัญ 4 กระบวนการคือ Classification Abstraction Association Abstraction Aggregation Abstraction และ Generalization Abstraction ซึ่งทำให้สามารถนิยามคลาสได้ทั้งหมดในระบบเพื่อดำเนินการในการพัฒนาระบบต่อไป

ยูเอ็มแอลเป็นภาษาสัญลักษณ์รูปภาพมาตรฐานที่ใช้เพื่อถ่ายทอดความคิดที่มีต่อระบบให้ออกมาเป็นแผนภาพประกอบด้วยรูปภาพหรือสัญลักษณ์ตามกฎการสร้างแผนภาพยูเอ็มแอลเป็นภาษาสำหรับการสร้างแบบจำลองเชิงวัตถุ ไดอะแกรมของยูเอ็มแอลประกอบด้วย 8 ไดอะแกรม โดยแต่ละไดอะแกรมเปรียบเสมือนมุมมองในด้านต่าง ๆ ของระบบที่กำลังพัฒนาช่วยให้การวิเคราะห์และออกแบบระบบเป็นไปได้อย่างมีประสิทธิภาพ ยูสเคสไดอะแกรม คือไดอะแกรมเชิงพฤติกรรมซึ่งแสดงถึงกลุ่มของยูสเคส แอ็กเตอร์ และความสัมพันธ์ ซึ่งเป็นมุมมองภายนอกของระบบ คลาสไดอะแกรม คือไดอะแกรมเชิงโครงสร้างที่แสดงถึงกลุ่มของคลาสและความสัมพันธ์ของคลาส บีเฮฟเวียอร์ไดอะแกรม (Behavioral Diagram) คือไดอะแกรมที่บ่งบอกพฤติกรรมของตัวระบบ ได้แก่ สเตตชาร์ตไดอะแกรม ซีควเอนซ์ไดอะแกรม คอลแลบอเรชันไดอะแกรม และแอ็กทิวิตีไดอะแกรม กลุ่มอิมพลีเมนเตชันไดอะแกรม (Implementation Diagram) ประกอบด้วยสัญลักษณ์ที่ใช้แสดงถึง โครงสร้างของซอร์สโค้ดหรือไฟล์และโครงสร้างของส่วนประกอบที่เชื่อมต่อกันในระบบ ซึ่งคือฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในระบบนั่นเองได้แก่ คอมโพเนนต์ไดอะแกรม และดีพลอยเมนต์ไดอะแกรม

## คำถามทบทวน

1. จงอธิบายความหมายของคำต่อไปนี้
  - 1.1 Object
  - 1.2 Class
  - 1.3 Method
  - 1.4 Polymorphism
  - 1.5 Encapsulation
2. จงอธิบายหลักการวิเคราะห์และออกแบบระบบเชิงวัตถุ
3. ยูเอ็มแอล คืออะไร มีข้อดีอย่างไร
4. จากกรณีศึกษาต่อไปนี้ ให้สร้างยูสเคสไดอะแกรม และคลาสไดอะแกรม

คณะวิทยาศาสตร์ของสถาบันการศึกษาแห่งหนึ่งมีบุคลากรหลายประเภท ได้แก่ อาจารย์ นักศึกษา และเจ้าหน้าที่ โดยอาจารย์แต่ละท่านมีหน้าที่ในการสอนอย่างน้อยหนึ่งวิชา นักศึกษามีหน้าที่ในการศึกษาวิชาใดวิชาหนึ่งหรือมากกว่า 1 วิชา ในขณะที่เจ้าหน้าที่ คือ เจ้าหน้าที่ห้องปฏิบัติการต่าง ๆ ซึ่งใน 1 ห้องต้องมีเจ้าหน้าที่อย่างน้อย 1 คนดูแล