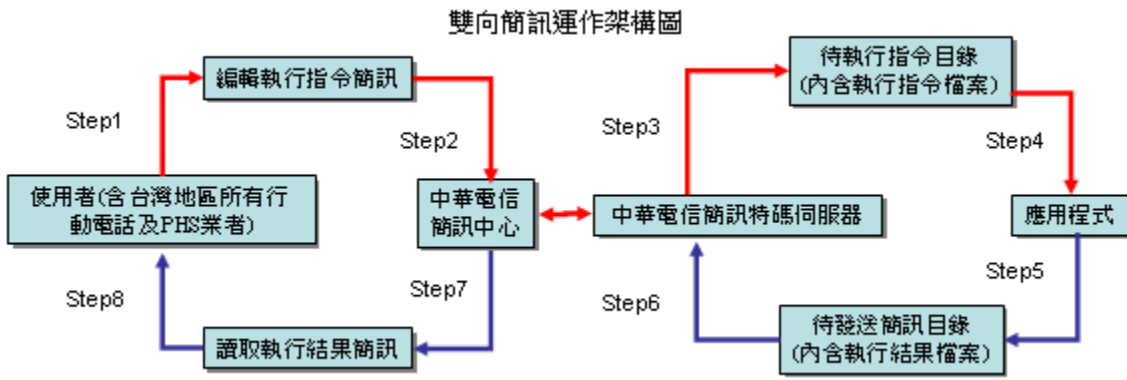


?????????????????. ??SMS, MMS?WAP Push???. ??????????(????). ??????????????



Step1:使用者編輯欲執行指令的簡訊

Step2:使用者發送編輯完成的簡訊至簡訊特碼

Step3:簡訊特碼伺服器定時接收執行指令的簡訊，並將待執行指令與使用者手機號碼存成檔案，放置於"待執行指令"目錄中

Step4:應用程式定時檢查"待執行指令"目錄中是否有待執行指令檔案，若有，則依指令執行應用程式

Step5:應用程式將執行結果與使用者手機號碼存成待發送檔案，放置於"待發送簡訊"目錄中

Step6:簡訊特碼伺服器定時於"待發送簡訊"目錄中，檢查是否有檔案，若有，則依檔案內容傳送簡訊內容至簡訊中心

Step7:簡訊中心發送簡訊至指定的使用者手機號碼

Step8:使用者讀取執行結果簡訊

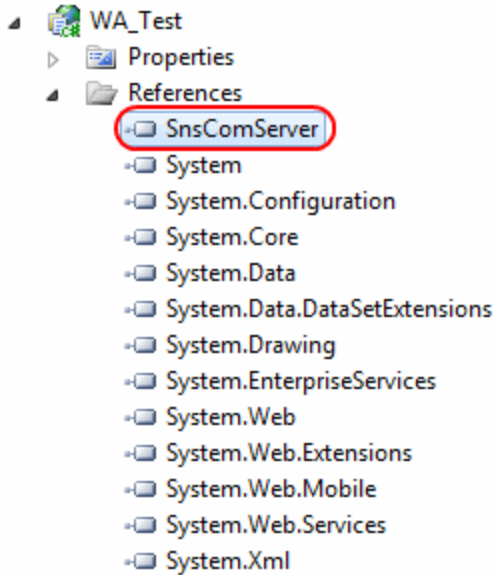
???

體育活動雙向簡訊運作流程(範例)



\* ?Messaging Server?????, ?????????1440??, ?????????source code????compile.

?????????????????SNS COM??, ??SNS COM+?.NET?????????



??API??????????????????:

<http://www.emome.net/channel?chid=212>

??Code Behind?????API???:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using SNSCOMSERVER;
```

??SNS??????????????????????:

```
private static SNSCOMSERVER.SnsComObject _snsComObject = new SNSCOMSERVER.SnsComObject();
private static string _connectionFlag;

protected void Page_Init(object sender, EventArgs e) {

    this.StdBtn_Login.Command += new CommandEventHandler(Button_Command);
    this.StdBtn_Logout.Command += new CommandEventHandler(Button_Command);
    this.StdBtn_Check.Command += new CommandEventHandler(Button_Command);
    this.StdBtn_Send.Command += new CommandEventHandler(Button_Command);
    this.StdBtn_Query.Command += new CommandEventHandler(Button_Command);
} //end Page_Init
```

????Web UI, ????????????, ??????:

SNS Sender Service

**Settings**

IP 
 User 
 Password 
 Port

Phone Number

Content

RespMsg

Query Status [EMPTY]

????????????:

```

protected void Button_Command(object sender, CommandEventArgs e) {
    string commandName = e.CommandName;
    switch (commandName) {
        case "Login":
            string ip = StdTxt_Ip.Text;
            string user = StdTxt_User.Text;
            string password = StdTxt_Password.Text;
            int port = 0;
            if (!int.TryParse(StdTxt_Port.Text, out port))
                port = 8001;
            _connectionFlag = _snsComObject.Login(ip, port, user, password).ToString();
            StdLbl_QueryStatus.Text = _connectionFlag;
            break;
        case "Logout":
            _snsComObject.Logout();
            _connectionFlag = "-1";
            StdLbl_QueryStatus.Text = _connectionFlag;
            break;
        case "Check":
            StdLbl_QueryStatus.Text = _connectionFlag;
            break;
        case "Send":
            string content = StdTxt_Content.Text;
            if (content != "") {
                string phoneNumber = StdTxt_PhoneNumber.Text;
                if (_connectionFlag == "0")
                {
                    _snsComObject.SubmitMessage(phoneNumber, content).ToString();
                    System.Threading.Thread.Sleep(3000);
                    string respMsg = _snsComObject.RespMessage;
                }
            }
    }
}

```

```

        StdTxt_RespMsg.Text = respMsg;
        StdLbl_QueryStatus.Text =
            _snsComObject.QryMessageStatus(phoneNumber, respMsg).ToString();
    } //end if
} //end if
break;
case "Query":
    if (_connectionFlag == "0") {
        string phoneNumber = StdTxt_PhoneNumber.Text;
        string respMsg = StdTxt_RespMsg.Text;
        StdLbl_QueryStatus.Text =
            _snsComObject.QryMessageStatus(phoneNumber, respMsg).ToString();
    } //end if
    break;
} //end switch
} //end Button_Command

```