โครงการวิจัยโดยใช้เงินรายได้คณะวิศวกรรมศาสตร์
ประจำปีงบประมาณ พ.ศ. 2549

ระบบ Real-Time เพื่อตรวจวัดความเสียหายที่เกิดจากการก่อสร้างอุโมงค์

REAL-TIME SYSTEM FOR MONITORING DAMAGES CAUSED BY TUNNELING

เสนอโดย
ผศ.ดร. สุชัชวีร์ สุวรรณสวัสดิ์
อาจารย์อุบะ ศิริแก้ว
ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โครงการวิจัยโดยใช้เงินรายได้คณะวิศวกรรมศาสตร์
ประจำปีงบประมาณ พ.ศ. 2549

ระบบ Real-Time เพื่อตรวจวัดความเสียหายที่เกิดจากการก่อสร้างอุโมงค์

REAL-TIME SYSTEM FOR MONITORING DAMAGES CAUSED BY TUNNELING

RCH
TA
656
ส756ร
เลขหมู่.....
เลขทะเบียน.....84505
วัน,เดือน,ปี.....13 ต.ค. 2551

เสนอโดย
ผศ.ดร. สุชัชวีร์ สุวรรณสวัสดิ์
อาจารย์อุบะ ศิริแก้ว
ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุ

สารบัญ

	หน้า
1. ชื่อโครงการวิจัย	2
2. หน่วยงานหลักที่รับผิดชอบงานวิจัย	2
3. คณะผู้วิจัย	2
4. ประเภทของการวิจัย	2
5. สาขาวิชาการที่ทำการวิจัย	2
6. คำสำคัญ (Keywords) ของโครงการวิจัย	2
7. ความสำคัญและที่มาของปัญหา (Problem Identification)	3
8. วัตถุประสงค์ของโครงการวิจัย	6
9. ขั้นตอนการวิจัย	6
10. คุณสมบัติและรายละเอียดการทำงานของชิ้นงาน โครงงานวิจัย	7
11. ประโยชน์ที่คาดว่าจะได้รับ	9
12. หน่วยงานที่จะนำงานวิจัยไปใช้ประโยชน์	9
13. ทฤษฎีและกรอบแนวความคิด (Theory and Conceptual Framework)	9
13.1 การพัฒนาระบบเชิงวัตถุ (Object Oriented)	9
13.2 ภาษายูเอ็มแอล	16
13.3 เทคโนโลยีด็อทเน็ตเฟรมเวิร์ค (.NET Framework)	20
13.4 สถาปัตยกรรมของด็อทเน็ตคอมแพคเฟรมเวิร์ค	27
13.5 เทคโนโลยีเอดีโอด็อทเน็ต (ADO.NET)	29
13.6 เอสคิวแอล เซิร์ฟเวอร์ 2000	32
13.7 การออกแบบฐานข้อมูลโดยการใช้ไนแอมโมเดล (NIAM Model)	34
13.8 Web Services	35
13.9 หลักการในการทำระดับ (Principles of Leveling)	37
13.10 พฤติกรรมของดินต่อการก่อสร้างอุโมงค์	41
14. การออกแบบ	46
14.1 การออกแบบระบบ	46
14.2 การออกแบบโครงงาน	47
15. ผลการทดสอบการทำงาน	59
15.1 ส่วนการทำงานของการเปลี่ยนแปลง Site งานการขุดอุโมงค์	59
15.2 ส่วนการทำงานของการเปลี่ยนแปลง Section งานการขุดอุโมงค์	63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

15.3 ส่วนการทำงานของการเปลี่ยนแปลง Instrument งานการขุดอุโมงค์ 64
15.4 ส่วนการเรียกดูข้อมูลเพื่อเปรียบเทียบการพังทลายของงานการขุดอุโมงค์ 64
15.5 ส่วนการบันทึกข้อมูลโดยตรงจากศูนย์ปฏิบัติการ (Direct Record) 66
15.6 ส่วนการบันทึกข้อมูลจาก Site งานผ่าน Pocket PC Phone 68
15.7 ส่วนการบันทึกข้อมูลจาก Site งานผ่าน Pocket PC Phone 70
16. แผนการถ่ายทอดเทคโนโลยีหรือผลการวิจัยสู่กลุ่มเป้าหมาย 71
17. เอกสารอ้างอิง 72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบ Real-Time เพื่อตรวจวัดความเสียหายที่เกิดจากการก่อสร้างอุโมงค์

REAL-TIME SYSTEM FOR MONITORING DAMAGES CAUSED BY TUNNELING

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดโครงการวิจัย

1. **ชื่อโครงการวิจัย** <u>ระบบ Real-Time เพื่อตรวจวัดความเสียหายที่เกิดจากการก่อสร้างอุโมงค์</u>
   <u>(REAL-TIME SYSTEM FOR MONITORING DAMAGES CAUSED BY TUNNELING)</u>

2. **หน่วยงานหลักที่รับผิดชอบงานวิจัย**
   ภาควิชาวิศวกรรมสารสนเทศ และ ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์
   สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

3. คณะผู้วิจัย
   หัวหน้าโครงการวิจัย ผศ.ดร. สุชัชวีร์ สุวรรณสวัสดิ์
   ผู้ร่วมโครงการวิจัย อาจารย์อุบะ ศิริแก้ว

4. ประเภทของการวิจัย การวิจัยและพัฒนาเพื่อประยุกต์ใช้ในงานจริง

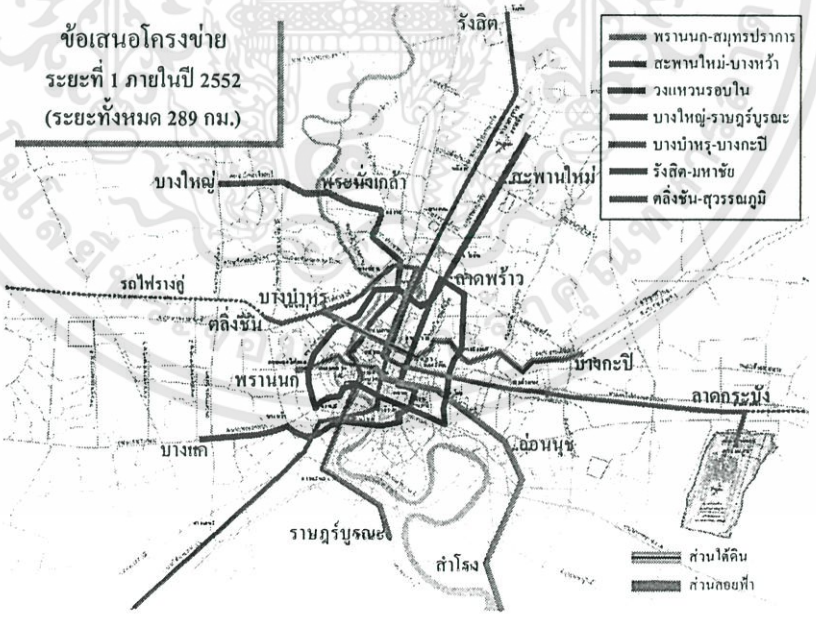
5. สาขาวิชาการที่ทำการวิจัย
   วิศวกรรมสารสนเทศ (Information Engineering)
   วิศวกรรมโทรคมนาคม (Telecommunication Engineering)
   วิศวกรรมเทคนิคธรณี (Geotechnical Engineering)
   วิทยาการคอมพิวเตอร์ (Computer Science)
   วิศวกรรมอิเล็กทรอนิกส์ (Electronics Engineering)

6. **คำสำคัญ (Keywords) ของโครงการวิจัย**
   Real-Time Monitoring
   Instrumentation
   Wireless System
   Tunneling
   Database
   Computer Software
   Web Application
   Mobile Phone Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

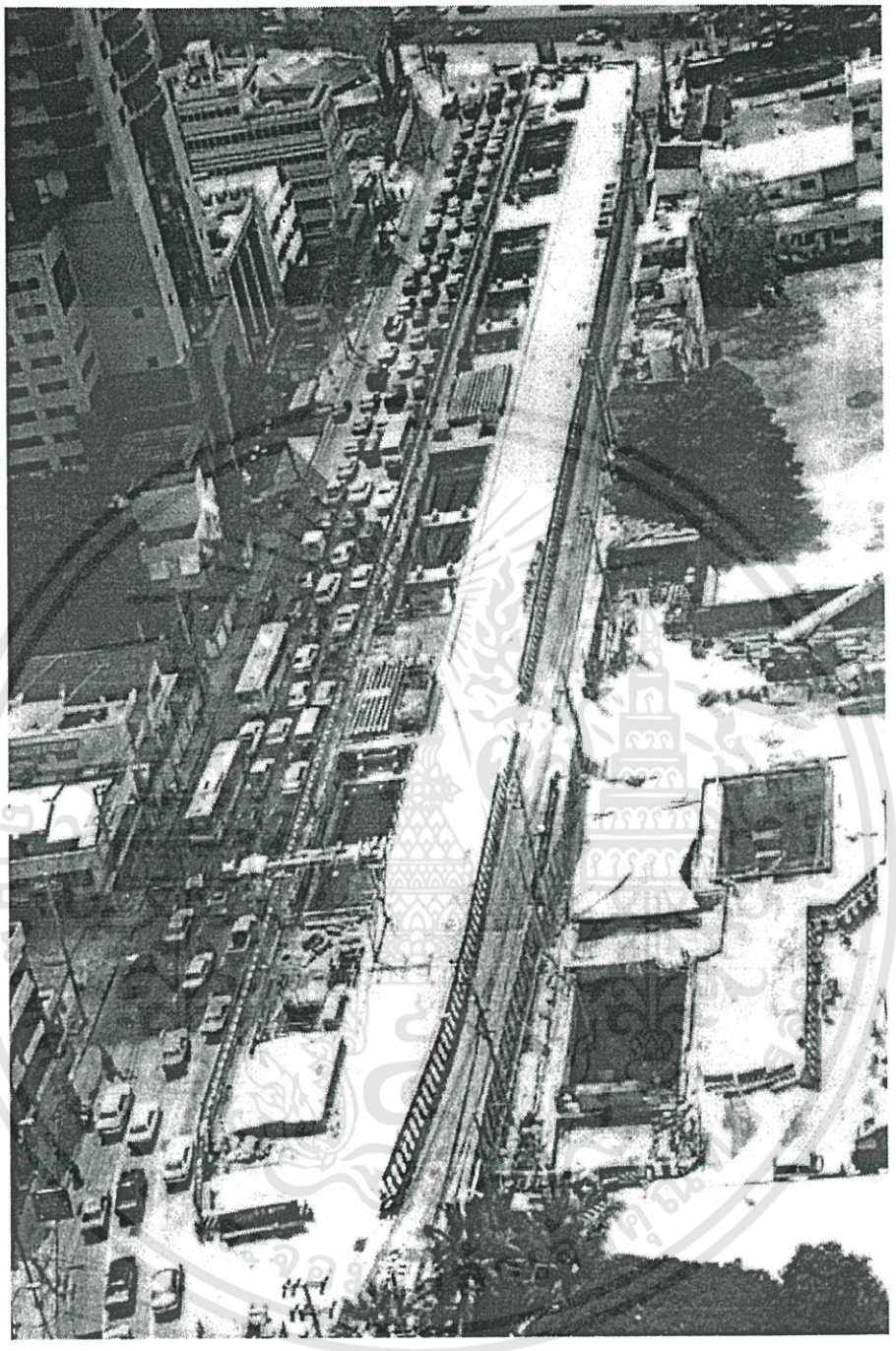
7. ความสำคัญและที่มาของปัญหา (Problem Identification)

ปัจจุบันรัฐบาลได้วางแผนพัฒนาระบบสาธารณูปโภค เพื่อรองรับการขยายตัวของกรุงเทพฯ และลดปัญหาการจราจรติดขัด โดยระบบดังกล่าวประกอบด้วย การก่อสร้างระบบรถไฟฟ้า ทั้งเป็นแบบโครงสร้างยกระดับและเป็นแบบโครงสร้างใต้ดิน (รูปที่ 1) อีกทั้งยังรวมไปถึง การก่อสร้างอุโมงค์ระบายน้ำเพื่อป้องกันน้ำท่วม อุโมงค์ส่งน้ำประปา และอุโมงค์สำหรับสายส่งกระแสไฟฟ้า ซึ่งโครงการส่วนใหญ่จะต้องก่อสร้างในเขตที่มีประชาชนอยู่อย่างหนาแน่น (รูปที่ 2) และมีสิ่งปลูกสร้างอยู่ใกล้เคียงจำนวนมาก เนื่องจากการก่อสร้างโครงสร้างสาธารณูปโภคขนาดใหญ่ จะต้องมีการขุดดินไม่ว่าจะเป็นกรณีของการก่อสร้างฐานรากเสาเข็มขนาดใหญ่ หรือขุดเจาะอุโมงค์ใต้ดิน ดังนั้นจึงมักสร้างความเสียหายต่อโครงสร้างใกล้เคียง ดังแสดงในรูปที่ 3 ถึงรูปที่ 4 ที่อาคารใกล้เคียงสถานีรถไฟฟ้าใต้ดินเกิดการทรุดตัวจนแตกร้าวจนอาจเป็นอันตรายต่อผู้อาศัย ซึ่งในอดีตวิธีการตรวจวัด (Monitoring System) ความเสียหายยังต้องใช้เวลา ในกระบวนการตรวจสอบและรายงานผล นานเป็นสัปดาห์ ซึ่งกว่าที่จะทราบถึงปัญหา ก็มักจะสายเกินการป้องกันหรือแก้ไขได้ทันท่วงที ดังนั้นโครงการวิจัยเพื่อสร้างระบบรายงานผลการตรวจวัดความเสียหายของโครงสร้างที่ได้รับผลกระทบจากการก่อสร้างขนาดใหญ่ จะสามารถช่วยให้วิศวกรสามารถตรวจสอบความเสียหายโดยใช้อุปกรณ์ไร้สาย เช่น Pocket PC หรือ Personal Digital Assistance (PDA) และเว็บไซต์ ที่สามารถเข้าถึงได้ทุกที่ทุกเวลา (รูปที่ 5) ซึ่งจะสามารถช่วยให้ประเมินความเสียหายได้ทันที (Real-time) โดยจะช่วยให้เกิดความปลอดภัยต่อชีวิตและทรัพย์สินได้อย่างมาก



รูปที่ 1 โครงข่ายการขนส่งมวลชนรวมส่วนลอยฟ้าและใต้ดิน

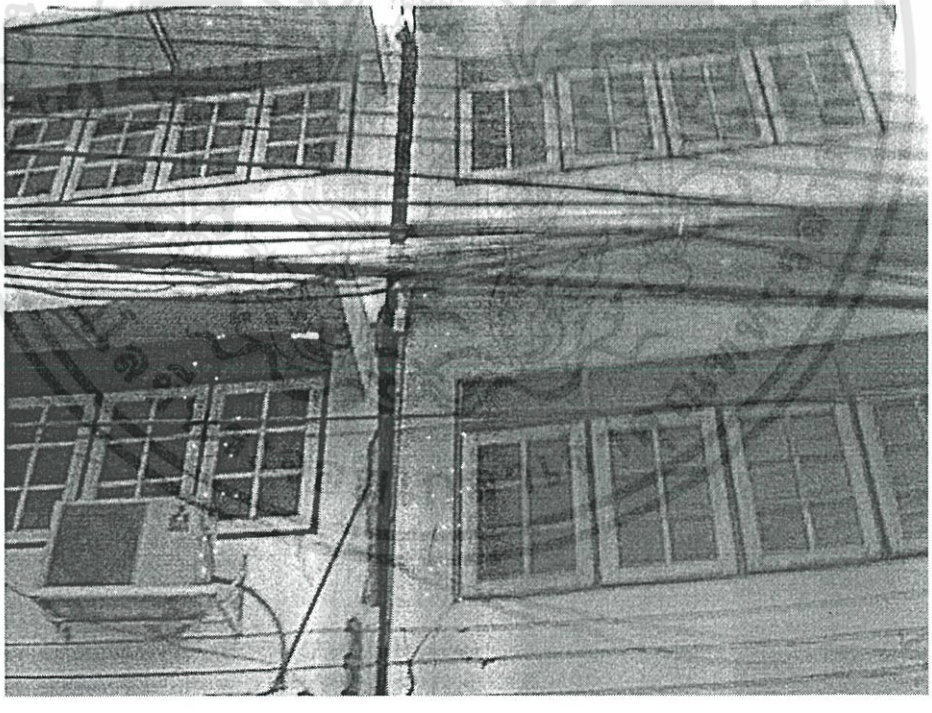
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 การก่อสร้างโครงการขนาดใหญ่ในเขตเมืองอาจสร้างความเสียหายแก่โครงสร้างข้างเคียงได้

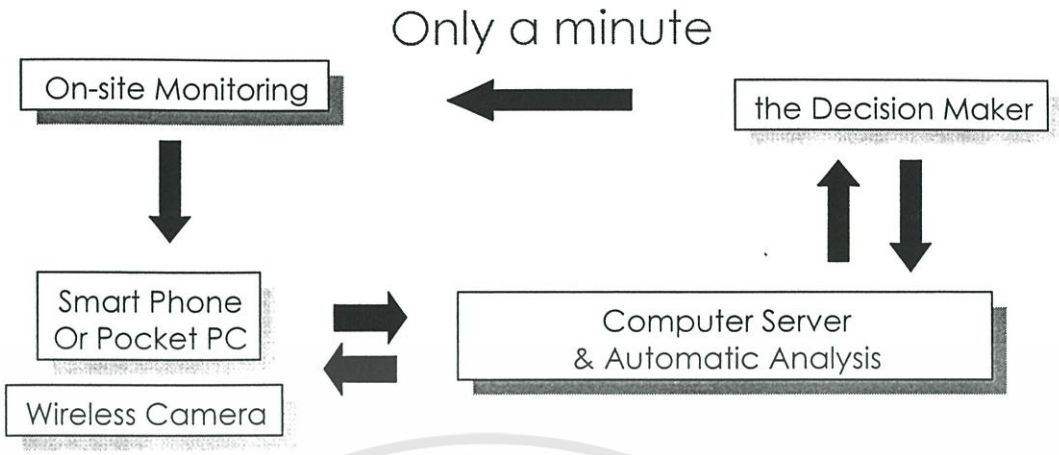
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3 รอยร้าวของผนังอาคารที่เกิดจากการก่อสร้างสถานีรถไฟฟ้าใต้ดินหัวลำโพง



รูปที่ 4 อาคารพาณิชย์แยกตัวที่เกิดจากการก่อสร้างสถานีรถไฟฟ้าใต้ดินสุขุมวิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5 กระบวณการตรวจสอบและรายงานผลกระทบจากการก่อสร้างขนาดใหญ่แบบที่จะถูกพัฒนาขึ้นใหม่

8. วัตถุประสงค์ของโครงการวิจัย

1. เพื่อพัฒนาอุปกรณ์สื่อสารแบบไร้สายและซอฟต์แวร์ เพื่อใช้ในงานตรวจสอบความเสียหายจากการก่อสร้างอุโมงค์และการก่อสร้างโครงการขนาดใหญ่
2. เพื่อพัฒนาซอฟต์แวร์ที่สามารถรับข้อมูลจากอุปกรณ์สื่อสารแบบไร้สายและนำมาวิเคราะห์และรายงานผลกระทบจากการก่อสร้างโครงการขนาดใหญ่ได้ทางอินเตอร์เน็ต
3. คิดค้นระบบสื่อสารแบบทันท่วงที (Real-time) ในการตรวจสอบความเสียหายจากการก่อสร้าง
4. เพื่อพัฒนาระบบฐานข้อมูล เพื่อรองรับข้อมูลขนาดใหญ่ที่ได้รับจากระบบการตรวจสอบความเสียหาย

9. ขั้นตอนการวิจัย

ขั้นตอนของการปฏิบัติจะแบ่งเป็น 2 ส่วนหลัก ดังนี้

1. งานทดสอบและเก็บรวบรวมข้อมูล (Testing and Data Collection)
   - สำรวจและทดสอบพฤติกรรมการเคลื่อนตัวของดิน
   - เก็บบันทึกค่าตัวแปรในการออกแบบระบบการตรวจวัด
   - เก็บบันทึกข้อมูลความเสียหายในอดีต
2. งานออกแบบและพัฒนาซอฟต์แวร์ระบบการตรวจวัดแบบทันท่วงที (Real-time)
   - พัฒนาอุปกรณ์สื่อสารแบบไร้สายและซอฟต์แวร์ เพื่อใช้ในงานตรวจสอบความเสียหายจากการก่อสร้าง

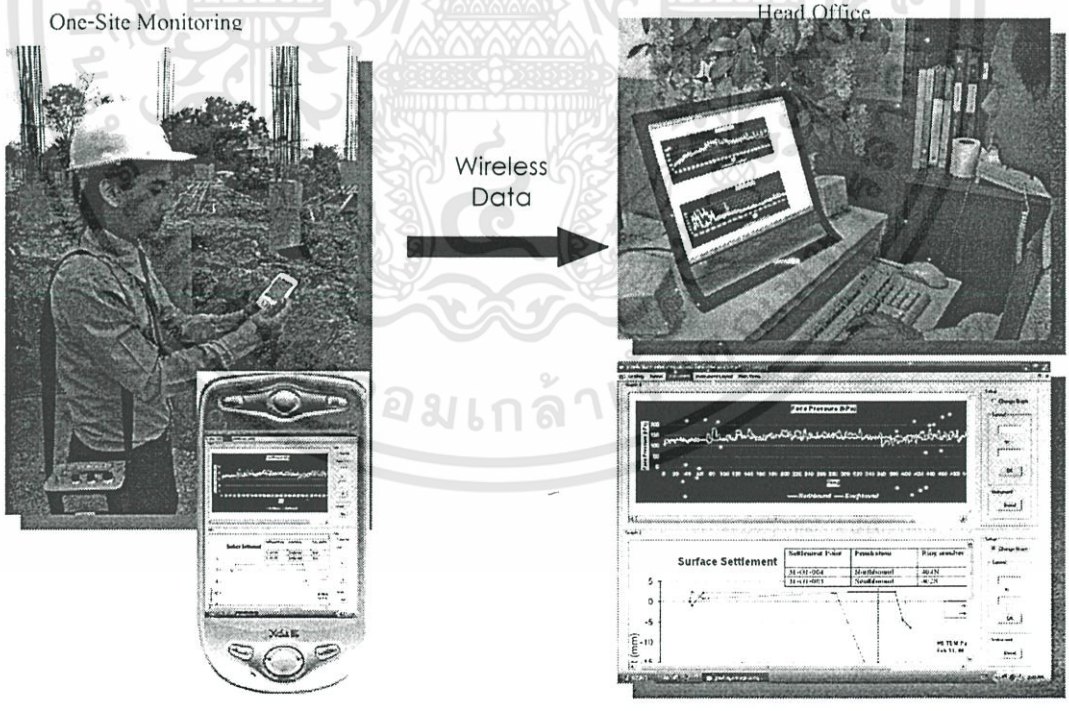
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- พัฒนาซอฟต์แวร์ที่สามารถรับข้อมูลจากอุปกรณ์สื่อสารแบบไร้สายและนำมาวิเคราะห์และรายงานผลกระทบจากการก่อสร้างโครงการขนาดใหญ่ได้ทางอินเตอร์เน็ต
- คิดค้นระบบสื่อสารแบบทันท่วงที (Real-time) ในการตรวจสอบความเสียหายจากการก่อสร้าง
- พัฒนาระบบฐานข้อมูล เพื่อรองรับข้อมูลขนาดใหญ่ที่ได้รับจากระบบการตรวจสอบความเสียหาย

10. คุณสมบัติและรายละเอียดการทำงานของชิ้นงานโครงงานวิจัย

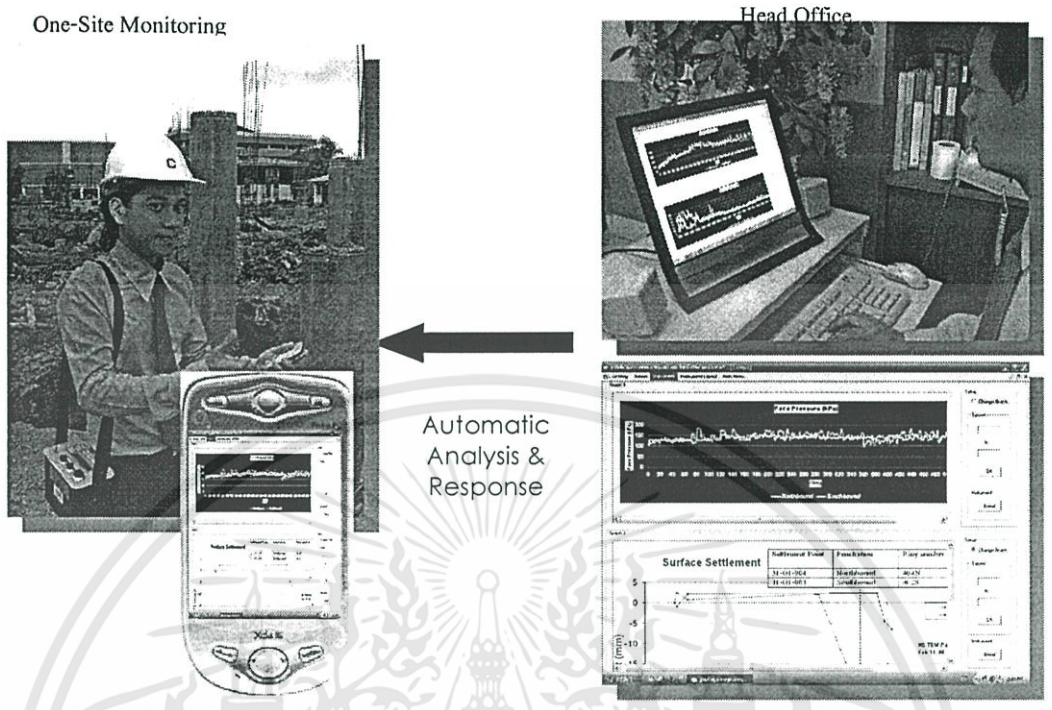
ระบบการตรวจสอบความเสียหายจากการก่อสร้างโครงการขนาดใหญ่ จะสามารถรายงานผลได้อย่างทันท่วงที โดยจะประกอบด้วยการพัมนาที่สำคัญดังนี้

1. ซอฟต์แวร์ที่สามารถส่งถ่ายข้อมูลจากอุปกรณ์ Pocket PC หรือ PDA การตรวจสอบความเสียหายภาคสนามได้
2. ซอฟต์แวร์ที่สามารถรับข้อมูลจากอุปกรณ์สื่อสารแบบไร้สาย เพื่อนำมาวิเคราะห์และรายงานผลได้แบบ Real-time โดยระบบการทำงานจะสามารถรายงานผลได้ทันท่วงที โดยมีกระบวนการดังแสดงในรูปที่ 6-8



รูปที่ 6 วิศวกรกำลังใช้อุปกรณ์สื่อสารไร้สายส่งข้อมูลความเสียหายไปยังระบบส่วนกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7 ระบบส่วนกลางวิเคราะห์ความเสียหายพร้อมทั้งคำแนะนำจากผู้เชี่ยวชาญส่งกลับมายังวิศวกร

รูปที่ 8 วิศวกรพร้อมปฏิบัติการแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ประโยชน์ที่คาดว่าจะได้รับ

1) ข้อมูลปัจจัยในการตรวจวัดความเสียหายจะถูกรวบรวมและสามารถนำไปใช้ประโยชน์ได้
2) เข้าใจพฤติกรรมการเคลื่อนตัวของดินอันเนื่องมาจากการสูญเสียมวลดินเพื่อสามารถกำหนดระบบและวิธีการวัดที่เหมาะสมได้
3) วิศวกรสามารถตรวจสอบความเสียหายโดยใช้อุปกรณ์ไร้สายและเว็บไซต์ ที่สามารถเข้าถึงได้ทุกที่ทุกเวลา ซึ่งจะสามารถช่วยให้ประเมินความเสียหายได้ทันที (Real-time)
4) นำไปสู่การพึ่งพาตนเองด้านเทคโนโลยีและความรู้ความเข้าใจ ส่งผลให้ประหยัดงบประมาณการออกแบบและก่อสร้างได้มหาศาล
5) สามารถใช้เป็นสื่อการเรียนการสอนให้นักศึกษาและวิศวกรเข้าใจถึงพฤติกรรมของดินขณะขุดเจาะอุโมงค์และผลกระทบที่จะเกิดขึ้น
6) เผยแพร่องค์ความรู้ด้านวิทยาการคอมพิวเตอร์ที่นำมาใช้ในการแก้ปัญหาด้านวิศวกรรมโยธา

12. หน่วยงานที่จะนำงานวิจัยไปใช้ประโยชน์

1) คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2) การรถไฟฟ้าขนส่งมวลชนแห่งประเทศไทย
3) การประปานครหลวง
4) หน่วยงานสังกัดกรุงเทพมหานคร
5) กรมทางหลวง
6) การรถไฟแห่งประเทศไทย
7) กระทรวงคมนาคม

13. ทฤษฎีและกรอบแนวความคิด (Theory and Conceptual Framework)

13.1 การพัฒนาระบบเชิงวัตถุ (Object Oriented)

การพัฒนาระบบซอฟต์แวร์หรือระบบสารสนเทศในอดีตนั้น จะใช้วิธีการพัฒนาแบบโครงสร้าง (Structural) ซึ่งมีข้อดีเหนือกว่าการพัฒนาระบบโดยไม่ใช้โมเดล (Model) แต่ในปัจจุบันพบว่า การพัฒนาแบบโครงสร้างนั้นมีปัญหาคือ โมเดลในการวิเคราะห์และออกแบบระบบจะไม่เชื่อมต่อไปถึงรายละเอียดในการพัฒนาระบบ ตัวอย่างเช่น เมื่อผู้พัฒนาระบบวิเคราะห์และออกแบบระบบได้เป็นแผนภาพคอนเท็กซ์ (Context Diagram) หรือแผนภาพ DFD ระดับสูงสุด แผนภาพกระแสข้อมูล (Data Flow) หรือแม้แต่แผนภาพที่แสดงความสัมพันธ์ขององค์ประกอบต่างๆเพื่อสร้างฐานข้อมูลเชิงสัมพันธ์ (Entity Relations) เองแผนภาพเหล่านี้เพียงบอกแต่ว่าระบบจะมีลักษณะการทำงานอย่างไร แต่ไม่ได้บอกวิธีในการพัฒนาระบบหรือรูปแบบในการเขียนโปรแกรมเลย ทำให้ระบบที่พัฒนาขึ้นมีปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากว่าการตรวจสอบความถูกต้องของระบบ (Validation) ในเชิงพฤติกรรม (Behavior) นั้นทำยาก และไม่มีโมเดลใดๆ ที่รอบรับจึงได้มีผู้คิดค้นและพัฒนาวิธีการหรือโมเดลในการพัฒนาระบบเชิงวัตถุขึ้นมา

การพัฒนาระบบเชิงวัตถุ เป็นแนวความคิดในการออกแบบระบบต่างๆโดยมองระบบต่างๆว่าเกิดจากการประกอบกันของวัตถุ (Object) ซึ่งหมายถึงสิ่งต่างๆ ที่อยู่รอบๆ ตัวเรา ออบเจ็กต์หนึ่งอาจจะประกอบด้วยออบเจ็กต์อื่นๆหลายๆออบเจ็กต์ ขนาดที่เล็กที่สุดหรือใหญ่ที่สุดของออบเจ็กต์จะขึ้นอยู่กับขอบเขตของระบบที่ต้องใช้งาน ว่าต้องการรายละเอียดระดับใด ในการกำหนดออบเจ็กต์จะต้องกำหนดคลาสขึ้นมาก่อน คลาสคือรูปแบบทั่วไปของออบเจ็กต์ หรือเป็นพิมพ์เขียวของออบเจ็กต์ และออบเจ็กต์คือ อินแสตนซ์ (Instant) ของคลาส เช่น Mr. John เป็นอินแสตนซ์ ของคลาส Person

ออบเจ็กต์จะมีโครงสร้างที่ประกอบด้วยแอททริบิวท์ และพฤติกรรม เช่นในคลาส Personจะมีแอททริบิวท์ คือ height, weight และ age และมีพฤติกรรมคือ eat, sleep, read, write เป็นต้น ซึ่งแอททริบิวท์และพฤติกรรมดังกล่าวจะมีอะไรบ้างนั้น ขึ้นอยู่กับระบบที่ต้องการออกแบบ เช่น Person ในระบบ Human Resourceก็จะมีรายละเอียดอื่นๆเกี่ยวข้องเพิ่มขึ้น

นอกจากนั้นการวิเคราะห์และออกแบบระบบเชิงวัตถุจะมีโมเดลที่รองรับ ซึ่งโมเดลที่เป็นมาตรฐานอยู่ในขณะนี้มีชื่อว่า UML (Unified Modeling Language) โดยกระบวนการความคิด (Methodology) และโมเดลที่สนับสนุนการเคราะห์และออกแบบเชิงวัตถุที่สามารถเข้าใจได้ง่ายและสามารถใช้ประกอบในขั้นตอนการลงมือเขียนโปรแกรมได้ด้วย

หลักการของการพัฒนาระบบเชิงวัตถุ

Abstraction

หมายถึงการกำหนดออบเจ็กต์ โดยไม่ได้ระบุรายละเอียดของแอททริบิวท์และพฤติกรรมและระบุเป็นส่วนน้อยเท่าที่สามารถทำได้ ซึ่งทำให้การออกแบบมีความยืดหยุ่นมากขึ้น และไม่เกิดการกำหนดออบเจ็กต์เดียวกันมากกว่า 1 ออบเจ็กต์ ผู้ใช้ในแต่ละฟังก์ชันสามารถกำหนดแอททริบิวท์ และพฤติกรรมเพิ่มเติมให้กับออบเจ็กต์ได้ ตามความจำเป็นที่ต้องมีในฟังก์ชันงานนั้นๆ

Inheritance

การสืบทอดคุณสมบัติ คือ เป็นวิธีการในการสร้างคลาสใหม่จากคลาสเดิมที่มีอยู่ ทั้งนี้คลาสที่สร้างขึ้นใหม่จะมีวัตถุประสงค์ในการทำงานที่เฉพาะเจาะจงมากยิ่งขึ้น โดยการที่ออบเจ็กต์มีคุณสมบัติทั้งหมดของคลาสที่ออบเจ็กต์นั้นเป็นอินแสตนซ์อยู่ คลาสที่สืบทอดจะเรียกว่าซับคลาสจะมีคุณสมบัติเหมือนกับซุปเปอร์คลาสทุกประการ โดยไม่ต้องกำหนดซ้ำ

Polymorphism

คุณสมบัติการเปลี่ยนรูป (Polymorphism) หมายถึงการเปลี่ยนรูปร่างของออบเจ็กต์หนึ่งๆซึ่งในเชิงโปรแกรมมิ่งจะเป็นการที่ตัวแปรออบเจ็กต์ของคลาสใดคลาสหนึ่ง สามารถเปลี่ยนรูปแบบไปจากคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดิมได้ โดยคลาสที่มีการสืบทอดแต่โอเปอร์เรชันในซับคลาส มีการทำงานที่แตกต่างกันกับโอเปอร์เรชันเดียวกันในซุปเปอร์คลาส เช่น ถ้าคลาส polygon มีโอเปอร์เรชัน draw และคลาส rectangle สืบทอดจากคลาส polygon ซึ่งจะทำให้มีโอเปอร์เรชั่น draw เช่นเดียวกัน แต่การ draw ของ polygon กับของ rectangle มีการทำงานคนละอย่างกัน ดังนั้นจึงต้องมีการกำหนดโอเปอร์เรชั่น draw ของคลาส rectangle ใหม่

Encapsulation

คุณสมบัติ (Encapsulation) หรือเรียกอีกอย่างนึงคือว่า การซ่อนข้อมูล (Information hiding) คือการจัดกลุ่มองค์ความคิดที่คล้ายคลึงกันเข้าเป็นหน่วยเดียวกันเพื่อทำการอ้างอิงถึงด้วยชื่อเดียวกัน เป็นการซ่อนข้อมูลหรือกระบวนการในการทำโอเปอร์เรชันจากผู้ใช้คลาสในการออกแบบเชิงออบเจ็กต์สามารถทำได้ 3 ระดับ คือ

- Private เป็นการกำหนดให้ข้อมูลหรือโอเปอร์เรชันนั้นเข้าถึงได้จากภายในคลาสเดียวกันเท่านั้น โดยสามารถทำได้ทั้งการอ่านและเขียน
- Protected เป็นการกำหนดข้อมูลหรือโอเปอร์เรชันนั้นเข้าถึงได้จากภายในคลาสเดียวกันและคลาสอื่น โดยถ้าเป็นคลาสเดียวกัน จะสามารถทำได้ทั้งการอ่านและเขียน แต่คลาสอื่นจะทำได้เพียงการอ่านอย่างเดียว
- Public เป็นการกำหนดให้ข้อมูลหรือโอเปอร์เรชันนั้น เข้าถึงได้จากภายในคลาสเดียวกันและคลาสอื่น โดยสามารถทำได้ทั้งอ่านและเขียน

Message Sending

การส่งข้อความทำให้ออบเจ็กต์ในระบบสามารถทำงานร่วมกันได้ โดยออบเจ็กต์หนึ่งจะส่งข้อความให้กับอีกออบเจ็กต์หนึ่งเพื่อให้ออบเจ็กต์นั้นทำโอเปอร์เรชันอย่างใดอย่างหนึ่ง ออบเจ็กต์ที่ได้รับข้อความก็จะทำโอเปอร์เรชั่นตามข้อความที่ได้รับแต่ในบางครั้งพบว่าการกำหนดให้ออบเจ็กต์ทำโอเปอร์เรชันอาจจะทำได้โดยเรียกใช้อินเตอร์เฟสของออบเจ็กต์นั้นโดยตรง เช่น การเรียกใช้ฟังก์ชันที่ประกาศเป็น Public ซึ่งถือเป็น message sending อย่างหนึ่ง

Association

ออบเจ็กต์ที่อยู่ในระบบเดียวกัน จะมีความสัมพันธ์กับออบเจ็กต์อื่นอย่างน้อย 1 ออบเจ็กต์เสมอ ในบางครั้งอาจมีความสัมพันธ์ระหว่างออบเจ็กต์ คือการกำหนดจำนวน (Multiplicity) ซึ่งเป็นการกำหนดจำนวนออบเจ็กต์ในคลาสหนึ่ง ที่มีความสัมพันธ์กับอีกออบเจ็กต์ในอีกคลาสหนึ่ง

Aggregation

เป็นรูปแบบความสัมพันธ์อีกแบบหนึ่ง โดยออบเจ็กต์หนึ่งประกอบขึ้นจากออบเจ็กต์อื่นๆตั้งแต่ 1 ออบเจ็กต์ขึ้นไป เช่นในออบเจ็กต์ Computer A มีออบเจ็กต์ Hard drive 2 ออบเจ็กต์เป็นต้นออบเจ็กต์ Computer เรียกว่า Aggregate Object และออบเจ็กต์ ที่เป็นองค์ประกอบจะเรียกว่า Component Object

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกรูปแบบหนึ่งของ Aggregation ที่เป็นความสัมพันธ์แบบความสัมพันธ์เชิงแข็ง (Strong relationship) คือ Composition โดย Component Object จะต้องมีใน Composite Object เช่น ออบเจ็กต์คอมพิวเตอร์ จะต้องมี ซีพียู (CPU) จึงเป็น Composite แต่ออบเจ็กต์คอมพิวเตอร์อาจจะไม่มีเครื่องพิมพ์ก็ได้ ทำให้ความสัมพันธ์ระหว่าง คอมพิวเตอร์ กับ เครื่องพิมพ์ เป็นความสัมพันธ์แบบ Aggregation แต่ไม่เป็น Composition

กล่าวโดยสรุป ข้อดีของการประยุกต์ใช้แนวคิดเชิงวัตถุในการพัฒนาระบบซอฟต์แวร์มีดังต่อไปนี้

- สนับสนุนการพัฒนาระบบที่ซับซ้อน (Problem Decomposition)

ในการพัฒนาโปรแกรมรูปแบบเดิม นักพัฒนาโปรแกรมจะทำงานกันในระดับฟังก์ชัน กล่าวคือตั้งแต่การเริ่มพัฒนาจนจบ จะเป็นการสร้างปรับปรุงแก้ไขรายละเอียดของฟังก์ชันต่างๆ ภายในตัวโปรแกรม แต่สำหรับแนวคิดเชิงวัตถุ นักพัฒนาจะทำการวิเคราะห์ออกแบบระบบกันในระดับออบเจ็กต์ซึ่งประกอบด้วยทั้งข้อมูลและฟังก์ชันภายในแต่ละออบเจ็กต์ ดังนั้นจึงกล่าวได้ว่าการอาศัยแนวคิดเชิงวัตถุจะช่วยให้นักพัฒนาสามารถสร้างโปรแกรมได้ง่ายขึ้นมาก

- สนับสนุนการนำกลับมาใช้งานอีกครั้ง(Promotion of Reusability)

เนื่องจากแต่ละคลาสหรือออบเจ็กต์ที่กำหนดขึ้นนั้น จะมีความสมบูรณ์อยู่ในตัวเองบนพื้นฐานแนวคิดของแต่ละออบเจ็กต์เอง รวมทั้งยังเป็นอิสระจากสภาพแวดล้อมอื่น ดังนั้นแต่ละคลาสจึงง่ายต่อการนำกลับมาใช้งานปรับปรุงเพิ่มเติม การนำกลับมาใช้งานอาจอยู่ในรูปแบบของการสืบทอดคุณสมบัติระหว่างออบเจ็กต์จากคลาสปุ่มเพื่อไปใส่ไว้ในโปรแกรมต่างๆ ได้ตามต้องการ นั่นคือคลาสปุ่มสามารถถูกนำกลับมาใช้ได้

- ปรับปรุงแก้ไขและบำรุงรักษาง่าย (Easy for Change and Extensibility)

เนื่องจากข้อมูล และฟังก์ชั่นการทำงานที่เกี่ยวข้องกับออบเจ็กต์หนึ่งๆจะถูกรวบรวมอยู่ที่เดียวกันการทำงานภายในแต่ละออบเจ็กต์จะไม่เกี่ยวข้องพึ่งพิงกับโค้ดที่อยู่ภายนอกออบเจ็กต์ดังนั้น นักพัฒนาสามารถทำการแก้ไขปรับปรุงลายละเอียดภายในของแต่ละคลาสได้ โดยไม่กระทบต่อส่วนที่เรียกใช้งานภายนอกแต่อย่างใด นอกจากนี้การขยายงานระบบก็สามารถทำได้โดยง่าย โดยการสร้างออบเจ็กต์หรือคลาสเพิ่มเติมลงไปในตัวโปรแกรมนั่นเอง

ขั้นตอนของกระบวนการพัฒนาซอฟต์แวร์เชิงวัตถุ

วัตถุประสงค์หลักของทุกกระบวนการพัฒนาซอฟต์แวร์ คือ การแปลงของผู้ใช้ให้เป็นระบบที่มีคุณภาพและสามารถใช้งานได้จริง อีกทั้งยังช่วยลดระยะเวลาที่ต้องทำการเขียนโปรแกรมและแก้ไขข้อผิดพลาดให้น้อยลงอีกด้วย ซึ่งในระหว่างการพัฒนาจะมีการใช้เครื่องมือช่วยต่างๆหลายอย่าง ในที่นี้ได้แก่ภาษามาตรฐานยูเอ็มแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการที่จะอธิบายต่อไปนี้ เป็นวิธีการทั่วไปในการพัฒนาซอฟต์แวร์เชิงวัตถุที่ประกอบไปด้วย 5 ขั้นตอน โดยในแต่ละตอนเรียกว่า เฟส (Phase) ซึ่งผลลัพธ์ที่ได้จากการพัฒนาในแต่ละเฟสจะต้องถูกนำไปพัฒนาต่อในเฟสถัดไป และมีเพียงเฟสแรกที่จะถูกดำเนินการในช่วงเริ่มต้นของการพัฒนาครั้งเดียวและในสี่เฟสที่เหลือจะถูกทำซ้ำ (Iteration) เพื่อทำการขยายต่อเติมฟังก์ชันของระบบจนกว่าจะได้ระบบที่สมบูรณ์ นั่นคือในแต่ระรอบของการวนซ้ำสิ่งที่ได้จะเป็นเพียงต้นแบบ (Prototype) อันประกอบไปด้วยฟังก์ชันบางอย่างสอดคล้องกับความต้องการของผู้ใช้ และสิ่งนี้เองที่ฝ่ายพัฒนาควรทำการส่งมอบให้ผู้ใช้พิจารณาเป็นระยะๆในช่วงของการพัฒนา และในความเป็นจริงสิ่งที่เกิดขึ้นตามมาซึ่งมักจะเลี่ยงไม่ได้คือความจำเป็นในการดำเนินการเฟสแรกมากกว่าหนึ่งครั้งอันเนื่องมาจากความเปลี่ยนแปลงความต้องการของผู้ใช้นั่นเอง

การวิเคราะห์ความต้องการของระบบหรือผู้ใช้ (Requirement Analysis)

โครงการที่ถือว่าประสบความสำเร็จจะต้องตอบสนองต่อความต้องการและความคาดหวังของผู้ใช้ได้อย่างครบถ้วน หรือมากกว่า อันเป็นจุดสำคัญของทุกกระบวนการของการพัฒนาซอฟต์แวร์ การวิเคราะห์ความต้องการของผู้ใช้เป็นการค้นหาขอบข่ายของระบบและเป็นการเตรียมข้อมูลของด้านความสามารถของระบบ (System Function) จากมุมมองภายนอกที่จะต้องถูกทำการพัฒนาโดยไม่คำนึงถึงรายละเอียดและกรรมวิธีทางเทคนิคต่างๆ

ในความเป็นจริงการเริ่มต้นเฟสแรกนี้จะต้องทำในรูปของการกำหนดเป็นข้อตกลง ด้วยการเจรจาระหว่างผู้ใช้งานและผู้พัฒนา ซึ่งฝ่ายผู้พัฒนาจะต้องบันทึกความต้องการของผู้ใช้อย่างละเอียดที่สุดเท่าที่จะเป็นไปได้ เรียกขั้นตอนย่อยนี้ว่า User Requirement Elicitation อันหมายถึงการสัมภาษณ์ผู้ใช้งานระบบถึงความต้องการต่างๆ เฟสนี้จำเป็นอย่างมากโดยเฉพาะการพัฒนาระบบด้านธุรกิจ แต่มักไม่จำเป็นสำหรับการสร้างระบบเชิงวิจัยพัฒนาซึ่งมีเป้าหมายที่ชัดเจนอยู่แล้ว สำหรับส่วนที่ไม่สามารถลงลายละเอียดของความต้องการได้ก็ควรกำหนดเป้าหมายให้ชัดเจนในส่วนนั้นๆ เพื่อป้องกันความไม่เข้าใจที่ไม่ตรงกันระหว่างผู้ใช้และผู้พัฒนา

นอกจากการค้นคว้าความสามารถที่ต้องมีอยู่ในตัวระบบแล้ว ผู้พัฒนายังจำเป็นต้องทำการสำรวจความต้องการในแง่อื่นๆ ด้วย (Nonfunctional Requirements) ตัวอย่างเช่น ประสิทธิภาพช่วงเวลาในการตอบสนอง ส่วนอินเทอร์เฟส การใช้งานร่วมกับระบบเดิม เป็นต้น อย่างไรก็ตามหากมองในภาพที่กว้างขึ้น จะพบว่าการดำเนินการพัฒนาระบบจะต้องเกี่ยวข้องกับงบประมาณระยะเวลาในการส่งมอบผลงานแต่ละช่วง การทำข้อสัญญา และการบริหารโครงการ เป็นต้น

การวิเคราะห์ระบบ (Domain Analysis)

การวิเคราะห์ระบบ (Domain Analysis) เป็นการวิเคราะห์โครงสร้าง (Structure) และพฤติกรรม (Behavior) ของระบบที่จะทำการพัฒนาซึ่งจะถูกนำไปกำหนดรายละเอียดเชิงเทคนิคในเฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบจะถูกสร้างจริงเป็นลำดับต่อไปในอิมพลีเมนต์ระบบ กิจกรรมในเฟสนี้จะเป็นการทำความเข้าใจกับระบบกำลังจะพัฒนาในสองด้าน กล่าวคือ

- การวิเคราะห์โครงสร้าง เป็นการทำความเข้าใจกับระบบที่กำลังพัฒนาเพื่อค้นหาคลาสและความสัมพันธ์ระหว่างคลาสในระบบ ซึ่งผลที่ได้เหล่านี้จะถูกนำไปอิมพลีเมนต์สร้างเป็นระบบขึ้นจริงในเฟสถัดๆไป

- การวิเคราะห์พฤติกรรม เนื่องจากออบเจ็กต์ที่เกิดจากคลาสต่างๆในโครงสร้างของระบบจะทำงานร่วมกันเพื่อให้ระบบโดยรวมสามารถตอบสนองต่อการใช้งานของผู้ใช้ได้ในทุกฟังก์ชัน โดยกิจกรรมระหว่างการปฏิบัติงานร่วมกันของแต่ละออบเจ็กต์จำเป็นต้องถูกทำความเข้าใจเพื่อนำไปอิมพลีเมนต์เป็นขั้นตอนการทำงานของแต่ละฟังก์ชันของระบบได้

การออกแบบ (Design)

ซึ่งเป็นกระบวนการคิดค้นวิธีแก้ไขปัญหาหรือพิจารณารายละเอียดเชิงเทคนิค เพื่อเตรียมที่จะอิมพลีเมนต์ระบบขึ้นจริง ซึ่งจะเป็นการนำผลการวิเคราะห์จากเฟสที่สองเพื่อนำมาแก้ไขเพิ่มเติมรายละเอียดเชิงเทคนิค เพื่อสามารถที่จะถูกนำไปสร้างขึ้นเป็นระบบซอฟต์แวร์จริงได้อย่างสมบูรณ์ ตัวอย่างกิจกรรมที่เกิดในเฟสนี้คือ

- การเลือกใช้ภาษาโปรแกรมคอมพิวเตอร์ที่จะใช้ในการพัฒนาระบบ
- การออกแบบฐานข้อมูล โดยเฉพาะในส่วนที่เรียกว่าระดับแนวคิด ซึ่งนิยมใช้ อีอาร์โมเดลเข้ามาช่วย
- การออกแบบสถาปัตยกรรมระบบ จำนวนเครื่องคอมพิวเตอร์
- การออกแบบลักษณะการเชื่อมต่อของเครือข่าย
- การนำกลับมาใช้งานของคลาสไลบรารีหรือซอฟต์แวร์คอมโพเนนต์ที่มีอยู่
- การออกแบบหน้าตาของตัวโปรแกรม (User Interface)
- รวมถึงการจัดการกับข้อผิดผลาดที่อาจเกิดขึ้นกับการใช้งานระบบ เช่น ระบบควรมีการจัดการอย่างไรหากผู้ใช้ป้อนข้อมูลผิดประเภท เป็นต้น

ดังนั้นเฟสนี้จำเป็นต้องอาศัยความรู้ทางเทคนิคและเทคโนโลยีต่างๆ ที่มีอยู่เพื่อสามารถเลือกได้อย่างเหมาะสม

การสร้างโปรแกรมระบบ (Construction, Coding, Implementation)

หลังจากที่ระบบถูกออกแบบไว้อย่างสมบูรณ์พร้อมที่จะถูกนำไปอิมพลีเมนต์จริง โดยโปรแกรมเมอร์ ในเฟสนี้สิ่งที่เกิดขึ้นจากการออกแบบทั้งหมดจะถูกแปลงไปสู่ระบบจริงโดยส่วนใหญ่ กิจกรรมในเฟสนี้จะเป็นการสร้างโปรแกรม หรือการอิมพลีเมนต์ระบบอันเป็นขั้นตอนของ OOP (Object Oriented Programming) ที่คุ้นเคยกันดี ซึ่งจะต้องอาศัยความรู้ในตัวภาษาโปรแกรมที่ใช้ในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สร้างโค้ด ในขั้นตอนนี้จะถูกดำเนินการโดยทีมโปรแกรมเมอร์ภายหลังจากได้รับข้อมูลการออกแบบที่เพียงพอจากฝ่ายวิเคราะห์ออกแบบ

การทดสอบระบบ (Testing)

เป็นการทดสอบความถูกต้องของระบบที่พัฒนาเพื่อค้นหาข้อผิดพลาดทางเทคนิค และการตรวจสอบความสอดคล้องกับความถูกต้องที่ถูกระบุอยู่ในความต้องการของผู้ใช้งาน ทั้งนี้การค้นพบข้อผิดพลาดจะถือว่าจะถือว่าเป็นความสำเร็จในการทำงานในเฟสนี้ มิใช่ความล้มเหลวแต่อย่างใด

นอกจากนี้ยังเป็นการประเมินความสมบูรณ์ของระบบ ว่าจำเป็นต้องทำการวิเคราะห์ออกแบบเพิ่มเติมอีกครั้งหรือไม่ โดยปกติในเฟสนี้จะต้องมีการจัดเตรียมข้อมูลที่จะใช้สำหรับการทดสอบและการผลลัพธ์ที่เรียกว่า Test Cases จะประกอบไปด้วยกิจกรรมในการทดสอบการป้อนข้อมูลและการทดสอบผลลัพธ์รวมถึงความคาดหวังจากการทำงานของระบบว่าน่าจะเป็นที่พึงพอใจแก่ผู้ใช้งานจริงหรือไม่ ผลของการทดสอบจะถูกบันทึกลงรายงานการทดสอบซึ่งรวมถึงการบรรยายรายละเอียดข้อผิดพลาดที่ปรากฏเพื่อทำการแก้ไขต่อไปโดยโปรแกรมเมอร์

ในเฟสนี้สิ่งที่ทำให้นักพัฒนาปวดหัวอยู่เสมอคือข้อผิดพลาด (Error) ซึ่งแบ่งได้เป็นสองด้าน ได้แก่

- Functional เป็นความผิดพลาดในระรับฟังก์ชันการทำงานของระบบ ซึ่งมักเกิดขึ้นบ่อยครั้ง

- Nonfunctional เป็นข้อผิดพลาดที่ไม่เกี่ยวกับระดับฟังก์ชันการทำงานของระบบที่พัฒนาโดยตรง เช่น การที่ประสิทธิภาพของการทำงานของระบบต่ำเกินไป ช่วงเวลาในการตอบสนองนานเกินไป เป็นต้น

โดยปกติความต้องการของผู้ใช้งานมักไม่ถูกค้นพบทั้งหมดในช่วงเริ่มต้นของการณ์สัมภาษณ์ผู้ใช้ จึงเป็นเหตุผลว่าทำไมผู้ใช้งานจึงมักร้องเรียนขอการเปลี่ยนแปลง ซึ่งเป็นปัญหาที่เกิดขึ้นกับนักพัฒนาทั้งหลายโดยตลอดมา ข้อเสนอแนะต่อไปนี้อาจช่วยป้องกันปัญหาดังกล่าวได้บ้างไม่มากก็น้อย

- กำหนดลายละเอียดความต้องการให้ชัดเจนถึงสิ่งที่ผู้ใช้ต้องการ และไม่ต้องการทั้งนี้ห้ามสมมติเอาเอง เพื่อใช้ในการป้องกันหรือชะลอความเสียหายจากผู้ใช้ที่มีความต้องการเปลี่ยนแปลงความต้องการบ่อยๆ

- สิ่งที่ผู้ใช้เกริ่นว่าจะต้องการแต่ไม่ใช่ตอนนี้อาจถูกออกแบบพัฒนาเพื่อไว้โดยปกติ 10% เพื่อลดผลกระทบที่อาจเกิดขึ้นเมื่อเวลาที่ต้องพัฒนาส่วนนั้นจริงๆมาถึง

- กำหนดข้อตกลงให้ชัดเจน ในการส่งมอบงาน หรือระบบตัวอย่างแต่ละช่วงเวลาในการพัฒนา

- การร้องขอให้แก้ไขข้อผิดพลาดในระบบที่เพิ่งส่งมอบ และตรวจรับเรียบร้อย อาจถูกคิดเป็นค่าบริการพิเศษ ทั้งนี้ต้องกำหนดไว้ในสัญญาอย่างชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจำเป็นต้องมองการเปลี่ยนแปลงความต้องการของผู้ใช้งานเป็นเรื่องปกติธรรมดาที่ผู้พัฒนาต้องเผชิญ หากแต่สิ่งที่ควรจะทำคือการหาวิธีรองรับที่จะทำให้ได้รับผลการทบน้อยที่สุดเมื่อความต้องการเปลี่ยนแปลงดังกล่าวเกิดขึ้น

13.2 ภาษายูเอ็มแอล

UML (Unified Modeling Language) หมายถึง ภาษาสัญลักษณ์รูปภาพมาตรฐาน (Standard Modeling Language) ที่ใช้กำหนดลักษณะคลาส การสร้างคลาสและเป็นเอกสารที่บอกถึงรายละเอียดของระบบโครงสร้าง โดยที่ยูเอ็มแอลจะแสดงความโครงสร้างของระบบเชิงวัตถุในรายละเอียดลึกๆได้ดี ในรูปแบบของแผนภาพไดอะแกรม แผนภาพเหล่านี้จะทำให้เกิดความเข้าใจที่ตรงกันระหว่างผู้ออกแบบระบบและโปรแกรมเมอร์ ทำให้การปรับปรุงแก้ไขโปรแกรมทำได้ง่ายขึ้น

ความเป็นมาของภาษายูเอ็มแอล

ยูเอ็มแอลถูกเริ่มครั้งแรกในปี 1994 ที่บริษัท Rational Software โดย Grady Booch และ James Rumbaugh วัตถุประสงค์เบื้องต้นในการร่วมงานกันระหว่างทั้งสองคน เป็นการพัฒนากระบวนการพัฒนาซอฟต์แวร์เชิงวัตถุที่เป็นหนึ่งเดียวกัน (Unified Method) โดยนำเอาวิธีของแต่ละคนคือวิธีของ Booch และวิธี OMT มารวมกันและปรับปรุงใหม่ ต่อมาในปี 1995 Ivar Jacobson ผู้พัฒนากระบวนการ OOSE หรือ Objectory ได้เข้าร่วมกับโครงการดังกล่าวซึ่งในครั้งนี้เป็นการสร้างภาษาโมเดลขึ้นมาใหม่เรียกว่า Unified Modeling Language (UML) และทั้งสามขนานนามตัวเองว่า Three Amigos

เนื่องจากวิธีกรพัฒนาซอฟต์แวร์เชิงวัตถุของทั้งสามมีชื่อเสียงอยู่แล้วในช่วงเวลานั้นดังนั้น ภาษายูเอ็มแอลที่พัฒนาขึ้นใหม่นี้จึงกลายเป็นที่นิยมอย่างแพร่หลายเช่นกัน ในปี 1996 มีหลายบริษัทขอเข่าวมในการพัฒนาภาษายูเอ็มแอล ซึ่งได้แก่ บริษัทดิจิตอลอีควิปเมนต์ บริษัทเอชพี ปริษัทไอบีเอ็ม รวมทั้งบริษัทออราเคิล บริษัทไมโครซอฟต์ และบริษัทอื่นๆอีกหลายบริษัท

ข้อดีของยูเอ็มแอล

ยูเอ็มแอลมีข้อดีหลายประการดังต่อไปนี้

- เป็นภาษารูปภาพมาตรฐานหรือภาษาสากลที่ใช้ในการพัฒนาซอฟต์แวร์เชิงวัตถุ และสามารถใช้ในการแลกเปลี่ยนโมเดลได้อย่างสื่อความหมาย รวมถึงการจัดสร้างเอกสารการวิเคราะห์ออกแบบระบบ
- สามารถนำเสนอและสนับสนุนหลักการเชิงวัตถุได้อย่างครบถ้วนชัดเจน ทำให้นักพัฒนาระบบสามารถทำความเข้าใจกับปัญหาและค้นพบวิธีแก้ไขได้อย่างรวดเร็วและง่ายยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

-ไม่ผูกติดกับภาษาโปรแกรมใดภาษาหนึ่ง กล่าวคือโมเดลที่สร้างขึ้นจากภาษามาตรฐานซึ่งสามารถแปลงไปเป็นระบบจริงที่ถูกสร้างขึ้นด้วยภาษาโปรแกรมเชิงวัตถุ

- เป็นภาษาที่ง่ายต่อการทำความเข้าใจ ผู้ที่ทำการศึกษาหรือนำไปใช้งานไม่จำเป็นต้องมีความรู้อื่นใดนอกจากแนวคิดเชิงวัตถุ

- สามารถถูกแปลงเป็นภาษาที่ใช้ในการสร้างระบบขึ้นได้จริงได้อย่างอัตโนมัติ จึงเป็นการช่วยลดภาระเวลา และค่าใช้จ่ายการพัฒนาระบบได้อย่างมาก

- ยูเอ็มแอลใช้ในการบันทึกความคิดของนักพัฒนาในลักษณะของเอกสาร ที่พร้อมจะถูกนำมาทำความเข้าใจหรือสานต่ออีกครั้งด้วยความรวดเร็ว

องค์ประกอบของยูเอ็มแอล

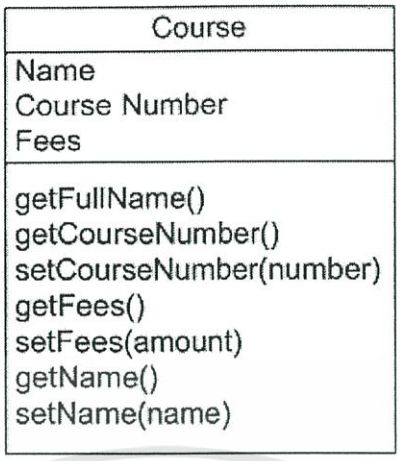
ยูเอ็มแอลเป็นภาษาการสร้างโมเดลในแบบกราฟฟิกซึ่งใช้ในการประกอบเป็น ไดอะแกรม โดยมีกฎในการประกอบกันของอีลิเมนต์ต่างๆ ไดอะแกรมจะแสดงถึงมุมมองต่างๆ (Multiple view) ของระบบ ซึ่งจะเรียกว่าโมเดล โมเดลยูเอ็มแอลจะบ่งบอกถึงรายละเอียดของระบบแต่จะไม่ระบุถึงวิธีการในการพัฒนาระบบ

1. คลาสไดอะแกรม (Class Diagram)

คลาสไดอะแกรมเป็นแผนภาพที่ใช้แสดงคลาส และความสัมพันธ์ในแง่ต่างๆระหว่างคลาสเหล่านั้น ซึ่งความสัมพันธ์ที่กล่าวถึงในคลาสไดอะแกรมนี้ ถือเป็นความสัมพันธ์เชิงสถิตย์ (Static Relationship) หมายถึงความสำคัญที่มีอยู่แล้วในคลาสต่างๆ ซึ่งเรียกว่าความสัมพันธ์เชิงกิจกรรม (Dynamic Relationship) สิ่งที่ปรากฏในคลาสไดอะแกรมนั้น ประกอบไปด้วยกลุ่มของคลาส และกลุ่มของความสัมพันธ์ โดยใช้สัญลักษณ์ที่ใช้ในการแสดงคลาสนั้น แทนด้วยสี่เหลี่ยมที่แบ่งออกเป็น 3 ส่วนโดยส่วนบนสุดจะเป็นชื่อคลาส ส่วนกลางจะแสดงแอททริบิวท์ ของคลาส และส่วนล่างจะแสดงโอเปอร์เรชันของคลาส

ตัวอย่างเช่น คลาสของ Course มีแอททริบิวต์คือ Name, Course Number และ Fees มีโอเปอร์เรชันคือ getFullName, getCourseNumber, setCourseNumber, getFees, setFees, getName, setName สามารถเขียนให้อยู่ในรูปแบบของยูเอ็มแอล คลาสไดอะแกรมได้ดังรูป

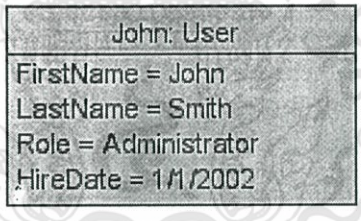
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
84505



รูปที่ 9 คลาสไดอะแกรมของระบบ Course

2. ออบเจ็กต์ไดอะแกรม (Object Diagram)

ออบเจ็กต์ คือ อินสแตนซ์ของคลาส การกำหนดรายละเอียดของออบเจ็กต์ คือการกำหนดค่าให้กับแอททริบิวต์ ออบเจ็กต์ไดอะแกรมของยูเอ็มแอล จะแสดงเป็นรูปสี่เหลี่ยม โดยชื่อจะประกอบด้วยชื่อออบเจ็กต์และชื่อคลาส คั่นระหว่างชื่อด้วยเครื่องหมาย ":" และขีดเส้นใต้ที่ชื่อด้วย ตัวอย่างเช่น ออบเจ็กต์ User ชื่อ John มี FirstName คือ John, LastName คือ Smith, Role คือ Administrator และ HireDate คือ 1/11/2002 สามารถแสดงได้ดังรูป



รูปที่ 10 ออบเจ็กต์ไดอะแกรมของระบบ User

3. ยูสเคสไดอะแกรม (Use Case Diagram)

เป็นแผนภูมิที่ออกแบบมาเพื่ออธิบายถึงกิจกรรมของระบบจากมุมมองของผู้ใช้ซึ่งจะทำให้ได้ระบบที่ตรงกับความต้องการและสามารถนำไปใช้งานได้จริง ยูสเคสไดอะแกรมจะประกอบด้วย แอ็กเตอร์ (Actor), ยูสเคส (Use Case), เส้นแสดงความสัมพันธ์ (Relationship)

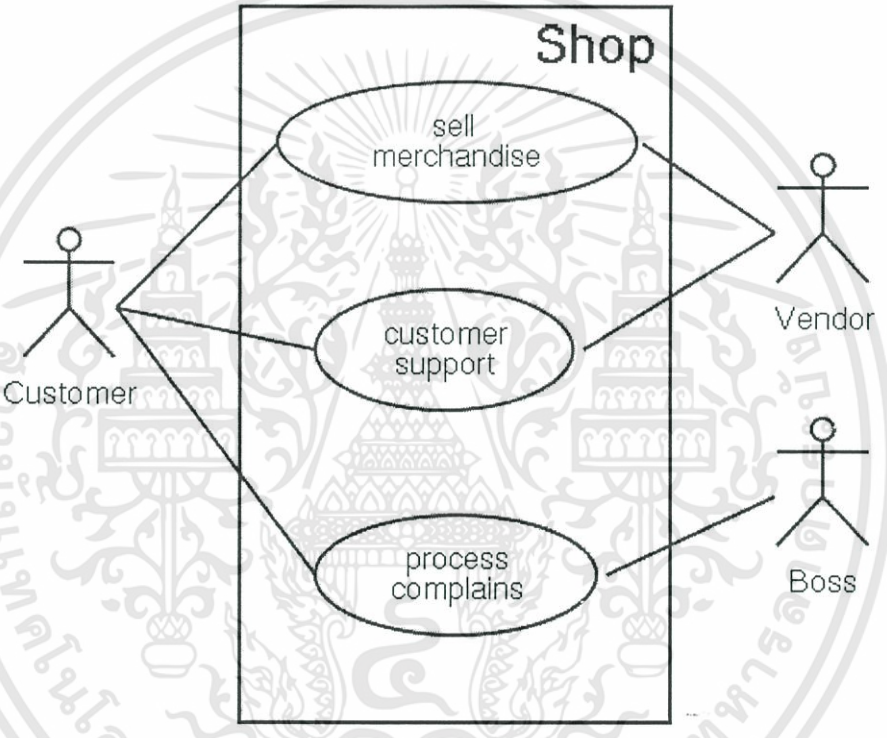
- แอ็กเตอร์ คือผู้ที่กระทำกับระบบ เป็นผู้ที่ใช้ยูสเคสโดยแอ็กเตอร์ที่เริ่มต้นเหตุการณ์จะอยู่ด้านซ้ายของยูสเคส และแอ็กเตอร์ ที่ได้รับผลจากเหตุการณ์นั้นจะอยู่ทางด้านขวาของยูสเคส ชื่อของแอ็กเตอร์จะอยู่ทางด้านล่างของรูปที่ใช้แทนแอ็กเตอร์นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ยูสเคส เป็นกิจกรรมที่ทำโดย แอ็กเตอร์ซึ่งเป็นฟังก์ชันต่างๆ ที่ซอฟต์แวร์ต้องทำได้ทั้งหมด จึงถือว่าซอฟต์แวร์ไม่มีข้อผิดพลาดหรืออีกนัยหนึ่งยูสเคส คือสิ่งที่บอกว่าระบบหรือซอฟต์แวร์ทำอะไรได้บ้าง โดย ยูสเคสจะใช้เครื่องหมายวงรีในการแสดง และชื่อของยูสเคสจะอยู่ภายในวงรีหรืออยู่ใต้วงรี

Use Case Diagram คือความสัมพันธ์ระหว่าง ยูสเคส ซึ่งมี 2 ลักษณะคือแบบ inclusion และแบบ extension แบบ include เป็นการทำขึ้นตอนนี้ทุกครั้ง ส่วนแบบ extend จะเป็นการทำงานนี้เป็นบางครั้ง

- เส้นแสดงความสัมพันธ์ คือ เส้นที่บอกความสัมพันธ์ระหว่างกันทั้งของคลาส หรือแอ็กเตอร์



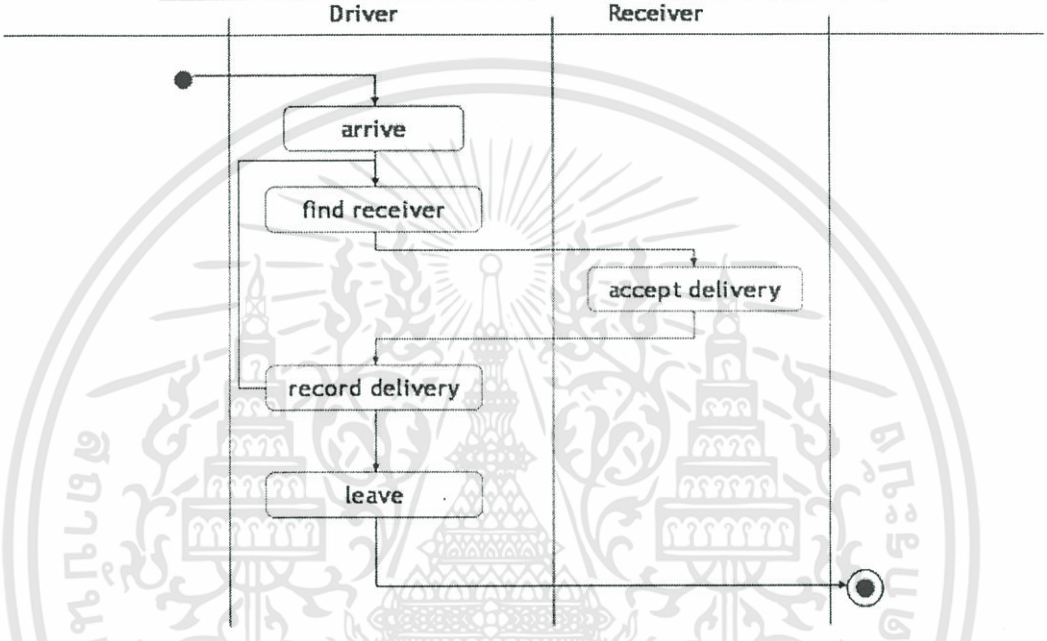
รูปที่ 11 ยูสเคสไดอะแกรม

4. แอ็กทิวิตี้ไดอะแกรม (Activity Diagram)

แอ็กทิวิตี้ไดอะแกรมจะแสดงขั้นตอนการทำงานของยูสเคส โดยจะเน้นที่งานย่อยของวัตถุซึ่งการเจาะจงไปที่งานๆ หนึ่งของวัตถุนั้นจะรู้สึกว่าเหมือนกับ เสตชาร์ตไดอะแกรมที่แสดงสถานะของวัตถุ แอ็กทิวิตี้ไดอะแกรมสามารถเปลี่ยนสถานะได้โดยไม่ต้องมีเหตุการณ์ที่กำหนดไว้ในไดอะแกรมมากระทำก่อน แต่จะเปลี่ยนสถานะเองตามกระบวนการทำงานคล้ายกับโฟลชาร์ตสัญลักษณ์ที่ใช้ในแอ็กทิวิตี้ไดอะแกรม มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญลักษณ์	ความหมาย
●	เป็นจุดเริ่มต้นเข้าสู่แอ็กทิวิตี้ไดอะแกรม
◉	เป็นจุดสำหรับออกจากแอ็กทิวิตี้ไดอะแกรม
◇	เป็นจุดตัดสินใจเงื่อนไขว่าเป็นจริงหรือเท็จ
Action	เป็นกิจกรรมที่ทำ
→	แสดงขั้นตอนในการทำงานถัดไป



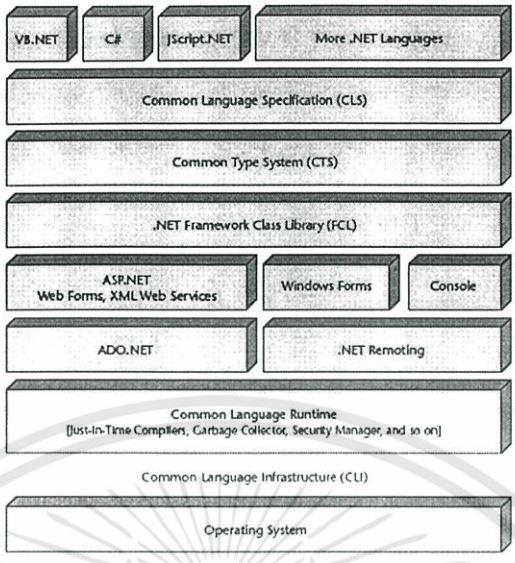
รูปที่ 12 แสดงตัวอย่างแอ็กทิวิตี้ไดอะแกรม มีดังนี้

13.3 เทคโนโลยีด็อทเน็ตเฟรมเวิร์ค (.NET Framework)

สถาปัตยกรรม .NET

โครงสร้างโดยรวมทั้งหมดของการสร้างแอพพลิเคชั่นด็อทเน็ตซึ่งแสดงได้ดังโครงสร้างด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 13 รูปแสดงโครงสร้างการพัฒนาแอพพลิเคชันดอทเน็ต

จากรูป เป็นการแสดงถึงสถาปัตยกรรมของแอพพลิเคชัน และสถาปัตยกรรมที่ใช้โดยมีเลเยอร์ล่างสุด คือ .NET Framework SDK เปรียบเสมือน Runtime Library ที่รันอยู่คอยสนับสนุนการทำงานของแอพพลิเคชันจากนั้นจะเป็นเลเยอร์ของ Common Language Runtime ซึ่งเป็นผลลัพธ์ของการคอมไพล์แอพพลิเคชันดอทเน็ต เลเยอร์ถัดขึ้นมาเป็นเครื่องมือ (Tools) และ เทคนิคต่างๆที่สามารถใช้พัฒนาแอพพลิเคชันได้ทั้งในเรื่องของ Web service, ADO.NET, ASP.NET จนกระทั่งถึงเลเยอร์บนสุดคือ ภาษาที่ใช้ในการพัฒนาแอพพลิเคชั่นด้วยวิชวลสตูดิโอดอทเน็ต

เลเยอร์ คอมมอนแลงเกวจรันไทม์ (Layer Common Language Run Time)

ก่อนที่จะมีการพัฒนาโปรแกรมเป็นเชิงวัตถุ (Object Oriented) นั้น แอพพลิเคชั่นแต่ละตัวเปรียบเสมือนกล่อง ภายในแอพพลิเคชั่นก็จะมีโค้ด (Code) และมีโครงสร้างข้อมูล (Data Structure) ต่างๆของตัวเอง มีฟังก์ชันต่างๆของตัวแอพพลิเคชันนั้น

การที่แอพพลิเคชันต่างๆ จะมีการเรียกใช้การทำงานของกันและกัน หรือมีการส่งผ่านข้อมูลถึงกันและกัน เป็นสิ่งที่ทำได้ยาก ซึ่งอาจต้องมีการกำหนดว่าอะไรขึ้นมาเองระหว่าง 2 แอพพลิเคชันนั้นๆ จนกระทั่งในยุคถัดมา ไมโครซอฟท์ได้คิดค้นเทคโนโลยี COM (Component Object Model) เป็นวิธีที่ทำให้เราเขียนโปรแกรมเป็นแบบเชิงวัตถุ และเรียกใช้การทำงานที่มาจากต่างแอพพลิเคชั่นได้

หากเราจะอธิบายให้ง่ายขึ้น ก็คือเปรียบเหมือนเราเอาแพ็กเกจอันหนึ่งห่อแอพพลิเคชั่นของเราไว้ โดยการพูดคุยหรือสื่อสารกันของแอพพลิเคชันก็พูดคุยผ่านแพคเกจที่เราห่อไว้ จนมาถึงตัววิชวล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

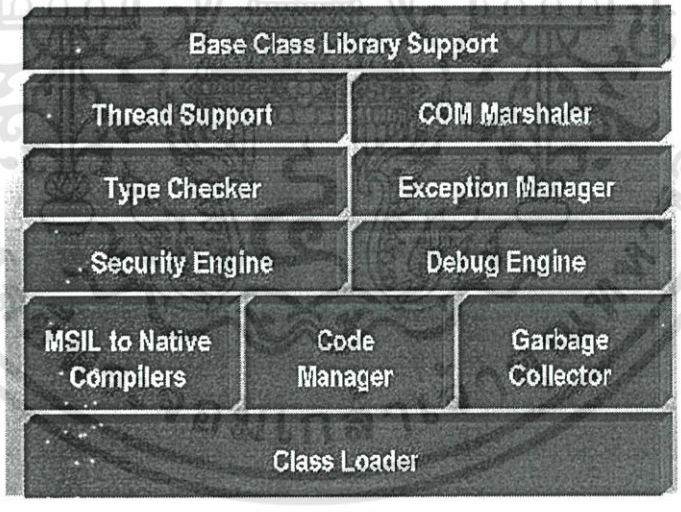
สตูดิโอด็อทเน็ต ที่ได้รับการออกแบบใหม่ จะเห็นว่าจากรูปเดิม กล่องแพคเกจหายไปถ้าเราพัฒนาด้วยรูปแบบเทคโนโลยีด็อทเน็ตนั้น คลาสต่างๆ สามารถติดต่อกันได้โดยตรง

การพัฒนาแอพพลิเคชันด้วย วิชวลสตูดิโอด็อทเน็ตนั้นเมื่อเราคอมไพล์สิ่งที่เราจะได้ จะไม่ใช่โค้ดไบนารี (Binary code) เลยทีเดียว แต่ได้เป็นภาษากลางอันหนึ่งเรียกว่า MSE (Microsoft Intermediate Language) ซึ่งเป็นภาษาในระดับเลเยอร์ต่างๆ โครงสร้างจะเหมือนภาษา Assembly ภายในสิ่งที่เกิดขึ้นก็จะเป็น Microsoft Intermediate Language ตัวนี้ ภายในตัวมันจะประกอบด้วยสองส่วนคือ โค้ดกับตัวแอททริบิวต์หรือพร็อพเพอร์ตี้ต่างๆที่ใช้อธิบายตัวโค้ดนั้นซึ่งเรียกว่า เมต้าดาต้า (Meta Data)

จากนั้น เมื่อโค้ดซึ่งเป็นภาษากลาง (Intermediate Language) ถูกเรียกใช้งานจริงๆ จะมีตัวแปลภาษาตัวหนึ่งมาทำการคอมไพล์โค้ดตัวนั้นให้เป็นโค้ดไบนารี ซึ่งตัวแปลภาษานั้นจะเรียกว่า JIT Compiler (Just In Time Compiler) เพราะว่าจะมีการคอมไพล์เมื่อมีการใช้งาน ฉะนั้นคลาสหรือโค้ดต่างๆ ที่เราพัฒนาขึ้นแล้วจะถูกคอมไพล์มาเป็นภาษากลาง ที่มีโครงสร้างภาษาแบบเดียวกันเพราะฉะนั้นคลาสต่างๆ ในแอพพลิเคชั่นจึงสามารถทำงานได้อย่างกลมกลืนกันและไม่มีข้อติดขัดใดๆ

การทำงานของ คอมมอนแลงเกวจรันไทม์ (Common Language Runtime)

ภายในตัวคอมมอนแลงเกวจรันไทม์ จะมีโมดูล (Module) ย่อยๆซึ่งเป็นสถาปัตยกรรมภายในดังรูป



รูปที่ 14 แสดงสถาปัตยกรรมของคอมมอนแลงเกวจรันไทม์

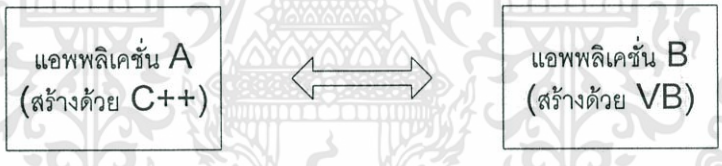
ด้านล่างสุดจะมี Class Loader ซึ่งเอาไว้โหลดโปรแกรมของเราขึ้นมาทำงาน นอกจากนี้ก็จะมีคอมไพล์เลอร์ ซึ่งจะทำการคอมไพล์ภาษากลาง ให้เป็นภาษาไบนารีโดยจะมีโค้ดเมเนเจอร์ (Code Manager) และ การ์เบจคอลเลคเตอร์ (Garbage Collector) คอยจัดการกับหน่วยความจำที่เราจองเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกใช้งาน นอกจากนี้ก็จะมีเรื่องความปลอดภัย (Security) ในการทำงาน รวมทั้ง Debug Engine ในการดักจับรันไทม์เออเรอร์ (Runtime Error) และ แอ็กเซ็บชั่นเมเนเจอร์ การตรวจเช็คชนิดของตัวแปรต่างๆ และด้านบนสุดจะเป็นการใช้งานร่วมกับไลบรารีคลาสต่างๆซึ่งจัดเตรียมมาให้ เพราะฉะนั้นส่วนของเลเยอร์ของคอมมอนแลงเกวจรันไทม์จะสามารถคอมไพล์แอพพลิเคชั่นใดก็ตาม โดยไม่ว่าจะเป็น ASP.NET หรือเขียนแอพพลิเคชันบนวินโดวส์ธรรมดา หรือจะเป็นการเขียนเว็บเซอร์วิสก็ตาม สิ่งที่ได้จากการคอมไพล์จะเป็น คอมมอนแลงเกวจรันไทม์ ตามแผนภาพนี้

นั่นคือ จากการออกแบบเพื่อสนับสนุนการทำงานร่วมกันใน คอมมอนแลงเกวจรันไทม์ จึงขจัดข้อเสียของ COM ออกไปได้ เนื่องจากข้อเสียของ COM คือเป็นเพียงการเอาอะไรบางอย่างมาห่อคลาสเอาไว้เท่านั้น ดังนั้นเวลาเอา COM ไปใช้งานจึงค้นข้างยุ่งยาก รวมทั้งถ้ามีการเปลี่ยนแปลง COM นั้นๆก็จะทำให้เกิดปัญหาเข้ากันไม่ได้ (compatibility) ระหว่างเวอร์ชันเดิมกับเวอร์ชันปัจจุบัน แต่ถ้าเราพัฒนาด้วยวิชวลสตูดิโอดอทเน็ต ข้อเสียของ COM ก็จะถูกกำจัดเสีย

ความจริงคอมมอนแลงเกวจรันไทม์ เป็นหลักการวิวัฒนาการมาจาก COM อีกทีหนึ่ง เป็นเชิงวัตถุที่แกนของภาษาเลย โดยวิชวลสตูดิโอดอทเน็ตนั้น ถูกออกแบบเพื่อสนับสนุนเชิงวัตถุโดยเฉพาะ คลาสต่างๆที่อยู่ในแต่ละแอพพลิเคชั่นสามารถติดต่อกันได้โดยตรงใน แอพพลิเคชัน A เราอาจเขียนคลาสด้วย ภาษา C++ และแอพพลิเคชั่น A อาจจะติดต่อกับ แอพพลิเคชั่น B ซึ่งเขียนด้วยภาษา Visual Basic ได้คือการสืบทอดข้ามภาษานั้นสามารถทำได้



รูปที่ 15 แสดงความสามารถในการติดต่อข้ามแอพพลิเคชั่น เมื่อเขียนด้วยภาษาที่ต่างกัน

ในวิชวลสตูดิโอดอทเน็ตจะคอมไพล์เป็นภาษาเดียวกันคือ ภาษากลาง ตามรูป นอกจากนี้ยังสามารถทำงานด้วยกันกับ COM แบบเดิมที่เราเคยเขียนมาแล้วได้ด้วยซึ่งใช้วิชวลสตูดิเดิมโค้ดเหล่านี้ก็ไม่จำเป็นต้องโยนทิ้ง ในวิชวลสตูดิโอดอทเน็ต เราสามารถเรียกใช้งาน คอมโพเนนต์ที่เขียนด้วยวิชวลสตูดิโอด็อทเน็ตได้เช่นกัน คือเป็นการเข้ากันได้ทั้งสองทาง (Backward Forward Compatibility) นี่คือข้อดีมากๆ ของ ด็อทเน็ต ทำให้เราไม่ต้องพัฒนาโค้ดใหม่

การจัดการเกี่ยวกับหน่วยความจำเมื่อทำการประมวลผล

เนื่องจากการทำงานทั้งหมดของตัวด็อทเน็ตจะมีการดูแลเรื่องคอมมอนแลงเกวจรันไทม์ ปัญหาหนึ่งที่มักจะพบเมื่อเราพัฒนาแอพพลิเคชั่น คือเรื่องของการ์เบจคอลเลคชั่นในการจักการกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำ โดยเฉพาะบางภาษาอย่างเช่น ภาษาซีพลัสพลัสที่ต้องมีการเรียกใช้งานพอยเตอร์ (Pointer) ค่อนข้างมาก ในการใช้งานพอยเตอร์นั้น ถ้าเราทำผิดไปนิดเดียวก็อาจทำให้แอพพลิเคชั่นของเราหยุดการทำงานในระหว่างรันหรือประมวลผลก็ได้ทำให้ถ้าแอพพลิเคชั่นของเรารันไปเรื่อยๆ หน่วยความจำเหล่านี้ก็จะขยายไปเรื่อยๆ และสักพักหนึ่งก็จะเกิดการหยุดไป

การแก้ไขข้อผิดพลาด (Debug) ตรงนี้ทำได้ยากมาก เพราะกว่าจะรู้ว่าเกิดความผิดพลาดขึ้นเราก็เสียเวลาในการรันไปแล้ว ซึ่งการจัดการกับหน่วยความจำนี้ ถ้าเราพัฒนาด้วยแพลตฟอร์มของด็อทเน็ตจะมี เครื่องมือหนึ่งเรียกว่า การ์เบจคอลเลคเตอร์ เป็นตัวคอยจัดการเกี่ยวกับหน่วยความจำให้เรา ความจริงตัวการ์เบจคอลเลคเตอร์ จะเป็นตัวทำการกำหนดค่าหน่วยความจำให้ว่างให้เราเองในส่วนที่เราไม่ใช้งานโดยอัตโนมัติ นี่คือข้อดีอันหนึ่งในการพัฒนาด้วย แพลตฟอร์มของด็อทเน็ต

ระบบตรวจจับความผิดพลาด

นอกจากนี้ในเรื่องของเอ็กเซ็บชั่นเมเนเจอร์ (Exception Manager) หรือการดักรันไทม์เออเรอร์ (Runtime Error) ในการสร้างแอพพลิเคชั่นที่มีการรันได้อย่างดีนั้น การดักรันไทม์เออเรอร์ที่จะเกิดขึ้นในระหว่างการทำงาน ก็เป็นปัจจัยหนึ่งที่เราต้องคำนึงถึง ซึ่งในตัวด็อทเน็ตก็จะมีโครงสร้างภาษาในการดักรันไทม์เออเรอร์ ที่มีลักษณะเป็นโครงสร้างทำให้หาได้ง่ายขึ้น เราเรียกว่าสตักเจอร์เอ็กซ์เซ็ปชั่นแฮนดิง (Structure Exception Handing)

รูปแบบการคอมไพล์แอพพลิเคชั่น

นอกจากนี้การคอมไพล์แอพพลิเคชั่นดอ็ทเน็ตมีให้เลือกหลายแบบ บางคนอาจจะคอมไพล์ให้เป็นแบบ โค้ดไบนารีเลย ไม่ต้องคอมไพล์ให้เป็นอินเทอร์มิเดียดท์แลงเกวจก็สามารถ จะพัฒนาได้โดยใช้วิชวลซีพลัสพลัสดอทเน็ท (Visual C++.NET) เรียกว่าเมเนจซีพลัสพลัส (Manage C++) หรือเราเลือกใช้เครื่องมือภาษาอื่นๆที่มีความง่ายในการพัฒนายิ่งกว่า คือ วิชวลเบสิค หรือ ซีชาร์ป ก็สามารถคอมไพล์เป็นอินเทอร์มีเดียดท์แลงเกวจได้

ในการคอมไพล์จะไม่มีตัวอินเทอร์รันไทม์ไลบรารี (Interpreter Runtime Library) ต่างๆ เช่น วิชวลเบสิค ที่ต้องมีรันไทม์ไลบรารีของตัวเองเวลาพัฒนาด้วย วิชวลสตูดิโออันนี้ก็จะไม่จำเป็นต้องทีอีกต่อไป เราใช้ตัวรันไทม์อันเดียวกัน ก็คือตัว ดอทเน็ทเฟรมเวิร์ค เป็นตัวรันไทม์ไลบรารี ที่กล่าวมาข้างต้น

รูปแบบการทำงานร่วมกับภาษาอื่นๆ

การพัฒนาแอพพลิเคชันในดอทเน็ต รวมทั้งเทคนิคต่างๆ ที่กล่าวมาแล้วนั้น ไม่จำเป็นจะใช้เฉพาะภาษาตระกูลในวิชวลสตูดิโอไม่ว่าจะเป็นวิชวลเบสิคดอทเน็ต (VB.NET), ซีชาร์ป (C#) หรือซีพลัสพลัส (C++) ถ้าภาษาใดสามารถคอมไพล์มาให้เป็นคอมมอนแลงเกวจรันไทม์ได้ภาษานั้นก็จะ

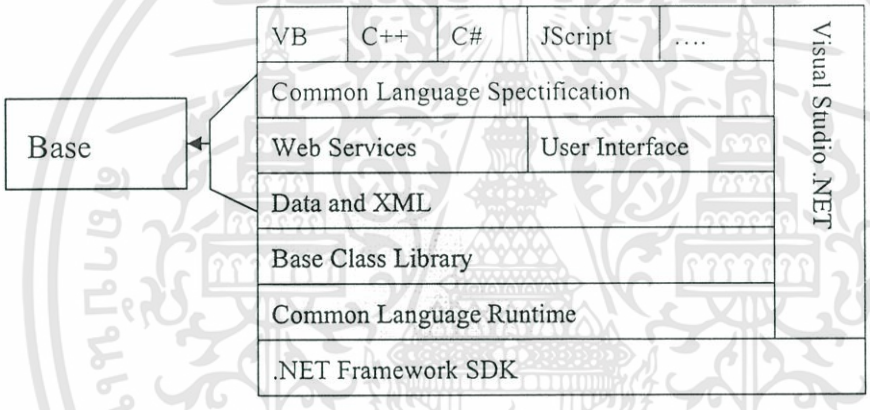
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถใช้คุณลักษณะทั้งหมดของดอทเน็ตได้ซึ่งตอนนี้มีภาษาอื่นๆที่สามารถทำงานร่วมกับแอพพลิเคชั่นด็อทเน็ต มากกว่า 20 ภาษาที่พัฒนาตัวคอมไพล์ขึ้นมาเพื่อคอมไพล์ภาษาต่างๆ ให้อยู่ในรูปของแพลตฟอร์ม ด็อทเนต ไม่ว่าจะเป็นภาษาโคบอล (COBOL), ปาสคาล(PASCAL) และอื่นๆ

ในตัววิชวลสตูดิโอด็อทเน็ต ก้มีเครื่องมือระดับสูง(High Level Tools) ที่จะทำให้เราพัฒนาแอพพลิเคชั่นได้สะดวกและง่ายมากขึ้น นี่คือเรื่องของคอมมอนแลงเกวจรันไทม์ ซึ่งถือได้ว่าเป็นเลเยอร์ล่างสุดของการคอมไพล์แอพพลิเคชั่นที่เขียนด้วยวิชวลสตูดิโอด็อทเน็ต

เลเยอร์เบสคลาสไลบรารีแลงเกวจ (Base Class Library Language)

เลเยอร์ถัดมาเป็นของโครงสร้างการพัฒนาแอพพลิเคชั่นด็อทเน็ตต่อจากคอมมอรันไทม์แลงเกวจ ก็คือเบสคลาสไลบรารี



รูปที่ 16 เบสคลาสไลบรารีของสถาปัตยกรรมด็อทเน็ต

เทคนิคต่างๆที่วิชวลสตูดิโอด็อทเน็ตจัดเตรียมให้เราในการใช้งานนั้นเบสคลาสไลบรารีเปรียบเสมือนการรวบรวมเอาฟังก์ชั่นของเอพีไอ (API: Application Programming Interface) ทั้งสมมติว่าตอนที่เราพัฒนาแอพพลิเคชั่นด้วยวิชวลสตูดิโอ เวลาเราเรียกใช้งานฟังก์ชั่นบางอย่างในระดับสูงหรือต้องการทำงานแบบลึกๆกับระบบเรามักจะเรียกใช้เอพีไอ ภาษาในวิชวลสตูดิโอมีความสามารถในการเรียกฟังก์ชั่นมีความสามารถในการเรียกฟังก์ชั่นเอพีไอได้ต่างกัน

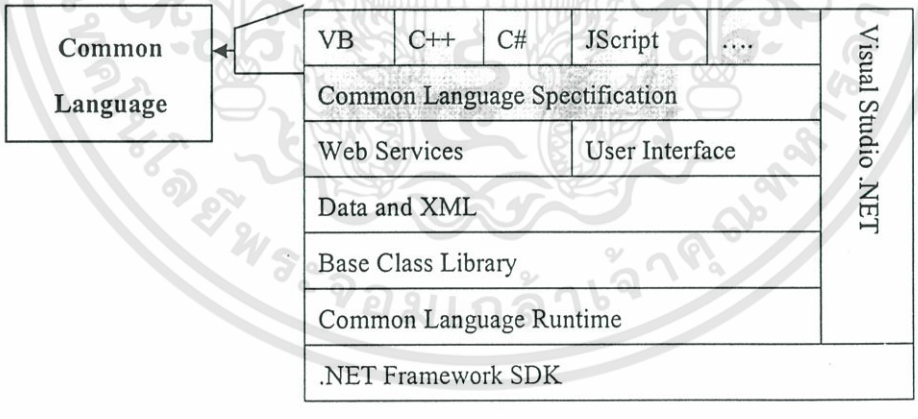
ตัวเบสคลาสไลบรารี ก็คือการที่เรารวบรวมฟังก์ชันเอพีไอซึ่งกระจัดกระจายอยู่เวลาจะเรียกใช้เราต้องไปค้นหาในเมนูช่วยเหลือ (Help) นั่นคือเบสคลาสไลบรารี พยามยามที่จะรวบรวมเอพีไอและฟังก์ชั่นทั้งหมดเกี่ยวกับระบบเข้ามาไว้ในลักษณะของเชิงวัตถุทำเป็นคลาสอันหนึ่งซึ่งเป็นมาตรฐานเป็นคลาสที่สร้างมาในตัวระบบเรียบร้อย ซึ่งคลาสทั้งหมดจะอยู่ภายใต้คลาสหลักอันหนึ่งที่เรียกว่า ซิสเต็ม (System) ทุกอย่างที่พัฒนาด้วยวิชวลสตูดิโอด็อทเนท จะเป็นเชิงวตถุทั้งหมด โดยมีคลาสที่ใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่สุดเรียกว่าคลาสซิสเต็ม ซึ่งภายในคลาสซิสเต็ม จะมีคลาสย่อยๆมากมาย ซึ่งแต่ละอันจะสนับสนุนฟังก์ชันเอพีไอหรือสนับสนุนการทำงานที่เราต้องการได้ไม่ว่าจะเป็นเรื่องของการทำกราฟิก การทำเกี่ยวกับโครงสร้างข้อมูลการทำเกี่ยวกับเรื่องเครือข่าย ฟังก์ชันเอพีไอเหล่านี้จะถูกจัดกลุ่มให้เป็นเชิงวัตถุ อยู่ภายใต้คลาสซิสเต็ม การเรียกช้านคลาสซิสเต็ม ถ้าเป็นวิชวลเบสิกกับซีชาร์ปก็ใช้งานได้ทั้ง 2 อย่าง

เลเยอร์คอมมอนแลงเกวจสเป็คซิฟิเคชัน (Common Language Specification Layer)

เลเยอร์สุดท้ายในสถาปัตยกรรมด็อทเน็ตที่เราจะพูดถึงคือเลเยอร์คอมมอนแลงเกวจสเป็คซิฟิเคชั่น เรื่องมาตรฐานบนพื้นฐานด็อทเน็ต ซึ่งคอมไพล์เลอร์จะต้องทำงานตามมาตรฐานดังกล่าวเพื่อให้สามารถทำงานร่วมกับภาษาบนพื้นฐานด็อทเน็ท และภาษาอื่นๆได้ ไมโครซอฟท์ได้ทำการปรับภาษาต่างๆ เช่น ซีชาร์ป, วิชวลเบสิค ให้เข้ากับมาตรฐานด็อทเน็ต นอกจากนั้น ผู้ผลิตรายอื่นสามารถพัฒนาตามข้อกำหนดนี้เพื่อสามารถทำงานบนพื้นฐานด็อทเน็ตได้ เลเยอร์ที่สามของสถาปัตยกรรมด็อทเน็ต ก็คือเครื่องมือในการสร้างแอพพลเคชั่น แต่สิ่งที่อยู่เหนือกว่าทุกๆอย่างก็คือภาษาที่เราใช้งานด็อทเน็ต นั้นมีข้อดีคือ ต้องสนับสนุนมาตรฐานเดียวกัน เรียกว่า คอมมอนแลงเกวจสเป็คซิฟิเคชัน ซึ่งไมโครซอฟต์ได้จดทะเบียนมาตรฐานนี้เข้ากับองค์กร ECMA แล้ว ซึ่งเป็นองค์กรดูแลเรื่องโครงสร้างภาษาต่างๆ และก็เป็นแบบเปิดด้วยเพราะฉะนั้นเจ้าของภาษาอื่นก็สามารถสร้างตัวแปลงภาษา หรือคอมไพล์เลอร์ เพื่อคอมไพล์ภาษาของเขาให้มาเป็นคอมมอนแลงเกวจสเป็คซิฟิเคชั่น อันนี้



รูปที่ 17 คอมมอนแลงเกวจสเป็คซิฟิเคชั่นของสถาปัตยกรรมด็อทเน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

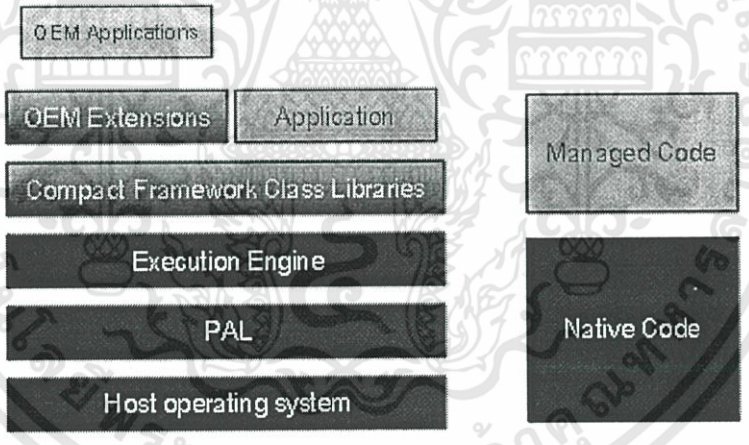
13.4 สถาปัตยกรรมของด็อทเน็ตคอมแพคเฟรมเวิร์ค (.NET Compact Framework)

ส่วนหนึ่งของดอทเน็ตคอมแพคเฟรมเวิร์ค ถูกสร้างขึ้นโดย เป็นส่วนย่อยของ เดสท็อป แฟรมเวิร์ค (Desktop Framework) โดยมีเป้าหมายหลักคือ เข้ากันได้กับทุกแพล็ตฟอร์ม (Platform)

.NET Framework คือ รูปแบบโครงสร้างพื้นฐานของคอมพิวเตอร์แบบใหม่ ออกแบบมาเพื่อรองรับการพัฒนาแอพพลิเคชั่นบนระบบปฏิบัติการวินโดวส์ ในรูปแบบขั้นสูงของสิ่งแวดล้อมบนอินเตอร์เน็ท โดยมี Common Language Runtime (CLR) คือ หัวใจของ .NET Framework โดยมีหน้าที่สำหรับcomplied โปรแกรม แล้วยังจัดการหน่อยความจำผ่าน garbage collection โปรแกรมใดๆที่ทำงานใช้ CLR จะเรียกว่า manage code ส่วนที่ไม่ใช่จะเรียกว่า unmanaged code

Namespaces คือ กลุ่มเทคนิคที่เกี่ยวข้องกันรวมกันอยู่ในรูปของ Namespace เพื่อใช้ค้นหาหรืออ้างอิงได้ง่ายขึ้น กล่าวคือทุกคลาสที่เขียนใน ภาษา C# จะเป็นของ Namespace โดยทั่วไปคลาสจะเป็นของ Namespace ที่มีชื่อเดียวกับโปรเจ็คนั้นๆ (ทั้งนี้รวมถึงชื่อที่ตั้งค่าเริ่มต้นไว้) อย่างไรก็ตามยังสามารถเปลี่ยนชื่อโปรเจ็คได้ โดยที่ Namespace จะใช้สำหรับคลาสทั้งหมดโดยใช้โปรเจ็ค properties

Namespaces จะจัดการเส้นทางการสืบทอดคุณสมบัติ ของชื่อคลาสเพื่อหลีกเลี่ยง 'Name pollution'



รูปที่ 18 แสดงสถาปัตยกรรมของด็อทเน็ตคอมแพ็คแฟรมเวิร์ค

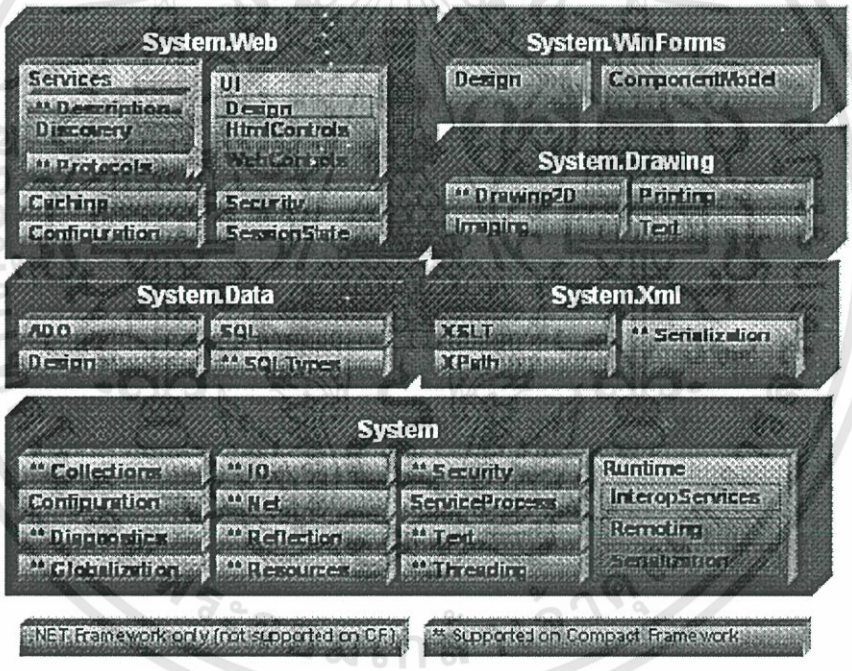
ขนาดของคอมแพคเฟรมเวิร์ค จะถูกติดตั้งบนอุปกรณ์ต่างๆ แต่ละอุปกรณ์ โดยปกติจะมีขนาด 1.7 MB - 2.6 MB และสามารถติดตั้งใน RAM, ROM หรือ FlashROM โดยในอนาคต คาดว่า ถ้าเราเริ่มด้วย RAM เราจะสามารถให้ใช้งานได้ทุกอุปกรณ์ทันที OEMs จะทำให้ FlashROM พัฒนาขึ้น และสามารถรวมอยู่ในคอมแพคแฟรมเวิร์ค สำหรับอุปกรณ์พวก Pocket PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Host Operating System เป็นชั้นล่างสุดของคอมแพคแฟรมเวิร์ค โดยส่วนที่เป็นโค้ด ที่เขียนสำหรับคอมแพคแฟรมเวิร์ค จะถูกประมวลผลบน Host Operating System เช่น Windows CE โดยคอมเพ็คแฟรมเวิร์คจะถูกรันบน Windows CE 3.0 และ Windows CE.NET 4.1

- PAL คือส่วนประกอบที่สำคัญ ที่จะทำให้พอร์ทของแพล็ตฟอร์มสามารถทำงานได้ ซึ่งจะบรรจุระบบย่อยๆ ที่เป็นฟังก์ชั่นการทำงานของระบบปฏิบัติการ (OS) และระบบฮาร์ดแวร์ (Hardware) โดยที่ฮาร์ดแวร์จะอยู่ในส่วนของ API, NSL, Execution Engine ตัวอย่างเช่น PAL จะประกอบด้วยอินเตอร์เฟส (Interface) สำหรับอุปกรณ์ต่างๆ, ระบบจัดการหน่วยความจำและพอร์ทติดต่ออุปกรณ์ภายนอก (I/O Port)

- Execution Engine จะมีกลุ่มของบริการที่สำคัญๆ ซึ่งแต่ละส่วนจะทำได้เหมือน CLR โดยให้บริการที่เหมือนกันสำหรับเดสท็อปและเซิร์ฟเวอร์แอพพลิเคชั่น ซึ่งต้องมีการประมวลผลบนด็อทเน็ทแอพพลิเคชั่น แต่อาจจะใช้ทรัพยากรมาก แต่ในความเป็นจริงแล้ว ตัวหน่วยความจำมันจะมีข้อจำกัด ซึ่งในตัวของ Execution Engine จะถูกสร้างมาโดยคำนึงถึงข้อจำกัดนี้



รูปที่ 19 แสดงคลาสไลบรารีของด็อทเน็ตคอมแพคแฟรมเวิร์ค

- Class Library เพื่อที่จะสร้างโรบัสโปรแกรมคอมแพคแฟรมเวิร์ค ก็มีกลุ่มของ Class Library เป็นภาษาในการสร้างไฮราคีคอลเนมสเปซ (Hierarchical namespace) ซึ่งก็จะเหมือนกับที่เราเจอในเดสท็อปเฟรมเวิร์ค แต่ก็จะมีส่วนที่ต่างกันอยู่ 4 อย่าง คือ

1. อาจจะมองได้ว่า คอมแพคเฟรมเวิร์ค เป็นซับเซตของ Desktop Framework

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนที่สองที่ทำให้ คอมแพคเฟรมเวิร์ค ต่างจาก Desktop Class Library โดยที่มีการพัฒนาต่างจากเดิม

3. คอมแพคเฟรมเวิร์คจะสามารถรองรับ Name space ขึ้นอีกสองชนิดโดยแต่ละชนิดมีวิธีการที่จะแอสเซมบลีของตัวมันเอง ซึ่งจะทำให้มีการประมวลผลที่ดีขึ้นคล้ายกับว่าทำให้อุปกรณ์ฉลาดขึ้น

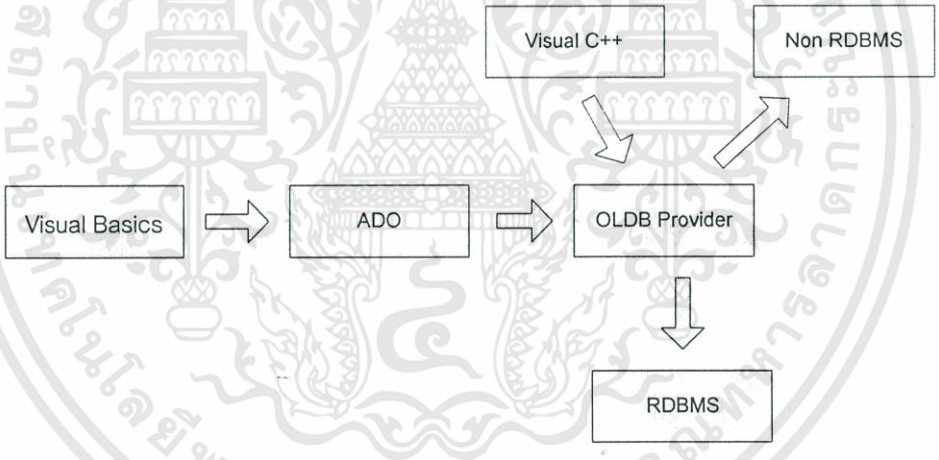
4. คอมแพคเฟรมเวิร์คจะสามารถรองรับการทำงานได้น้อยกว่าของ

เดสท็อปเฟรมเวิร์ค ซึ่งส่วนที่ คอมแพคแฟรมเวิร์คสามารถรองรับได้มักจะถูกอ้างอิงในนามของ Common Type System (CTS) เพราะว่าเป็นส่วนสำคัญที่สามารถรองรับได้

13.5 เทคโนโลยีเอดีโอด็อทเน็ต (ADO.NET)

เทคโนโลยีเอดีโอด็อทเน็ต (ActiveX Data Object Dot Net) เป็น เอพีไอ ที่ใช้ในการติดต่อกับฐานข้อมูลต่างๆ ซึ่งตัวเอดีโอด็อทเน็ต นี้ถูกสร้างมาเพื่อทดแทนอินเทอร์เฟสแบบเก่าที่มีความจำกัดในหลายๆอย่างคือ DAO (Data Access Objects) และ RDO (Remote Data Objects: R)

ตัวเอดีโอด็อทเน็ตเป็นเทคโนโลยีใหม่อันหนึ่งที่ช่วยในเรื่องการติดต่อกับแหล่งข้อมูลต่างๆ และเอดีโอด็อทเน็ต ยังมีโครงสร้างเหมือนเอดีโอ เพียงแต่การออกแบบนั้นมีความแตกต่างกันบ้าง



รูปที่ 20 สถาปัตยกรรมการเชื่อมต่อข้อมูลผ่านเอดีโอ

การเก็บข้อมูลในเอดีโอ เมื่อมีการเลือกข้อมูลนั้น ผลลัพธ์ที่ออกมาจะเก็บข้อมูลในโครงสร้างของ COM (Common Object Model) แต่ตัวเอดีโอด็อทเน็ต เมื่อเราเลือกข้อมูลออกมาแถวต่างๆที่เลือกออกมาจะเก็บไว้ในรูปข้อมูลของ เอ็กซ์เอ็มแอล (XML) เพราะฉะนั้น เอดีโอด็อทเน็ตจึงเหมาะกับการทำงานที่มี เป็นแอพพลิเคชั่นบนอินเตอร์เน็ต

ถ้าเป็นเอดีโอปกติ จะเหมาะกับการทำงานที่เป็น ไคลเอนต์/เซิร์ฟเวอร์ ที่อยู่ในเครือข่ายแลนหรือเครือข่ายอินทราเน็ต (Internet) ในองค์กรของเรามากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อฐานข้อมูลเครือข่ายภายในองค์กรนั้น เราสามารถกำหนดการเชื่อมต่อได้คือเมื่อปิดแอพพลิเคชั่น เราก็ค่อยมาปิดการเชื่อมต่อข้อมูล ซึ่งเรียกว่าเป็น Connection Oriented แต่เมื่อเอาไปใช้กับการทำงานที่เป็นแอพพลิเคชั่นบนอินเตอร์ ซึ่งเป็นระบบที่จำนวนผู้ใช้ไม่แน่นอน และสูงกว่าระบบเครือข่ายในบริษัทแน่นอน เพราะฉะนั้นการที่เรามาเชื่อมต่อกับฐานข้อมูลทิ้งไว้จึงไม่เหมาะสมกับการทำงานในเครือข่ายอินเตอร์เน็ตนัก เอดีโอด็อทเน็ตจึงมีคุณสมบัติที่สนับสนุน Disconnected Connection ด้วย

นอกจากนั้นยังสามารถใช้กับแหล่งข้อมูลหลายๆแบบเหมือนเดิม เช่น Access กับ OLE DB Provider ของหลายๆแบบ หรือกระทั่งข้อมูลใน Outlook Exchange เราก็สามารถใช้ เอดีโอด็อทเน็ตไปเลือกออกมาได้เช่นเดียวกัน

ออบเจ็กต์ที่สำคัญในเอดีโอด็อทเน็ต

ออบเจ็กต์ที่ใช้อยู่บ่อยๆใน เอดีโอด็อทเน็ตซึ่งจะกล่าวถึงในส่วนนี้คือ Connection, Dataset และ Data Adapter

ออบเจ็กต์คอนเน็คชั่น (Connection Object)

ออบเจ็คตัวแรกก็คือ Connection เป็นออบเจ็กต์ที่อยู่ใน เอดีโอด็อทเน็ตหน้าที่คือ การเปิดการเชื่อมต่อ (Connection) ระหว่างไคลเอ็นต์ของเรากับแหล่งข้อมูล โดยที่เราสามารถกำหนดรูปแบบเชื่อมต่อไปยังฐานข้อมูลที่ต้องการ พร้อมค่ารายละเอียดต่างๆ เช่น รหัสผ่านเมื่อจะเชื่อมต่อกับฐานข้อมูล ซึ่งยังคงคล้ายกับ เอดีโอแบบเดิมนั่นเอง

ออบเจ็กต์คอมมานด์ (Command Object)

ออบเจ็กต์ตัวนี้มีหน้าที่เอาไว้เขียนคำสั่ง เอสคิวแอล (SQL) จากไคลเอ็นต์ไปทำงานบนแหล่งข้อมูล เช่นเดียวกับออบเจ็กต์ Connection คือใน เอดีโอปกติก็ยังมีออบเจ็กต์นี้อยู่

การใช้งานออบเจ็กต์ Command ในเอดีโอด็อทเน็ตก็เหมือนกับการทำงานใน เอดีโอเดิม คือในเวลาที่เราจะส่งคำสั่งเอสคิวแอลที่ไม่ใช่การเลือก (Select) เช่น Insert, Update หรือ Delete ที่ไม่มีการคืน (Return) ค่ากลับมานั้น ซึ่งส่วนใหญ่ในทุกฐานข้อมูลจะมีโครงสร้างภาษาเอสคิวแอล ให้เราสามารถสร้างฟังก์ชั่นฝังไว้ในตัวฐานข้อมูลได้

ออบเจ็กต์ดาต้าเซต (Dataset Object)

ออบเจ็กต์ดาต้าเซ็ตนี้มีเฉพาะในเอดีโอด็อทเน็ตเท่านั้น ดาต้าเซตเปรียบเสมือนฐานข้อมูลทั้งก้อนเลย คือเมื่อเรา เลือก (Select) ออกมา เราไม่ได้เฉพาะแถวข้อมูลออกมา แต่ได้ข้อมูลอันใหม่อันหนึ่งโดยเป็นฐานข้อมูลที่อยู่ในหน่วยความจำของเรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในดาต้าเซ็ตนั้น จะมองว่าเป็นฐานข้อมูลเหมือนกัน ในนั้นมีตาราง (Table), คอลัมน์ (Columns) และแถว (rows) ต่างๆอยู่ภายใน มีความสัมพันธ์ระหว่างตารางได้มีการกำหนดข้อบังคับความถูกต้อง (Constrain), ดรรชนี (Index key), คีย์ร่วม (Public key) ได้โดยการเก็บข้อมูลของดาต้าเซต จะเก็บโครงสร้าง (schema) แข้อมูลของตัวมันด้วย เราสามารถเลือกข้อมูลออกมาเก็บไว้ในดต้าเซต จากนั้นเราก็สามารถออกจากการเชื่อมต่อได้โดยเราก็จะมีดาต้าเซตเอาไว้ใช้งานจากนั้นภายในโปรแกรมเราก็สามารถใช้คำสั่ง Insert, Update, Delete ลงไปใน Dataset ได้ จนเสร็จเรียบร้อยแล้วเราสามารถกลับมาเชื่อมต่อกับฐานข้อมูลอีกทีหนึ่ง แล้วทำการซิงค์ (Synch) ข้อมูลกลับขึ้นไปได้ โดยตัวเอดีโอดอทเน็ตจะคอยจัดการความแตกต่างระหว่างข้อมูลด้วยกันเอง

การซิงค์ข้อมูลกลับขึ้นไปในลักษณะนี้ คือการเขียนโปรแกรมแบบ Disconnected Programming โดยไม่จำเป็นต้องเชื่อมต่อกับฐานข้อมูลตลอดเวลา เพราะแอพพลิเคชั่นบนอินเตอร์เน็ตนั้น ถ้าเราเชื่อมต่อฐานข้อมูลไว้ตลอดเวลาจะเป็นการสิ้นเปลืองทรัพยากรของเซิร์ฟเวอร์มาก

ออบเจ็กต์ดาต้าอะแด็ปเตอร์ (Data Adapter Object)

ออบเจ็กต์ ดาต้าอะแด็ปเตอร์ มีเฉพาะในเอดีโอด็อทเน็ตเช่นกัน ออบเจ็กต์ดาต้าอะแด็ปเตอร์ก็เหมาะสำหรับการเลือก (Select) ทั่วๆไป ซึ่งเปรียบเสมือนเป็นสะพานเชื่อมระหว่างแหล่งข้อมูลที่มีข้อมูลกับตัวดาต้าเซ็ต ของเราโดยลำเลียงข้อมูลจากแหล่งข้อมูลเอามาเก็บในตัวดาต้าเซ็ต จากนั้นจึงสามารถใช้คำสั่งเอสคิวแอลในดาต้าเซ็ต ทำการเปลี่ยนแปลงข้อมูลในแถวนั้นๆได้ และหากต้องการซิงค์ข้อมูลจากดาต้าเซ็ต กลับไปในแหล่งข้อมูลใหม่ เราก็ใช้ดาต้าอะแด็ปเตอร์ ทำการอัพเดตข้อมูลกลับขึ้นไปในแหล่งข้อมูล

หากเปรียบเทียบภาพของดาต้าอะแด็ปเตอร์ ก็เหมือนสะพานเชื่อมโยงระหว่างฐานข้อมูลของเรากับดาต้าเซ็ต นั่นเอง การที่จะเอาข้อมูลมาเก็บไว้ดาต้าเซ็ต เราต้องสร้างออบเจ็กต์ขึ้นมา 2 ตัว ตัวแรกเป็นดาต้าเซ็ต ตัวที่ 2 เป็นดาต้าอะแด็ปเตอร์ หลังสร้างเสร็จแล้วให้ใช้คำสั่งเอสคิวแอลเพื่อดึงข้อมูลมาจากนั้นใช้คำสั่ง Fill เพื่อนำข้อมูลที่ได้จากการอ่านเข้ามาใส่ในดาต้าเซ็ต

คุณลักษณะ	ADO	ADO.NET
สถาปัตยกรรมที่ใช้	บนพื้นฐานคอม(COM) การเชื่อมโยง Connection Oriented	XML กับ HTTP การเชื่อมโยงแบบ Connectionless
การเข้าถึงระบบป้องกัน	ไม่ผ่านระบบ Firewall	ใช้ HTTP และสามารถผ่าน Firewall ได้เป็นอย่างดี
ออบเจ็กต์ที่ใช้	Connection, Command, Record set	Connection, Command, Dataset, Dataset view, Data view

ตารางที่ 1 เปรียบเทียบการเชื่อมต่อฐานข้อมูลระหว่างเทคโนโลยี ADO และ ADO.NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13.6 เอสคิวแอล เซิร์ฟเวอร์ 2000

เอสคิวแอลเซิร์ฟเวอร์ 2000 (SQL Sever 2000 : Sequential Query Language Server 2000) จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDMBS : Relation Database Management System) ตัวหนึ่ง ซึ่งเป็นที่นิยมกันมากในปัจจุบัน สามารถจัดการฐานข้อมูลอย่างมีประสิทธิภาพ และสนับสนุนการใช้งานบนระบบปฏิบัติการวินโดวส์ (Window) ได้เป็นอย่างดีโดยเอสคิวแอล เซิร์ฟเวอร์ 2000 (SQL Sever 2000 : Sequential Query Language Server 2000) ถูกออกแบบมาให้ทำงานในลักษณะที่เป็นระบบฐานข้อมูลระหว่างเครื่องเซิร์ฟเวอร์ กับเครื่องลูกข่าย (Client Server Database) ในระบบปฏิบัติการ (Windows NT) และระบบปฏิบัติการวินโดวส์ 2000 (Windows 2000) นอกจากนี้ เอสคิวแอลเซิร์ฟเวอร์ 2000 ยังสามารถใช้งานร่วมกับเว็บดีวีล็อปเม็นท์แพลตฟอร์ม (Web Development Platform) ทั้งหลายไม่ว่าจะเป็นภาษาซี (C) ซีพลัสพลัส (C++) จาวา (Java) เพิร์ล (Perl) ทีซีแอล (Tcl) หรือ เอเอสพี (ASP) ก็ตาม

เอสคิว แอลเซิร์ฟเวอร์ 2000 เป็นระบบจัดการฐานข้อมูลแบบ Client Server Relation Database ทำให้ช่วยเพิ่มประสิทธิภาพในการทำงาน และระบบการจัดการแบบควบคุมจากศูนย์กลาง (Centralize Management) โดยระบบรักษาความปลอดภัย ระบบจัดสรรการทำงาน และใช้ข้อมูลจากหลายๆงานพร้อมกันได้ ซึ่งประกอบด้วย 3 ส่วนหลักคือส่วนของผู้ให้บริการ (Sever) ส่วนของผู้ใช้บริการ (Client) และข่ายการสื่อสารหรือเน็ตเวิร์ค โดในแต่ละส่วนก็จะมีโปรแกรมสำหรับการทำงานตามหน้าที่ของตน

ส่วนของผู้ให้บริการ (Server)

เป็นคอมพิวเตอร์หลักในระบบเครือข่าย (Network) ที่ทำการติดตั้งโปรแกรมการทำงานเอสคิวแอลเซิร์ฟเวอร์ โดยทำหน้าที่จัดเก็บรวบรวม ค้นหา เรียกดู จัดการข้อมูลในรูปของ ฐานข้อมูล และจัดเก็บข้อมูลที่เกิดจากการที่ผู้ใช้แต่ละคนสร้างขึ้นมา เพื่อที่คอมพิวเตอร์ในเครือข่ายจะมาเรียกไปใช้ได้

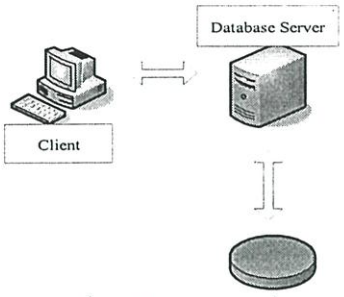
ส่วนของผู้ใช้บริการ (Client)

เป็นเครื่องคอมพิวเตอร์ที่ใช้ทำการติดตั้ง เว็บดีวีล็อปเมนท์แพลตฟอร์มต่างๆ เช่น จาวา เพิร์ล เอเอสพี เป็นต้น และ เชื่อมต่อกับเอสคิวแอลเซิร์ฟเวอร์ ฝั่งเซิร์ฟเวอร์ โดยโปรแกรมฝั่งไคลเอ็นต์จะทำหน้าที่ส่งต่อข้อมูลจากดาต้าเบส และมีโปรแกรมสำหรับเรียกดู และจัดการข้อมูล โดยมีสิทธ์เรียกดูข้อมูลบนเครื่องเซิร์ฟเวอร์ได้

ข่ายการสื่อสารหรือเนทเวิร์ค (Communication Network)

เป็นเส้นทางในการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องเซิร์ฟเวอร์และเครื่องไคลเอ็นต์ ดังนี้

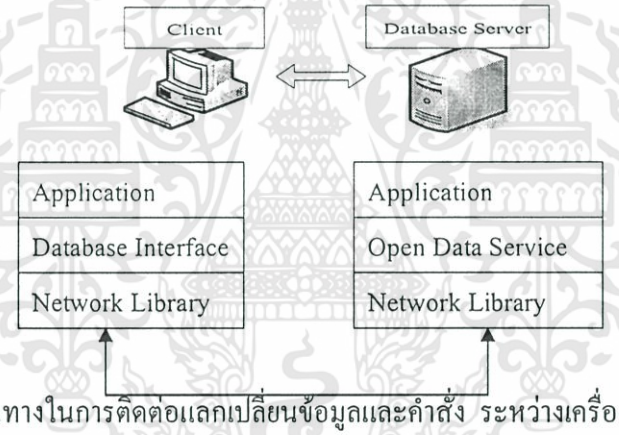
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 21 แสดงเส้นทางในการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องเซิร์ฟเวอร์และเครื่องไคลเอ็นต์

สถาปัตยกรรมของเอสคิวแอล เซิร์ฟเวอร์ (SQL Sever)

สถาปัตยกรรมหรือโครงสร้างภายในของเอสคิวแอลเซิร์ฟเวอร์ จะมีส่วนที่ทำหน้าที่ติดต่อสื่อสารกับทางฝั่งไคลเอ็นต์อยู่ด้วย แต่ก็แยกส่วนจัดการเน็ตเวิร์คและโปรโตคอลออกจากส่วนที่เป็นแอพพลิเคชั่น ทำให้แอพพลิเคชั่นสามารถทำงานอยู่บนเน็ตเวิร์คแบบใดก็ได้



รูปที่ 22 แสดงเส้นทางในการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องดาต้าเบสเซิร์ฟเวอร์และเครื่องไคลเอ็นต์

จากรูปที่ 22 แอพพลิเคชั่น (Application) หมายถึง โปรแกรมที่พัฒนาขึ้นเพื่อใช้ข้อมูลจากดาต้าเบสผ่นอินเตอร์เฟสของโปรแกรมที่เรียกว่า เอพีไอ (API : Application Program Interface) เพื่อใช้งานกับดีวีล็อปเมนท์แพลตฟอร์มต่างๆมากมาย ไม่ว่าจะเป็นภาษาซี ซีพลัสพลัส จาวา ไอเฟล เพิร์ล หรือ ทีซีแอล เป็นต้น และนอกจากนี้ยังสามารถใช้งานร่วมกับโอดีบีซี (ODBC : Open Database Connectivity) ซึ่งทำให้เราสามารถใช้งานได้กลับเครื่องมืออื่นๆ บนวินโดวส์แพลตฟอร์ม เช่น Access เป็นต้น รวมทั้งสามารถนำมาประยุกต์เพื่อใช้งานร่วมกับ ASP (Active Server Page)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดาต้าเบสอินเตอร์เฟส (Database Interface) หมายถึง อินเตอร์เฟสที่ใช้โดยแอพพลิเคชั่นที่ติดต่อไปยังเอสคิวแอลเซิร์ฟเวอร์ เช่น ODBC โดยสร้างการติดต่อระหว่าง แอพพลิเคชั่น และ OLE DB (Object Linking and Embedding Database)

เน็ตเวิร์คไลบรารี (Network Library) หมายถึง โปรแกรมที่ทำหน้าที่จัดส่งข้อมูลผ่านไปทางเน็ตเวิร์ค โดย เอสคิวแอล เซิร์ฟเวอร์ ซึ่งเซิร์ฟเวอร์ มีเน็ตเวิร์คไลบรารี ที่ทำงานร่วมกับโปรโตคอลได้หลายประเภท เช่น ทีซีพีไอพี (TCP/IP) เนมไปป์ (Named Pipe หรือ NT) เป็นต้น

โอเพนดาต้าเซอร์วิส (Open Data Service) เป็นอินเตอร์เฟสระหว่างเน็ตเวิร์คไลบรารีกับแอพพลิเคชั่นฝั่งเซิร์ฟเวอร์

13.7 การออกแบบฐานข้อมูลโดยการใช้ไนแอมโมเดล (NIAM Model)

NIAM (Nijssen's Information Analysis Methodology) คิดค้นโดย ศาสตราจารย์ G.M.Nijssen และทีมงาน เป็นวิธีการในการออกแบบฐานข้อมูลโดยใช้สัญลักษณ์ต่างๆในการแทนการจำลองข้อมูล ความสำคัญและข้อกำจัดต่าง ของข้อมูล เป็นวิธีที่มีอัลกอริทึมที่สามารถออกแบบฐานข้อมูลแบบความสัมพันธ์(Relational)ให้อยู่ในรูปแบบบรรทัดฐานที่ 5 (5 NF) โดยไม่ต้องทำการนอร์มาไลซ์ (Normalization) และมีอินเทอร์เฟสที่คนสามารถเข้าใจได้ง่ายเนื่องจากเป็นภาษาธรรมชาติ(Natural Language Interface)

ส่วนประกอบของไนแอมโมเดล

-Entity type คือ เซตของสิ่งที่สนใจซึ่งอาจจะเป็นสิ่งที่จับต้องได้หรือไม่ได้

-Label (Value type) คือเซตของสิ่งของที่ใช้บอกความแตกต่างหรือชื่อของแต่ละ Entity โดย Entity type ทุกตัวจะต้องมีlabel type

-Role ใช้ในการแสดงความสัมพันธ์ที่เกี่ยวข้องกับ Entity type ที่เชื่อมต่ออยู่

-Fact type คือ เซตของความสัมพันธ์ระหว่างสมาชิกของ Entity type ตั้งแต่2 Entity ขึ้นไป

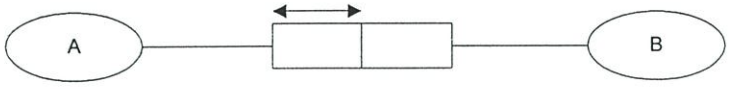
-Reference type คือ เซตของความสัมพันธ์ระหว่างสมาชิกของ Entity type กับ สมาชิกของ Label type

-Nested Fact type เป็น Entity type ชนิดหนึ่งที่แสดงความสัมพันธ์ในการกำหนดกลุ่มของFact type

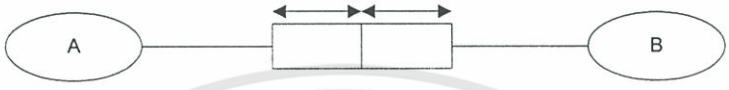
การเขียนความสัมพันธ์ในรูปแบบของไนแอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One to many relationship 1:M)



ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to one relationship 1:1)



ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many to many relationship M:M)

13.8 Web Services

แอพพลิเคชั่นบนเว็บหมายถึงการที่จะสามารถร้องขอแอพพลิเคชั่นนี้อย่างแบบโปรแกรมบนเว็บโดยใช้โปรโตคอล SOAP

XML + HTTP = SOAP

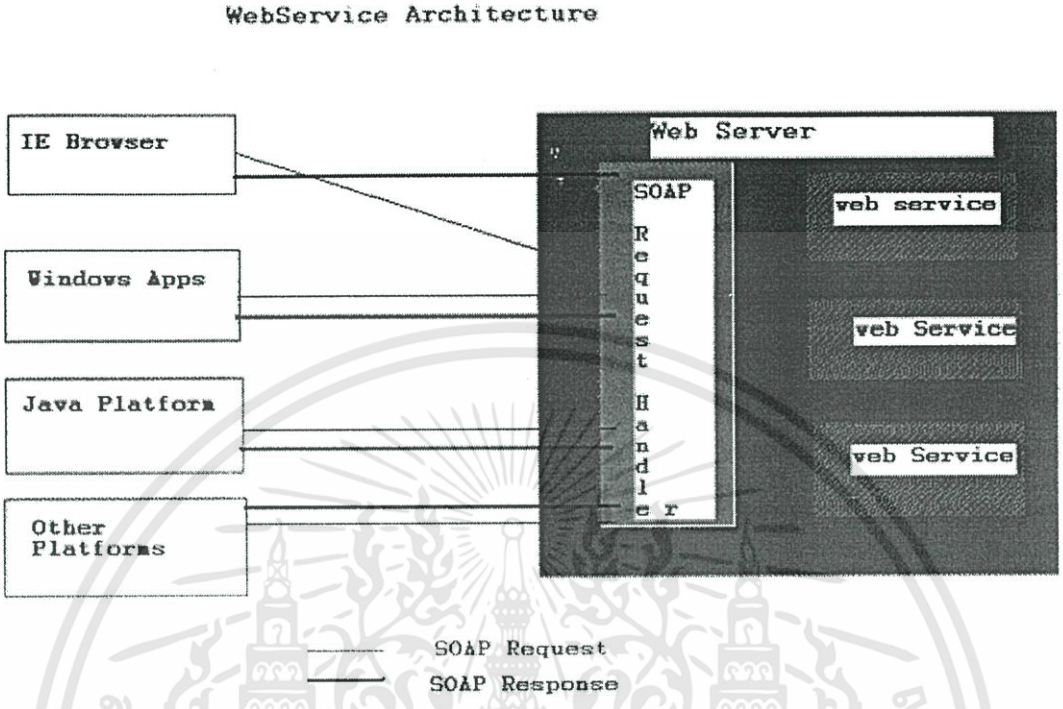
เว็บเซอร์วิสอนุญาตให้แอพพลิเคชั่นใช้ข้อมูลร่วมกันได้ และสามรถถูกเรียกใช้งานข้ามรูปแบบพื้นฐาน(Platform) และระบบปฏิบัติการได้โดยอ้างเป็นภาษาโปรแกรม โดย .NET เป็นโครงสร้างพื้นฐานของ Microsoft สำหรับ XML เว็บเซอร์วิส ใช้ SOAP เพื่อร้องขอเว็บเซอร์วิส (HTTP-GET,HTTP-POST,SOAP)

SOAP – Simple Object Access Protocol เป็นโปรโตคอลแบบข้อความพื้นฐาน (Message-Base) ขึ้นกับการร้องขอการตอบสนอง

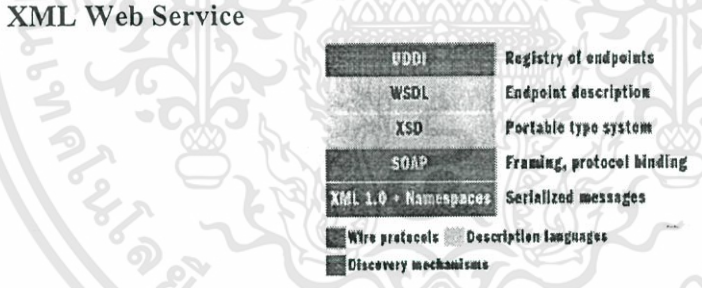
ข้อได้เปรียบของการใช้เว็บเซอร์วิส อยู่ตรงที่การจ่ายเมื่อใช้ ตัวอย่าง เช่น การผลิตแอพพลิเคชั่นออกมาเป็นซอร์ฟแวร์แล้วจะใช้ต้องเสียค่าลิขสิทธิ์เพื่อซื้อมาใช้ ซึ่งอาจจะแพงเกินไป แต่ถ้าเป็นเว็บเซอร์วิสแล้วละก็เมื่อใช้ก้ค่อยจ่ายค่าทำเนียมตามครั้งที่ใช้ก็พอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Web Services Architecture



รูปที่ 23 สถาปัตยกรรมเว็บเซอร์วิส



รูปที่ 24 สถาปัตยกรรม XML เว็บเซอร์วิส

คือ โครงสร้างพื้นฐานในการเคลื่อนย้าย เพื่อกระจายข้อมูลบนอินเตอร์เน็ต เป็นส่วนประกอบของโปรแกรมที่อนุญาติให้สร้าง Application ที่ขยายได้, ยืดหยุ่น, ไม่ขึ้นกับ platform ใด เว็บเซอร์วิสมีการให้ใช้ แอพพลิเคชั่นเพื่อแลกเปลี่ยนข้อความโดยใช้โปรโตคอลพื้นฐาน เช่น HTTP, XML, XSD, SOAP และ WSDL

เว็บบริการ XML สามารประกอบด้วยการเจาะจงการเขียนโปรแกรมเหตุผล เพื่อ จัดทำคุณสมบัติการใช้งาน เช่น การคำนวณภาษีเงินได้ XML เว็บเซอร์วิสที่คำนวณภาษีเงินได้ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอพพลิเคชั่นทางลูกค้าเพื่อขอข้อมูลของลูกค้า เช่น รายได้, เงินออม และ การลงทุนระหว่างปี แอพพลิเคชั่นทางลูกค้าสามารถเรียกวิธีการบนบริการเพื่อเรียกข้อมูลที่จำเป็น ในการทำงาน call (Method Call) และข้อมูลที่เกี่ยวกับการทำงานนั้นจะถูกดึงมาที่เว็บเซอร์วิสในรูปแบบ XML โดยใช้ SOAP บน HTTP

XML เว็บเซอร์วิส มีส่วนประกอบ 3 ส่วน

-Directories เป็นส่วนกลางที่ใช้เก็บข้อมูลสาธารณะ เกี่ยวกับ XML เว็บเซอร์วิส UDDI จะระบุเฉพาะเจาะจงเกี่ยวกับแนวทางสำหรับข้อมูลสาธารณะ เกี่ยวกับ XML เซอร์วิส

-Discovery เป็นกระบวนการอนุญาติให้ลูกค้าใช้ลูกค้าได้ทราบเกี่ยวกับรูปลักษณ์ของ XML เว็บเซอร์วิสและเกี่ยวกับที่ตั้งเฉพาะเจาะจง XML เซอร์วิส

-Description เป็นเอกสาร XML ที่เจาะจงรูปแบบของข้อความที่ XML เว็บเซอร์วิส สามารถเข้าใจได้ เช่นเอกสารบรรยายเกี่ยวกับรูปแบบข้อความ SOAP ที่ใช้ร้องขอวิธีการ บน XML เว็บเซอร์วิส

-Communication Wire Formats การติดต่อแบบใช้สายสามารถเข้าใจได้ทุกระบบที่สนับสนุนการใช้ เว็บธรรมดา เช่น HTTP, SOAP โดย HTTP POST เป็นโปรโตคอล ที่อนุญาตให้ส่งพารามิเตอร์ และ โปรโตคอล HTTP GET อนุญาติให้ส่ง URL-encode พารามิเตอร์ไปยัง XML เว็บเซอร์วิส โปรโตคอล SOAP อนุญาตใช้แลกเปลี่ยนโครงสร้าง และรูปแบบข้อมูลระหว่างแอพพลิเคชั่น บนอินเตอร์เน็ต

13.9 หลักการในการทำระดับ (Principles of Leveling)

หลักการเบื้องต้นเกี่ยวกับเครื่องมือก็ คือ เมื่อตั้งกล้องและตั้งระดับของกล้องแล้ว แนวเล็งกล้อง (Collimation line หรือ Line of sight) จะเป็นแนวเดียวกันกับแนวราบ (Horizontal line)

สำหรับไม้วัดระดับ เมื่อตั้งฟองกลมของไม้วัดระดับ (Staff) เราจะถือว่าไม้วัดระดับนั้นอยู่ ในแนวดิ่งและไม้วัดระดับหน้า-หลัง (FS/BS) จะขนานกันในแนวดิ่ง Staff บางทีเรียกว่า Leveling rod

พื้นหลักฐานการระดับ (Datum) จะอ้างอิงจากระดับน้ำทะเลปานกลาง (Mean sea Level = MSI หรือ รทก.) ที่ระดับ รทก. จะถือว่ามีค่าระดับเท่ากับศูนย์เสมอ ระดับน้ำทะเลปานกลางจะทำการวัดประมาณ 18.6ปี (Mutation period) จึงจะให้ค่าที่ดี ทั้งนี้เพราะขั้วโลกจะหมุนมาอยู่ตำแหน่งเดิมพอดี

ในประเทศไทยสถานีวัดระดับน้ำทะเลจะอยู่ที่เกาะหลัก จังหวัดประจวบคีรีขันธ์ ในแต่ละประเทศก็จะมีจุดระดับเป็นของตัวเอง เพราะฉะนั้นค่าระดับจะเปรียบเทียบกันไม่ได้ ถ้าทำการก่อสร้างสะพานหรืออุโมงค์ข้ามประเทศ จึงต้องให้ค่าระดับมาจากจุดจุดเดียวจึงจะใช้ได้

ค่าความต่างระดับ ($A\Delta h$) = BS – FS

ค่าระดับของจุด B = ค่าระดับที่A + BS – FS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายของคำต่างๆที่ใช้ในการถ่ายระดับ มีดังนี้

1. ความสูงของแกนกล้อง (Collimation line) ความสูงนี้จะนับจากระดับน้ำทะเลปานกลางจนถึงแกนกล้อง ทุกครั้งที่มีการทำระดับ จะคำนวณหาค่าระดับของจุดตั้ง Staff ต่อเนื่องกันไป บางทีเรียกว่าวิธีความสูงแกนกล้อง (Height of Instrument) HI หรือ HPC (Height of Plane of Collimation) เพราะส่องรอบตัว .

2. หมุดหลักฐานการระดับ (มฐ.) หรือ Bench Mark (BM) จะเป็นหมุดอ้างอิงค่าระดับจริงที่นับเนื่องจากระดับน้ำทะเลปานกลาง กรมแผนที่ทหารจะทำการถ่ายระดับและสร้างหมุดหลักฐานกระจายไปทั่วประเทศ

3. หมุดถ่ายระดับ (Turning Point (TP)) หรือหมุดเปลี่ยนระดับ หรือหมุดตั้ง Staff (Change Point (CP)) หมายถึง จุดตั้ง Staff เพื่อทำการถ่ายระดับ ซึ่งจะต้องวางTPหรือGround Plate เพื่อตั้ง Staff หัวหมุดจะต้องเป็นทรงกลมเสมอ เพราะจะต้องมีจุดสูงสุดเพียงจุดเดียว เราจะใช้ถ่ายระดับต่อเนื่องกันไป โดยการส่อง Back Sight(BS) หรือ Staff หลังและส่อง Fore Sight(FS) หรือ Staff หน้า

4. ค่า Back Sight (BS) หรือไม้หลังหรือ Staff หลัง หมายถึงค่า Staff ที่อ่านได้ครั้งแรก หลังจากตั้งกล้องตั้งระดับแล้วจึงส่องอ่านค่าดังกล่าว

5.ค่า Fore Sight (FS) หรือไม้หน้า หมายถึง ค่าStaff ที่อ่านได้ครั้งสุดท้ายก่อนใช้ถ่ายระดับในจุดนั้น

6.ค่า Intermediate Fore Sight (IFS) หรือค่าไม้กลาง บางทีเรียกว่า Intermediate Sight (IS) หมายถึงค่า Staff ที่อ่านได้หลังจากการส่อง Back Sight (BS) แล้ว จะส่องกี่ค่าก็ได้ ใช้ในการทำระดับเพื่อหาระดับดินเดิมของพื้นดิน (NGL = Natural Ground Level) ซึ่งเราเรียกว่า การทำระดับพื้นที่ (Area leveling) และการทำระดับตามแนวศูนย์กลาง (Profile leveling) เพื่อหาระดับดินเดิม

7. ตอนการระดับ (Section) หมายถึงการถ่ายระดับระหว่างหมุดระดับ (Bench Mark) สองหมุดปกติความยาวจะประมาณ 2 ถึง 3 กิโลเมตร

8. ช่วงการระดับ (Subsection) ถ้าตอนการระดับยาว 2 กิโลเมตร เราอาจจะแบ่งย่อยลงอีกเป็นระยะ 1 กิโลเมตรเพราะถ้าถ่ายระดับไป 2 กิโลเมตร และกลับอีก 2 กิโลเมตร ถ้าทำผิดจะต้องถ่ายระดับใหม่อีก 2 กิโลเมตร ถ้าแบ่งย่อย 1 กิโลเมตร จะทำให้ทราบว่าช่วงไหนทำผิด สามารถแก้โดยถ่ายเพียง 1 กิโลเมตร

9. สายการระดับ (Leveling line) คือ ระยะทางรวมของการถ่ายระดับแต่ละตอนต่อเนื่องกันไป เช่น สมมติถ่ายระดับจากกรุงเทพมหานคร ไปสระบุรี เราจะเรียกว่าสายการระดับ ถ้าถ่ายไป-กลับทีละสองกิโลเมตร เราเรียกว่าตอนการระดับ ดังกล่าวแล้ว

10. หมุดหลักฐานการระดับชั่วคราว (Temporary Bench Mark = TBM) เป็นหมุด BM (มฐ.) ชั่วคราวจะทำโดยใช้ตะปูตอกบนรากต้นไม้ และจะกำหนดหมุดขึ้นเมื่อสิ้นสุดการทำงานแต่ละวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการปฏิบัติของกล้องระบบ Mechanics

1. จะต้องตั้งกล้องอยู่กึ่งกลางระหว่าง Staff ทาง BS และ FS ฉะนั้นการตั้งกล้องผู้ส่องกล้องจะต้องประมาณระยะให้ดี ให้จำเป็นหลักการง่ายๆว่า ระยะหน้าหลังต้องเท่ากัน

2. การตั้งกล้องอยู่กึ่งกลางจะกำจัดความผิดที่เกิดจากความโค้งของโลก การหักเหของแสง และความผิดที่เกิดจากสายใยเอียง (Collimation error)

3. การตั้งหรือตอก TP จะต้องให้แน่น เพื่อป้องกันไม่ให้เกิดการทรุดตัว จะทำให้ค่า Staff ผิด

4. ปรับภาพ Staff ภาพของสายใยให้ชัดเท่ากันโดยการเหลือบตาขึ้นลงถ้าภาพของสายใยและภาพของStaff เคลื่อนที่ไปด้วยกัน แสดงว่าชัดเท่ากัน ถ้าเคลื่อนไปคนละทาง แสดงว่ายังชัดไม่เท่ากัน

5. กางร่มให้กล้องทุกครั้งและตลอดเวลาที่ส่องอ่าน

6. จดค่า Staff ให้ถูกต้อง คนจดจะต้องอ่านทวนให้คนกล้องฟังเสมอ

7. ถ้า Ground plate มี2ปุ่มควรจะตั้งstaff 2ครั้ง จะแทนการถ่ายไป-กลับได้เลย จะได้ไม่เสียเวลาถ่ายระดับกลับ

กฎการจดสมุดสนาม(Field book rule)

1. จะต้องไม่ลืมวิธีการจด เช่น ค่า BS FS ใส่ให้ถูกช่อง
2. บันทึก วันที่ เวลา และอากาศ
3. เขียนคำอธิบายเพื่อความเข้าใจ
4. ใช้สัญลักษณ์ทั่วไปที่คนอื่นเข้าใจได้
5. จะต้องเขียนรูปมาด้วย เช่น ตำแหน่ง BM, ตำแหน่งของสิ่งก่อสร้างที่อยู่ใกล้
6. อย่าเขียนค่าต่างๆ ปนกัน หรือมากจนล้นหน้ากระดาษ
7. ห้ามลบขูดตัวเลขที่ผิด ข้อมูลที่บันทึกผิดให้ขีดฆ่าไว้
8. บันทึกทุกสิ่งที่เกี่ยวข้อง เช่น คนส่องกล้อง คนถือ Staff คนทาสี
9. ใช้ตัวหนังสือมาตรฐาน เช่น ถ้าภาษาอังกฤษควรจะเป็นตัวพิมพ์ใหญ่ ควรใช้คนที่มีลายมือสวยงาม
10. ถ้านำค่าไปเขียนแบบ เช่น Profile ควรจะใช้อักษรโตตามมาตรฐานของกรม กอง นั้นๆ

การทำ Cross section คือการทำระดับของดินเดิม (Existing ground) ไปในแนวตั้งฉากกับเส้นฐานของแนวสำรวจ (Base line) ค่าระดับของดินเดิมที่ได้มาจะนำมาเขียนลงในกระดาษเขียนแบบ จุดประสงค์ก็เพื่อที่จะหาจำนวนปริมาณงานดินเป็นประการสำคัญ และเพื่อให้เป็นแนวทางพิจารณาการออกแบบทางด้านการระบายน้ำบ้างในบางแห่ง และกำหนดแนวทาง และอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทราบจุดประสงค์ที่ต้องทำ Cross section แล้วช่างสำรวจก็ต้องทำ Cross section ให้ได้ค่าที่จะนำไปหาปริมาณงานดิน ที่ใกล้เคียงความจริงมากที่สุด การเก็บค่าระดับดินเดิมในบางจุดบน Station ที่จะมีผลทำให้ได้ระดับดินเดิม (NGL) เมื่อนำไปคำนวณหาปริมาณงานดินแล้วจะทำให้ผิดจากความจริงไปมาก ก็ให้เลี่ยงไปหาระดับของจุดที่ใกล้เคียง ซึ่งเมื่อพิจารณาแล้วเห็นว่าได้ค่าระดับดินเดิมนั้นถูกต้อง เมื่อนำไปหาปริมาณงานดินแล้วจะได้ค่าถูกต้องกว่า

การทำ Cross section ด้วยกล้องระดับ

วิธีปฏิบัติในสนาม

หมุดระดับที่จะต้องใช้ในการทำ Cross Section ก็คือ BM หรือ TBM หรือ Station ที่ได้ทำการส่อง Profile ดังนั้นจะสามารถนำมาใช้ได้ เพราะทุกจุดทราบค่า Elevation บน HUB แล้ว การปฏิบัติทำได้ดังนี้

1. คนกล้องตั้งกล้องส่อง BS ไปยัง BM หรือ HUB หรือ TBM
2. หมุนกล้องส่องค่า IFS บน Staff ที่ตั้งบน Cross Line หรือ Spur Line ในขณะเดียวกัน Chain Man ก็วัดระยะจาก Base Line ไปยังจุดตั้ง Staff นั้น
3. ข้อมูลต่างๆผู้จดบนทึกจะต้องจด และเขียนภาพ Sketch ต่างๆ อย่างละเอียด
4. การจับฉากถ้าระยะใกล้จะใช้ Optical square ถ้าไกลมากจะใช้กล้อง Theodoline วางแนว

จุดที่ต้องทำ Cross Section

1. ทุก Station 25 เมตร(Sta. เดียวกับ Profile)
2. ทุกจุดที่มีท่อ
3. ทุกจุดที่มีทางน้ำ
4. ตอสะพานทั้งสองข้าง
5. ทุกจุดของทางแยก ที่ไม่ได้วาง Spur Line

ขอบเขตของการเก็บ Cross Section

การเก็บระดับจะต้องเก็บกว้างบ้างข้างละเท่ากับ ROW หรือมากกว่าถ้าจำเป็น ในกรณี Spur Line ที่เป็นถนนอาจเก็บออกไปไม่เท่ากับ ROW จาก Center line เช่น ประมาณ 15-20 เมตร ดูข้อกำหนดทางหลวง ในกรณี Spur Line ที่เป็นคลองให้เก็บคลุมขอบตลิ่งไปข้างละไม่น้อยกว่า 10 เมตร

จุดที่ต้องเก็บระดับในแต่ละ Cross Section

1. ในกรณีที่มีถนนเดิมอยู่ให้เก็บที่ Base line กลางถนน ไหล่ทาง ขอบร่องน้ำข้างถน ก้นร่องน้ำข้างถนน และบนพื้นดินทุกจุดที่เปลี่ยนความลาด (Slope)
2. ในกรณีที่ไม่มีต้นทาง เก็บ Base line และตามจุดที่ดินเปลี่ยนความลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ในกรณีที่ไม่มีต้นทางและพื้นดินราบเสมอกัน สังเกตการณ์เปลี่ยนความลาดของพื้นดินได้ยาก เช่นทุ่งนา หรือ ป่า ที่พื้นดินเสมอกัน ก็ให้เก็บที่ Base line และทุก 5, 10, 20 เมตรและที่ ROW
4. หลีกเลี่ยงการเก็บระดับบนพื้นที่ที่เปลี่ยนความลาด ซึ่งจะทำให้ค่างานดินผิดไปมาก เช่น จอมปลวก เนินดิน ที่มีความยาวไปตาม Station ไม่ถึง 10 เมตร และคันนา หลุมหรือบ่อที่มีความยาวไปตาม Station ไม่ถึง 10 เมตร
5. ใน Cross Section บนท่อ นอกจากเก็บที่ Base line กลางถนนและไหล่ทางแล้ว ต้องตั้ง Staff เก็บที่ปลายปากท่อ (หลังท่อ) และที่ Inlet และ Outlet และบนพื้นดินที่เปลี่ยนความลาดจนถึง ROW

13.10 พฤติกรรมของดินต่อการก่อสร้างอุโมงค์

พฤฒิกรรมของดินต่อการเจาะอุโมงค์ในดินอ่อนคือ ดินเคลื่อนตัวไปทางช่องเปิด เนื่องมาจากเป็นจุดที่ความเครียดจะลดน้อยลง นอกจากนี้ ถ้าดินอยู่ใต้ระดับน้ำ ระดับน้ำจะมีการเคลื่อนตัวไปทางช่องเปิด ภายใต้กฎง่ายๆ นี้ รายละเอียดของการตอบสนองของพื้นดินจะแปรไปตามเทคนิคที่ใช้ในการเจาะดิน วัตถุประสงค์ของทฤษฎีและกรอบแนวความคิด ก็คือเพื่อแนะนำหลักการของการตอบสนองของดินที่เกิดขึ้นจากหัวเจาะซึ่งพัฒนาขึ้นมาเพื่อทำนายและอธิบายการเสียรูปของดิน

โดยทั่วๆ ไป การขุดเจาะดินอ่อนส่งผลต่อการเสียรูปของดินในสองทาง ก็คือ การเสียรูปที่มีสาเหตุโดยตรงมาจากการเจาะอุโมงค์ นี่นับได้ว่าเป็นการเสียรูปที่มีสำคัญและมักจะอธิบายได้โดย การสูญเสียมวลดิน (Ground loss) อีกข้อหนึ่งก็คือ การทรุดตัว ซึ่งจัดเป็นการทรุดตัวในระยะยาว ที่มีสาเหตุมาจาก การเพิ่มขึ้นของความเครียดรอบๆ อุโมงค์ การทรุดตัวมักจะเกิดในระยะยาวหลังจากที่มีการขุดเจาะ ซึ่งขึ้นอยู่กับ เงื่อนไขของ

การสูญเสียดินมีสองประเภท ประเภทแรกเกิดขึ้นในปีแรกของพัฒนาหัวเจาะ การสูญเสียมวลดินอาจเกิดขึ้นโดยทันที ไม่สามารถควบคุมได้และมีลักษณะรุนแรง เนื่องมาจากการเคลื่อนตัวของหัวเจาะ การไหลของดินละน้ำลงสู่ผิวหน้าของอุโมงค์ เหมือนกับในกรณีของ Marc Brunel ที่ขุดเจาะอุโมงค์ใต้แม่น้ำเทมส์ในปี 1828 แต่มันสามารถเกิดจากการไหลของดินลงสู่เส้นทางขุดเจาะหรือเกิดจากความไม่แข็งแรงของผนังอุโมงค์ อย่างไรก็ตาม หลังจากที่ เทคนิค close face shield tunneling ได้ถูกแนะนำ ซึ่งผิวหน้าของอุโมงค์อยู่ภายใต้การควบคุมโดยแรงดันอากาศ แรงดันของโคลน หรือแรงดันของดินที่ถูกให้ที่ด้านหน้าของหัวเจาะ การเคลื่อนตัวของดินอย่างรุนแรงจะสามารถป้องกันได้

การสูญเสียมวลดินในประเภทที่สองเกิดขึ้นมาจากกระบวนการการขุดเจาะดินดังรูปที่ 25-29 การสูญเสียดินเหนือหัวเจาะเกิดขึ้นได้ในห้าระยะที่แตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระยะที่ 1 การสูญเสียมวลดินด้านหน้าหัวเจาะเกิดขึ้นเมื่อใช้หัวเจาะแบบเปิดหน้า หรือเมื่อหัวเจาะมีความดันที่ต่ำ ทำให้ดินสามารถเคลื่อนตัวไปที่ผิวหน้า ในกรณีนี้ ปริมาตรเป็นลบ หรือปริมาตรส่วนมากจะถูกเอาออก

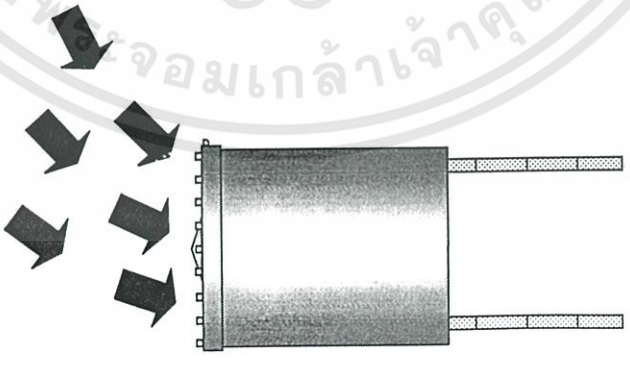
ระยะที่ 2 การสูญเสียมวลดินจากการตัดดินเกินขนาดด้านนอกอุโมงค์โดยรอบบริเวณผิวหน้าของเครื่องจักรซึ่งถูกทำให้เกิดโดยการยื่นใบตัดมากเกินไป (ขนาดใหญ่จนถึง 100 มิลลิเมตรด้านนอกของของพื้นผิว)

ระยะที่ 3 การสูญเสียมวลดินจากการก้มเงยของหัวเจาะสามารถตัดให้เกิดวงรีขนาดใหญ่ซึ่งมีเส้นผ่าศูนย์กลางมากกว่าหัวเจาะ ในทางทฤษฎี หัวเจาะที่มีขนาดยาวจะทำให้เกิดช่องว่างขนาดใหญ่ได้มากกว่าหัวเจาะที่มีขนาดสั้น

ระยะที่ 4 การสูญเสียมวลดินจากการรบกวนดินเนื่องมาจากการเลื่อนอย่างแรงของหัวเจาะที่มีเส้นศูนย์กลางขนาดใหญ่ สามารถเกิดการเคลื่อนตัวของดินเหนือหัวเจาะ

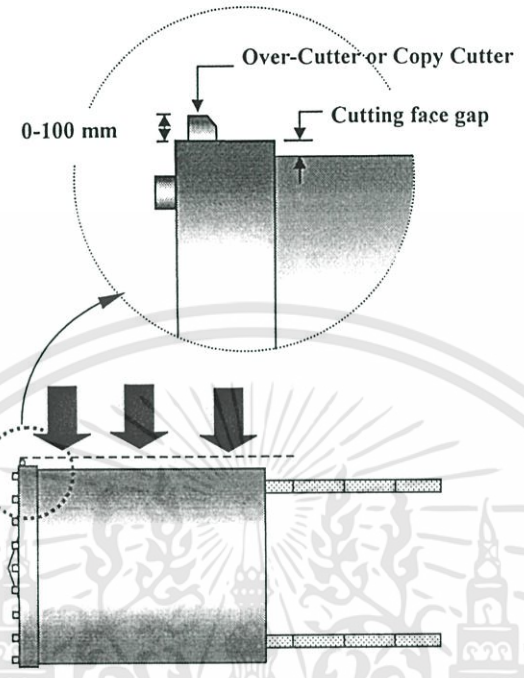
ระยะที่ 5 การสูญเสียมวลดินจากช่องว่างส่วนท้ายของหัวเจาะหลังจากที่หัวเจาะผ่านเข้าไปทำให้เกิด การเสียรูปของดินที่มีองค์ประกอบเพิ่มขึ้น เนื่องมาจากดินที่อยู่ด้านนอกเคลื่อนเข้ามาในช่องว่างที่เกิดขึ้นระหว่างดินที่อยู่รอบๆ การขุดเจาะกับผนังอุโมงค์ด้านนอก หนึ่งในวิธีการ ที่มักจะใช้ก็คือพยามยามที่จะลดช่องว่างโดยการขยายตัวอุโมงค์หรือการฉีดน้ำปูนรอบผนังอุโมงค์ เหมือนกับที่มันเกิดจากด้านท้ายของหัวเจาะก่อนที่ดินเคลื่อนตัวเข้ามาแทนที่ในช่องว่าง

การเคลื่อนตัวของดินที่มีปริมาณมากซึ่งเกิดจากการสูญเสียมวลดินเหล่านี้ มีอิทธิพลมาจากสภาพของดิน วิธีการก่อสร้าง การควบคุมหัวเจาะ การควบคุมนี้รวมไปถึงการควบคุมที่ความดันที่ส่วนหัว การควบคุมที่หัวเจาะ อัตราการผ่านเข้าไปของหัวเจาะ และคุณภาพของคนงาน ดังนั้น ทำให้เกิดการเคลื่อนตัวเข้าสู่ด้านหน้าของอุโมงค์ ส่งผลให้เกิดช่องว่างด้านท้ายที่มีขนาดใหญ่ขึ้น การรบกวนดินอย่างรุนแรงโดยการควบคุมที่แย่ และการติดตั้งผนังอุโมงค์ที่ช้าเนื่องจากระบบการทำงานที่ไม่ดี ทั้งหมดนี้ล้วนแต่ส่งผลเพิ่มการเคลื่อนตัวของดิน

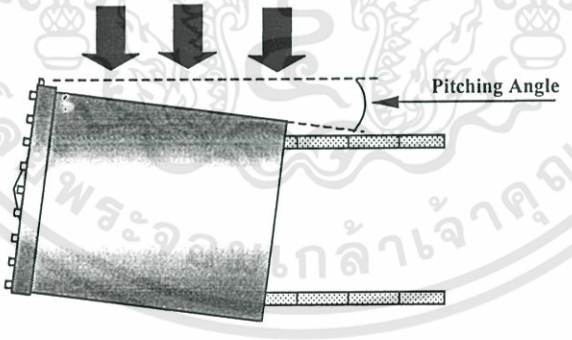


รูปที่ 25 การสูญเสียมวลดินด้านหน้าหัวเจาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

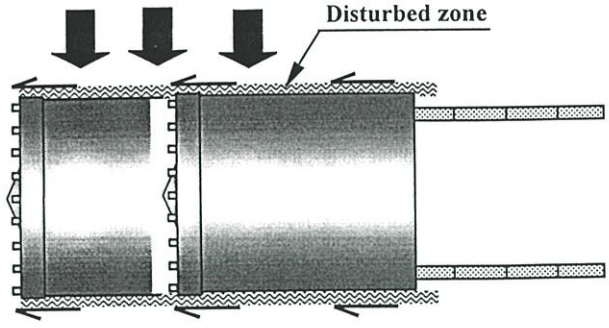


รูปที่ 26 การสูญเสียมวลดินจากการตัดดินเกินขนาด

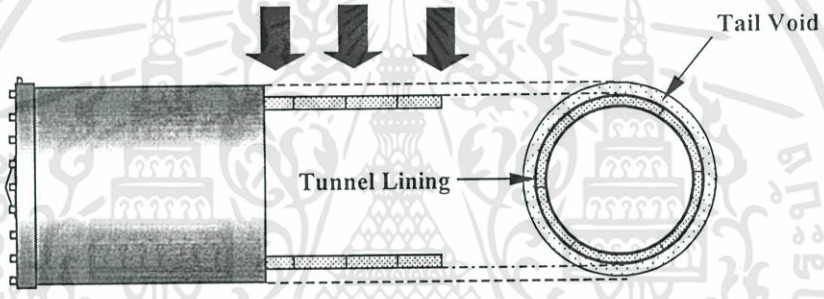


รูปที่ 27 การสูญเสียมวลดินจากการก้มเงยของหัวเจาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



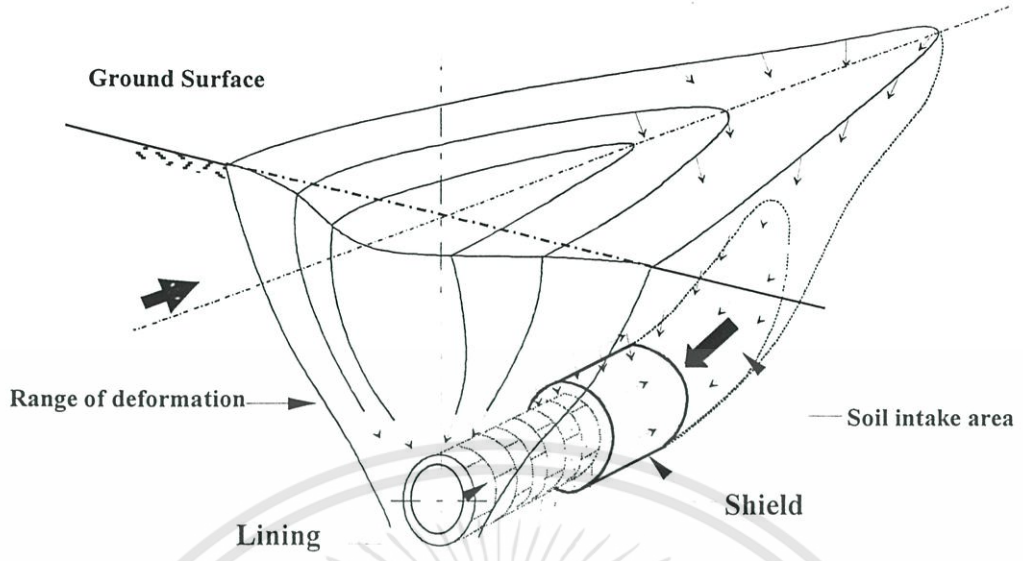
รูปที่ 28 การสูญเสียมวลดินจากการรบกวนดิน



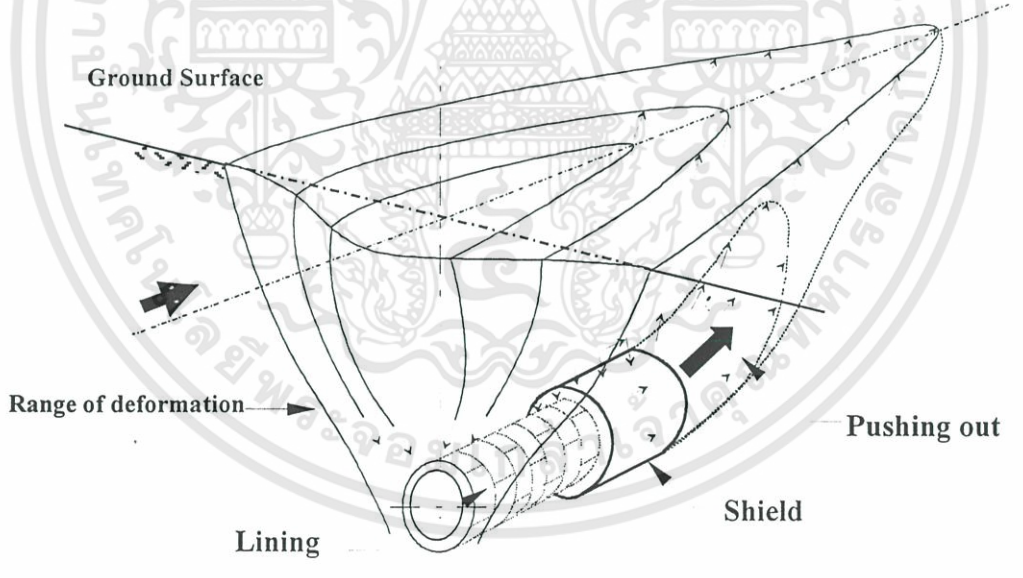
รูปที่ 29 การสูญเสียมวลดินจากช่องว่างส่วนท้ายของหัวเจาะ

การเคลื่อนตัวของดินอันเนื่องมาจากการสูญเสียมวลดินโดยตัวมันเองที่ผิวหน้าในช่องที่ขยายไปทางด้านข้างและด้านเหนือของผิวที่เกิดขึ้นมาก่อน รูปที่ 30 แสดงให้เห็น รูปสามมิติของการเคลื่อนตัวของดินซึ่งถูกทำให้เกิดโดยการสร้างอุโมงค์ในดินอ่อน อย่างไรก็ตาม ในกรณีที่ปริมาตรของดินที่ขุดออกไปน้อยกว่าปริมาตรของหัวเจาะที่เข้าไปอยู่ในดิน (อย่างเช่น หัวเจาะถูกกำหนดให้มีปริมาตรสมดุลเป็นบวก) ดินจะถูกดึงออก ในกรณีนี้สามารถเป็นผลของการให้แรงดันมากเกินไปที่ด้านหน้าของหัวเจาะ ดังนั้นการตรวจวัดความเสียหายที่เกิดจากการก่อสร้างอุโมงค์จะต้องคำนึงถึงพฤติกรรมการเคลื่อนตัวของดินอันเนื่องมาจากการสูญเสียมวลดินเพื่อสามารถกำหนดระบบและวิธีการวัดที่เหมาะสมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 30 การเคลื่อนตัวของในกรณีที่ปริมาตรของดินที่ขุดออกไปมากกว่าปริมาตรของหัวเจาะที่เข้าไปอยู่ในดิน



รูปที่ 31 การเคลื่อนตัวของในกรณีที่ปริมาตรของดินที่ขุดออกไปน้อยกว่าปริมาตรของหัวเจาะที่เข้าไปอยู่ในดิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14. การออกแบบ

14.1 การออกแบบระบบ

การออกแบบระบบการตรวจสอบความเสียหายจากการก่อสร้างอุโมงค์ใต้ดินแบบ Real – Time จะมีส่วนหลักๆคือ

. *ส่วนสำหรับพนักงานภาคสนาม* โดยจะมีรายละเอียดการทำงานดังนี้

-สามารถเลือก site และ section จากหน้าจอของ pocket pc phone

-สามารถกรอกรายละเอียดเกี่ยวกับการวัดผ่านหน้าจอ pocket pc phone

-สามารถประมวลผลการทรุดตัวของดินผ่านหน้าจอ pocket pc phone

-สามารถวาดกราฟการทรุดตัวของดินผ่านหน้าจอ pocket pc phone

ส่วนสำหรับพนักงานศูนย์ปฏิบัติการ โดยจะมีรายละเอียดการทำงานดังนี้

-สามารถสร้าง, เลือก และลบข้อมูลของ site, section และ Instrument จากหน้าจอของ PC server

-สามารถกรอกรายละเอียดเกี่ยวกับ site, section และ Instrument จากหน้าจอของ PC server

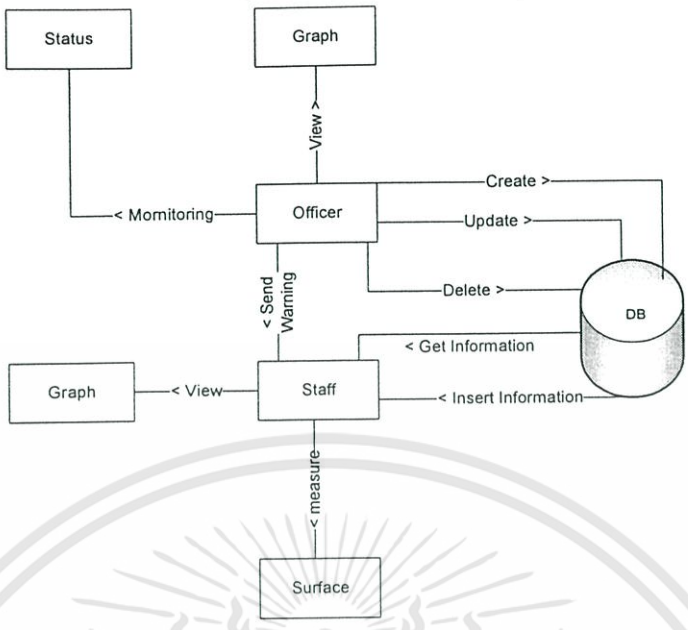
-สามารถประมวลผลการทรุดตัวของดินผ่านหน้าจอจากหน้าจอของ PC server

-สามารถสร้างรายงานการทรุดตัวของดินผ่านหน้าจอ PC server

-สามารถวาดกราฟเปรียบเทียบการทรุดตัวของดินจากการวัดหลายครั้งในผ่านหน้าจอ PC server

โครงสร้างทั่วไปของระบบ

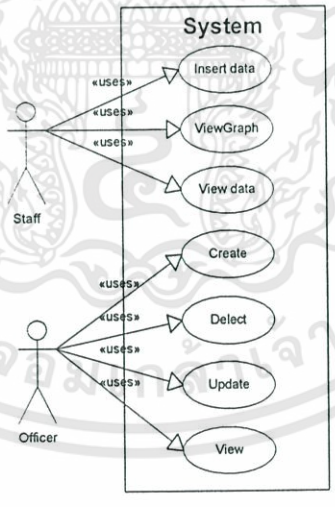
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 32 โดเมนโมเดล (Domain Model)

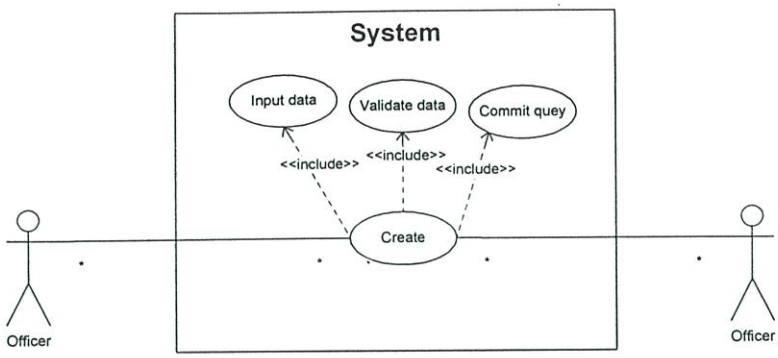
14.2 การออกแบบโครงงาน

ยูสเคสไดอะแกรม (Use case diagram)

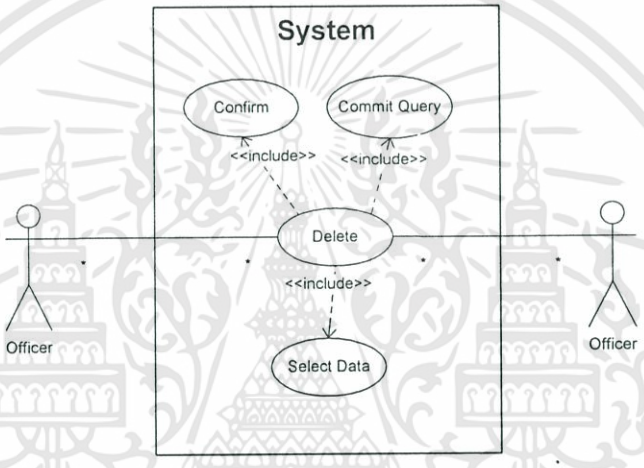


รูปที่ 33 ยูสเคสไดอะแกรมสำหรับการใช้ข้อมูลของ Staff (Survey Man) และ พนังงานประจำศูนย์ปฏิบัติการ

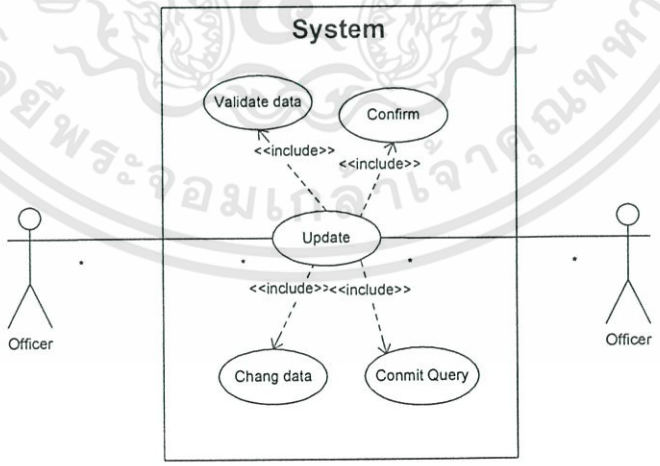
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 34 ยูสเคสไดอะแกรมสำหรับการสร้าง site station ของ พนังงานประจำศูนย์ปฏิบัติการ

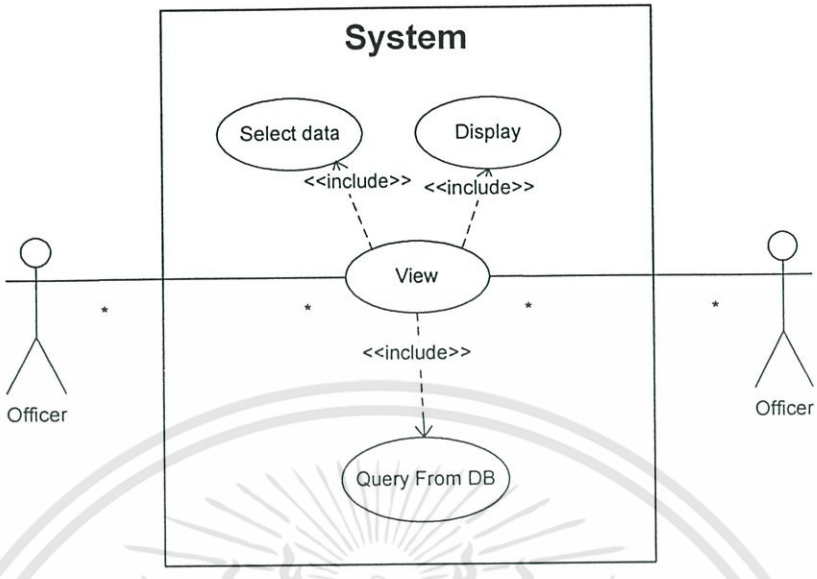


รูปที่ 35 ยูสเคสไดอะแกรมสำหรับการลบค่าภายในของ Site station ของ พนังงานประจำศูนย์ปฏิบัติการ

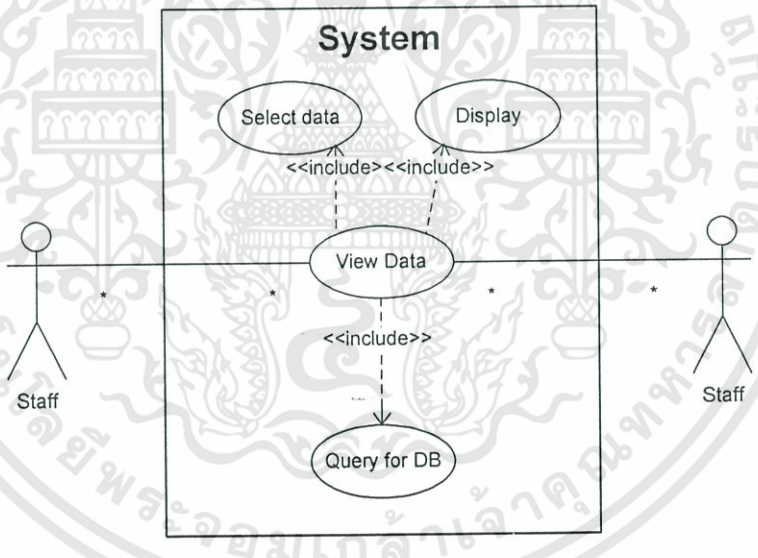


รูปที่ 36 ยูสเคสไดอะแกรมสำหรับการUpdate ค่าภายในของ Site station ของ พนังงานประจำศูนย์ปฏิบัติการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

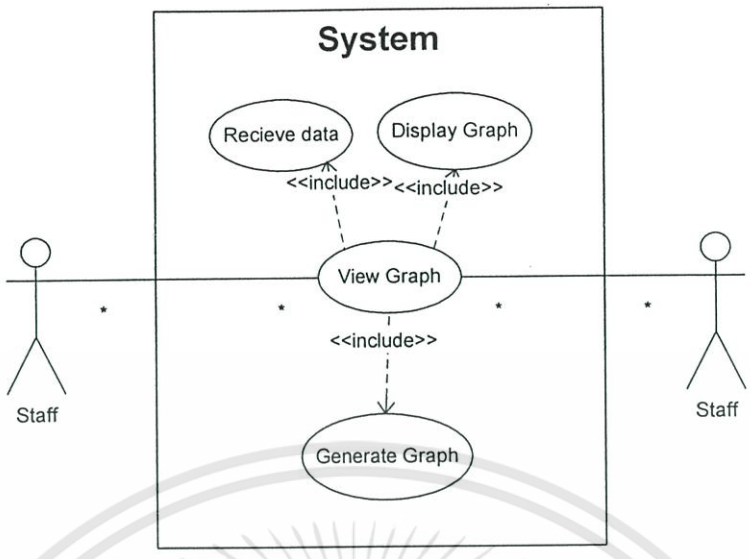


รูปที่ 37 ยูสเคสไดอะแกรมสำหรับการเรียกดูค่า Site station ของ พนังงานประจำศูนย์ปฏิบัติการ

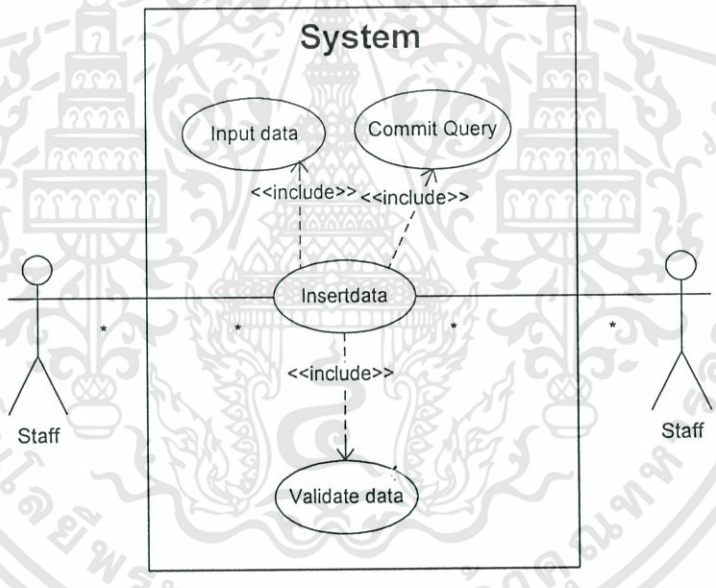


รูปที่ 38 ยูสเคสไดอะแกรมสำหรับการเรียกดูค่าภายในของ Site station ของ staff (Survey Man)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



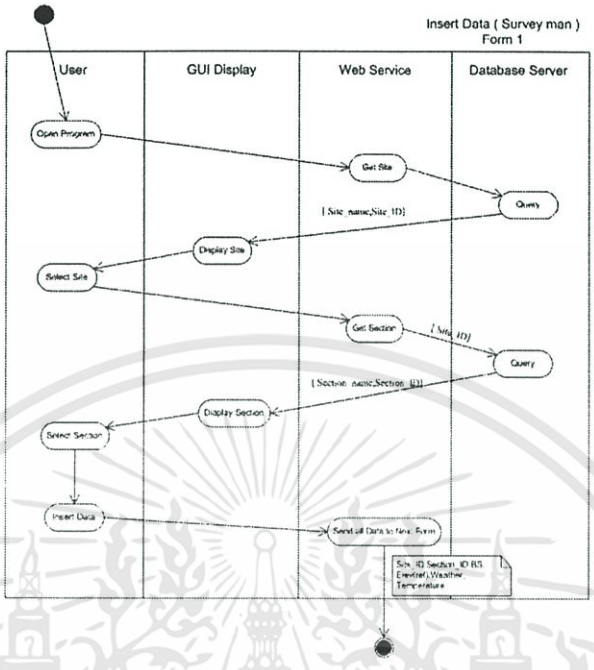
รูปที่ 39 ยูสเคสไดอะแกรมสำหรับการเรียกดูกราฟข้อมูล Settlement ของ staff (Survey Man)



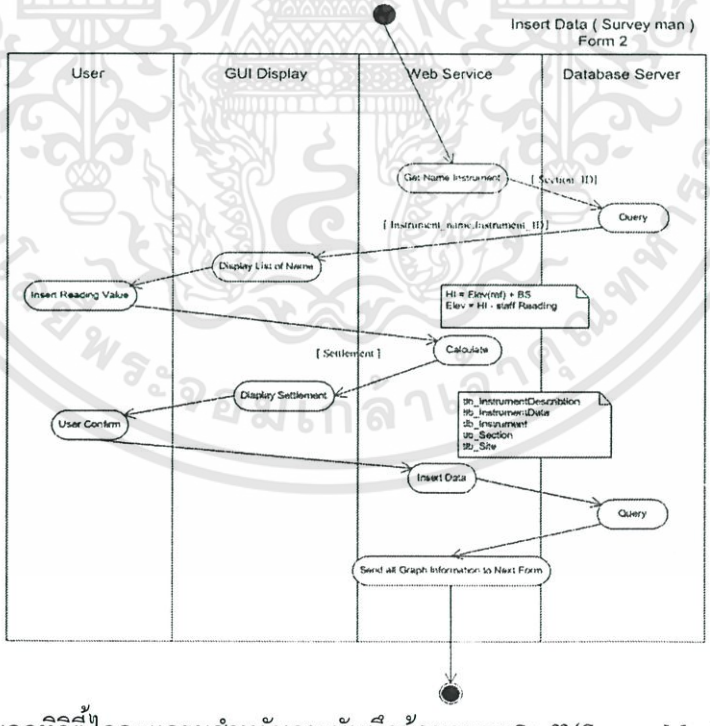
รูปที่ 40 ยูสเคสไดอะแกรมสำหรับการใส่ข้อมูลของ Staff (Survey Man)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอคทิวิตี้ไดอะแกรม (Activity diagram)

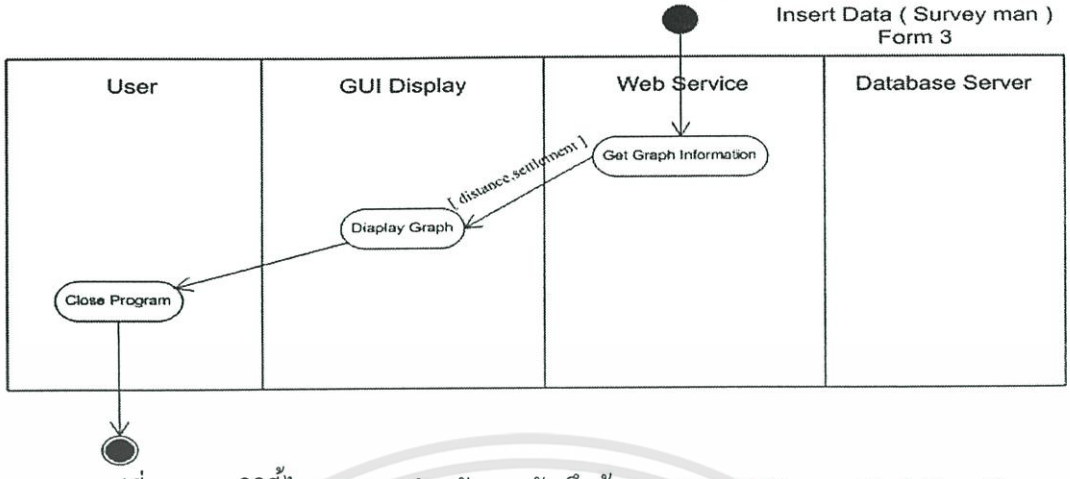


รูปที่ 41 แอคทิวิตี้ไดอะแกรมสำหรับการบันทึกข้อมูลของ Staff (Survey Man)

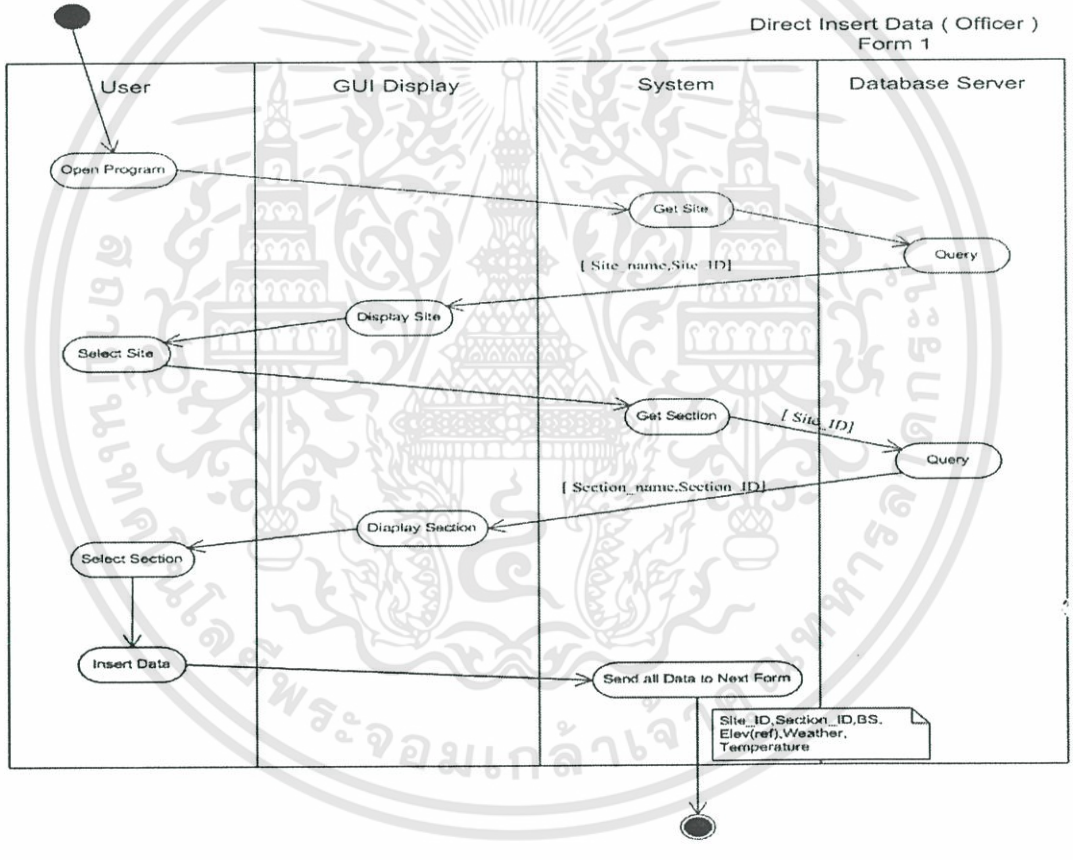


รูปที่ 42 แอคทิวิตี้ไดอะแกรมสำหรับการบันทึกข้อมูลของ Staff (Survey Man) (Form2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

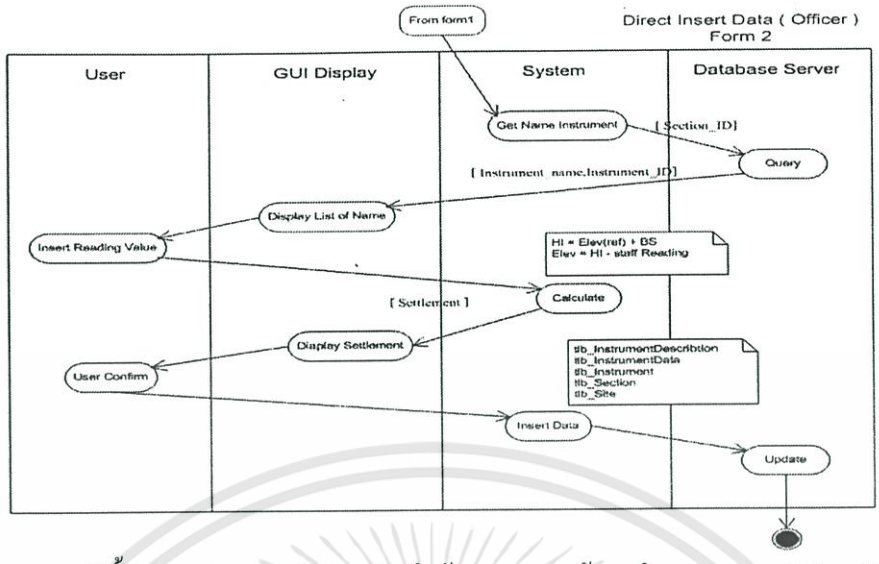


รูปที่ 43 แอคทิวิตี้ไดอะแกรมสำหรับการบันทึกข้อมูลของ staff (Survey Man) (Form3)

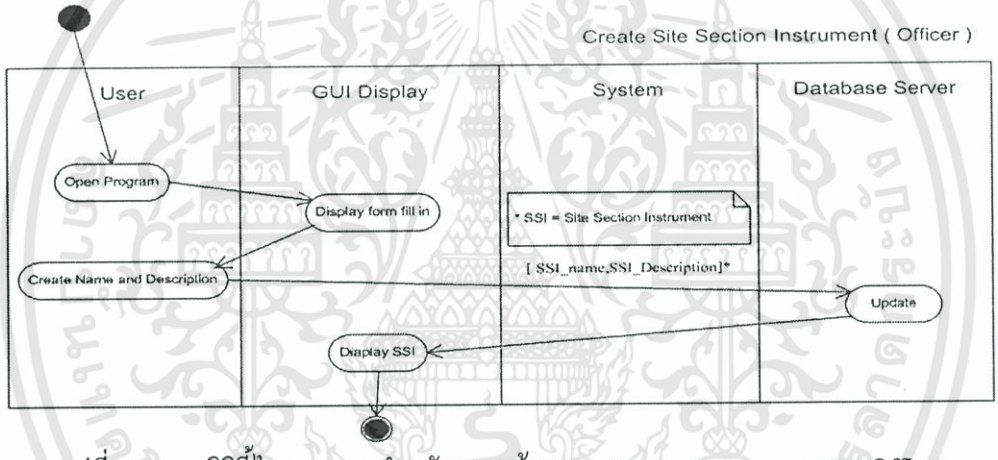


รูปที่ 44 แอคทิวิตี้ไดอะแกรมสำหรับการบันทึกข้อมูลลงฐานข้อมูลโดยตรงของ Officer (Form1)

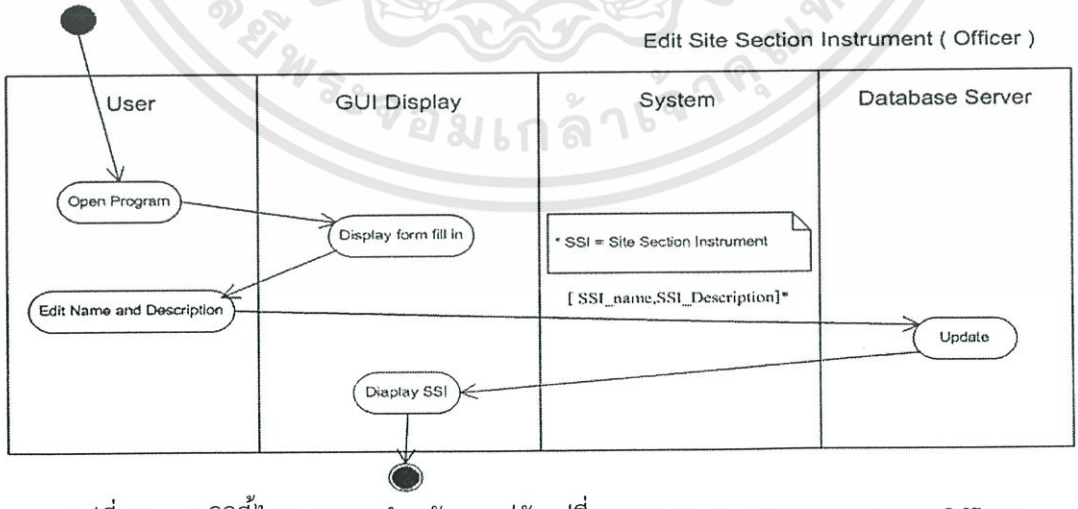
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 45 แอคทิวิตี้ไดอะแกรมสำหรับการบันทึกข้อมูลลงฐานข้อมูลโดยตรงของ Officer (Form2)

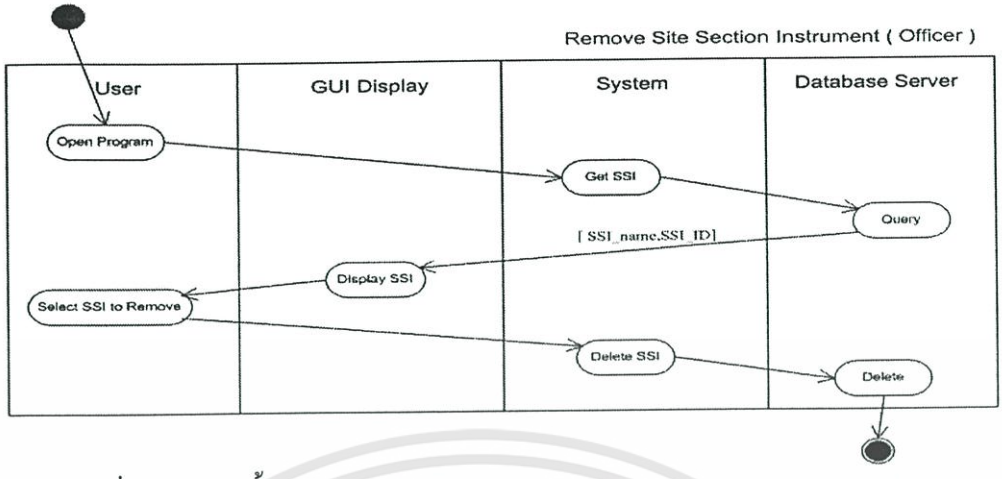


รูปที่ 46 แอคทิวิตี้ไดอะแกรมสำหรับการสร้าง Site Section Instrument ของ Officer

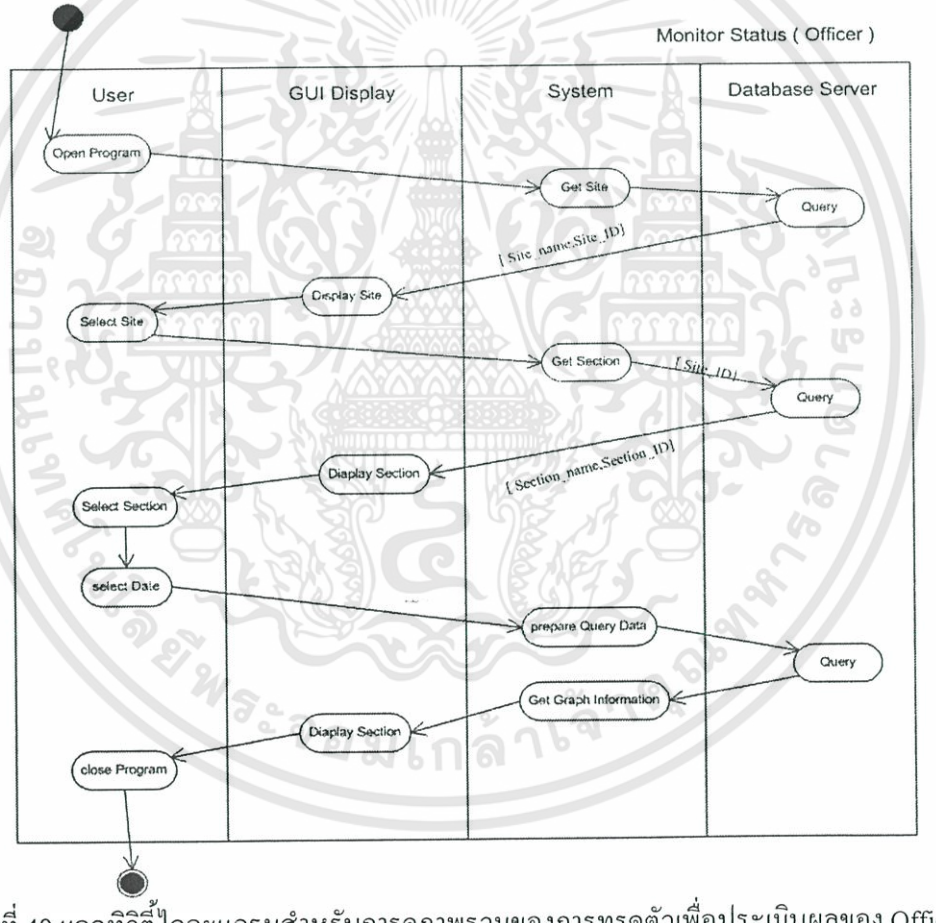


รูปที่ 47 แอคทิวิตี้ไดอะแกรมสำหรับการปรับเปลี่ยน Site Section Instrument ของ Officer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

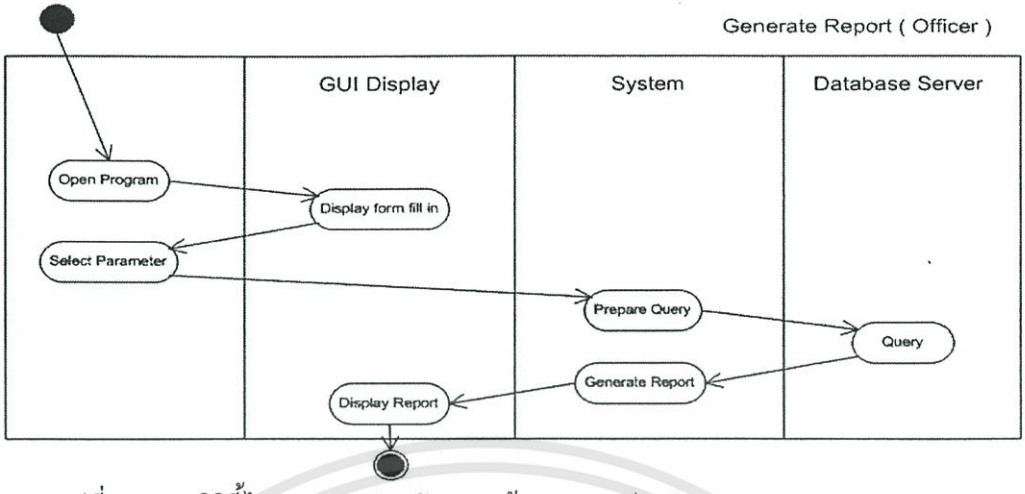


รูปที่ 48 แอคทิวิตี้ไดอะแกรมสำหรับการลบ Site Section Instrument ของ Officer



รูปที่ 49 แอคทิวิตี้ไดอะแกรมสำหรับการดูภาพรวมของการทรุดตัวเพื่อประเมินผลของ Officer

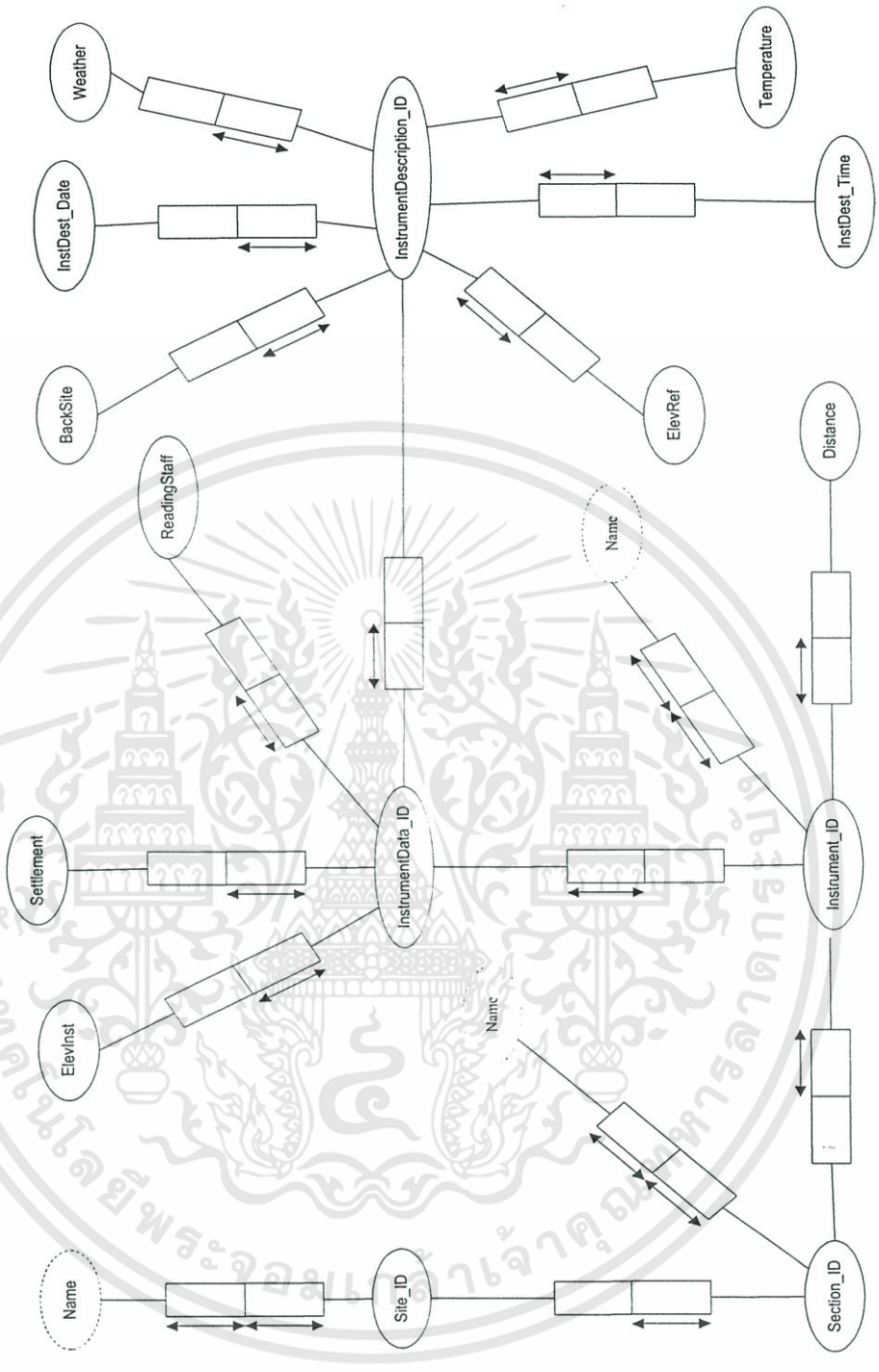
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 50 แอคทิวิตี้ไดอะแกรมสำหรับการสร้างรายงานประจำการตรวจสอบของ Officer

การออกแบบฐานข้อมูล (Database Design)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 ไนแอม โมเดล (Niam Model) ของระบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2 ตารางข้อมูลเครื่องมือ(Instrument Data)

Attribute	Data type	Null	Key	Meaning
InstrumentData_ID	Numeric[9]	ไม่	PK	รหัสของข้อมูลจุดวัด
ReadingStaff	Float[8]	ไม่		ค่าที่ staff อ่านมาจากจุดที่วัด
ElevInst	Float[8]	ไม่		ค่า ELE
Settlement	Float[8]	ไม่		ค่าทรุดตัวของดิน
InstrumentDescription_ID	Varchar[14]	ไม่	FK	รหัสที่อ้างอิงถึงรายละเอียดการวัดในครั้งนั้นๆ
Instrument_ID	Numeric[9]	ไม่	FK	รหัสอ้างอิงถึงที่จุดวัด

ตารางที่ 3 ตารางรายละเอียดเครื่องมือ (InstrumentDescription)

Attribute	Data type	Null	Key	Meaning
InstrumentDescription_ID	Varchar[14]	ไม่	PK	รหัสรายละเอียดการวัด
InstDest_Date	Varchar[10]	ไม่		วันที่
InstDest_Time	Varchar[8]	ไม่		เวลา
BackSite	Float[8]	ไม่		ค่า BS
ElevRef	Float[8]	ไม่		ค่า ELE อ้างอิง
Temperature	Float[8]			ค่าอุณหภูมิ
Weather	Nvarchar[50]			สภาพภูมิอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4 ตารางเครื่องมือ (Instrument)

Attribute	Data type	Null	Key	Meaning
Instrument_ID	Numeric[9]	ไม่	PK	รหัสประจำจุดวัด
Name	Nvarchar[50]	ไม่		ชื่อจุดวัด
Section_ID	Numeric[9]	ไม่	FK	รหัสอ้างอิงถึง section ที่วัด
Distance	Float[8]	ไม่		ระยะห่างระหว่างจุดวัด

ตารางที่ 5 ตารางส่วนย่อย (Section)

Attribute	Data type	Null	Key	Meaning
Section_ID	Numeric[9]	ไม่	PK	รหัสของ section
Name	Nvarchar[50]	ไม่		ชื่อของ section
Site_ID	Numeric[9]	ไม่	FK	รหัสอ้างอิงของ site

ตารางที่ 6 ตารางข้อมูล Site (Site)

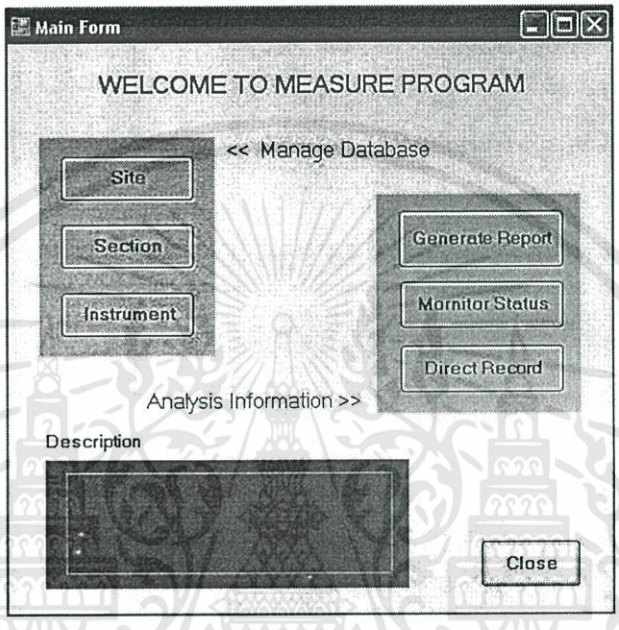
Attribute	Data type	Null	Key	Meaning
Site_ID	Numeric[9]	ไม่	PK	รหัสของ site
Name	Nvarchar[50]	ไม่		ชื่อของ site

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. ผลการทดสอบการทำงาน

ผลการทดสอบนั้นได้มาจากการรันโปรแกรมเพื่อสร้างSiteงานในการขุดอุโมงค์ การบันทึกค่าลงในฐานข้อมูล และการแสดงผลเป็นกราฟเพื่อดูค่าการทรุดตัวของดินเนื่องมาจากการขุดอุโมงค์

โดยมีลักษณะของแอพพลิเคชั่นด้าน Server ดังนี้



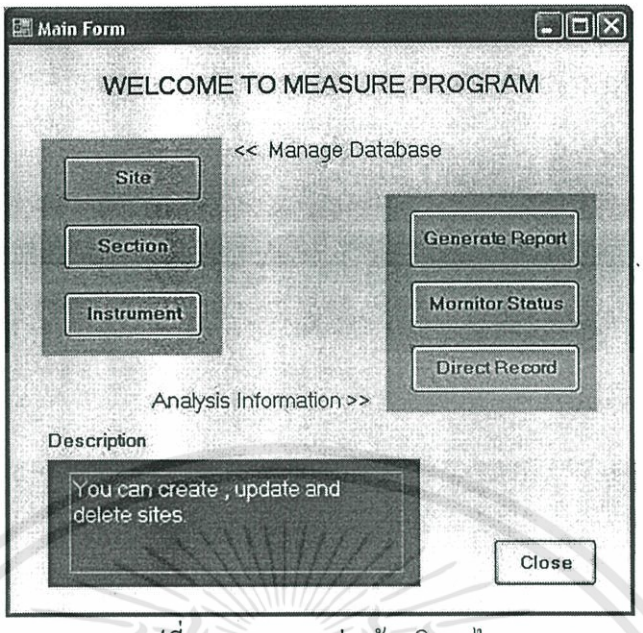
รูปที่ 4.1 หน้าแรกโปรแกรมด้านServer

15.1 ส่วนการทำงานของการเปลี่ยนแปลง Site งานการขุดอุโมงค์

การเปลี่ยนแปลง Site งานนั้นจะทำโดยพนักงานที่ศูนย์ปฏิบัติการเท่านั้นไม่สามารถสร้างจากที่ pocket PC phone ได้โดยมีขั้นตอนดังนี้

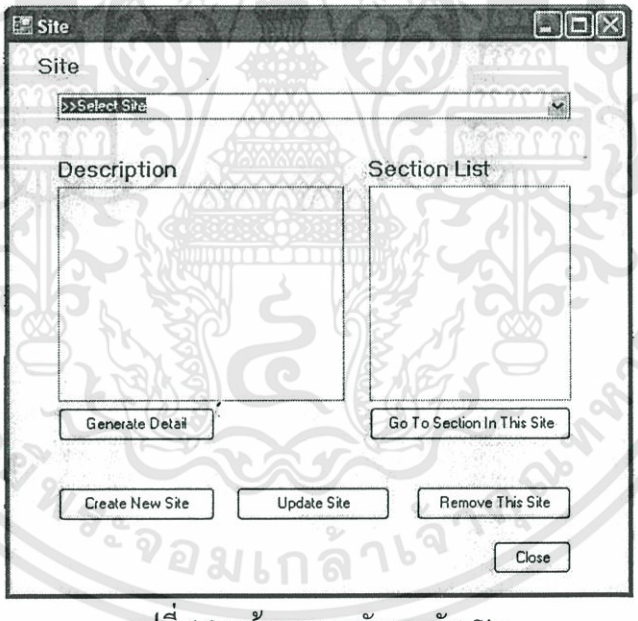
-เมื่อเลื่อนเมาส์ไปที่ปุ่ม จะมีคำอธิบายขึ้นด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



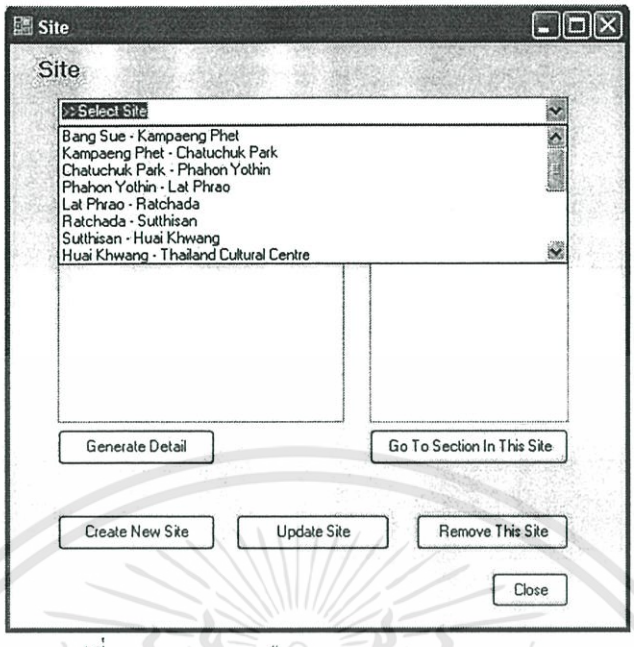
รูปที่ 4.2 แสดงการจ่อเม้าบริเวณไอคอน Site

-และเมื่อคลิ๊กเข้าไป จะเป็นหน้าต่างเพื่อเลือกรายละเอียด



รูปที่ 4.3 หน้าแรกการจัดการกับ Site

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



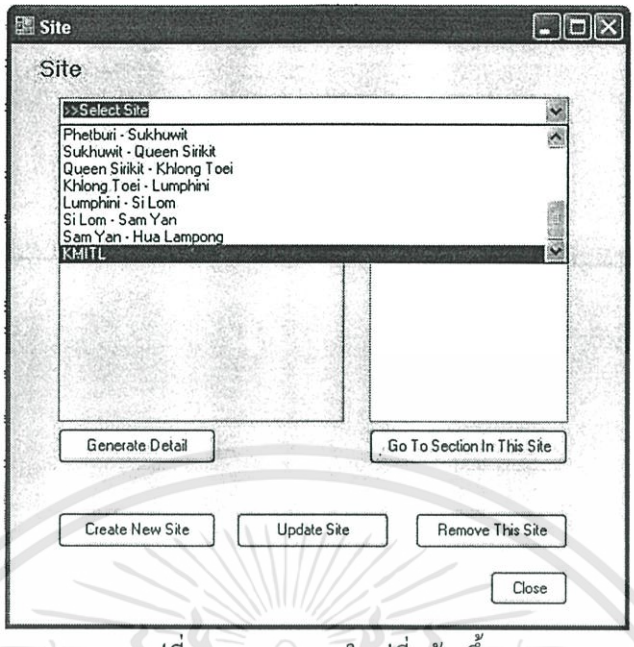
รูปที่ 4.4 แสดงการเลือก Site จาก Drop down List

จะมีฟังก์ชั่นการใช้งานดังนี้

- Generate Detail คือ การเพิ่มคำอธิบายรายละเอียดของ Site งานนั้น
- Go To Section In This Site คือ การเลือกที่จะไปยังรายละเอียดของ section ใน site งานนั้น
- Create New Site คือ การสร้าง site งานใหม่ที่เกิดขึ้นเพื่อวัดการทรุดตัวเมื่อขุดอุโมงค์

รูปที่ 4.5 แสดงการสร้าง Site งานใหม่ และ คำอธิบาย Site

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

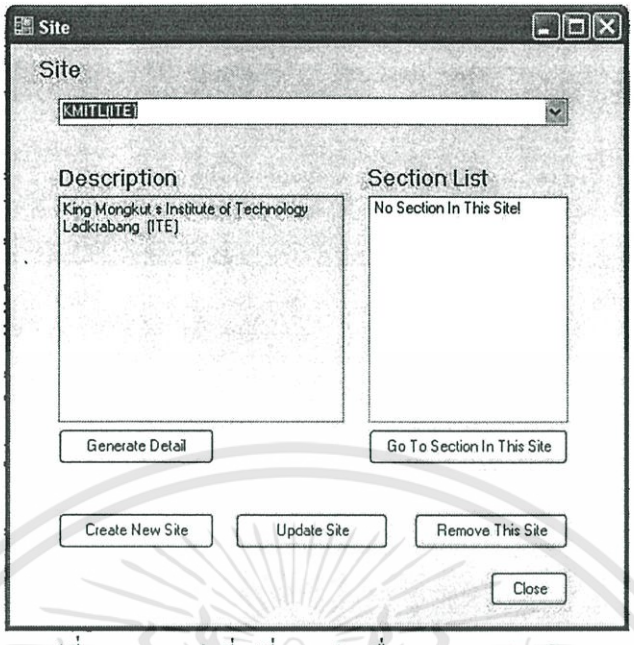


รูปที่ 4.6 แสดง Site ใหม่ที่สร้างขึ้น

- Update Site คือ การปรับเปลี่ยนรายละเอียดของ site งาน

รูปที่ 4.7 แสดงการทำการเปลี่ยนแปลงค่าชื่อ Site และคำอธิบาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

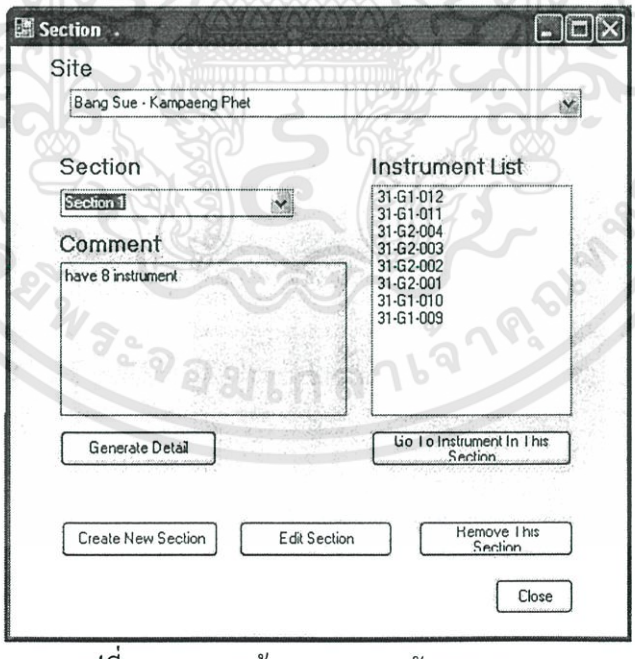


รูปที่ 4.8 แสดงค่าที่เปลี่ยนแปลงชื่อ Site และคำอธิบาย

- Remove This Site คือ การลบรายละเอียดของ site งานนั้น

15.2 ส่วนการทำงานของการเปลี่ยนแปลง Section งานการขุดอุโมงค์

จะเป็นส่วนที่เกี่ยวข้องกับการเปลี่ยนแปลงรายละเอียดของ section ซึ่งจะมีฟังก์ชั่นการที่งานคล้ายคลึงกับการทำงานของ site ดังรูป

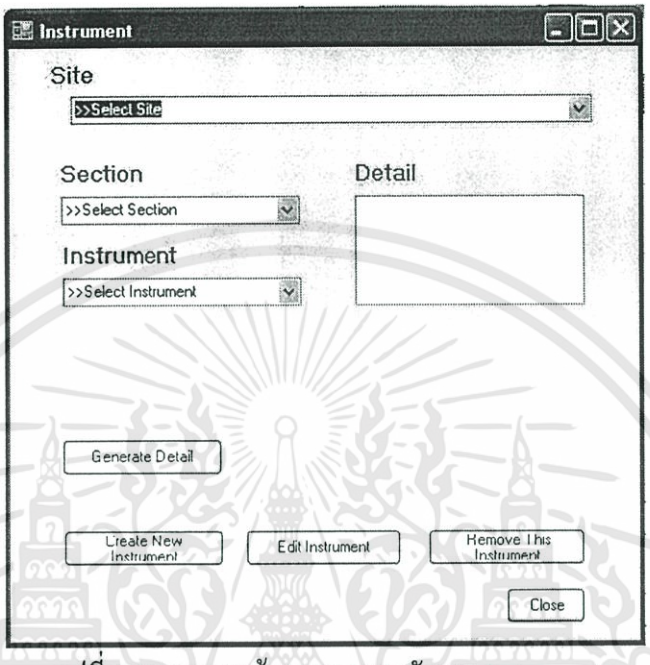


รูปที่ 4.9 แสดงหน้าแรกของการจัดการ Section

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15.3 ส่วนการทำงานของการเปลี่ยนแปลง Instrument งานการขุดอุโมงค์

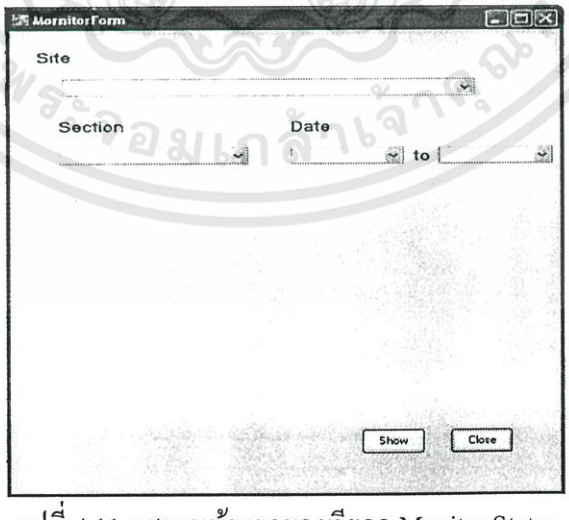
จะเป็นส่วนที่เกี่ยวข้องกับการเปลี่ยนแปลงรายละเอียดของ section ซึ่งจะมีฟังก์ชั่นการที่งานคล้ายคลึงกับการทำงานของ site ดังรูป



รูปที่ 4.10 แสดงหน้าแรกของการจัดการ Instrument

15.4 ส่วนการเรียกดูข้อมูลเพื่อเปรียบเทียบการพังทลายของงานการขุดอุโมงค์ (Monitor Status)

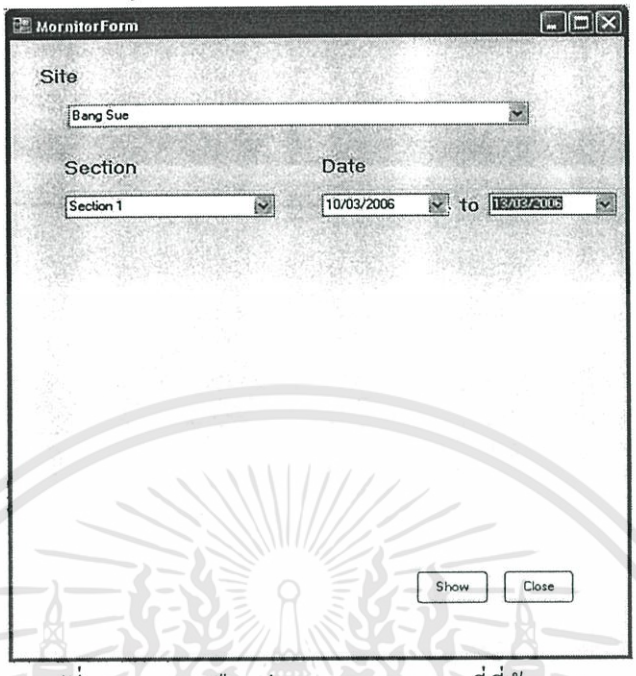
การเรียกดูผลการวัดการพังทลาย เพื่อทำการเปรียบเทียบระดับการทรุดตัวของอุโมงค์จากฐานข้อมูล แล้วจึงทำการประเมินผลความเสียหายของการทำงานของแท่นขุด โดยมีรูปแบบการทำงานดังนี้



รูปที่ 4.11 แสดงหน้าแรกของเรียกดู Monitor Status

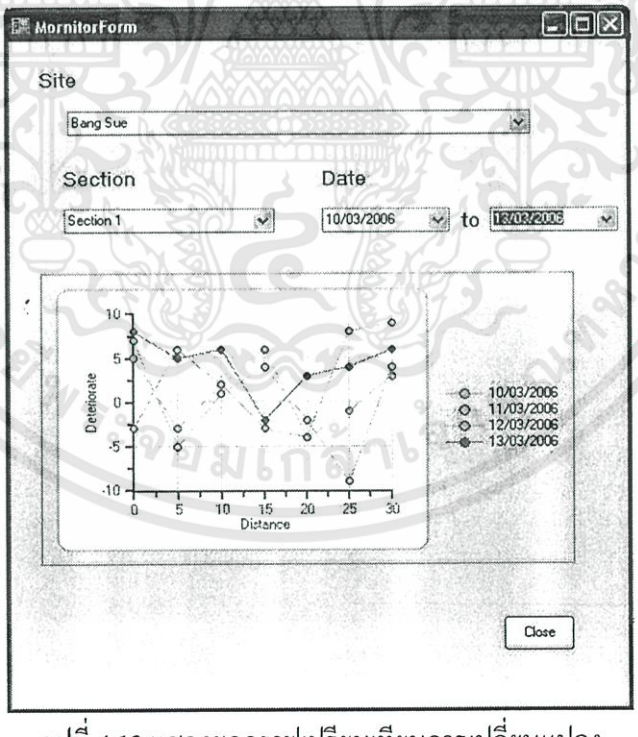
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จากนั้นทำการเลือก ข้อมูลสถานที่ช่วงเวลา ที่ต้องการดูข้อมูล



รูปที่ 4.12 แสดงเลือกช่วงเวลา และ สถานที่ที่ต้องการดู

- แสดงผลออกเป็นกราฟข้อมูลของค่าจากการวัดในช่วงเวลาที่เลือก



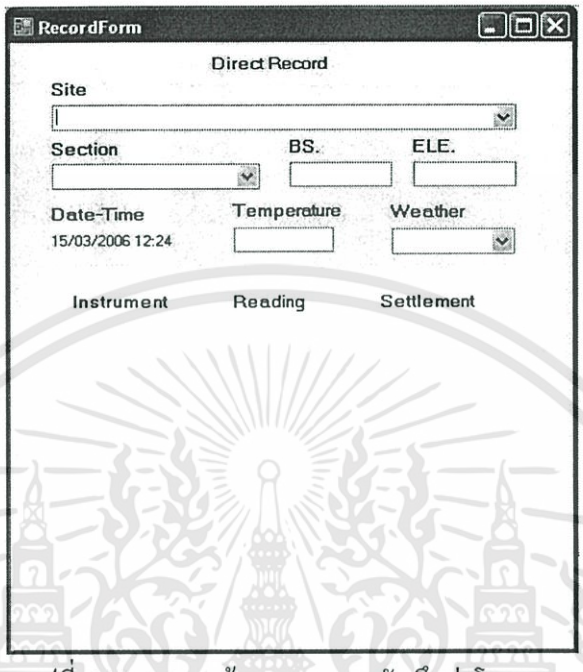
รูปที่ 4.13 แสดงผลกราฟเปรียบเทียบการเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15.5 ส่วนการบันทึกข้อมูลโดยตรงจากศูนย์ปฏิบัติการ (Direct Record)

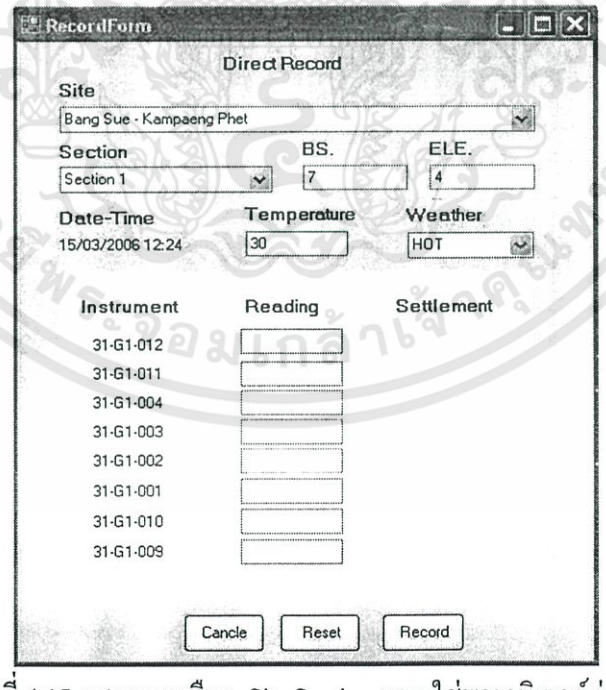
เป็นการบันทึกค่าลงฐานข้อมูลจากศูนย์ปฏิบัติการโดยตรง

- ใส่รายละเอียดต่างๆของ Site



รูปที่ 4.14 แสดงหน้าแรกของการบันทึกค่าโดยตรง

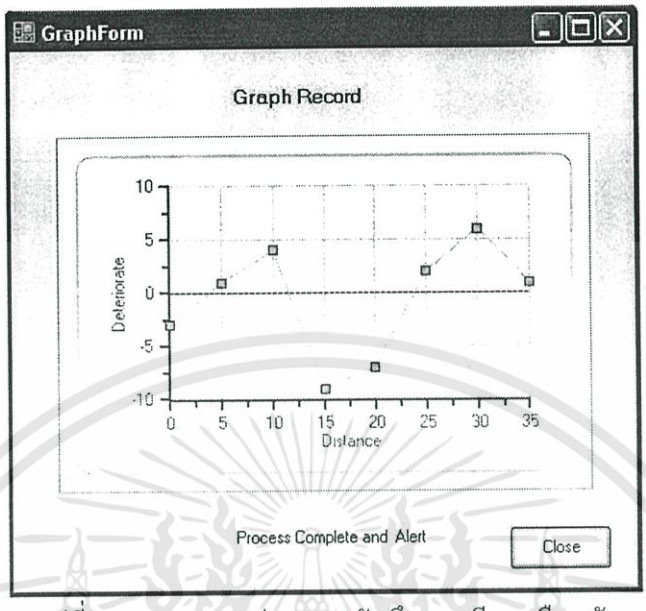
- ใส่ค่าความเสียหายจากการทรุดตัว



รูปที่ 4.15 แสดงการเลือก Site Section และ ใส่พารามิเตอร์ต่างๆ

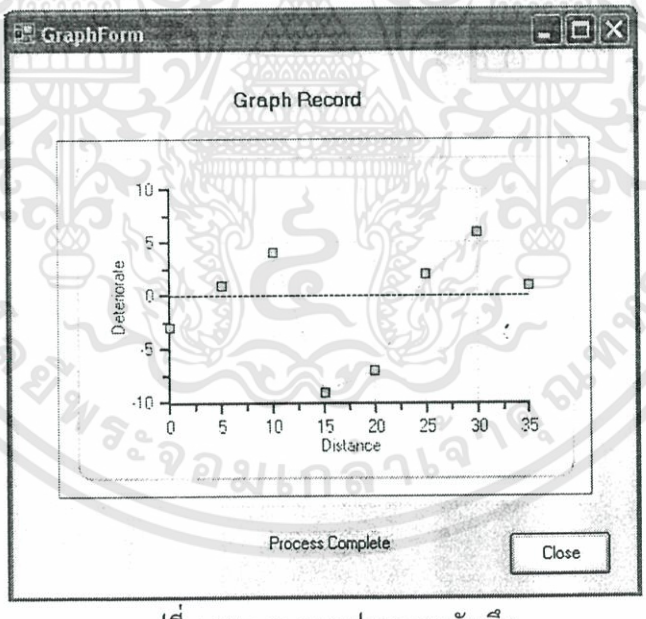
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บันทึกค่าแล้วมีการส่งข้อความเตือนภัย



รูปที่ 4.16 แสดงกราฟจากการบันทึก และมีการเตือนภัย

- บันทึกค่าแล้วไม่มีการส่งข้อความเตือนภัย



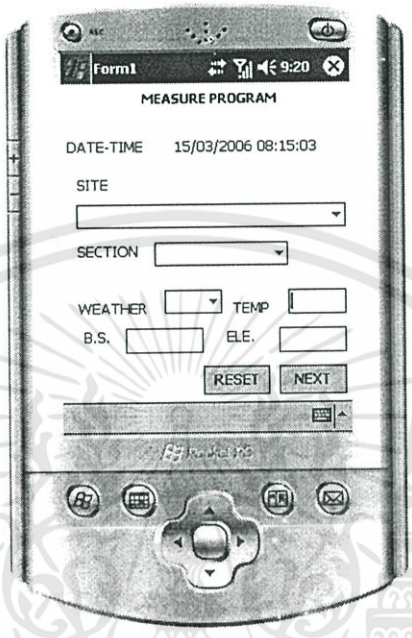
รูปที่ 4.17 แสดงกราฟจากการบันทึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15.6 ส่วนการบันทึกข้อมูลจาก Site งานผ่าน Pocket PC Phone (PDA Record)

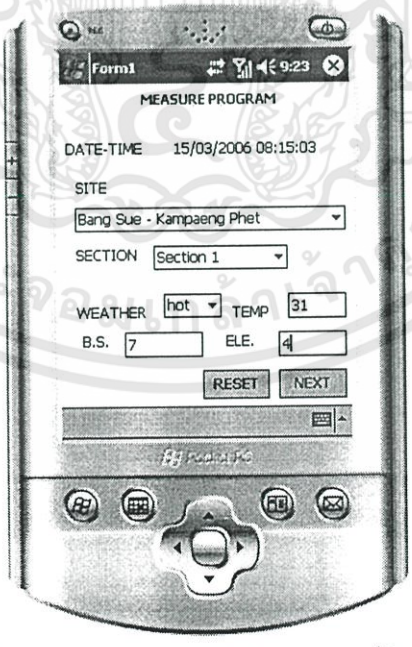
การจดบันทึกค่าจาก Site งานโดย Survey Man ลง Pocket PC Phone ส่งผ่านเว็บเซอร์วิสแล้ว บันทึกลงในฐานข้อมูลที่ศูนย์ปฏิบัติการ

- หน้าหลัก



รูปที่ 4.18 แสดงหน้าแรกของการบันทึกค่าผ่าน Pocket PC Phone

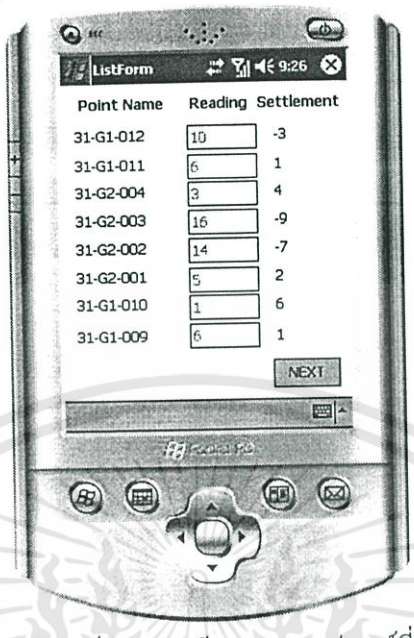
- กรอกค่าพารามิเตอร์ต่างๆ ลงในแบบฟอร์ม



รูปที่ 4.19 แสดงการเลือก Site Section และ ใส่พารามิเตอร์ต่างๆลง Pocket PC Phone

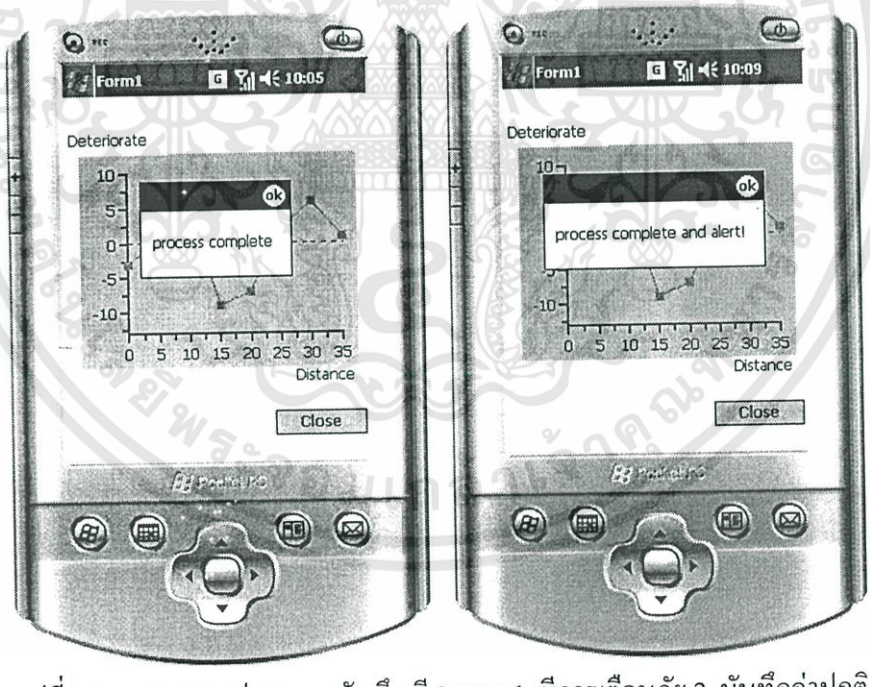
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กรอกค่าของความเสียหายจากอุปกรณ์บริเวณ Site งาน



รูปที่ 4.20 แสดงการใส่พารามิเตอร์ค่าความเสียหายจากอุปกรณ์ที่ Site ลง Pocket PC Phone

- แสดงผลกราฟมี 2 ชนิดมีการเตือนภัยและแบบปกติ



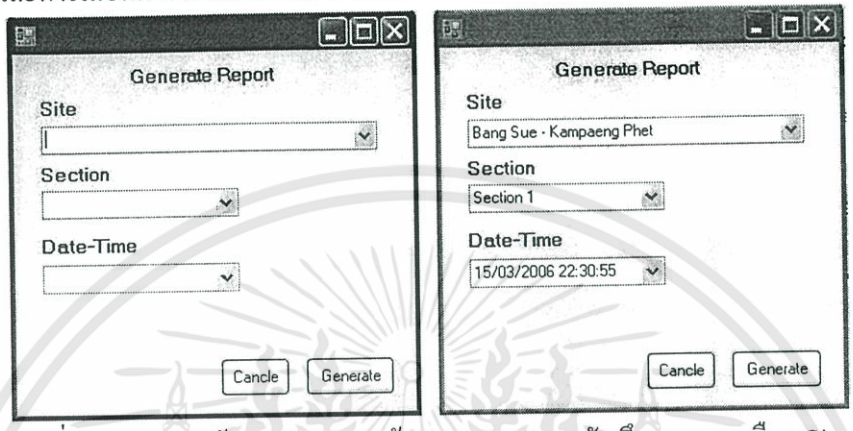
รูปที่ 4.21 แสดงกราฟจากการบันทึก มี 2 แบบ 1. มีการเตือนภัย 2. บันทึกค่าปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15.7 ส่วนการบันทึกข้อมูลจาก Site งานผ่าน Pocket PC Phone (PDA Record)

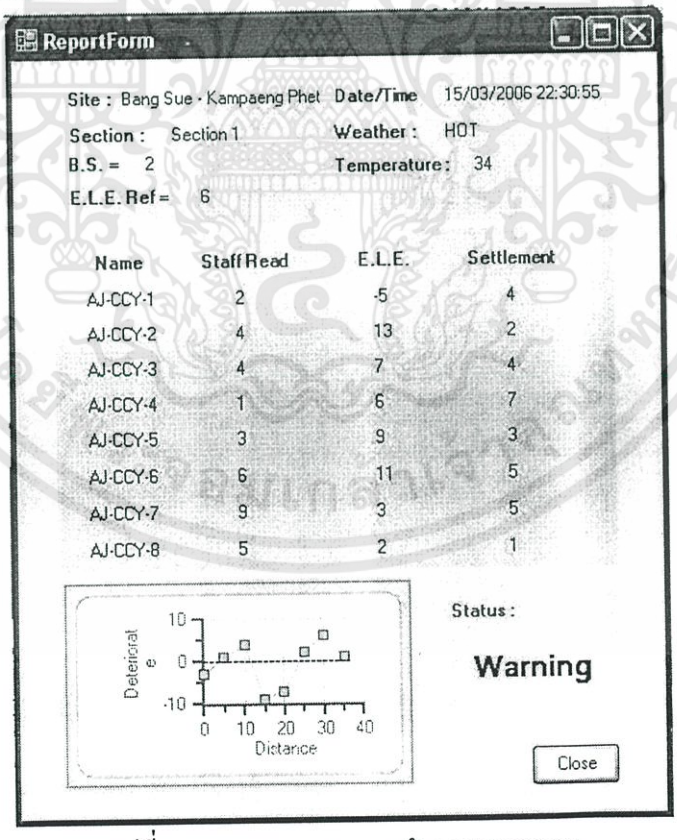
การจดบันทึกค่าจาก Site งานโดย Survey Man ลง Pocket PC Phone ส่งผ่านเว็บเซอร์วิสแล้วบันทึกลงในฐานข้อมูลที่ศูนย์ปฏิบัติการ

- หน้าแรก และการเลือกค่าวันที่ต้องการทำรายงาน



รูปที่ 4.22 แสดงหน้าแรกของการสร้างรายงานผลการบันทึก และการเลือก Site

- หน้ารายงานผลการตรวจสอบ



รูปที่ 4.23 แสดงรายงานแระจำการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16. แผนการถ่ายทอดเทคโนโลยีหรือผลการวิจัยสู่กลุ่มเป้าหมาย

ระบบ Real-Time เพื่อตรวจวัดความเสียหายที่เกิดจากการก่อสร้างขนาดใหญ่ ที่จะพัฒนาขึ้นนี้จะถูกนำไปใช้จริงในโครงการรถไฟฟ้าใต้ดินส่วนต่อขยายและอุโมงค์สาธารณูปโภคอื่นที่สำคัญ เช่น อุโมงค์ระบายน้ำเพื่อป้องกันน้ำท่วม อุโมงค์ประปา และอุโมงค์สำหรับสายส่งกระแสไฟฟ้า ซึ่งเป็นความร่วมมือระหว่างคณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบังกับหน่วยงานรัฐบาลและเอกชน เพื่อใช้ในกระบวนการตรวจสอบและรายงานผล อีกทั้งระบบต้นแบบนี้ยังสามารถช่วยให้ประหยัดงบประมาณประเทศได้จำนวนมาก เพราะไม่ต้องนำเข้าเทคโนโลยีที่มีค่าใช้จ่ายสูงจากต่างประเทศ ผลของงานวิจัยยังมีประโยชน์ต่อการเรียนการสอนในห้องเรียนและแก่วิศวกร โดยจะสามารถช่วยให้วิศวกรสามารถตรวจสอบความเสียหายโดยใช้อุปกรณ์ไร้สายและเว็บไซต์ ที่สามารถเข้าถึงได้ทุกที่ทุกเวลา ซึ่งจะสามารถช่วยให้ประเมินความเสียหายได้ทันที (Real-time) โดยจะช่วยให้เกิดความปลอดภัยต่อชีวิตและทรัพย์สินได้อย่างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17. เอกสารอ้างอิง

Clough, G. W. and Schmidt, B., Design and Performance of Excavations and Tunnels in Soft Clay, Soft Clay Engineering, Elsevier, 1981, pp. 569-634.

Cording, E. J. et al., Displacement around Tunnels in Soil, Report to US. Department of Transportation prepared at the University of Illinois at Urbana-Champaign, DoT-TST-76T-22, 1976.

Hudson, J. A. et al., Understanding Ground Movements caused by Tunneling, Proc. of Conf. on Underground Engineering, Paper No. D21G11, London, 1976.

Maidl, B., Herrenknecht, M., and Anheuser, L., Mechanised Shield Tunneling, Ernst & Sohn, Berlin, 1996.

Mair, R. J. and Taylor, R. N., Prediction of Clay Behavior around Tunnels using Plastic Solution, Proc. Worth Memorial Symposium, Oxford, Thomas Telford, 1992, pp. 449-463.

Peck, R. B., Deep Excavations and Tunneling in Soft Ground, Proc. 7th Int. Conf. on Soil Mechanics and Foundation Engineering, Mexico City, State of the Art Volume, 1969, pp. 225-290.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T084505

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้