

ภาคผนวก ง

การเขียนโปรแกรมและ Source Code ของโปรแกรม KU-Bridge Abutment 1.0

1. โปรแกรม Visual C#

โปรแกรม Visual C# ประกอบด้วย 4 ส่วนหลัก คือ (1) การเขียนโปรแกรม (2) ทิศทางการทำงาน (3) ข้อผิดพลาดและการแก้ไข และ (4) การใช้งานร่วมกับ Microsoft Office โดยมีรายละเอียดดังนี้

1.1 การเขียนโปรแกรมด้วย Visual C#

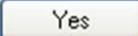
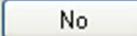
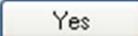
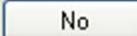
การเขียนโปรแกรมเพื่อสร้างแอปพลิเคชันเพื่อทำงานบน Windows เป็นลักษณะ Event Drive Programming ซึ่งจะเหมาะสมกับโปรแกรมที่มีความซับซ้อน หลากหลาย มีการทำงานแบบเลือกทิศทางได้ โดยหลักการนี้จะคิดว่าเมื่อผู้ใช้งานคลิกปุ่มหนึ่ง (เมื่อมีทางเลือกที่มากกว่าหนึ่งปุ่ม) การทำงานของโปรแกรมก็จะมุ่งไปที่เหตุการณ์นั้น โดยโปรแกรมจะไม่สนใจเหตุการณ์ในส่วนอื่นๆ

การเริ่มต้นเขียนโปรแกรมจากการออกแบบหน้าต่างและการทำงานของโปรแกรม จะทำงานบน Form Designer โดยการนำเอาคำสั่งในส่วนต่างๆ มาวางบนฟอร์ม ตามที่ออกแบบไว้ดังแสดงในภาพผนวกที่ 1 ซึ่งสามารถปรับแต่งคุณสมบัติต่างๆ ทั้งขนาดและรูปร่างตามต้องการได้

หลังจากการออกแบบและปรับแต่งหน้าต่างของโปรแกรมแล้วจะเข้าสู่การเขียนโค้ด โดยดับเบิลคลิกคอนโทรลในเหตุการณ์นั้น จะมีหน้าต่าง Code Editor เข้ามาพร้อมทั้งมีโค้ดบางส่วนมาให้และทำการเขียนโค้ดเพิ่มเติมเพื่อให้โปรแกรมมีการทำงานในเหตุการณ์ส่วนนั้น ดังแสดงในภาพผนวกที่ 2 หลังจากเสร็จสิ้นจะทำการทดสอบการทำงานของโปรแกรมโดยคลิกปุ่ม  หรือกด F5 โปรแกรมที่เขียนไว้จะปรากฏขึ้นให้ทดสอบใช้งานดังแสดงในภาพผนวกที่ 3 เมื่อเสร็จสิ้นก็บันทึกการทำงาน

Message Box ใช้ในการตอบโต้กับผู้ใช้โปรแกรมเพื่อเตือนข้อผิดพลาดที่เกิดขึ้นจากการใช้งานโปรแกรม ซึ่งรูปแบบต่างๆ ถึง 21 รูปแบบ โดย Code Editor จะช่วยแนะนำในขั้นตอนการเขียนโปรแกรม ในส่วนนี้จะทำให้การใช้งานโปรแกรมมีความผิดพลาดน้อยลง มีรายละเอียดของปุ่มดังแสดงในตารางผนวกที่ 1 และ ไอคอนที่ใช้งานควบคู่กับปุ่มดังแสดงในตารางผนวกที่ 2

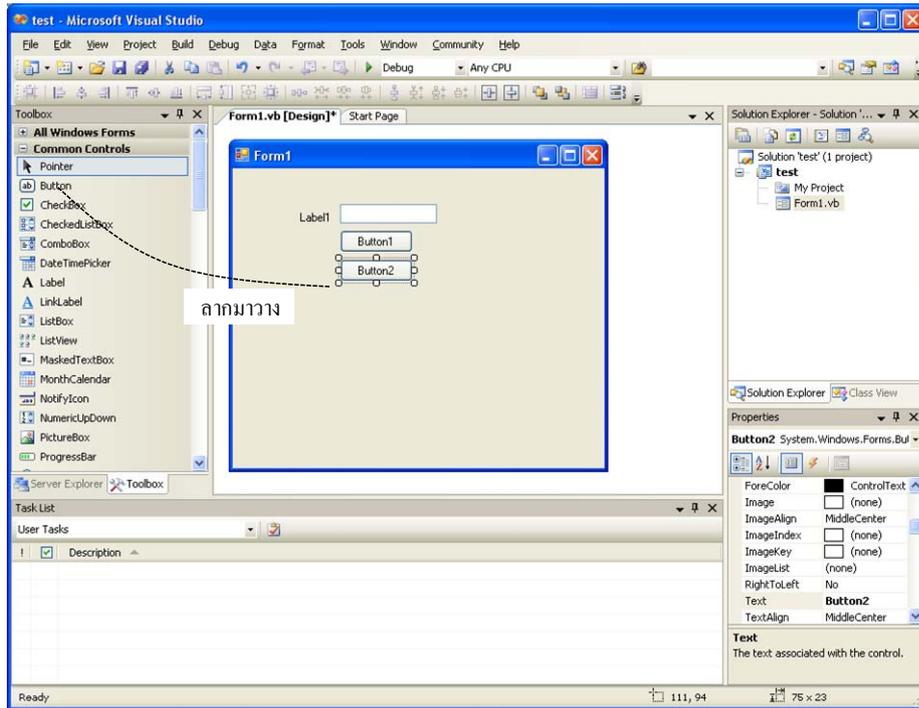
ตารางผนวกที่ ๑1 ปุ่ม Message Box (สัจจะ, 2552)

การเรียกใช้งาน	ปุ่มที่ปรากฏ
Message Box Buttons. Abort Retry Ignore	  
Message Box Buttons. OK	
Message Box Buttons. OK Cancel	 
Message Box Buttons. Retry Cancel	 
Message Box Buttons. Yes No	 
Message Box Buttons. Yes No Cancel	  

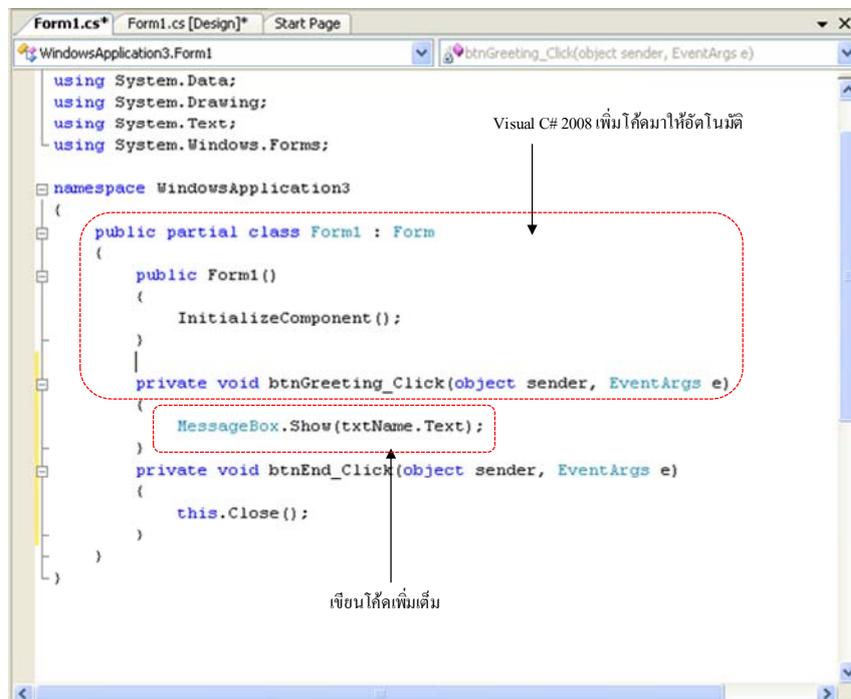
ตารางที่ 5.2 ไอคอน Message Box (สัจจะ, 2552)

การเรียกใช้งาน	ไอคอนที่ปรากฏ
Message Box Icon. Asterisk	
Message Box Icon. Error	
Message Box Icon. Exclamation	
Message Box Icon. Hand	
Message Box Icon. Information	
Message Box Icon. None	ไม่แสดงไอคอนใดๆ
Message Box Icon. Question	
Message Box Icon. Stop	
Message Box Icon. Warning	

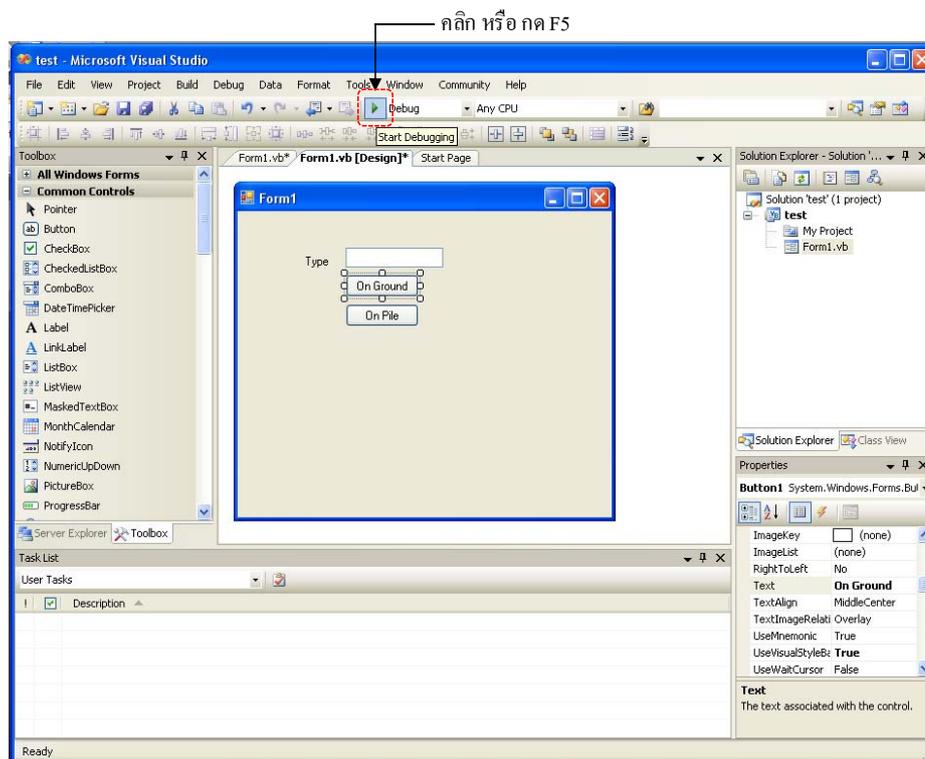
โครงการปรับปรุงการออกแบบและมาตรฐานโครงสร้างโปรแกรมการทดสอบตัวบริเวณคอสะพาน (โครงการวิจัยต่อเนื่อง)



ภาพผนวกที่ 11 ลากคำสั่งมาวางบนฟอร์ม



ภาพผนวกที่ 12 เขียนคำสั่งการทำงานของโปรแกรม



ภาพผนวกที่ 3 ทดสอบการทำงานของโปรแกรม

1.2 ทิศทางการทำงานของโปรแกรม

การควบคุมทิศทางการทำงานของโปรแกรม หรือที่เรียกว่า Control Flow เป็นส่วนที่เลียนแบบการทำงานของมนุษย์ คือต้องเลือกว่าจะทำงานในทิศทางใดโดยใช้การตัดสินใจเข้ามาช่วย ซึ่งจะอาศัย Logical Operator มาพิจารณาทิศทางในการตัดสินใจ มีด้วยกัน 2 รูปแบบ คือ 1) การตัดสินใจ 2) การทำงานแบบวนซ้ำ ดังนี้

1) การตัดสินใจ (Decision) เป็นการเลือกพิจารณาทิศทางหรือรูปแบบที่ต้องการให้โปรแกรมทำงาน โดยพิจารณา 1 ทางเลือกจาก 2 ทางเลือกหรือมากกว่า ซึ่งเหมือนกับการตอบคำถามประเภท Yes/No เช่น ในที่นี้ต้องการเลือกการทำงานของ Approach Slab on Pile หรือ Approach Slab on Ground หรือ การทำงานร่วมกันทั้ง 2 ทางเลือก เป็นต้น

2) การทำงานแบบวนซ้ำ (Iteration) เป็นการทำงานวนซ้ำไปซ้ำมาตามเงื่อนไขที่กำหนดไว้ให้ โปรแกรมทำงาน แบ่งลักษณะการทำงานออกเป็น 2 ส่วน คือ การวนซ้ำที่มีจำนวนรอบที่แน่นอน และการวนซ้ำที่มีจำนวนรอบที่ไม่แน่นอน

1.3 การตรวจสอบและแก้ไขข้อผิดพลาด

ข้อผิดพลาดของโปรแกรมสามารถเกิดขึ้นได้จากการพัฒนาแอปพลิเคชันซึ่งถือเป็นเรื่องปกติ แต่การตรวจสอบและแก้ไขข้อผิดพลาดที่เกิดขึ้น สามารถถูกค้นพบได้โดยความสามารถของเครื่องมือในการพัฒนาแอปพลิเคชันด้วยโปรแกรมภาษา C# แบ่งประเภทความผิดพลาดได้ 3 ส่วนดังนี้

1) Syntax Error เป็นความผิดพลาดเนื่องจากการพิมพ์คำสั่งผิดพลาด เช่น การเรียกใช้งานตัวแปรโดยไม่ประกาศค่า การพิมพ์คำสั่งตกหล่น เป็นต้น ซึ่งถ้าไม่ทำการแก้ไขให้ถูกต้อง โปรแกรมจะไม่สามารถทำงานได้

2) Runtime Error เป็นความผิดพลาดที่เกิดจากสภาวะแวดล้อมในขณะที่ทำงาน เช่น การป้อนค่าที่หารด้วยศูนย์ เป็นต้น จะทำให้โปรแกรมแฮก แต่ใน Visual Studio 2008 จะรายงานและแนะนำวิธีในการแก้ไข โดยจะแสดงไว้ในหน้าต่าง Exception Assistance

3) Logical Error เป็นความผิดพลาดที่เกิดจากการออกแบบ หรือการสร้างแอปพลิเคชันที่ผิดพลาดแม้ว่าโปรแกรมจะทำงานได้จริง แต่ผลลัพธ์ที่ได้อาจจะผิดพลาดหรือไม่ถูกต้อง ซึ่งเป็นส่วนที่ตรวจพบได้ยาก ผู้เขียนหรือผู้ใช้โปรแกรมจะต้องเป็นผู้ตรวจสอบเอง

การตรวจสอบและแก้ไขข้อผิดพลาด โดย Visual Studio 2008 มีเครื่องมือที่เรียกว่า Debugger สำหรับตรวจสอบเพื่อแก้ไขข้อผิดพลาดต่างๆ ที่เกิดขึ้นในขั้นตอนการเขียนโปรแกรม ซึ่งเมื่อสงสัยว่าสิ่งที่เขียนน่าจะมีข้อผิดพลาด เราสามารถหยุดการทำงานชั่วคราว (Break Point) เพื่อตรวจสอบว่าสิ่งที่เขียนนั้นทำงานได้ถูกต้องหรือไม่โดยเลือกคำสั่ง Debug

1.4 พัฒนาแอปพลิเคชันร่วมกับ Microsoft Office

เนื่องจาก Microsoft Office เป็นชุด Software ที่มีความนิยมมาก จึงมีการพัฒนาแอปพลิเคชัน (เขียนโปรแกรม) ร่วมกับ Office 2007 ด้วย VSTO (Visual Studio Tools for Office) ซึ่งเป็นตัวเชื่อมระหว่างชุด Software ทั้ง 2 ส่วน โดยมีรูปแบบของ VSTO 4 รูปแบบดังนี้

1) Document-Level Customization เป็นการปรับแต่งเอกสารที่สร้างขึ้นจากโปรแกรมต่างๆ ใน Microsoft Office ให้เอกสารเหล่านั้นทำงานได้มากกว่าการจัดเก็บข้อมูล

2) Application-Level Add-ins เป็นการพัฒนาส่วนเสริมการทำงานของโปรแกรมใน Microsoft Office หรือเรียกว่า Plug-in

3) SharePoint Workflow Template เป็นการพัฒนาแอปพลิเคชันที่เน้นการสื่อสารระหว่างกันภายในองค์กร

4) InfoPath form Template เป็นการพัฒนาแอปพลิเคชันร่วมกับ InfoPath ซึ่งใช้ในการจัดการระบบเอกสารในองค์กร

การสร้างแอปพลิเคชันด้วย VSTO ร่วมกับ Office จะทำงานคล้ายๆ กับการเริ่มต้นสร้าง New Project จะปรากฏหน้าต่าง Form Designer ขึ้นมา แต่ในที่นี้จะเป็นการเลือกหน้าต่าง Form Designer ที่อยู่ในรูปของหน้าเอกสาร (Word, Excel และ Power Point) ในชุด Software ของ Microsoft Office

2. การเขียนโปรแกรม และ Source Code

การเขียนโปรแกรมคำนวณการรดน้ำของคอกสะพานนั้น เป็นการเขียนโปรแกรมแบบ Object Oriented Programming (OOP) โดยขั้นตอนที่สำคัญจะเกี่ยวข้องกับลักษณะของเหตุการณ์ที่เกิดขึ้นกับออบเจกต์ต่างๆ ของโปรแกรม ซึ่งสามารถอธิบาย Tab ต่างๆ ในโปรแกรมได้ดังต่อไปนี้

2.1 Tab Soil Profile

1) ปุ่ม Add Ground Water Elevation

```
bool gnd_check = true;

private void bt_add_gnd_we_Click(object sender, EventArgs e)
{
    if (gnd_check == true)
    {
        sheet1.Cells[2, 3] = tb_Ground_Watre_Elevation.Text;
        bt_add_gnd_we.Text = "Edit Data";
        tb_Ground_Watre_Elevation.ReadOnly = true;
        gnd_check = false;
    }
    else
    {
        bt_add_gnd_we.Text = "Update Data";
        tb_Ground_Watre_Elevation.ReadOnly = false;
        gnd_check = true;
    }
}
```

เมื่อมีการ click ที่ Add Ground Water Elevation ทำการตรวจสอบเงื่อนไข

ถ้า `gnd_check == true`

กำหนดข้อความใน sheet1 แถวที่ 2 หลักที่ 3

แสดงข้อความของปุ่มกด `bt_add_gnd_we` เป็น Edit Data

กำหนดให้ `tb_Ground_Watre_Elevation` แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

ถ้า `gnd_check == false`

แสดงข้อความของปุ่มกด `bt_add_gnd_we` เป็น Update Data

กำหนดให้ `tb_Ground_Watre_Elevation` แสดงข้อความและสามารถแก้ไขข้อความได้

2) ปุ่ม Add Layer

```

private void bt_add_Click(object sender, EventArgs e)
{
    try
    {
        Pls_wait.Show();
        Input_Data_tab1();
        Pls_wait.Hide();
        if (layer > 1)
        {
            bt_del_layer.Visible = true;
        }
        else
        {
            bt_del_layer.Visible = false;
        }

        if (layer == 16)
        {
            bt_add_layer.Visible = false;
        }
    }
    catch (Exception ex)
    {
        KillExcel();
    }
}

```

เมื่อมีการ click ที่ Add Layer ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น

ทำการตรวจสอบเงื่อนไข

แสดง pop up ข้อความ Please Wait ที่หน้าจอ

เรียกใช้คำสั่ง Input_Data_tab1()

ยกเลิกการแสดงผลของข้อความ Please Wait ที่หน้าจอ

ถ้า layer > 1

แสดงปุ่มกด bt_del_layer

เงื่อนไขอื่น

ยกเลิกการแสดงผลปุ่มกด bt_del_layer

ถ้า layer == 16

ยกเลิกการแสดงผลปุ่มกด bt_aad_layer

เมื่อเกิดข้อผิดพลาดขึ้น ทำการลบค่าใน Excel

3) ปุ่ม Delete Layer

```
private void bt_del_layer_Click(object sender, EventArgs e)
{
    Pls_wait.Show();

    if (layer > 1)
    {
        row = layer + 6;
        layer--;

        dataGridView1.Rows.RemoveAt(layer-1);

        sheet1.Cells[row, 1] =
        sheet1.Cells[row, 2] =

        sheet1.Cells[row, 4] =
        sheet1.Cells[row, 5] =
        sheet1.Cells[row, 6] =
        sheet1.Cells[row, 7] =
        sheet1.Cells[row, 9] =
        sheet1.Cells[row, 10] =
        sheet1.Cells[row, 11] = "";

        tb_Elevation_From.Text = Convert.ToString(sheet1.Cells[row - 1, 3].Value2);

        dataGridView1.FirstDisplayedScrollingRowIndex = dataGridView1.Rows.Count - 1;

        lb_layer.Text = Convert.ToString(layer);

        if (dataGridView1.RowCount > 0)
        {
            dataGridView1.CurrentCell = dataGridView1[0, dataGridView1.Rows.Count - 1];
        }
    }
    if (layer == 1)
    {
        bt_del_layer.Visible = false;
    }
    Pls_wait.Hide();
}
```

เมื่อมีการ click ที่ Delete Layer จะทำการแสดง pop up ข้อความ Please Wait ที่หน้าจอ แล้วทำการตรวจสอบเงื่อนไข

ถ้า $layer > 1$

ทำการกำหนดค่า $row = layer + 6$,

ลดค่า layer ลงครั้งละ 1,

ลบแถวของ dataGridView1 ที่ตำแหน่ง layer-1,

กำหนดข้อความใน sheet1 แถวที่ row หลักที่ 1,2,4,5,6,7,9,10,11

กำหนดข้อความที่อยู่ใน tb_Elevation_From = ข้อความที่อยู่ใน sheet1 แถวที่ row - 1

หลักที่ 3

แสดงข้อความที่อยู่ใน dataGridView1 และเลื่อน DisplayedScrolling มาที่ Count - 1
กำหนดข้อความของ lb_layer

ถ้า dataGridView1.RowCount > 0
กำหนดค่าปัจจุบันของ dataGridView1

ถ้า layer == 1
ยกเลิกการแสดงผลปุ่มกด bt_del_layer
ยกเลิกการแสดงผลข้อความ Please Wait ที่หน้าจอ

4) ปุ่ม GeoGraph

```

private void bt_GeoGraph_Click(object sender, EventArgs e)
{
    try
    {
        Export_Chart();
        Process process1 = new Process();
        process1.StartInfo.FileName = @"c:\bridge\exe\Output_GeoGraph.exe";
        process1.Start();
    }
    catch (Exception ex)
    {
        KillExcel();
    }
}

```

เมื่อมีการ click ที่ GeoGraph ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น

ทำการตรวจสอบเงื่อนไข

เรียกใช้คำสั่ง Export_Chart()

กำหนดตัวแปร Process ชื่อ process1

กำหนดที่อยู่ของ process1

ทำการเริ่มต้น process1

เมื่อเกิดข้อผิดพลาดขึ้น

ทำการลบค่าใน Excel

2.2 Tab Longitudinal Profile

1) ปุ่ม Add Data

```
bool tab2begin_check = true;

private void bt_add_tab2_begin_Click(object sender, EventArgs e)
{
    try
    {
        Pls_wait.Show();

        if (tab2begin_check == true)
        {
            sheet2.Cells[8, 4] = tb_begin_pvc1.Text;
            sheet2.Cells[9, 4] = tb_ige.Text;
            bt_add_tab2_begin.Text = "Edit";
            tb_begin_pvc1.ReadOnly = true;
            tb_ige.ReadOnly = true;
            tab2begin_check = false;

            tb_st_pvc1.Text = Convert.ToString(sheet2.Cells[20, 5].Value2);
            tb_st_pvi1.Text = Convert.ToString(sheet2.Cells[21, 5].Value2);
            tb_st_pvt1.Text = Convert.ToString(sheet2.Cells[22, 5].Value2);

            tb_st_pvc2.Text = Convert.ToString(sheet2.Cells[24, 5].Value2);
            tb_st_pvi2.Text = Convert.ToString(sheet2.Cells[25, 5].Value2);
            tb_st_pvt2.Text = Convert.ToString(sheet2.Cells[26, 5].Value2);

        }
        else
        {
            bt_add_tab2_begin.Text = "Update";
            tb_begin_pvc1.ReadOnly = false;
            tb_ige.ReadOnly = false;
            tab2begin_check = true;
        }
        Pls_wait.Hide();
    }
    catch (Exception ex)
    {
        KillExcel();
    }
}
```

เมื่อมีการ click ที่ Add Data ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น

ทำการตรวจสอบเงื่อนไข

แสดง pop up ข้อความ Please Wait ที่หน้าจอ

ถ้า tab2begin_check == true

กำหนดข้อความใน sheet2 แถวที่ 8 หลักที่ 4

กำหนดข้อความใน sheet2 แถวที่ 9 หลักที่ 4

แสดงข้อความของปุ่มกด bt_add_tab2_begin เป็น edit

กำหนดให้ tb_begin_pvc1 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_ige แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดข้อความของ tb_st_pvc1

กำหนดข้อความของ tb_st_pvi1

กำหนดข้อความของ tb_st_pvt1

กำหนดข้อความของ tb_st_pvc2

กำหนดข้อความของ tb_st_pvi2

กำหนดข้อความของ tb_st_pvt2

ถ้ามีเงื่อนไขอื่น

แสดงข้อความของปุ่มกด bt_add_tab2_begin เป็น Update

กำหนดให้ tb_begin_pvc1 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_ige แสดงข้อความและสามารถแก้ไขข้อความได้

ยกเลิกการแสดง pop up ข้อความ Please Wait ที่หน้าจอ

เมื่อเกิดข้อผิดพลาดขึ้น

ทำการลบค่าใน Excel

2) ปุ่ม Add Data

```
bool tab2add_check = true;

private void add_data_tab2_Click(object sender, EventArgs e)
{
    Pls_wait.Show();
    if (tab2add_check == true)
    {
        Input_Data_tab2();
        label13.Visible =
        tb_grad_l1.Visible =
        tb_grad_l2.Visible =
        tb_grad_l3.Visible =
        tb_grad_l4.Visible =
        tb_grad_l5.Visible = true;

        add_data_tab2.Text = "Edit";
        tab2add_check = false;

        tb_l1_sub.ReadOnly =
        tb_l2_sub.ReadOnly =
        tb_l3_sub.ReadOnly =
        tb_l4_sub.ReadOnly =
        tb_l5_sub.ReadOnly = true;

        tb_l1_length.ReadOnly =
        tb_l2_length.ReadOnly =
        tb_l3_length.ReadOnly =
        tb_l4_length.ReadOnly =
        tb_l5_length.ReadOnly = true;

        tb_el_pvc1.ReadOnly =
        tb_el_pvi1.ReadOnly =
        tb_el_pvt1.ReadOnly = true;

        tb_el_pvc2.ReadOnly =
        tb_el_pvi2.ReadOnly =
        tb_el_pvt2.ReadOnly = true;
    }
    else
    {
        add_data_tab2.Text = "Update";
        tab2add_check = true;

        tb_l1_sub.ReadOnly =
        tb_l2_sub.ReadOnly =
        tb_l3_sub.ReadOnly =
        tb_l4_sub.ReadOnly =
        tb_l5_sub.ReadOnly = false;

        tb_l1_length.ReadOnly =
        tb_l2_length.ReadOnly =
        tb_l3_length.ReadOnly =
        tb_l4_length.ReadOnly =
        tb_l5_length.ReadOnly = false;

        tb_el_pvc1.ReadOnly =
        tb_el_pvi1.ReadOnly =
        tb_el_pvt1.ReadOnly = false;

        tb_el_pvc2.ReadOnly =
        tb_el_pvi2.ReadOnly =
        tb_el_pvt2.ReadOnly = false;
    }
    Pls_wait.Hide();
}
```

เมื่อมีการ click ที่ Add Data ทำการแสดง pop up ข้อความ Please Wait ที่หน้าจอ แล้วทำการตรวจสอบเงื่อนไข

ถ้า tab2add_check == true

เรียกใช้คำสั่ง Input_Data_tab2

แสดง label13

แสดง tb_grad_11

แสดง tb_grad_12

แสดง tb_grad_13

แสดง tb_grad_14

แสดง tb_grad_15

แสดงข้อความของปุ่มกด add_data_tab2 เป็น Edit

กำหนดให้ tb_11_sub แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_12_sub แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_13_sub แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_14_sub แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_15_sub แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_11_length แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_12_length แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_13_length แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_14_length แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_15_length แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_e1_pvc1 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_e1_pvi1 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_e1_pvt1 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_e1_pvc2 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_e1_pvi2 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_e1_pvt2 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

ถ้ามีเงื่อนไขอื่น

แสดงข้อความของ add_data_tab2 เป็น Update

กำหนดให้ tb_11_sub แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_12_sub แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_13_sub แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_14_sub แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_15_sub แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_11_length แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_12_length แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_13_length แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_14_length แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_15_length แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_el_pvc1 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_el_pvi1 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_el_pvt1 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_el_pvc2 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_el_pvi2 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_el_pvt2 แสดงข้อความและสามารถแก้ไขข้อความได้

ยกเลิกการแสดง pop up ข้อความ Please Wait ที่หน้าจอ

2.3 Tab Embankment Geometry

1) เช็คบล็อกร Type 1

```

private void cb_type1_CheckedChanged(object sender, EventArgs e)
{
    if ((cb_type1.Checked == true) && ((add_data_tab3.Text == "Add Data") || (add_data_tab3.Text == "Update")))
    {
        tb_type1_from.ReadOnly =
        tb_type1_ss.ReadOnly =
        //tb_type1_ssvh.ReadOnly =
        tb_type1_to.ReadOnly =
        tb_type1_u1l.ReadOnly =
        tb_type1_uw.ReadOnly =
        tb_type1_woh.ReadOnly = false;
    }
    else if (cb_type1.Checked == false)
    {
        tb_type1_from.ReadOnly =
        tb_type1_ss.ReadOnly =
        //tb_type1_ssvh.ReadOnly =
        tb_type1_to.ReadOnly =
        tb_type1_u1l.ReadOnly =
        tb_type1_uw.ReadOnly =
        tb_type1_woh.ReadOnly = true;
    }
}

```

เมื่อ Type 1 ถูกเลือกจะทำการตรวจสอบเงื่อนไข

ถ้า (cb_type1.Checked == true)&&((add_data_tab3.Text == "Add Data") || (add_data_tab3.Text == "Update"))

กำหนดให้ tb_type1_from แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_ss แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_to แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_u1l แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_uw แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_woh แสดงข้อความและสามารถแก้ไขข้อความได้

ถ้า cb_type1.Checked == false

กำหนดให้ tb_type1_from แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_ss แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_to แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_u1l แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_uw แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type1_woh แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

2) เช็คบล็อกรูป Type 2

```

private void cb_type2_CheckedChanged(object sender, EventArgs e)
{
    if ((cb_type2.Checked == true) && ((add_data_tab3.Text == "Add Data") || (add_data_tab3.Text == "Update")))
    {
        tb_type2_from.ReadOnly =
        tb_type2_ss.ReadOnly =
        //tb_type2_ssvh.ReadOnly =
        tb_type2_to.ReadOnly =
        tb_type2_u11.ReadOnly =
        tb_type2_uw.ReadOnly =
        tb_type2_woh.ReadOnly = false;
    }
    else if (cb_type2.Checked == false)
    {
        tb_type2_from.ReadOnly =
        tb_type2_ss.ReadOnly =
        //tb_type2_ssvh.ReadOnly =
        tb_type2_to.ReadOnly =
        tb_type2_u11.ReadOnly =
        tb_type2_uw.ReadOnly =
        tb_type2_woh.ReadOnly = true;
    }
}

```

เมื่อ Type 2 ถูกเลือกจะทำการตรวจสอบเงื่อนไข

ถ้า (cb_type2.Checked == true)&&((add_data_tab3.Text == "Add Data") ||
(add_data_tab3.Text == "Update"))

กำหนดให้ tb_type2_from แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_ss แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_to แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_u11 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_uw แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_woh แสดงข้อความและสามารถแก้ไขข้อความได้

ถ้า cb_type2.Checked == false

กำหนดให้ tb_type2_from แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_ss แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_to แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_u11 แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_uw แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

กำหนดให้ tb_type2_woh แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

3) ปุ่ม Add Data

```
bool tab3add_check = true;

private void add_data_tab3_Click(object sender, EventArgs e)
{
    try
    {
        if (tab3add_check == true)
        {
            Pls_wait.Show();
            Input_Data_tab3();
            Pls_wait.Hide();
            add_data_tab3.Text = "Edit";
            if (cb_type1.Checked == true)
            {
                tb_type1_from.ReadOnly =
                tb_type1_to.ReadOnly =
                tb_type1_who.ReadOnly =
                tb_type1_ss.ReadOnly =
                tb_type1_uw.ReadOnly =
                tb_type1_u11.ReadOnly = true;
            }
            if (cb_type2.Checked == true)
            {
                tb_type2_from.ReadOnly =
                tb_type2_to.ReadOnly =
                tb_type2_who.ReadOnly =
                tb_type2_ss.ReadOnly =
                tb_type2_uw.ReadOnly =
                tb_type2_u11.ReadOnly = true;
            }
            tab3add_check = false;
        }
        else
        {
            add_data_tab3.Text = "Update";
            if (cb_type1.Checked == true)
            {
                tb_type1_from.ReadOnly =
                tb_type1_to.ReadOnly =
                tb_type1_who.ReadOnly =
                tb_type1_ss.ReadOnly =
                tb_type1_uw.ReadOnly =
                tb_type1_u11.ReadOnly = false;
            }
            if (cb_type2.Checked == true)
            {
                tb_type2_from.ReadOnly =
                tb_type2_to.ReadOnly =
                tb_type2_who.ReadOnly =
                tb_type2_ss.ReadOnly =
                tb_type2_uw.ReadOnly =
                tb_type2_u11.ReadOnly = false;
            }
            tab3add_check = true;
        }
    }
    catch (Exception ex)
    {
        KillExcel();
    }
}
```

เมื่อมีการ click ที่ Add Data ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น

ทำการตรวจสอบเงื่อนไข

ถ้า tab3add_check == true

 แสดง pop up ข้อความ Please Wait ที่หน้าจอ

 เรียกใช้คำสั่ง Input_Data_tab3

 ยกเลิกการแสดง pop up ข้อความ Please Wait ที่หน้าจอ

 แสดงข้อความของปุ่มกด add_data_tab3 เป็น Edit

ถ้า cb_type1.Checked == true

 กำหนดให้ tb_type1_from แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type1_to แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type1_woh แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type1_ss แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type1_uw แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type1_ull แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

ถ้า cb_type2.Checked == true

 กำหนดให้ tb_type2_from แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type2_to แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type2_woh แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type2_ss แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type2_uw แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

 กำหนดให้ tb_type2_ull แสดงข้อความโดยที่ไม่สามารถแก้ไขข้อความได้

ถ้ามีเงื่อนไขอื่น

 แสดงข้อความของปุ่มกด add_data_tab3 เป็น Update

ถ้า `cb_type1.Checked == true`

กำหนดให้ `tb_type1_from` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type1_to` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type1_woh` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type1_ss` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type1_uw` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type1_ull` แสดงข้อความและสามารถแก้ไขข้อความได้

ถ้า `cb_type2.Checked == true`

กำหนดให้ `tb_type2_from` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type2_to` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type2_woh` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type2_ss` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type2_uw` แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ `tb_type2_ull` แสดงข้อความและสามารถแก้ไขข้อความได้

เมื่อเกิดข้อผิดพลาดขึ้น

ลบค่าใน Excel

2.4 Tab Criteria

1) ปุ่ม Select

```
private void bt_Select_Analyze_Click(object sender, EventArgs e)
{
    try
    {

        MemoryStream ms1 = new MemoryStream();
        if (rb_oneway.Checked == true)
        {
            sheet4.Cells[21, pp] = 1;
        }
        else if (rb_twoway.Checked == true)
        {
            sheet4.Cells[21, pp] = 2;
        }

        double temp_sheet1 = sheet6.Cells[20, ii].Value2;
        double temp_sheet2 = sheet6.Cells[26, ii].Value2;

        sheet6.Cells[20, ff].Value2 = temp_sheet1;
        sheet6.Cells[26, ff].Value2 = temp_sheet2;

        /*

        Path_Initial_Profile_Graph = @"c:\bridge\pic\Initial_Profile_Graph" + Convert.ToString(numberPIC) + ".bmp";
        Path_Initial_End_Prof_Graph = @"c:\bridge\pic\Initial_End_Prof_Graph" + Convert.ToString(numberPIC) + ".bmp";

        Export_Initial_Profile_Graph();
        Image img1 = Image.FromFile(@"c:\bridge\pic\Initial_Profile_Graph" + Convert.ToString(numberPIC) + ".bmp");
        img1.Save(ms1, System.Drawing.Imaging.ImageFormat.Bmp);
        img1 = null;
        output_int_end.SizeMode = PictureBoxSizeMode.StretchImage;
        output_int_end.Image = Image.FromStream(ms1);
        ms1.Dispose();
        ms1 = new MemoryStream();

        img1 = Image.FromFile(@"c:\bridge\pic\Initial_End_Prof_Graph" + Convert.ToString(numberPIC) + ".bmp");
        img1.Save(ms1, System.Drawing.Imaging.ImageFormat.Bmp);
        img1 = null;
        Output_Initial_End_Graph.SizeMode = PictureBoxSizeMode.StretchImage;
        Output_Initial_End_Graph.Image = Image.FromStream(ms1);
        ms1.Dispose();
        ms1 = new MemoryStream();

        //MessageBox.Show(Convert.ToString(numberPIC));
        numberPIC++;
        */

        tb_ay_el_pvi1.ReadOnly = false;
        tb_ay_el_pvt1.ReadOnly = false;

        tb_ay_el_pvc2.ReadOnly = false;
        tb_ay_el_pvi2.ReadOnly = false;

        tb_ay_st_pvc1.Text = Convert.ToString(sheet6.Cells[20, ee].Value2);
        tb_ay_st_pvi1.Text = Convert.ToString(sheet6.Cells[21, ee].Value2);
        tb_ay_st_pvt1.Text = Convert.ToString(sheet6.Cells[22, ee].Value2);
        tb_ay_st_pvc2.Text = Convert.ToString(sheet6.Cells[24, ee].Value2);
        tb_ay_st_pvi2.Text = Convert.ToString(sheet6.Cells[24, ee].Value2);
        tb_ay_st_pvt2.Text = Convert.ToString(sheet6.Cells[26, ee].Value2);

        tb_ay_el_pvc1.Text = Convert.ToString(sheet6.Cells[20, ff].Value2);
        tb_ay_el_pvt2.Text = Convert.ToString(sheet6.Cells[26, ff].Value2);

        //File.Delete(@"c:\bridge\pic\Initial_Profile_Graph" + Convert.ToString(numberPIC - 1) + ".bmp");
        //File.Delete(@"c:\bridge\pic\Initial_Profile_Graph" + Convert.ToString(numberPIC - 1) + ".bmp");
    }
    catch (Exception ex)
    {
        //string temp = Convert.ToString(ex);
        //KillExcel();
        //MessageBox.Show(Convert.ToString(numberPIC-1));
        //MessageBox.Show(ex.Message);
    }
    tabControl1.SelectedTab = tabPage1;
}
}
```

เมื่อมีการ click ที่ Select_ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น
ทำการตรวจสอบเงื่อนไข

ถ้า rb_oneway.Checked == true

กำหนดให้ sheet4 แถวที่ 21 หลักที่ pp = 1

ถ้า rb_twoway.Checked == true

กำหนดให้ sheet4 แถวที่ 21 หลักที่ pp = 2

กำหนดให้ tb_ay_el_pvi1 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_ay_el_pvt1 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_ay_el_pvc2 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดให้ tb_ay_el_pvi2 แสดงข้อความและสามารถแก้ไขข้อความได้

กำหนดข้อความใน tb_ay_st_pvc1

กำหนดข้อความใน tb_ay_st_pvi1

กำหนดข้อความใน tb_ay_st_pvt1

กำหนดข้อความใน tb_ay_st_pvc2

กำหนดข้อความใน tb_ay_st_pvi2

กำหนดข้อความใน tb_ay_st_pvt2

กำหนดข้อความใน tb_ay_el_pvc1

กำหนดข้อความใน tb_ay_el_pvt2

แสดง tabControl1

2.5 Analysis

1) ปุ่ม Add Data (Step 1)

```
private void bt_add_step3_ay_Click(object sender, EventArgs e)
{
    try
    {
        sheet6.Cells[21, ff] = tb_ay_el_pvi1.Text;
        sheet6.Cells[22, ff] = tb_ay_el_pvt1.Text;
        sheet6.Cells[24, ff] = tb_ay_el_pvc2.Text;
        sheet6.Cells[25, ff] = tb_ay_el_pvi2.Text;

        Path_Initial_Profile_Graph = @"c:\bridge\pic\Initial_Profile_Graph" + Convert.ToString(numberPIC) + ".bmp";
        Path_Initial_End_Prof_Graph = @"c:\bridge\pic\Initial_End_Prof_Graph" + Convert.ToString(numberPIC) + ".bmp";

        MemoryStream ms1 = new MemoryStream();
        Export_Initial_Profile_Graph();
        Image img1 = Image.FromFile(Path_Initial_Profile_Graph);
        img1.Save(ms1, System.Drawing.Imaging.ImageFormat.Bmp);
        img1 = null;
        output_int_end.SizeMode = PictureBoxSizeMode.StretchImage;
        output_int_end.Image = Image.FromStream(ms1);
        ms1.Dispose();
        ms1 = new MemoryStream();

        img1 = Image.FromFile(Path_Initial_End_Prof_Graph);
        img1.Save(ms1, System.Drawing.Imaging.ImageFormat.Bmp);
        img1 = null;
        Output_Initial_End_Graph.SizeMode = PictureBoxSizeMode.StretchImage;
        Output_Initial_End_Graph.Image = Image.FromStream(ms1);
        ms1.Dispose();
        ms1 = new MemoryStream();

        numberPIC++;

        File.Delete(@"c:\bridge\pic\Initial_Profile_Graph" + Convert.ToString(numberPIC - 1) + ".bmp");
        File.Delete(@"c:\bridge\pic\Initial_End_Prof_Graph" + Convert.ToString(numberPIC - 1) + ".bmp");
    }
    catch (Exception ex)
    {
        string temp = Convert.ToString(ex);
        KillExcel();
    }
}
```

เมื่อมีการ click ที่ Add Data (Step 1) ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น

ทำการตรวจสอบเงื่อนไข

กำหนด sheet6 แถวที่ 21 หลักที่ ff มีค่าเท่ากับข้อความของ tb_ay_el_pvi1

กำหนด sheet6 แถวที่ 22 หลักที่ ff มีค่าเท่ากับข้อความของ tb_ay_el_pvt1

กำหนด sheet6 แถวที่ 24 หลักที่ ff มีค่าเท่ากับข้อความของ tb_ay_el_pvc2

กำหนด sheet6 แถวที่ 25 หลักที่ ff มีค่าเท่ากับข้อความของ tb_ay_el_pvi2

กำหนดที่อยู่ของ Path_Initial_Profile_Graph

กำหนดที่อยู่ของ Path_Initial_End_Prof_Graph

ประกาศตัวแปร MemoryStream ชื่อ ms1
เรียกใช้คำสั่ง Export_Initial_Profile_Graph()
ประกาศตัวแปร Image ชื่อ img1
กำหนดภาพ img1
กำหนดที่บันทึกของ img1
กำหนดขนาดของ output_int_end
กำหนดภาพของ output_int_end

ประกาศตัวแปร ms1 ใหม่
กำหนดภาพ img1
กำหนดที่บันทึกของ img1
กำหนดขนาดของ Output_Initial_End_Graph
กำหนดภาพของ Output_Initial_End_Graph

เมื่อเกิดข้อผิดพลาดขึ้น

กำหนดข้อความให้กับตัวแปร temp
ทำการลบค่าใน Excel

2) ปุ่ม Analysis (Step 2)

```
private void Analyze_next_tab_Click(object sender, EventArgs e)
{
    try
    {
        Run_Analyze();
        tabControl3.SelectedTab = tabPage8;
    }
    catch (Exception ex)
    {
        string temp = Convert.ToString(ex);
        KillExcel();
    }
}
```

เมื่อมีการ click ที่ Analysis (Step 2) ทำการตรวจสอบและจัดการข้อผิดพลาดที่เกิดขึ้นในรูปแบบ Exception Handling โดยกำหนดจุดเริ่มต้น

ทำการตรวจสอบเงื่อนไข

เรียกใช้คำสั่ง Run_Analyze()

กำหนดการแสดงผลของ tabControl3

เมื่อเกิดข้อผิดพลาดขึ้น

กำหนดข้อความให้กับตัวแปร temp

ทำการลบค่าใน Excel
