

บทที่ 5

การออกแบบการแสดงผลและการพัฒนาระบบ

การออกแบบการแสดงผล

ในการพัฒนาโปรแกรมจัดรูปแบบเอกสารวิทยานิพนธ์/การค้นคว้าแบบอิสระของนักศึกษาระดับบัณฑิตศึกษา มหาวิทยาลัยเชียงใหม่ ผู้ศึกษาได้ออกแบบระบบเพื่อลดความซับซ้อน และง่ายต่อการใช้งานและการพัฒนา โดยแบ่งออกได้ดังนี้

5.1 การออกแบบจอภาพบนเว็บเพจ สำหรับนักศึกษา

- 5.1.1 การออกแบบจอภาพเข้าสู่ระบบ
- 5.1.2 การออกแบบจอภาพเมนูหลักของนักศึกษา
- 5.1.3 การออกแบบจอภาพในส่วนของโปรแกรมจัดรูปแบบเอกสาร
- 5.1.4 การออกแบบจอภาพเมนูการอัปโหลดและดาวน์โหลดไฟล์เอกสาร

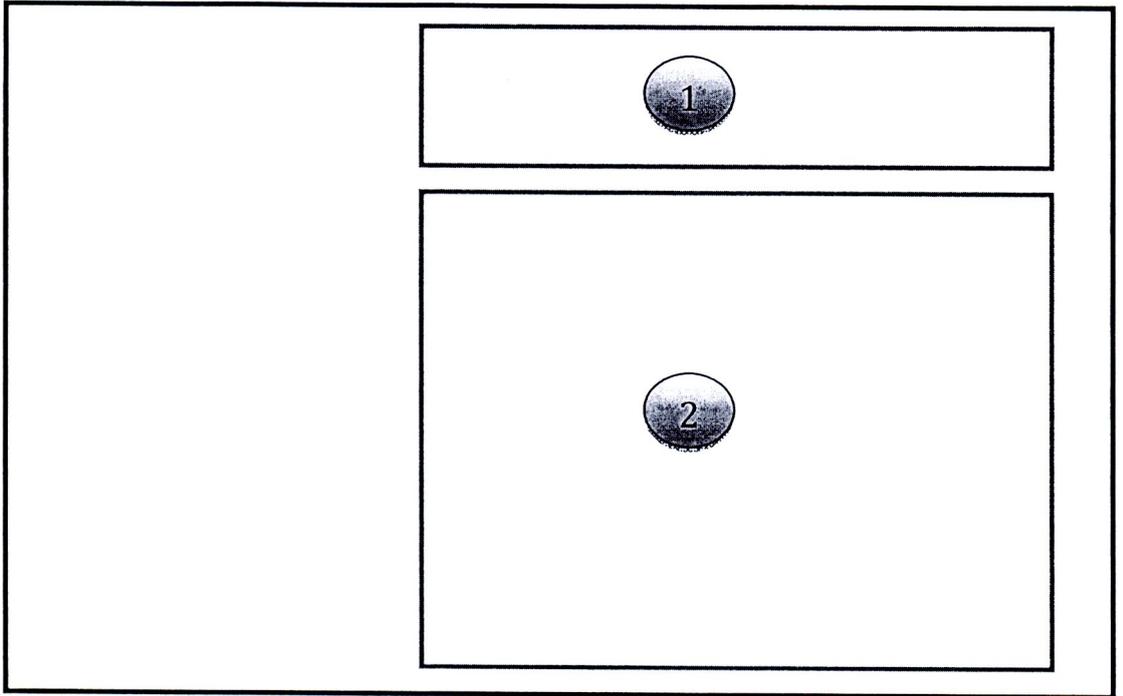
5.2 การออกแบบจอภาพบนเว็บเพจ สำหรับเจ้าหน้าที่ที่เกี่ยวข้อง

- 5.2.1 เมนูการตรวจสอบสถานะเอกสาร
- 5.2.2 การออกแบบจอภาพเมนูการอัปโหลดและดาวน์โหลดไฟล์เอกสาร

5.1 การออกแบบจอภาพบนเว็บเพจ สำหรับนักศึกษา

ส่วนการออกแบบจอภาพบนเว็บเพจสำหรับนักศึกษา เพื่อให้ระบบติดต่อกับผู้ใช้งานได้สะดวก โดยสามารถบันทึก จัดการข้อมูล รวมถึงการเรียกใช้ข้อมูลได้อย่างรวดเร็ว ซึ่งระบบได้ถูกออกแบบจอภาพเพื่อให้ผู้ใช้สะดวกในการใช้งาน โดยแบ่งออกดังนี้

5.1.1 การออกแบบหน้าจอเข้าสู่ระบบ



รูปที่ 5.1 หน้าจอเข้าสู่ระบบเว็บไซต์

จากรูป 5.1 ประกอบด้วย 2 ส่วน ดังรายละเอียดดังนี้

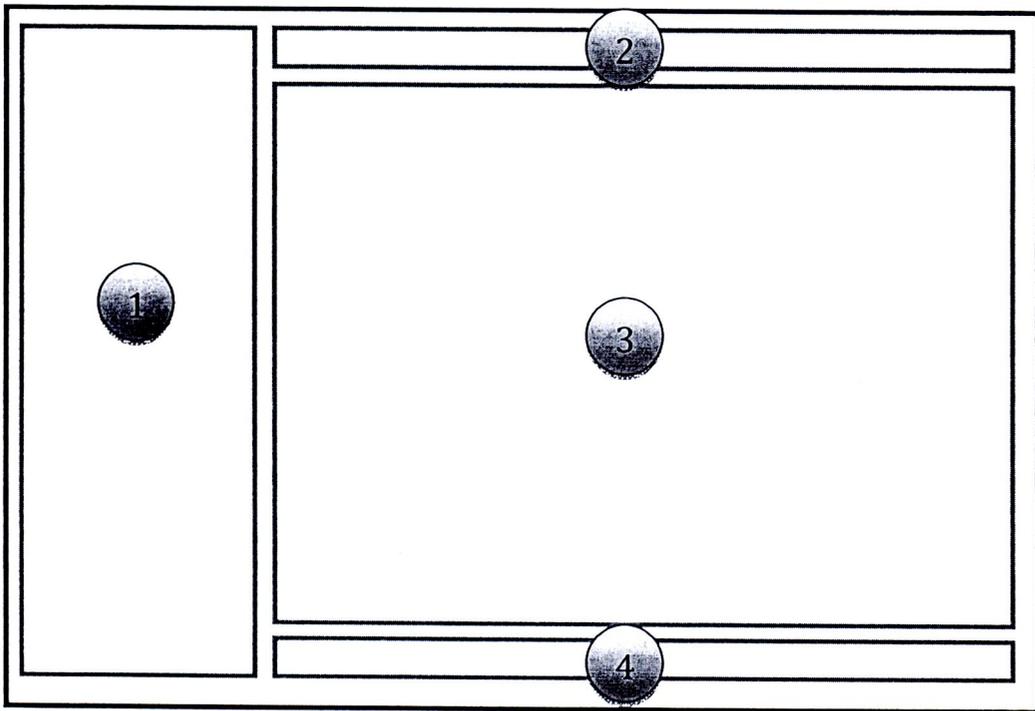
1) ส่วนที่ 1 เมนูต่างๆ ประกอบด้วยดังนี้

- (1) คิวบอร์ดโปรแกรมจัดการเอกสารการค้นคว้าแบบอิสระ
- (2) ตรวจสอบสถานะ
- (3) อีพโหลดไฟล์การค้นคว้าแบบอิสระในแต่ละบท
- (4) คิวบอร์ดไฟล์การค้นคว้าแบบอิสระในแต่ละบท

2) ส่วนที่ 2 ส่วนของการลงทะเบียนเข้าใช้โปรแกรม ประกอบด้วย ดังนี้

- (1) รหัสนักศึกษา
- (2) ชื่อนักศึกษา
- (3) นามสกุล
- (4) คณะวิชา
- (5) สาขาวิชา
- (6) อีเมลล์
- (7) หมายเลขโทรศัพท์

5.1.2 การออกแบบจอภาพเมนูหลักของนักศึกษา



รูปที่ 5.2 หน้าจอเมนูหลักของนักศึกษา

จากรูป 5.2 ประกอบด้วย 4 ส่วน ดังมีรายละเอียดดังนี้

1) ส่วนที่ 1 ส่วนของเมนูต่างๆ

- (1) คิวรี่โหลดโปรแกรมจัดการเอกสาร
- (2) ตรวจสอบสถานะ
- (3) อัปเดตไฟล์เอกสารการค้นคว้าแบบอิสระ
- (4) คิวรี่โหลดไฟล์เอกสารการค้นคว้าแบบอิสระ

2) ส่วนที่ 2 ส่วนของผู้ดูแลระบบ

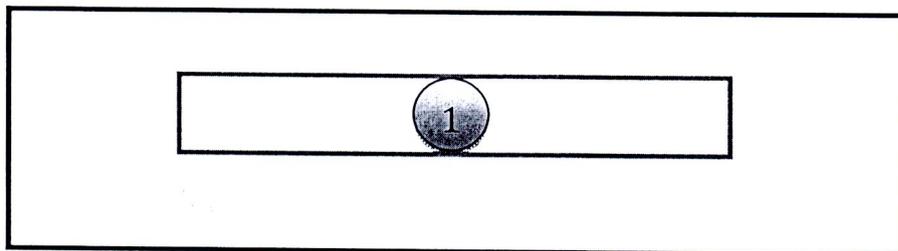
แสดงส่วนการทำงาน ดูแลและจัดการข้อมูลของนักศึกษาและเจ้าหน้าที่ที่เกี่ยวข้อง

5.1.3 การออกแบบจอภาพในส่วนของโปรแกรมจัดรูปแบบเอกสาร

ในส่วนของโปรแกรมจัดรูปแบบเอกสารสามารถแยกออกเป็น 2 ส่วนดังนี้

- (1) ในส่วนของโปรแกรม

หน้า Login

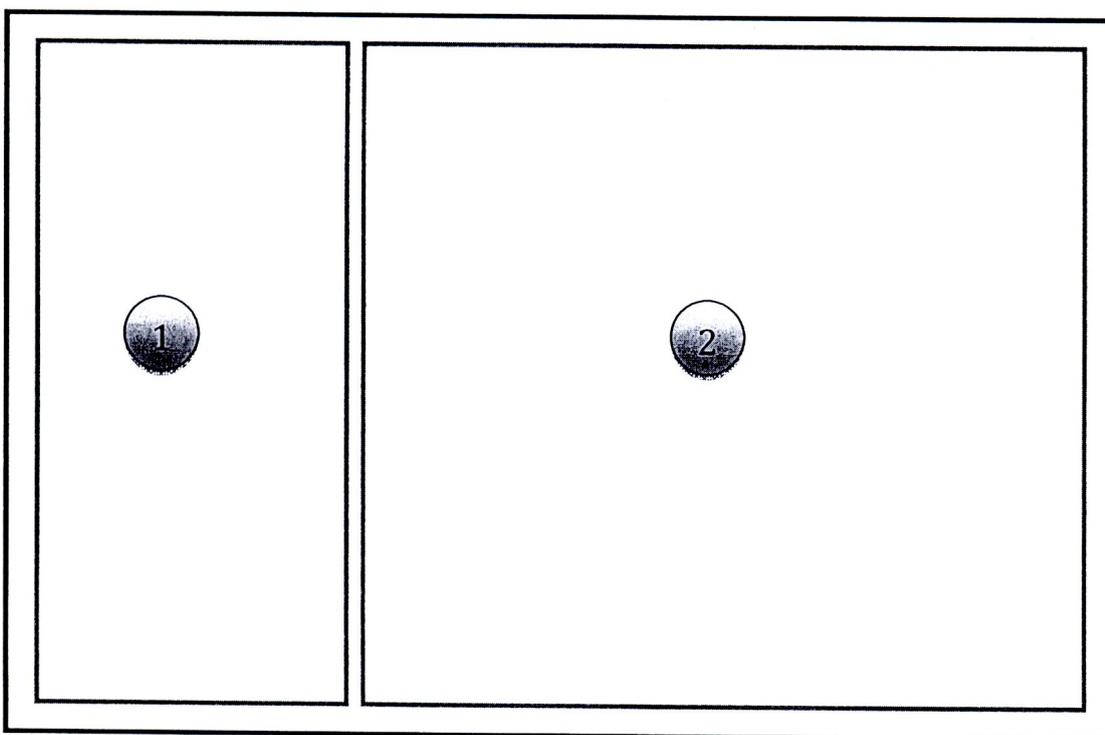


รูปที่ 5.3 หน้า Login

ประกอบด้วย รหัสนักศึกษา

(2) ในส่วนของหน้าในการทำงานจัดการเอกสารการค้นคว้าแบบอิสระ

หน้าการทำงาน

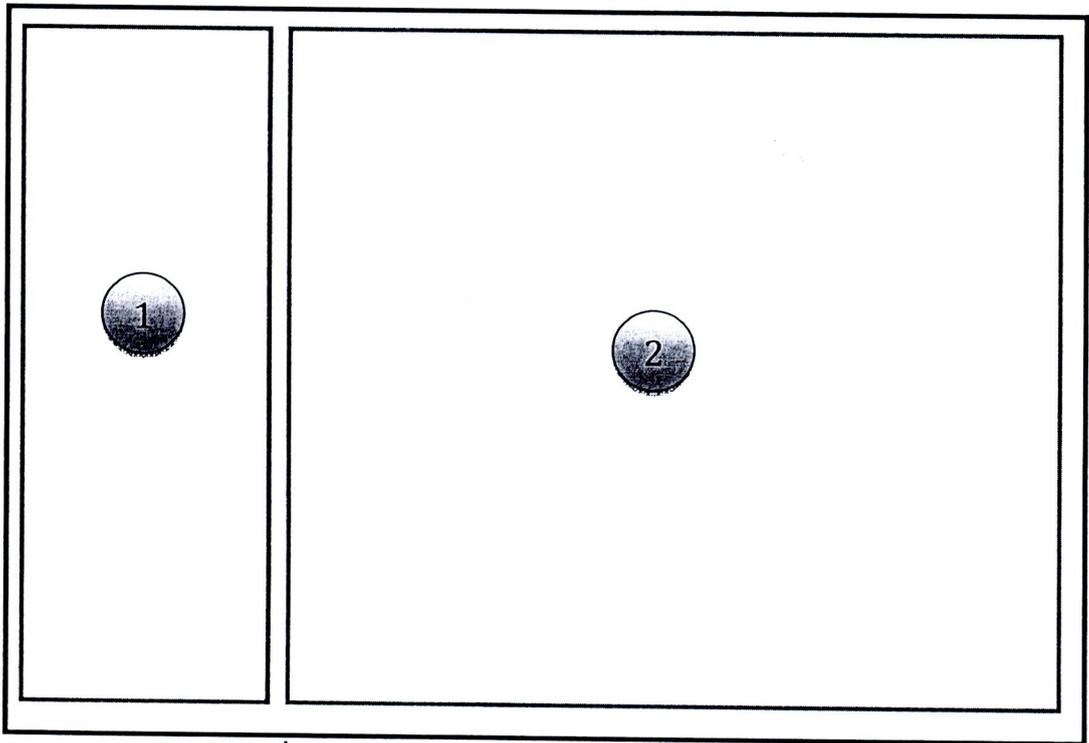


รูปที่ 5.4 หน้าจอในการทำงานจัดการเอกสารการค้นคว้าแบบอิสระ

ประกอบด้วย

- 1.เมนูเกี่ยวกับเอกสารวิทยานิพนธ์
- 2.หน้ารูปแบบเอกสารเอกสารวิทยานิพนธ์

5.1.4 การออกแบบจอภาพหน้าจอภาพเมนูการอัปเดตและดาวน์โหลดไฟล์เอกสาร



รูปที่ 5.5 หน้าจอภาพเมนูการอัปเดตและดาวน์โหลดไฟล์เอกสาร

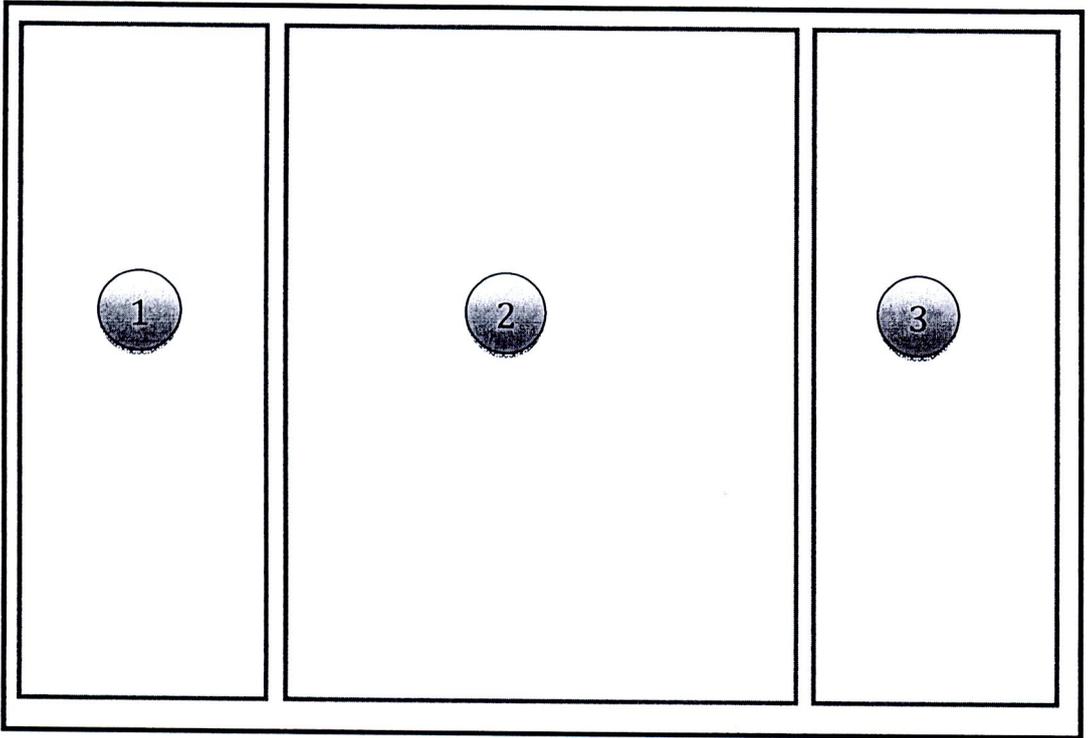
จากรูป 5.5 หน้าจอภาพเมนูการอัปเดตและดาวน์โหลดไฟล์เอกสาร ดังมีรายละเอียด
ดังนี้

ส่วนที่ 1 ส่วนเมนูการอัปเดตและดาวน์โหลดเอกสาร

ส่วนที่ 2 ส่วนการอัปเดตและดาวน์โหลดเอกสาร

5.2 การออกแบบจอภาพบนเว็บเพจ สำหรับเจ้าหน้าที่

5.2.1 เมนูการตรวจสอบสถานะเอกสาร



รูปที่ 5.6 หน้าจอตรวจสอบสถานะเอกสาร

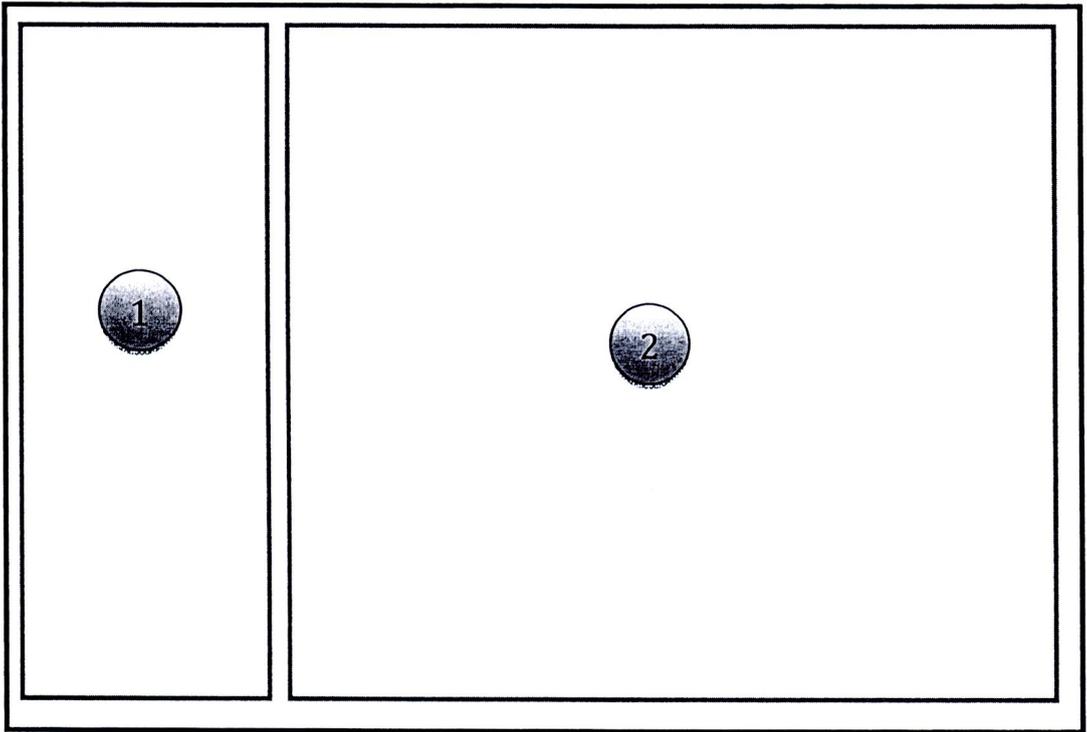
จากรูป 5.6 หน้าจอภาพเมนูตรวจสอบสถานะเอกสาร ดังมีรายละเอียดดังนี้

ส่วนที่ 1 ส่วนเมนูการตรวจสอบเอกสาร

ส่วนที่ 2 ส่วนรายละเอียด(ชื่อเอกสาร)

ส่วนที่ 3 สถานการณ์ตรวจสอบ/หมายเหตุ

5.2.2 การออกแบบจอภาพหน้าจอกาจอภาพเมนูการอัปโหลดและดาวน์โหลดไฟล์เอกสาร



รูปที่ 5.7 หน้าจอภาพเมนูการอัปเดตและดาวน์โหลดไฟล์เอกสาร

จากรูป 5.7 หน้าจอภาพเมนูการอัปเดตและดาวน์โหลดไฟล์เอกสาร ดังมีรายละเอียดดังนี้
ส่วนที่ 1 ส่วนเมนูการอัปเดตและดาวน์โหลดเอกสาร
ส่วนที่ 2 ส่วนการอัปเดตและดาวน์โหลดเอกสาร

การพัฒนาระบบ

การพัฒนาโปรแกรมจัดรูปแบบเอกสารวิทยานิพนธ์/การค้นคว้าแบบอิสระของนักศึกษา บัณฑิตวิทยาลัย มหาวิทยาลัยเชียงใหม่ มีลักษณะเป็นเว็บแอปพลิเคชันและโปรแกรมติดตั้งบนเครื่องไคลเอนต์(client) โดยในส่วนของเว็บแอปพลิเคชันเลือกใช้โปรแกรมภาษาพีเอชพี (PHP) ร่วมกับ ภาษาเอชทีเอ็มแอล (HTML) และภาษาจาวาสคริปต์(JAVA Script) และในส่วนของโปรแกรมติดตั้งบนเครื่องไคลเอนต์(client) เลือกใช้ภาษาไมโครซอฟต์วิซวลสตูดิโอ2008 (C#.net) โดยมีการพัฒนาดังนี้

5.3 การใช้โปรแกรมภาษาและเทคโนโลยีในการพัฒนาระบบ

5.4 การพัฒนาระบบ

5.5 โครงเว็บไซต์และโปรแกรม

5.3 การใช้โปรแกรมภาษาและเทคโนโลยีในการพัฒนาระบบ

การพัฒนาโปรแกรมจัดรูปแบบเอกสารวิทยานิพนธ์/การค้นคว้าแบบอิสระของนักศึกษา บัณฑิตวิทยาลัย มหาวิทยาลัยเชียงใหม่ มีรายละเอียดดังต่อไปนี้

ในส่วนของเว็บแอปพลิเคชัน

1) โปรแกรมภาษาพีเอชพี (PHP) ใช้ในการเขียนคำสั่งรับส่งข้อมูลไปยังเครื่องคอมพิวเตอร์แม่ข่าย (Server) โดยการเขียนโปรแกรมภาษาพีเอชพีใช้ในการสร้างเว็บฟอร์ม (Web Form) สำหรับการรับส่งข้อมูลในระบบ โดยมีนามสกุล .php

2) โปรแกรมภาษาจาวาสคริปต์ (Java Script) จาวาสคริปต์เป็นฟังก์ชันที่พัฒนาขึ้นเพื่อตรวจสอบและควบคุมการแสดงผลไฟล์ต่างๆผ่านหน้าจอ ซึ่งภาษาสคริปต์จะทำงานในฝั่งไคลเอนต์(Client) เท่านั้น

3) โปรแกรมภาษาเอชทีเอ็มแอล(HTML) ใช้ในการพัฒนาส่วนติดต่อกับผู้ใช้งาน การแสดงผลต่างๆ ในระบบ โดยไม่เป็นส่วนที่มีการคำนวณ

ในส่วนของโปรแกรมติดตั้งบนเครื่องไคลเอนต์(client)

1) ภาษาไมโครซอฟต์วิซวลสตูดิโอ2008 (C#.net) ใช้ในการเขียนคำสั่งเพื่อบันทึกข้อมูลไปยังฐานข้อมูล โดยการเขียนโปรแกรมภาษาไมโครซอฟต์วิซวลสตูดิโอ2008 (C#.net) ใช้ในการสร้างเป็นวินโดวฟอร์ม (Windows Form) โดยมีนามสกุลเป็น .Cs

5.4 การพัฒนาระบบ

โครงสร้างการพัฒนาระบบสนับสนุนการบริหารระบบการตรวจรูปเล่มวิทยานิพนธ์/การค้นคว้าแบบอิสระ งานบริการการศึกษา บัณฑิตวิทยาลัย มหาวิทยาลัยเชียงใหม่ แบ่งส่วนการพัฒนาออกเป็น 3 ส่วนหลักๆ ได้แก่ ส่วนติดต่อกับผู้ใช้ ส่วนการควบคุมการแสดงผล และส่วนติดต่อกับฐานข้อมูลเนื่องจากมีการแบ่งการทำงานออกเป็น ส่วนข้อมูล ส่วนจัดการข้อมูล ทำให้ง่ายต่อการปรับปรุงและเพิ่มเติมซอร์สโค้ดในภายหลัง

ส่วนของการแสดงผล(รายชื่อนักศึกษา)

ส่วนที่แสดงรายชื่อนักศึกษาที่มีการลงทะเบียนเข้าใช้ระบบ

```
<table width="100%" height="42" border="0" cellpadding="0" cellspacing="0">
```

```
<?php
```

```
$sql="SELECT tbstudents.*, tbstudentstatus.*
```

```
FROM tbstudentstatus INNER JOIN tbstudents ON tbstudentstatus.stu_Code =
tbstudents.stu_Code";
```

```
$result=mysql_query($sql);
```

```
// ===== กำหนด ข้อมูลในการแบ่งหน้า =====
```

```
$totalrecord=mysql_num_rows($result);
```

```
$pagesize =35; //กำหนดจำนวนเรคอร์ดที่ต้องการแสดงผลใน 1 หน้า
```

```
// คำนวณหาค่าของ page เริ่มต้น
```

```
/* คำนวณจำนวนหน้าที่ต้องใช้ในการแสดงผลเรคอร์ดทั้งหมด โดยเก็บไว้ในตัวแปร
```

```
ฟังก์ชัน ceil เป็นฟังก์ชันที่ใช้ในการปัดเศษขึ้น ให้เป็นจำนวนเต็ม และจะปัดขึ้นทั้งหมด*/
```

```
$totalpage = ceil($totalrecord / $pagesize);
```

```
/* $pageid คือตัวแปรที่จะถูกส่งผ่านทาง query string เมื่อผู้ใช้คลิกไฮเปอร์ลิงก์เพื่อเลือกหน้าที่ต้องการ */
```

```
pageid=$_GET["pageid"];
```

```

if (isset($pageid)) {
//คำนวณหาหมายเลขเรคอร์ดแรกที่เราจะเลือกมา
$start = $pagesize * ($pageid - 1);
}else {
$pageid = 1;
$start = 0;
}
// =====
$count=1;
// ===== แสดงผล การแบ่งหน้า =====

if($txtsearch != ""){ $sql.=" And stu_Name like '%".$txtsearch.'%';}
$sql.=" order by stu_RegisDate LIMIT $start,$pagesize";
$result=mysql_query($sql);
$count=1;
while ($dbarr=mysql_fetch_array($result)){
    ?>
    <tr>
        <td width="36%" valign="middle"><span class="style2"><a href="<?
if($_SESSION['yurname']!= 'user'){ echo "main-page.php?code=";}else{echo "upload\manage-
downloadfile.php?code=";}?><? echo $dbarr['stu_Code']; ?> " ><?php echo
$count.".)&nbsp;".$dbarr['stu_Code'].'.'.$dbarr['stu_Name'].' '.$dbarr['stu_SurName'];
?></a></span></td>
        <td width="25%" align="left"><?php echo $dbarr['stu_Faculty']; ?></td>
        <td width="25%" align="left"><?php echo $dbarr['stu_Department']; ?></td>
        <td width="14%" align="left"><?php if($dbarr['stu_StatusDoc']=="เอกสารสมบูรณ์"){ echo
" <strong style="color:#090">เอกสารสมบูรณ์</strong>"; }else{ echo $dbarr['stu_StatusDoc'];}
?> </td>
    </tr>
    <?php $count++;} ?>

```

ส่วนของการ Upload เอกสารของนักศึกษา

ในส่วนของการอัปโหลดไฟล์เอกสารวิทยานิพนธ์/การค้นคว้าแบบอิสระ ในแต่ละบท เพื่อให้เจ้าหน้าที่ตติวิทยาลัยทำการดาวน์โหลดเพื่อนำไปตรวจสอบความถูกต้องตามระเบียบของมหาวิทยาลัย

```
<table width="810" height="42" border="0" cellpadding="0" cellspacing="0">
  <?php
    $sql="Select * From tbdocumentfile Where doc_stuCode=" .$_GET['code']. " ";
    $result=mysql_query($sql);
    $count=1;
    $index=0;
    $arrday=array("", "ปกนอก,ปกใน,หน้าอนุมัติรายเซ็นต์,กิตติกรรม", "บทคัดย่อ(ไทย-อังกฤษ)", "
    สารบัญ", "บทที่1", "บทที่2", "บทที่3", "บทที่4", "บทที่5", "บทที่6", "บทที่7", "บทที่8", "
    บรรณานุกรม", "ภาคผนวก", "ภาคผนวก1", "ภาคผนวก2", "ภาคผนวก3", "ภาคผนวก4");
    $dbarr=mysql_fetch_array($result);
    $sql1="Select
    doc_Check1,doc_Check3,doc_Check2,doc_Check9,doc_Check10,doc_Check11,doc_Check12,do
    c_Check13,doc_Check14,doc_Check18,doc_Check19,doc_Check15,doc_Check4,doc_Check5,do
    c_Check6,doc_Check7,doc_Check8 from tbdocumentfile Where
    doc_stuCode=" .$_GET['code']. " ";
    $result1=mysql_query($sql1);
    $dbarr1=mysql_fetch_array($result1);
    while ($count<18){
    ?>
      <tr>
        <td width="2%" height="42" valign="middle"><span class="style2">&nbsp;<?php echo
        $count.". "; ?></span></td>
        <td width="47%" valign="middle"><?php echo $arrday[$count]; ?><font
        color="#FF0000">
          <input name="txtstatus[]" type="hidden" value="<?php echo $dbarr1[$index]; ?>"
          id="txtstatus[]" />
```

```

</font></td>
<td width="28%" align="left" valign="middle">
<?
if($dbarr1[$index] ==1) { echo " <strong style=\"color:#090\">สมบูรณ์</strong>";}
if($dbarr1[$index] ==2) { echo " <strong style=\"color:#F00\">รอการแก้ไข</strong>";}
if($dbarr1[$index] ==3) { echo "ส่งเอกสาร";}
if($dbarr1[$index] ==4) { echo " <strong style=\"color:#F00\">แก้ไขเอกสารกลับ</strong>";}
?></td>
<td width="23%" align="left" valign="middle"><input name="item_file[]" type="file"
class="contacttextsearch" size="30" id="item_file[]" /></td>
</tr>
<?php $count++;$index++;} ?>

```

5.5 เทคนิคการพัฒนาระบบ

ในการพัฒนาระบบผู้ค้นคว้าได้นำเทคนิคต่างๆมาใช้เพื่ออำนวยความสะดวกแก่ผู้ใช้ให้สามารถใช้งานระบบได้อย่างง่ายดาย อีกทั้งยังช่วยประหยัดเวลาและเพิ่มประสิทธิภาพในการทำงานของนักศึกษาและเจ้าหน้าที่ที่เกี่ยวข้อง ซึ่งประกอบด้วย

5.5.1 ส่วนของหน้าบันทึก ปกนอก ปกใน หน้าอนุมัติลายเซ็น กิตติกรรมประกาศ

เป็นส่วนการจัดการข้อมูลในส่วนของปกนอก ปกใน หน้าอนุมัติลายเซ็น กิตติกรรมประกาศ ตามรูปแบบที่ทางมหาวิทยาลัยกำหนดเพื่อบันทึกเป็นไฟล์ข้อมูลอัฟโหลดให้กับเจ้าหน้าที่บัณฑิตวิทยาลัยตรวจสอบความถูกต้องต่อไป

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using DevExpress.XtraEditors;
using System.Diagnostics;

```

```

using ISITProject.Lesson;

namespace ISITProject
{
    public partial class frmabstract : DevExpress.XtraEditors.XtraForm
    {
        string setID;
        bool IsUpdate = false;
        public frmabstract(string getID,string getName)
        {
            InitializeComponent();
            setID = getID;
            txtAuthorTH.Text = getName;
        }

        private void barButtonItem1_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
        {
            SaveData();
            barButtonItem2.Enabled = true;
            IsUpdate = false;
        }
        private void SaveData()
        {
            string SQL = "";
            if (!IsUpdate)
            {
                SQL = "INSERT INTO
tblabstract(Code,TitleTH,AuthorTH,DegreeTH,BranchTH,IndependentTH,AbstractTH,TitleENG,AuthorENG,
DegreeENG,BranchENG,IndependentENG,AbstractENG)";
                SQL = SQL + "VALUES('" + setID + "','" + txtTitleTH.Text + "','" + txtAuthorTH.Text + "','" +
txtDegreeTH.Text + "','" + txtBranchTH.Text + "','" + txtIndependentTH.Text + "','" +
                SQL = SQL + "'" + txtabstractTH.Text + "','" + txtTitleENG.Text + "','" + txtAuthorENG.Text + "','"
+ txtDegreeENG.Text + "','" + txtBranchENG.Text + "','" + txtIndependentENG.Text + "','" +
                SQL = SQL + "'" + txtabstractENG.Text + "')";
            }
        }
    }
}

```



```

}
else
{
    SQL = "UPDATE tabstract SET TitleTH=" + txtTitleTH.Text + ",AuthorTH=" +
txtAuthorTH.Text + ",DegreeTH=" + txtDegreeTH.Text + ",BranchTH=" + txtBranchTH.Text +
",IndependentTH=" + txtIndependentTH.Text + ",";

    SQL = SQL + "AbstractTH=" + txtabstractTH.Text + ",TitleENG=" + txtTitleENG.Text +
",AuthorENG=" + txtAuthorENG.Text + ",DegreeENG=" + txtDegreeENG.Text + ",BranchENG=" +
txtBranchENG.Text + ",";

    SQL = SQL + "IndependentENG=" + txtIndependentENG.Text + ",AbstractENG=" +
txtabstractENG.Text + " WHERE Code=" + setID + " ";
}
if (new Connection().Execute(SQL))
{
    MessageBox.Show("จัดเก็บเรียบร้อย", "เสร็จสิ้น");
}
}

private void barButtonItem4_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    setID = txtCode.EditValue.ToString();
    string SQL = "SELECT * FROM tabstract WHERE Code=" + setID + " ";
    DataTable dt = new Connection().Query(SQL, "abstract");
    if (dt.Rows.Count > 0)
    {
        txtTitleTH.Text = dt.Rows[0]["TitleTH"].ToString();
        txtAuthorTH.Text = dt.Rows[0]["AuthorTH"].ToString();
        txtDegreeTH.Text = dt.Rows[0]["DegreeTH"].ToString();
        txtBranchTH.Text = dt.Rows[0]["BranchTH"].ToString();
        txtIndependentTH.Text = dt.Rows[0]["IndependentTH"].ToString();
        txtabstractTH.Text = dt.Rows[0]["AbstractTH"].ToString();
        txtTitleENG.Text = dt.Rows[0]["TitleENG"].ToString();
        txtAuthorENG.Text = dt.Rows[0]["AuthorENG"].ToString();
    }
}

```

```

txtDegreeENG.Text = dt.Rows[0]["DegreeENG"].ToString();
txtBranchENG.Text = dt.Rows[0]["BranchENG"].ToString();
txtIndependentENG.Text = dt.Rows[0]["IndependentENG"].ToString();
txtabstractENG.Text = dt.Rows[0]["AbstractENG"].ToString();
barButtonItem2.Enabled = true;
IsUpdate = true;
}
else
{
    MessageBox.Show("ไม่พบข้อมูล", "ลองใหม่");
}
}

private void barButtonItem2_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    saveFileDialog.Filter = "doc files (*.doc)*.doc";
    saveFileDialog.FileName = "Abstract.doc";
    if (saveFileDialog.ShowDialog() == DialogResult.OK)
    {
        AbstractPage rpt = new AbstractPage();
        rpt.FilterString = "[Code]=" + setID + "";
        rpt.CreateDocument();
        rpt.PrintingSystem.ExportToRtf(saveFileDialog.FileName + ".doc");

        MessageBox.Show("เอกสารนี้สามารถปรับปรุงหรือแก้ไขใน Microsoft Work ได้", "เปิด",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        if (MessageBox.Show("คุณต้องการเปิดไฟล์นี้หรือไม่?", "เปิด", MessageBoxButtons.YesNo,
        MessageBoxIcon.Question, MessageBox.DefaultButton.Button1) == DialogResult.Yes)
        {
            try
            {
                System.Diagnostics.Process.Start(saveFileDialog.FileName + ".doc");
            }
        }
    }
}

```

```

        catch (Exception)
        {
            MessageBox.Show("เปิดไฟล์ไม่สำเร็จ", "Error");
        }
    }
}

private void barButtonItem3_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    string pathname = Application.StartupPath + "\\Abstract.doc";
    Process.Start(pathname);
}

private void frmabstract_Load(object sender, EventArgs e)
{
    txtCode.EditValue=setID ;
    string SQL = "SELECT * FROM tbabstract WHERE Code='" + setID + "' ";
    DataTable dt = new Connection().Query(SQL, "abstract");
    if (dt.Rows.Count > 0)
    {
        txtTitleTH.Text = dt.Rows[0]["TitleTH"].ToString();
        txtAuthorTH.Text = dt.Rows[0]["AuthorTH"].ToString();
        txtDegreeTH.Text = dt.Rows[0]["DegreeTH"].ToString();
        txtBranchTH.Text = dt.Rows[0]["BranchTH"].ToString();
        txtIndependentTH.Text = dt.Rows[0]["IndependentTH"].ToString();
        txtabstractTH.Text = dt.Rows[0]["AbstractTH"].ToString();
        txtTitleENG.Text = dt.Rows[0]["TitleENG"].ToString();
        txtAuthorENG.Text = dt.Rows[0]["AuthorENG"].ToString();
        txtDegreeENG.Text = dt.Rows[0]["DegreeENG"].ToString();
        txtBranchENG.Text = dt.Rows[0]["BranchENG"].ToString();
        txtIndependentENG.Text = dt.Rows[0]["IndependentENG"].ToString();
        txtabstractENG.Text = dt.Rows[0]["AbstractENG"].ToString();
        barButtonItem2.Enabled = true;
    }
}

```

```

    IsUpdate = true;
}

```

5.5.2 ส่วนของหน้าบันทึก บทความย่อ (ไทย-อังกฤษ)

เป็นส่วนการจัดการข้อมูลในส่วนของบทความย่อ(ภาษาไทย-อังกฤษ) ตามรูปแบบที่ทางมหาวิทยาลัยกำหนดเพื่อบันทึกเป็นไฟล์ข้อมูลออฟไลน์ให้กับเจ้าหน้าที่บัณฑิตวิทยาลัยตรวจสอบความถูกต้องต่อไป

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using DevExpress.XtraEditors;
using System.Diagnostics;
using ISITProject.Lesson1;

namespace ISITProject
{
    public partial class frmApprove : DevExpress.XtraEditors.XtraForm
    {
        string setID;
        bool IsUpdate = false;
        public frmApprove(string getID, string getName)
        {
            InitializeComponent();
            dtDateTesis.DateTime = DateTime.Now;
            setID = getID;
            txtStuname.Text = getName;
        }
    }
}

```

```

private void barButtonItem1_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    SaveData();
    IsUpdate = false;
    barButtonItem2.Enabled = true;
}
private void SaveData()
{
    string SQL = "";
    if (!IsUpdate)
    {
        SQL = "INSERT INTO
tbsignature(Code,Title,DateTesis,Stuname,Approve,Degree,Brach,Advisor,PositionAdvisor,Board,PositionBoar
d,Directors1,PositionDirectors1,Directors2,PositionDirectors2,CoBoard,PositionCoBoard,memotitle)";
        SQL=SQL + "VALUES('" + setID + "','" + txtTitle.Text + "','" + dtDateTesis.DateTime + "','" +
txtStuname.Text + "','" + txtApprove.Text + "','" + txtDegree.Text + "','" +
        SQL = SQL + "'" + txtBrach.Text + "','" + txtAdvisor.Text + "','" + txtPositionAdvisor.Text + "','" +
txtBoard.Text + "','" + txtPositionBoard.Text + "','" + txtDirectors1.Text + "','" +
        SQL = SQL + "'" + txtPositionDirectors1.Text + "','" + txtDirectors2.Text + "','" +
txtPositionDirectors2.Text + "','" + txtCoBoard.Text + "','" + txtPositionCoBoard.Text + "','" + memo.Text + "')";
    }
    else
    {
        SQL="UPDATE tbsignature SET Title='"+ txtTitle.Text+"',DateTesis='"+ dtDateTesis.DateTime
+ "',Stuname='"+ txtStuname.Text + "',Approve='"+ txtApprove.Text + "',Degrce='"+ txtDegree.Text+",";
        SQL = SQL + "Brach='" + txtBrach.Text + "',Advisor='" + txtAdvisor.Text + "',PositionAdvisor='" +
txtPositionAdvisor.Text + "',Board='" + txtBoard.Text + "',PositionBoard='" + txtPositionBoard.Text + "','" +
        SQL = SQL + "Directors1='" + txtDirectors1.Text + "',PositionDirectors1='" +
txtPositionDirectors1.Text + "',Directors2='" + txtDirectors2.Text + "',PositionDirectors2='" +
txtPositionDirectors2.Text + "','" +
        SQL = SQL + "CoBoard='" + txtCoBoard.Text + "',PositionCoBoard='" + txtPositionCoBoard.Text +
" WHERE Code='" + setID + "'";
    }
}

```

```

if (new Connection().Execute(SQL))
{
    MessageBox.Show("จัดเก็บเรียบร้อยแล้ว", "เสร็จสิ้น");
}
}

private void barButtonItem3_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    string pathname=Application.StartupPath + "\\ApprovePage.doc";
    Process.Start(pathname);
}

private void barButtonItem4_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    setID = txtCode.EditValue.ToString();
    string SQL="SELECT * FROM tbsignature WHERE Code=" + setID + " ";
    DataTable dt= new Connection().Query(SQL, "tbsignature");
    if (dt.Rows.Count > 0)
    {
        txtTitle.Text = dt.Rows[0]["Title"].ToString();
        dtDateTesis.EditValue = dt.Rows[0]["DateTesis"];
        txtStuname.Text = dt.Rows[0]["Stuname"].ToString();
        txtApprove.Text = dt.Rows[0]["Approve"].ToString();
        txtDegree.Text = dt.Rows[0]["Degree"].ToString();
        txtBrach.Text = dt.Rows[0]["Brach"].ToString();
        txtAdvisor.Text = dt.Rows[0]["Advisor"].ToString();
        txtPositionAdvisor.Text = dt.Rows[0]["PositionAdvisor"].ToString();
        txtBoard.Text = dt.Rows[0]["Board"].ToString();
        txtPositionBoard.Text = dt.Rows[0]["PositionBoard"].ToString();
        txtDirectors1.Text = dt.Rows[0]["Directors1"].ToString();
        txtPositionDirectors1.Text = dt.Rows[0]["PositionDirectors1"].ToString();
        txtDirectors2.Text = dt.Rows[0]["Directors2"].ToString();
        txtPositionDirectors2.Text = dt.Rows[0]["PositionDirectors2"].ToString();
        txtCoBoard.Text = dt.Rows[0]["CoBoard"].ToString();
    }
}

```

```

txtPositionCoBoard.Text = dt.Rows[0]["PositionCoBoard"].ToString();
memo.Text = dt.Rows[0]["memotitle"].ToString();
IsUpdate = true;
barButtonItem2.Enabled = true;
}
else
{
    MessageBox.Show("ไม่พบข้อมูล","ลองใหม่");
}
}

private void barButtonItem2_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)
{
    saveFileDialog.Filter = "doc files (*.doc)*.doc";
    saveFileDialog.FileName = "Cover.doc";
    if (saveFileDialog.ShowDialog() == DialogResult.OK)
    {
        ApprovePage rpt = new ApprovePage();
        rpt.FilterString = "[Code]=" + setID + "";
        rpt.CreateDocument();
        rpt.PrintingSystem.ExportToRtf(saveFileDialog.FileName + ".doc");

        MessageBox.Show("เอกสารนี้สามารถปรับปรุงหรือแก้ไขใน Microsoft Work ได้", "เปิด",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        if (MessageBox.Show("คุณต้องการเปิดไฟล์นี้หรือไม่?", "เปิด", MessageBoxButtons.YesNo,
        MessageBoxIcon.Question, MessageBoxDefaultButton.Button1) == DialogResult.Yes)
        {
            try
            {
                System.Diagnostics.Process.Start(saveFileDialog.FileName + ".doc");
            }
            catch (Exception)
            {
            }
        }
    }
}

```

```

        MessageBox.Show("เปิดไฟล์ไม่สำเร็จ", "Error");
    }
}
}
}

```

```

private void frmApprove_Load(object sender, EventArgs e)
{
    txtCode.EditValue = setID;
    //setID = txtCode.EditValue.ToString();
    string SQL = "SELECT * FROM tbsignature WHERE Code='" + setID + "' ";
    DataTable dt = new Connection().Query(SQL, "tbsignature");
    if (dt.Rows.Count > 0)
    {
        txtTitle.Text = dt.Rows[0]["Title"].ToString();
        dtDateTesis.EditValue = dt.Rows[0]["DateTesis"];
        txtStuname.Text = dt.Rows[0]["Stuname"].ToString();
        txtApprove.Text = dt.Rows[0]["Approve"].ToString();
        txtDegree.Text = dt.Rows[0]["Degree"].ToString();
        txtBrach.Text = dt.Rows[0]["Brach"].ToString();
        txtAdvisor.Text = dt.Rows[0]["Advisor"].ToString();
        txtPositionAdvisor.Text = dt.Rows[0]["PositionAdvisor"].ToString();
        txtBoard.Text = dt.Rows[0]["Board"].ToString();
        txtPositionBoard.Text = dt.Rows[0]["PositionBoard"].ToString();
        txtDirectors1.Text = dt.Rows[0]["Directors1"].ToString();
        txtPositionDirectors1.Text = dt.Rows[0]["PositionDirectors1"].ToString();
        txtDirectors2.Text = dt.Rows[0]["Directors2"].ToString();
        txtPositionDirectors2.Text = dt.Rows[0]["PositionDirectors2"].ToString();
        txtCoBoard.Text = dt.Rows[0]["CoBoard"].ToString();
        txtPositionCoBoard.Text = dt.Rows[0]["PositionCoBoard"].ToString();
        memo.Text = dt.Rows[0]["memotitle"].ToString();
        barButtonItem2.Enabled = true;
        IsUpdate = true;
    }
}

```

5.5.3 ส่วนของการเชื่อมต่อฐานข้อมูล

เป็นส่วนที่ใช้ในการเชื่อมต่อระหว่างเว็บแอปพลิเคชันกับฐานข้อมูล

เชื่อมต่อเซิร์ฟเวอร์

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.OleDb;
using System.Data;

namespace ISITProject
{
    public class Connection
    {
        public System.Data.OleDb.OleDbConnection ConnODBE = new System.Data.OleDb.OleDbConnection();
        public string StrCon;
        public void ConnectionString()
        {
            try
            {
                StrCon = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" + Application.StartupPath +
                "\\sitDB.acedb;Persist Security Info=False;";
                return;
            }
            catch (Exception ex)
            {
                MessageBox.Show("ไม่สามารถเชื่อมต่อกับฐานข้อมูลได้ กรุณาตรวจสอบ " + ex.ToString(), "คำ
                เตือน");
                System.Environment.Exit(0);
            }
        }
        public bool Execute(string Sqldata)
```

```

{
    bool functionReturnValue = false;
    OleDbCommand mycommand = new OleDbCommand();
    OpenData();
    try
    {
        var _with1 = mycommand;
        _with1.CommandType = CommandType.Text;
        _with1.CommandText = Sqldata;
        _with1.Connection = ConnODBE;
        _with1.ExecuteNonQuery();
        functionReturnValue = true;
    }
    catch (OleDbException ex)
    {
        functionReturnValue = false;
        MessageBox.Show(ex.Message);
    }
    ConnODBE.Close();
    return functionReturnValue;
}

public void OpenData()
{
    ConnectionString();
    var _with1 = ConnODBE;
    if (_with1.State == ConnectionState.Open)
        _with1.Close();
    _with1.ConnectionString = StrCon;
    _with1.Open();
}

public DataTable Query(string Sql, string table)
{

```

```
OpenData();  
System.Data.OleDb.OleDbDataAdapter AddDataAdapter = new  
System.Data.OleDb.OleDbDataAdapter(Sql, StrCon);  
System.Data.DataSet DataSetq = new System.Data.DataSet();  
try  
{  
  
    AddDataAdapter.Fill(DataSetq, table);  
  
}  
catch (Exception ex)  
{  
    MessageBox.Show("ไม่สามารถเชื่อมต่อกับฐานข้อมูลได้ กรุณา Config ใหม่อีกครั้ง..." + ex.ToString(),  
"แจ้งเตือน", MessageBoxButtons.OK, MessageBoxIcon.Error);  
  
    System.Environment.Exit(0);  
}  
  
return DataSetq.Tables[table];  
}  
}  
}
```