

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

การวิจัยโครงการ การพัฒนาระบบประกันอุบัติเหตุออนไลน์ โดยแอปพลิเคชัน แอนดรอยด์ ได้ทำการค้นคว้า รวบรวมข้อมูล และทฤษฎีที่เกี่ยวข้อง ดังต่อไปนี้

1. ความเป็นมาของประกันชีวิต
2. ระบบปฏิบัติการแอนดรอยด์
3. วงจรการพัฒนาาระบบ (System Development Life Cycle: SDLC)
4. การวิเคราะห์และออกแบบระบบด้วย UML (Unified Modeling Language)
5. ความรู้พื้นฐานเกี่ยวกับ Java 2 Enterprise Edition (J2EE)
6. ความรู้เบื้องต้นเกี่ยวกับ MySQL
7. การประเมินเครื่องมือในการวิจัย
8. งานวิจัยที่เกี่ยวข้อง

ความเป็นมาของประกันชีวิต

รายละเอียดต่อไปนี้เป็นารแสดงของรายละเอียดความเป็นมาของประกัน (พุทธชงษ์ นิจบรรณ)

ปัจจุบันภัยอันตรายต่างๆ เกิดขึ้นได้ตลอดเวลาให้กับทุกคน และดูเหมือนชีวิตจะมีการเสี่ยงภัยกันมากขึ้นทุกวัน เช่น การเสี่ยงภัยจากมลพิษโรคร้ายไข้เจ็บต่างๆ และภัยอันสยองอันเกิดจากการประสบอุบัติเหตุต่างๆ ซึ่งเกิดขึ้นไม่ละเว้นในแต่ละวัน และตราบดีที่มนุษย์ยังไม่สามารถเอาชนะภัยเหล่านี้ได้ ก็ย่อมเกิดความหวาดกลัว อันเป็นสัญชาตญาณของมนุษย์อยู่ เพราะไม่ว่าภัยจะเกิดขึ้นกับตนและครอบครัวเมื่อใด ซึ่งผู้ที่เป็นหัวหน้าครอบครัวก็ย่อมต้องเป็นผู้ที่มีความห่วงใยบุคคลและสมาชิกในครอบครัวมากขึ้น ปัจจุบันนี้เหตุการณ์ผันแปรไม่แน่นอนซึ่งเป็นอุทากรณ์สอนใจให้คิดอยู่เสมอๆ ขณะเดียวกันวันข้างหน้าก็ยังไม่ทราบว่าจะเคราะห์กรรมที่มนุษย์ไม่สามารถหลีกเลี่ยงได้จะเกิดขึ้นกับตนเมื่อใด มนุษย์ผู้มีความเชี่ยวชาญ คาดการณ์ ฉลาดไม่มีความประมาท รู้จักป้องกันความเสี่ยงที่เกิดขึ้นจากความไม่แน่นอนใน

อนาคตที่อาจเจ็บป่วย ประสบอุบัติเหตุ สูญเสียความสามารถการทำงานหรือมีอันต้องล้มตาย ก่อนเวลาอันควร ก็ย่อมต้องมีการเตรียมตัวไว้ล่วงหน้า ก่อนที่จะมีภัยมาถึงตน คณะบุคคลที่ตก นั้นเองอยู่ในภัยประเภทเดียวกัน ได้คิดเสียสละเงินไว้คนละเล็กละน้อยจนได้เป็นเงินก้อนใหญ่ ก้อนหนึ่ง และเมื่อหากมีภัยอันตรายเกิดขึ้นแก่บุคคลใดบุคคลหนึ่งในคณะบุคคลนั้น ก็จะได้รับ การบรรเทาความช่วยเหลือความเสียหายจากเงินที่ช่วยกันสะสมไว้ โดยลักษณะและวิธีนี้ เรียกว่า การประกัน

1. การประกันชีวิตในประเทศไทย

การประกันชีวิตในประเทศไทยเริ่มมีในสมัยรัชกาลที่ 5 โดยประเทศอังกฤษได้ส่ง คณะทูตมาเจริญสัมพันธไมตรี และถือโอกาสอันเดี่ยวนี้นำร่างบังคับมทูลขอพระบรมราชานุญาต ให้บริษัท The Equitable Assurance Society Of London บริษัทประกันชีวิตแห่งแรกในโลก ซึ่งก่อตั้งในกรุงลอนดอน ประเทศอังกฤษ เมื่อปี พ.ศ.2305 แต่งตั้งตัวแทนประกอบการ ประกันชีวิตขึ้นในประเทศไทย โดยมีสมเด็จพระยาบรมมหาศรีสุริยวงศ์ ได้ทรงเอาประกัน เป็นรายแรก แต่การประกันชีวิตในสมัยนั้นยังไม่เจริญเท่าที่ควร ทั้งนี้ประชาชนส่วนมากยังไม่ เข้าใจในธุรกิจประเภทนี้ ซึ่งบริษัทใหญ่ในลอนดอนเห็นว่า ถ้าดำเนินธุรกิจต่อไปจะไม่คุ้มกับ ค่าใช้จ่าย จึงถอนตัวแทนประกันชีวิตออก แต่ยังคงรักษาผลประโยชน์และบริการผู้ทำประกันชีวิตอยู่ ตามเงื่อนไขเดิม

2. นโยบายการประกันอุบัติเหตุ

2.1 สิทธิพิเศษที่จะได้รับ

2.1.1 ไม่ต้องตรวจสุขภาพ

2.1.2 คุ้มครองอุบัติเหตุ 24 ชั่วโมงทั่วโลก

2.1.3 สามารถใช้สิทธิพิเศษไม่ต้องสำรองจ่ายค่ารักษา (ตามวงเงินคุ้มครอง บนบัตร) หรือ Fax Claim กับโรงพยาบาลกว่า 200 แห่งทั่วประเทศไทย

2.1.4 ในการต่ออายุกรมธรรม์ จะได้รับการเพิ่มความคุ้มครอง การเสียชีวิต การสูญเสียอวัยวะ สายตา หรือทุพพลภาพถาวรสิ้นเชิง เพิ่มขึ้นปีละ 5% ของจำนวนเงินเอา ประกันภัยต่อเนื่องเป็นเวลา 5 ปี

2.2 การกระทำของผู้เอาประกันภัยขณะอยู่ภายใต้ฤทธิ์สุราหรือยาเสพติด การฆ่าตัวตาย พยายามฆ่าตัวตาย หรือการทำร้ายร่างกายตนเอง

- 2.2.1 การแทงลูก
- 2.2.2 สงคราม การปฏิบัติการกบฏ
- 2.2.3 การจลาจล การนัดหยุดงาน การที่ประชาชนก่อความวุ่นวายลุกฮือต่อต้านรัฐบาล
- 2.2.4 การแผ่รังสีหรือกัมมันตภาพรังสีจากเชื้อเพลิงนิวเคลียร์ อาวุธนิวเคลียร์
- 2.2.5 การเล่นหรือแข่งกีฬาอันตราย เช่น การดำน้ำ การเล่นบันจี้จัมพ์ เล่นสกี การแข่งรถ แข่งเรือ แข่งสเก็ต เป็นต้น
- 2.2.6 ขณะขับขีหรือโดยสารรถจักรยานยนต์
- 2.2.7 ขณะโดยสารอยู่ในเครื่องบินที่มีใช้สายการบินพาณิชย์ เช่น เฮลิคอปเตอร์
- 2.2.8 ขณะเข้าร่วมการทะเลาะวิวาท ก่ออาชญากรรม หรือหลบหนีการจับกุม
- 2.2.9 ขณะเข้าปราบปรามหรือปฏิบัติการทางสงคราม หากผู้เอาประกันภัยเป็นทหาร ตำรวจ หรืออาสาสมัคร

ระบบปฏิบัติการแอนดรอยด์

1. ความเป็นมาของระบบปฏิบัติการแอนดรอยด์

แอนดรอยด์ (Android) กูเกิลแอนดรอยด์ (Google Android) หรือระบบปฏิบัติการแอนดรอยด์ (Android Operating System) เป็นชื่อเรียกชุดซอฟต์แวร์หรือแพลตฟอร์ม (Platform) สำหรับอุปกรณ์อิเล็กทรอนิกส์ที่มีหน่วยประมวลผลเป็นส่วนประกอบ เช่น คอมพิวเตอร์ โทรศัพท์ (Telephone) โทรศัพท์เคลื่อนที่ (Cell phone) อุปกรณ์เล่นอินเทอร์เน็ตขนาดพกพา เป็นต้น แอนดรอยด์ถือกำเนิดอย่างเป็นทางการในวันที่ 5 พฤศจิกายน 2550 โดยบริษัทกูเกิล จุดประสงค์ของแอนดรอยด์มีจุดเริ่มต้นมาจากบริษัท Android Inc. ที่ได้นำเอาระบบปฏิบัติการลินุกซ์ (Linux) ซึ่งนำไปใช้งานกับเครื่องเซิร์ฟเวอร์ (Server) เป็นหลัก นำมาลดทอนขนาดตัว (แต่ไม่ลดทอนความสามารถ) เพื่อให้เหมาะสมแก่การนำไปติดตั้งบนอุปกรณ์พกพา ที่มีขนาดพื้นที่จัดเก็บข้อมูลที่จำกัด โดยที่แอนดรอยด์จะใช้สัญลักษณ์หุ่นยนต์เป็นหุ่นยนต์ที่คอยช่วยเหลืออำนวยความสะดวกแก่ผู้ที่พกพาในทุกที่ทุกเวลา

กูเกิลแอนดรอยด์เป็นชื่อเรียกอย่างเป็นทางการของแอนดรอยด์ เนื่องจากปัจจุบันบริษัทกูเกิลเป็นผู้ถือสิทธิบัตรในตราสัญลักษณ์ชื่อและรหัสต้นฉบับ (Source Code) ของแอนดรอยด์ภายใต้เงื่อนไขการพัฒนาแบบ GN/L โดยเปิดให้นักพัฒนา (Developer) สามารถนำรหัสต้นฉบับไปพัฒนาปรับแต่งได้อย่างเปิดเผย (Open Source) ทำให้แอนดรอยด์มีผู้เข้าร่วมพัฒนาเป็นจำนวนมากและพัฒนาไปได้อย่างรวดเร็ว

แอนดรอยด์เปิดตัวอย่างเป็นทางการเมื่อวันที่ 5 พฤษภาคม พุทธศักราช 2550 ปัจจุบันมีผู้ร่วมพัฒนากว่า 52 องค์กร ประกอบด้วยบริษัทซอฟต์แวร์ บริษัทผู้ผลิตอุปกรณ์ บริษัทผู้ผลิตชิ้นส่วนอิเล็กทรอนิกส์ บริษัทผู้ให้บริการเครือข่าย และบริษัทที่เกี่ยวข้องกับการสื่อสาร

ประเภทของชุดซอฟต์แวร์ เนื่องจากแอนดรอยด์นั้นเปิดให้นักพัฒนาเข้าไปชมรหัสต้นฉบับได้ ทำให้มีผู้พัฒนาจากหลายฝ่ายนำเอารหัสต้นฉบับมาปรับแต่งและสร้างแอนดรอยด์ในแบบฉบับของตนเอง จึงแบ่งประเภทของแอนดรอยด์ออกได้เป็น 3 ประเภท ดังนี้

1.1 Android Open Source Project (AOSP) เป็นแอนดรอยด์ประเภทแรกที่กูเกิลเปิดให้สามารถนำต้นฉบับแบบเปิดไปติดตั้งและใช้งานในอุปกรณ์ต่างๆ โดยไม่ต้องเสียค่าใช้จ่ายใดๆ

1.2 Open Handset Mobile (OHM) เป็นแอนดรอยด์ที่ได้รับการพัฒนาร่วมกับกลุ่มบริษัทผู้ผลิตอุปกรณ์พกพาที่เข้าร่วมกับกูเกิลในนาม Open Handset Alliances ซึ่งบริษัทจะพัฒนาแอนดรอยด์ในแบบฉบับของตนเองออกมา โดยรูปร่างหน้าตาการแสดงผลและฟังก์ชันการใช้งานจะมีความเป็นเอกลักษณ์และมีลิขสิทธิ์เป็นของตนเอง พร้อมได้รับสิทธิในการ มีบริการเสริมต่างๆ จากกูเกิลที่เรียกว่า Google Mobile Service (GMS) ซึ่งเป็นบริการเสริมที่ทำให้แอนดรอยด์มีประสิทธิภาพเป็นไปตามจุดประสงค์ของแอนดรอยด์แต่การจะได้มา 38 ซึ่ง GMS ผู้ผลิตจะต้องทำการทดสอบระบบและขออนุญาตกับทางกูเกิลก่อนนำออกสู่ตลาดได้

1.3 Cooking หรือ Customize เป็นแอนดรอยด์ที่นักพัฒนานำเอารหัสต้นฉบับจากแหล่งต่างๆ มาปรับแต่งในแบบฉบับของตนเอง โดยจะต้องทำการปลดล็อคสิทธิ์ (Unlock) การใช้งานอุปกรณ์ จึงจะสามารถติดตั้งได้ โดยแอนดรอยด์ประเภทนี้ถือเป็นประเภทที่มีความสามารถมากที่สุดเท่าที่อุปกรณ์เครื่องนั้นๆ จะรองรับได้เนื่องจากได้รับการปรับแต่งให้เข้ากับอุปกรณ์นั้นๆ จากผู้ใช้งานจริง สิทธิในการใช้งานระบบปฏิบัติการทั่วไปที่มีการใช้งานและการเข้าถึงส่วนต่างๆ ภายในระบบเพื่อความปลอดภัยของระบบและผู้ใช้งาน อุปกรณ์ที่ติดตั้งบน

ระบบแอนดรอยด์จึงมีการจำกัดสิทธิ์ไว้ (เว้นแต่ได้ทำการปลดล็อคสิทธิ์) สามารถแบ่งสิทธิ์ของผู้ใช้ในการเข้าถึงระบบได้ดังนี้

1.3.1 สิทธิ์ root สิทธิ์การใช้งานระดับราก ซึ่งถือว่าเป็นรากฐานของระบบ จึงมีความสามารถในการเข้าถึงทุกๆ ส่วนของระบบ

1.3.2 สิทธิ์ ADB (Android Develop Bridge) นักพัฒนาสามารถเข้าถึงส่วนต่างๆ ของระบบได้ผ่านสิทธิ์นี้

1.3.3 Application & System สิทธิ์ของโปรแกรมในการเข้าถึงระบบและสิทธิ์ของระบบในการเข้าถึงอุปกรณ์โดยสิทธิ์เหล่านี้ ตัวระบบจะเป็นตัวจัดมอบและถอนสิทธิ์ตามเงื่อนไขที่กำหนดซึ่งจะถูกแบ่งย่อยออกเป็นหลายหัวข้อ

1.3.4 End-user ผู้ใช้งานขั้นสุดท้าย ซึ่งก็คือผู้ใช้งานทั่วไป การเข้าถึงส่วนต่างๆ ของระบบผ่านช่องทางสิทธิ์ที่โปรแกรมได้รับอีกที โดยจะถูกจำกัดไม่ให้เข้าถึงในส่วนที่เป็นอันตรายต่อแก่นระบบและอุปกรณ์

2. ประวัติแอนดรอยด์

ประวัติแอนดรอยด์ไม่ได้ถูกเริ่มต้นพัฒนาที่ Google แต่เริ่มต้นกับบริษัท Android Inc. โดยผู้ที่เริ่มต้นพัฒนาและก่อตั้งแอนดรอยด์นั้น คือ Andy Rubin (co-founder of Danger Inc.) และมีผู้ร่วมก่อตั้งอีก 4 คน คือ Rich Miner, Nick, and Chris White ซึ่งการที่ Android Inc. ได้ขายลิขสิทธิ์แอนดรอยด์ให้กับ Google นั้น เป็นเพราะว่า Andy Rubin เอง น่าจะเล็งเห็นถึงความสามารถของทรัพยากรบุคคล และ เทคโนโลยีต่างๆ ที่มีใน Google ซึ่งในขณะนั้นเอง Android ถูกเข้าใจว่าเป็นเพียงระบบปฏิบัติการบนอุปกรณ์เคลื่อนที่เท่านั้นซึ่งเหตุการณ์เริ่มต้นทั้งหมดนี้เกิดขึ้นในปี ค.ศ. 2005

ต่อมาในปลายปี ค.ศ. 2007 ก็เกิดปรากฏการณ์ของแอนดรอยด์แพลตฟอร์มอีก 39 ครั้งซึ่งในครั้งนี้ได้มีการเปิดเผยถึงระบบปฏิบัติการบนโทรศัพท์เคลื่อนที่ที่เป็นในลักษณะ Open Platform ซึ่งผู้ใช้สามารถพัฒนาซอฟต์แวร์มาใช้งานเองได้ออกมาจาก กลุ่มบริษัทพัฒนาเทคโนโลยีเกี่ยวกับอุปกรณ์เคลื่อนที่ 34 บริษัทที่มีการเรียกตัวเองว่า Open Handset Alliance และจากการประกาศออกมาในครั้งนี้นำให้ทั่วโลกเริ่มจับตามองกับการเคลื่อนไหวของแอนดรอยด์ มากขึ้น และในปีนี้อังยังมีการประกาศเป็นครั้งแรกของ Android SDK อีกด้วย

3. ข้อเด่นของแอนดรอยด์

เนื่องจากระบบปฏิบัติการแอนดรอยด์มีการเจริญเติบโตอย่างรวดเร็วและมีส่วนแบ่งตลาดของอุปกรณ์ด้านนี้ ขึ้นทุกขณะ ทำให้กลุ่มผู้ใช้งาน และกลุ่มนักพัฒนาโปรแกรม ให้ความสำคัญกับระบบปฏิบัติการแอนดรอยด์เพิ่มมากขึ้นเมื่อมองในด้านของกลุ่มผลิตภัณฑ์ บริษัทที่มีการพัฒนาผลิตภัณฑ์รุ่นใหม่ ได้มีการนำเอาระบบปฏิบัติการแอนดรอยด์ไปใช้ในพร้อม ทั้งยังมีการปรับแต่งให้ระบบปฏิบัติการมีความสามารถ การจัดวาง โปรแกรม และลูกเล่นใหม่ ๆ ที่แตกต่างจากคู่แข่งในท้องตลาด โดยเฉพาะอย่างยิ่ง กลุ่มสินค้าที่เป็น มือถือรุ่นใหม่ และ อุปกรณ์จอสัมผัส โดยมีคุณลักษณะแตกต่างกันไป เช่นขนาดหน้าจอ ระบบโทรศัพท์ ความเร็วของหน่วยประมวลผล ปริมาณหน่วยความจำ แม้กระทั่งอุปกรณ์ตรวจจับต่างๆ (Sensor) หากมองในด้านของการพัฒนาโปรแกรม ทางบริษัทกูเกิ้ล ได้มีการพัฒนา ApplicationFramework ไว้สำหรับนักพัฒนาใช้งาน ได้อย่างสะดวก และไม่เกิดปัญหาเมื่อนาซุดโปรแกรมที่พัฒนาขึ้นมา ไปใช้กับอุปกรณ์ที่มีคุณลักษณะต่างกัน เช่น ขนาดจออุปกรณ์ ไม่เท่ากัน ก็ยังสามารถใช้งาน โปรแกรมได้เหมือนกัน

3.1 ข้อดีของแอนดรอยด์

- 3.1.1 มีลักษณะเป็นโอเพนซอร์สซึ่งทำให้แอนดรอยด์แพลตฟอร์ม ได้รับความนิยอย่างรวดเร็ว
- 3.1.2 มีการเชื่อมต่อที่กลมกลืนกันระหว่าง เว็บแอปพลิเคชัน และโทรศัพท์
- 3.1.3 มีเครื่องมือที่ใช้พัฒนาซอฟต์แวร์ที่ง่ายและมีประสิทธิภาพ

3.2 ข้อจำกัดของแอนดรอยด์

- 3.2.1 อุปกรณ์ที่พัฒนาขึ้นมาอาจจะทำงานไม่รองรับกับระบบแอนดรอยด์ ได้ครบทุกฟังก์ชัน เพราะ Google ไม่ได้มีส่วนร่วมในการควบคุมการผลิต Hardware โดยตรง
- 3.2.2 ในบางครั้งแอนดรอยด์แพลตฟอร์มมีการทำงานที่ยังไม่ดีนัก

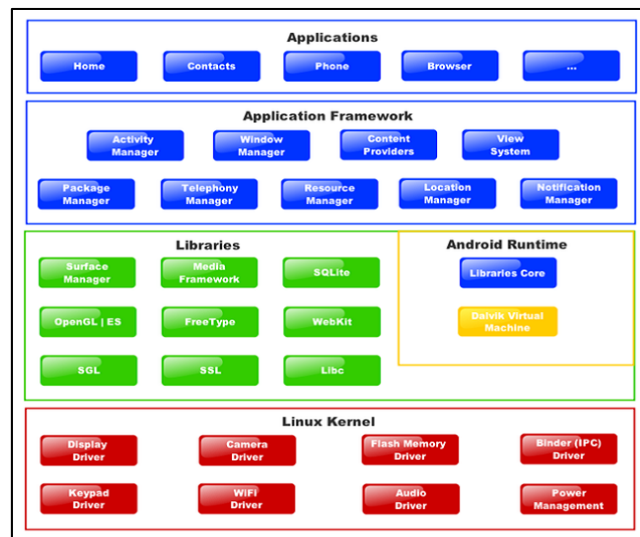
3.3 แผนกลยุทธ์ของแอนดรอยด์

- 3.3.1 กรกฎาคม ค.ศ. 2005 -Google เริ่มถือลิขสิทธิ์ใน Android. Inc.
- 3.3.2 12 พฤศจิกายน ค.ศ. 2007 -SDK ตัวแรกของแอนดรอยด์ถูกเปิดเผย
- 3.3.3 มกราคม ถึง สิงหาคม ค.ศ. 2008 -มีการจัดการแข่งขัน Android Developer Challenge

- 3.3.4 18 สิงหาคม ค.ศ. 2008 -มีการพัฒนาตัว SDK 0.9 beta
- 3.3.5 กันยายน ค.ศ. 2008 -Android 1.0 (pre) SDK ถูกผลิตออกมา
- 3.3.6 ไตรมาสที่ 4 ค.ศ. 2008 -อุปกรณ์ Android 1.0 เริ่มมีการขาย
- 3.3.7 ไตรมาสที่ 4 ค.ศ. 2008 -มีการเผยแพร่ Source Code

4. สถาปัตยกรรมแอนดรอยด์

การที่จะเริ่มต้นพัฒนาซอฟต์แวร์บนแอนดรอยด์แพลตฟอร์มจำเป็นต้องทำความรู้จักและเข้าใจกับโครงสร้างพื้นฐานต่างๆ เพื่อที่จะได้นำมาประยุกต์ใช้กับซอฟต์แวร์ โดยสิ่งสำคัญสิ่งหนึ่งที่ทำให้เข้าใจรูปแบบของระบบการทำงานของแอนดรอยด์แพลตฟอร์ม ก็คือ สถาปัตยกรรมของแอนดรอยด์ แอนดรอยด์เป็นซอฟต์แวร์ที่มีโครงสร้างแบบเรียงทับซ้อนหรือแบบสแต็ก ซึ่งรวมเอาระบบปฏิบัติการ, มิดเดิลแวร์ และแอปพลิเคชันที่สำคัญเข้าไว้ด้วยกัน เพื่อใช้สำหรับทำงานบนอุปกรณ์พกพาเคลื่อนที่ เช่น โทรศัพท์มือถือ เป็นต้น การทำงานของแอนดรอยด์มีพื้นฐานอยู่บนระบบลินุกซ์ เคอร์เนล ซึ่งใช้ Android SDK เป็นเครื่องมือสำหรับการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ Android และใช้ภาษา Java ในการพัฒนาสถาปัตยกรรมของแอนดรอยด์ นั้นถูกแบ่งออกเป็นลำดับชั้นออกเป็น 4 ชั้นหลักดังในตารางด้านล่าง



ภาพที่ 1 สถาปัตยกรรมแอนดรอยด์ (Android Architecture)

ที่มา: คมสันต์ ผุยลานวงศ์. (2555 : 8)

4.1 ลินุกซ์ เคอร์เนล

ส่วน Kernel ซึ่งเป็นระดับล่างสุดของแอนดรอยด์นั้นจะถูกสร้างขึ้นมาจาก Standard Linux 2.6.24 Kernel และได้ส่งผลให้นักพัฒนาสามารถ Patch Kernel ของลินุกซ์ได้เช่นเดียวกับการ Patch บนระบบปฏิบัติการลินุกซ์โดยทั่วไป แต่ถึงแม้ว่าแอนดรอยด์จะนำ Kernel ของ Linux มาใช้ในการพัฒนาระบบ ตัวแอนดรอยด์แพลตฟอร์มนั้นก็ยังไม่ถือว่าเป็นระบบปฏิบัติการที่เป็นลินุกซ์โดยสมบูรณ์ เนื่องจากบนแอนดรอยด์แพลตฟอร์มนั้น จะไม่รองรับการใช้งาน glibc เหมือนใน Linux ซึ่งส่งผลให้แอนดรอยด์ไม่มีฟังก์ชันในการทำงานต่างๆ สมบูรณ์เทียบเคียงได้กับลินุกซ์ จึงทำให้สรุปเหตุผลที่แอนดรอยด์เลือกใช้ Linux Kernel ได้ดังนี้

4.1.1 ตัวระบบลินุกซ์เป็นโอเพนซอร์ส ซึ่งเป็นส่วนสำคัญหนึ่งที่ทำให้ระบบแอนดรอยด์แพลตฟอร์มนั้น อยู่ภายใต้เงื่อนไขการใช้งานซอฟต์แวร์โอเพนซอร์ส และทำให้แอนดรอยด์แพลตฟอร์มมีการเผยแพร่ได้รวดเร็วอีกด้วย

4.1.2 รองรับฟังก์ชันการทำงานในส่วนของ Driver ได้อย่างดี ซึ่งทำให้แอนดรอยด์แพลตฟอร์มรองรับกับการใช้งานของชิ้นส่วนต่างๆที่มีบนอุปกรณ์สื่อสารเคลื่อนที่

4.1.3 มีการใช้งานระบบ Permission-Based Security Model เพื่อที่แอนดรอยด์แพลตฟอร์มจะมีการกำหนดสิทธิ์ในการใช้งานทรัพยากรของระบบได้ ซึ่งจะทำให้ระบบแอนดรอยด์แพลตฟอร์มมีความปลอดภัยสูงอีกด้วย โดยนอกจากนี้เราสามารถจะดาวน์โหลดตัว Android Kernel มาทำการศึกษาได้จาก <http://git.android.com>

4.1.4 Media Framework จะช่วยรองรับเกี่ยวกับการทำงานกับไฟล์ในฟอร์แมตต่างๆไม่ว่าจะเป็นในด้าน Video, Audio หรือว่าจะเป็น Frame-Format ดังรูปที่มีการนำภาพถ่ายที่มีชนิดไฟล์เป็น JPG



ภาพที่ 2 การแสดงผลไฟล์รูปภาพ

ที่มา: คมสันต์ พุยลานวงค์. (2555 : 9)

4.1.5 SQLite เป็นไลบรารี ที่เก็บฟังก์ชันการใช้ระบบฐานข้อมูลบนแอนดรอยด์แพลตฟอร์ม ซึ่ง SQLite นี้เป็นระบบฐานที่ได้รับความนิยมในอุปกรณ์ที่มีหน่วยความจำน้อย, ใช้งานง่าย และเหมาะสมกับการใช้งานกับฐานข้อมูลแบบ Standalone ด้วยการทำงานของไลบรารี SQLite บนแอนดรอยด์แพลตฟอร์มนั้นจะช่วยให้ นักพัฒนาสามารถสร้างฐานข้อมูลส่วนตัวเอาไว้ใช้กับโปรแกรมได้โดยง่ายไม่ว่าจะผ่านทางภาษา Sql หรือด้วยเมธอดที่มีให้ในไลบรารี

4.1.6 WebKit เป็น Open Source Browser ที่ช่วยให้แอนดรอยด์แพลตฟอร์ม รองรับการใช้งาน JavaScript, AJAX และ CSS รวมทั้งสามารถเปิดหน้าเว็บไซต์ได้อย่างมีประสิทธิภาพ เช่นเดียวกันกับ Safari ที่ทำงานอยู่บน OS X นอกจากนั้นแล้วการที่แอนดรอยด์แพลตฟอร์มใช้ Webkit ยังส่งผลให้การทำงานของซอฟต์แวร์ที่ Google ได้พัฒนาขึ้นมาอย่าง Chrome และ Gear ร่วมกันได้ดีอีกด้วย ซึ่งการทำงานหลักของไลบรารีนี้จะประกอบไปด้วยการแสดงผลของหน้าเว็บ และการจัดการกับ Cookie

4.2 แอนดรอยด์รันไทม์

ในส่วนของแอนดรอยด์รันไทม์ นั้นจะมีการทำงานควบคู่ไปกับส่วนของ Libraries โดยจะมีหน้าที่หลักคือการ Interpret โค้ดจาวาถูก Compile แล้วมาทำงาน ซึ่งในส่วนนี้จะมีระบบการทำงานย่อยที่แบ่งได้อีกสองส่วนคือ

4.2.1 Dalvik Virtual Machine จะสามารถเปรียบได้กับ Java Virtual Machine ในจาวา แต่ว่าในส่วนของ Dalvik Virtual Machine นั้นได้ถูกพัฒนาขึ้นมาโดย Google ซึ่งจะแตกต่างจากจาวาทั่วไปตรงที่จะทำการรันไฟล์ .dex ไฟล์ ที่ถูกแปลงมาจาก .class เพื่อให้เหมาะสมกับอุปกรณ์ที่ใช้งานแอนดรอยด์แพลตฟอร์ม ที่มี หน่วยความจำน้อย

4.2.2 Core Libraries เป็นองค์ประกอบที่ทำให้แอนดรอยด์สามารถใช้งานฟังก์ชันพื้นฐานของจาวา ซึ่งมีพื้นฐานมาจาก Java SE 5 แต่ฟังก์ชันพื้นฐานจะถูกลดให้เหลือเพียงเท่าที่จำเป็น อย่างเช่น Input/output (java.io), Security (java. Security) และ Utility (java.util) เป็นต้น

4.3 แอปพลิเคชันเฟรมเวิร์ค

ในส่วนนี้เป็นระดับที่รวบรวม Service ทั้งหมดที่แอนดรอยด์แพลตฟอร์มมีให้พร้อมสำหรับซอฟต์แวร์เรียกใช้ ซึ่งในระดับนี้เองที่ทำให้ นักพัฒนาซอฟต์แวร์สามารถเรียกใช้ APIs และเข้าถึงข้อมูลต่างๆได้ง่าย ไม่ว่าจะเป็น Activity Manager ที่เป็นตัวกำหนดการทำงาน

ในแต่ละ Activity, Content Providers ที่ทำให้ซอฟต์แวร์แต่ละตัวสามารถส่งผ่านข้อมูลกันได้ และ View System ที่รวบรวม UI ชนิดต่างๆเอาไว้เพื่อให้นักพัฒนาเรียกใช้ เป็นต้น ซึ่งในส่วน ของ Application Framework ยังสามารถแบ่งได้อีก 2 ส่วนคือ

4.3.1 Core Platform Service หมายถึง Service ที่ใช้ในการจัดการ เกี่ยวกับการรันโปรแกรมและควบคุมการทำงานของโปรแกรม ซึ่งจะมีส่วนประกอบเช่น Activity Manager จะเป็นส่วนที่ใช้ควบคุมการทำงานของแต่ละ Activity ซึ่งจะอธิบายใน หัวข้อต่อไป Content Providers จะเป็น Service หนึ่งที่ทำให้ซอฟต์แวร์บนแอนดรอยด์ แพลตฟอร์มนั้น สามารถส่งผ่านข้อมูลถึงกันได้ เช่น การไปดึงข้อมูลในส่วนของ Contact List มาเพื่อใช้งานในซอฟต์แวร์อื่นๆ เป็นต้น View System จะเป็นส่วนที่รวบรวม User Interface ต่างๆไว้ให้เรียกใช้ไม่ว่าจะเป็น Text View และ Button เป็นต้น นอกจากนี้ ข้างต้น แล้ว ในส่วนนี้ยังประกอบไปด้วย Package Manager, Window Manager และ Resource Manager

4.3.2 Hardware Service หมายถึง Service ที่เกี่ยวข้องกับการใช้งาน Hardware ต่างๆ โดยในประเภทนี้จะมีส่วนประกอบเด่นๆดังนี้ Telephony Service ที่จะใช้ ควบคุมฟังก์ชันที่ใช้โทรศัพท์ Bluetooth Service ที่ใช้ควบคุมการใช้งาน Bluetooth นอกจากนี้ แล้วยังมี Location Service, Wi-Fi Service, USB service และ Sensor Service อีก เป็นต้น

4.4 แอปพลิเคชันเลเยอร์

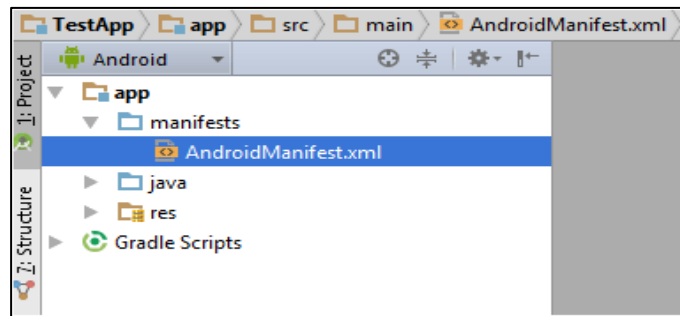
คือระดับที่หมายถึง ซอฟต์แวร์ หรือ โปรแกรมที่ใช้อยู่โดยทั่วไป อย่างเช่น โปรแกรมรับส่งอีเมล, SMS, แผนที่, Browser เครื่องมือจัดการสมุดโทรศัพท์และโปรแกรมหลัก อื่นๆ ซึ่งในชั้นนี้ยังรวมถึงซอฟต์แวร์ที่เราได้ทำการพัฒนาขึ้นมาด้วย

5. แอนดรอยด์ Building Blocks

จากการที่ซอฟต์แวร์ที่ทำงานบนแอนดรอยด์แพลตฟอร์มนั้น เกิดจากการทำงาน ที่รวมกันขึ้นมาจาก Service ต่างๆเข้าด้วยกันในระดับ Application Platform ซึ่งผู้พัฒนา ซอฟต์แวร์จะต้องคำนึงถึงการทำงานของซอฟต์แวร์เพื่อที่จะเรียกใช้งาน Service ต่างๆ เหล่า นั้นจาก Android APIs ให้เหมาะสมกับซอฟต์แวร์ โดยรายละเอียดต่อไปนี้จะ เป็นลักษณะต่างๆ ของแต่ละ Android APIs เพื่อการนำไปใช้ในการทำงานที่แตกต่างกัน

5.1 AndroidManifest.xml

จากการที่ซอฟต์แวร์ที่ทำงานบนแอนดรอยด์แพลตฟอร์มนั้น เกิดจากการทำงานที่รวมกันขึ้นมาจาก Service ต่างๆ เข้าด้วยกันในระดับ Application Platform ซึ่งผู้พัฒนาซอฟต์แวร์จะต้องคำนึงถึงการทำงานของซอฟต์แวร์ เพื่อที่จะเรียกใช้งาน Service ต่างๆ เหล่านั้นจาก Android APIs ให้เหมาะสมกับซอฟต์แวร์โดยรายละเอียดต่อไปนี้ จะเป็นลักษณะต่างๆของแต่ละ Android APIs เพื่อการนำไปใช้ในการทำงานที่แตกต่างกัน ซึ่งการที่ได้นำองค์ประกอบที่เรียกใช้ Android APIs ต่างๆ นั้นก็คือ Building Block 2.4.1 AndroidManifest.xml ตัวไฟล์ AndroidManifest.xml เป็นไฟล์ที่จะใช้ควบคุมการทำงานของซอฟต์แวร์ที่ได้พัฒนาขึ้นให้ทำงานไปตามลำดับ เป็นไฟล์ที่ใช้สำหรับประกาศเพื่อเรียกใช้งาน ส่วนประกอบอื่นๆ หรือ Service ต่างๆ ให้ทำงานร่วมกัน และยังเป็นไฟล์ที่เปรียบเสมือนแหล่งอ้างอิงของซอฟต์แวร์ที่พัฒนาขึ้นมาอีกด้วยซึ่งไฟล์ AndroidManifest.xml จะมีรายละเอียดที่สำคัญในส่วนต่างๆ ดังนี้



ภาพที่ 3 แสดงตำแหน่งของ AndroidManifest.xml บน Project

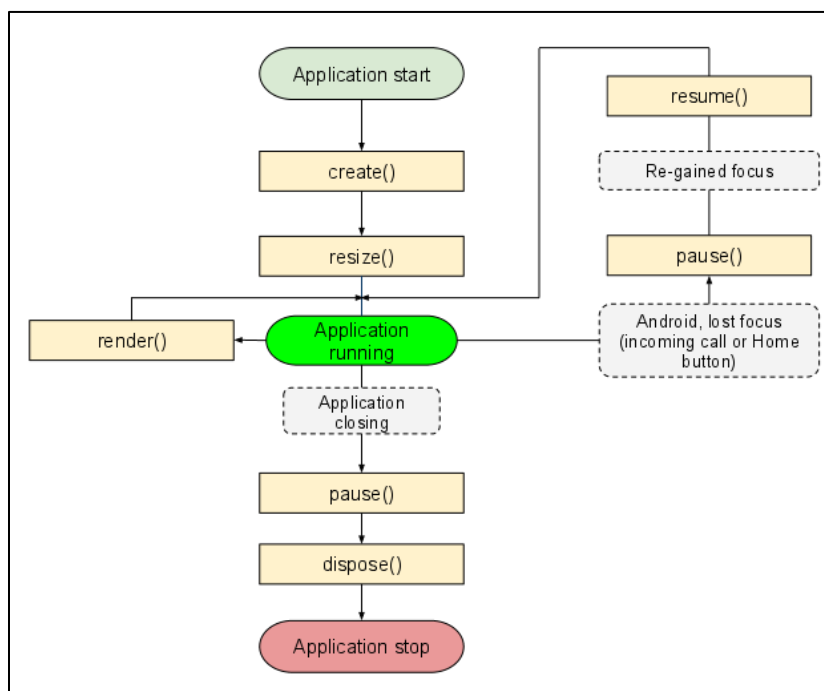
ที่มา: พรพรรณ มณีจันทร์. (2558 : 14)

สิ่งสำคัญอีกสิ่งหนึ่งของ Android Manifest คือการประกาศ intent-filters โดยจะใช้เป็นตัวเรียกให้ Activity อื่นที่อยู่ในซอฟต์แวร์ทำงาน การทำงานของ intent-filters จะเริ่มต้นขึ้นเมื่อมีการเรียกใช้ intent หลังจากนั้นจะทำการเปรียบเทียบกับ intent-filters เพื่อดำเนินการต่อไป นอกจากนี้แล้วในการทำงานด้านอื่นๆ ของ Building Block ด้วย

5.2 Activity

โดยทั่วไปความหมายของ Activity ในแอนดรอยด์แพลตฟอร์มนั้นจะหมายถึง วัตถุหนึ่งๆ ที่มีการทำงานหลายๆอย่าง และเป็นการทำงานที่มีการแสดงผลรวมอยู่ด้วย ซึ่งในแอปพลิเคชันหนึ่งๆนั้นอาจจะประกอบไปด้วยหลายๆ Activity ได้ และในหนึ่ง

แอปพลิเคชันนั้นจำเป็นต้องมีอย่างน้อย 1 Activity ซึ่งในการทำงานกับ Activity นั้น จะมีลักษณะการทำงานเป็นวงจร จึงทำให้การทำงานของซอฟต์แวร์บนระบบแอนดรอยด์แพลตฟอร์มมีระเบียบแบบแผน Activity Life cycle เมื่อ Activity เริ่มมีการทำงานจะถูกเก็บไว้ในส่วนของ System ที่เรียกว่า Activity stack ซึ่งเมื่อ Activity เริ่มทำงานก็就会被จัดเก็บไว้ในส่วนบนสุดของ Stack นั้น แต่ในส่วนของ Activity ที่ถูกเรียกใช้งานก่อนหน้าก็จะถูกจัดเก็บเอาไว้ใน Stack ชั้นต่อไป จากการทำงานของวงจรเบื้องต้นดังกล่าวนี้ จะสามารถศึกษาได้จากวงจรที่เรียกว่า Activity Life cycle ดังแผนภาพข้างล่าง



แผนภาพที่ 1 แสดง Activity Life cycle ของ Android

ที่มา: คมสันต์ ฝูยลานวงศ์. (2555 : 13)

วงจรการพัฒนาระบบ (System Development Life Cycle: SDLC)

วงจรการพัฒนาระบบ (System Development Life Cycle : SDLC) คือ กระบวนการทางความคิด (Logical Process) ในการพัฒนาระบบสารสนเทศเพื่อแก้ปัญหา และตอบสนองความต้องการของผู้ใช้ได้ โดยระบบที่จะพัฒนาอาจเริ่มด้วยการพัฒนาระบบใหม่ หรือนำระบบเดิมที่มีอยู่แล้วมาปรับเปลี่ยนให้ดียิ่งขึ้น ภายในวงจรมีจะแบ่งกระบวนการพัฒนาออกเป็นระยะ ได้แก่ ขั้นตอนการสำรวจระบบ (System investigation) ขั้นตอนการวิเคราะห์

ระบบ (System Analysis) ขั้นการออกแบบและพัฒนาระบบสารสนเทศ (System design) ระยะการสร้างและพัฒนา (Implementation Phase) การทดลองใช้และติดตั้งระบบ (System implementation) และการบำรุงรักษาระบบและการประเมินผล (System maintenance and review) โดยแต่ละระยะจะประกอบไปด้วยขั้นตอนต่างๆ แตกต่างกันไป ตาม Methodology ดังนั้น การศึกษาครั้งนี้ได้ประยุกต์ใช้วงจรการพัฒนาาระบบสารสนเทศ (System development life cycle : SDLC) (Stair 1996 : 411-412) ซึ่งประกอบด้วย กิจกรรมต่างๆ 5 ขั้น ดังนี้

1. ขั้นตอนการวางแผนระบบ (Systems Planning)

การวางแผนระบบ (Systems Planning) เป็นสิ่งสำคัญที่จะทำให้การพัฒนา ระบบสารสนเทศเป็นไปด้วยดีตามแผนเพราะหากไม่มีการวางแผนที่ดีอาจเกิดความล่าช้าของ การพัฒนาระบบสารสนเทศได้และเป็นขั้นตอนแรกสำหรับเตรียมความพร้อมในการพัฒนา ระบบสารสนเทศด้วยการตั้งประเด็นคำถามที่ว่ามีความต้องการอะไรบ้างในระบบสารสนเทศที่ จะพัฒนาขึ้น

1.1 กำหนดโอกาสของระบบสารสนเทศในการใช้งาน (Identify Opportunity) การศึกษาระบบงานปัจจุบันเป็นการศึกษาปัญหาที่เกิดขึ้นจากการใช้งานระบบที่มีอยู่จุดบันทึก ถึงความต้องการสารสนเทศที่จะต้องปรับปรุงเพิ่มเติมอาจสำรวจจากการสัมภาษณ์แบบ สอภ

1.2 วิเคราะห์ความเป็นไปได้ (Analyze Feasibility) กำหนดขอบเขตของการ พัฒนาระบบการกำหนดรายละเอียดและขั้นตอนของการดำเนินงานและระยะเวลาที่ใช้และ หากผลการสำรวจพบว่าระบบงานนั้นมีความเป็นไปได้ที่จะพัฒนานักวิเคราะห์ระบบจะ วิเคราะห์ ต้นทุน/ผลตอบแทน (Cost-Benefit Analysis) ในการศึกษาเพื่อหาแนวทางที่ เหมาะสมของระบบสารสนเทศที่จะพัฒนาขึ้น

1.3 พัฒนาแผนการทำงาน (Develop Work plan) การจัดทำข้อเสนอโครงการ สำหรับพัฒนาระบบสารสนเทศซึ่งเป็นการเก็บรวบรวมข้อมูลตามความต้องการของผู้บริหารทุก ระดับและบุคลากรระดับปฏิบัติการจากนั้นนำมาวิเคราะห์เพื่อวางแผนถึงทางเลือกต่างๆ ของ ระบบสารสนเทศที่จะพัฒนาขึ้นพร้อมทั้งสรุปผลและนำเสนอแนวทางที่เหมาะสมที่สุดเพื่อนำไป วิเคราะห์และออกแบบระบบต่อไป

2. ขั้นตอนการวิเคราะห์ระบบ (Systems Analysis)

การวิเคราะห์ระบบ (Systems Analysis) คือ การศึกษาและทำความเข้าใจถึงระบบงานเดิมที่ใช้อยู่ซึ่งอาจเป็นระบบการทำงานด้วยมือหรือเป็นระบบสารสนเทศเดิมที่ใช้อยู่ก็ได้ การวิเคราะห์ระบบงานเดิมจะทำให้นักวิเคราะห์ระบบทราบถึงสภาพปัญหาที่เกิดขึ้นและใช้เป็นข้อมูลในการวิเคราะห์และออกแบบระบบงานใหม่ต่อไปนอกจากนี้ งานของนักวิเคราะห์ระบบคือการพิจารณาถึงความต้องการของผู้ใช้ระบบซึ่งจะทำให้ผู้ใช้ทราบว่าต้องการอะไรบ้าง ทำให้นักวิเคราะห์ระบบสามารถกำหนดองค์ประกอบต่างๆ ที่ใช้ในการทำงานของระบบได้เช่น ข้อมูลและสิ่งที่จะต้องนำสู่ระบบลักษณะของแฟ้มข้อมูลลักษณะการประมวลผลและผลลัพธ์ ที่ระบบสร้างให้แก่ผู้ใช้

3. ขั้นตอนการออกแบบระบบ (Systems Design)

การออกแบบระบบ (Systems Design) เป็นการนำข้อมูลจากการวิเคราะห์ระบบที่เป็นแนวคิด (Concept) มาออกแบบให้เห็นรูปร่างของระบบสารสนเทศโดยนักวิเคราะห์ระบบจะออกแบบระบบทีละส่วนโดยเริ่มจากส่วนที่เป็นผลลัพธ์ (Output) ก่อน เพราะผลลัพธ์นั้นเกิดจากการนำข้อมูลเข้าระบบแล้วไปประมวลผลดังนั้นการออกแบบผลลัพธ์หรือส่วนแสดงผลจะทำให้ทราบถึงการออกแบบในส่วนอื่นๆ ที่เกี่ยวข้อง

4. ขั้นตอนการพัฒนา ระบบ (Systems Development)

การพัฒนา ระบบ (Systems Development) หลังจากทีนักวิเคราะห์ระบบได้ออกแบบระบบใหม่และจัดการสั่งซื้ออุปกรณ์ต่างๆ เสร็จเรียบร้อยแล้วในขั้นตอนนี้คือการนำระบบที่ได้ออกแบบมาแล้วมาพิจารณาเพื่อสร้าง Program Software ที่จะใช้งานโดยนักเขียนโปรแกรมจะเขียนโปรแกรมตามที่นักวิเคราะห์ระบบได้ออกแบบไว้เมื่อสร้างระบบเสร็จเรียบร้อยแล้วจะต้องนำ Software ที่สร้างไว้แล้วมาทดสอบ

5. ขั้นตอนการติดตั้งและดำเนินการใช้ระบบ (Systems Implementation & Operation)

การติดตั้งและดำเนินการใช้ระบบ (Systems Implementation & Operation) เมื่อดำเนินการสร้างระบบและทำการตรวจสอบแก้ไขเสร็จเรียบร้อยแล้วระบบงานจะถูกส่งมอบและทำการติดตั้งระบบ (Installed System) ลงคอมพิวเตอร์เพื่อนำไปใช้ในการปฏิบัติงาน

ควรมีการประเมินและสร้างการยอมรับระบบงานใหม่ให้กับบุคลากรที่ใช้ระบบสารสนเทศซึ่ง
การดำเนินการใช้ระบบ

จากการศึกษาวิธีการเชิงระบบ สรุปได้ว่าการวิจัยในครั้งนี้ ผู้วิจัยได้ดำเนินการ
วิจัยตามขั้นตอนวิธีการเชิงระบบแบบ SDLC มี 5 ขั้นตอน ได้แก่ การวางแผนระบบ การ
วิเคราะห์ระบบ การออกแบบระบบ การพัฒนาระบบ การติดตั้งและดำเนินการใช้ระบบ โดย
นำมาใช้ในกระบวนการวิจัยและกระบวนการพัฒนาเครื่องมือในการวิจัย

การวิเคราะห์และออกแบบระบบด้วย UML : (Unified Modeling Language)

การวิเคราะห์และออกแบบระบบเชิงวัตถุเป็นวิธีที่นิยมกันมากในปัจจุบัน และมี
แนวโน้มที่จะทดแทนการออกแบบระบบแบบเดิม กระบวนการพัฒนาระบบตามแบบวิธี
Rational Unified Process หรือ Rational Objectory Process เป็นกระบวนการที่
ครอบคลุมกระบวนการพัฒนาระบบทั้งหมด โดยการพิจารณาทั้งงานด้านการบริหารและงาน
ด้านเทคนิค กระบวนการพัฒนาจะมีลักษณะการทำซ้ำ (Iterative) และการเพิ่มขึ้น
(Incremental) ดังนั้นงานที่ทำจะไม่มีมากในคราวเดียวกันในตอนสุดท้ายของโครงการ แต่จะมี
การแบ่งงานออกเป็นช่วงๆ (Phase) ในช่วงของการสร้างระบบ (Construction Phase) การ
ทดสอบและการรวบรวมส่วนย่อยเข้ากับระบบรวม จะมีการทำซ้ำหลายๆ ครั้ง เพื่อให้ได้
โปรแกรมที่มีคุณภาพ และตรงตามความต้องการในการทำซ้ำแต่ละรอบจะประกอบด้วย การ
วิเคราะห์ (Analysis) การออกแบบ (Design) การเขียนโปรแกรมที่ใช้ในการสร้างและการ
พัฒนาซอฟต์แวร์ (Implement) และการทดสอบระบบ (Testing) โดยสามารถแสดงได้ ดังนี้
(ชาลี และเทพฤทธิ์, 2544 : 38 - 80)

1. ช่วงของการพัฒนาระบบ

1.1 อินเซพชันเฟส (Inception Phase) เป็นการเก็บข้อมูลพื้นฐานเกี่ยวกับ
ระบบที่ต้องการ โดยจะมีความเกี่ยวข้องกับฟังก์ชันการทำงานต่างๆ ความสามารถ
ประสิทธิภาพเทคโนโลยีที่ใช้และคุณสมบัติอื่นๆ อีกทั้งยังเป็นการกำหนดแนวคิดเพิ่มเติมและ
แสดงวิธีที่ใช้ในการพัฒนาในขั้นตอนต่อไป และแสดงวิธีการที่ทำให้ระบบมีความสามารถมาก
ขึ้นโดยผลลัพธ์ที่ได้จากกระบวนการนี้จะปรากฏอยู่ในรูปของงานโดยรวม ซึ่งแสดงว่าจะต้อง
สร้างอะไรขึ้นมาบ้างกำหนดว่าจะสร้างได้อย่างไร และมีการทำงานอย่างไร กระบวนการนี้
จำเป็นต้องมีทักษะในการวิเคราะห์ระบบให้ออกมาอยู่ในรูปของฟังก์ชันหลักของระบบ และผู้
ติดต่อกับระบบ (Actor) ซึ่งอธิบายอยู่ในรูปของมุมมองการใช้งาน (Use Case View) และยัง

ต้องมีการวางแผนด้านงบประมาณค่าใช้จ่ายในการพัฒนาระบบ ความสามารถทางการตลาด การวิเคราะห์ความเสี่ยง และผลิตภัณฑ์ของคู่แข่งในกรณีการพัฒนาระบบเพื่อธุรกิจ

1.2 อีลาโบเรชันเฟส (Elaboration Phase) จะประกอบไปด้วยรายละเอียดของการวิเคราะห์ระบบ การกำหนด และวางแผนก่อนการทำงานขั้นตอนต่างๆ ได้แก่

1.2.1 แผนผังที่แสดงภาพในเชิงสถิตยของระบบ (Static Diagram) โดยจะแสดงถึงการมีอยู่ของคลาส และความสัมพันธ์ระหว่างคลาส แต่จะไม่แสดงถึงกิจกรรมที่จะเกิดขึ้นซึ่งมี 2 แผนผังที่ใช้งาน คือ แผนผังการใช้งานของระบบ (Use Case Diagram) และ แผนผังอธิบายความสัมพันธ์ของเอนทิตี (Entity) ต่างๆ ของระบบ (Class Diagram)

1.2.2 แผนผังที่แสดงภาพในเชิงกิจกรรมของระบบ (Dynamic Diagram) โดยเป็นการแสดงถึงสิ่งที่เกิดขึ้นจากกิจกรรมของ Class ต่างๆ ที่มีอยู่ในระบบ ซึ่งมีแผนผังที่ใช้งาน คือ แผนผังแสดงการทำงานระหว่างออบเจกต์ (Sequence Diagram) และ แผนผังแสดงสถานะ (Statechart Diagram) ซึ่งแสดงสถานะต่างๆ ที่คลาสหนึ่งคลาสจะเป็นได้ในระหว่างช่วงชีวิตในการตอบสนองต่อเหตุการณ์ (Event) ที่เกิดขึ้น

1.3 คอนสตรัคชันเฟส (Construction Phase) เป็นการพัฒนาระบบจริงขึ้น โดยเป็นการเขียนโปรแกรม ซึ่งมีการพัฒนาแบบทำซ้ำและเพิ่มขึ้นเรื่อยๆ ซึ่งกระบวนการที่ทำซ้ำจะประกอบด้วย ขั้นตอนการวิเคราะห์ ออกแบบ เขียนโปรแกรม และการทดสอบ จากนั้นทำการรวมเป็นระบบใหญ่ขึ้นจนได้ระบบที่ต้องการผลลัพธ์ของการทำงานช่วงนี้คือ ระบบที่ต้องการ

1.4 ทรานซิชันเฟส (Transition Phase) เป็นกระบวนการของการส่งผลิตภัณฑ์ไปสู่ผู้ใช้งานจริง รวมไปถึงการหาคัดลอกหรือการแพ็คเกจจิ้ง (Packing) และการบำรุงรักษาและการสอนการใช้โปรแกรมและจัดทำคู่มือการใช้โปรแกรม

2. ส่วนประกอบของ UML

2.1 มุมมอง (View) เป็นระบบงานทั้งหมดอาจมีหลายส่วนที่ต้องพิจารณา เพราะอาจมีขอบข่ายงานที่กว้างขวางและซับซ้อนการอธิบายกระบวนการทำงานต่างๆ ของระบบไม่สามารถอธิบายได้เพียงแค่มุมมองเดียว ดังนั้นการมองระบบควรจะต้องเป็นมุมมองต่างๆ กัน เช่น มุมมองด้าน Functional, Nonfunctional มุมมองขององค์กร เป็นต้น ซึ่งแต่ละไดอะแกรมที่จะมีมุมมองของผู้ใช้งานระบบ ผู้เขียนโปรแกรมพัฒนาระบบ ซึ่งแต่ละมุมมองทำให้ผู้ทำระบบเข้าใจระบบในแง่มุมมองที่ต่างๆ กัน มุมมองต่างๆ ของ UML มีดังนี้

2.1.1 มุมมองการใช้งาน (Use Case View) เป็นการมองระบบจากผู้ใช้ภายนอกหรือผู้ใช้ระบบซึ่งไดอะแกรมที่ใช้อธิบายคือ ยูสเคสไดอะแกรม (Use-Case Diagram) หรือบางครั้งแอกทิวิตี้ไดอะแกรม (Activity Diagram) ตัวอย่างผู้ใช้ระบบ เช่น ลูกค้าผู้ออกแบบ ผู้ทดสอบระบบนักเรียน อาจารย์ เป็นต้น ยูสเคส (Use Case) ในยูสเคสไดอะแกรมเป็นตัวกำหนดเป้าหมายของระบบ จึงเป็นตัวกลางของมุมมองอื่นๆ ที่จะต้องมีการทำงานต่างๆ ควบคู่กันตามที่กำหนดไว้ในยูสเคสไดอะแกรม

2.1.2 มุมมองทางตรรกะ (Logical View) ใช้อธิบายว่าสามารถที่จะจัดการทำงานของของระบบให้เป็นไปตามที่ต้องการได้อย่างไรและมีบริการอะไรให้กับผู้ใช้บ้าง Logical View ต่างจาก Use Case View เนื่องจากเป็นมุมมองของผู้ออกแบบและพัฒนาระบบ โดยจะแสดงในรูปแบบของโครงสร้างแบบสถิต (Static) เช่น คลาส ออบเจกต์ (Object) ความสัมพันธ์ระหว่างการทำงานร่วมกันแบบไดนามิก (Dynamic Collaboration) ซึ่งเกิดเมื่อออบเจกต์ส่งเมสเสจระหว่างการทำงาน

2.1.3 มุมมองในการนำไปใช้ (Deployment View) เป็นการแสดงการจัดระบบในระดับกายภาพ (Physical) ให้เหมาะสม เช่น การเชื่อมต่อระหว่างคอมพิวเตอร์และโหนดต่างๆ และรวมถึงการแมพ (Map) คอมโพเนนต์ต่างๆ ในระดับโครงสร้างทางกายภาพ เช่น ลำดับของหรือโปรแกรมในแต่ละเครื่องคอมพิวเตอร์ใช้สำหรับผู้พัฒนาระบบ ผู้ร่วมพัฒนาระบบ ผู้ทดสอบระบบอธิบายโดยดีพลอยเมนต์ไดอะแกรม (Deployment Diagram)

2.1.4 มุมมองของกระบวนการ (Process View) ไดอะแกรมเป็นกราฟซึ่งแสดงโดยสัญลักษณ์ที่จัดเรียงขึ้น เพื่อใช้อธิบายระบบในมุมมองต่างๆ ในระบบหนึ่งๆ จะประกอบไปด้วยหลายๆ ไดอะแกรม แต่ละไดอะแกรมยังสามารถมองได้หลายๆ มุมมองด้วย

3. ไดอะแกรมใน UML

3.1 ยูสเคสไดอะแกรม (Use-Case Diagram) สิ่งที่สำคัญในการสร้างยูสเคส คือ การค้นหาว่าระบบทำงานอะไรได้บ้าง โดยไม่สนใจว่าข้างในสิ่งที่ระบบต้องทำมีกลไกการทำงานอย่างไรหรือใช้เทคนิคการสร้างอย่างไรเปรียบเสมือนเป็น “กล่องดำ” (Black Box) ยูสเคสไดอะแกรมจะแสดงความสัมพันธ์ระหว่างผู้ใช้งานกับระบบ ซึ่งจะมีแอกเตอร์ (Actor) กับระบบ โดยติดต่อผ่านยูสเคสต่างๆ ที่เกี่ยวข้องและจะใช้ในการสื่อสารกับผู้ใช้ เพื่ออธิบายถึงฟังก์ชันการทำงานหลักของระบบยูสเคสไดอะแกรมก็คือ การทำงานต่างๆ ที่ผู้ใช้ต้องการ ซึ่งจะได้มาจากการสอบถามจากผู้ใช้

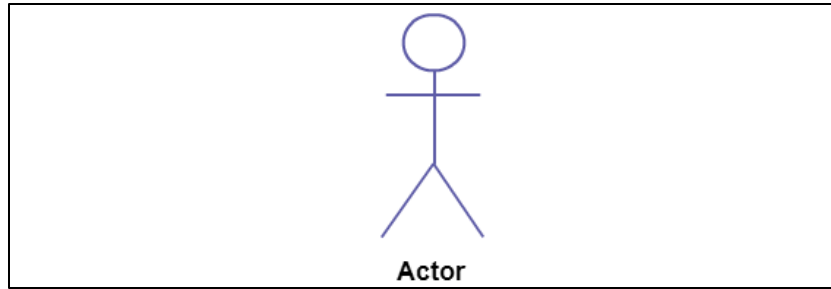
3.1.1 ยูสเคส (Use Case) คือ ความสามารถหรือฟังก์ชันที่ระบบซอฟต์แวร์จะต้องทำได้ เช่น ค้นหาข้อมูลของนักศึกษา คุณสมบัติของยูสเคส จะต้องถูกกระทำโดยแอกเตอร์ และแอกเตอร์เป็นผู้ติดต่อกับระบบตามยูสเคสที่กำหนดไว้ ยูสเคสรับข้อมูลจากแอกเตอร์และส่งข้อมูลให้แอกเตอร์นั้นคือ แอกเตอร์กระทำกับยูสเคสโดยการส่งข้อมูลเข้าสู่ระบบตามยูสเคสหรือรอรับค่าที่ระบบจะส่งกลับให้ยูสเคส ถือว่าเป็นการรวบรวมเอาคุณลักษณะความต้องการในระบบอย่างสมบูรณ์เปรียบเสมือนเป็นการสรุปความต้องการของผู้ใช้ออกเป็นข้อๆ อย่างครบถ้วน โดยการเขียนยูสเคสใช้สัญลักษณ์รูปวงรี และคำอธิบายฟังก์ชันการทำงานอยู่ในวงรีนั้น ดังภาพที่ 4 แสดงยูสเคสการสร้างบัญชีชื่อผู้ใช้บล็อก (Blog)



ภาพที่ 4 แสดงยูสเคสการสร้างบัญชีชื่อผู้ใช้บล็อก (Blog)

ที่มา : ธีรพล ด่านวิริยะกุล. (2549 : 26)

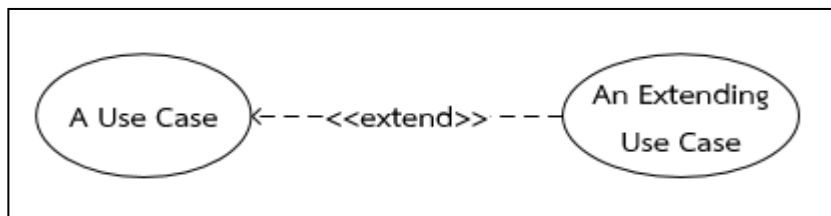
3.1.2 แอกเตอร์ (Actor) คือ ผู้ที่กระทำกับยูสเคสนั้นๆ เช่น นักศึกษา อาจารย์เจ้าหน้าที่ ไม่ใช่ส่วนประกอบของระบบ แต่เป็นส่วนที่ใช้ติดต่อกับระบบ ซึ่งอาจเป็นเพียงการป้อนข้อมูลเข้าสู่ระบบ หรือการส่งข้อมูลออกจากระบบ หรืออาจเป็นทั้งสองอย่างอาจมองได้เป็น แอกเตอร์หลัก หมายถึง แอกเตอร์ที่มีความสำคัญโดยตรงต่อความสามารถหลักของระบบ ซึ่งถูกแสดงด้วยยูสเคสผู้ใช้งานระบบจะให้ความสำคัญกับงานที่แอกเตอร์หลักจะต้องกระทำมากที่สุด แอกเตอร์รอง หมายถึง แอกเตอร์ที่มีหน้าที่สำคัญรองลงไปจาก แอกเตอร์หลัก โดยการเขียนแอกเตอร์จะใช้สัญลักษณ์รูปคน ดังภาพที่ 5 แสดงแอกเตอร์



ภาพที่ 5 ตัวอย่างแอกเตอร์

ที่มา : ธีรพล ตำนาวิริยะกุล. (2549 : 26)

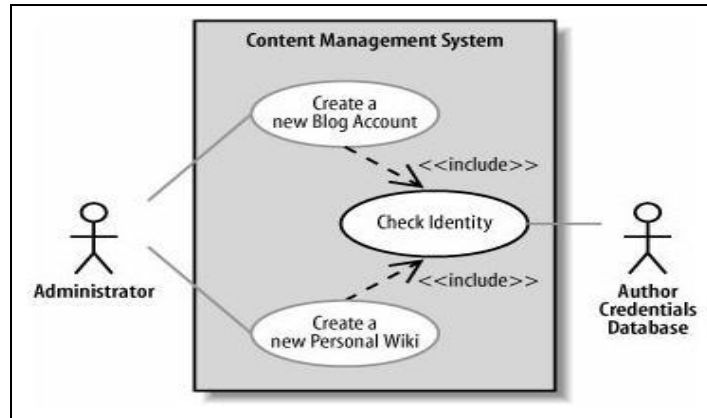
3.1.3 เส้นความสัมพันธ์ระหว่างยูสเคส (Relationship) คือ เส้นเพื่อแสดงความสัมพันธ์ระหว่างแอกเตอร์กับแอกเตอร์ หรือ ยูสเคสกับยูสเคส ซึ่งมีอยู่สองชนิด ได้แก่ ความสัมพันธ์แบบขยาย (Extend Relationship) ยูสเคสหนึ่งอาจถูกช่วยเหลือโดยการทำงาน ยูสเคสอื่น สัญลักษณ์ใน UML คือ ลูกศรเส้นประที่ชี้จากยูสเคสแรกไปยังยูสเคสที่ถูกช่วยเหลือ หรือถูกขยาย โดยมีคำว่า “extend” อยู่ในเครื่องหมายสเตรียโอไทป์ (Stereotype) <<extend>> อยู่กึ่งกลางลูกศร ดังแผนภาพที่ 2 แสดงถึงยูสเคสด้านซ้ายได้รับฟังก์ชันการทำงานจากยูสเคสด้านขวา



แผนภาพที่ 2 ความสัมพันธ์แบบขยาย

ที่มา : ธีรพล ตำนาวิริยะกุล. (2549 : 27)

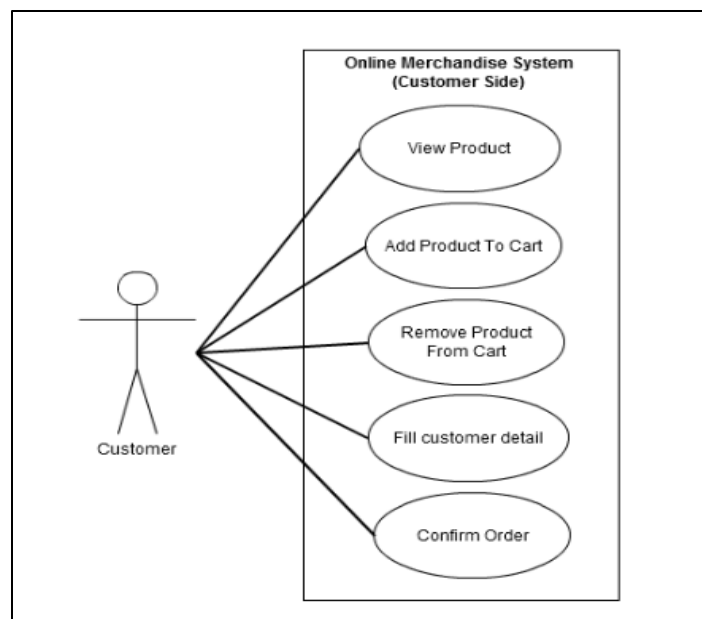
ความสัมพันธ์แบบรวม (Include Relationship) ยูสเคสหนึ่งๆ อาจจำเป็นต้องอาศัยการทำงานของยูสเคสอื่นๆ สำหรับยูสเคสที่ถูกเรียกใช้โดยยูสเคสอื่น สัญลักษณ์ใน UML ของความสัมพันธ์ดังกล่าวคือ ลูกศรเส้นประที่ชี้จากยูสเคสที่ถูกเรียกใช้ โดยมีคำว่า “include” อยู่ในเครื่องหมายสเตรียโอไทป์ <<include>> อยู่กึ่งกลางลูกศร ดังภาพที่ 8 ในการสร้างบล็อก (Blog) ใหม่และสร้างข้อมูลส่วนตัวในเว็บวิกิพีเดียจำเป็นต้องผ่านการตรวจสอบ (Check Identity) ทุกครั้ง



แผนภาพที่ 3 ความสัมพันธ์แบบรวม

ที่มา : อีรพล ด้านวิริยะกุล. (2549 : 27)

ตัวอย่างการเขียนยูสเคสไดอะแกรมของระบบขายสินค้าออนไลน์ ดังแผนภาพที่ 4 เมื่อลูกค้าดูผลิตภัณฑ์ ทำการเพิ่มผลิตภัณฑ์ในรถเข็น สามารถย้ายผลิตภัณฑ์จากรถเข็น ลูกค้าทำการกรอกรายละเอียด และทำการยืนยันการสั่งซื้อ



แผนภาพที่ 4 ตัวอย่างการเขียนยูสเคสไดอะแกรม

ที่มา : ณรงค์ฤทธิ์ คุณแก้ว. (2554 : 28)

3.2 คลาสไดอะแกรม (Class Diagram) แสดงโครงสร้างของส่วนที่ไม่เปลี่ยนแปลงของระบบในมุมมองของผู้พัฒนาระบบ ซึ่งสามารถแสดงความสัมพันธ์ได้หลายวิธี ได้แก่ การเชื่อมต่อระหว่างกัน (Association) การพึ่งพาเรียกใช้คลาสอื่น (Dependent) ความเป็นลักษณะเฉพาะของคลาสอื่น (Specialized) รวมกันเป็นหน่วย (Package) ความสัมพันธ์ระหว่างคลาสต่างๆ เหล่านี้จะถูกแสดงโดยคลาสไดอะแกรม โดยรวมเข้าเป็นโครงสร้างภายในของคลาสเป็นกลุ่มแอททริบิวต์ (Attribute) และกลุ่มโอเปอเรชัน (Operation) ในระบบหนึ่งสามารถประกอบด้วยหลายคลาสไดอะแกรม

3.2.1 คลาส (Class) คือ กลุ่มของออบเจกต์ที่มีคุณสมบัติ (Attributes) และพฤติกรรม (Behavior) ร่วมกันรายละเอียดของสัญลักษณ์คลาส ชื่อของคลาสจะขึ้นต้นด้วยตัวใหญ่แบบหนาและเอียง หากเป็น Abstract Class แอททริบิวต์ประกอบด้วยชนิดของการเข้าถึง (Visibility) ของแอททริบิวต์ ได้แก่ Public ซึ่งถูกแสดงด้วยเครื่องหมาย (+) Private ซึ่งถูกแสดงด้วยเครื่องหมายลบ (-) และโปรเทกต์แสดงด้วยเครื่องหมาย (#) ชื่อของแอททริบิวต์ประเภทของแอททริบิวต์ ซึ่งจะอยู่ต่อจากเครื่องหมายโคลอน (:.) โดยอาจเป็น Primitive Data Type ของแต่ละภาษาโปรแกรมมิ่งซึ่งมักจะคล้ายคลึงกัน เช่น Integer, Boolean, Real เป็นต้น ค่าเริ่มต้นของแอททริบิวต์ คือ Public จะถูกแสดงด้วยเครื่องหมายเท่ากับ

3.2.2 โอเปอเรชันมีชนิดและสัญลักษณ์การเข้าถึงเช่นเดียวกับแอททริบิวต์ มีชื่อโอเปอเรชัน พารามิเตอร์ (Parameters) ประเภทของค่าที่ส่งคืน (Return Type)

3.2.3 ความสัมพันธ์ระหว่างคลาส (Relationships) สามารถแบ่งออกได้เป็นความสัมพันธ์แบบพึ่งพิง (Dependent) การเปลี่ยนแปลงที่เกิดขึ้นกับคลาสที่ถูกพึ่งพิง (Independent Class) จะส่งผลต่อคลาสที่พึ่งพิง (Dependent Class) การโมเดลความสัมพันธ์แบบนี้สามารถทำได้โดยวาดเส้นตรงแบบมีหัวลูกศรเป็นเส้นโค้งชี้จากซับคลาสที่พึ่งพิงไปยังคลาสที่ถูกพึ่งพิงความสัมพันธ์แบบทั่วไป (Generalization) คือ ความสัมพันธ์ระหว่าง Super Class และ Sub Class การโมเดลความสัมพันธ์แบบนี้วาดเส้นตรงหัวทึบที่มีหัวลูกศรเป็นรูปสามเหลี่ยมโค้งชี้จากคลาสไปยัง Super Class ความสัมพันธ์แบบมีความสัมพันธ์กัน (Association) สามารถแบ่งได้เป็น

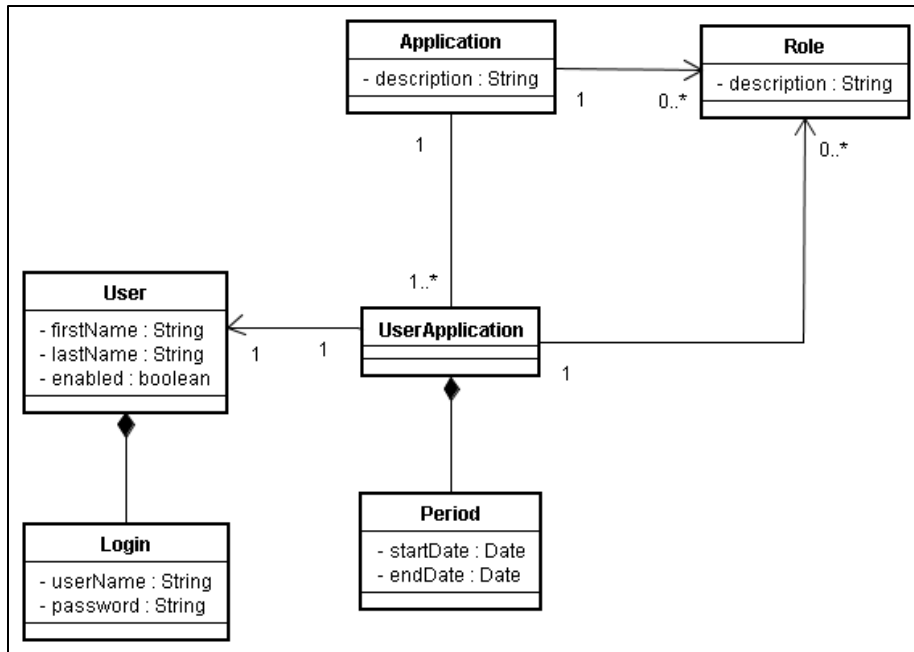
ความสัมพันธ์แบบปกติ (Normal Association) มักใช้ในระบบโมเดลที่ซับซ้อน โดยเฉพาะระบบสารสนเทศ ปกติจะเป็นความสัมพันธ์แบบสองทิศทาง จะวาดด้วยเส้นตรงทึบเชื่อมระหว่างสองคลาสและมีชื่อความสัมพันธ์กำกับอยู่ โดยชื่อนี้มักเป็นคำกริยาเป็นส่วนใหญ่

นอกจากนี้ยังมีการกำหนดปริมาณของคลาสหรือออบเจ็กต์ที่สัมพันธ์กันอยู่ เรียกว่า

Multiplicity

- 1 หมายถึง จะมีออบเจ็กต์ในคลาสไดอะแกรมได้หนึ่งออบเจ็กต์เท่านั้น
- 0...1 หมายถึง จะมีออบเจ็กต์ในคลาสไดอะแกรมได้หนึ่งหรืออาจจะไม่มีก็ได้
- M...N หมายถึง จะมีออบเจ็กต์ในคลาสไดอะแกรมได้ตั้งแต่ M ถึง N (เมื่อ M, N เป็นจำนวนเต็มบวก)
- * หมายถึง จะมีออบเจ็กต์ในคลาสไดอะแกรมได้ตั้งแต่ศูนย์ขึ้นไป
- 0...* หมายถึง จะมีออบเจ็กต์ในคลาสไดอะแกรมได้ตั้งแต่ศูนย์ขึ้นไป
- 1...* หมายถึง จะมีออบเจ็กต์ในคลาสไดอะแกรมได้ตั้งแต่หนึ่งขึ้นไป

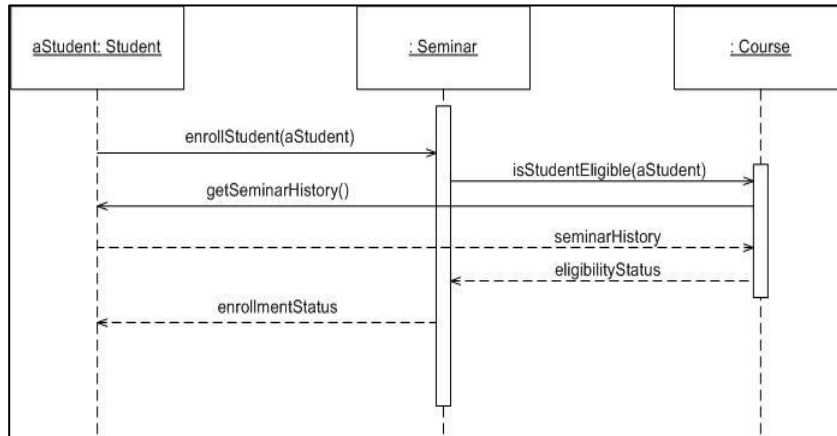
การรวมกัน (Aggregation) เป็นความสัมพันธ์ระหว่างคลาสหรือออบเจ็กต์ในแง่ของการรวมกันแสดงด้วยเส้นทึบโยงระหว่างคลาสโดยมีสัญลักษณ์ข้าวหลามตัดติดอยู่ระหว่างปลายเส้นความสัมพันธ์กับคลาสที่หมายถึงสิ่งที่ใหญ่กว่า และส่วนประกอบ (Composition) คล้ายคลึงกับความสัมพันธ์แบบ Normal Aggregation แต่คลาสที่เป็นองค์ประกอบจะเป็นส่วนหนึ่งของคลาสที่ใหญ่กว่าและเมื่อคลาสที่ใหญ่กว่าถูกทำลายคลาสที่เป็นองค์ประกอบจะถูกทำลายด้วยเส้นที่ใช้แสดงการส่งข้อมูลมีอยู่ 4 ชนิด ได้แก่ เส้นทั่วไป เป็นเส้นที่ใช้ส่งเมสเสจแบบทั่วไปไม่เฉพาะเจาะจงจะถูกแสดงเป็นหัวลูกศรธรรมดา คำอธิบายประกอบเป็นคำอธิบายทั่วไป เส้นชิงโครนัส เป็นเส้นที่ส่งข้อมูลไปแล้วจำเป็นต้องรอผลการตอบกลับเหมาะสำหรับงานแบบเรียลไทม์ (Real Time) ที่หลายๆ งานอย่างน้อยต้องทำพร้อมกันลักษณะเป็นหัวเส้นตรงโปร่งครึ่งซีก และเส้นตรงส่งกลับจากการเรียกใช้ฟังก์ชัน ลักษณะเป็นเส้นตรงประหัวลูกศรหัวโปร่งซีก จากขวามาซ้ายเป็นการ Return From Method Call มักใช้คู่กับเส้นที่ 1 เมื่อเมธอดที่ถูกเรียกใช้มีค่าบางอย่างที่ต้องการส่งกลับมาตัวอย่างการเขียนคลาสไดอะแกรม



แผนภาพที่ 5 ตัวอย่างการเขียนคลาสไดอะแกรม

ที่มา : ชาคริต กุลไกรศรี. (2556 : 30)

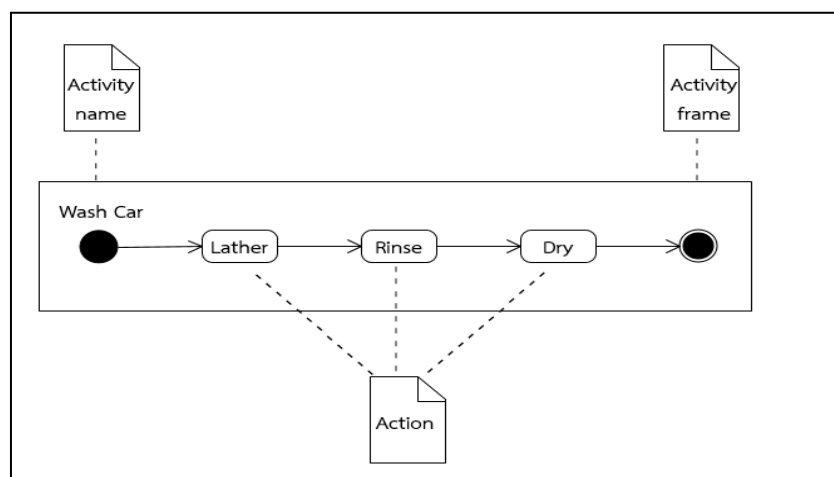
3.3 ซีควেনซ์ไดอะแกรม (Sequence Diagram) จะบอกว่าในยูสเคสนั้นวัตถุแต่ละตัวจะติดต่อสื่อสารกันอย่างไร มีขั้นตอนการทำงานอย่างไร โดยจะเน้นไปที่แกนเวลาเป็นสำคัญ ถ้าเวลาเปลี่ยนขั้นตอนการทำงานจะเปลี่ยนโดยมีแอกเตอร์เป็นผู้เริ่มกระทำเริ่มต้น ซีควেনซ์ไดอะแกรมใน UML จะมีแกนสมมุติ 2 แกนคือแกนตั้ง และแกนนอน แกนนอนจะแสดงขั้นตอนการทำงานหรือการส่งเมสเสจระหว่างวัตถุ โดยแต่ละวัตถุจะส่งข้อมูลถึงกันว่าต้องทำอะไรเมื่อใด ส่วนแกนตั้งเป็นแกนเวลา แกนนอนและแกนตั้งต้องสัมพันธ์กันส่วนวัตถุหรือคลาสแทนด้วยรูปสี่เหลี่ยมเรียงกันตามแนวนอน ภายในบรรจุชื่อออบเจกต์ตามด้วยเครื่องหมายโคลอนและชื่อคลาส เส้นประที่อยู่ในแนวแกนเวลาซึ่งแสดงถึงชีวิตวัตถุ สี่เหลี่ยมแนวตั้งที่อยู่ในตำแหน่งเดียวกับวัตถุหรือคลาส เรียกว่า Activation ซึ่งใช้แสดงช่วงเวลาวัตถุกำลังปฏิบัติงานและส่งข้อมูลระหว่างวัตถุรวมถึงแสดงการสิ้นสุดลงของออบเจกต์หรือการถูกทำลายด้วยเครื่องหมายกากบาทไว้ที่ปลายเส้นชีวิตของออบเจกต์



แผนภาพที่ 6 ตัวอย่างการเขียนซีควเอนซ์ไดอะแกรม

ที่มา : ซีรพล ด่านวิริยะกุล. (2549 : 31)

3.4 แอคทิวิตี้ไดอะแกรม (Activity Diagram) แสดงลำดับการไหลของกิจกรรมต่างๆ โดยจะอธิบายกิจกรรมในลักษณะของการกระทำจะมีเงื่อนไขและการตัดสินใจกำหนดไว้เพื่อควบคุมการไหลของกิจกรรม รวมถึงแมสเชสที่รับส่งระหว่างแต่ละกิจกรรมแสดงด้วยสี่เหลี่ยมมนเหมือนแคปซูล เชื่อมโยงกันด้วยลูกศรเพื่อแสดงลำดับการทำแอคทิวิตี้ (Activity) ถัดไปได้ โดยจะมีเส้นลูกศรชี้เข้ามารวมที่จุดเดียว (เส้นตรงแนวนอน) นั่นคือ แอคทิวิตี้ที่ชี้เข้ามาที่เส้นที่บดงกล่าวเสร็จแล้วก่อน จึงทำให้แอคทิวิตี้ถัดไปได้ การแบ่งเป็นสวิมเลนส์ (Swimlanes) เหมือนสระว่ายน้ำโดยแบ่งช่องในแนวดิ่งและกำหนดแต่ละช่องด้วยชื่อของออบเจกต์ไว้แถบบนสุด



แผนภาพที่ 7 ตัวอย่างการเขียนแอคทิวิตี้ไดอะแกรม

ที่มา : <https://msdn.microsoft.com>. (2558 : 31)

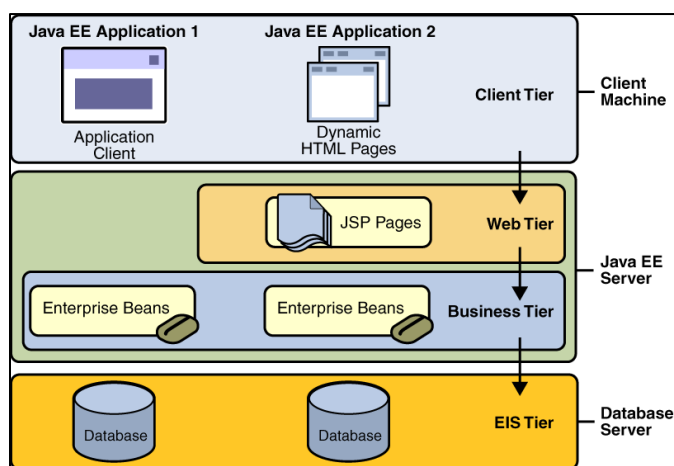
ความรู้พื้นฐานเกี่ยวกับ Java 2 Enterprise Edition (J2EE)

Java 2 Enterprise Edition เป็นกลุ่มของเทคโนโลยีภาษาจาวาที่ใช้การพัฒนาโปรแกรมประยุกต์ด้วยภาษาจาวาเพื่อรองรับการใช้งานระดับองค์กรที่มีผู้ใช้ทั้งภายในและภายนอก องค์กรเป็นจำนวนหลายร้อยจนถึงหลายพันคน ซึ่งระบบจะต้องมีประสิทธิภาพในการทำงานสูงมีความเชื่อถือได้ มีความปลอดภัยสูงสุดแล้ร้กษาไม่ยากและบางครั้งต้องสามารถทำงานร่วมกับระบบเดิมได้ด้วย

Java2EE ได้กำเนิดขึ้นเพื่อลดความยุ่งยากในการสร้างโปรแกรม Enterprise applications โดยการกำหนดมาตรฐานสำหรับ Applications รวมถึงสภาวะแวดล้อมสำหรับการทำงานของ Applications ซึ่งประกอบด้วยข้อกำหนด (specification) ของ Application Server ซึ่งมีบริการ (Services) มาตรฐานสนับสนุนการทำงานของ Applications ทำให้ผู้เขียน J2EE Applications สามารถเน้นการสร้างระบบไปที่ Business Logic โดยไม่ต้องสร้างโปรแกรมเกี่ยวกับการติดต่อบริบบอื่นเช่น ฐานข้อมูล Mail Server เป็นต้น จึงช่วยให้ J2EE Applications สามารถนำไป Reused ได้ง่าย

1. J2EE Architecture

J2EE Architecture ได้ถูกเสนอขึ้นเพื่อลดกลไกในการสร้าง Enterprise Application โดยใช้การกำหนดมาบังคับใช้ ว่าหากบริษัทใดจะนำเอา J2EE Technology ไปผลิตต่อจะต้องมีการจัดการสภาวะแวดล้อมต่างๆ ให้เป็นไปตามที่มาตรฐานกำหนดไว้จึงเรียกว่า J2EE Architecture ซึ่งสามารถแสดงได้ดังภาพ

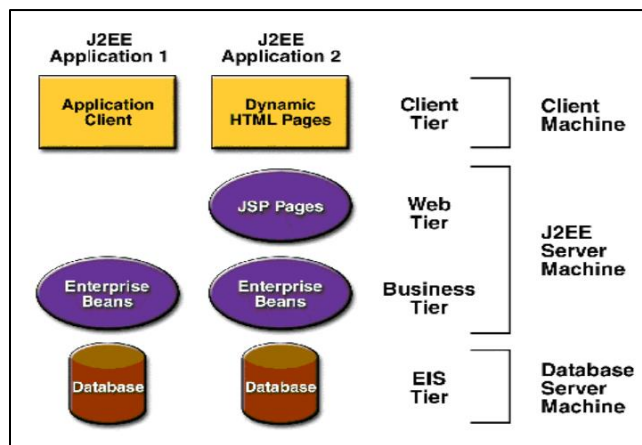


ภาพที่ 6 แสดง J2EE Architecture

ที่มา : ศิริศักดิ์ เสนาราช. (2553 : 32)

เป็นมาตรฐานที่กำหนดให้บริษัทต่างๆ ที่จะจัดทำ Application Server ขึ้นมาว่า จะต้องมีการ API ดังภาพไว้รองรับเรียก API ในภาพรวมกันได้ ว่า Container Service จะสังเกตเห็นได้ว่าการแยก Presentation Logic ออกจาก Business Logic โดยแท้จริง โดยสามารถให้บริการ Presentation Logic ได้ ตาม Request ไม่ว่าจะ เป็น Server page ต่างๆ หรือแม้กระทั่งจะเป็น Application ก็ตามที่และภายใน Container นั้นก็จะไปติดต่อกับบริการอื่นๆ ให้อีกทีโดยใช้ API ที่มีอยู่ใน Server เช่นติดต่อกับฐานข้อมูลใช้ Java mail ฯลฯ

J2EE คือ สถาปัตยกรรมที่มีการกำหนดคุณลักษณะของเครื่องมือที่จะใช้ ในการสร้างระบบงานแบบมัลติเทียร์ (Multi-tier) โดยอยู่บนพื้นฐานของภาษาจาวาและสนับสนุนการทำงานแบบกระจาย



ภาพที่ 7 แสดงโครงสร้างของสถาปัตยกรรมแบบ Multitier Applications
ที่มา : ศิริศักดิ์ เสนาราช. (2553 : 33)

จากภาพ J2EE แบ่งส่วนการทำงานหลักๆ ออกเป็น 4 ส่วน ดังนี้

1.1 ชั้นไคลเอนท์ (Client Tier) เป็นส่วนหน้าของระบบหรือที่เรียกว่า Front end ซึ่งทำหน้าที่ในการติดต่อกับผู้ใช้โดยตรงโดยการประมวลผลของชั้นนี้จะอยู่ที่เครื่องของผู้ใช้หรือไคลเอนท์

1.2 ชั้นเว็บ (Web Tier) เป็นส่วนที่ทำการรับคำร้องขอ (Request) จากชั้นไคลเอนท์แล้วส่งไปประมวลผลในชั้นถัดๆ ไปเมื่อได้ผลลัพธ์แล้วจะทำการส่งผลลัพธ์กลับไปยังชั้นไคลเอนท์ โดยการประมวลผลของชั้นนี้จะอยู่ที่เครื่อง J2EE Server

1.3 ชั้นบิสซิเนส (Business Tier) เป็นส่วนการประมวลผลทางธุรกิจ (Business Logic) เช่น คำนวณอัตราดอกเบี้ยการฝาก และถอนเงินจากธนาคาร เป็นต้น โดยการ

ประมวลผลของชั้นนี้จะอยู่ที่เครื่อง J2EE Server

1.4 ชั้นดาต้า (Enterprise Information System Tier : EIS) เป็นส่วนที่ใช้ในการจัดการข้อมูลในฐานข้อมูลโดยการประมวลผลของชั้นนี้จะอยู่ที่เครื่อง Database Server J2EE มีลักษณะคล้ายกับระบบ Web-base และ N-tier เรียกว่า Distributed Multitiered Applications โดย J2EE จะสร้างสภาพแวดล้อมสำหรับการทำงานของ J2EE Applications เรียกว่า J2EE Containers ซึ่งประกอบด้วยส่วนประกอบดังต่อไปนี้

1.4.1 Web Containers เป็นสภาพแวดล้อมที่รองรับการทำงานของ Servlet และ JSP

1.4.2 Enterprise Java Bean Containers เป็นสภาพแวดล้อมที่รองรับการทำงานของ Enterprise Java Bean (EJB)

1.4.3 Application client Container เป็นสภาพแวดล้อมที่รองรับการทำงานของ Java Application client

1.4.4 Applet container เป็นสภาพแวดล้อมที่รองรับการทำงานของ Java Applet Java Plug in

2. J2EE Container Service

ในหัวข้อนี้จะเป็นการกล่าวถึงการให้บริการของ Container นั่นก็คือ API ทั้งหมดที่มีในการจัดไว้ให้บริการ API ต่างๆ มีหน้าที่ดังต่อไปนี้

2.1 JDBC (Java Database Connectivity) สำหรับการติดต่อ Database Server

2.2 RMI-IIOP (Remote Method Invocation over the Internet Inter-ORB Protocol) เป็นการเรียกใช้ Method ระยะไกลที่อยู่บนละ JVM

2.3 JNDI (Java Naming and Directory Interface) สำหรับติดต่อกับ Naming Directory Interface

2.4 JMS (Java Message Service) สำหรับติดต่อกับ Message Servers

2.5 JTA (Java Transaction) สำหรับจัดการ Distributed Transaction

2.6 Java Mail สำหรับติดต่อกับ Mail Server

3. J2EE Application Component

ใน J2EE Server ก็สามารถแบ่งได้ เป็น 2 Container ด้วยกัน ก็คือ Web-Container สำหรับ Server Page Technology และสำหรับ Distributed Programming อย่าง EJB- Container ทั้งสอง Container เป็นเพียงการเตรียมสถานะแวดล้อมให้ Component ด้านในทำงานได้อย่างสะดวกเท่านั้น จะมีบางกรณีเท่านั้นที่ Container ต้องจัดการทำอะไรเองบางอย่าง ถึงอย่างไรก็ตามในการพัฒนางานจริงไม่จำกัดว่า Presentation Logic จะเป็น Application หรือว่า Server Page แต่สามารถ Tunneling ใช้ งานได้ทั้งสอง Container ซึ่งปัจจุบันก็ยังคงใช้ Browser เป็นหลักอยู่ โดยขอแนะนำ Component ทั้งหมดที่อยู่ใน J2EE Server Web Component เป็น Component ที่รองรับ request จาก Client ในลักษณะของ Server Page เป็นหลัก โดย Application Server ทุกตัวที่ให้บริการภาษา Enterprise ของ Java จะสามารถใช้บริการ Component เหล่านี้ได้ หมดทุก Application Server เช่น Tomcat Sun Application Server ฯลฯ โดยด้านใน Web Container จะประกอบด้วย Component ดังต่อไปนี้

3.1 Servlets เมื่อก่อนที่ CGI (Common Gateway Interface) เป็นที่นิยมใช้ ไม่ว่าจะ เป็น ASP หรือ PHP ก็ตามที่ Client จะสามารถติดต่อกับ Server ได้ ทาง CGI โดยการขอ Request ผ่าน Browser ไปที่ Server และรับ Service ก่อนที่จะ Response กลับคืนมา แต่ CGI จะมีข้อเสียหลักๆ ดังต่อไปนี้

- 3.1.1 ต้องโหลดและประมวลผลคำสั่งใหม่ทุกครั้งที่ใช้งาน
- 3.1.2 ไม่สามารถทำ MVC Programming ได้
- 3.1.3 ขาดเสถียรภาพและการควบคุมทางการรักษาความปลอดภัย
- 3.1.4 ภาษาที่ใช้มักจะเป็น Imperative Programming
- 3.1.5 ใช้งานได้เฉพาะ Presentation Logic Technology ที่สามารถทำงานได้ บน Browser เท่านั้น

ปัญหาดังกล่าวถูกแก้ไขโดยการสร้าง Component ตัวหนึ่งขึ้นมาบน web container มีชื่อว่า Servlet โดยมีลักษณะการทำงานแบบ OOP ซึ่ง Servlet เป็น Object หนึ่งที่มีหน้าที่รับ Request จาก Client ซึ่งอาจจะผ่านทาง Presentation ที่เป็น Server Page หรือ Application ก็ได้โดย Servlet จะเสียเวลาในการ Compile ตัวเองหนึ่งครั้งและจะได้ตัว Servlet ที่ Compile แล้วเป็น class เพราะ Servlet เป็น Class ใน Java จากนั้นพอสั่งทำงานอีกครั้ง Servlet ก็สามารถทำงานได้ เลยโดยไม่ต้อง Compile อีกและนี่คือข้อดีที่มี

เหนือ CGI โดยเราสามารถเรียกใช้งาน Servlet โดยการใช้กฎของ Encapsulation เรียกใช้ Method ธรรมดา

3.2 JSP (Java Server Page) จริงๆ แล้ว JSP (Java Server Page) ก็คือ Servlet นั่นเอง ที่ JSP ถูกสร้างขึ้นมานั้นก็เพื่อความสะดวกในการพัฒนางานในส่วนที่ Client ต้องการใช้งานในลักษณะของ Server Page การที่จะให้ Web Designer ซึ่งมีหน้าที่ในการตกแต่งหน้า web โดยตรงมาเขียนโปรแกรม Servlet ในภาพแบบของภาษา Java ก็คงจะไม่สะดวก JSP Technology จึงเป็นการใช้เทคโนโลยี การแสดงผลเดิมของ HTML รวมเข้ากับการใช้ Java ใน Server page ได้ เลยจนเกิดเป็น JSP ในที่สุด

3.3 EJB Component ในส่วนของ EJB Component เป็นส่วนที่ต้องอาศัยความสามารถของ EJB Container อยู่มาก เพราะเป็นการคุยกันระหว่าง Server กับอีก Server หนึ่งที่กำลังทำงานอยู่โดยอีก Server หนึ่งอาจจะเป็น Server จากค่ายอื่นค่ายเดียวกัน หรืออาจจะเป็น Database Server ก็เป็นได้ ตามปกติแล้ว Concept ที่ Sun วางไว้สำหรับการให้บริการของ EJB Component คือต้องการให้คุยกับ Application โดยตรงแต่ในประเทศไทยส่วนใหญ่แล้วถือว่ายังไม่พร้อมที่จะใช้ Presentation Logic ที่เป็น Application จึงยังใช้การ Tunneling โดยใช้ Web Browser เป็นหลักอยู่เช่นเคย โดยจะกล่าวถึง EJB Component ดังต่อไปนี้

3.4 Session Bean หรือเป็นบริการเป็น Bean ที่มีมักจะเป็น Component ตัวแรกที่คุยกับ Client และเป็นหน้าที่ของ Session Bean อีกนั่นเองที่จะเป็นตัวเชื่อมให้การทำงานของ Client กับ Component อื่นต่อไปโดยสามารถแบ่งการทำงานตามลักษณะการรักษา Session ของ Client ออกเป็นสองอย่าง คือ

3.4.1 Session Bean Stateless เป็นการรักษา Session แบบไม่ถาวร

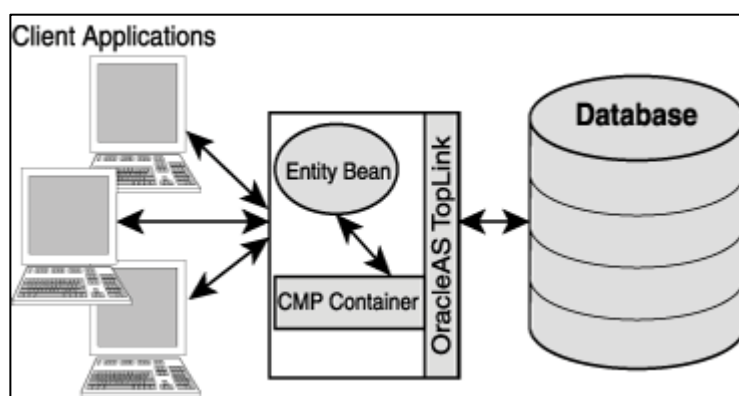
3.4.2 Session Bean State full เป็นการรักษา Session แบบถาวร ซึ่งส่วนใหญ่ในการใช้งานจริงจะใช้ Stateless เป็นหลัก

3.5 Entity Bean ดังที่ได้ กล่าวไปแล้วว่า EJB Container มีความสามารถพิเศษที่สามารถจะไปคุยกับ Server อื่นได้อย่างเต็มที่ โดยเฉพาะในเรื่องการติดต่อกับ Database Server ซึ่งในเรื่องนี้ มีแนวคิดคล้ายๆ กับ OODBMS โดยเวลาที่มีการติดต่อ Database Server จาก EJB Container จะต้องใช้คำสั่งทุกสิ่งทุกอย่างเป็น Model เชิงวัตถุทั้งหมด เช่น หากใช้คำสั่ง sql ในการ query ข้อมูลจาก RDBMS ก็จะต้องสร้าง Collection Framework หรือ Data Structure อะไรบางอย่างมาเก็บไว้เอง แต่ถ้าหากเป็น EJB Container จะต้องใช้ EJB-QL ในการ query ข้อมูลขึ้นมาโดยส่ง Object เข้าไป query จาก

ฐานข้อมูล สามารถแบ่ง Entity Bean เป็น 2 ประเภท ดังนี้

3.5.1 CMP (Container – Managed Persistence) โดยใช้ Container ในการสร้าง Bean เอง และเป็นที่นิยมใช้มากที่สุดเพราะมีความสะดวก

3.5.2 BMP (Container – Managed Persistence) ใช้ การเขียนคำสั่ง โดยใช้ Bean เป็นตัวติดต่อฐานข้อมูลเอง เป็นวิธีที่เก่าและล้าสมัยไม่มีคนใช้แต่ก็มีข้อดีอยู่บ้าง ตรงที่มีความยืดหยุ่นในการใช้ ฐานข้อมูลมากกว่าโดยสามารถแสดงภาพภาพแสดงแนวคิดของ Entity Bean ได้ ดังนี้



แผนภาพที่ 8 แสดงการทำงานของ Entity Bean ใน EJB Container
ที่มา : ศิริศักดิ์ เสนาราช. (2553 : 36)

4. J2EE Modules

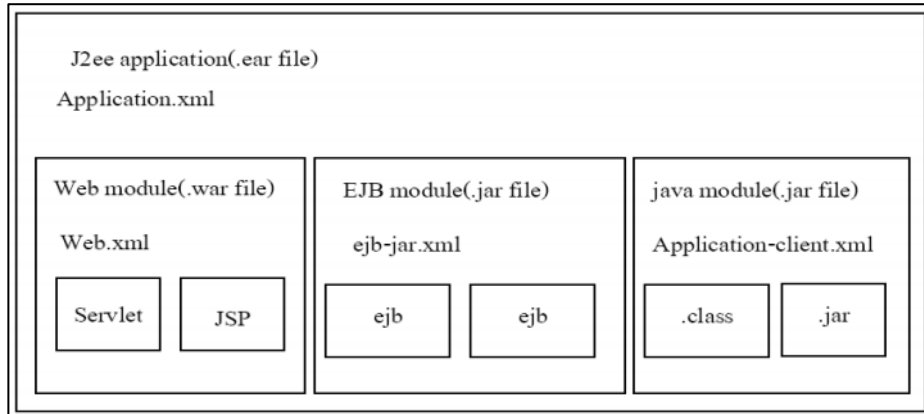
ในการ Deploy หรือการเตรียมการให้สามารถใช้งานได้จริงเมื่อ Programming เสร็จแล้ว ผู้ใช้จะต้องมีความรู้และความเข้าใจในการจัดการ Programming ในแต่ละส่วนแยกเป็นหมวดหมู่ ความยุ่งยากตรงนี้เกิดขึ้นเพราะ J2EE Platform นั้นทำงานอยู่ บน Application Server โดยสามารถแบ่งได้ เป็น 3 Modules ใหญ่ๆดังต่อไปนี้

4.1 Web Modules ใช้ ในการจัดเก็บ Modules ที่ Programming ขึ้นมาโดย Servlet และ JSP โดยจะได้ File .WAR ขึ้นมาพร้อมใช้งาน

4.2 EJB Modules ใช้ ในการจัดเก็บ Modules ที่ Programming ขึ้นมาโดย Session Bean Entity Bean และJMS โดยจะได้ File .jar ขึ้นมาพร้อมใช้ งาน

4.3 Java Modules ใช้ในการจัดเก็บ Modules ที่ Programming ขึ้นมาโดย Java Programming ส่วนใหญ่จะเป็น Client Presentation โดยจะได้ File .jar ขึ้นมาพร้อมใช้งานเท่านั้นยังไม่พอใน Server จะยังไม่รู้จัก File เหล่านี้ จนกว่าจะมีการสร้าง XML File

ขึ้นมาเพื่อเป็น Deployment Descriptor จึงจะสามารถทำงานได้ ดังภาพต่อไปนี้
(ศิริศักดิ์ เสนาราช, 2553 : 18-24)



ภาพที่ 8 แสดง EJB Module

ที่มา : ศิริศักดิ์ เสนาราช. (2553 : 18-24)

ความรู้เบื้องต้นเกี่ยวกับ MySQL

MySQL จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS: Relational Database Management System) ตัวหนึ่ง ซึ่งเป็นที่นิยมกันมากในปัจจุบัน โดยเฉพาะอย่างยิ่งในโลกของอินเทอร์เน็ต สาเหตุเพราะว่า MySQL เป็นฟรีแวร์ทางด้านฐานข้อมูลที่มีประสิทธิภาพสูง เป็นทางเลือกใหม่จากผลิตภัณฑ์ระบบจัดการฐานข้อมูลในปัจจุบัน ที่มักจะเป็นการผูกขาดของผลิตภัณฑ์เพียงไม่กี่ตัว นักพัฒนาระบบ

ฐานข้อมูลที่เคยใช้ MySQL ในความสามารถความรวดเร็ว การรองรับจำนวนผู้ใช้ และขนาดของข้อมูลจำนวนมาก ทั้งยังสนับสนุนการใช้งานบนระบบปฏิบัติการมากมาย ไม่ว่าจะเป็น Unix, OS/2, Mac OS หรือ Windows ก็ตามนอกจากนี้ MySQL ยังสามารถใช้งานร่วมกับ Web Development Platform ทั้งหลายไม่ว่าจะเป็น C, C++, Java, Perl, PHP, Python, Tel หรือ ASP ก็ตามที ดังนั้นจึงไม่น่าแปลกใจเลยว่าทำไม MySQL จึงได้รับความนิยมอย่างมากในปัจจุบันและมีแนวโน้มสูงยิ่งขึ้นไปในอนาคต MySQL จัดเป็นซอฟต์แวร์ประเภท Open Source Software สามารถดาวน์โหลด Source Code ต้นฉบับได้จากอินเทอร์เน็ตโดยไม่เสียค่าใช้จ่ายใดๆ การแก้ไขก็สามารถทำได้ตามความต้องการ MySQL ยึดถือสิทธิบัตรตาม GPL (GNU General Public License) ซึ่งเป็นข้อกำหนดของ

ซอฟต์แวร์ประเภทนี้ส่วนใหญ่โดยจะเป็นการชี้แจงว่า สิ่งใดทำได้หรือทำไม่ได้สำหรับการใช้งานในกรณีต่างๆ

MySQL ได้รับการยอมรับและทดสอบเรื่องของความรวดเร็วในการใช้งาน โดยจะมีการทดสอบและเปรียบเทียบกับผลิตภัณฑ์ทางด้านฐานข้อมูลอื่นอยู่เสมอ มีการพัฒนาอย่างต่อเนื่อง โดยเริ่มตั้งแต่เวอร์ชันแรกๆ ที่ยังไม่ค่อยมีความสามารถมาก มาจนถึงทุกวันนี้ MySQL ได้รับการพัฒนาให้มีความสามารถมากยิ่งขึ้น รองรับข้อมูลจำนวนมหาศาล สามารถใช้งานหลายผู้ใช้ได้พร้อมๆ กัน (Multi-user) มีการออกแบบให้สามารถแตกงานออกเพื่อช่วยการทำงานให้เร็วยิ่งขึ้นรองรับข้อมูลจำนวนมหาศาลเพื่อช่วยการทำงานเร็วยิ่งขึ้น (Multithreaded) วิธีและการเชื่อมต่อที่ดีขึ้น การกำหนดสิทธิและการรักษาความปลอดภัยของข้อมูลมีความรัดกุมน่าเชื่อถือยิ่งขึ้นเครื่องมือหรือโปรแกรมสนับสนุนทั้งของตัวเองและของผู้พัฒนาอื่นๆ มีมากยิ่งขึ้น นอกจากนี้สิ่งหนึ่งที่สำคัญคือ “MySQL ได้รับการพัฒนาไปในแนวทางตามข้อกำหนดมาตรฐาน SQL ดังนั้นเราสามารถใช้คำสั่ง SQL ในการทำงาน MySQL ได้” นักพัฒนาที่ใช้ SQL มาตรฐานอยู่แล้วไม่ต้องศึกษาคำสั่งเพิ่มเติม แต่อาจจะต้องเรียนรู้ถึงรูปแบบและข้อจำกัดบางอย่างโดยเฉพาะทางทีมงานผู้พัฒนา MySQL มีเป้าหมายอย่างชัดเจนที่จะพัฒนาให้ MySQL มีความสามารถสนับสนุนตามข้อกำหนด SQL92 มากที่สุดและจะพัฒนาให้เป็นไปตามข้อกำหนด SQL99 ต่อไป

1. ลักษณะเด่นของ MySQL

1.1 MySQL เป็นระบบจัดการฐานข้อมูล MySQL เป็นโปรแกรมสำหรับใช้ในการสร้างฐานข้อมูล โดยมีคุณลักษณะของระบบจัดการฐานข้อมูลที่ไม่ยิ่งหย่อนไปกว่าระบบจัดการฐานข้อมูลชนิดอื่นๆ รวมทั้งสามารถสร้างและจัดการฐานข้อมูลขนาดใหญ่ได้อย่างรวดเร็วอีกด้วย โดยที่ MySQL มีระบบสืบค้นข้อมูลที่รวดเร็วและแม่นยำ สามารถใช้งานได้กับคอมพิวเตอร์ระบบ Stand-Alone และ Network รวมทั้งทำงานร่วมกับ Application

1.2 MySQL เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ MySQL มีความสามารถในการเชื่อมโยงข้อมูลระหว่างตาราง จัดเก็บข้อมูลจำนวนมาก สะดวก และค้นหาง่าย ซึ่งเป็นคุณลักษณะปกติของโปรแกรมที่พัฒนาขึ้นมาจาก SQL แต่การสร้างฐานข้อมูลเชิงสัมพันธ์ของ MySQL ให้ทางเลือกในการออกแบบ และพัฒนาฐานข้อมูลใช้มากกว่าโปรแกรมจัดการฐานข้อมูลชนิดอื่น

1.3 MySQL เป็นซอฟต์แวร์แบบฟรีแวร์และเป็น Open Source ผู้ใช้ MySQL สามารถพัฒนาโปรแกรมต่อเนื่องได้อย่างอิสระและทุกคนมีสิทธิ์ที่จะ Download ระบบจัดการ

ฐานข้อมูลที่ผ่านมาทาง Internet หรือทำสำเนาได้แต่โปรแกรม MySQL ที่การจดลิขสิทธิ์ ดังนั้น สิทธิทางประการ เช่น การจัดจำหน่ายซอฟต์แวร์ ซึ่งพัฒนามาจาก MySQL หรือการจำหน่าย ซอฟต์แวร์เสริมทางทำงานของ MySQL จะถูกสงวนไว้โดยบริษัทผู้ผลิต

2. โครงสร้างของภาษา MySQL

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบรองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูลที่ต้องใช้ ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการ ของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา asp.net หรือภาษาเจ เอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษาวิซวล เบสิกดอทเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงาน ได้บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนซอร์ซ (Open Source) ที่ถูกนำไปใช้งานมากที่สุด MySQL: มายเอสคิวแอล เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL. แม้ว่า MySQL เป็นซอฟต์แวร์โอเพนซอร์ซ แต่แตกต่างจากซอฟต์แวร์โอเพนซอร์ซทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ ให้ใช้ฟรี และแบบที่ใช้ในเชิงธุรกิจ MySQL สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius. ปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.) เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว ฉะนั้น ผลิตภัณฑ์ภายใต้ MySQL AB ทั้งหมดจะตกเป็นของซันชื่อ "MySQL" อ่านออกเสียงว่า "มายเอสคิวแอล" หรือ "มายเอสคิวแอล" (ในการอ่านอักษร L ในภาษาไทย) ซึ่งทางซอฟต์แวร์ไม่ได้ อ่าน มายซีคิวล หรือ มายซีควล เหมือนกับซอฟต์แวร์จัดการฐานข้อมูลตัวอื่น

2.1 ความสามารถและการทำงานของโปรแกรม MySQL มีดังต่อไปนี้

2.1.1 MySQL ถือเป็นระบบจัดการฐานข้อมูล (Database Management System (DBMS) ฐานข้อมูลมีลักษณะเป็นโครงสร้างของการเก็บรวบรวม ข้อมูล การที่จะเพิ่มเติม เข้าถึงหรือประมวลผลข้อมูลที่เก็บในฐานข้อมูลจำเป็นต้องอาศัย ระบบจัดการ ฐานข้อมูล ซึ่งจะทำหน้าที่เป็นตัวกลางในการจัดการกับข้อมูลในฐานข้อมูลทั้ง สำหรับการ ใช้งานเฉพาะ และรองรับการทำงานของแอปพลิเคชันอื่นๆ ที่ต้องการใช้งานข้อมูล

ในฐานะข้อมูล เพื่อให้ได้รับความสะดวกในการจัดการกับข้อมูลจำนวนมาก MySQL ทำหน้าที่เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล

2.1.2 MySQL เป็นระบบจัดการฐานข้อมูลแบบ relational

ฐานข้อมูลแบบ relational จะทำการเก็บข้อมูลทั้งหมดในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์ เพียงไฟล์เดียว ทำให้ทำงานได้รวดเร็วและมีความยืดหยุ่น นอกจากนี้ แต่ละตารางที่เก็บข้อมูลสามารถเชื่อมโยงเข้าหากันทำให้สามารถรวมหรือจัด กลุ่มข้อมูลได้ตามต้องการ โดยอาศัยภาษา SQL ที่เป็นส่วนหนึ่งของโปรแกรม MySQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงฐานข้อมูล

2.1.3 MySQL แจกจ่ายให้ใช้งานแบบ Open Source นั่นคือ ผู้ใช้งาน

MySQL ทุกคนสามารถใช้งานและปรับแต่งการทำงานได้ตามต้องการ สามารถดาวน์โหลดโปรแกรม MySQL ได้จากอินเทอร์เน็ตและนำมาใช้งานโดยไม่มีค่าใช้จ่ายใดๆ ในระบบปฏิบัติการ Red Hat Linux นั้น มีโปรแกรมที่สามารถใช้งานเป็นฐานข้อมูลให้ผู้ดูแลระบบสามารถเลือกใช้งานได้ หลายโปรแกรม เช่น MySQL และ PostgreSQL ผู้ดูแลระบบสามารถเลือกติดตั้งได้ทั้งในขณะที่ติดตั้งระบบปฏิบัติการ Red Hat Linux หรือจะติดตั้งภายหลังจากที่ติดตั้งระบบปฏิบัติการก็ได้ อย่างไรก็ตาม สาเหตุที่ผู้ใช้งานจำนวนมากนิยมใช้งานโปรแกรม MySQL คือ MySQL สามารถทำงานได้อย่างรวดเร็ว น่าเชื่อถือและใช้งานได้ง่าย เมื่อเปรียบเทียบประสิทธิภาพในการทำงานระหว่างโปรแกรม MySQL และ PostgreSQL นอกจากนี้ MySQL ถูกออกแบบและพัฒนาขึ้นมาเพื่อทำหน้าที่เป็นเครื่องให้บริการรองรับการจัดการกับ ฐานข้อมูลขนาดใหญ่ ซึ่งการพัฒนายังคงดำเนินอยู่อย่างต่อเนื่อง ส่งผลให้มีฟังก์ชันการทำงานใหม่ๆ ที่อำนวยความสะดวกแก่ผู้ใช้งานเพิ่มขึ้นอยู่ตลอดเวลา รวมไปถึงการปรับปรุงด้านความต่อเนื่อง ความเร็วในการทำงาน และความปลอดภัย ทำให้ MySQL เหมาะสมต่อการนำไปใช้งานเพื่อเข้าถึงฐานข้อมูลบนเครือข่ายอินเทอร์เน็ต

การประเมินเครื่องมือในการวิจัย

1. การหาคุณภาพเครื่องมือความเที่ยงตรงตามเนื้อหา

พิสุทธา อารีราษฎร์ (2551 : 119-121) ได้กล่าวว่า ความเที่ยงตรงตามเนื้อหา (Content Validity) หมายถึง การที่ผู้สอนออกแบบทดสอบได้ตรงกับเนื้อหาที่สอน ในการทดสอบความเที่ยงตรงตามเนื้อหา สามารถดำเนินการได้โดยใช้ผู้เชี่ยวชาญในด้านเนื้อหา พิจารณาถึงความสอดคล้องระหว่างวัตถุประสงค์กับแบบทดสอบ โดยพิจารณาเป็นรายชื่อ

วิธีการพิจารณาแบบนี้เรียกว่า การหาค่าสัมประสิทธิ์ความสอดคล้อง (Index of Item – Congruence : IOC) หาค่าความเที่ยงตรงเชิงเนื้อหาของแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน และแบบสอบวัดเจตคติต่อการเรียน

ไพศาล วรคำ (2554 : 260-262) ได้กล่าวว่า เครื่องมือในการเก็บรวบรวมข้อมูลสำหรับการวิจัยทุกประเภทไม่ว่าจะเป็นแบบทดสอบ แบบสอบถาม แบบสังเกต แบบสัมภาษณ์ สามารถหาความเที่ยงตรงเชิงเนื้อหาได้เช่นเดียวกัน การตรวจสอบความเที่ยงตรงเชิงเนื้อหานี้จะต้องดำเนินการก่อนไปทดลองใช้ โดยการนำนิยามเชิงทฤษฎี นิยามเชิงปฏิบัติการ โครงสร้างของข้อคำถาม (รวมทั้งคำตอบ สำหรับกรณีที่กำหนดคำตอบให้ผู้ตอบเลือก) ให้ผู้เชี่ยวชาญพิจารณาความสอดคล้องระหว่างข้อคำถามกับประเด็นที่ต้องการวัดวัตถุประสงค์หรือนิยามศัพท์ในแบบฟอร์มรายงานผลการตรวจสอบเครื่องมือ

สำหรับผู้เชี่ยวชาญที่จะให้ทำการตรวจสอบความสอดคล้องควรมีตั้งแต่ 3 คนขึ้นไป เพื่อหลีกเลี่ยงความคิดเห็นที่แบ่งเป็น 2 ด้าน ดังนั้นควรใช้จำนวนผู้เชี่ยวชาญเป็นจำนวนคี่ เช่น 3 คน 5 คน 7 คน เป็นต้น ส่วนคุณสมบัติของผู้เชี่ยวชาญจะต้องสอดคล้องกับสาขาวิชาของเครื่องมือที่ต้องการตรวจสอบ เช่น ถ้าเป็นการวัดตัวแปรทางจิตวิทยาก็ควรมีผู้เชี่ยวชาญทางด้านจิตวิทยา ผู้เชี่ยวชาญด้านการวัดและประเมินผล และผู้เชี่ยวชาญทางด้านภาษา เป็นต้น ซึ่งจะสามารถประเมินความสอดคล้องและให้ข้อเสนอแนะเกี่ยวกับเครื่องมือครบทั้งในส่วนของเนื้อหา ลักษณะข้อคำถามและความเหมาะสมของภาษาที่ใช้ นำผลการตรวจสอบของผู้เชี่ยวชาญมาคำนวณหาดัชนีที่บ่งบอกถึงความเที่ยงตรงเชิงเนื้อหา ซึ่งคำนวณได้จากความสอดคล้องระหว่างประเด็นที่ต้องการวัดกับข้อคำถามที่สร้างขึ้น ดัชนีนี้เรียกว่า ดัชนีความสอดคล้องระหว่างข้อคำถามกับวัตถุประสงค์ (Index of Item –Congruence : IOC)

สรุป วิธีการหาสัมประสิทธิ์ความสอดคล้อง (Index of Item –Congruence : IOC) สามารถใช้ได้กับเครื่องมือในการเก็บรวบรวมข้อมูลสำหรับการวิจัยทุกประเภท ไม่ว่าจะเป็นแบบทดสอบ แบบสอบถาม แบบสังเกต แบบสัมภาษณ์ ในการศึกษาครั้งนี้ ผู้ศึกษาได้นำวิธีการหาสัมประสิทธิ์ความสอดคล้อง (Index of Item –Congruence : IOC) ในการหาดัชนีความสอดคล้องระหว่างเนื้อหากับแบบประเมินและแบบสอบถามที่พัฒนาขึ้น โดยให้ผู้เชี่ยวชาญ จำนวน 3 คน ประเมินดัชนีความสอดคล้อง (IOC) เพื่อตรวจสอบหาคุณภาพเครื่องมือที่ใช้ในการประเมิน ก่อนทำการจัดทำเป็นเครื่องมือฉบับสมบูรณ์เพื่อใช้ในการเก็บรวบรวมข้อมูลในการวิจัยต่อไป

2. การประเมินผลบทเรียนคอมพิวเตอร์ตามแนวทางวิศวกรรมซอฟต์แวร์

มนต์ชัย เทียนทอง (2554 : 283) ได้กล่าวว่า การประเมินผลบทเรียนคอมพิวเตอร์ตามแนวทางวิศวกรรมซอฟต์แวร์ มีวิธีการทดสอบ ดังต่อไปนี้

- 2.1 White box Testing
- 2.2 Black box Testing
- 2.3 Integration Testing
- 2.4 Performance Testing
- 2.5 Usability Testing

Black box Testing แปลว่า การทดสอบแบบกล่องดำ เปรียบเสมือนการทดสอบภายนอกกล่อง โดยพิจารณาเฉพาะส่วนของการนำเข้าและส่วนของการแสดงผลไม่พิจารณาภายในกล่องอันได้แก่ โครงสร้างของโปรแกรมและรหัสของโปรแกรม ซึ่งเป็นการประเมินผลตรงกันข้ามกับ White box Testing เมื่อนำไปประเมินผลบทเรียนคอมพิวเตอร์ จึงมุ่งเน้นเฉพาะผลลัพธ์ที่ได้จากบทเรียนและส่วนของการนำเข้าเท่านั้น ไม่ได้ให้ความสำคัญกับส่วนประกอบภายในตัวบทเรียนแต่อย่างไร วิธี Black box Testing จึงใช้พิจารณาด้านการทำงาน (Functionality) ตลอดจนคุณสมบัติของบทเรียน (Behavioral) เป็นหลัก ประเด็นทั้งสองนี้จะเกี่ยวข้องกับผู้ออกแบบบทเรียน ผู้เชี่ยวชาญ ผู้สอน และผู้ใช้บทเรียนทั่วไป ซึ่งจะเป็นผู้ประเมินผลบทเรียนหลังจากได้ศึกษาบทเรียนแล้ว ไม่จำเป็นต้องใช้โปรแกรมเมอร์ให้เป็นผู้ประเมินแต่อย่างใด ในการประเมินแบบ White box Testing และ Black box Testing จะใช้แบบสอบถามแบบมาตราส่วนประเมินค่า เพื่อสอบถามความคิดเห็นในประเด็นต่างๆตามที่กำหนดไว้

สรุปในการศึกษานี้ ผู้ศึกษาได้ใช้วิธีการประเมินแบบ Black box Testing หรือ การทำสอบแบบกล่องดำ โดยนำมาใช้พิจารณาทางด้านหน้าที่การทำงาน (Functionality) ตลอดจนคุณสมบัติของบทเรียน (Behavioral) เป็นหลัก ประเด็นที่ใช้ในการประเมิน ประกอบด้วย การประเมินฟังก์ชันการทำงานของระบบ (Functional Test) ด้านการใช้งานของโปรแกรม (Usability Test) ด้านผลลัพธ์ที่ได้จากโปรแกรม (Result Test) ด้านความปลอดภัย (Security Test) และด้านคู่มือการใช้งานระบบ (Documentation)

3. การประเมินความพึงพอใจ

3.1 ความหมายของความพึงพอใจ

ความพึงพอใจ หมายถึง พอใจ ชอบใจ (ราชบัณฑิตยสถาน. 2542 : 775) นอกจากนี้ นักการศึกษาได้ให้ความหมายของความพึงพอใจไว้ดังนี้

ศุภสิริ โสมาเกต (2544 : 9) ได้กล่าวว่า ความพึงพอใจ หมายถึงความรู้สึกนึกคิด หรือเจตคติของบุคคลที่มีต่อการทำงาน หรือการปฏิบัติกิจกรรมในเชิงบวก

ศิริพรรณ ชูติมันตานนท์ (2545 : 32) ได้กล่าวว่า ความพึงพอใจ หมายถึง การแสดงออกถึง ความรู้สึกชอบ มีความสุข ที่เกิดขึ้นเมื่อบุคคลได้รับการสนองความต้องการ หรือได้รับผลสำเร็จตามความมุ่งหมายที่ตนเองได้ตั้งไว้

พิสุทธา อารีราษฎร์ (2551 : 174) ได้กล่าวว่า ความพึงพอใจ (Satisfaction) หมายถึง ความรู้สึกของบุคคลที่มีต่อสิ่งหนึ่งสิ่งใดโดยเฉพาะ ความรู้สึกนั้นทำให้บุคคลเอาใจใส่และอาจกระทำการบรรลุถึงความมุ่งหมายที่บุคคลมีต่อสิ่งนั้น ซึ่งพอสรุปได้ว่า ความพึงพอใจเป็นเรื่องของความรู้สึก ทักษะคติ หรือระดับความพึงพอใจที่มีต่อสิ่งนั้น สามารถตอบสนองความต้องการหรือทำให้บรรลุจุดมุ่งหมายนั้นๆ ได้

มนต์ชัย เทียนทอง (2554 : 306) ได้กล่าวว่า ความพึงพอใจ หมายถึง ความชื่นชม ความนิยม หรือความรู้สึกยอมรับในสิ่งที่ได้เห็นหรือได้สัมผัส สำหรับความพึงพอใจของผู้เรียน จะเป็นการสอบถามความคิดเห็นของผู้เรียนหลังจากการทดลองใช้บทเรียน ซึ่งเป็นการประเมินผลในภาพรวมว่าผู้เรียนพึงพอใจหรือไม่

สรุป ความหมายของความพึงพอใจ หมายถึง สภาวะของอารมณ์ความรู้สึกของคนที่มีความพอใจที่ได้พบเห็น หรือได้รับสิ่งที่ต้องการให้เกิดขึ้น เช่น ชอบหรือไม่ชอบ ซึ่งเกิดมาจากความสนใจและเจตคติของแต่ละบุคคลที่อาจไม่เท่ากัน ขึ้นอยู่กับการได้รับการตอบสนองในสิ่งที่สงสัยอยากรู้ หรือบรรลุในจุดมุ่งหมายที่ตนตั้งเอาไว้หรือไม่มากนักน้อยเพียงใด

3.2 การวัดความพึงพอใจ

ซึ่งนักวิชาการหลายคนได้กล่าวถึง การวัดความพึงพอใจไว้สรุปได้ดังนี้

โยธิน คັນสนยุท (2530 : 77-86) ได้กล่าวถึงเครื่องมือวัดความพึงพอใจว่าการจะค้นหาว่าบุคคลมีความพึงพอใจหรือไม่ วิธีที่ง่ายที่สุดก็คือการถาม ซึ่งการศึกษาในระยะหลัง ๆ ที่ต้องมีผู้บอกข้อมูลจำนวนมากๆ มักใช้แบบสอบถามที่ใช้มาตราส่วนประมาณค่าตามแบบของลิเคอร์ท (Likert) ประกอบด้วยชุดของคำถาม และมีตัวเลือก 5 ตัว สำหรับเลือกตอบ

คือมากที่สุด มากปานกลาง น้อย น้อยที่สุด และคะแนนความพึงพอใจนั้นสามารถนำมาวิเคราะห์ได้ว่า บุคคลมีความพึงพอใจในด้านใดสูง และด้านใดต่ำโดยใช้วิธีการทางสถิติ

พิสุทธา อารีราษฎร์ (2550 : 178) ได้กล่าวว่า การวัดหรือประเมินประสิทธิภาพของบทเรียนคอมพิวเตอร์ช่วยสอน ด้านความพึงพอใจในการใช้บทเรียนคอมพิวเตอร์ช่วยสอนโดยอาจจะเป็นผู้สอนหรือนักเรียน ถือเป็นวิธีการหนึ่งในการวัดประสิทธิภาพของบทเรียนคอมพิวเตอร์ช่วยสอน ถ้าผู้ใช้มีความพึงพอใจต่อบทเรียนคอมพิวเตอร์ช่วยสอนจะเป็นผลให้นักเรียนยอมรับและตอบสนองการเรียนด้วยความเต็มใจ โดยการสนใจในการเรียน หรือการเข้าร่วมกิจกรรม ซึ่งมีผลทำให้นักเรียนมีผลการเรียนที่ดีขึ้น ในการวัดความพึงพอใจ จะใช้แบบสอบถามวัดทัศนคติ ตามมาตรวัดของลิเคิร์ต (Likert Scales) ซึ่งแบ่งความรู้สึกเป็น 5 ช่วงหรือ 5 ระดับ ดังนี้

ระดับ 5 หมายถึง มีความพึงพอใจมากที่สุด

ระดับ 4 หมายถึง มีความพึงพอใจมาก

ระดับ 3 หมายถึง มีความพึงพอใจปานกลาง

ระดับ 2 หมายถึง มีความพึงพอใจน้อย

ระดับ 1 หมายถึง มีความพึงพอใจน้อยที่สุด

สรุปได้ว่า การวัดความพึงพอใจ เป็นการตรวจสอบความชอบหรือไม่ชอบของบุคคลที่มีต่อสิ่งหนึ่งสิ่งใดตามลำดับขั้นของความพึงพอใจ ซึ่งขึ้นอยู่กับว่าสิ่งนั้นมีสำคัญอยู่ในระดับใด เช่น มาก ปานกลาง หรือน้อยเพียงใด

งานวิจัยที่เกี่ยวข้อง

วิไลพร ไชยสิทธิ์ (2554) การพัฒนาแอปพลิเคชันฝึกทักษะทางคณิตศาสตร์บนเครื่องคอมพิวเตอร์แท็บเล็ตสำหรับนักเรียนระดับประถมศึกษาปีที่ 1 มีวัตถุประสงค์ดังนี้ 1) เพื่อหาประสิทธิภาพของแอปพลิเคชันฝึกทักษะทางคณิตศาสตร์บนเครื่องคอมพิวเตอร์แท็บเล็ตปฏิบัติการแอนดรอยด์ 2) เพื่อเปรียบเทียบผลสัมฤทธิ์ทางการเรียนรู้ด้วยแอปพลิเคชันฝึกทักษะทางคณิตศาสตร์ 3) เพื่อศึกษาความพึงพอใจของนักเรียนที่มีต่อแอปพลิเคชันฝึกทักษะทางคณิตศาสตร์ กลุ่มตัวอย่างในการวิจัยครั้งนี้เป็นนักเรียนชั้นประถมศึกษาปีที่ 1 โรงเรียนบ้านจอมบึง (วาปีพร้อมประชาศึกษา) อ.จอมบึง จ.ราชบุรี จำนวน 30 คน ได้มาโดยการสุ่มอย่างง่าย (Sample Random Sampling) เครื่องมือที่ใช้ในการรวบรวมข้อมูลประกอบด้วย แอปพลิเคชันฝึกทักษะทางคณิตศาสตร์ แบบประเมินประสิทธิภาพของแอปพลิเคชันและ

แบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน สถิติที่ใช้ในการวิเคราะห์ข้อมูลได้แก่ ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐาน และสถิติทดสอบค่าที (t-test Statistic) ผลการวิจัยพบว่า 1) ผลการประเมินคุณภาพของแอปพลิเคชันโดยผู้เชี่ยวชาญ 3 ด้าน คือด้านเนื้อหา ด้านการออกแบบ แอปพลิเคชันและด้านการวัดผลและประเมินผล ภาพรวมทุกด้านมีคุณภาพอยู่ในระดับดีมาก ที่ค่าเฉลี่ยเท่ากับ 4.79 2) ผลการประเมินประสิทธิภาพของแอปพลิเคชันมีประสิทธิภาพเท่ากับ 80.78/95.11 ซึ่งสูงกว่าเกณฑ์ 80/80 3) ผลสัมฤทธิ์ทางการเรียนรู้อีก่อนกับหลังเรียนด้วยแอปพลิเคชัน พบว่า คะแนนหลังการเรียนรู้อีกสูงกว่าก่อนเรียน อย่างมีนัยสำคัญทางสถิติที่ระดับ 0.01 4) ความพอใจของผู้เรียนที่มีต่อแอปพลิเคชัน อยู่ในระดับมาก ที่ค่าเฉลี่ย 4.28

สุกัญญา สิงห์คำ (2557) การพัฒนาระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ต แบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส มีวัตถุประสงค์ดังนี้ 1) การพัฒนาระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส 2) เพื่อศึกษาคุณภาพระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส 3) เพื่อศึกษาความพอใจของผู้ใช้ระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส กับระบบที่พัฒนาขึ้น โดยประชากรและกลุ่มตัวอย่างที่ใช้ในการศึกษาคือ นักศึกษาสาขาเทคโนโลยีสารสนเทศ เครื่องมือที่ใช้ในการศึกษาระบบคือ ระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส แบบประเมินคุณภาพระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส และแบบวัดความพึงพอใจของนักศึกษาสาขาเทคโนโลยีสารสนเทศ สถิติที่ใช้ในการวิเคราะห์ข้อมูล คือ ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน ผลการศึกษาพบว่า 1)ระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส สามารถใช้งานได้อย่างมีคุณภาพ 2)ผลการประเมินคุณภาพจากผู้เชี่ยวชาญ ผลการประเมินอยู่ในระดับมากที่สุด 3)ผลการประเมินความพึงพอใจต่อการใช้งานระบบจ่ายประกันภัยรถยนต์ผ่านอินเทอร์เน็ตแบนด์กิ้ง ด้วยเทคโนโลยีเว็บเซอร์วิส ผลการประเมินระดับมากที่สุด

สุรศักดิ์ แผงเมือง (2558) มีวัตถุประสงค์ คือ 1) เพื่อพัฒนาระบบการอนุมัติสินเชื่อบ้านโดยใช้เทคโนโลยีเว็บเซอร์วิส 2) เพื่อหาคุณภาพของระบบการอนุมัติสินเชื่อบ้านโดยใช้เทคโนโลยีเว็บเซอร์วิส กลุ่มเป้าหมายคือ อาจารย์คณะเทคโนโลยีสารสนเทศมหาวิทยาลัยราชภัฏมหาสารคามซึ่งเป็นผู้เชี่ยวชาญด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศเพื่อประเมินคุณภาพของระบบการอนุมัติสินเชื่อบ้านโดยใช้เทคโนโลยีเว็บเซอร์วิส เครื่องมือในการศึกษาได้แก่ ระบบการอนุมัติสินเชื่อบ้านโดยใช้เทคโนโลยีเว็บเซอร์วิส และแบบประเมินคุณภาพระบบการอนุมัติสินเชื่อบ้าน โดยใช้เทคโนโลยีเว็บเซอร์วิส สถิติที่ใช้ในการวิเคราะห์ข้อมูล คือ

ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน

ผลการศึกษาพบว่า 1) ระบบการอนุมัติสินเชื่อบ้านโดยใช้เทคโนโลยีเว็บเซอร์วิสสามารถใช้งานได้อย่างมีคุณภาพครบทุกระบบงานตามขอบเขตของการศึกษา 2) ผลการประเมินคุณภาพจากผู้เชี่ยวชาญ 3 ท่าน มีผลการประเมินโดยรวมทุกด้านอยู่ในระดับดีมาก ($\bar{X} = 4.50$, S.D. = 0.40)

สรุปผลการศึกษาการพัฒนากระบวนการอนุมัติสินเชื่อบ้าน โดยใช้เทคโนโลยีเว็บเซอร์วิสมีคุณภาพ สามารถทำงานได้ตามขอบเขตของระบบงานที่ได้ศึกษา บรรลุตามจุดประสงค์ของการศึกษาที่ตั้งไว้

พรทิพย์ ทั้งถิร (2558) การพัฒนาแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ มีวัตถุประสงค์ดังนี้ คือ 1) เพื่อพัฒนาแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ 2) เพื่อศึกษาคุณภาพแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ 3) เพื่อศึกษาความพึงพอใจของผู้ใช้งานที่มีต่อการแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ ประชากรและกลุ่มตัวอย่างที่ใช้ในการศึกษาประชากร คือ นักศึกษาสาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยราชภัฏมหาสารคาม จำนวน 500 คน กลุ่มตัวอย่าง คือ นักศึกษาสาขาเทคโนโลยีสารสนเทศชั้นปีที่ 1 จำนวน 30 คน เลือกแบบเจาะจงจากนั้นใช้วิธีการสุ่มอย่างง่ายด้วยวิธีการจับฉลาก เครื่องที่ใช้ในการศึกษาได้แก่ แอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ แบบประเมินคุณภาพของแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ และแบบประเมินความพึงพอใจของแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ สถิติที่ใช้ในการวิเคราะห์ข้อมูล คือ ค่าเฉลี่ย (\bar{X}) และส่วนเบี่ยงเบนมาตรฐาน (S.D.) ผลการศึกษาพบว่า 1) ได้แอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์สามารถใช้งานได้อย่างมีคุณภาพ 2) ผลการประเมินคุณภาพจากผู้เชี่ยวชาญ 3 คน มีความคิดเห็นต่อแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ ระดับความคิดเห็นโดยรวมประเมินอยู่ในระดับดีมาก ($\bar{X} = 4.63$, S.D. = 0.17) 3) ผลการประเมินความพึงพอใจต่อการใช้งานแอปพลิเคชันการสอบรายวิชาเทคโนโลยีเว็บ บนระบบปฏิบัติการแอนดรอยด์ 30 คน ผลการประเมินความพึงพอใจอยู่ในระดับพอใจมากที่สุด ($\bar{X} = 4.70$, S.D. = 0.05)

พิพัฒน์ ประทุมวัน (2558) การพัฒนาระบบงานที่คืนโดยใช้เทคโนโลยีเว็บ
 เซอร์วิส กรณีศึกษา : ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตรจังหวัดมหาสารคาม มี
 วัตถุประสงค์ดังนี้ คือ 1) เพื่อพัฒนาระบบงานที่คืนโดยใช้เทคโนโลยีเว็บเซอร์วิส กรณีศึกษา
 : ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตรจังหวัดมหาสารคาม 2) เพื่อศึกษาคุณภาพ
 ระบบงานที่คืนโดยใช้เทคโนโลยีเว็บเซอร์วิส กรณีศึกษา : ธนาคารเพื่อการเกษตรและ
 สหกรณ์การเกษตรจังหวัดมหาสารคาม โดยกลุ่มเป้าหมายในการศึกษา กลุ่มเป้าหมาย คือ กลุ่ม
 ผู้เชี่ยวชาญด้านคอมพิวเตอร์และเทคโนโลยี เพื่อประเมินระบบงานที่คืนโดยใช้เทคโนโลยีเว็บ
 เซอร์วิส กรณีศึกษา : ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตรจังหวัดมหาสารคาม เป็น
 อาจารย์จากมหาวิทยาลัยราชภัฏมหาสารคามมีคุณสมบัติ คือ สำเร็จการศึกษาระดับปริญญา
 เอกและระดับปริญญาโท สาขาเทคโนโลยีสารสนเทศ หรือ สาขาที่เกี่ยวข้องกับคอมพิวเตอร์
 จำนวน 3 คน ได้แก่ ดร.ธวัชชัย สหพงษ์ วุฒิการศึกษา (ปร.ด) คอมพิวเตอร์ศึกษา อาจารย์
 สาขาวิชาเทคโนโลยีมีเดียและแอนิเมชัน มหาวิทยาลัยราชภัฏมหาสารคาม อาจารย์จารุ
 กิตติ์ สายสิงห์ วุฒิการศึกษา (วท.ม) เทคโนโลยีสารสนเทศ อาจารย์ประจำสาขาเทคโนโลยี
 สารสนเทศ มหาวิทยาลัยราชภัฏมหาสารคาม และอาจารย์วินัย โกหล่า วุฒิการศึกษา (วท.ม)
 เทคโนโลยีคอมพิวเตอร์เพื่อการศึกษา อาจารย์ประจำสาขาเทคโนโลยีคอมพิวเตอร์เพื่อการ
 สื่อสาร มหาวิทยาลัยราชภัฏมหาสารคามเครื่องมือที่ใช้ในการวิเคราะห์ข้อมูล คือ ค่าเฉลี่ย (\bar{X})
 และส่วนเบี่ยงเบนมาตรฐาน (S.D.) ผลการศึกษาพบว่า 1) ระบบงานที่คืนโดยใช้เทคโนโลยี
 เว็บเซอร์วิส กรณีศึกษา : ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตรจังหวัดมหาสารคาม
 สามารถใช้งานได้อย่างมีคุณภาพ 2) ผลการประเมินคุณภาพจากผู้เชี่ยวชาญ 3 คน โดยแบบ
 ประเมินแบ่งออกเป็น 5 ด้าน ได้แก่ ด้านความสามารถในการทำงานตามระบบผู้ใช้งาน ด้าน
 การใช้งานของโปรแกรม ด้านผลลัพธ์ที่ได้จากโปรแกรม ด้านความปลอดภัย และด้านคู่มือการ
 ใช้งานและการติดตั้งระบบผลการประเมินอยู่ในระดับดีมาก ($\bar{X} = 4.66, S.D. = 0.14$)