

TẬP SLIDE BÀI GIẢNG

MÔN LẬP TRÌNH MẠNG

Người soạn : TS. Nguyễn Văn Hiệp



MÔN LẬP TRÌNH MẠNG

Đối tượng : SV đại học ngành Công nghệ thông tin

Nội dung chính :

1. Lập trình ứng dụng mạng theo mô hình Client/Server dùng giao thức TCP/IP.
2. Lập trình Web chạy trên Client và trên Server bằng các công nghệ phổ biến.

Tài liệu tham khảo chính :

- [1] Computer Networks, A.S. Tanenbaum, Prentice-Hall, Edition 3.
- [2] Bộ CD MSDN của Microsoft.
- [3] Online Help của môi trường lập trình JBuilder
- [4] CDROM chứa RFCs.
- [5] Core Servlets and JavaServer Pages, Marty Hall, Sun



MÔN LẬP TRÌNH MẠNG

Đối tượng : SV đại học ngành Công nghệ thông tin

Nội dung chính gồm 10 chương :

1. Lập trình mạng theo mô hình Client/Server dùng giao thức TCP/IP (cụ thể là Winsock)
2. Viết hệ thống MiniChatter bằng VC++ & dùng kỹ thuật xử lý sự kiện.
3. Viết hệ thống MiniChatter bằng Java & dùng kỹ thuật xử lý multi-thread.
4. Tổng quát về lập trình Web.
5. DHTML & Client Script.
6. Lập trình Web chạy trên Client dùng Applet Java.
7. Lập trình Web chạy trên Client dùng ActiveX.
8. Lập trình Web chạy trên Server bằng công nghệ Microsoft.
9. Lập trình Web chạy trên Server bằng PHP.
10. Lập trình Web chạy trên Server bằng Java.



MÔN LẬP TRÌNH MẠNG

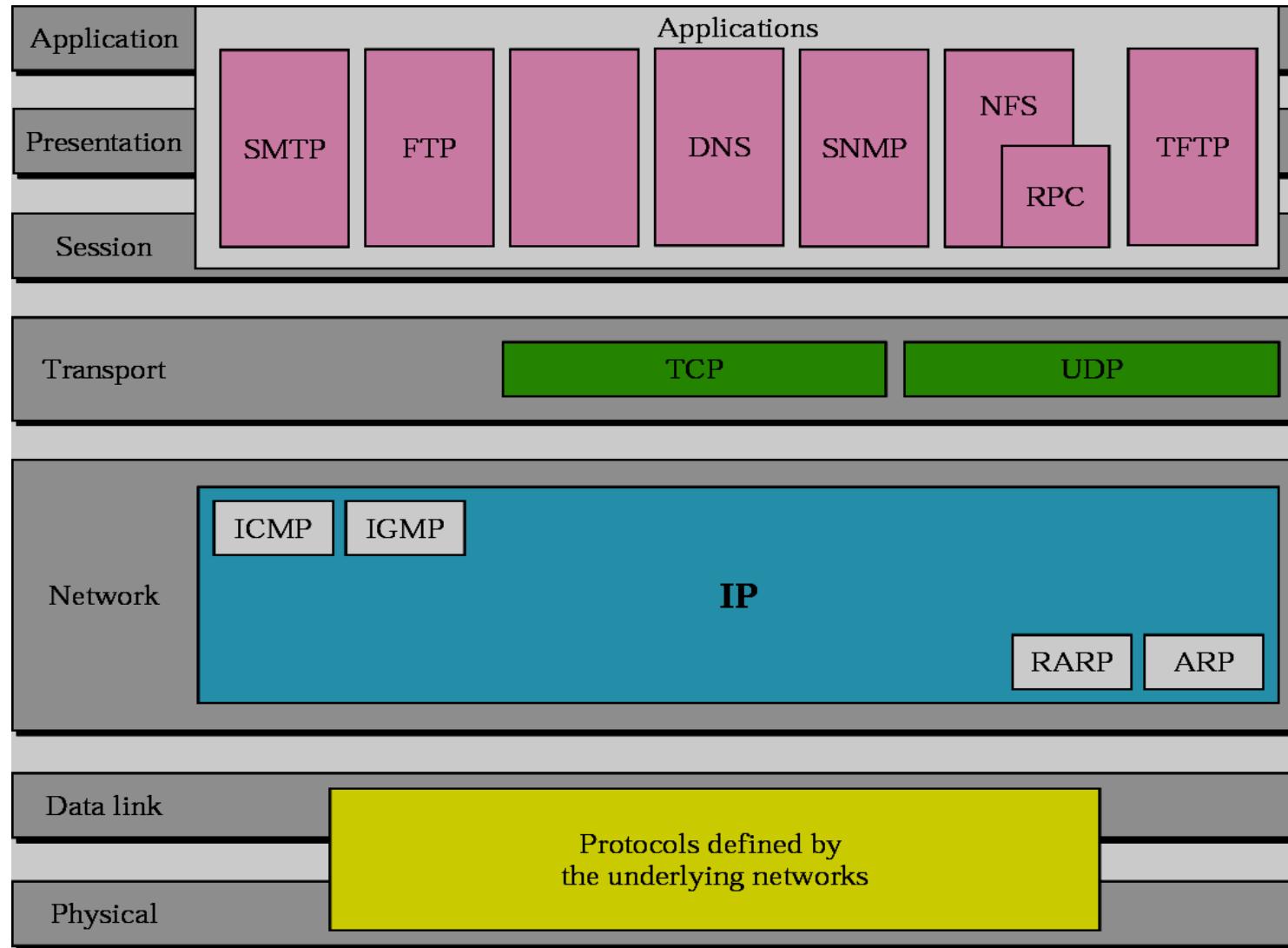
Chương 1

LẬP TRÌNH MẠNG THEO MÔ HÌNH CLIENT/SERVER DÙNG GIAO THỨC TCP/IP (CỤ THỂ LÀ WINSOCK)

- I.1 Ôn lại kiến trúc mạng Internet
- I.2 Mô hình hoạt động client/server dùng socket
- I.3 Đặc tả các hàm Winsock cơ bản
- I.4 Vấn đề bất đồng bộ trong gửi/nhận thông tin
- I.5 Thí dụ về ứng dụng mạng cơ bản



I.1 Ôn lại kiến trúc mạng Internet



Tổng quát về lập trình mạng trên Internet

- ❑ Xem kiến trúc của mạng Internet trong slide trước, ta thấy việc lập trình ứng dụng sẽ dựa vào 1 trong 2 giao thức TCPIUDP của cấp TCP.
- ❑ Giao thức TCP dùng cầu nối nên rất tin cậy (không mất, không sai, không thay đổi thứ tự truyền/nhận).
- ❑ Giao thức UDP không dùng cầu nối nên không tin cậy, code của ứng dụng cần kiểm soát lỗi trong quá trình gửi/nhận thông tin (nếu muốn).
- ❑ Hiện trên các platform khác nhau, người ta cung cấp giao tiếp lập trình của thư viện socket để lập trình trên cấp TCP. Thư viện socket trên Windows được gọi là Winsock. Trong phần còn lại của chương 1, chúng ta sẽ trình bày chi tiết về các hàm thư viện winsock cơ bản và cách sử dụng chúng để lập trình 1 ứng dụng nhỏ. Các thông tin của chương này có thể được áp dụng trên các platform khác với sự thay đổi nhỏ (do có sự khác biệt nhỏ giữa các thư viện socket trên các platform khác nhau).

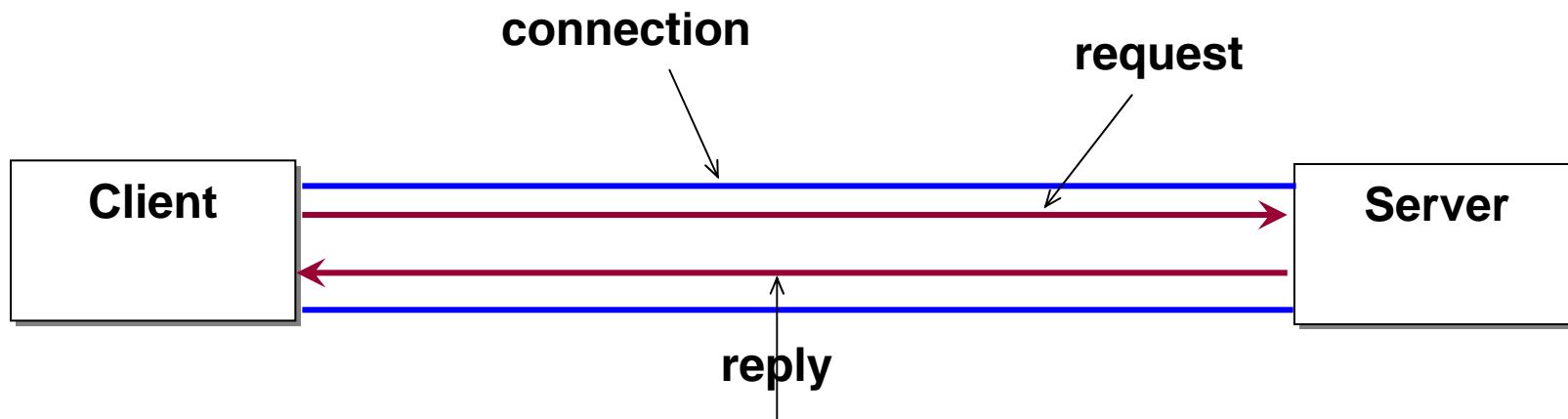


I.2 Mô hình hoạt động client/server dùng socket

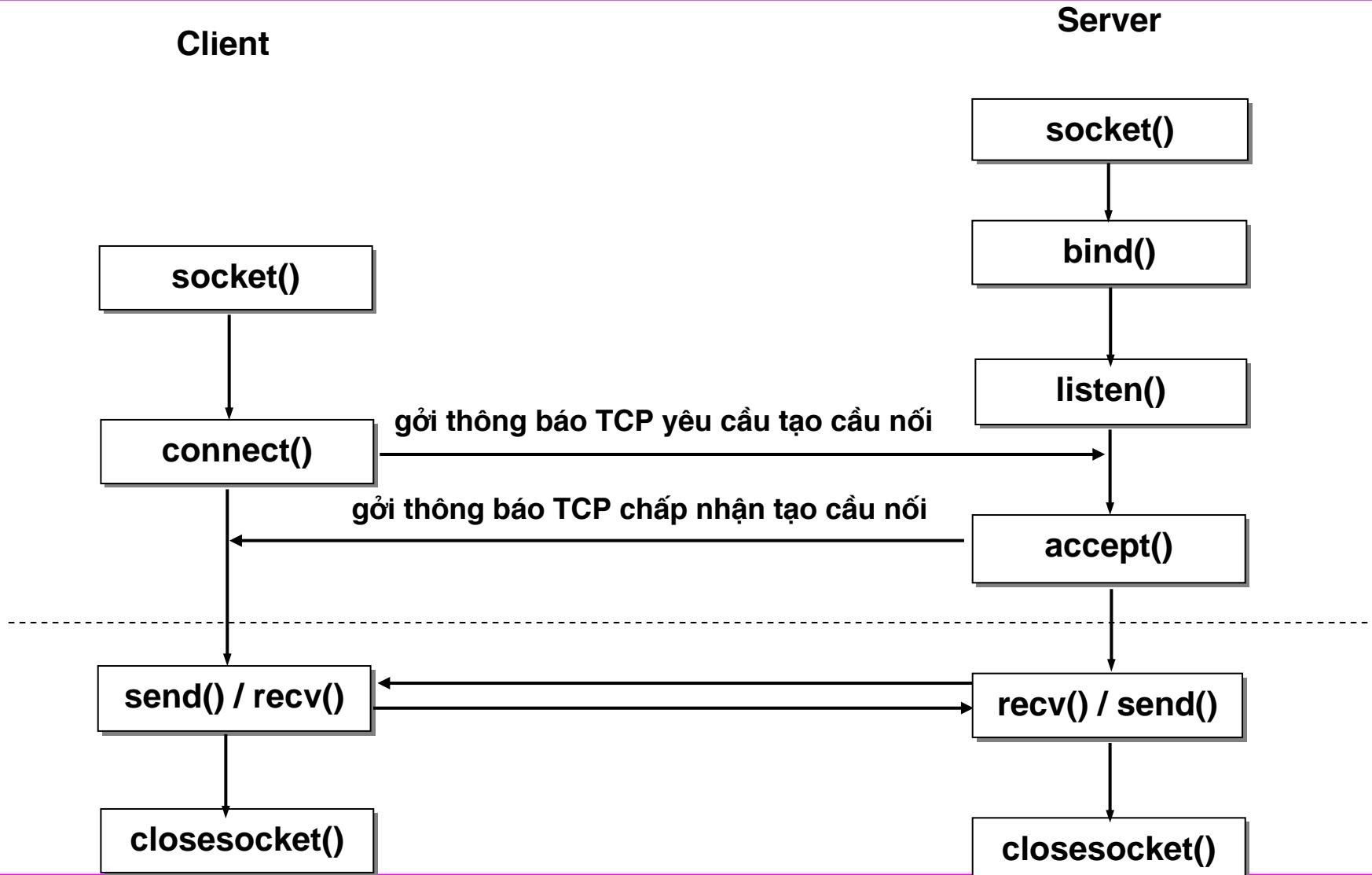
Hai ứng dụng client/server lúc còn độc lập nhau



Hai ứng dụng client/server lúc giao tiếp nhau (dùng cầu nối TCP và giao thức request/reply)



Mô hình hoạt động client/server dùng socket



I.3 Đặc tả các hàm Winsock

- ❑ Để biết chi tiết và chính xác về đặc tả sử dụng & tính năng của các hàm trong thư viện Winsock, bạn nên đọc trang Web tương ứng trong đĩa CD MSDN của Microsoft (hoặc vào website <http://www.msn.com> của Microsoft).
- ❑ Trong các slide ở mục I.3, chúng tôi chỉ cố gắng tóm tắt lại các điểm cơ bản nhất về đặc tả sử dụng cũng như tính năng của các hàm winsock thiết yếu nhất.



Đặc tả hàm socket()

- Chức năng : tạo record socket để chứa các thông tin về cổng giao tiếp của ứng dụng.

SOCKET socket (int af, int type, int protocol);

af : Họ địa chỉ, thường là AF_INET : Internet

type : Kiểu socket (SOCK_STREAM, SOCK_DGRAM)

protocol : giao thức được dùng, default = 0

return - INVALID_SOCKET : error

- handle của socket vừa được tạo, ứng dụng sẽ lưu giữ handle này để gọi hàm dịch vụ của socket khi cần.

Thí dụ :

```
// Tạo socket mới, nếu thất bại báo sai  
ser_sock=socket(AF_INET,SOCK_STREAM,0);  
if(ser_sock==INVALID_SOCKET) {  
    MessageBox("Không tạo được socket");  
    return TRUE;  
}
```



Đặc tả hàm bind()

- Chức năng : khởi động các thông tin trạng thái ban đầu cho socket server.

int bind (SOCKET s, const struct sockaddr FAR* name, int namelen);

name : record chứa thông tin cần khởi động

namelen : độ dài của record "name", return SOCK_ERROR nếu bị lỗi

Thí dụ :

```
// thiết lập địa chỉ điểm đầu mút và bind nó với socket
SOCKADDR_IN local_addr;
local_addr.sin_family=AF_INET;
local_addr.sin_port=256;
local_addr.sin_addr.s_addr=INADDR_ANY;
if(bind(ser_sock,(LPSOCKADDR)&local_addr,sizeof(local_addr))==SOCKET_ERROR) {
    MessageBox("Không bind socket được");
    return TRUE;
}
```



Đặc tả cấu trúc sockaddr_in

- Chức năng : được dùng trong tham số của nhiều hàm socket.

```
struct sockaddr_in {  
    short sin_family;          // họ socket Internet  
    unsigned short sin_port;   // cổng giao tiếp  
    struct in_addr sin_addr;   // địa chỉ IP của máy  
    char sin_zero[8];          // 8 byte 0  
};
```



Đặc tả hàm listen()

❑ Chức năng : khai báo độ dài hàng chờ cho các yêu cầu nối kết.

int listen (SOCKET s, int backlog);

backlog = độ dài hàng chờ chứa các yêu cầu nối kết (nên dùng hằng mặc định SOMAXCONN)

return < 0 error

Thí dụ :

```
// Khai báo độ dài hàng chờ chứa các yêu cầu kết nối  
if (listen(ser_sock,SOMAXCONN)==SOCKET_ERROR) {  
    MessageBox("Không listen được");  
    return TRUE;  
}
```



Đặc tả hàm accept ()

- Chức năng : lắng nghe và phục vụ yêu cầu kết nối.

SOCKET accept (SOCKET s, struct sockaddr FAR* addr, int FAR* addrlen);

addr : record chứa thông tin về cổng từ xa yêu cầu kết nối

addrlen : độ dài record "addr"

return handle socket phục vụ giao tiếp với client tương ứng

Thí dụ :

```
SOCKADDR_IN remote_addr;
```

```
SOCKET sock;
```

```
// Lắng nghe và phục vụ yêu cầu kết nối
```

```
int len=sizeof(remote_addr);
```

```
sock=accept(ser_sock,(LPSOCKADDR)&remote_addr,&len);
```

```
if(sock==INVALID_SOCKET) {
```

```
    MessageBox("Không accept được");
```

```
    return;
```

```
}
```



Đặc tả hàm connect ()

- Chức năng : yêu cầu tạo cầu nối tới server.

```
int connect (SOCKET s, const struct sockaddr FAR* name,  
           int namelen );
```

s : socket client

name : record chứa thông tin về cổng giao tiếp từ xa cần nối kết

namelen : độ dài của vùng name

return <0 : error

Thí dụ :

```
// thiết lập địa chỉ cổng giao tiếp của server từ xa  
SOCKADDR_IN ser_addr;  
ser_addr.sin_family=AF_INET;  
ser_addr.sin_port=condlg.m_port;  
if ('0'<=condlg.m_ipaddr[0] && condlg.m_ipaddr[0]<='9') // địa chỉ IP  
    ser_addr.sin_addr.s_addr=inet_addr(condlg.m_ipaddr);
```



Đặc tả hàm connect (tt)

```
else { // tên gọi nhớ của server
    PHOSTENT phe = gethostbyname(condlg.m_ipaddr);
    char szTemp[128];
    if (phe == NULL) {
        wsprintf(szTemp,"Khong co may '%s'", condlg.m_ipaddr);
        MessageBox(szTemp);
        return;
    }
    memcpy((char FAR *)&(ser_addr.sin_addr), phe->h_addr, phe->h_length);
}
// Yêu cầu kết nối với server
if(connect(sock,(LPSOCKADDR)&ser_addr,sizeof(ser_addr))==SOCKET_ERROR)
{
    MessageBox("Khong ket noi duoc");
    return;
}
```



Đặc tả hàm send ()

- Chức năng : gởi 1 chuỗi byte ra cầu nối để đến đối tác.

int send (SOCKET s, const char FAR * buf, int len, int flags);

buf : pointer tới bộ đệm dữ liệu

len : độ dài bộ đệm

flags : NO_FLAGS_SET, return số byte gởi được thực sự

Thí dụ :

```
CString mesg;
```

```
// Xây dựng thông báo cần gởi
```

```
mesg = _T("LOGIN ") + condlg.m_groupname + _T(",") +  
    condlg.m_username;
```

```
if (send(sock,mesg,strlen(mesg),0) <0) {
```

```
    MessageBox("Không gởi được lệnh LOGIN");
```

```
    return;
```

```
}
```



Đặc tả hàm recv ()

- Chức năng : chờ nhận thông tin từ xa gửi tới.

int recv (SOCKET s, char FAR* buf, int len, int flags);

buf : pointer tới bộ đệm dữ liệu nhận

len : độ dài bộ đệm

flags : NO_FLAGS_SET

return SOCKET_ERROR : error

Thí dụ :

```
status = recv(sock, mesg, MSG_LENGTH, 0);
```

```
if (status==0) return;
```

```
// tiếp tục xử lý thông tin nhận được
```

```
...
```



Đặc tả hàm closesocket ()

- Chức năng : yêu cầu đóng và xóa socket.

int closesocket (SOCKET s);

Thí dụ :

```
LRESULT CMiniChatClientDlg::WindowProc(UINT message, WPARAM wParam,  
LPARAM lParam) {  
switch (message) {  
case WSA_RDCLOSE :  
    if (WSAGETSELECTEVENT(lParam) == FD_READ) {  
        status = recv(sock, mesg, MSG_LENGTH, 0);  
        if (status==0) return;  
        // tiếp tục xử lý thông tin nhận được  
    } else // đóng socket  
        closesocket (wParam);  
    return 1;  
}  
return CDialog::WindowProc(message, wParam, lParam);  
}
```



I.4 Vấn đề bất đồng bộ trong gửi/nhận thông tin

Hãy quan sát 2 hàm accept() và recv(), mặc định chúng có hành vi đặc biệt : bị kẹt nếu đối tác không giao tiếp với mình (blocking). Để tránh bị kẹt khi gọi 1 trong 2 hàm này, người ta dùng 1 trong 4 phương pháp sau :

1. dùng socket ở chế độ non-blocking :

```
int non_block = 1;  
rc = ioctlsocket(s, FIONBIO, &non_block);  
if (rc == SOCKET_ERROR) {  
    dos_net_perror("ioctlsocket() call failed");  
    closesocket(s);  
    exit(1);  
}
```

//trên unix, thay vì gọi hàm ioctlsocket, ta gọi hàm :

```
//fcntl(sockfd, F_SETFL, O_NONBLOCK);
```

Ở chế độ non-blocking, khi gọi các hàm accept(), recv(),... mà dữ liệu chưa có, hàm sẽ trả điều khiển về ngay với mã lõi tương ứng.



Vấn đề bất đồng bộ trong gửi/nhận thông tin

2. gọi hàm select() để khảo sát trạng thái sẵn sàng đọc, sẵn sàng ghi,... của 1 số socket xác định.

```
int select (int nfds,  
fd_set FAR *readfds,  
fd_set FAR *writefds,  
fd_set FAR *exceptfds,  
const struct timeval FAR *timeout);
```

Một số macro thêm/xóa 1 socket vào/ra 1 tập hợp :

FD_SET(socket,&readfds) : thêm socket vào tập hợp

FD_ISSET(socket, &readfds) : kiểm tra xem socket có trong tập hợp

FD_CLR(socket,&readfds) : xóa socket khỏi tập hợp



Vấn đề bất đồng bộ trong gói/nhận thông tin

3. dùng cơ chế lập trình Multi-thread : thread chính sẽ tạo thread con và nhờ thread con này thực hiện hàm accept() hay recv(). Thread con bị kẹt khi gọi hàm nhưng thread cha thì vẫn chạy bình thường.
4. Tạo thông báo Windows kết hợp với sự kiện liên quan đến cổng giao tiếp. Khi có sự kiện qui định, Windows sẽ gửi thông báo về cửa sổ ứng dụng để kích khởi hàm xử lý sự kiện, code của hàm này sẽ kiểm tra sự kiện, và gọi hàm accept() hay recv(). 2 hàm winsock được dùng thông thường cho cơ chế này là WSAAsyncSelect() và WSAGETSELECTEVENT().



Đặc tả hàm WSAAsyncSelect ()

- Chức năng : khai báo các biến cố bất đồng bộ kết hợp với socket.

**int WSAAsyncSelect (SOCKET s, HWND hWnd, unsigned int wMsg,
long lEvent);**

hWnd : cửa sổ chương trình sẽ nhận message.

wMsg : thông báo sẽ tạo ra

lEvent : tổ hợp các biến cố network sẽ gây ra thông báo.

Thí dụ :

```
// Khai báo chờ yêu cầu kết nối  
if (WSAAsyncSelect(ser_sock, m_hWnd, WSA_ACCEPT,  
FD_ACCEPT) > 0) {  
    MessageBox("Error on WSAAsyncSelect()");  
    closesocket(ser_sock);  
}
```



I.5 Thí dụ về ứng dụng mạng cơ bản

Hệ thống MiniChatter

Chức năng: cho phép nhiều user đăng ký vào các nhóm để trò chuyện với nhau.

Mô hình chọn lựa: client/server

Server: quản lý các nhóm và các user cùng nhóm, phân phối các chuỗi thông tin từ một user đến các user khác cùng nhóm...

Client: giao tiếp với user, cho phép họ đăng ký nhóm, gửi/nhận thông tin lẫn nhau.



Thí dụ về ứng dụng mạng cơ bản

Định nghĩa giao thức được dùng bởi hệ thống MiniChatter :

Gồm 5 thông báo request sau :

1. Lệnh GLIST <LF>
2. Lệnh ULIST <LF>
3. Lệnh LOGIN <tên group> "," <tên user> <LF>
4. Lệnh SEND <string> <LF>
5. Lệnh LOGOU <LF>

Và định dạng thông báo reply cho tất cả các request :

n <chuỗi dữ liệu phụ trợ kèm theo>

n = 1 : thành công, n = 0 : thất bại.

Chương 2 sẽ trình bày qui trình viết hệ thống MiniChatter bằng VC++ và dùng kỹ thuật xử lý sự kiện.

Chương 3 sẽ trình bày qui trình viết hệ thống MiniChatter bằng Java (qua môi trường JBuilder) và dùng kỹ thuật xử lý multi-thread.



MÔN LẬP TRÌNH MẠNG

Chương 2

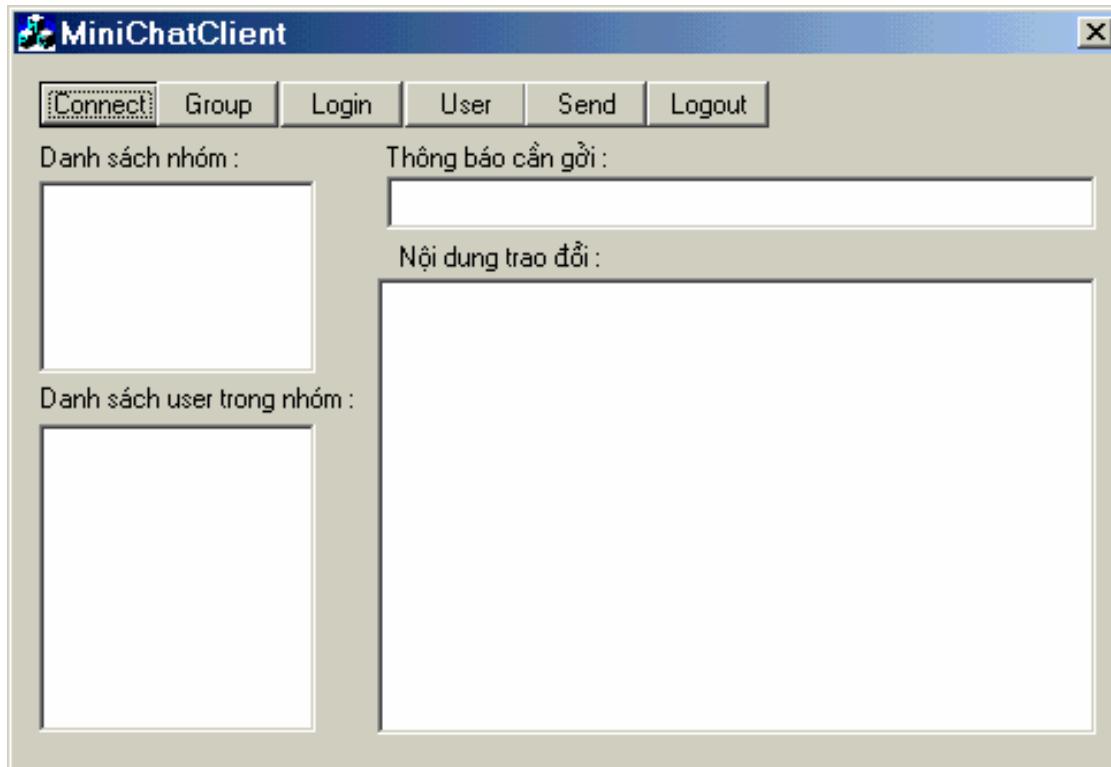
Viết hệ thống MiniChatter bằng VC++ & dùng kỹ thuật xử lý sự kiện

- 2.1 Xây dựng module client MiniChatter
- 2.2 Xây dựng module server MiniChatter

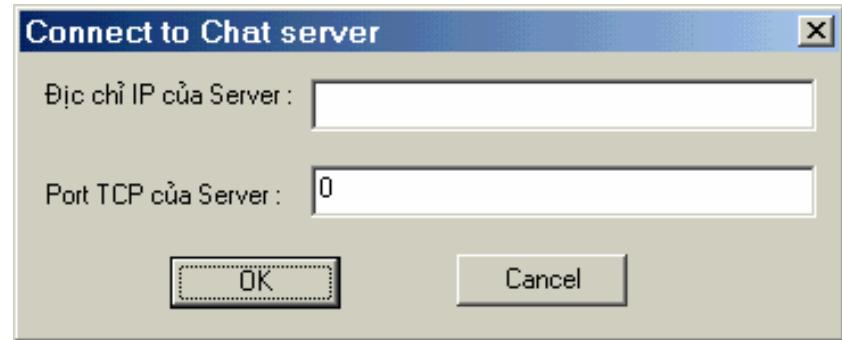


2.1 Xây dựng module client MiniChatter

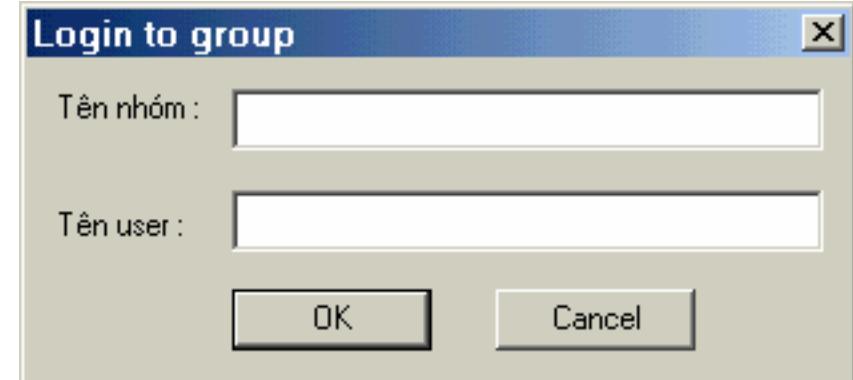
Chúng ta sẽ thiết kế giao diện cho Form ứng dụng MiniChatClient như sau :



Cửa sổ phục vụ lệnh Connect :

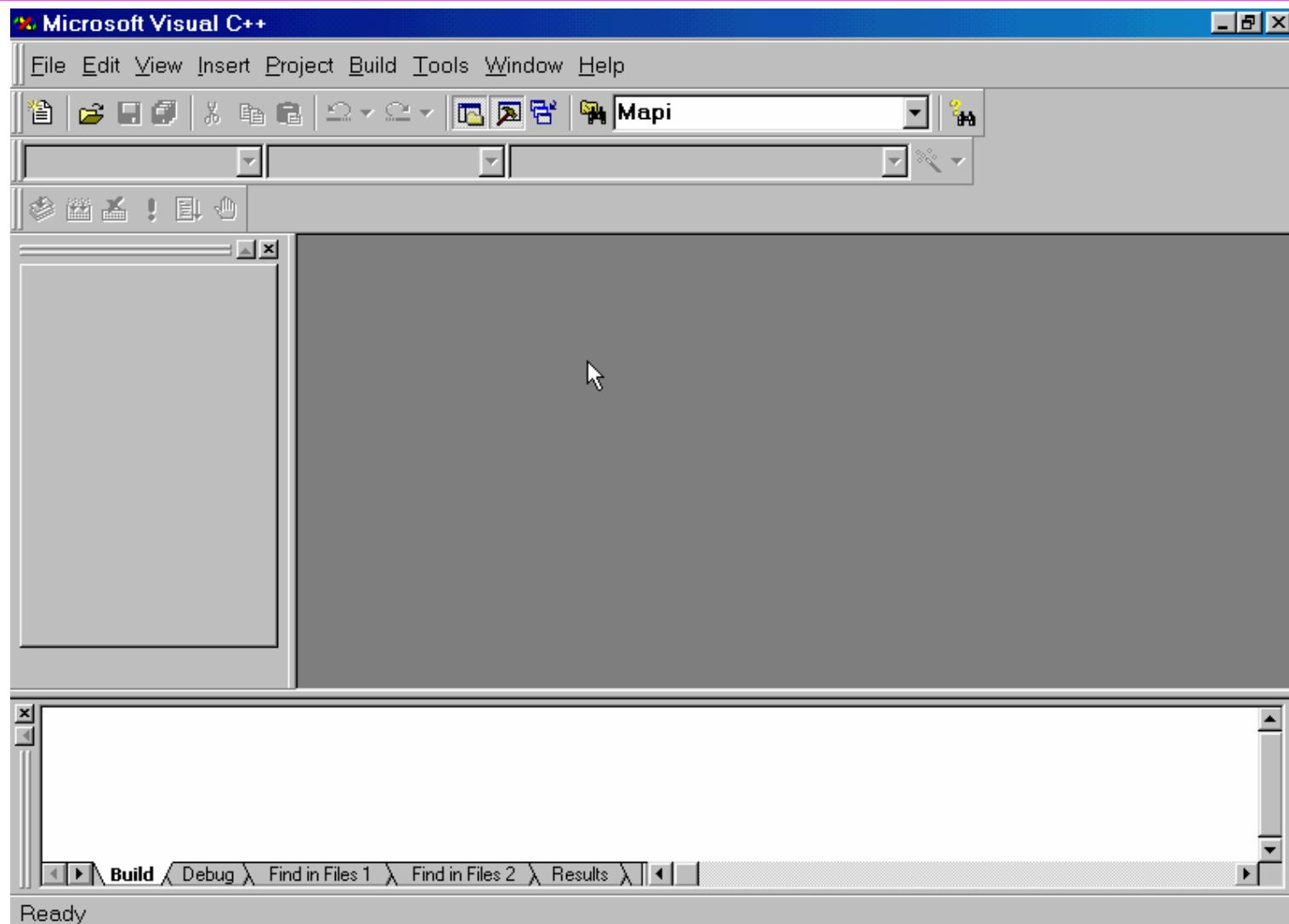


Cửa sổ phục vụ lệnh Login :



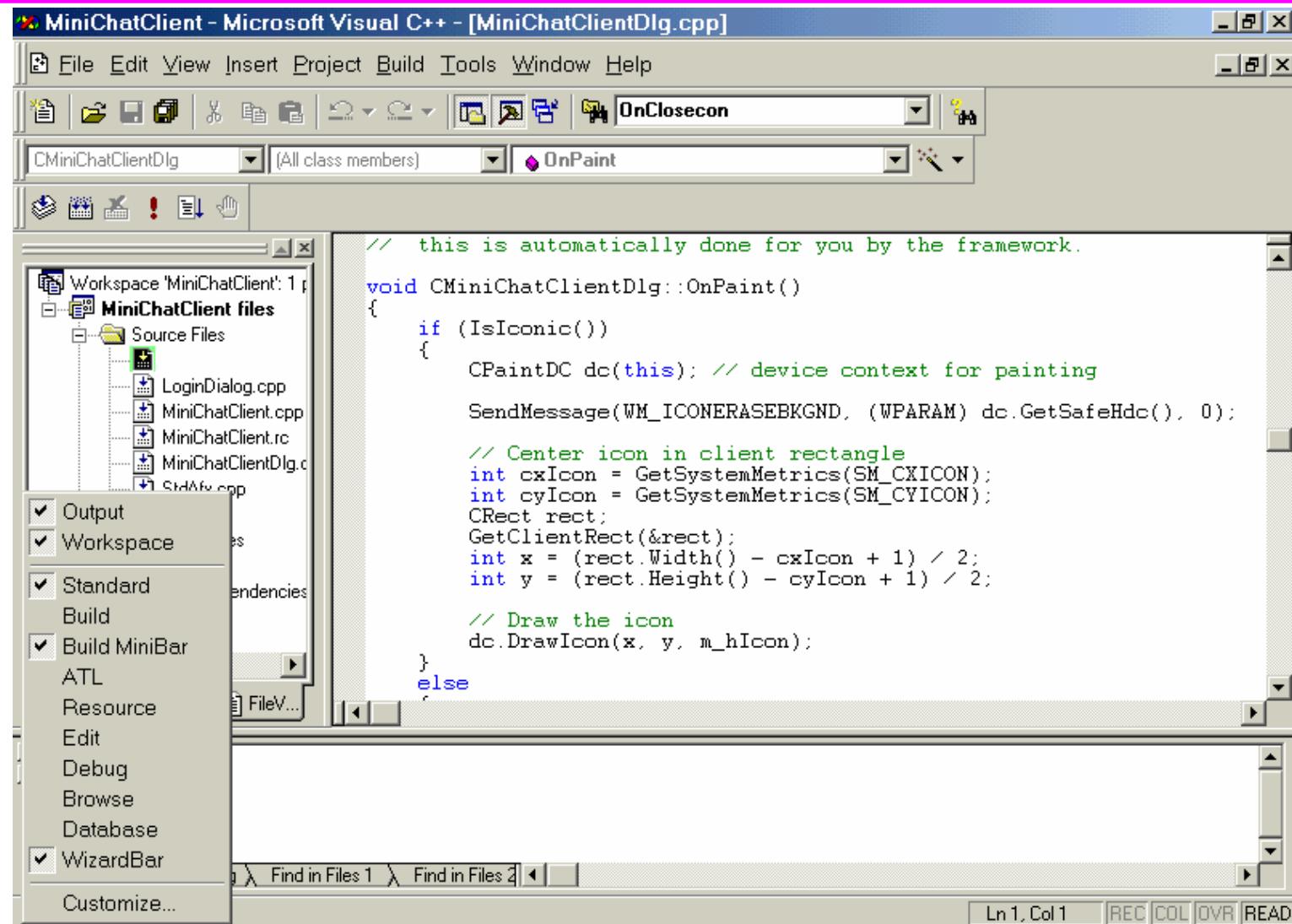
Xây dựng module client MiniChatter

1. Chạy Visual C++ 6.0 từ Windows (thí dụ chọn mục Start Programs Microsoft Visual Studio 6.0/Microsoft Visual C++ 6.0). Màn hình của VC++ được hiển thị như sau :



Xây dựng module client MiniChatter

2. Có nhiều cửa sổ làm việc khác nhau, ta có thể bật/tắt từng cửa sổ theo nhu cầu từng thời điểm.



The screenshot shows the Microsoft Visual Studio interface for the 'MiniChatClient' project. The title bar reads 'MiniChatClient - Microsoft Visual C++ - [MiniChatClientDlg.cpp]'. The menu bar includes File, Edit, View, Insert, Project, Build, Tools, Window, Help. The toolbar has icons for New, Open, Save, Print, etc. The status bar at the bottom shows 'Ln 1, Col 1' and buttons for REC, COL, DVR, READ.

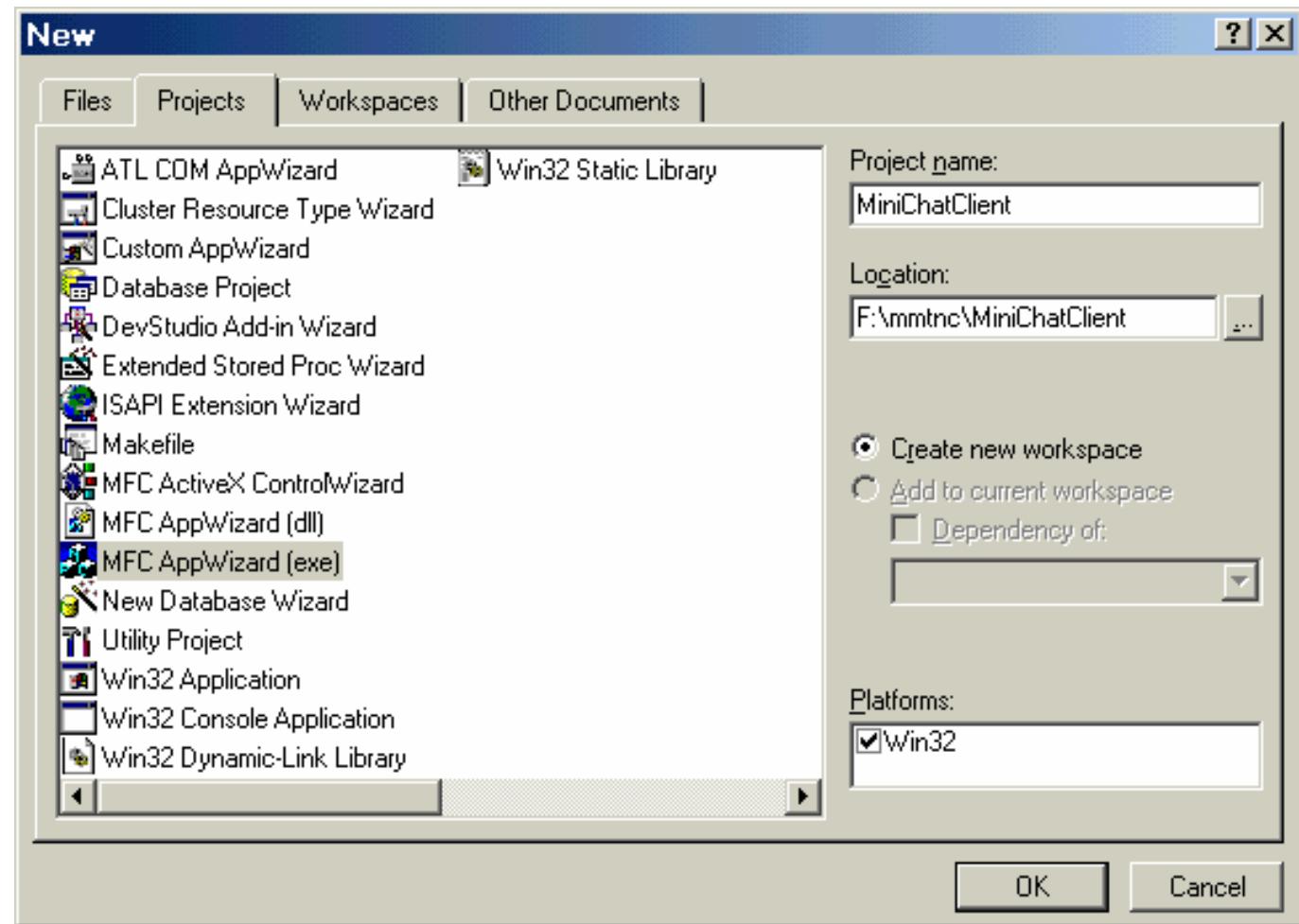
The code editor displays the implementation of the `OnPaint()` method:

```
// this is automatically done for you by the framework.  
void CMiniChatClientDlg::OnPaint()  
{  
    if (IsIconic())  
    {  
        CPaintDC dc(this); // device context for painting  
  
        SendMessage(WM_ICONERASEBKGND, (WPARAM) dc.GetSafeHdc(), 0);  
  
        // Center icon in client rectangle  
        int cxIcon = GetSystemMetrics(SM_CXICON);  
        int cyIcon = GetSystemMetrics(SM_CYICON);  
        CRect rect;  
        GetClientRect(&rect);  
        int x = (rect.Width() - cxIcon + 1) / 2;  
        int y = (rect.Height() - cyIcon + 1) / 2;  
  
        // Draw the icon  
        dc.DrawIcon(x, y, m_hIcon);  
    }  
    else  
    {  
        // Drawing logic for non-iconic state  
    }  
}
```



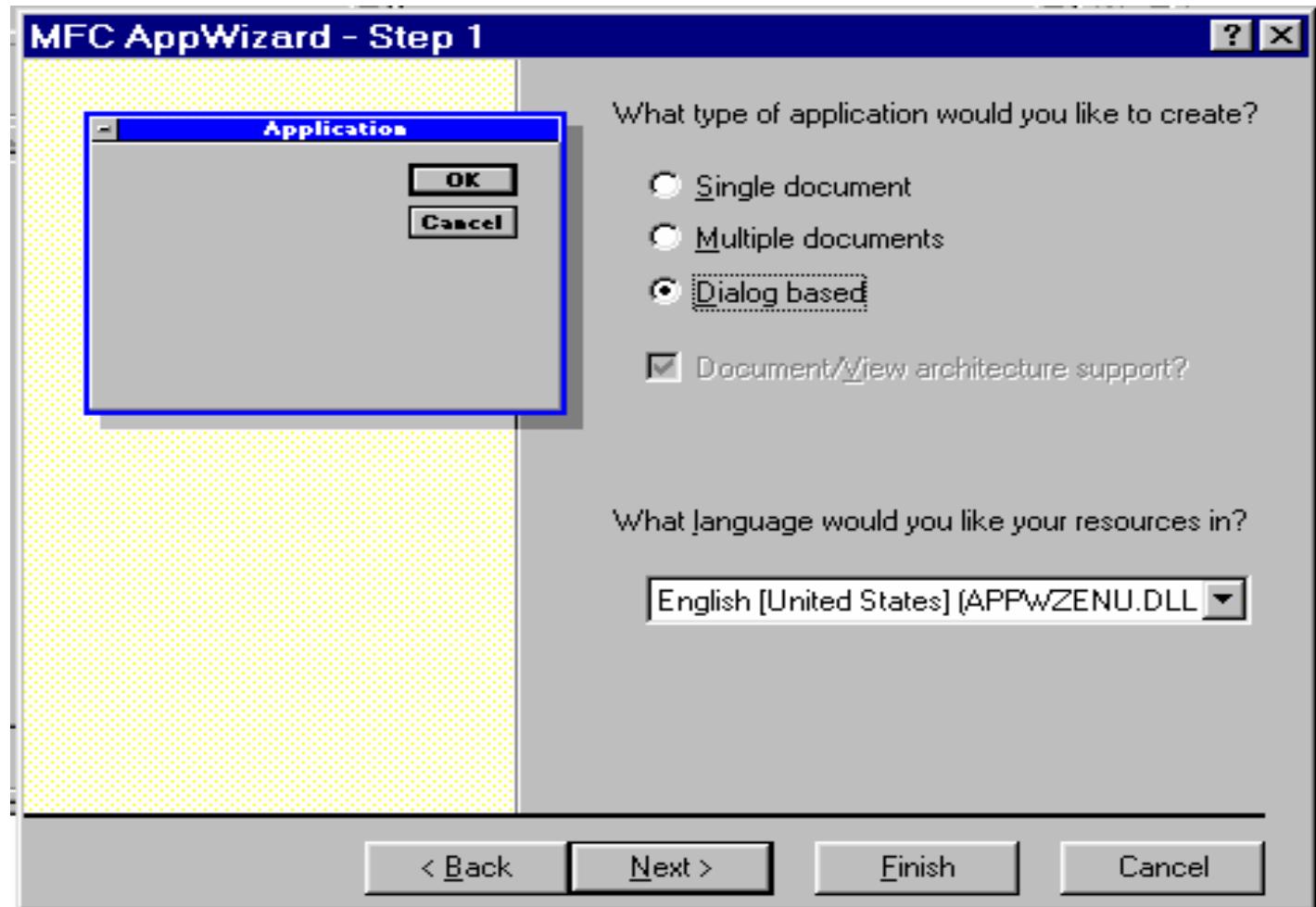
Xây dựng module client MiniChatter

3. Bước 1 : Dùng AppWizard để tạo project. Chọn menu File.New..., cửa sổ bên dưới được hiển thị. Chọn tab Projects, chọn mục MFC AppWizard (exe), chọn thư mục chứa project chương trình cần tạo ở mục "Location". Nhập tên project (MiniChatClient) vào textbox "Project Name".



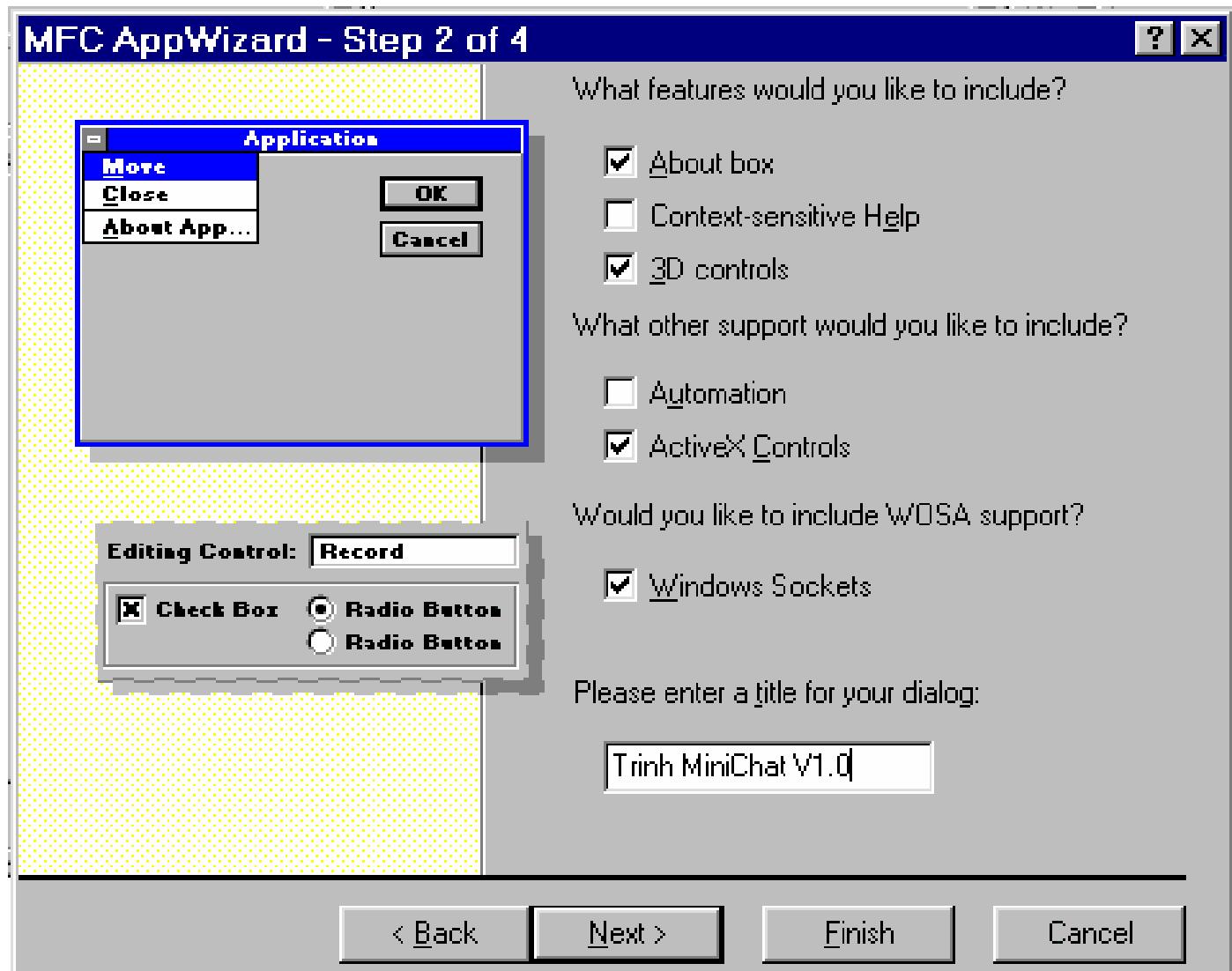
Xây dựng module client MiniChatter

4. Chọn button Ok, cửa sổ sau được hiển thị để bạn chọn style cửa sổ làm việc :



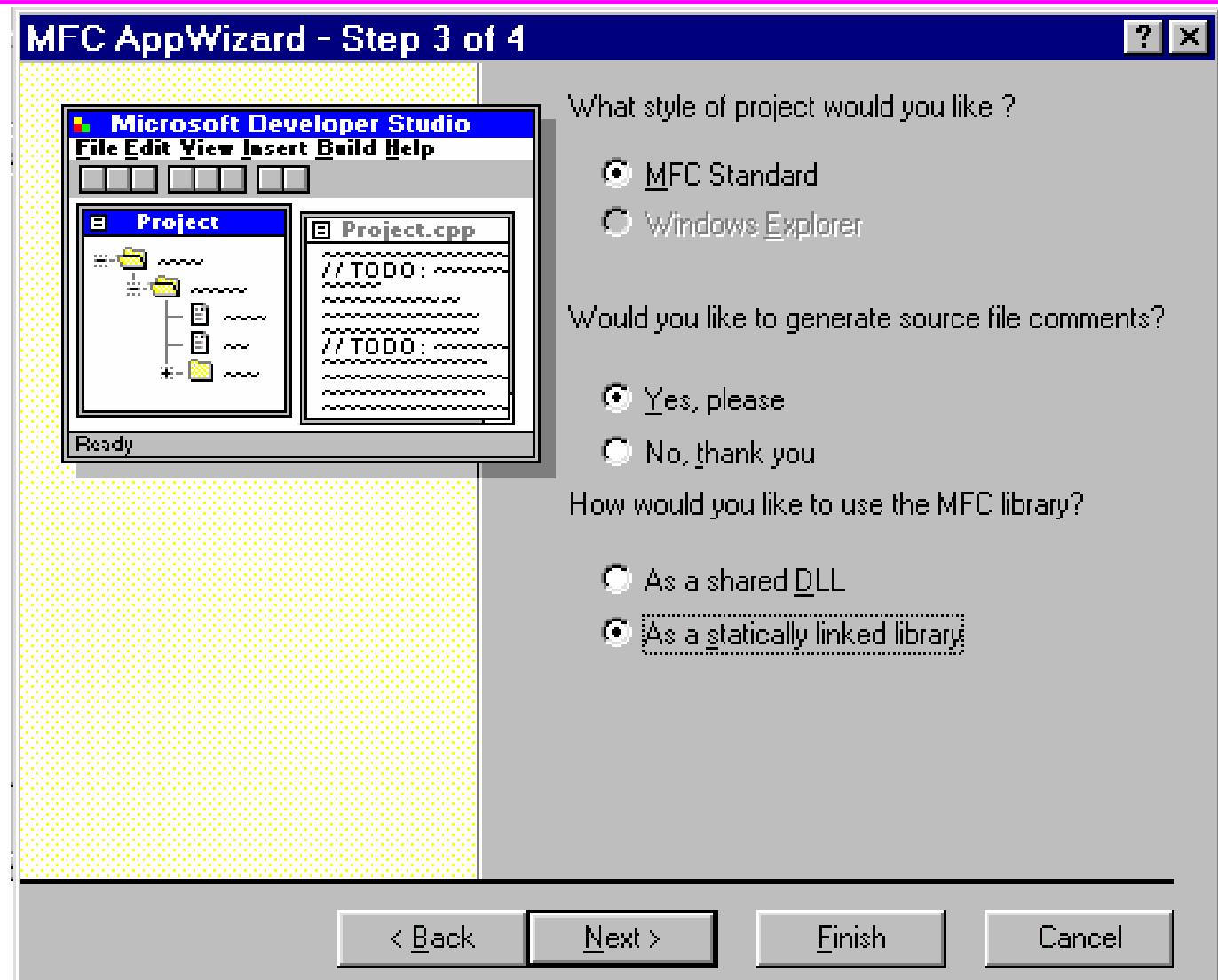
Xây dựng module client MiniChatter

5. Thí dụ chương trình của ta dùng cửa sổ dạng hộp thoại, chọn mục "Dialog based" rồi chọn button Next, bạn sẽ được cửa sổ sau :



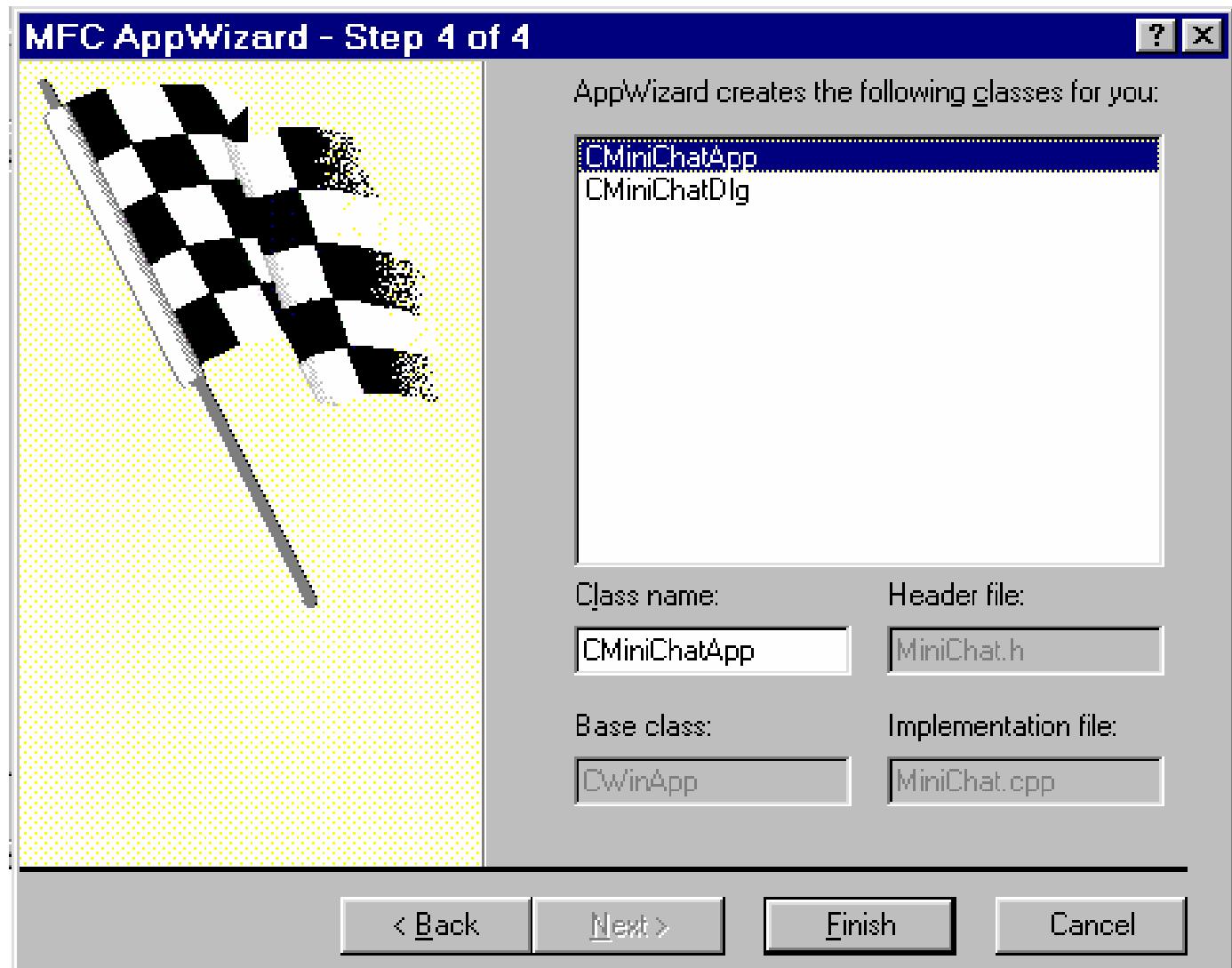
Xây dựng module client MiniChatter

6. Chọn mục "Windows Sockets", hiệu chỉnh lại Title của dialog box, rồi chọn button Next, bạn sẽ được cửa sổ sau:



Xây dựng module client MiniChatter

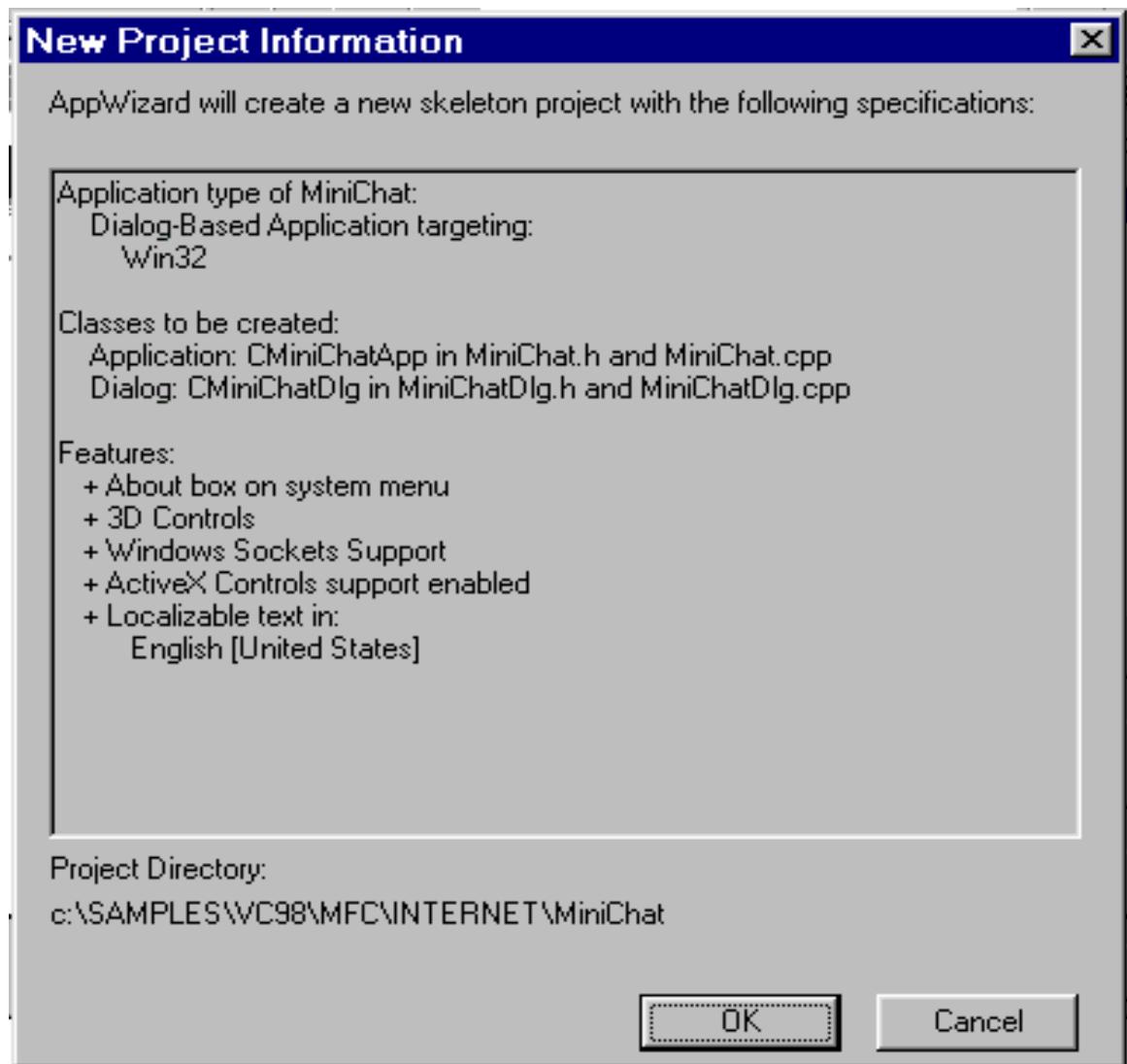
7. Chọn mục "As a statically linked library" hay "As a shared DLL" tùy yêu cầu cần liên kết tĩnh hay động ứng dụng với thư viện MFC, rồi chọn button Next, bạn sẽ được cửa sổ sau :



Xây dựng module client MiniChatter

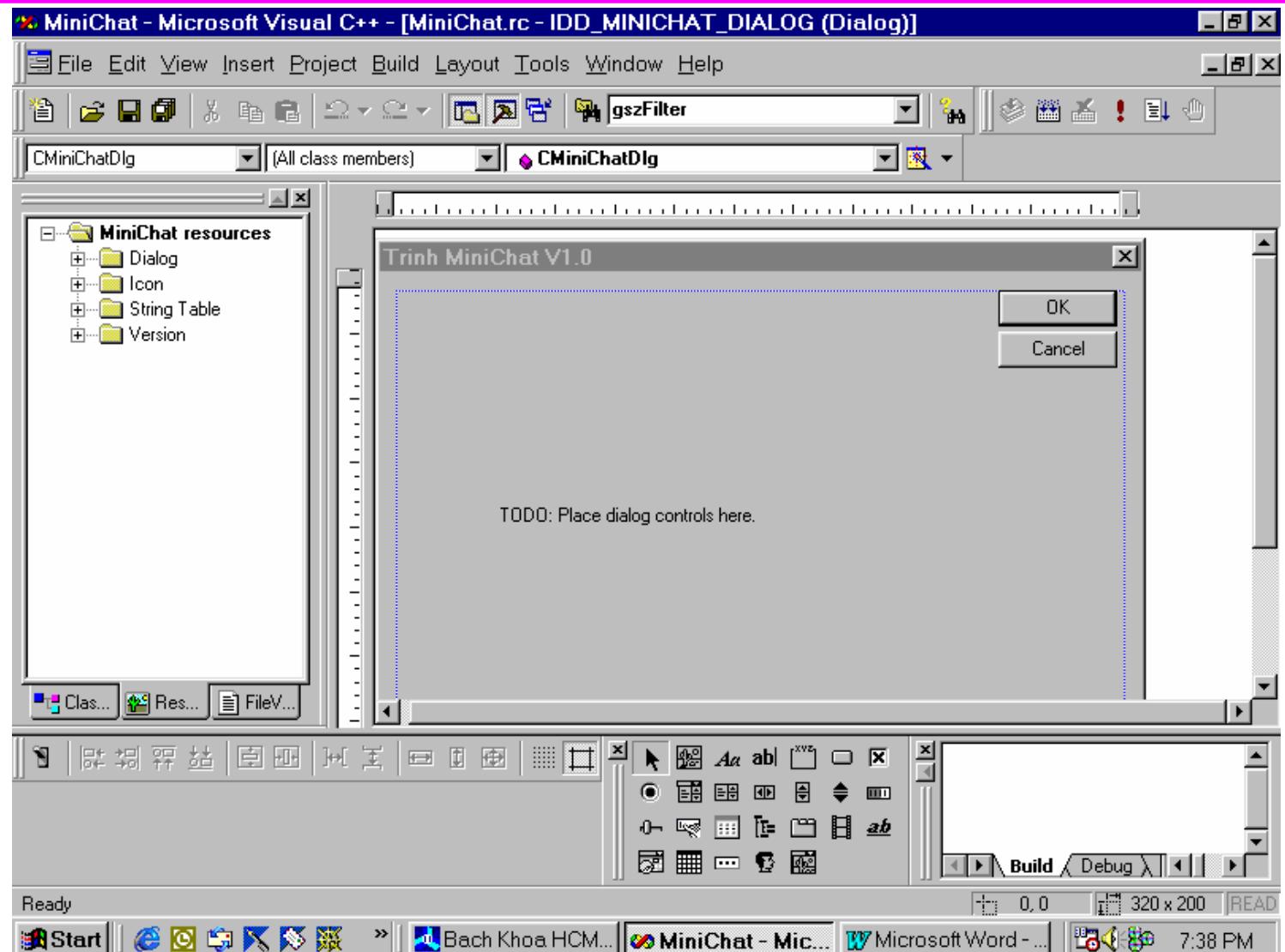
8. Chọn button

Finish, bạn sẽ được
cửa sổ sau :



Xây dựng module client MiniChatter

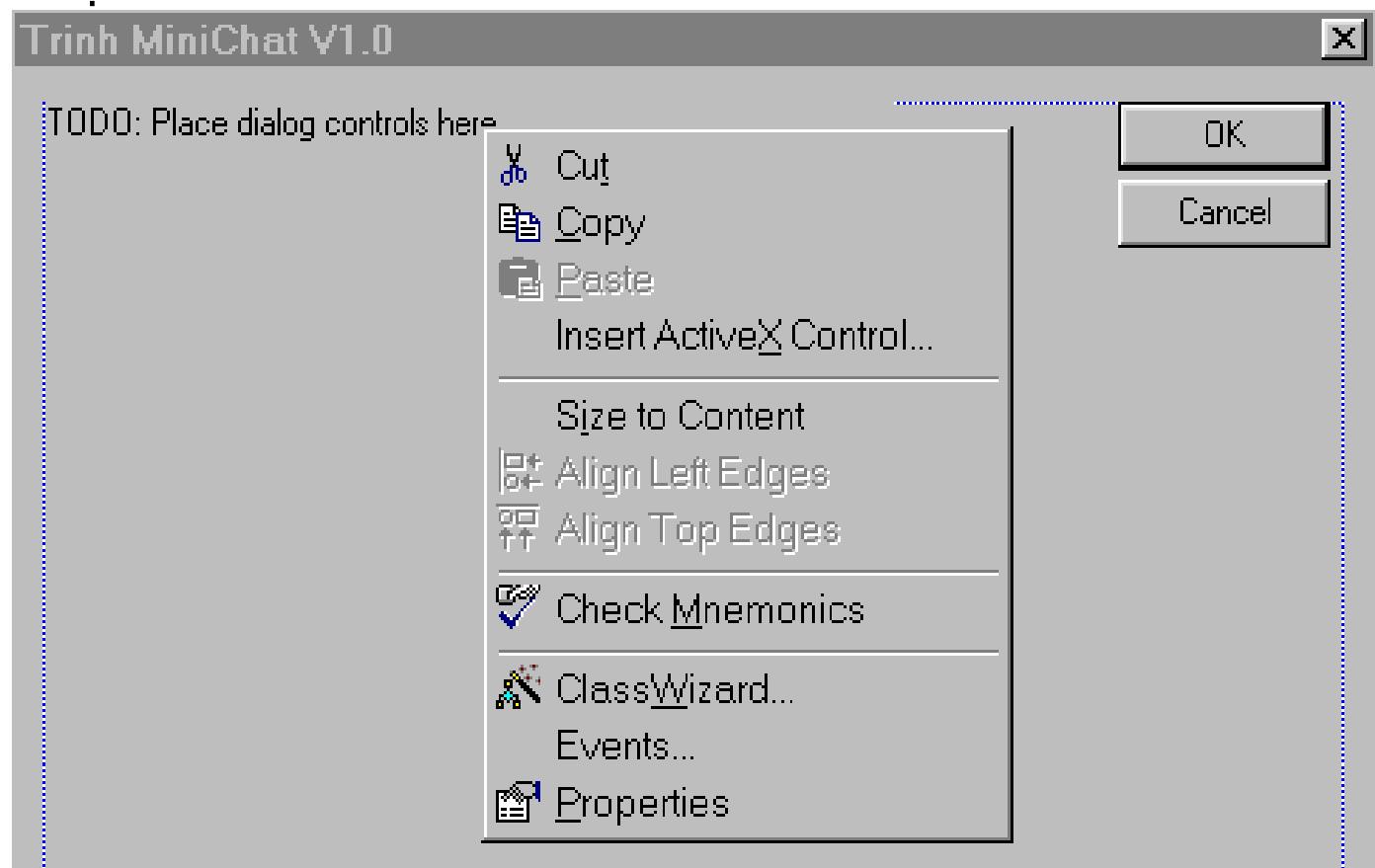
9. Chọn button Ok, màn hình cho chương trình sẽ được hiển thị như sau :



Xây dựng module client MiniChatter

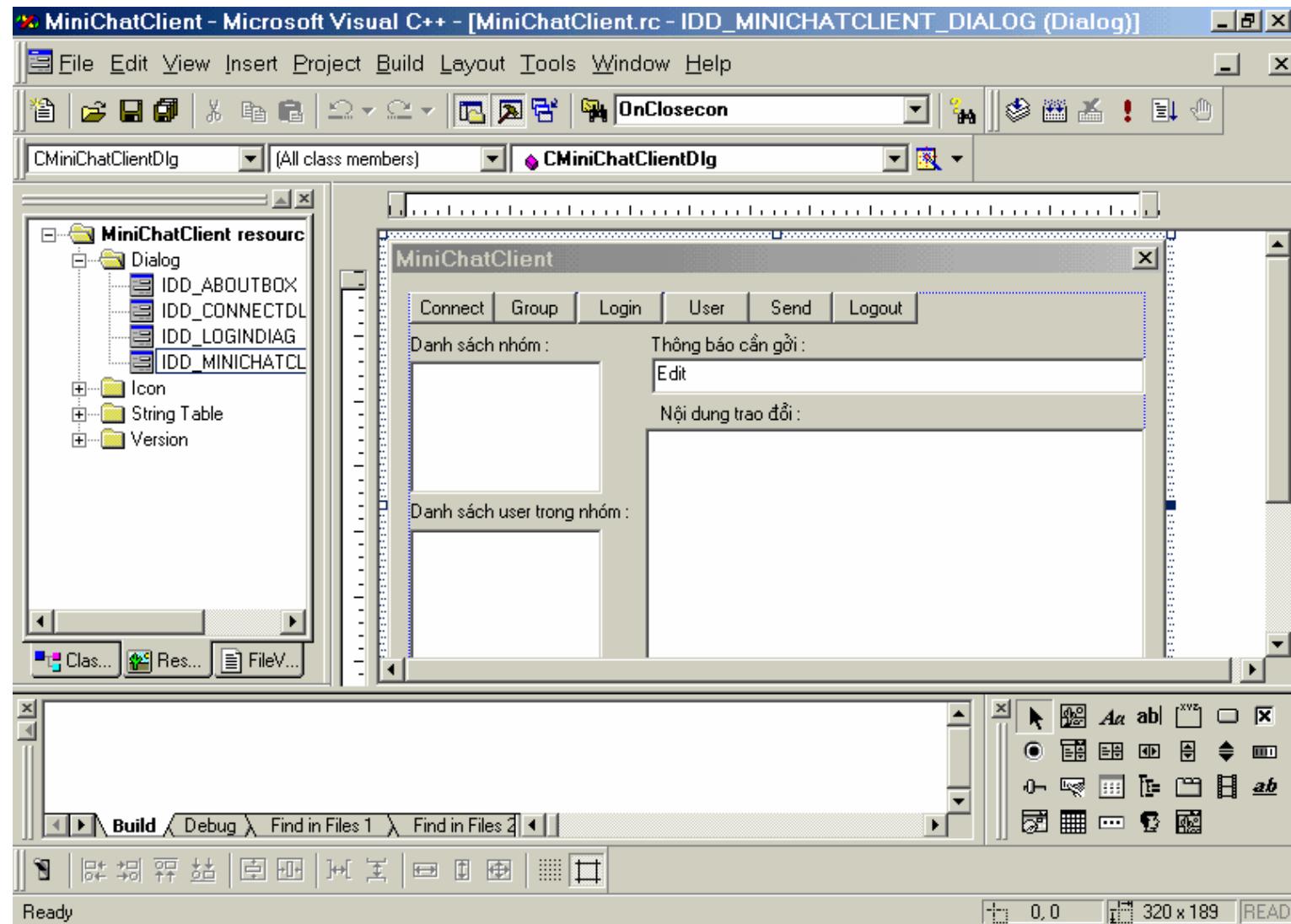
10. Bước 2 : Thiết kế trực quan các phần tử giao diện. Bắt đầu thiết kế giao diện chương trình, dùng các control định sẵn trong cửa sổ Control. Để xóa 1 control, dùng mouse chọn nó rồi ấn button "Cut".

Để thêm 1 control, dùng mouse drag nó từ cửa sổ Control vào vị trí trong dialog box. Để xem hay hiệu chỉnh các thuộc tính của một control, dời mouse tới nó rồi ấn phải chuột, chọn mục Properties để hiển thị cửa sổ thuộc tính.



Xây dựng module client MiniChatter

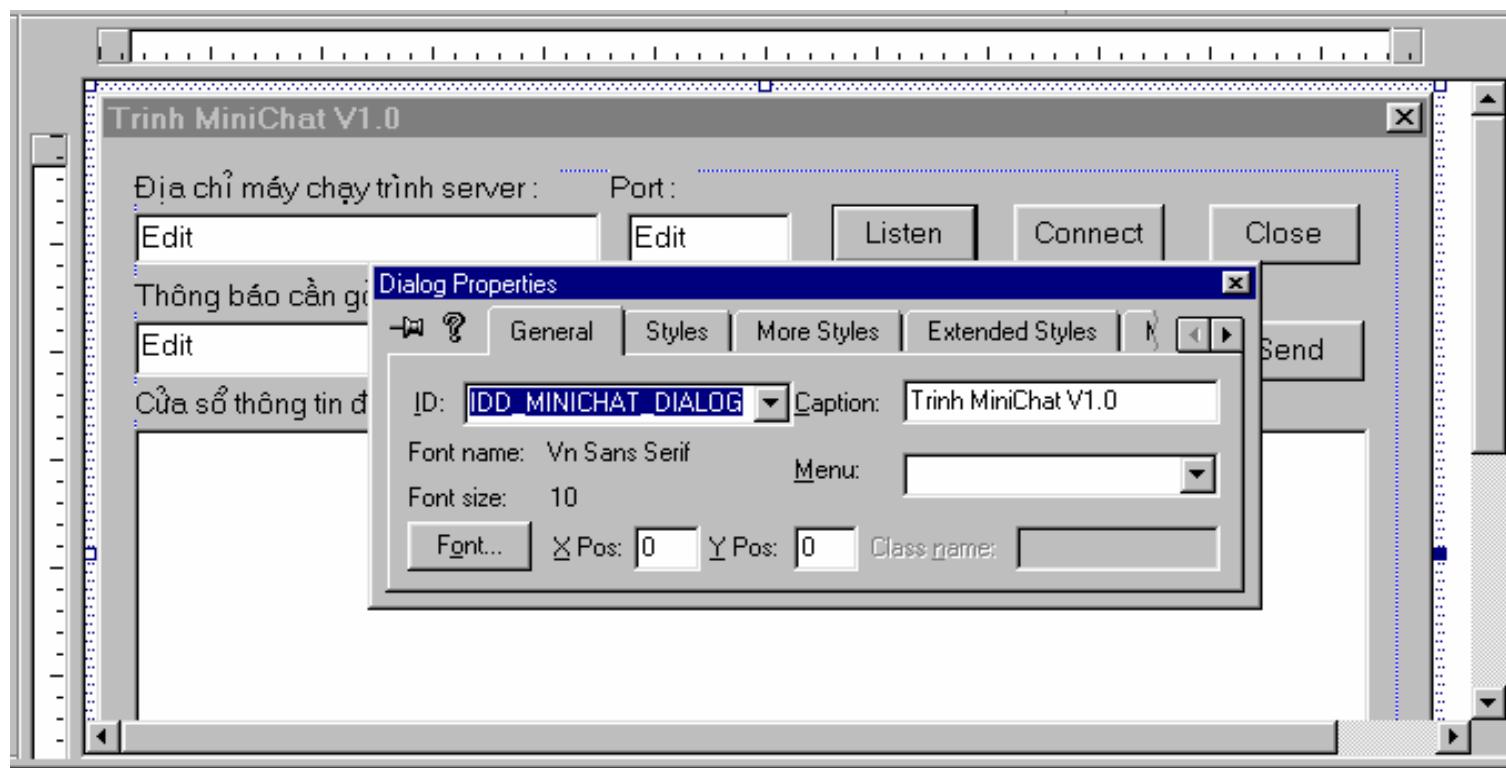
11. Dùng chuột drag các đối tượng (control) từ cửa sổ Control và đặt ở vị trí mong muốn trong dialogbox :



Xây dựng module client MiniChatter

12. Để các phần tử trong DialogBox dùng font chữ xác định (tiếng Việt), chọn dialogbox rồi ấn phải chuột.

Cửa sổ thuộc tính sẽ được hiển thị :



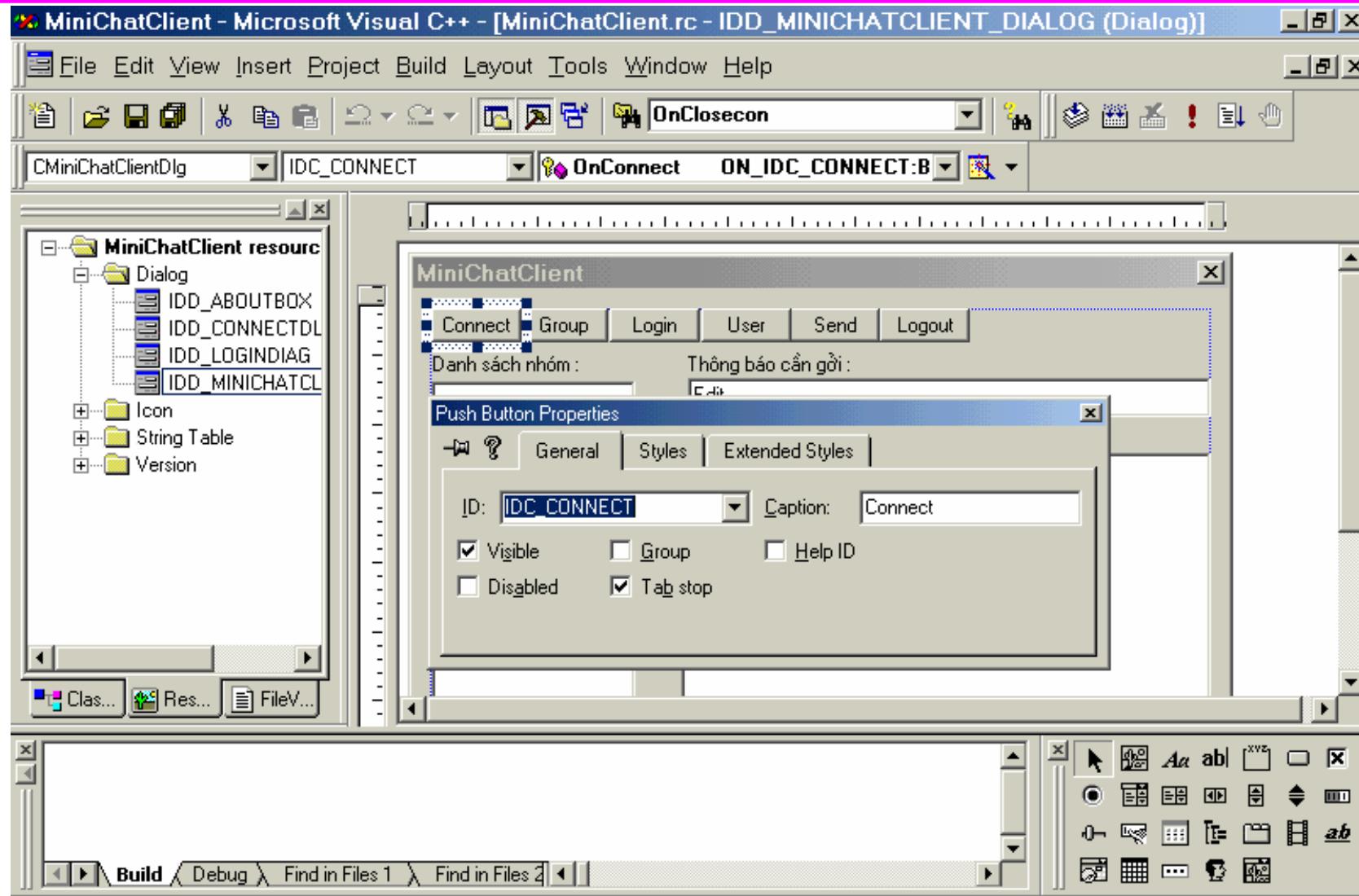
Xây dựng module client MiniChatter

13. Bước 3: Hiệu chỉnh các thuộc tính các phần tử giao diện (nhất là ID, caption).
Cụ thể như sau :

- Button "Connect" với tên IDC_CONNECT, "Group" với tên IDC_GROUPREQ, "Login" với tên IDC_LOGIN
- Button "User" với tên IDC_USERREQ, Button "Send" với tên IDC_SENDMSG, "Logout" với tên IDC_LOGOUT
- Textbox "Thông báo cần gửi" với tên IDC_MESG, Listbox "Danh sách nhóm" với tên IDC_GROUPLIST,
- Listbox "Danh sách user trong nhóm" tên IDC_USERLIST, Listbox "Nội dung trao đổi" tên IDC_CONTENT.



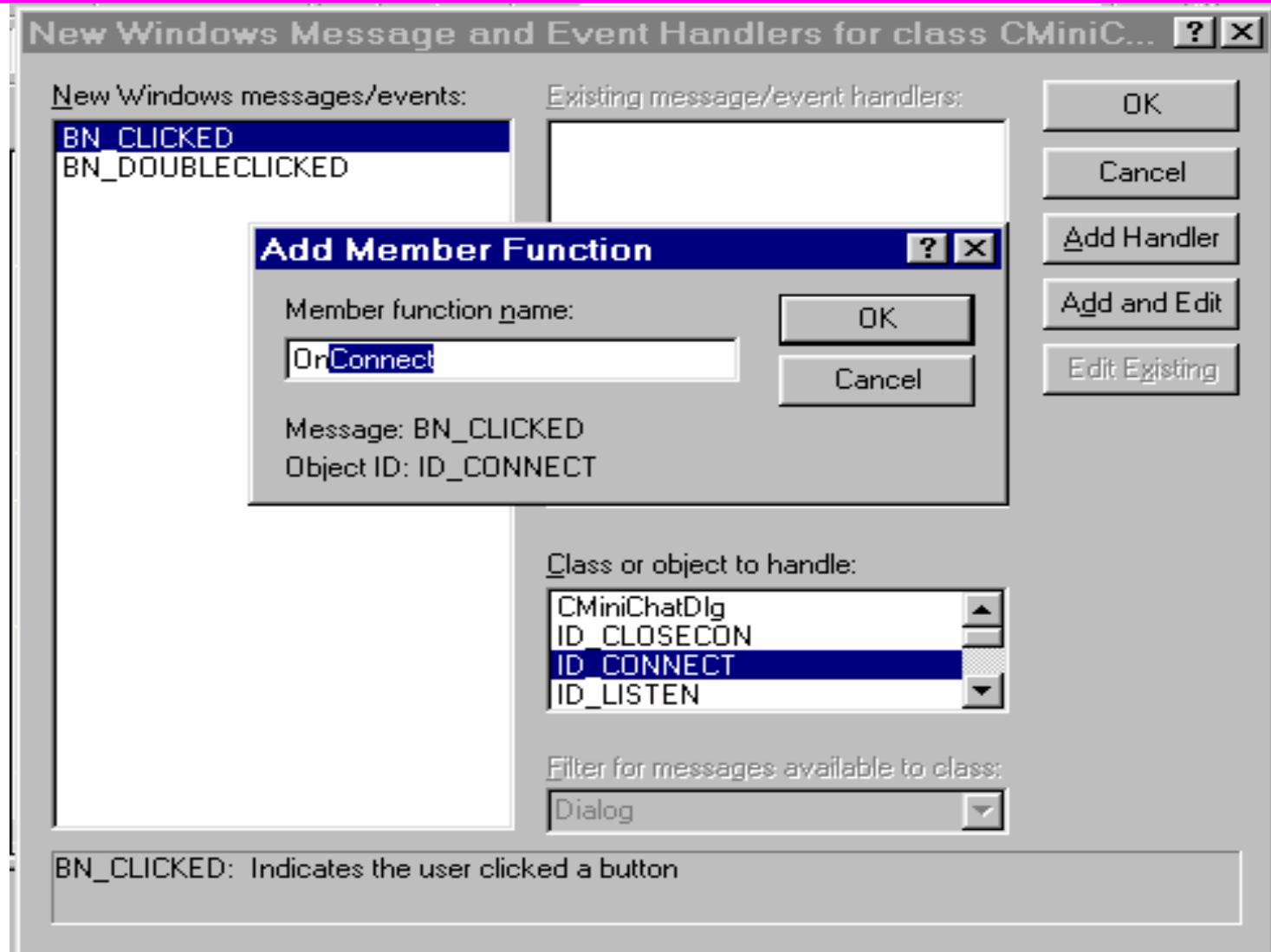
Xây dựng module client MiniChatter



Xây dựng module client MiniChatter

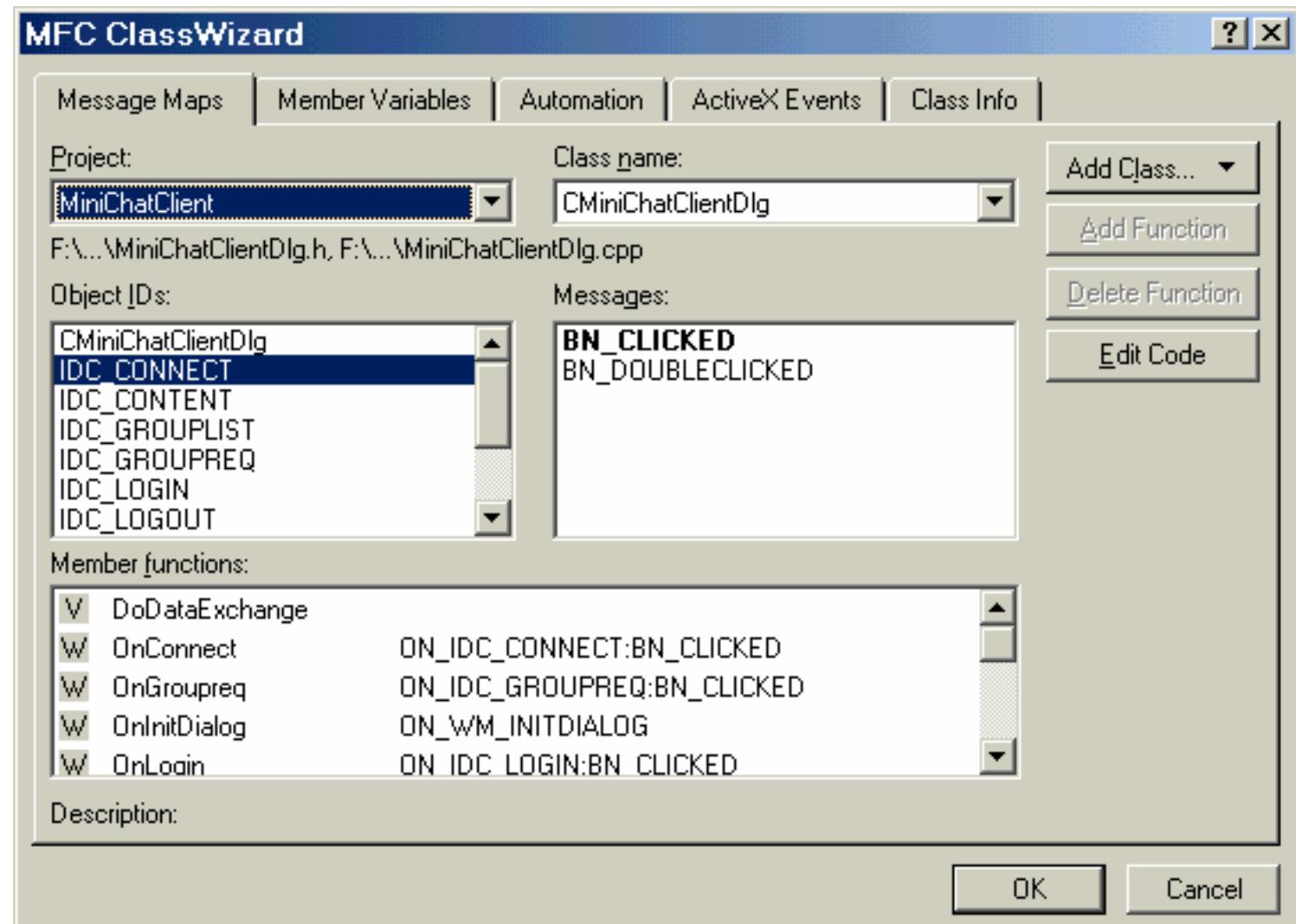
14. Bước 4: Kết hợp biến và hàm xử lý biến cố với các phần tử giao diện :Tạo hàm xử lý biến cố cho từng button bằng cách đưa cursor về button, ấn phải chuột, chọn mục Event trong cửa sổ Properties :

Cửa sổ sau sẽ xuất hiện :



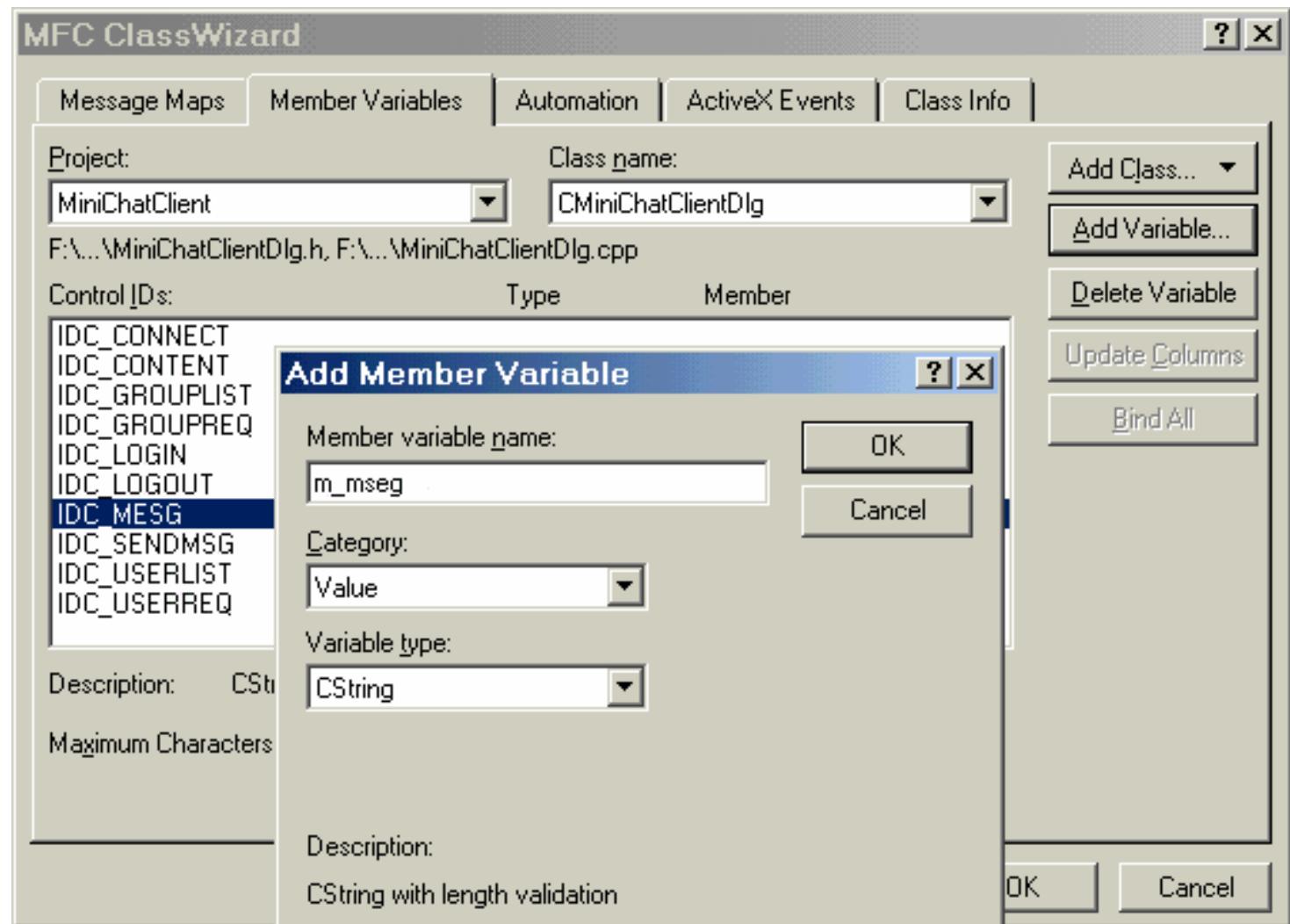
Xây dựng module client MiniChatter

15. Phương pháp chính quy để khai báo biến và hàm xử lý biến cố với các phần tử giao diện là dùng menu View.ClassWizard, cửa sổ ClassWizard sẽ xuất hiện, trang Message Maps cho phép khai báo các hàm xử lý biến cố.



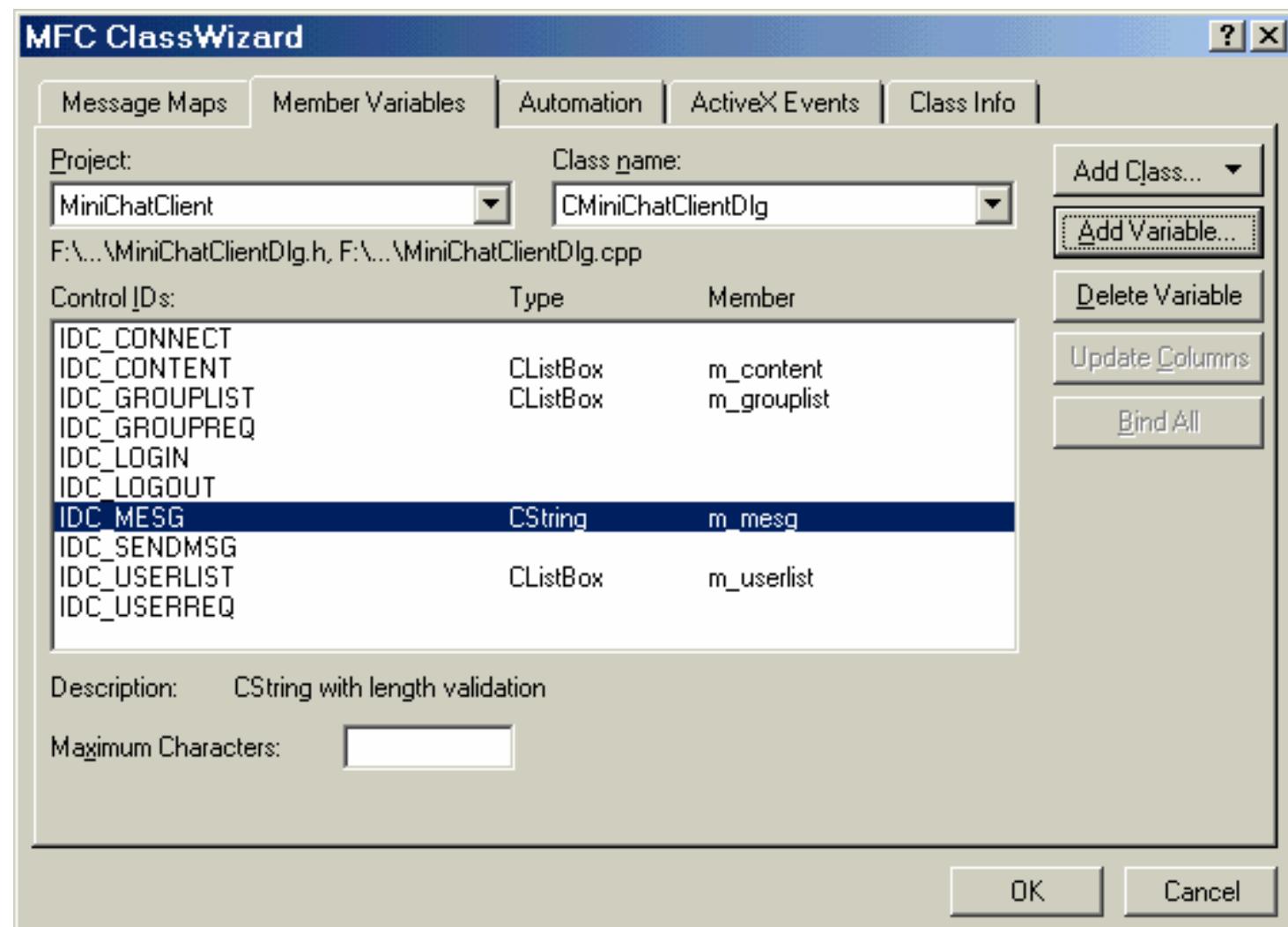
Xây dựng module client MiniChatter

16. Để tạo các biến dữ liệu kết hợp với các phần tử giao diện, chọn Project, class chứa biến, trang Member Variables, sau đó chọn từng ID phần tử rồi ấn button "Add Variable", cửa sổ sau sẽ xuất hiện :



Xây dựng module client MiniChatter

17. Kết quả sau khi tạo 4 biến kết hợp với 4 phần tử giao diện :

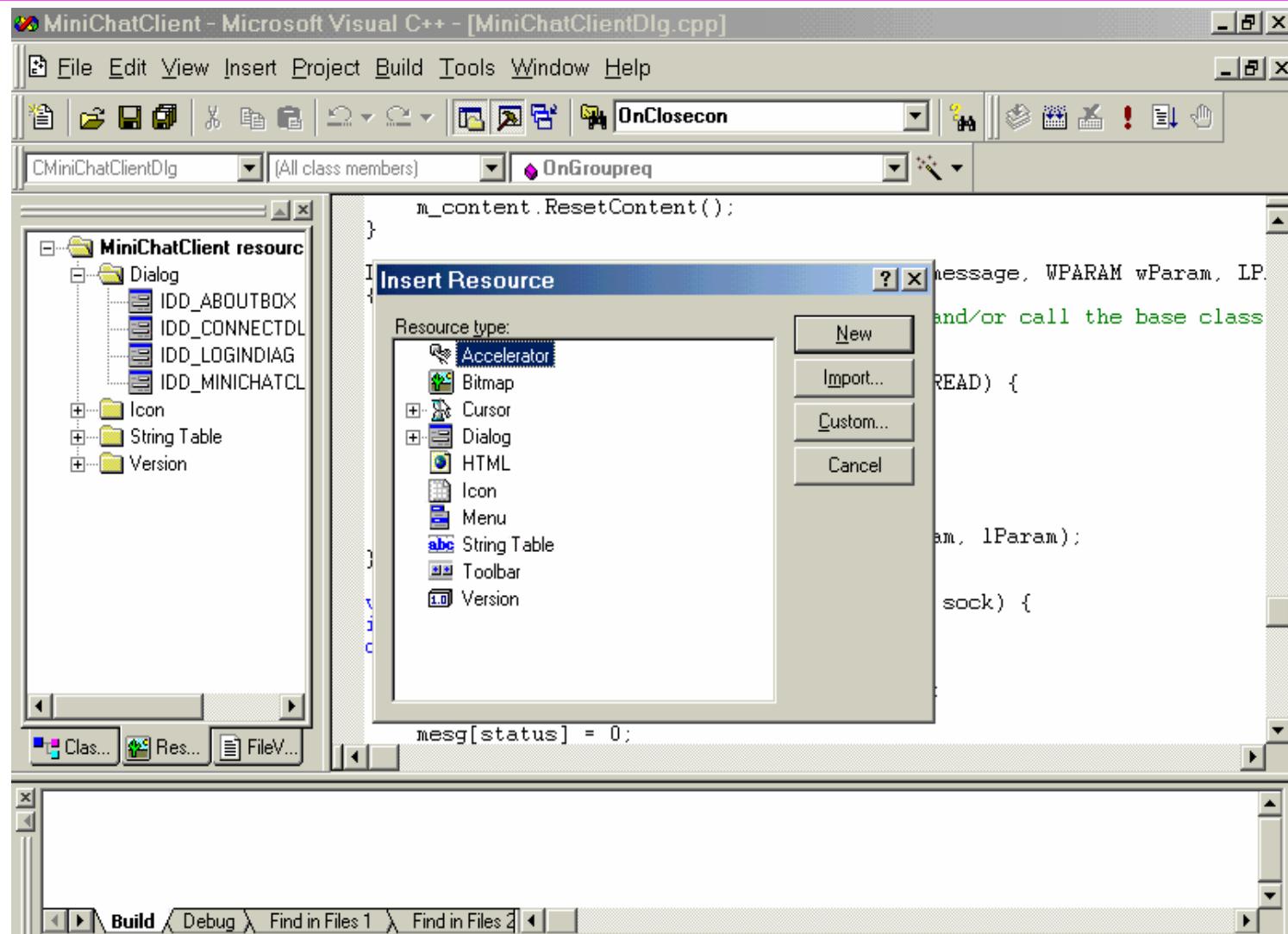


Xây dựng module client MiniChatter

18. Chọn menu

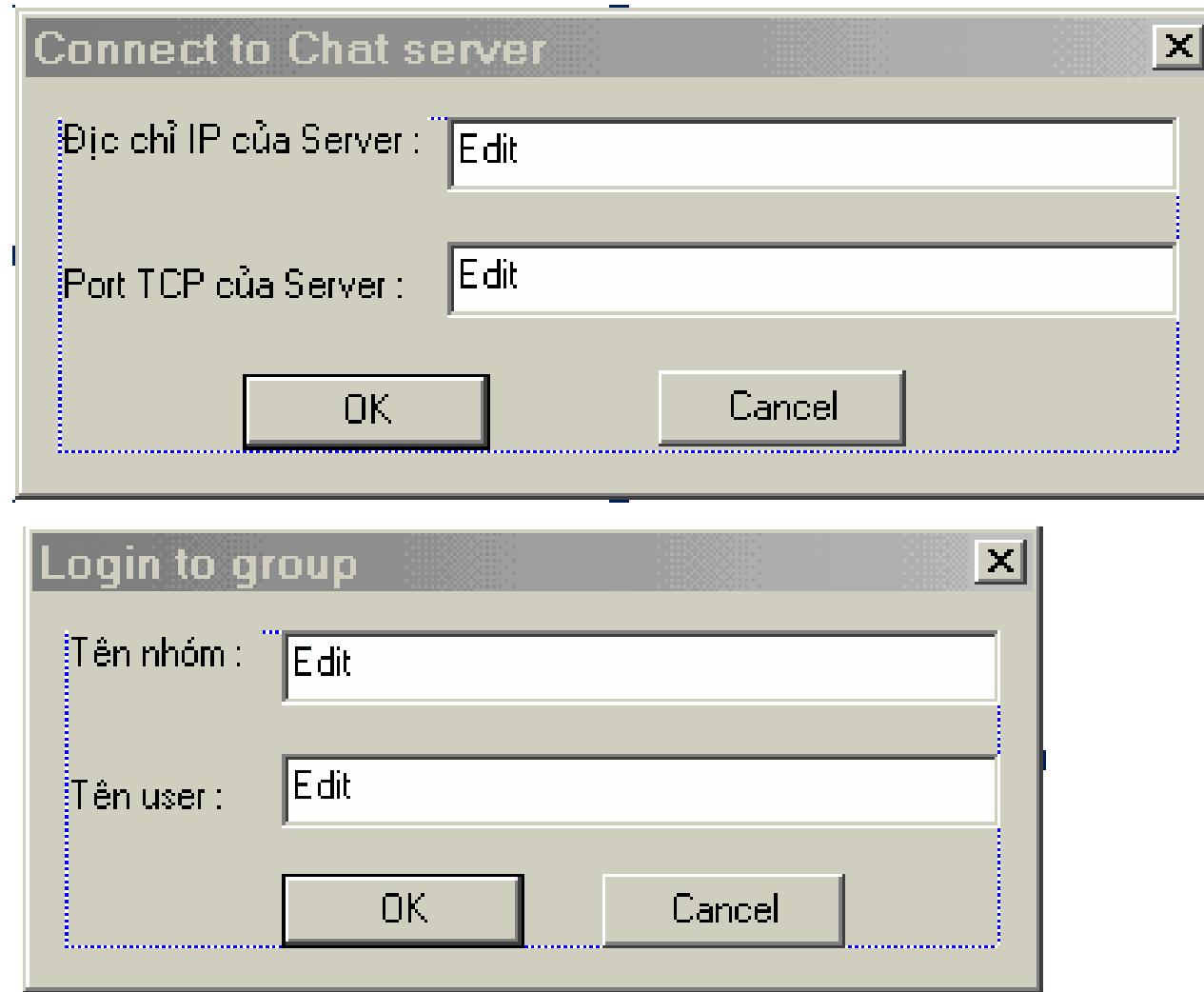
Insert.Resource để
tạo tuân tự các hộp
thoại nhập liệu sau

:



Xây dựng module client MiniChatter

19. Khuôn dạng
mẫu của 2 hộp
thoại nhập liệu :



Xây dựng module client MiniChatter

20. Bắt đầu viết code cho chương trình :

- Phần khởi động ở hàm OnInitDialog nếu cần.
- Các hàm xử lý biến cố : OnConnect, OnGroupreq, OnLogin, OnUserreq, OnSend, OnLogout, Windowproc

21. Code cho hàm OnConnect() :

```
void CMiniChatClientDlg::OnConnect() {
    // TODO: Add your control notification handler code here
    CConnectDlg condlg;
    if (condlg.DoModal() != IDOK) return;
    // Tao socket moi, neu that bai bao sai
    sock=socket(AF_INET,SOCK_STREAM,0);
    if(sock==INVALID_SOCKET) {
        MessageBox("Khong tao duoc socket");
        return;
}
```

Xây dựng module client MiniChatter

```
// thiet lap dia chi diem dau mut ben server
SOCKADDR_IN ser_addr;
ser_addr.sin_family=AF_INET;
ser_addr.sin_port=condlg.m_port;
if ('0'<=condlg.m_ipaddr[0] && condlg.m_ipaddr[0]<='9') // dia chi IP
    ser_addr.sin_addr.s_addr=inet_addr(condlg.m_ipaddr);
else { // dia chi goi nho
    PHOSTENT phe = gethostbyname(condlg.m_ipaddr);
    char szTemp[128];
    if (phe == NULL) {
        wsprintf(szTemp,"Khong co may '%s'", condlg.m_ipaddr);
        MessageBox(szTemp);
        return;
    }
    memcpy((char FAR *)&(ser_addr.sin_addr), phe->h_addr,phe-
>h_length);
}
```



Xây dựng module client MiniChatter

```
// Yeu cau noi ket toi server
if(connect(sock,(LPSOCKADDR)&ser_addr,sizeof(ser_addr))==SOCKET_
ERROR) {
    MessageBox("Khong ket noi duoc");
    return;
}
m_content.AddString("I: Da ket noi thanh cong. Bat dau noi chuyen.");
// Khai bao nhan du lieu bat dong bo + dong cau noi bat dong bo
if (WSAAAsyncSelect(sock, m_hWnd, WSA_RDCLOSE, FD_READ | FD_CLOSE) > 0) {
    MessageBox("Error on WSAAAsyncSelect()");
    closesocket(sock);
}
}
```



Xây dựng module client MiniChatter

```
void CMiniChatClientDlg::OnGroupreq() {  
CString mesg;  
    mesg = _T("GLIST ");  
    if (send(sock,mesg,strlen(mesg),0) <0) {  
        MessageBox("Khong goi duoc lenh GLIST");  
        return;  
    } else fstat = FSGLIST;  
}  
void CMiniChatClientDlg::OnLogin() {  
CString mesg;  
char gname[50];  
int i = m_grouplist.GetCurSel();  
    m_grouplist.GetText(i,gname);  
CLoginDialog condlg(gname);  
    if (condlg.DoModal() != IDOK) return;  
    mesg = _T("LOGIN ") + condlg.m_groupname + _T(",") + condlg.m_username;  
    if (send(sock,mesg,strlen(mesg),0) <0) {  
        MessageBox("Khong goi duoc lenh LOGIN");  
        return;  
    } else fstat = FSLOGIN;  
}
```



Xây dựng module client MiniChatter

```
void CMiniChatClientDlg::OnUserreq() {
    // TODO: Add your control notification handler code here
    CString mesg;
    mesg = _T("ULIST ");
    if (send(sock,mesg,strlen(mesg),0) <0) {
        MessageBox("Khong goi duoc lenh ULIST");
        return;
    } else fstat = FSULIST;
}

void CMiniChatClientDlg::OnSendmsg() {
    // TODO: Add your control notification handler code here
    UpdateData(TRUE);
    if (m_mesg != "") {
        if (send(sock,m_mesg,strlen(m_mesg),0) <0) {
            MessageBox("Khong goi duoc lenh SENDMESG");
            return;
        } else fstat = FSMESG;
        UpdateData(FALSE);
    }
}
```



Xây dựng module client MiniChatter

```
void CMiniChatClientDlg::OnLogout() {  
CString mesg;  
    mesg = _T("LOGOU ");  
    if (send(sock,mesg,strlen(mesg),0) <0) {  
        MessageBox("Khong goi duoc lenh LOGOUT");  
    } else fstat = FSMESG;  
    m_content.ResetContent();  
}  
LRESULT CMiniChatClientDlg::WindowProc(UINT message, WPARAM wParam, LPARAM  
IParam) {  
    switch (message) {  
    case WSA_RDCLOSE :  
        if (WSAGETSELECTEVENT(IParam) == FD_READ) {  
            Reply_Process(wParam);  
        } else {  
            CloseSock(wParam);  
        }  
    return 1;  
    }  
    return CDialog::WindowProc(message, wParam, IParam);  
}
```



Xây dựng module client MiniChatter

```
void CMiniChatClientDlg::Reply_Process(SOCKET sock) {  
    int status;  
    char mesg[MSG_LENGTH];  
    status = recv(sock, mesg, MSG_LENGTH, 0);  
    if (status==0) return;  
    mesg[status] = 0;  
    switch (fstat) {  
        case FSLOGIN :  
            Do_login(mesg); break;  
        case FSLOGOUT: // logout  
            Do_logout(mesg); break;  
        case FSMESG : // group list  
            Do_receive(mesg); break;  
        case FSGLIST: // user list  
            Do_glist(mesg); break;  
        case FSULIST: // broadcast message  
            Do_ulist(mesg);  
    }  
    fstat = FSMESG;  
}
```

}



Xây dựng module client MiniChatter

```
void CMiniChatClientDlg::Reply_Process(SOCKET sock) {  
    int status;  
    char mesg[MSG_LENGTH];  
    status = recv(sock, mesg, MSG_LENGTH, 0);  
    if (status==0) return;  
    mesg[status] = 0;  
    switch (fstat) {  
        case FSLOGIN :  
            Do_login(mesg); break;  
        case FSLOGOUT: // logout  
            Do_logout(mesg); break;  
        case FSMESG : // group list  
            Do_receive(mesg); break;  
        case FSGLIST: // user list  
            Do_glist(mesg); break;  
        case FSULIST: // broadcast message  
            Do_ulist(mesg);  
    }  
    fstat = FSMESG;  
}
```

}



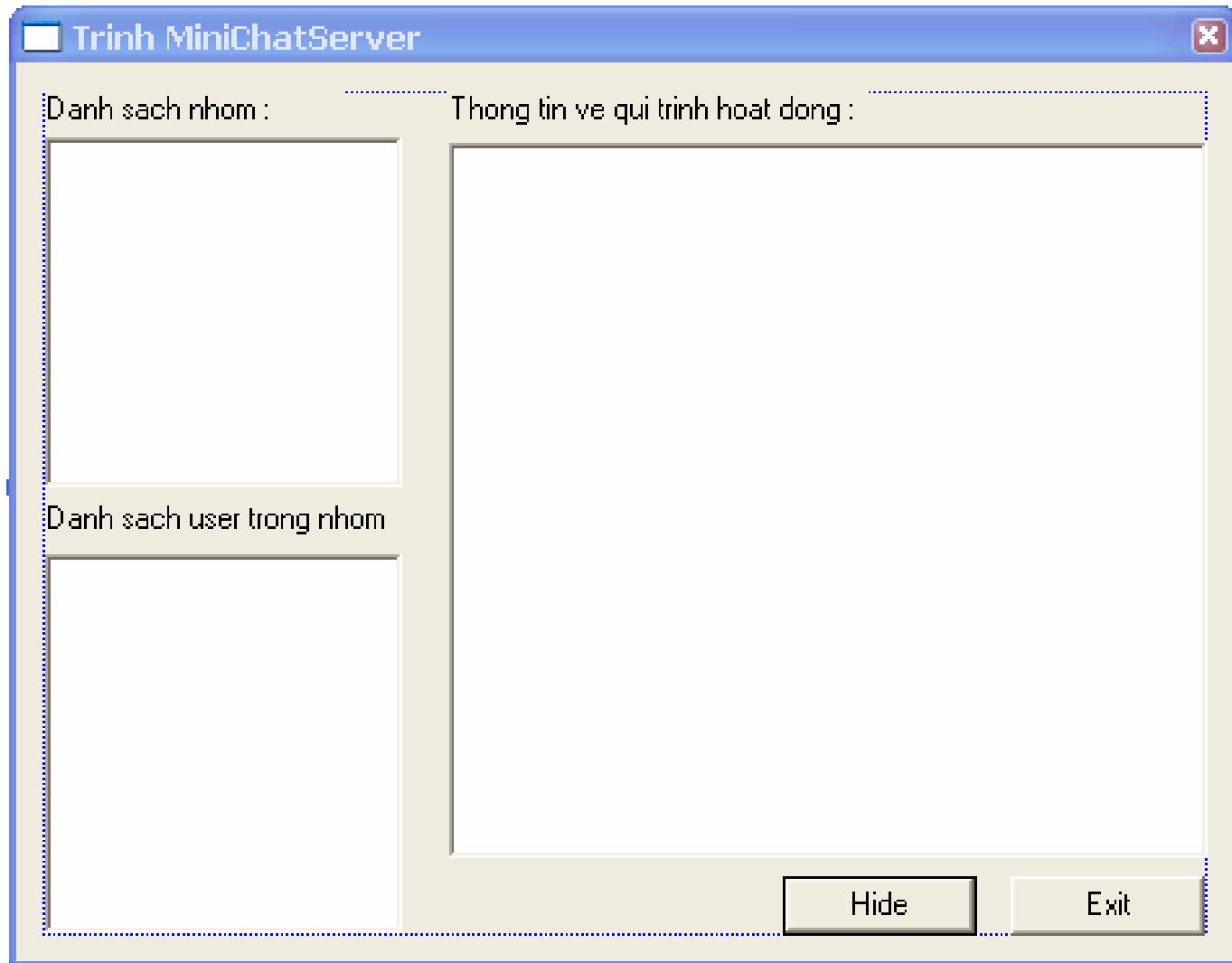
Xây dựng module client MiniChatter

22. Dịch, chạy và kiểm thử từng chức năng của ứng dụng. Khó khăn của việc kiểm thử module client (hay module server) là cần phải có đối tác chạy đồng thời với mình. Do đó, chúng ta sẽ chạy một instance VC++ 6.0 khác để tạo và quản lý module MiniChatter Server. Các slide còn lại sẽ miêu tả thông tin về qui trình xây dựng module server.



2.2 Xây dựng module Server MiniChatter

1. Chạy một instance VC++ 6.0 khác, tạo và quản lý module MiniChatter Server. Thiết kế form giao diện của server như sau :



Xây dựng module Server MiniChatter

```
//Code cua MiniChatServer
void CMiniChatServerDlg::OnHide() {
    // TODO: Add your control notification handler code here
    ShowWindow(SW_HIDE);
}

void CMiniChatServerDlg::OnCancel() {
    // TODO: Add extra cleanup here
    Shell_NotifyIcon(NIM_DELETE, &tnd);
    CDialog::OnCancel();
}
```



Xây dựng module Server MiniChatter

```
BOOL CMiniChatServerDlg::OnCommand(WPARAM wParam, LPARAM IParam) {  
    // TODO: Add your specialized code here and/or call the base class  
    switch (wParam) {  
        case SC_CLOSE :  
            Shell_NotifyIcon(NIM_DELETE, &tnd);  
            return CDialog::OnCommand(WM_CLOSE,IParam);  
            break;  
        case IDM_ABOUTBOX :  
            CAaboutDlg dlgAbout;  
            dlgAbout.DoModal();  
            break;  
    }  
    return CDialog::OnCommand(wParam, IParam);  
}
```



Xây dựng module Server MiniChatter

```
BOOL CMiniChatServerDlg::OnInitDialog() {  
    //...  
    // TODO: Add extra initialization here  
    // Init group list  
    sock_no_user = 0;  
    //Đọc danh sách nhóm cần quản lý từ database  
    ReadDisplayGroups();  
    // Tao socket moi, neu that bai bao sai  
    ser_sock=socket(AF_INET,SOCK_STREAM,0);  
    if(ser_sock==INVALID_SOCKET) {  
        MessageBox("Khong tao duoc socket"); return TRUE;  
    }  
    // thiet lap dia chi diem dau mut va bind no voi socket  
    SOCKADDR_IN local_addr;  
    local_addr.sin_family=AF_INET;  
    local_addr.sin_port=256;  
    local_addr.sin_addr.s_addr=INADDR_ANY;  
    if(bind(ser_sock,(LPSOCKADDR)&local_addr,sizeof(local_addr))==SOCKET_ERROR) {  
        MessageBox("Khong bind socket duoc");  
        return TRUE;  
    }  
}
```



Xây dựng module Server MiniChatter

```
// Khai bao so yeu cau ket noi dong thoi
if(listen(ser_sock,SOMAXCONN)==SOCKET_ERROR) {
    MessageBox("Khong listen duoc"); return TRUE;
}
// Khai bao nhan du lieu bat dong bo + dong cau noi bat dong bo
if (WSAAAsyncSelect(ser_sock, m_hWnd, WSA_ACCEPT, FD_ACCEPT) > 0) {
    MessageBox("Error on WSAAAsyncSelect()");
    closesocket(ser_sock);
}
// Display cac nhom
for (i=0; i< groupcnt; i++) m_groupbox.AddString(m_grouplist[i].name);
// Add to System tray
tnd.cbSize = sizeof(NOTIFYICONDATA);
tnd.hWnd = m_hWnd; tnd.uID = 3200; tnd.hIcon = m_hIcon;
tnd.uFlags = NIF_MESSAGE|NIF_ICON|NIF_TIP;
tnd.uCallbackMessage = MYWM_NOTIFYICON;
strcpy(tnd.szTip,"MiniChatServer");
Shell_NotifyIcon(NIM_ADD, &tnd);
// ShowWindow(SW_HIDE);
return TRUE; // return TRUE unless you set the focus to a control
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::ReadDisplayGroups(){
_ConnectionPtr Con; _RecordsetPtr Rs;
::CoInitialize(NULL);
if (Rs.CreateInstance (_uuidof(Recordset))!= S_OK) {
    MessageBox("Khong khai dong duoc Recordset"); return;
}
Rs->Open ("Select * from GroupList", _bstr_t("DSN=GroupList;UID=;PWD="),
adOpenForwardOnly, adLockReadOnly, adCmdUnknown);
VARIANT vGName; CString m_GName;
groupcnt = 0;
while (VARIANT_FALSE == Rs->GetadoEOF()) { groupcnt++; Rs->MoveNext(); }
m_grouplist = (T_GroupList*)malloc(groupcnt*sizeof(T_GroupList));
Rs->MoveFirst();
int i = 0;
while (VARIANT_FALSE == Rs->GetadoEOF()) {
    vGName = Rs->Fields->GetItem(_variant_t((long)1))->Value;
    m_GName = vGName.bstrVal; Rs->MoveNext();
    strcpy(m_grouplist[i].name,m_GName); m_grouplist[i++].userlist = 0;
}
Rs->Close();
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::OnSysCommand(UINT nID, LPARAM lParam) {  
    if ((nID & 0xFFFF0) == IDM_ABOUTBOX) {  
        CAboutDlg dlgAbout;  
        dlgAbout.DoModal();  
    } else CDialog::OnSysCommand(nID, lParam);  
}  
//  
void CMiniChatServerDlg::HandlePopupMenu (void) {  
    HMENU hMenu; POINT point;  
    int ketqua;  
    GetCursorPos (&point);  
    // Get the menu for the windows  
    hMenu = ::GetSystemMenu (m_hWnd, FALSE);  
    if (!hMenu) return;  
    //SetForegroundWindow ();  
    ketqua = TrackPopupMenu (hMenu, TPM_RIGHTBUTTON, point.x, point.y, 0, m_hWnd,  
    NULL);  
    // This is required when using a notify icon -- see KB article  
    // PRB: Menus for Notification Icons Don't Work Correctly  
    PostMessage (WM_USER, 0, 0);  
}
```



Xây dựng module Server MiniChatter

```
LRESULT CMiniChatServerDlg::WindowProc(UINT message, WPARAM wParam, LPARAM IParam) {
    switch (message) {
        case WSA_ACCEPT : OnAccept(); return 1;
        case WSA_RDCLOSE :
            if (WSAGETSELECTEVENT(IParam) == FD_READ) Read_Process(wParam);
            else CloseSock(wParam);
            return 1;
        case MYWM_NOTIFYICON : switch (IParam) {
            case WM_RBUTTONDOWN:
                // Handle popup here
                HandlePopupMenu (); return TRUE;
            case WM_LBUTTONDOWN:{ 
                // Diplay the dialog box on Left mouse click
                ShowWindow (SW_RESTORE);
                SetForegroundWindow ();
                return TRUE;
            } // IParam switch
        }
    return CDialog::WindowProc(message, wParam, IParam);
}
```



Xây dựng module Server MiniChatter

```
// Accept 1 yêu cầu nối kết
void CMiniChatServerDlg::OnAccept(void) {
    SOCKADDR_IN remote_addr; SOCKET sock;
    // Cho kết nối
    int len = sizeof(remote_addr);
    sock = accept(ser_sock, (LPSOCKADDR)&remote_addr, &len);
    if (sock == INVALID_SOCKET) {
        MessageBox("Không accept được"); return;
    }
    T_UserRec *puser = new(T_UserRec);
    puser->sock = sock; puser->next = sock_no_user; sock_no_user = puser;
    // Khai báo nhân duyên bắt đồng bộ + đồng cao nối bắt đồng bộ
    if (WSAAAsyncSelect(sock, m_hWnd, WSA_RDCLOSE, FD_READ | FD_CLOSE) > 0) {
        MessageBox("Error on WSAAAsyncSelect()");
        closesocket(sock);
    }
    // Gọi reply ban đầu về client
    if (send(sock, startmesg, strlen(startmesg), 0) < 0)
        MessageBox("Không gọi được dữ liệu");
}
```



Xây dựng module Server MiniChatter

```
// Doc vao request va xu ly
void CMiniChatServerDlg::Read_Process(SOCKET sock) {
int status;
char mesg[MSG_LENGTH];

status = recv(sock, mesg, MSG_LENGTH, 0);
if (status==0) return;
mesg[status] = 0;
if (strncmp(mesg,"LOGIN",5)==0) { // login
    Do_login(sock,mesg);
} else if (strncmp(mesg,"LOGOU",5)==0) { // logout
    Do_logout(sock);
} else if (strncmp(mesg,"GLIST",5)==0) { // group list
    Do_glist(sock);
} else if (strncmp(mesg,"ULIST",5)==0) { // user list
    Do_ulist(sock);
} else { // broadcast message
    Do_broadcastMesg(sock,mesg);
}
}
```

}



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::Do_glist(SOCKET sock) {  
    char mesg[1024]; char *ps; int i;  
    wsprintf(mesg,"1 %s",m_grouplist[0].name);  
    ps = &mesg[strlen(mesg)];  
    for (i=1;i < groupcnt;i++) {  
        wsprintf(ps,"%s",m_grouplist[i].name); ps += strlen(ps);  
    }  
    if (send(sock,mesg,strlen(mesg),0) <0) MessageBox("Khong goi duoc du lieu");  
}
```

```
int CMiniChatServerDlg::Findgroup(SOCKET sock, char** uname) {  
    int i; T_UserRec *pu;  
    for (i=0; i<groupcnt; i++) {  
        pu = m_grouplist[i].userlist;  
        while (pu) {  
            if (pu->sock == sock) { *uname = pu->name; return i; }  
            pu = pu->next;  
        }  
    }  
    return -1;  
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::Do_ulist(SOCKET sock) {  
    char mesg[1024];  
    char *ps;  
    int i = Findgroup(sock,&ps);  
    if (i <0 || m_grouplist[i].userlist ==0) {  
        strcpy (mesg,"0 ");  
    } else {  
        T_UserRec *pu = m_grouplist[i].userlist;  
        wsprintf(mesg,"1 %s",pu->name);  
        ps = &mesg[strlen(mesg)];  
        pu = pu->next;  
        while (pu) {  
            wsprintf(ps,"%s",pu->name);  
            ps += strlen(ps);  
            pu = pu->next;  
        }  
    }  
    if (send(sock,mesg,strlen(mesg),0) <0) {  
        MessageBox("Khong goi duoc du lieu ulist");  
    }  
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::Do_broadcastMesg(SOCKET sock, char *mesg) {  
    char *ps;  
    int i = Findgroup(sock,&ps);  
    if (i <0) return;  
    char bmesg[256];  
    wsprintf(bmesg,"%s: %s",ps,mesg);  
    T_UserRec *pu = m_grouplist[i].userlist;  
    while (pu) {  
        send(pu->sock,bmesg,strlen(bmesg),0);  
        pu = pu->next;  
    }  
}
```



Xây dựng module Server MiniChatter

```
// Dong cau noi tuong ung
void CMiniChatServerDlg::CloseSock(SOCKET sock) {
char *ps;
int i = Findgroup(sock,&ps);
T_UserRec *pu, *pup;
if (i >= 0) {
    pu = m_grouplist[i].userlist;
    while (pu && pu->sock != sock) pup = pu; pu = pu->next;
    if (pu==m_grouplist[i].userlist) m_grouplist[i].userlist = pu->next;
    else pup->next = pu->next;
    delete(pu);
} else {
    pu = sock_no_user;
    while (pu && pu->sock != sock) pup = pu; pu = pu->next;
    if (pu==sock_no_user)
        sock_no_user = pu->next;
    else pup->next = pu->next;
    delete(pu);
}
closesocket(sock);
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::Do_login(SOCKET sock, char *mesg) {  
T_UserRec *pup,*pu;  
char gname[80], uname[80];  
int i;  
    if (mesg[6] == '0') goto errordisp;  
    // tim ten group  
    i = 0; mesg += 6;  
    do gname[i++] = *mesg++;  
    while (*mesg && *mesg!=',');  
    gname[i] = 0;  
    if (*mesg++==0) goto errordisp;  
    // tim ten user  
    i = 0;  
    do uname[i++] = *mesg++;  
    while (*mesg && *mesg!=',');  
    uname[i] = 0;  
    // tim group tuong ung  
    for (i=0; i<groupcnt; i++)  
        if (strcmp (gname,m_grouplist[i].name)==0) break;  
    if (i > groupcnt) goto errordisp;
```



Xây dựng module Server MiniChatter

```
// tim sokcet trong danh sach chua co user
pup = pu = sock_no_user;
while (pu && pu->sock != sock) pup = pu; pu = pu->next;
if (pu) {
    if (pu == sock_no_user)
        sock_no_user = pu->next;
    else pup->next = pu->next;
    pu->next = m_grouplist[i].userlist;
    strcpy(pu->name,uname);
    m_grouplist[i].userlist = pu;
    strcpy(uname,"1 ");
    if (send(sock,uname,strlen(uname),0) <0)
        MessageBox("Khong goi duoc reply cho ULOGIN");
    return;
}
errordisp:
strcpy(uname,"0 ");
if (send(sock,uname,strlen(uname),0) <0)
    MessageBox("Khong goi duoc reply cho ULOGIN");
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::Do_logout(SOCKET sock) {  
    char *ps;  
    int i = Findgroup(sock,&ps);  
    T_UserRec *pu, *pup;  
    if (i >= 0) {  
        pu = m_grouplist[i].userlist;  
        while (pu && pu->sock != sock) {  
            pup = pu; pu = pu->next;  
        }  
        if (pu==m_grouplist[i].userlist)  
            m_grouplist[i].userlist = pu->next;  
        else pup->next = pu->next;  
        pu->next = sock_no_user;  
        sock_no_user = pu;  
    }  
}
```



Xây dựng module Server MiniChatter

```
void CMiniChatServerDlg::OnSelchangeGroup()
{
    // TODO: Add your control notification handler code here
char gname[50];
int i;
    i=m_groupbox.GetCurSel();
    m_groupbox.GetText(i,gname);
    // tim group tuong ung
    for (i=0; i<groupcnt; i++)
        if (strcmp (gname,m_grouplist[i].name)==0) break;
    if (i > groupcnt) return;
T_UserRec *pu = m_grouplist[i].userlist;
    m_userbox.ResetContent ();
    while (pu) {
        m_userbox.AddString (pu->name);
        pu = pu->next ;
    }
}
```



Xây dựng module client MiniChatter

2. Dịch, chạy và kiểm thử từng chức năng của ứng dụng. Khó khăn của việc kiểm thử module client (hay module server) là cần phải có đối tác chạy đồng thời với mình. Do đó, chúng ta sẽ debug module MiniChatter Server đồng thời với module MiniChatter Client đã viết được ở mục 2.1 trước đây.



MÔN LẬP TRÌNH MẠNG

Chương 3

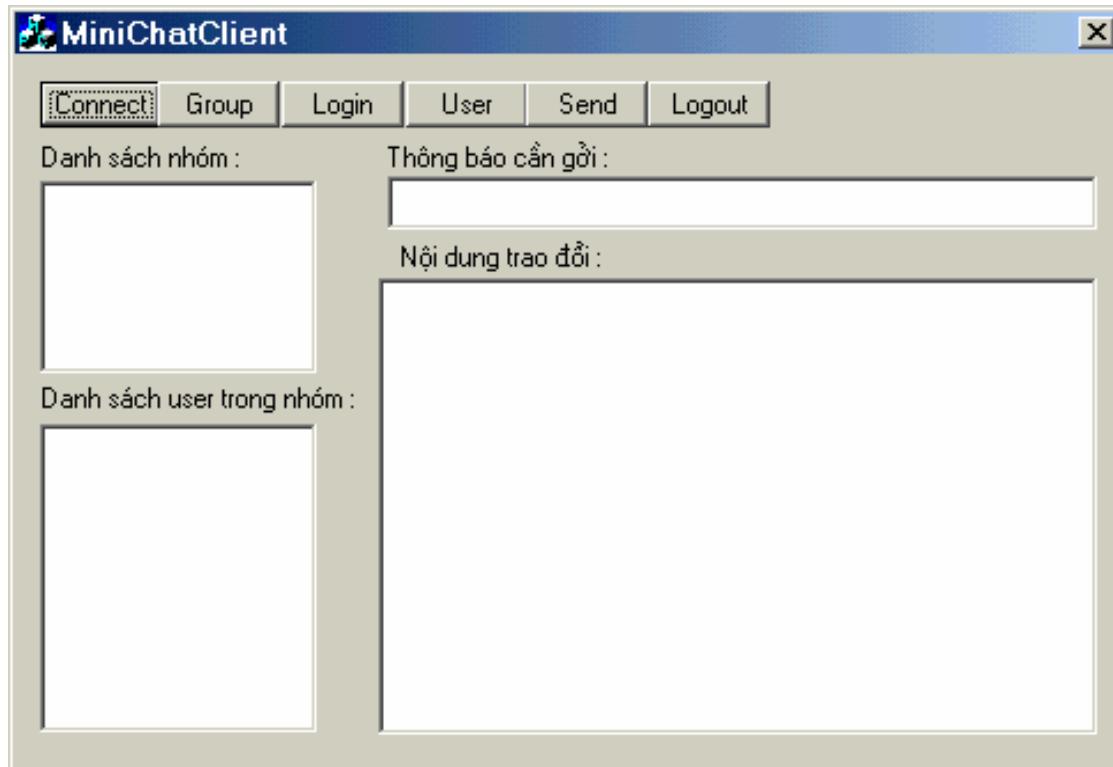
Viết hệ thống MiniChatter bằng Java & dùng kỹ thuật multi-thread

- 3.1 Xây dựng module MiniChatter Client
- 3.2 Xây dựng module MiniChatter Server

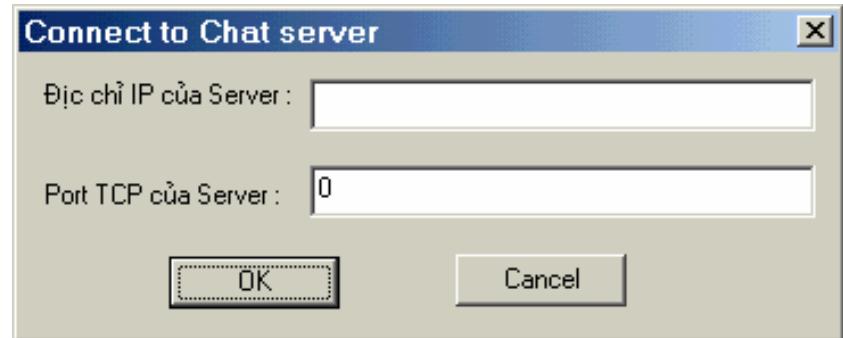


3.1 Xây dựng module client MiniChatter

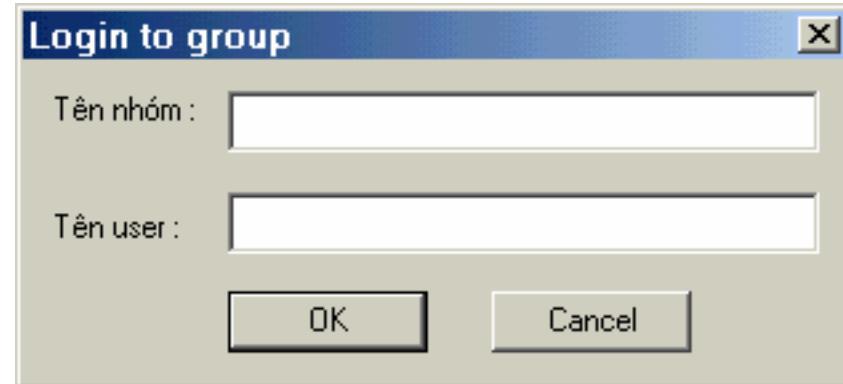
Chúng ta sẽ thiết kế giao diện cho Form ứng dụng MiniChatClient như sau :



Cửa sổ phục vụ lệnh Connect :

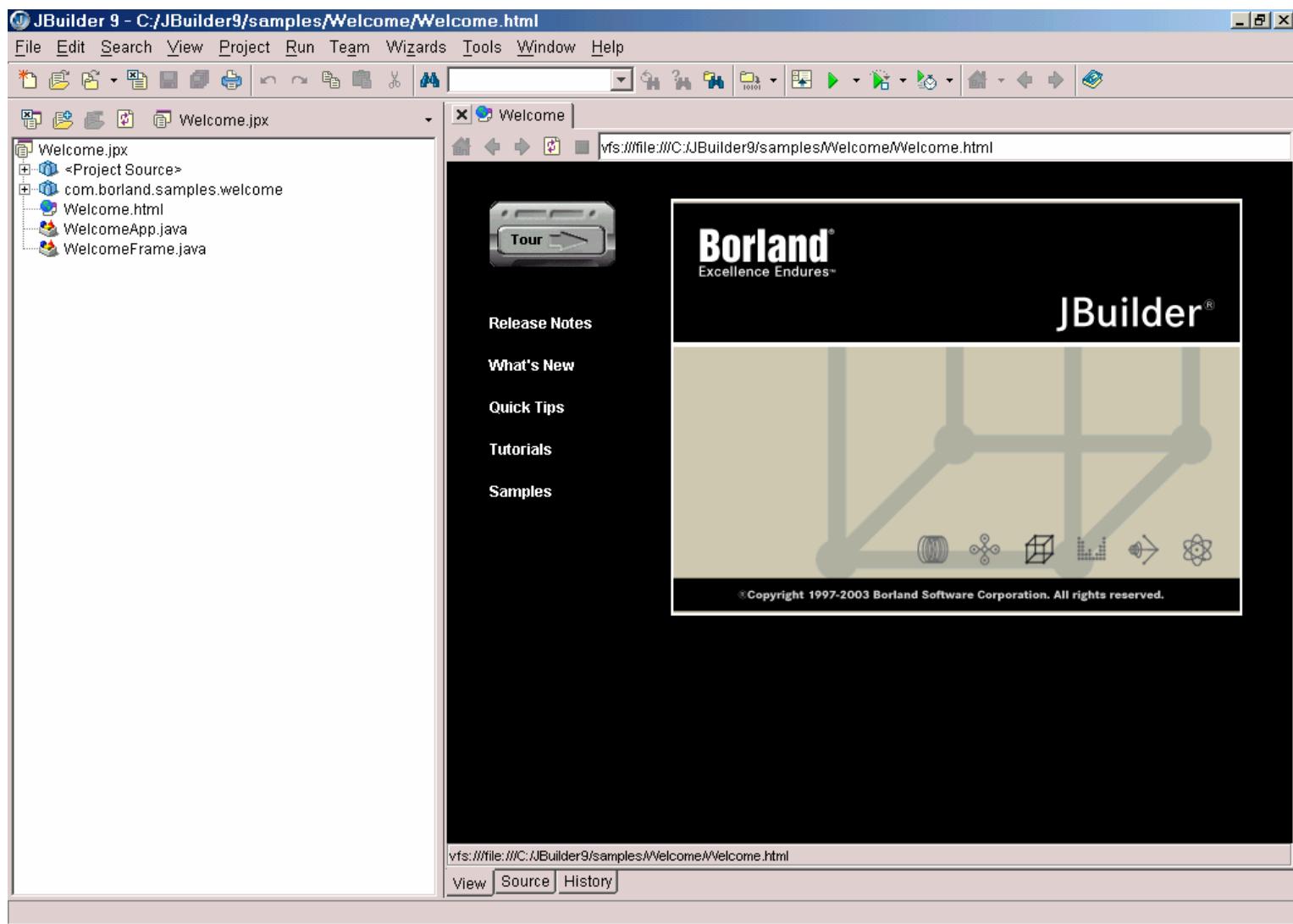


Cửa sổ phục vụ lệnh Login :



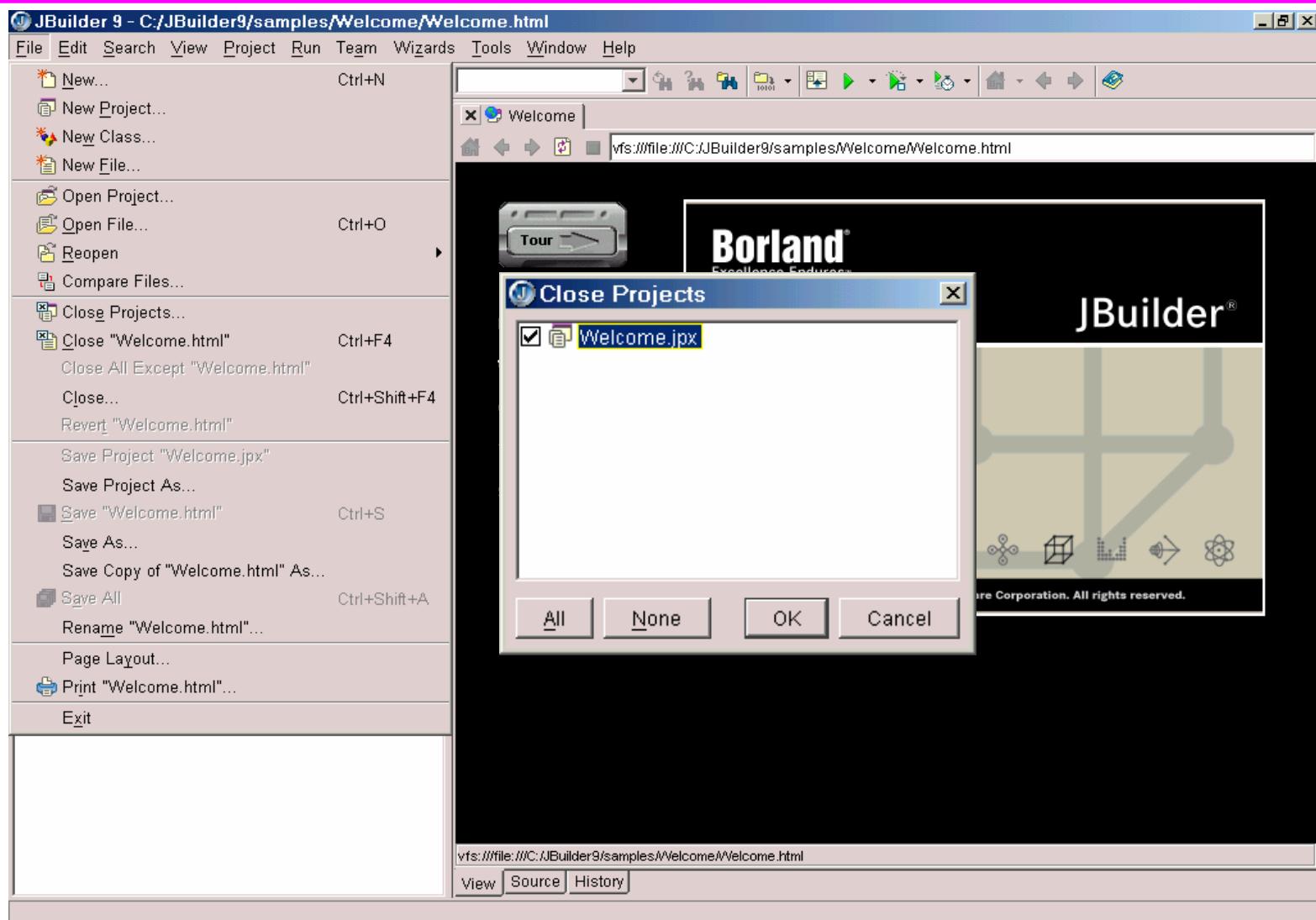
Xây dựng module client MiniChatter

1. Chạy JBuilder từ Windows. Màn hình của JBuilder được hiển thị như sau :



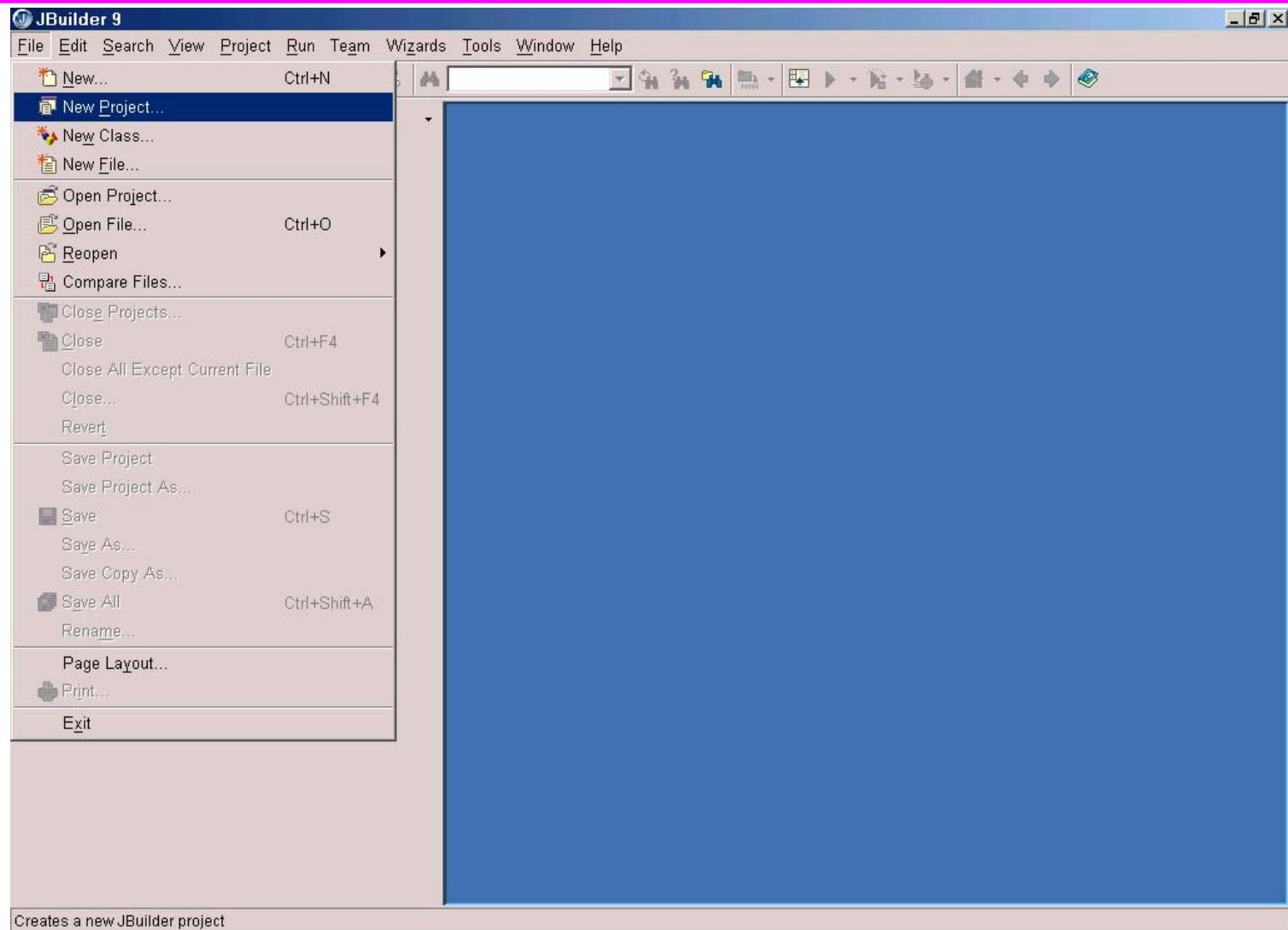
Xây dựng module client MiniChatter

2. Chọn menu File.Close Projects... để hiển thị cửa sổ "Close Projects", chọn button "All" rồi Ok để đóng các Project đang mở :



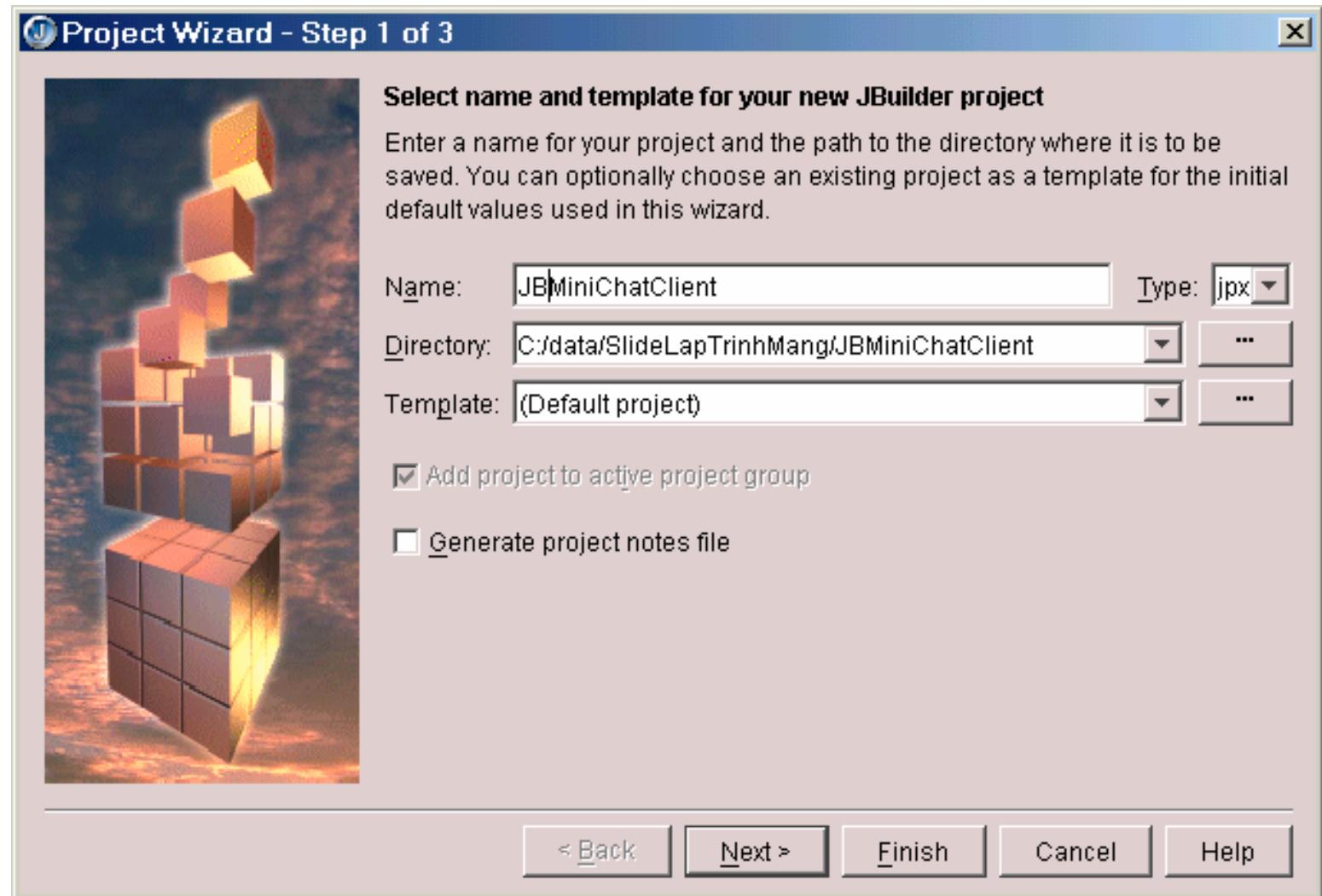
Xây dựng module client MiniChatter

3. Chọn menu
File.New
Project... để tạo
Project mới
quản lý ứng
dụng
MiniChatClient :



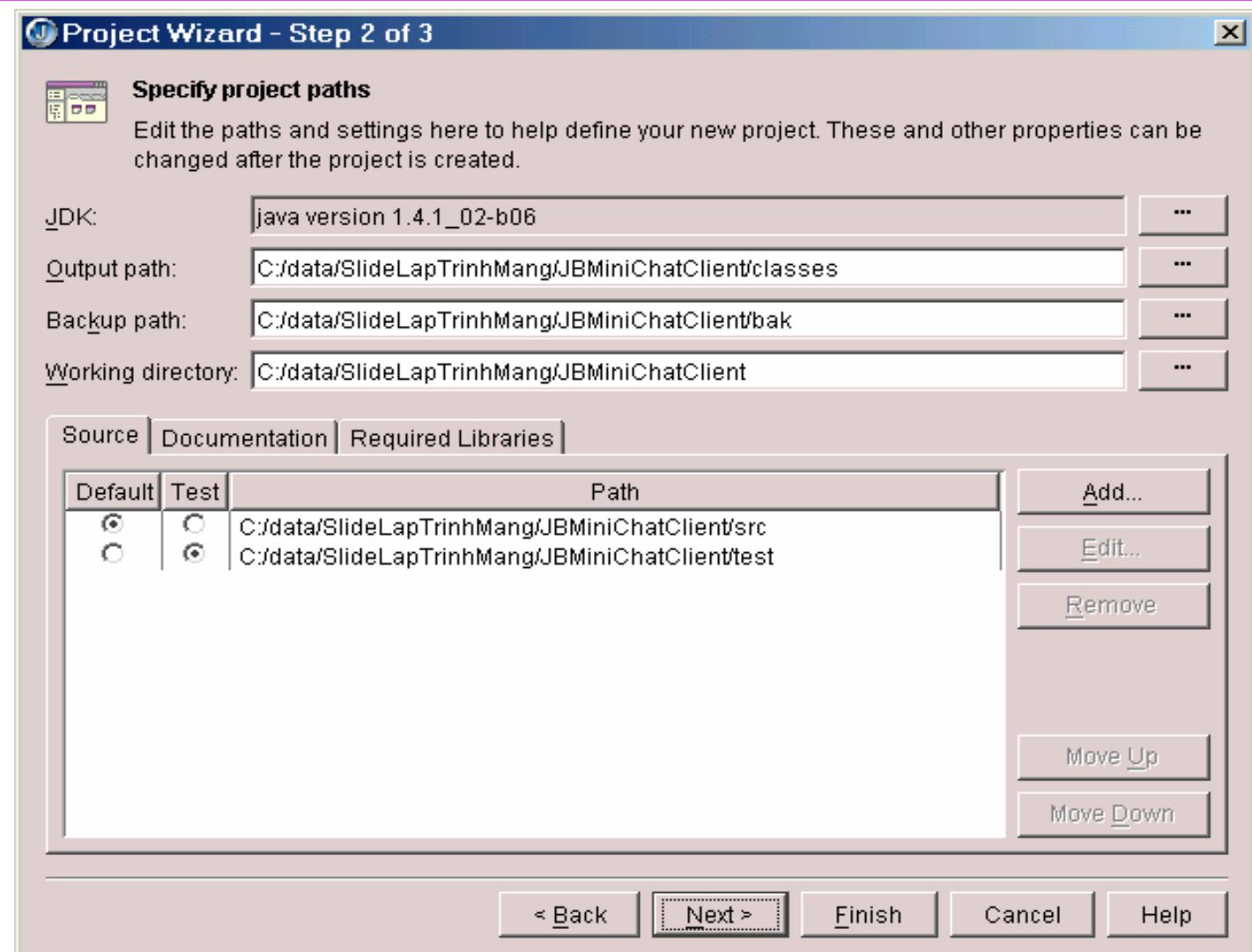
Xây dựng module client MiniChatter

4. Nhập tên Project, chọn thư mục chứa Project, chọn Template qui định cấu hình ban đầu cho Project (nên để mặc định), rồi ấn button Next :



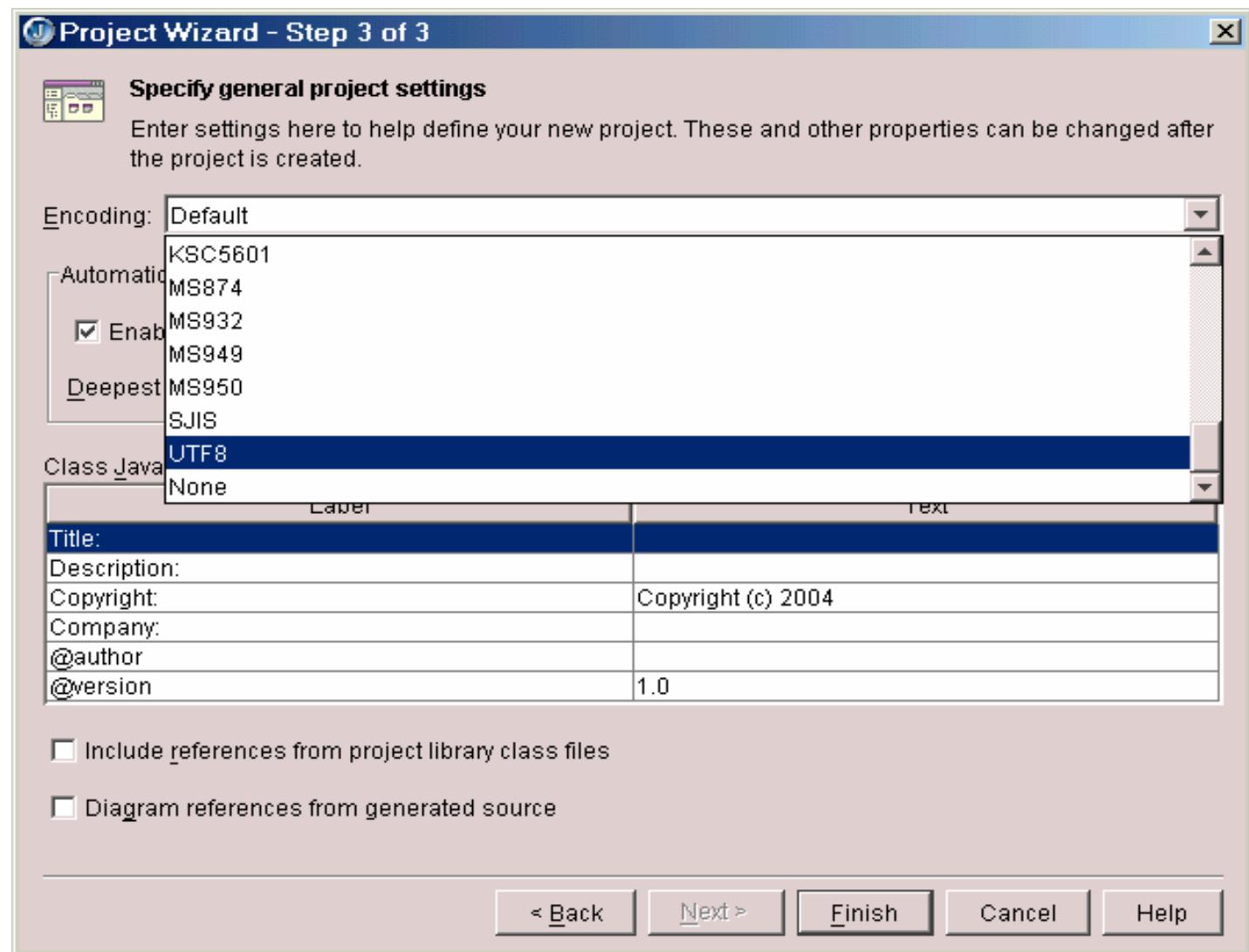
Xây dựng module client MiniChatter

5. Chọn version JDK, chọn thư mục chứa các file Output, Backup, chọn thư mục Working (nên để mặc định), rồi ấn button Next :



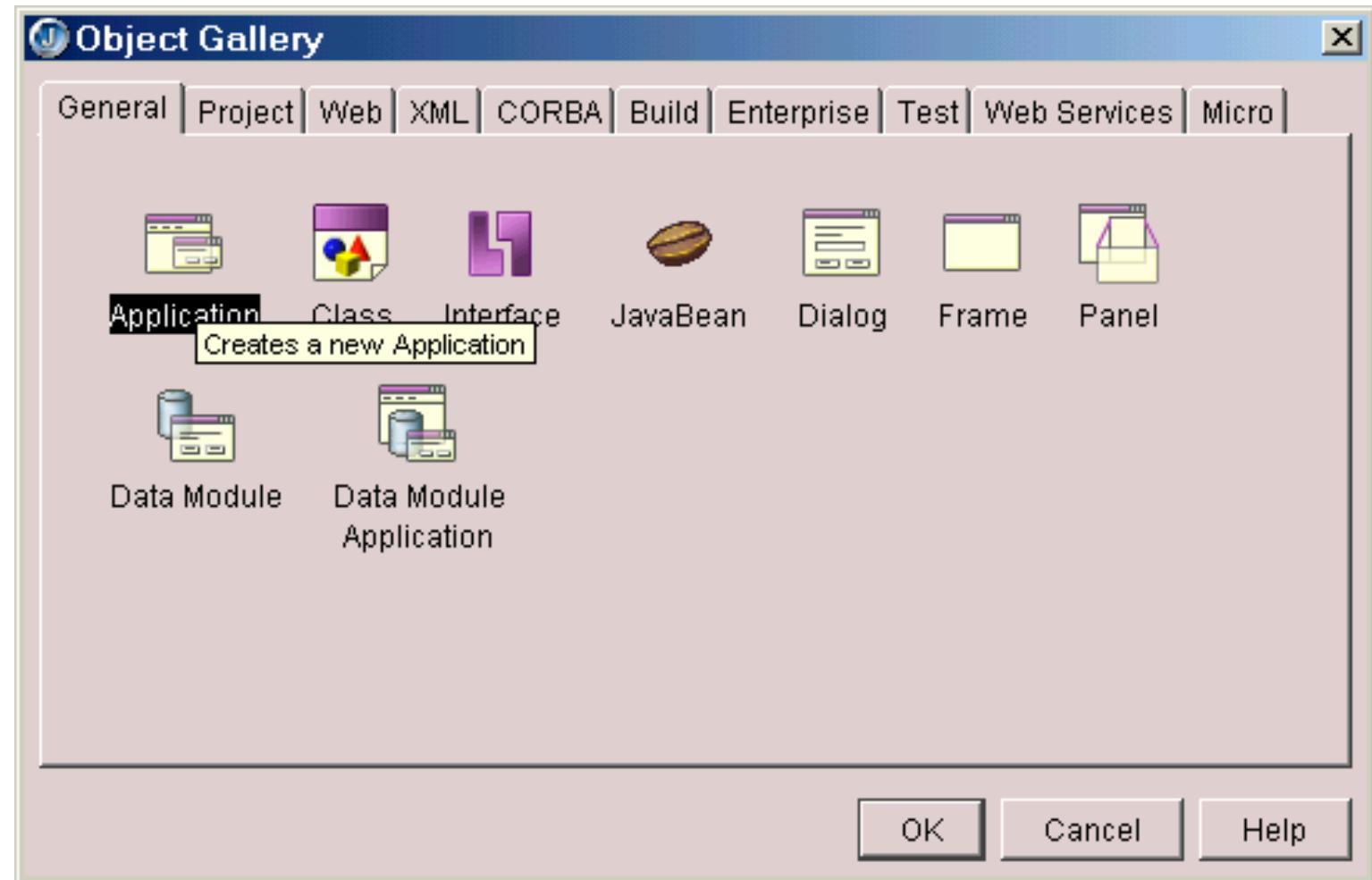
Xây dựng module client MiniChatter

6. Chọn cách mã hóa chuỗi là UTF-8, nhập Title và các thông tin khác về ứng dụng, nếu muốn, rồi ấn nutton Finish :



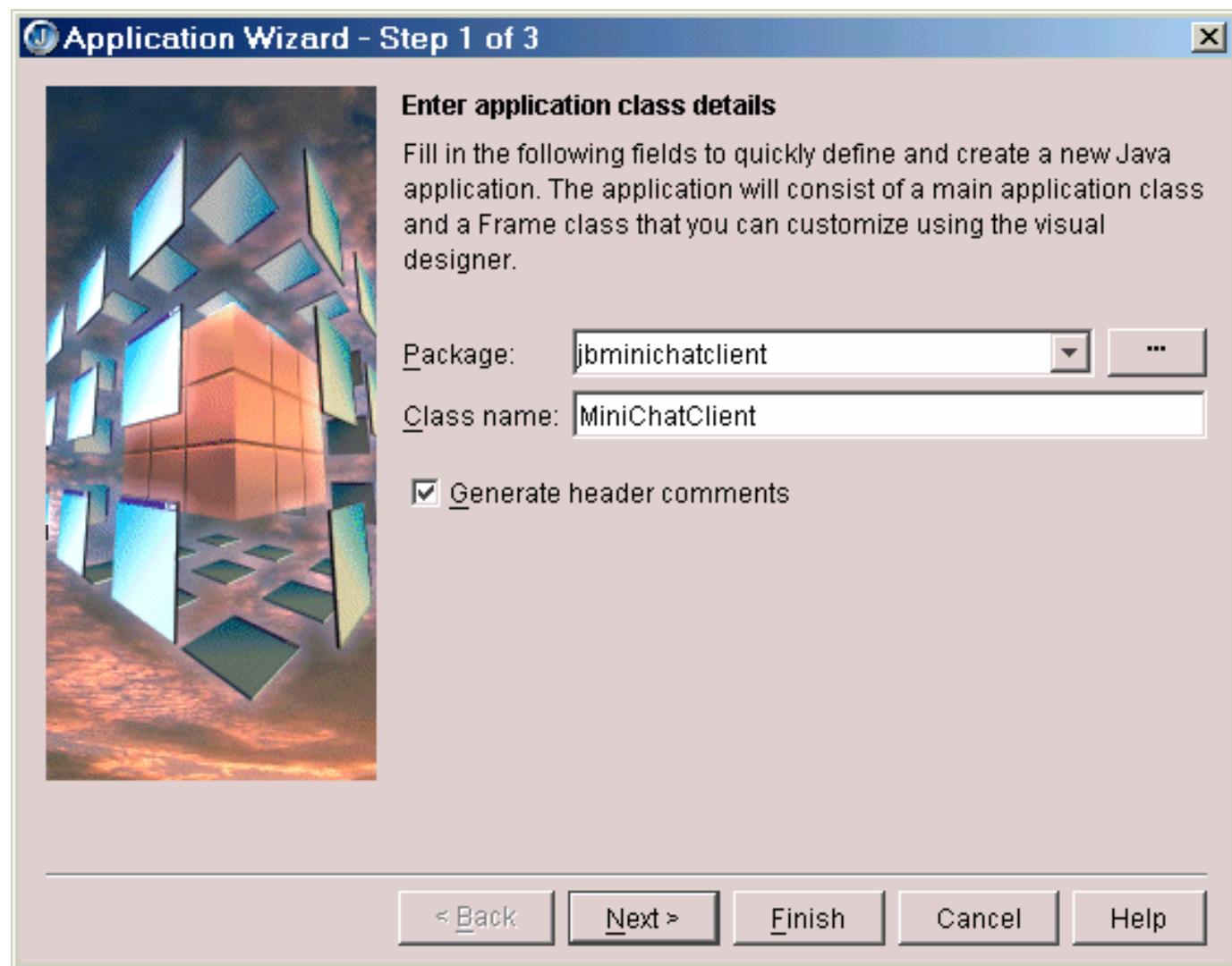
Xây dựng module client MiniChatter

7. Chọn menu
File.New..., cửa
sổ Object
Gallery sẽ hiển
thị, chọn icon
"Application" để
tạo 1 ứng dụng
mới trong
Project :



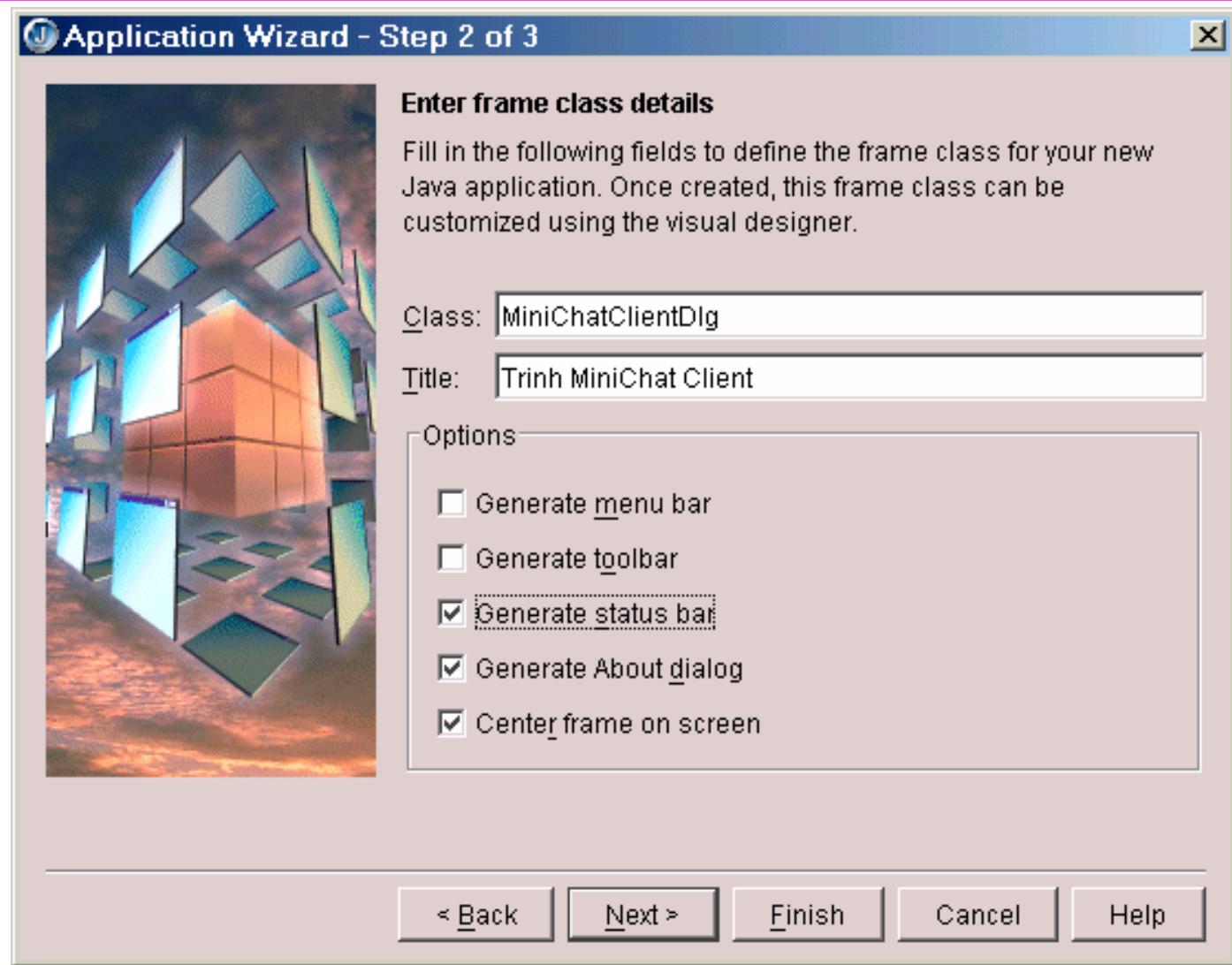
Xây dựng module client MiniChatter

8. Nhập tên ứng dụng, chọn Package chứa ứng dụng (nên để mặc định) rồi ấn button Next :



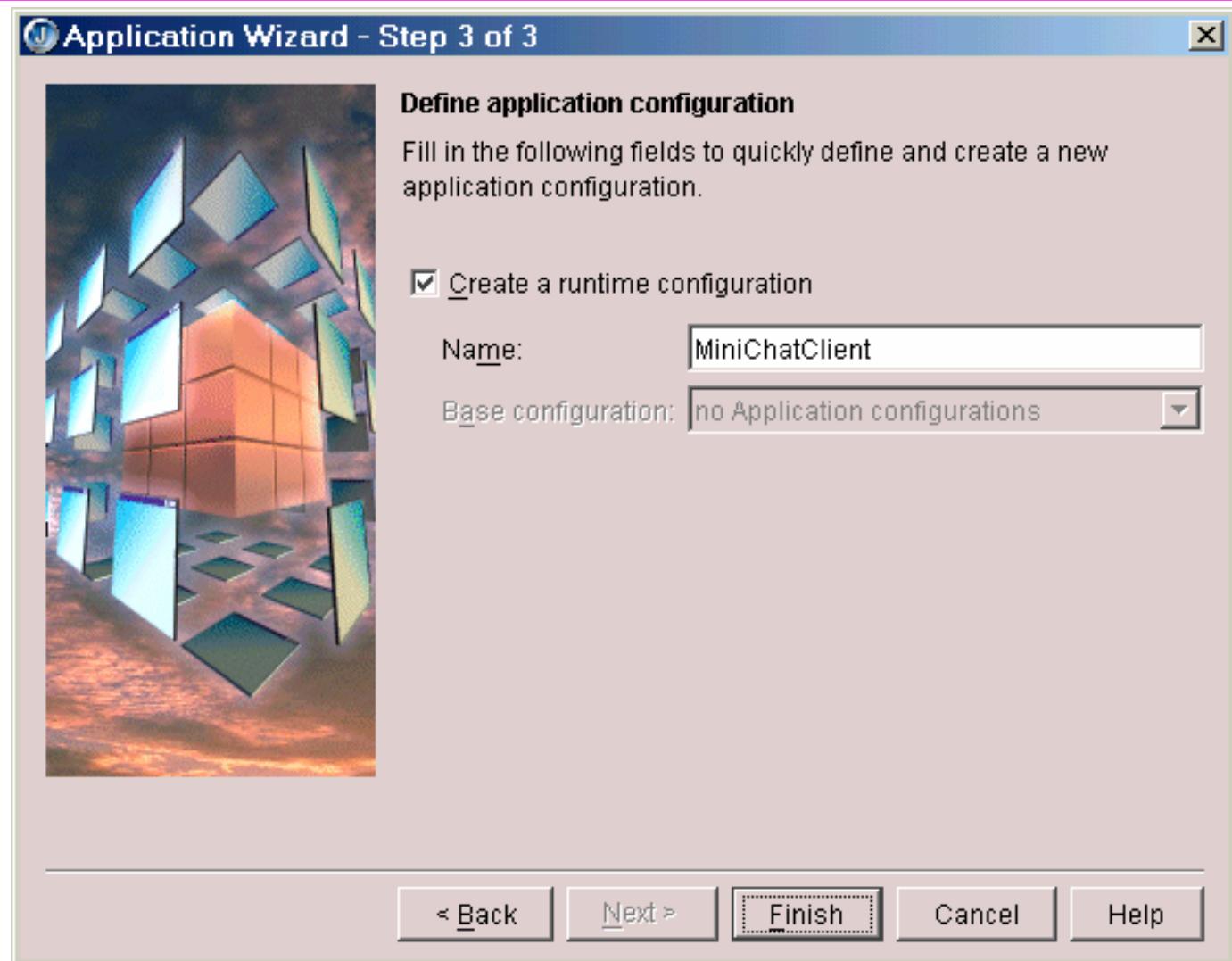
Xây dựng module client MiniChatter

9. Nhập tên class và title bar cho dialog của ứng dụng, chọn các options cần thiết rồi ấn button Next :



Xây dựng module client MiniChatter

10. Nhập tên file quản lý cấu hình run-time cho ứng dụng vừa tạo (nên để mặc định) rồi ấn button Finish :



Xây dựng module client MiniChatter

11. Cửa sổ JBuilder lúc này sẽ có dạng như sau :

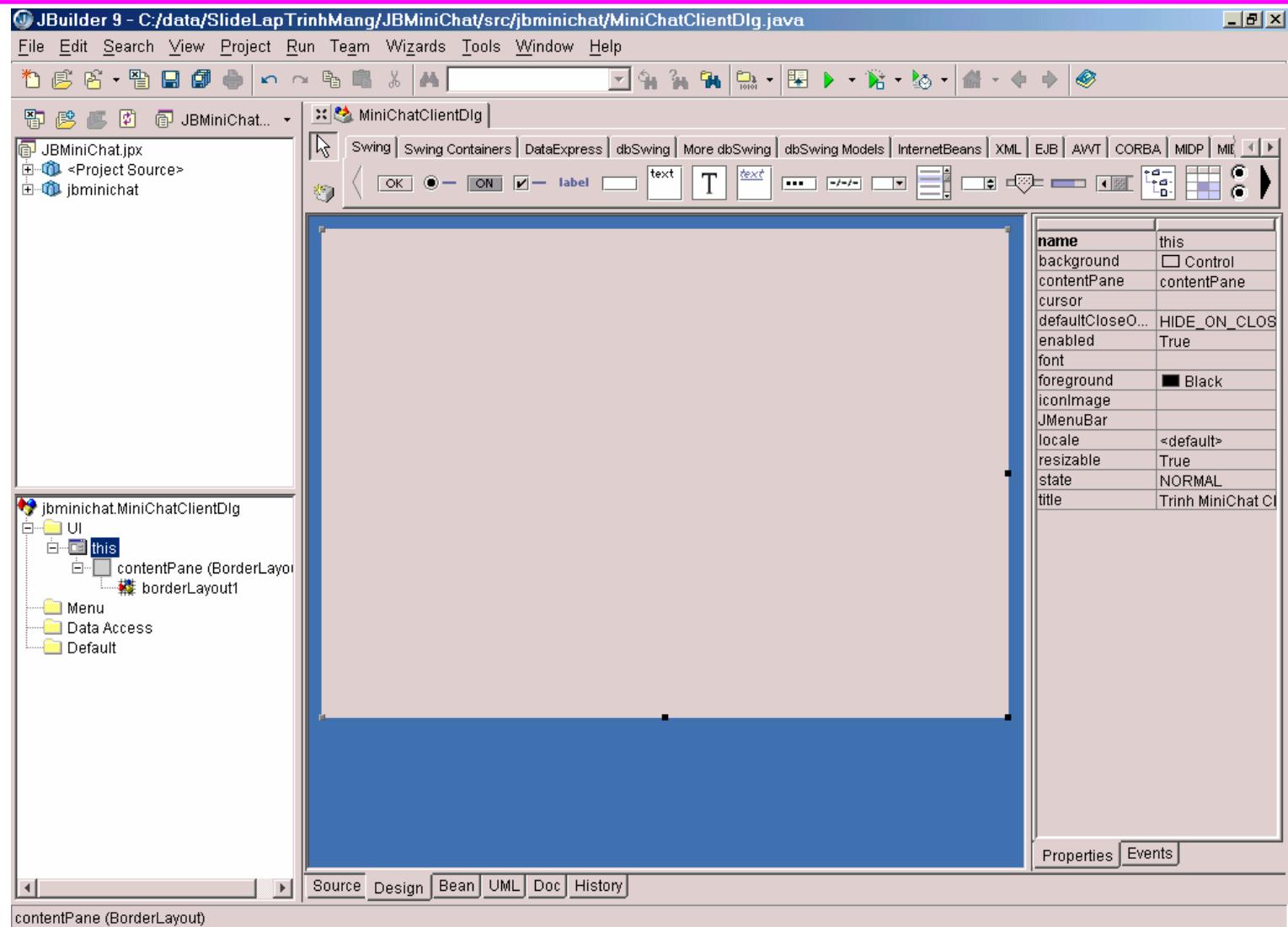
The screenshot shows the JBuilder 9 IDE interface. The title bar reads "JBuilder 9 - C:/data/SlideLapTrinhMang/JBMiniChatClient/src/jbminichatclient/MiniChatClientDlg.java". The menu bar includes File, Edit, Search, View, Project, Run, Team, Wizards, Tools, Window, Help. The toolbar has various icons for file operations. The left pane shows the project structure with files like JBMiNiChatClient.jpx, jbminichatclient, about.png, MiniChatClient.java, MiniChatClientDlg.java, and MiniChatClientDlg_AboutBox.java. The center pane displays the Java code for MiniChatClientDlg.java, and the bottom pane shows the component tree for MiniChatClientDlg.

```
1 package jbminichatclient;
2
3 import java.awt.*;
4 import java.awt.event.*;
5 import javax.swing.*;
6
7 /**
8 * <p>Title: </p>
9 * <p>Description: </p>
10 * <p>Copyright: Copyright (c) 2004</p>
11 * <p>Company: </p>
12 * @author not attributable
13 * @version 1.0
14 */
15
16 public class MiniChatClientDlg extends JFrame {
17     JPanel contentPane;
18     JLabel statusBar = new JLabel();
19     BorderLayout borderLayout1 = new BorderLayout();
20
21     //Construct the frame
22     public MiniChatClientDlg() {
23         enableEvents(AWTEvent.WINDOW_EVENT_MASK);
24         try {
25             jbInit();
26         }
27         catch(Exception e) {
28             e.printStackTrace();
29         }
30     }
31     //Component initialization
32     private void jbInit() throws Exception {
33         contentPane = (JPanel) this.getContentPane();
```



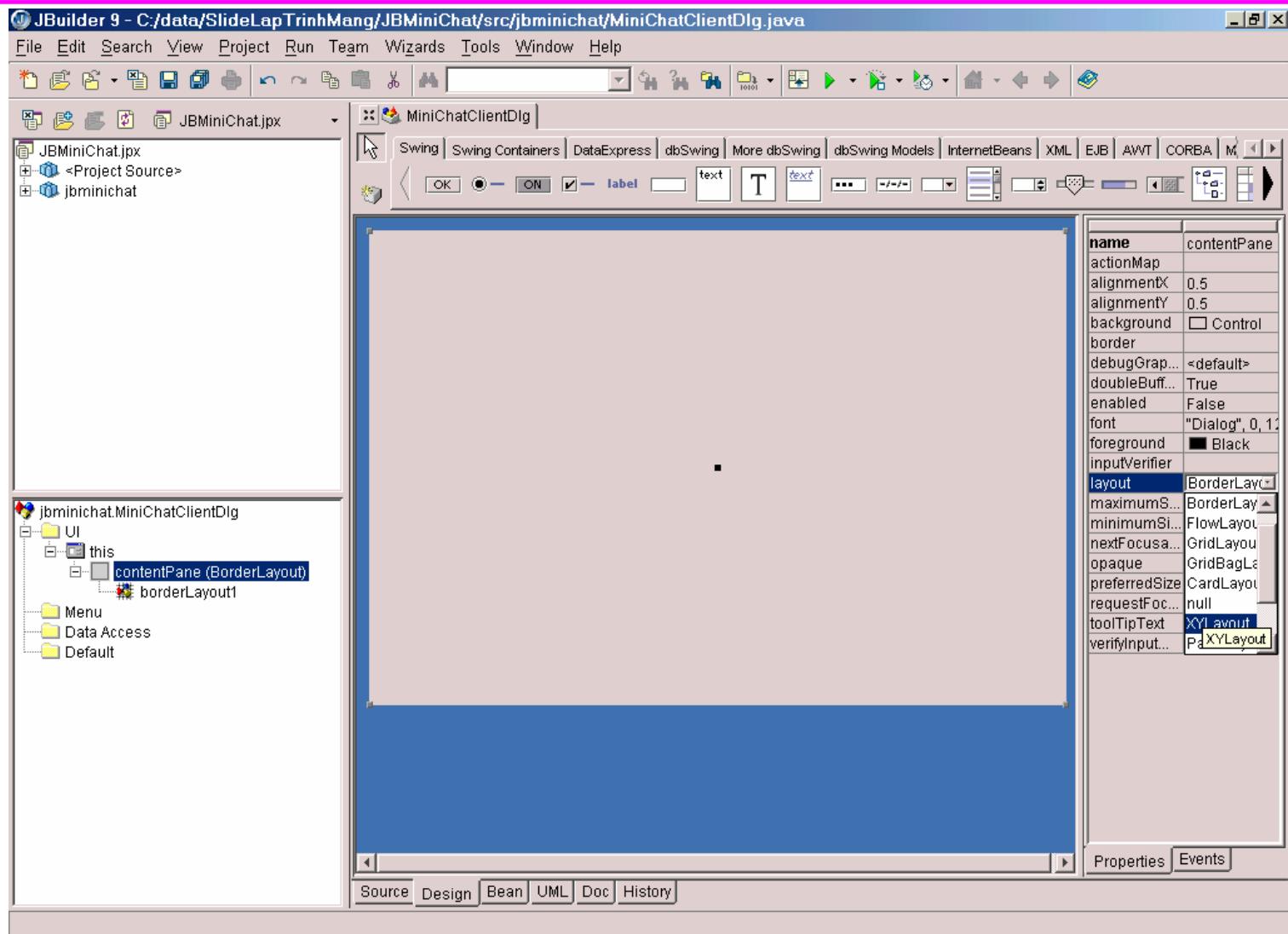
Xây dựng module client MiniChatter

12. Chọn tab "Design" để chuyển sang cửa sổ thiết kế form giao diện :



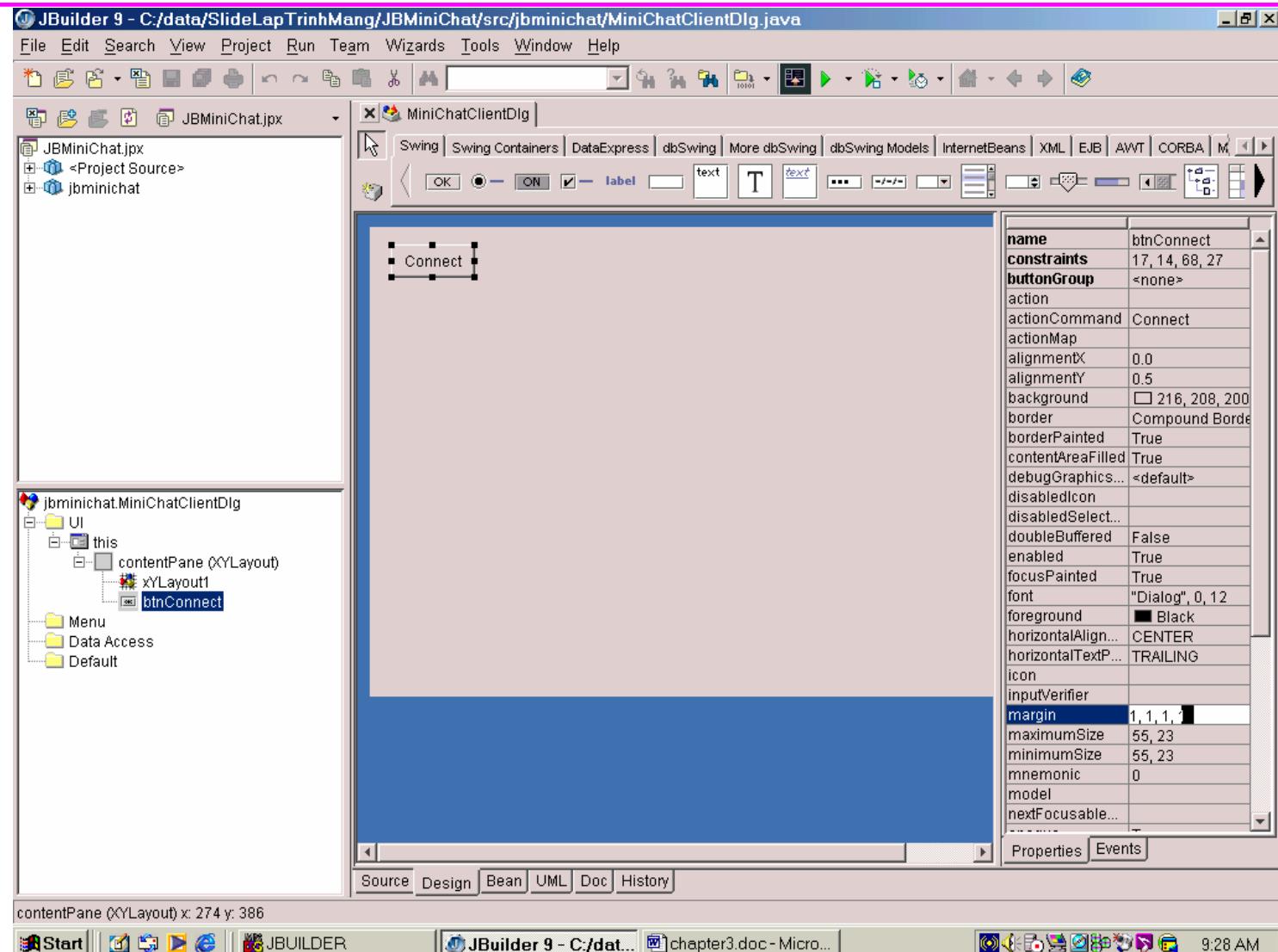
Xây dựng module client MiniChatter

13. Chọn mục "contentPane", thay đổi giá trị thuộc tính layout về "XYLayout" để ta có thể thiết kế các phần tử giao diện 1 cách tự do :



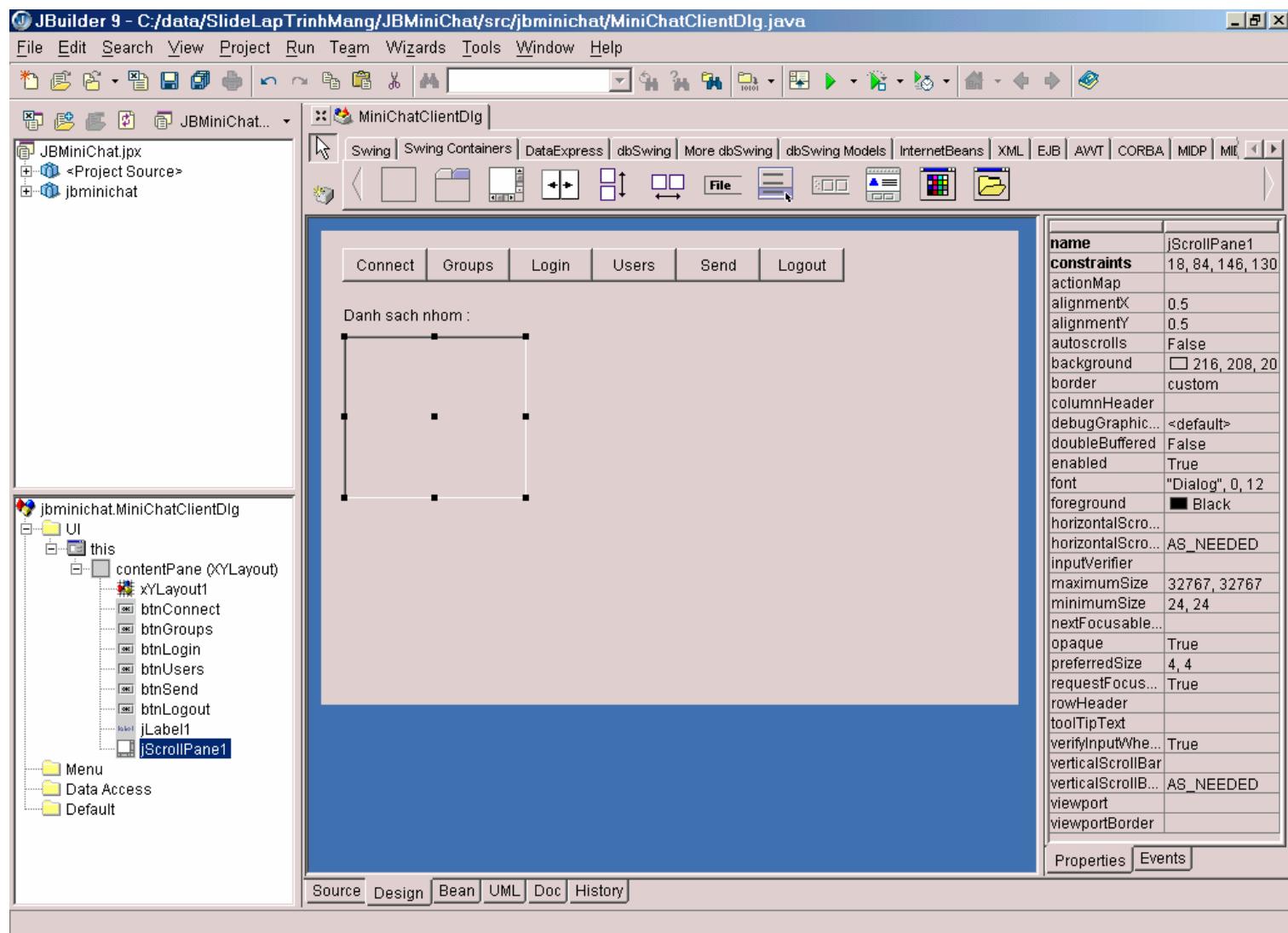
Xây dựng module client MiniChatter

14. Chọn tab "Swing", chọn icon "JButton" rồi vẽ nó ở góc trên trái Form, chọn button vừa vẽ, quan sát và thiết lập giá trị cho thuộc tính name=btnConnect, margin = 1,1,1,1, thuộc tính text=Connect :



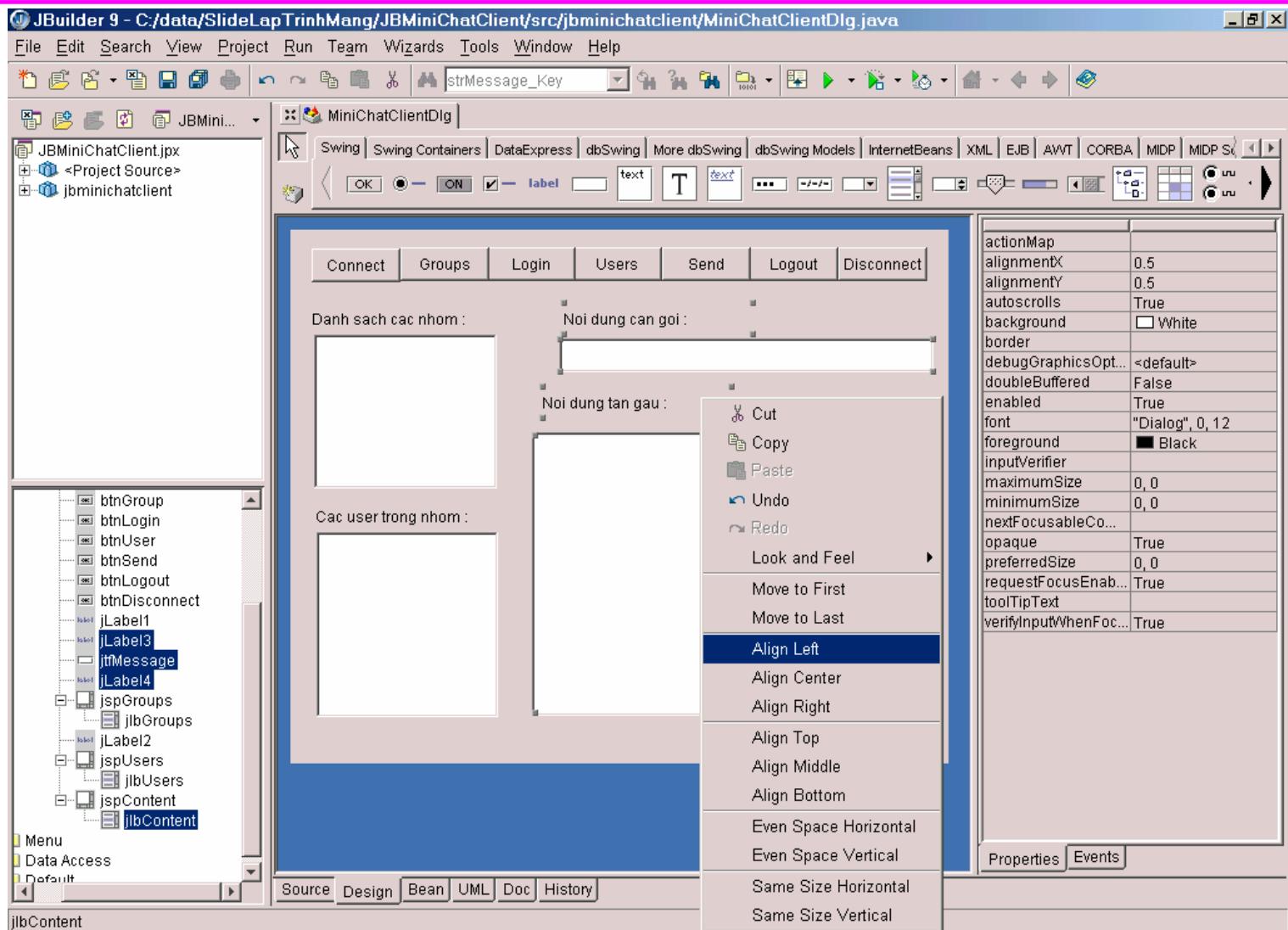
Xây dựng module client MiniChatter

15. Lập lại bước 14
để vẽ các button
Groups, Login,
Users, Send, Logout,
Disconnect. Chọn tab
"Swing Containers",
chọn icon
JScrollPane rồi vẽ 1
scrollpane chứa
listbox hiển thị danh
sách nhóm, thiết lập
field
name=jspGroups. Vẽ
1 JList vào trong
scrollpane và thiết
lập field
name=jlbGroups :



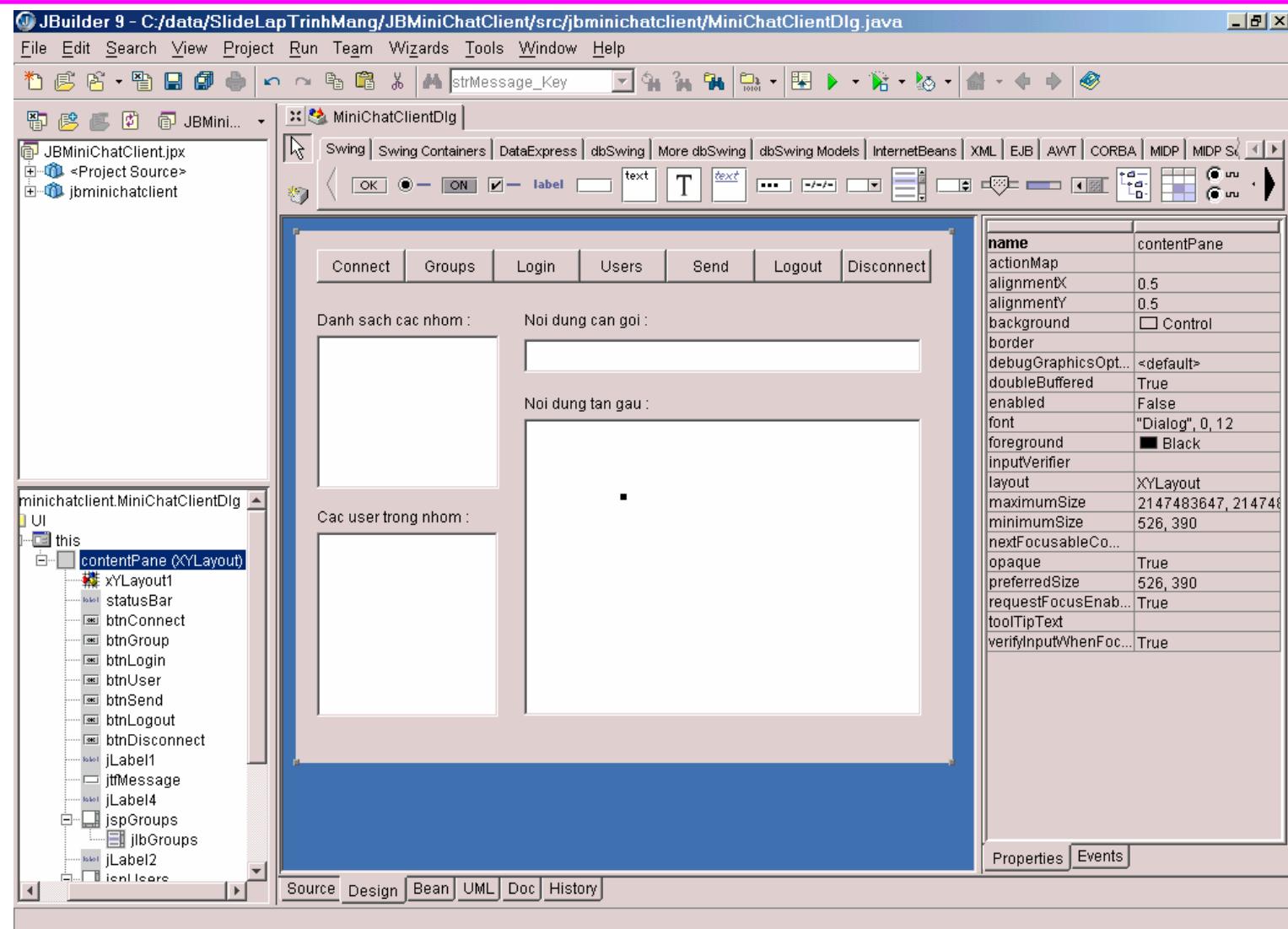
Xây dựng module client MiniChatter

16. Lập lại bước 15
để vẽ các listbox
jlbUsers,
jlbContent, vẽ 1
JTextField và thiết
lập field
name=jtfMessage.
Có thể đóng trái,
trên,... các đối
tượng bằng cách
chọn chúng, ấn
phải chuột để hiển
thị cửa sổ lệnh rồi
chọn mục tương
ứng (thí dụ đóng
trái các đối tượng
sau đây) :



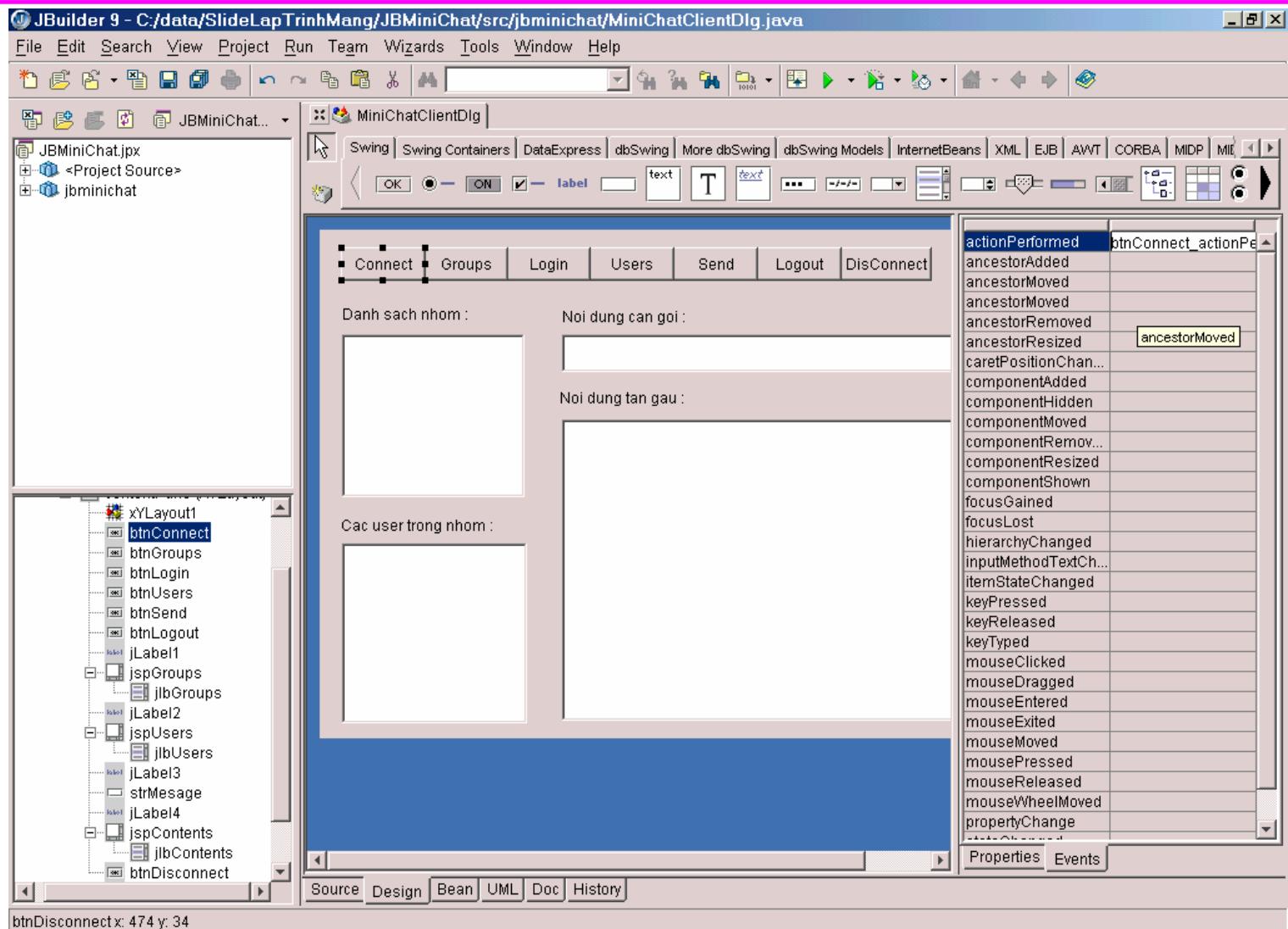
Xây dựng module client MiniChatter

17. Sau khi thiết kế hoàn chỉnh, ta có Form như sau :



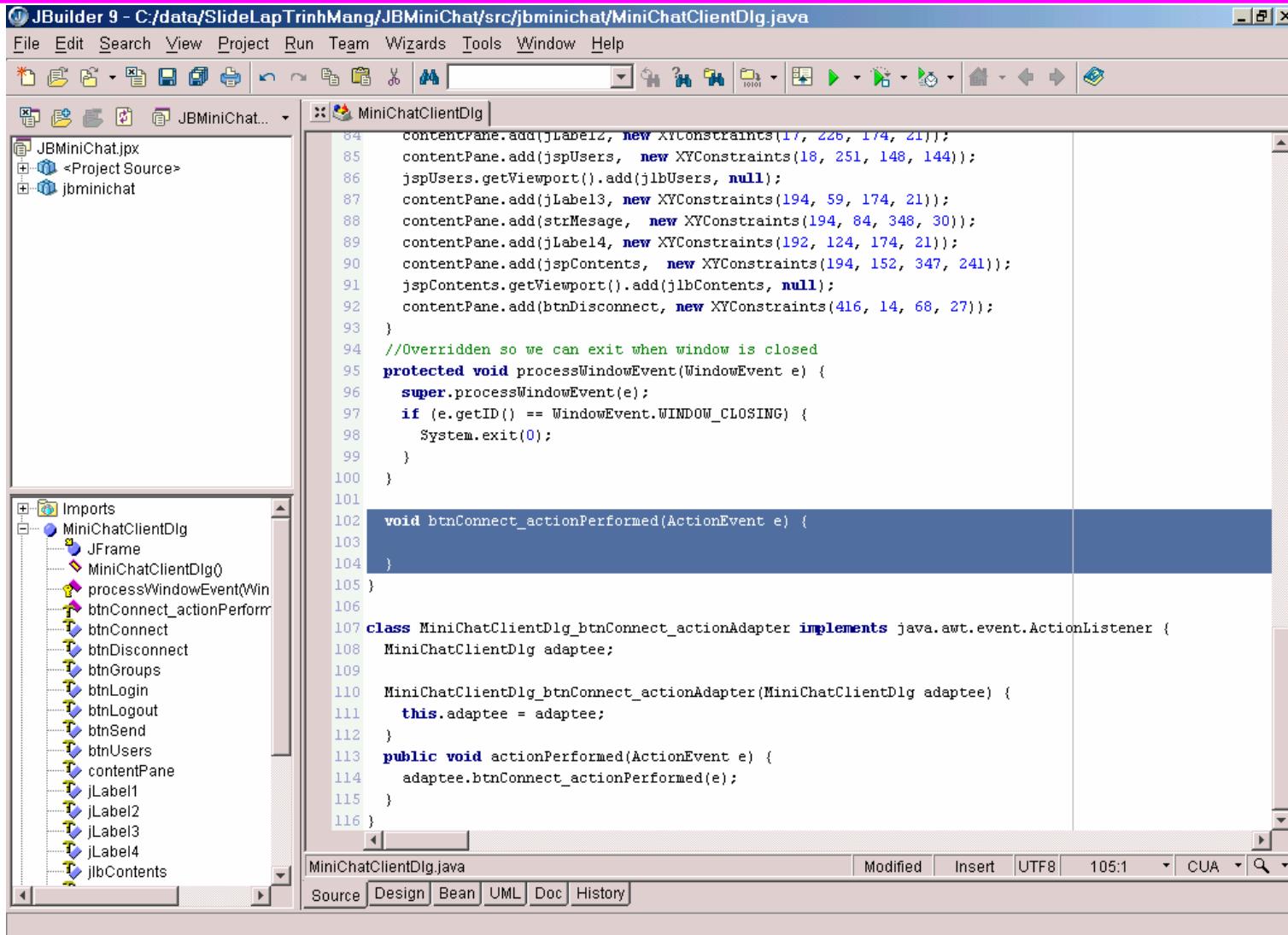
Xây dựng module client MiniChatter

18. Tạo hàm xử lý sự kiện ấn chuột vào button
Connect : chọn button Connect, chọn tab Events, đưa cursor về textbox actionPerformed rồi ấn Enter để tạo hàm xử lý với tên mặc định :



Xây dựng module client MiniChatter

19. Cửa sổ code sẽ hiển thị để bạn có thể hiện thực thân của hàm xử lý vừa tạo. Bạn có thể viết code ngay hay bất kỳ lúc nào sau đó. Lặp lại bước 18 để tạo các hàm xử lý cho các button còn lại :



The screenshot shows the JBuilder 9 IDE interface. The title bar reads "JBuilder 9 - C:/data/SlideLapTrinhMang/JBMiniChat/src/jbminichat/MiniChatClientDlg.java". The menu bar includes File, Edit, Search, View, Project, Run, Team, Wizards, Tools, Window, Help. The toolbar has various icons for file operations. The left pane shows the project structure with "JBMiniChat.jpx" and "jbminichat" selected. The right pane displays the Java code for "MiniChatClientDlg.java". The code includes imports for JFrame, MiniChatClientDlg, processWindowEvent, and several JButton instances. It contains a constructor for the content pane and a protected method processWindowEvent. The main part of the code is a actionPerformed method for the btnConnect button, which calls the btnConnect_actionPerformed method of a MiniChatClientDlg_btnConnect_adapter object. The code uses XYConstraints for layout management.

```
contentPane.add(jLabel1, new XYConstraints(17, 226, 174, 21));
contentPane.add(jspUsers, new XYConstraints(18, 251, 148, 144));
jspUsers.setViewport().add(jlbUsers, null);
contentPane.add(jLabel3, new XYConstraints(194, 59, 174, 21));
contentPane.add(strMesage, new XYConstraints(194, 84, 348, 30));
contentPane.add(jLabel4, new XYConstraints(192, 124, 174, 21));
contentPane.add(jspContents, new XYConstraints(194, 152, 347, 241));
jspContents.setViewport().add(jlbContents, null);
contentPane.add(btnDisconnect, new XYConstraints(416, 14, 68, 27));
}
//Overridden so we can exit when window is closed
protected void processWindowEvent(WindowEvent e) {
    super.processWindowEvent(e);
    if (e.getID() == WindowEvent.WINDOW_CLOSING) {
        System.exit(0);
    }
}

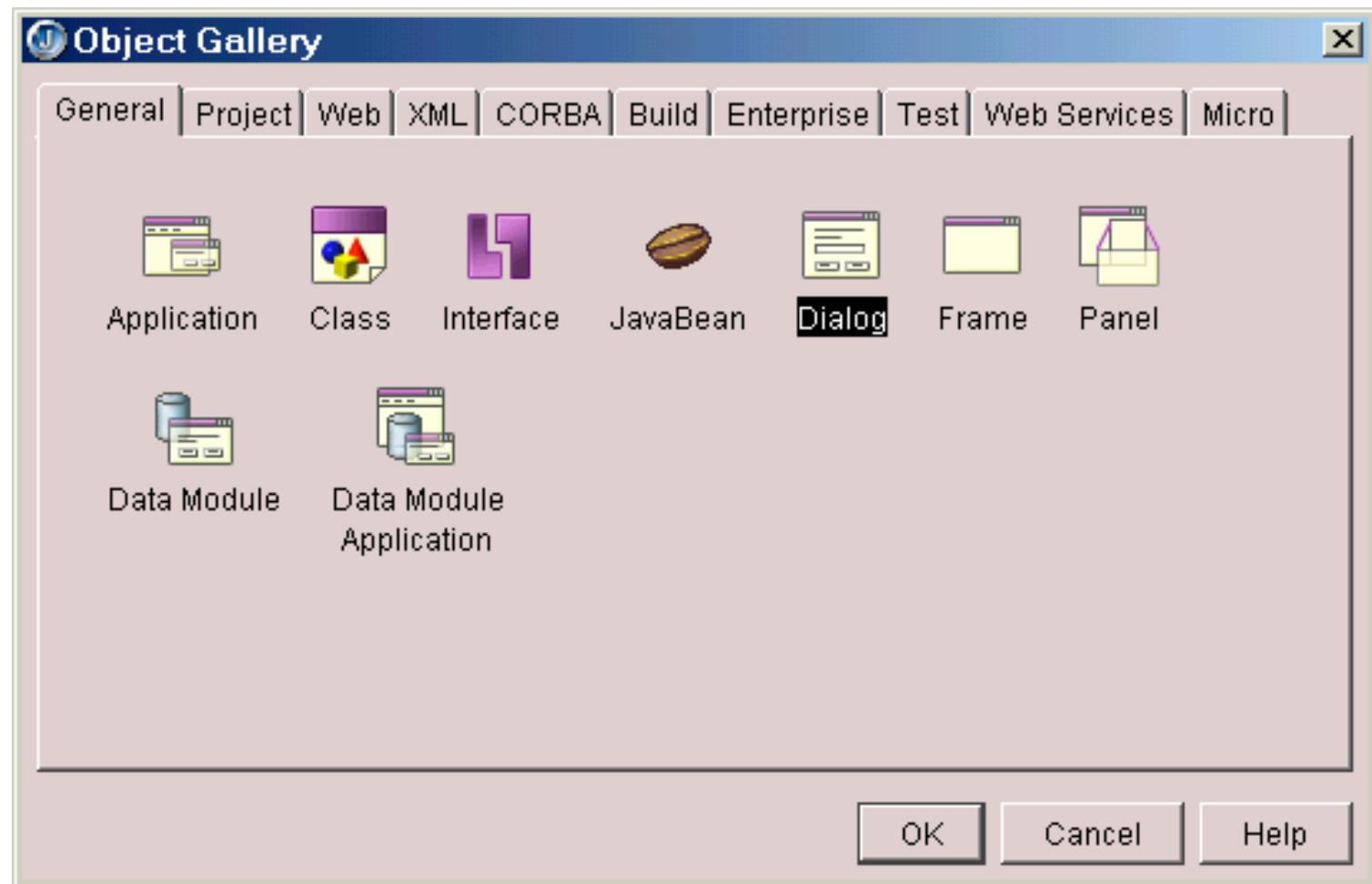
void btnConnectActionPerformed(ActionEvent e) {
}

class MiniChatClientDlg_btnConnect_actionAdapter implements java.awt.event.ActionListener {
    MiniChatClientDlg adaptee;
    MiniChatClientDlg_btnConnect_actionAdapter(MiniChatClientDlg adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnConnectActionPerformed(e);
    }
}
```



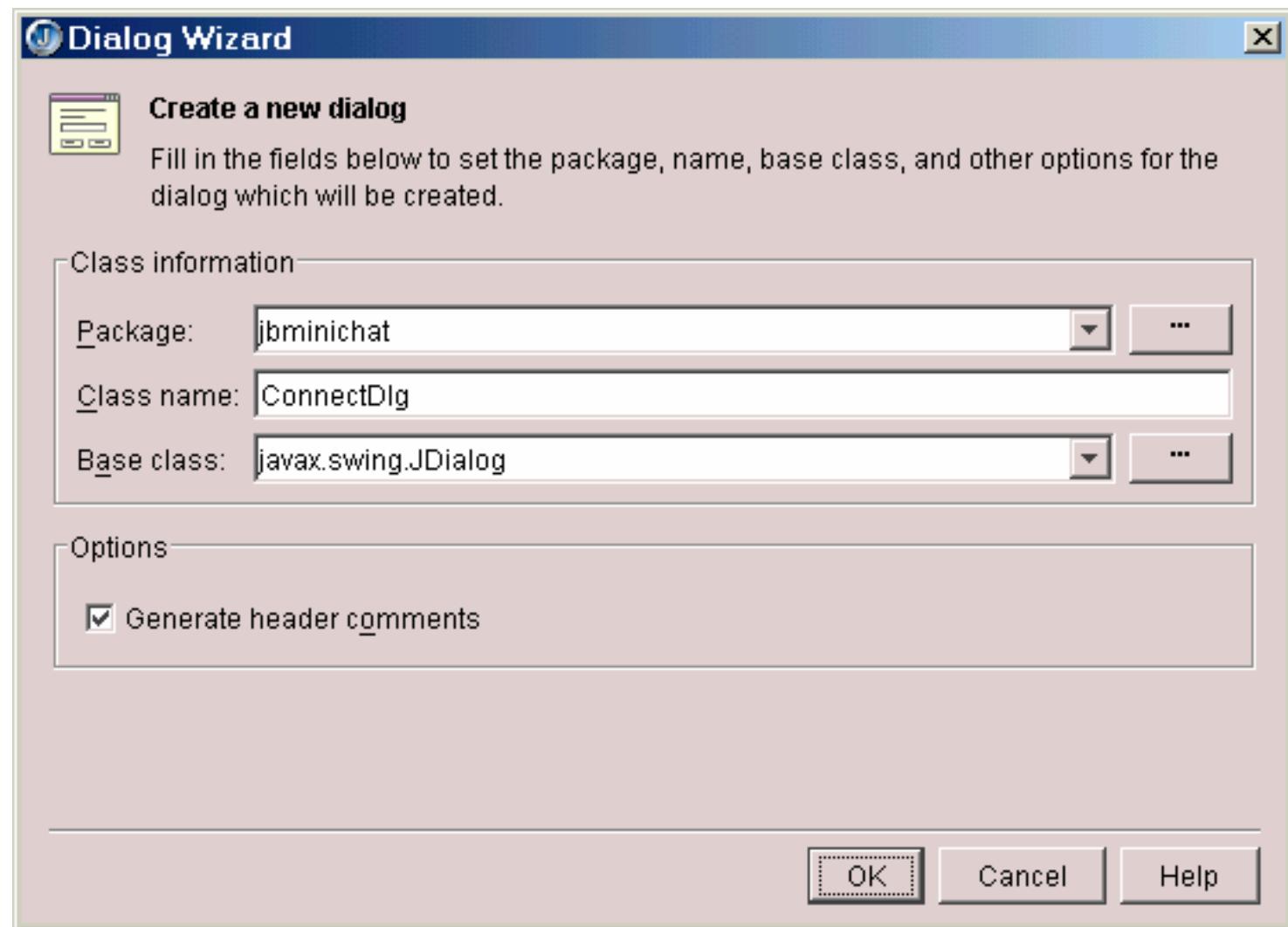
Xây dựng module client MiniChatter

20. Để tạo form nhập liệu "Connect", chọn menu File.New... để hiển thị cửa sổ Object Gallery. Chọn tab General, chọn icon Dialog :



Xây dựng module client MiniChatter

21. Nhập tên class,
chọn class cha cần
thừa kế rồi ấn
button Ok :



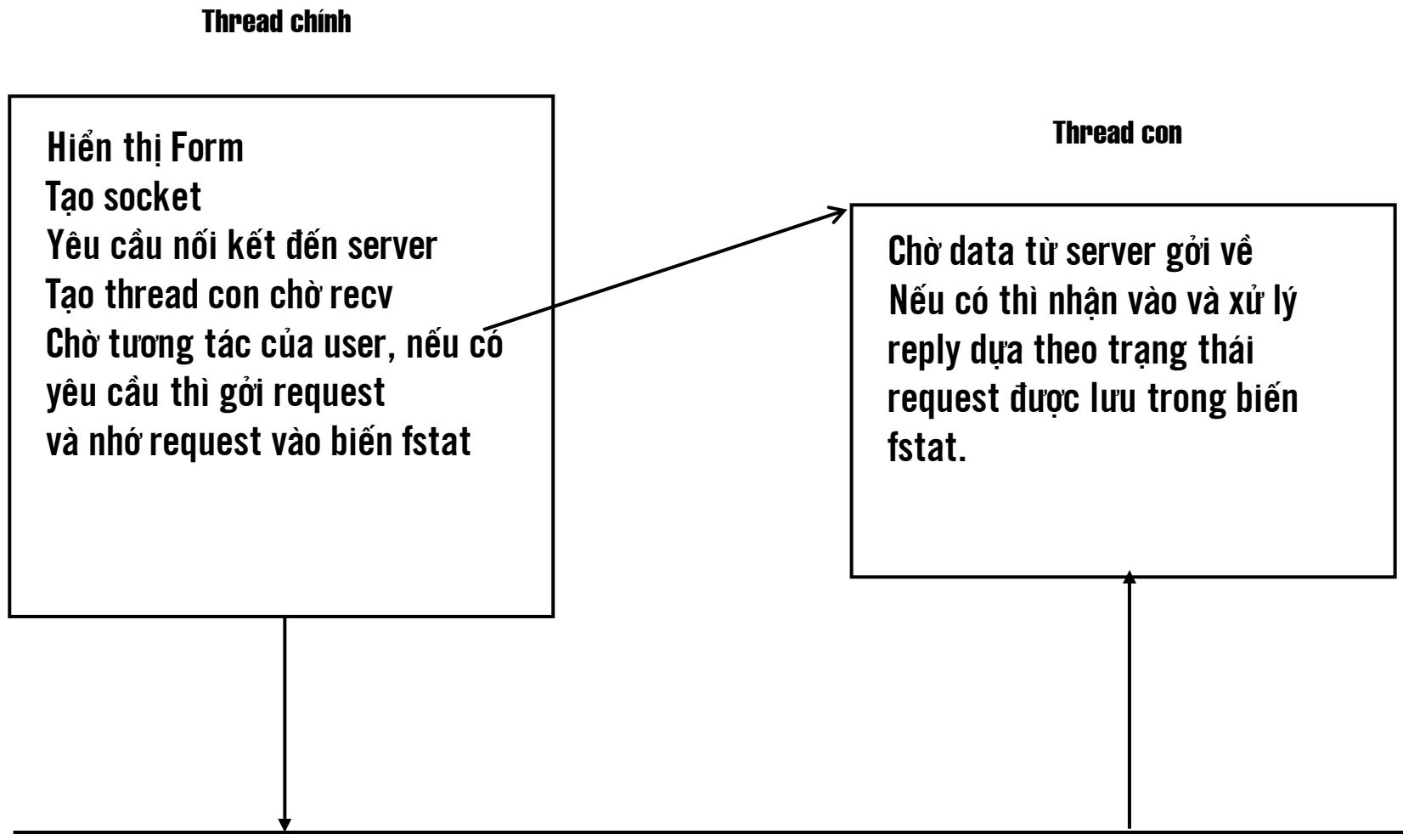
Xây dựng module client MiniChatter

22. Sau khi Dialog mới được tạo ra, bạn thiết kế trực quan và viết code cho nó giống như Form đầu tiên miêu tả ứng dụng. Lặp lại bước 20, 21, 22 cho Form nhập liệu Login. Viết code cho các hàm xử lý sự kiện đã tạo ra trên từng Form giao diện. Nếu cần thiết, viết code tăng cường cho Constructor của Form để thiết lập 1 số giá trị ban đầu cho Form đó. Chọn menu Run.Run Project, chọn tên file quản lý run-time của ứng dụng để chạy và debug nó.



Xây dựng module client MiniChatter

Mô hình multi-thread ở client :



Xây dựng module client MiniChatter

```
// khai báo package của ứng dụng
package jbminichatclient;
// khai báo các package cần dùng trong ứng dụng
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import com.borland.jbcl.layout.*;
import java.net.*;
import java.util.*;
import java.io.*;

// khai báo class miêu tả Form của client
public class MiniChatClientDlg extends JFrame implements MessageListener {
    private Socket socket; // socket giao tiếp với server
    private boolean connected = false; // cờ trạng thái làm việc của client
    private ReceivingThread receivingThread; // handle đến thread con chờ recv
    // các hằng gọi nhớ miêu tả lệnh request cuối cùng đã gửi đi
    private static final int FSGLIST = 0;
    private static final int FSLOGIN = 1;
    private static final int FSLOGOUT = 2;
    private static final int FSULIST = 3;
    private static final int FSMESG = 4;
    private int fstate=FSMESG;
```



Xây dựng module client MiniChatter

```
// các thuộc tính được tạo tự động do thiết kế trực quan  
JPanel contentPane;  
  
...  
//Constructor của Form client  
public MiniChatClientDlg() {  
    enableEvents(AWTEvent.WINDOW_EVENT_MASK);  
    try {  
        jbInit();  
    }  
    catch(Exception e) {  
        e.printStackTrace();  
    }  
    // thiết lập kích thước cho Form  
    this.setSize(new Dimension(540, 440));  
    // tạo Model chứa nội dung cho các đối tượng JList (listbox)  
    DefaultListModel model = new DefaultListModel();  
    jlbGroups.setModel((ListModel)model);  
    model =new DefaultListModel();  
    jlbUsers.setModel((ListModel)model);  
    model =new DefaultListModel();  
    jlbContent.setModel((ListModel)model);  
}
```



Xây dựng module client MiniChatter

```
//Code của các hàm xử lý sự kiện  
// hàm xử lý button btnConnect  
void btnConnectActionPerformed(ActionEvent e) {  
    ConnectDlg dlg = new ConnectDlg(new javax.swing.JFrame(),"Nhập thông tin về  
server",true);  
    dlg.show();  
    if (dlg.fOk) { // thực hiện kết nối  
        try {  
            socket = new Socket(InetAddress.getByName(dlg.txtIPAddress),dlg.intPort);  
            // create Thread for receiving incoming messages  
            receivingThread = new ReceivingThread(this,socket );  
            receivingThread.start();  
        }  
        // handle exception connecting to server  
        catch ( IOException ioException ) {  
            ioException.printStackTrace();  
        }  
        // update connected flag  
        connected = true;  
    }  
}
```



Xây dựng module client MiniChatter

```
// hàm xử lý button btnGroup
void btnGroupActionPerformed(ActionEvent e) {
    String mesg = "GLIST ";
    SendMessage(socket,mesg);
    fstate = FSGLIST;
}

// hàm xử lý button btnLogin
void btnLoginActionPerformed(ActionEvent e) {
    LoginDlg dlg;
    if (jlbGroups.getSelectedItem().toString() == null)
        dlg = new LoginDlg(new javax.swing.JFrame(),"Nhập thông tin login",true,"");
    else
        dlg = new LoginDlg(new javax.swing.JFrame(),"Nhập thông tin
login",true,jlbGroups.getSelectedItem().toString());
    dlg.show();
    if (dlg.fOk) { // thực hiện kết nối
        String request;
        request = "LOGIN "+dlg.jtfGroupName.getText() + "," +
        dlg.jtfUserName.getText();
        SendMessage(socket,request); fstate = FSLOGIN;
    }
}
```



Xây dựng module client MiniChatter

```
// hàm xử lý button btnUser
void btnUserActionPerformed(ActionEvent e) {
    String mesg = "ULIST ";
    SendMessage(socket,mesg);
    fstate = FSULIST;
}

// hàm xử lý button btnSend
void btnSendActionPerformed(ActionEvent e) {
    SendToGroup();
}

// hàm xử lý button btnLogout
void btnLogoutActionPerformed(ActionEvent e) {
    String mesg;
    mesg = "LOGOU ";
    SendMessage(socket,mesg);
    fstate = FSLOGOUT;
}
```



Xây dựng module client MiniChatter

```
// hàm xử lý button btnDisconnect
void btnDisconnectActionPerformed(ActionEvent e) {
    try {
        SendMessage(socket,"LOGOU "); fstate = FSLOGOUT;
    }
    catch ( IOException ioException ) { ioException.printStackTrace(); }
}

// hàm xử lý sự kiện ấn phím trên textbox
void jtfMessageKeyPressed(KeyEvent e) {
    if (e.getKeyCode()==10) SendToGroup();
}

// hàm gửi 1 message đến server
private void SendMessage(Socket sock, String mesg) {
    // send message and flush PrintWriter
    try {
        PrintWriter writer = new PrintWriter( sock.getOutputStream() );
        writer.println(mesg); writer.flush();
    }
    // handle exception sending message
    catch ( IOException ioException ) { ioException.printStackTrace(); }
}
```



Xây dựng module client MiniChatter

```
// hàm gửi 1 message đến các thành viên của nhóm
private void SendToGroup() {
    String mesg = jtfMessage.getText();
    if (mesg != "") SendMessage(socket,"SEND "+mesg); fstate = FSMESG;
}
// hàm nhận 1 message từ Server và xử lý reply (do thread con gọi)
public void messageReceived( Socket sock, String mesg) {
int status;
    if (mesg.compareTo("CLOSE ")==0) return;
    switch (fstate) {
        case FSLOGIN : Do_login(mesg); break;
        case FSLOGOUT: // logout
            Do_logout(mesg); break;
        case FSMESG : // group list
            Do_receive(mesg); break;
        case FSGLIST: // user list
            Do_glist(mesg); break;
        case FSULIST: // broadcast message
            Do_ulist(mesg);
    }
    fstate = FSMESG;
}
```



Xây dựng module client MiniChatter

```
private void Do_login(String mesg) {  
}  
private void Do_logout(String mesg) {  
    if (mesg.charAt(0) == '0') return;  
    DefaultListModel lmUsers = (DefaultListModel)jlbUsers.getModel();  
    lmUsers.clear();  
    DefaultListModel lmContent = (DefaultListModel)jlbContent.getModel();  
    lmContent.clear();  
}  
private void Do_glist(String mesg) {  
    int i;  
    DefaultListModel lmGroups = (DefaultListModel)jlbGroups.getModel();  
    lmGroups.clear();  
    if (mesg.charAt(0) == '0') return;  
    mesg = mesg.substring(2);  
    // tokenize message to retrieve user name and message body  
    StringTokenizer tokenizer = new StringTokenizer(mesg, "," );  
    int size =tokenizer.countTokens();  
    for (i=0; i<size;i++ )  
        lmGroups.addElement( tokenizer.nextToken());  
}
```



Xây dựng module client MiniChatter

```
private void Do_ulist(String mesg) {  
    String gname;  
    int i;  
    DefaultListModel lmUsers = (DefaultListModel)jlbUsers.getModel();  
    lmUsers.clear();  
    if (mesg.charAt(0) == '0') return;  
    if (mesg.length()<3) return;  
    mesg = mesg.substring(2);  
    // tokenize message to retrieve user name and message body  
    StringTokenizer tokenizer = new StringTokenizer(mesg, "," );  
    int size = tokenizer.countTokens();  
    for (i=0; i<size;i++ )  
        lmUsers.addElement( tokenizer.nextToken());  
}  
  
private void Do_receive(String mesg) {  
    DefaultListModel lmContent = (DefaultListModel)jlbContent.getModel();  
    // append new message  
    lmContent.addElement(mesg);  
}
```



Xây dựng module client MiniChatter

Để đặc tả thread xử lý recv (là 1 class Java), bạn chọn menu File.New... để hiện thị cửa sổ Object Gallery, chọn tab General, chọn icon Class rồi trả lời 1 số thông số để xác định class cần tạo ra. Sau đó viết code sau cho class :

// Code của class ReceivingThread : miêu tả hoạt động của thread con chờ nhận reply

package jbminichatclient;

import java.io.*;

import java.net.*;

import java.util.StringTokenizer;

```
public class ReceivingThread extends Thread {
```

```
    private BufferedReader input;
```

```
    private MessageListener messageListener;
```

```
    private boolean keepListening = true;
```

```
    Socket socket;
```

```
    // ReceivingThread constructor
```

```
    public ReceivingThread( MessageListener listener, Socket clientSocket ) {
```

```
        // invoke superclass constructor to name Thread
```

```
        super( "ReceivingThread: " + clientSocket );
```

```
        // set listener to which new messages should be sent
```

```
        messageListener = listener;
```

```
        socket = clientSocket;
```



Xây dựng module client MiniChatter

```
// set timeout for reading from clientSocket and create  
// BufferedReader for reading incoming messages  
try {  
    socket.setSoTimeout( 5000 );  
    input = new BufferedReader( new InputStreamReader(  
        socket.getInputStream() ) );  
}  
// handle exception creating BufferedReader  
catch ( IOException ioException ) { ioException.printStackTrace(); }  
} // end ReceivingThread constructor  
  
// listen for new messages and deliver them to MessageListener  
public void run() {  
    String message;  
    // listen for messages until stopped  
    while ( keepListening ) {  
        // read message from BufferedReader  
        try {  
            message = input.readLine();  
        }
```



Xây dựng module client MiniChatter

```
// handle exception if read times out
catch ( InterruptedIOException interruptedIOException ) {
    // continue to next iteration to keep listening
    continue;
}
// handle exception reading message
catch ( IOException ioException ) {
    messageListener.messageReceived(
        socket, // user name
        "CLOSE "); // message body
    break;
}
// ensure non-null message
if ( message != null )
    // send message to MessageListener
    messageListener.messageReceived(socket, // user name
        message); // message body
} // end while
```



Xây dựng module client MiniChatter

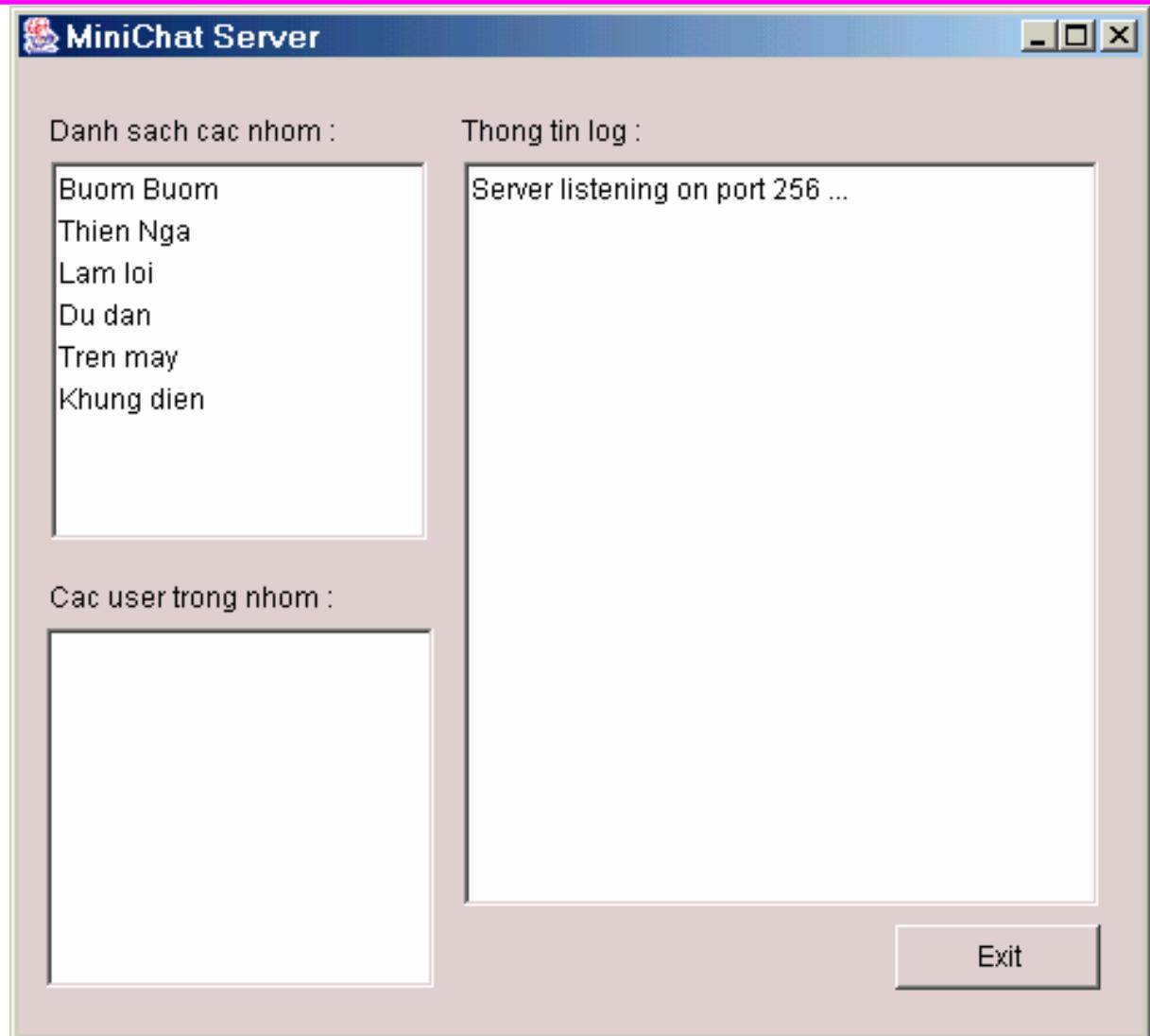
```
// close BufferedReader (also closes Socket)
try {
    input.close();
}
// handle exception closing BufferedReader
catch ( IOException ioException ) {
    ioException.printStackTrace();
}
} // end method run

// stop listening for incoming messages
public void stopListening()
{
    keepListening = false;
}
} // end class ReceivingThread
```



Xây dựng module Server MiniChatter

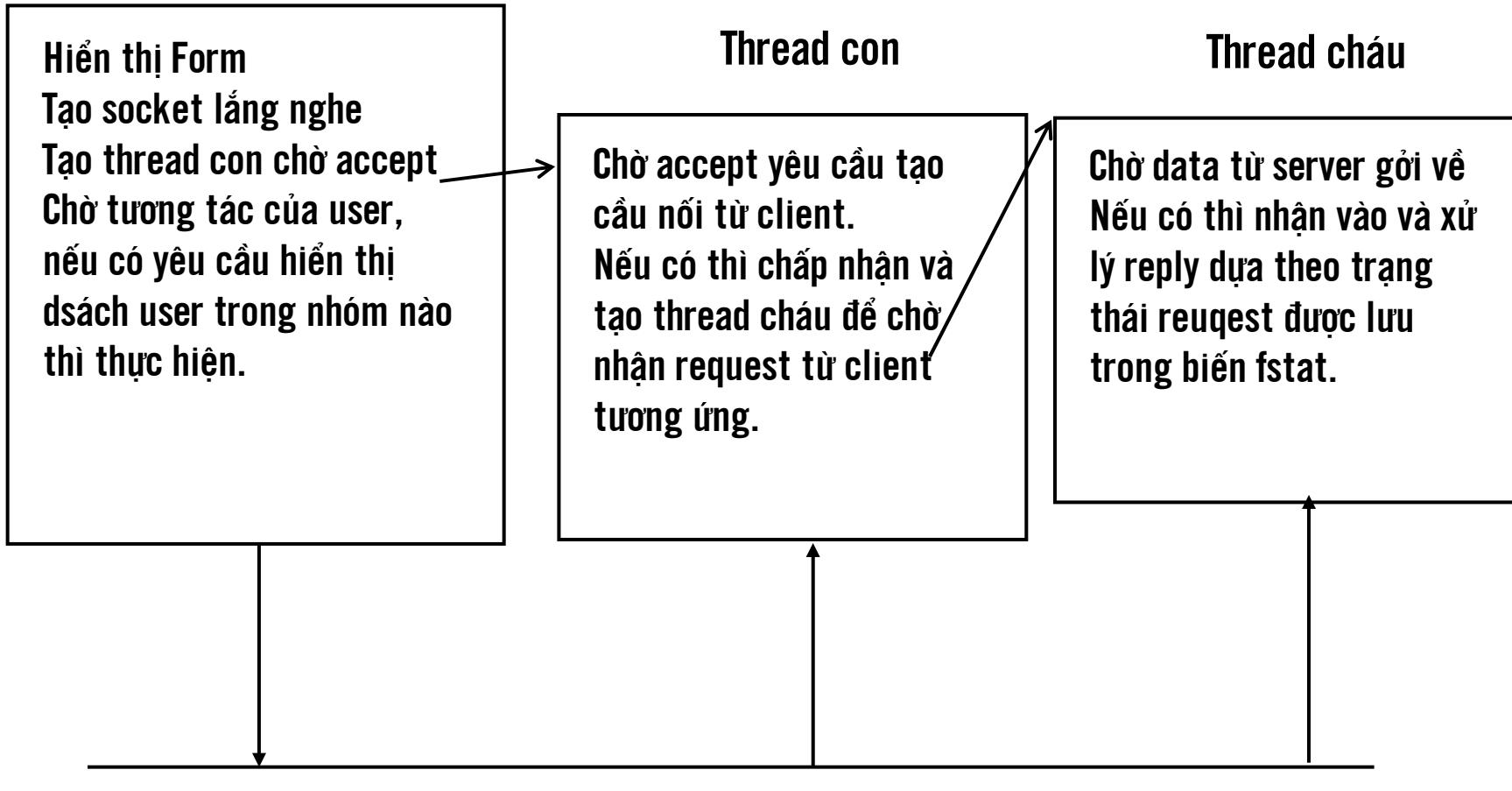
Qui trình tạo ứng dụng Server cũng giống như Client. Bạn tạo Form giao diện có dạng như sau :



Xây dựng module Server MiniChatter

Thread chính

Mô hình các thread ở Server :



Xây dựng module Server MiniChatter

```
// code của file MiniChatServerDlg
package jbminichatclient;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import com.borland.jbcl.layout.*;
import java.util.*;
import java.net.*;
import java.io.*;
import java.sql.*;
public class MiniChatServerDlg extends JFrame implements MessageListener {
    //các thuộc tính cần dùng cho Server
    int m_groupcnt; // số nhóm
    T_UserRec m_sock_no_user=null; // danh sách các user chưa login
    T_GroupList m_grouplist[] = new T_GroupList[10]; // danh sách nhóm
    String uname; // tên user
    ServerSocket serverSocket; // socket mà server dùng để accept
    final int SERVER_PORT = 256; // port lắng nghe của server
    //các thuộc tính cần dùng cho Server
    JPanel contentPane;
    ...
}
```



Xây dựng module Server MiniChatter

```
//Constructor của frame  
public MiniChatServerDlg() {  
    enableEvents(AWTEvent.WINDOW_EVENT_MASK);  
    try {  
        jbInit();  
    }  
    catch(Exception e) {  
        e.printStackTrace();  
    }  
    // thiết lập kích thước của form  
    this.setSize(new Dimension(490, 445));  
    // tạo các model chứa nội dung của các listbox  
    DefaultListModel model = new DefaultListModel();  
    jlbGroups.setModel((ListModel)model);  
    model =new DefaultListModel();  
    jlbUsers.setModel((ListModel)model);  
    model =new DefaultListModel();  
    jlbContent.setModel((ListModel)model);  
    // truy xuất danh sách nhóm từ database và hiển thị lên listbox Groups  
    ReadDisplayGroups();
```



Xây dựng module Server MiniChatter

```
// tao sersersocket lắng nghe cho server
try {
    serverSocket = new ServerSocket( SERVER_PORT, 100 );
    DefaultListModel lmContent = (DefaultListModel)jlbContent.getModel();
    lmContent.addElement("Server listening on port " + SERVER_PORT + " ...");
    // tạo thread con để chờ accept
    new ServerAcceptThread(this,serverSocket).start();
} // end try
// handle exception creating server and connecting clients
catch ( IOException ioException ) {
    ioException.printStackTrace();
}
// đọc danh sách nhóm từ database và hiển thị
private void ReadDisplayGroups(){
    String conStr = "jdbc:odbc:GroupList";
    Connection con;
    String newSQL = "Select * from GroupList";
    String[] data = {"dummy"};
    DefaultListModel lmGroups = (DefaultListModel)jlbGroups.getModel();
```



Xây dựng module Server MiniChatter

```
try {  
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
    //1. Tao connection mieu ta database can truy xuat  
    con = DriverManager.getConnection(conStr,"","","");  
    //2. Tao 1 doi tuong Statement lien ket den connection  
    java.sql.Statement stmt = con.createStatement();  
    //4. Tao doi tuong recordset chua ket qua cua lenh SQL  
    ResultSet rs =stmt.executeQuery(newSQL);  
    //5. Duyet recordset de xu ly cac record cua no  
    int i = 0; lmGroups.clear();  
    if (rs != null) while (rs.next()) {  
        m_grouplist[i] = new T_GroupList();  
        m_grouplist[i].name = rs.getString("groupname");  
        lmGroups.addElement(m_grouplist[i].name);  
        i++;  
    }  
    m_groupcnt = i;  
    //6. Dong cac dong tuong da tao ra  
    rs.close(); stmt.close(); con.close();  
} catch (Exception e) {System.out.println("Error : "+e);}  
}
```

}



Xây dựng module Server MiniChatter

```
// hàm xử lý request đến từ socket sock
public void messageReceived( Socket sock, String mesg ) {
int status;
String opcode = mesg.substring(0,5);
if (opcode.compareTo("LOGIN")==0) { // login
    Do_login(sock,mesg);
} else if (opcode.compareTo("LOGOU")==0) { // logout
    Do_logout(sock);
} else if (opcode.compareTo("GLIST")==0) { // group list
    Do_glist(sock);
} else if (opcode.compareTo("ULIST")==0) { // user list
    Do_ulist(sock);
} else if (opcode.compareTo("CLOSE")==0) { // user list
    Do_CloseSock(sock);
} else { // broadcast message
    Do_MulticastMesg(sock,mesg.substring(5));
}
}
```



Xây dựng module Server MiniChatter

```
private void Do_glist(Socket sock) {  
    String mesg,mesg1;  
    int i;  
    mesg = "1 "+m_grouplist[0].name;  
    for (i=1;i <m_groupcnt;i++) mesg = mesg + "," +m_grouplist[i].name;  
    SendMessage(sock,mesg);  
}  
  
private int Findgroup(Socket sock) {  
    int i; T_UserRec pu;  
    for (i=0; i<m_groupcnt; i++) {  
        pu = m_grouplist[i].userlist;  
        while (pu != null) {  
            if (pu.sock == sock) {  
                uname = pu.name; return i;  
            }  
            pu = pu.next;  
        }  
    }  
    return -1;  
}
```



Xây dựng module Server MiniChatter

```
private void Do_ulist(Socket sock) {  
    String mesg; int i = Findgroup(sock);  
    if (i < 0 || m_grouplist[i].userlist == null) mesg = "0 ";  
    else {  
        T_UserRec pu = m_grouplist[i].userlist;  
        mesg = "1 "; while (pu != null) {  
            mesg = mesg + pu.name; pu = pu.next;  
            if (pu != null) mesg = mesg + ",";  
        }  
    }  
    SendMessage(sock,mesg);  
}  
  
private void Do_MulticastMesg(Socket sock, String mesg) {  
    int i = Findgroup(sock);  
    if (i < 0) return;  
    mesg = uname+ ":" + mesg;  
    T_UserRec pu = m_grouplist[i].userlist;  
    while (pu != null) {  
        SendMessage(pu.sock,mesg); pu = pu.next;  
    }  
}
```



Xây dựng module Server MiniChatter

```
// Dong cau noi tuong ung
private void Do_CloseSock(Socket sock) {
int i = Findgroup(sock);
T_UserRec pu, pup = null;
if (i >= 0) {
    pu = m_grouplist[i].userlist;
    while (pu!=null && pu.sock != sock) {
        pup = pu; pu = pu.next;
    }
    if (pu==m_grouplist[i].userlist) m_grouplist[i].userlist = pu.next;
    else pup.next = pu.next;
} else {
    pu = m_sock_no_user;
    while (pu!= null && pu.sock != sock) { pup = pu; pu = pu.next; }
    if (pu == m_sock_no_user) m_sock_no_user = pu.next;
    else pup.next = pu.next;
}
try {
    sock.close();
}
```



Xây dựng module Server MiniChatter

```
// handle exception connecting to server
catch ( IOException ioException ) {
    ioException.printStackTrace();
}
}

private void Do_login(Socket sock, String mesg) {
T_UserRec pup,pu;
String gname, uname;
int i;
mesg = mesg.substring(6);
// tokenize message to retrieve user name and message body
 StringTokenizer tokenizer = new StringTokenizer(mesg, ",");
// ignore messages that do not contain a user
// name and message body
if ( tokenizer.countTokens() == 2 ) {
    gname = tokenizer.nextToken();
    uname = tokenizer.nextToken();
    // tim group tuong ung
    for (i=0; i<m_groupcnt; i++)
        if (gname.compareTo(m_grouplist[i].name) == 0) break;
    System.out.println(gname + " " + uname + " " + i);
}
```



Xây dựng module Server MiniChatter

```
if (i < m_groupcnt) {  
    // tim sokcet trong danh sach chua co user  
    pup = pu = m_sock_no_user;  
    while (pu!= null && pu.sock != sock) { pup = pu; pu = pu.next; }  
    if (pu!= null) {  
        if (pu == m_sock_no_user)  
            m_sock_no_user = pu.next;  
        else pup.next = pu.next;  
        pu.next = m_grouplist[i].userlist;  
        pu.name = uname;  
        m_grouplist[i].userlist = pu;  
        uname = "1 ";  
        SendMessage(sock,uname);  
        return;  
    }  
}  
}  
}  
}  
errordisp:  
    uname ="0 ";  
    SendMessage(sock,uname);  
}
```



Xây dựng module Server MiniChatter

```
private void Do_logout(Socket sock) {  
    int i = Findgroup(sock);  
    T_UserRec pu, pup=null;  
    if (i >= 0) {  
        pu = m_grouplist[i].userlist;  
        while (pu!= null && pu.sock != sock) { pup = pu; pu = pu.next; }  
        if (pu==m_grouplist[i].userlist) m_grouplist[i].userlist = pu.next;  
        else pup.next = pu.next;  
        pu.next = m_sock_no_user; m_sock_no_user = pu; SendMessage(sock,"1 ");  
    } else SendMessage(sock,"0 ");  
}  
public void SendMessage(Socket sock, String mesg) {  
    // send message and flush PrintWriter  
    try {  
        PrintWriter writer = new PrintWriter( sock.getOutputStream() );  
        writer.println(mesg); writer.flush();  
    }  
    // handle exception sending message  
    catch ( IOException ioException ) { ioException.printStackTrace(); }  
}
```



Xây dựng module Server MiniChatter

```
// hàm xử lý sự kiện ấn chuột vào 1 phần tử của listbox Group
void jlbGroups_mouseClicked(MouseEvent e) {
    String gname = jlbGroups.getSelectedValue().toString();
    int i,j;
    if (gname == null) return;
    for (i = 0; i < m_groupcnt; i++)
        if (gname.compareTo(m_grouplist[i].name) == 0) break;
    if (i >= m_groupcnt) return;
    T_UserRec ulist = m_grouplist[i].userlist;
    DefaultListModel lmUsers = (DefaultListModel)jlbUsers.getModel();
    lmUsers.clear();
    while (ulist != null) {
        lmUsers.addElement(ulist.name);
        ulist = ulist.next;
    }
}
```



Xây dựng module Server MiniChatter

```
// code của thread chờ accept
package jbminichatclient;
import java.util.*;
import java.net.*;
import java.io.*;
import javax.swing.*;
public class ServerAcceptThread extends Thread {
    ServerSocket serverSocket;
    MiniChatServerDlg serverChat;

    public ServerAcceptThread(MiniChatServerDlg server, ServerSocket sock) {
        serverSocket = sock;
        serverChat = server;
    }

    public void run() {
        T_UserRec puser;
        try {
            // listen for clients constantly
            while (true) {
```



Xây dựng module Server MiniChatter

```
// accept new client connection
Socket clientSocket = serverSocket.accept();
puser = new T_UserRec();
puser.sock = clientSocket;
puser.next = serverChat.m_sock_no_user;
serverChat.m_sock_no_user = puser;
// create new ReceivingThread for receiving messages from client
new ReceivingThread(serverChat, clientSocket).start();
// print connection information
DefaultListModel lmContent =
    (DefaultListModel)serverChat.jlbContent.getModel();
lmContent.addElement("Connection received from: " +
    clientSocket.getInetAddress());
serverChat.SendMessage(clientSocket,"Request accepted");
} // end while
}
// handle exception creating server and connecting clients
catch ( IOException ioException ) { ioException.printStackTrace(); }
}
```



Xây dựng module Server MiniChatter

```
// accept new client connection
Socket clientSocket = serverSocket.accept();
puser = new T_UserRec();
puser.sock = clientSocket;
puser.next = serverChat.m_sock_no_user;
serverChat.m_sock_no_user = puser;
// create new ReceivingThread for receiving messages from client
new ReceivingThread(serverChat, clientSocket).start();
// print connection information
DefaultListModel lmContent =
    (DefaultListModel)serverChat.jlbContent.getModel();
lmContent.addElement("Connection received from: " +
    clientSocket.getInetAddress());
serverChat.SendMessage(clientSocket,"Request accepted");
} // end while
}
// handle exception creating server and connecting clients
catch ( IOException ioException ) { ioException.printStackTrace(); }
}
```



MÔN LẬP TRÌNH MẠNG

Chương 4

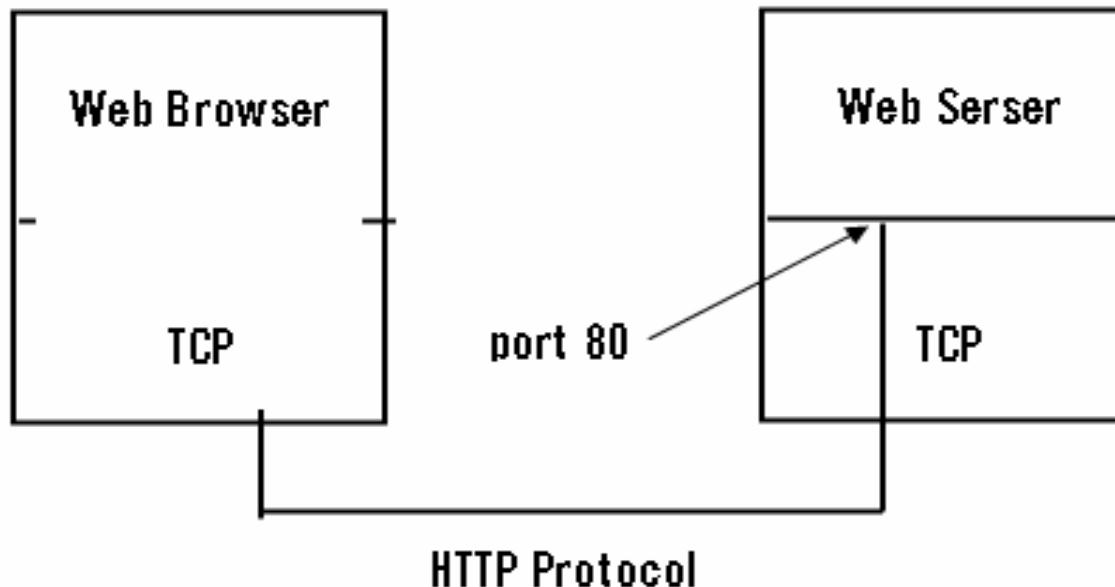
TỔNG QUÁT VỀ LẬP TRÌNH WEB

- 4.1 Ôn lại hệ thống Web
- 4.2 Ôn lại ngôn ngữ HTML
- 4.3 Tổng quát về lập trình Web
- 4.4 Các công nghệ lập trình Web chạy ở Client
- 4.5 Các công nghệ lập trình Web chạy ở Server



4.1 Ôn lại hệ thống Web

IV.1 Mô hình hoạt động



Gồm 2 loại phần tử :

- Web server : chương trình quản lý một cây thư barker các trang Web và phục vụ yêu cầu truy xuất chúng từ các client từ xa.
- Web Browser : chương trình giao tiếp với người dùng, nhận yêu cầu từ user rồi truy xuất trang Web ở server tương ứng để phân giải và hiển thị nội dung lên màn hình.

Giao thức HTTP

HTTP (HyperText Transfer Protocol)

Method	Description
GET	Request to read a Web page
HEAD	Request to read a Web page's header
PUT	Request to store a Web page
POST	Append to a named resource (e.g., a Web page)
DELETE	Remove the Web page
LINK	Connects two existing resources
UNLINK	Breaks an existing connection between two resources



4.2 Ôn lại ngôn ngữ HTML

<HTML> ... </HTML>	Declares the Web page to be written in HTML
<HEAD> ... </HEAD>	Delimits the page's head
<TITLE> ... </TITLE>	Defines the title (not displayed on the page)
<BODY> ... </BODY>	Delimits the page's body
<H _n > ... </H _n >	Delimits a level <i>n</i> heading
 ... 	Set ... in boldface
<I> ... </I>	Set ... in italics
 ... 	Brackets an unordered (bulleted) list
 ... 	Brackets a numbered list
<MENU> ... </MENU>	Brackets a menu of items
	Start of a list item (there is no)
 	Force a break here
<P>	Start of paragraph
<HR>	Horizontal rule
<PRE> ... </PRE>	Preformatted text; do not reformat
	Load an image here
 ... 	Defines a hyperlink



Ôn lại ngôn ngữ HTML

<TABLE> </TABLE>	Bảng thông tin 2 chiều
<TR> </TR>	hàng thuộc bảng
<TH> ... </TH>	phần tử header
<TD> ... </TD>	phần tử dữ liệu

Mục đích của các tag lệnh HTML cơ bản :

- Miêu tả nội dung của trang Web cùng định dạng của từng nội dung.
- chèn nội dung ngoại trực tiếp vào trang Web (thay vì miêu tả trực tiếp)
- tag
- liên kết đến nội dung ngoại và cho phép người dùng quyết định truy xuất nội dung ngoại - tag



Khái niệm URL

URL (Uniform Resource Locator) là phương tiện giải quyết đồng thời 3 chức năng :

- Xác định giao thức được dùng để truy xuất tài nguyên.
- Xác định địa chỉ máy (thường là địa chỉ DNS) chứa tài nguyên.
- Xác định vị trí cụ thể của tài nguyên trên máy (pathname).

Name	Used for	Example
http	Hypertext (HTML)	http://www.cs.vu.nl/~ast/
ftp	FTP	ftp://ftp.cs.vu.nl/pub/minix/README
file	Local file	/usr/suzanne/prog.c
news	News group	news:comp.os.minix
news	News article	news:AA0134223112@cs.utah.edu
gopher	Gopher	gopher://gopher.tc.umn.edu/11/Libraries
mailto	Sending email	mailto:kim@acm.org
telnet	Remote login	telnet://www.w3.org:80



Thí dụ 1 trang Web cơ bản

http://hiepcomp1/DanhbaDienthoai/webbase.htm - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Search Favorites History Go Links

Address http://hiepcomp1/DanhbaDienthoai/webbase.htm

Đây là tiêu đề lớn cấp 1



Đây là đoạn văn chữ thường, còn đây là đoạn văn chữ nghiêng, và đây là đoạn văn chữ đậm, Đây là đoạn văn được gạch dưới.

Đây là một bảng 2 chiều 3x3 :

phần tử 1,1	phần tử 1,2	phần tử 1,3
phần tử 2,1	phần tử 2,2	phần tử 2,3
phần tử 3,1	phần tử 3,2	phần tử 3,3

Đây là một danh sách đánh số thứ tự :

1. Mục 1
2. Mục 2

Đây là một danh sách dùng bullet :

- [Liên kết đến trang tradthoai.asp](#)
- [Liên kết đến trang isapiDBDTHOAI.HTM](#)

Done Local intranet



Source code của trang Web cơ bản

Source code HTML của trang Web cơ bản :

```
<HTML><HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=x-user-defined">
<TITLE>Trang Web demo các phần tử cơ bản</TITLE>
</HEAD>
<BODY>
<FONT face=VnTimes size=4>
<H1 align=center>Đây là tiêu đề lớn cấp 1</H1>

<BR>Đây là đoạn văn chữ thường, <EM>còn đây là đoạn văn chữ nghiêng</EM>,
<STRONG>và đây là đoạn chữ in đậm, </STRONG>
<U>đây là đoạn chữ gạch dưới</U>.
<BR>Đây là 1 bảng dữ liệu 2 chiều 3x3 :
<TABLE cellSpacing=1 cellPadding=1 width="75%" border=1>
<TR>
<TD>phần tử 1,1</TD>
<TD>phần tử 1,2</TD>
<TD>phần tử 1,3</TD></TR>
```



Source code của trang Web cơ bản (tt)

```
<TR><TD>phần tử 2,1</TD> <TD>phần tử 2,2</TD> <TD>phần tử 2,3</TD></TR>
<TR>
  <TD>phần tử 3,1</TD>
  <TD>phần tử 3,2</TD>
  <TD>phần tử 3,3</TD></TR>
</TABLE>
```

Đây là danh sách có đánh số thứ tự :

```
<OL>
  <LI>MÖc 1
  <LI>MÖc 2
</OL>
```

Đây là danh sách dùng bullet :

```
<UL>
  <LI><A href="tradthoai.asp">Liên kết đến trang tradthoai.asp </A>
  <LI><A href="isapiDBDTHOAI.HTM">Liên kết đến trang isapidbdthoai.htm </A>
</UL> </FONT>
</BODY></HTML>
```



4.3 Tổng quát về lập trình Web

Các tag HTML cơ bản chưa thể biến trang Web thành phần mềm tương tác trực quan với người dùng như các ứng dụng truyền thống trên Windows ⇒ cần 1 số tag lệnh mới để biến trang Web thành ứng dụng. Trang Web nằm ở Web server, nhưng khi cần nó sẽ được download về phía client ⇒ Thành phần phần mềm được nhúng vào trang Web có thể được xử lý trước ở Server hay sau khi đã được download về phía client ⇒ Có 2 công nghệ lập trình Web khác nhau :

- lập trình Web chạy ở client và lập trình web chạy ở server.

<OBJECT> </OBJECT>	tạo 1 phần tử COM, ActiveX Control,...
<APPLET code=URL ...>	tạo 1 Applet Java
<SCRIPT> </SCRIPT>	tạo 1 đoạn chương trình (thường là n hàm xử lý)
<FORM></FORM>	tạo 1 form giao diện người dùng và đặc tả hàm xử lý form (thường nằm ở máy server)
.....	



4.4 Lập trình Web chạy ở máy client

Hiện có 3 công nghệ lập trình Web chạy ở phía client khác nhau :

- **DHTML và script** : mỗi tag <script> cho phép chèn nhiều hàm xử lý vào trang Web, các hàm này được viết bằng ngôn ngữ script như javascript, VBscript. DHTML nói rộng nhiều tham số vào từng tag lệnh HTML, 1 trong các tham số nói rộng là định nghĩa hàm nào sẽ xử lý sự kiện nào trên phần tử tương ứng.
- **Java Applet** : mỗi tag <Applet> cho phép nhúng vào trang Web 1 đối tượng java đặc biệt mà ta gọi là Applet. Applet là 1 form chương trình viết bằng Java, nó có giao diện nào đó tùy theo chức năng xử lý. Khi được nhúng vào trang Web, nó sẽ chiếm không gian với vị trí và kích thước được qui định bởi tag <Applet> và sẽ tương tác với người dùng trực tiếp.
- **Microsoft object** : tag <Object> cho phép nhúng vào trang Web 1 đối tượng COM hay ActiveX. Cũng giống như Applet, ActiveX là 1 form chương trình viết bằng công nghệ Microsoft (VC++, VB,...), nó có giao diện nào đó tùy theo chức năng xử lý. Khi được nhúng vào trang Web, nó sẽ chiếm không gian với vị trí và kích thước được qui định bởi tag <Object> và sẽ tương tác với người dùng trực tiếp.



4.4 Lập trình Web chạy ở máy client

Các công nghệ lập trình Web chạy ở phía client khác nhau thường có các khuyết điểm sau :

- **không tin cậy** : phụ thuộc vào cấu hình máy client vốn rất đa dạng về công suất tính toán, tình trạng nhiễm virus...
- **không hiệu quả** : tổn thời gian download phần tử phần mềm từ server về client nhất là phần tử phần mềm có lượng dữ liệu lớn. Nếu dữ liệu xử lý nằm ở server thì phần mềm phải tốn nhiều thời gian truy xuất dữ liệu thông qua mạng.
- **khó bảo mật** : vì thành phần phần mềm phải được download về máy client nên client có thể tìm hiểu, nghiên cứu được thuật toán của phần mềm, nhất là nó được viết bằng ngôn ngữ script ở dạng rõ (mã nguồn).

Với các đặc điểm của lập trình Web chạy ở phía client, người ta thường chỉ dùng nó để xử lý sơ bộ quá trình tương tác và nhập liệu của người dùng, còn để xử lý các vấn đề còn lại, người ta dùng các công nghệ lập trình Web ở Server.



4.5 Lập trình Web chạy ở máy Server

Hiện có rất nhiều công nghệ lập trình Web chạy ở phía server khác nhau :

- **CGI** : là 1 phần mềm xử lý hàng lệnh truyền thống cho phép chuyển hướng (redirection) thiết bị nhập/xuất chuẩn. Mỗi lần Web server nhận yêu cầu từ client, nó chuẩn bị dữ liệu trong 1 file nhập, thiết lập file nhập và file xuất thành 2 thiết bị I/O chuẩn của module CGI rồi kích hoạt CGI chạy. Trong quá trình chạy, CGI nhập dữ liệu từ file nhập và xuất dữ liệu ra file xuất do Server thiết lập. Sau khi CGI chạy xong, Web server sẽ lấy dữ liệu từ file xuất và gửi về client.
- **ISAPI Extension** : mỗi thành phần xử lý được đóng gói thành 1 hàm và được đính kèm trong thư viện *.dll. Mỗi lần Web server cần thành phần nào (do client yêu cầu), nó sẽ liên kết động với hàm đó và gọi hàm cục bộ ⇒ hệ thống không tốn chi phí quản lý process như CGI.
- **ASP, JSP, PHP,...** : thành phần phần mềm là những đoạn script được nhúng thẳng vào trang Web nhưng sẽ được server chạy trước, kết quả thường được thay thế vào vị trí của thành phần phần mềm trước khi gửi về client.



Lập trình Web chạy ở máy Server

Các công nghệ ASP, JSP, PHP,... dùng ngôn ngữ script để miêu tả chương trình ⇒ không đủ mạnh để giải quyết những vấn đề xử lý phức tạp ⇒ thường nhờ sự trợ giúp của các thành phần chuyên dụng khác :

- **ASP** : dùng COM và *.dll.
- **JSP** : dùng Servlet Java và JavaBean.
- **PHP** : dùng các thư viện hàm API viết sẵn.

Tóm lại, thông qua dịch vụ Web, từ client, ta có thể kích hoạt chạy 1 ứng dụng hay 1 hàm trong thư viện nào đó ở máy server.



Môn học : Lập trình mạng

Chương 5

DHTML & CLIENT SCRIPT

- 5.1 Các tính chất chính của DHTML
- 5.2 Các sự kiện có thể kết hợp hàm xử lý
- 5.3 Dynamic Style
- 5.4 Dynamic font
- 5.5 Data Binding
- 5.6 Thí dụ về viết script : trò chơi dò mìn đơn giản



5.1 CÁC TÍNH CHẤT CHÍNH CỦA DHTML

DHTML (Dynamic HTML) là sự nới rộng từ HTML do Microsoft khởi xướng và hiện mới chỉ được hiện thực bởi trình duyệt IE của Microsoft, các trình duyệt khác thường không hiểu các tính chất DHTML. Ý tưởng của DHTML gồm 4 lĩnh vực chính :

1. kết hợp từ 0 đến n hàm xử lý sự kiện với từng phần tử **hầu** cho phép người dùng tương tác với nó. Các hàm xử lý sự kiện được đặc tả bằng 1 ngôn ngữ script nào đó và được đặt trong các tag `<script>...</script>`. Hiện 2 ngôn ngữ script phổ dụng nhất là javascript và VBscript, trong đó javascript phổ dụng hơn vì được hỗ trợ bởi hầu hết các trình duyệt Web, trong khi VBscript thì hầu như chỉ có IE hỗ trợ. Lập trình script chạy ở phía client chủ yếu là viết các hàm xử lý sự kiện cho các tag lệnh trong trang Web.
2. kết hợp 1 Style với từng phần tử (tag HTML). Style là tập các thuộc tính miêu tả cách thức hiển thị phần tử. Style của từng phần tử được định nghĩa tại thời điểm xây dựng trang Web nhưng có thể được thay đổi động trong quá trình tương tác với người dùng.



CÁC TÍNH CHẤT CHÍNH CỦA DHTML

3. Font động. Thường thì khi trang web được download về máy client, trình duyệt Web sẽ thông dịch nó và hiển thị kết quả lên màn hình cho người dùng xem, nếu chuỗi văn bản yêu cầu dùng font xác định, trình duyệt Web sẽ yêu cầu HĐH cung cấp, nếu có font thì không sao, còn nếu không có, HĐH sẽ thay thế bằng font có tính chất gần giống, nhưng thường không đạt yêu cầu. Để khắc phục yếu điểm này, tính chất "dynamic font" cho phép trình duyệt Web download font từ server về để hiển thị đúng theo yêu cầu của trang Web đang xử lý.
4. Liên kết dữ liệu (data binding).

Chi tiết về 4 tính chất trên sẽ được trình bày trong các slide còn lại của chương này.



5.2 Các sự kiện có thể kết hợp hàm xử lý

onafterupdate :	xảy ra sau khi phần tử đã được cập nhật (database)
onbeforeupdate :	trước khi phần tử đã được cập nhật (database)
onclick :	sau khi ấn chuột trên phần tử
ondblclick :	sau khi ấn kép chuột trên phần tử
ondragstart :	khi bắt đầu drag phần tử
onerrorupdate :	khi có sai trong việc cập nhật phần tử
onhelp :	khi có yêu cầu trợ giúp
onkeydown :	khi ấn phím
onkeypress :	khi ấn phím
onkeyup :	khi nhả phím
onmousedown :	khi ấn mouse
onmousemove :	khi dời mouse
onmouseout :	khi dời mouse ra khỏi phần tử
onmouseover :	khi dời mouse tới phần tử
onmouseup :	khi nhả mouse
onreadystatechange :	khi đối tượng thay đổi trạng thái
onrowenter :	khi record hiện hành đã thay đổi nội dung
onrowexit :	khi điều khiển datasource thay đổi nội dung record.
onselectstart :	khi đối tượng được chọn.



5.3.1 Style động : ẩn/hiện đối tượng

Một trong các thuộc tính trong style kết hợp với từng tag lệnh là thuộc tính "display", nó cho phép ẩn/hiện phần tử tương ứng. Thí dụ trong trang Web kế bên, bài thơ được để trong tag `` và ta có thể ẩn/hiện bài thơ bằng cách thay đổi giá trị của thuộc tính `display` của tag `` tương ứng.

Style động : Thuộc tính ẩn/hiện

Mùa Thi

Tho ta ho hó chưa chồng
Ta yêu, muốn cưới, mà không thì giờ

Xuân Diệu

Click chuột vào đây để ẩn bài thơ



Ẩn/hiện đối tượng (mã nguồn trang Web)

```
<html> <head> <script language=javascript>
function HideDisp() {
    if (Baito.style.display == "none") {
        Baito.style.display = ""; cmdHide.value ="Click chuột vào đây để ẩn bài thơ";
    } else {
        Baito.style.display = "none";
        cmdHide.value ="Click chuột vào đây để hiện bài thơ";
    }
}
</script> </head> <body>
<center> <h3>Style động : Thuộc tính ẩn/hiện </h3>
<span id=Baito style="display: "">
<p><B>Mùa Thi</B>
<p>Thơ ta hờ hớ chưa chồng <br>Ta yêu, muốn cưới, mà không thì giờ
<p>Xuân Diệu </span>
<p> <input id=cmdHide type=button value="Click chuột vào đây để ẩn bài thơ"
onclick=HideDisp()
</center> </body> </html>
```



5.3.2 Style động : thay đổi font/co chữ

Để đặc tả font chữ được dùng cho phần tử, ta có thể dùng các thuộc tính liên quan như color, font-style, font-variant, font-weight, font-size, font-family,... Thí dụ trong trang Web phía dưới, khi dời chuột vào/ra chuỗi "Hãy dời chuột vào/ra chuỗi này", thuộc tính fontsize và color kết hợp với chuỗi sẽ được hiệu chỉnh.

Style động : Thay đổi font/co chữ

HÃY DỜI CHUỘT VÀO/RA CHUỖI NÀY

Style động : Thay đổi font/co chữ

HÃY DỜI CHUỘT VÀO/RA CHUỖI NÀY



Thay đổi font/co chữ (mã nguồn trang Web)

```
<html> <head> <script language=javascript>
function zoomin() {
    document.all.myText.style.color = "red";
    document.all.myText.style.fontSize = "40px";
}
function dorestore() {
    document.all.myText.style.color = "black";
    document.all.myText.style.fontSize = "14px";
}
</script> </head>
<body>
<center>
<h3>Style động : Thay đổi font/co chữ </h3>
<h3 id=myText onmouseover=zoomin() onmouseout=dorestore()
style="color:black; font-size:18px" align=center> HÃY DỜI CHUỘT VÀO/RA
CHUỖI NÀY </h3>
</center> </body> </html>
```



5.3.3 Style động : thay đổi nội dung

Ta có thể thay đổi nội dung của tag lệnh bằng cách dùng 1 trong 4 thuộc tính sau đây của nó :

1. innerText : text của nội dung tag lệnh được hiểu theo dạng text thô.
2. innerHTML : text của nội dung tag lệnh được hiểu theo dạng HTML.
3. outerText : text cho toàn tag lệnh được hiểu theo dạng text thô.
4. outerHTML : text cho toàn tag lệnh được hiểu theo dạng HTML.

Thay đổi nội dung động

Hello! Dynamic Content

innerText

Hello! Dynamic Content

innerHTML

Hello! Dynamic Content

outerText

Hello! Dynamic Content

outerHTML



Thay đổi nội dung

Ta cũng có thể thêm động tag lệnh vào vị trí xác định trong trang Web bằng cách dùng tác vụ tagi.insertAdjacentHTML(swhere, sText), trong đó swhere có thể là:

1. beforeBegin : ngay trước tag lệnh liên quan.
2. afterBegin : trước ngay sau khi bắt đầu tag lệnh liên quan.
3. beforeEnd : thêm ngay trước khi kết thúc tag lệnh liên quan.
4. afterEnd : thêm ngay sau đối tượng liên quan.

Thay đổi nội dung động

Goodbye! <I>Dynamic Content</I>

innerText

Goodbye! *Dynamic Content*

innerHTML

Goodbye! <I>Dynamic Content</I>

outerText

Goodbye! *Dynamic Content*

outerHTML

Thay đổi nội dung (mã nguồn trang Web)

```
<html> <head> <script language=javascript>
function ChangeContent3() {
    myH3.insertAdjacentHTML("AfterEnd", "<br>");
    myH3.outerText="Goodbye! <l>Dynamic Content</l>";
}
function ChangeContent4() {
    myH4.insertAdjacentHTML("AfterEnd", "<br>");
    myH4.outerHTML="Goodbye! <l>Dynamic Content</l>";
}
</script> </head> <body> <center> <h3>Thay đổi nội dung động </h3>
<h3 id=myH1>Hello! Dynamic Content</h3>
<input type=button value="innerText"
onclick="document.all.myH1.innerText='Goodbye! <l>Dynamic Content</l>'">
<br> <h3 id=myH2>Hello! Dynamic Content</h3>
<input type=button value="innerHTML"
onclick="document.all.myH2.innerHTML='Goodbye! <l>Dynamic Content</l>'">
<br> <h3 id=myH3>Hello! Dynamic Content</h3>
< input type=button value="outerText" onclick=ChangeContent3()>
<br> <h3 id=myH4>Hello! Dynamic Content</h3>
< input type=button value="outerHTML" onclick=ChangeContent4()>
</center> </body> </html>
```



5.3.4 Style động : thay đổi vị trí

Thuộc tính "position" giúp IE hiển thị phần tử tương ứng ở ví trí tương đối/tuyệt đối :

1. position=relative (mặc định), phần tử sẽ được hiển thị ở ngay sau phần tử đi trước nó.
2. position=absolute, phần tử sẽ được hiển thị ở vị trí được qui định bởi các thuộc tính top, left, z-index bắt chấp nó nằm ở đâu trong mã nguồn HTML của trang Web.

Thuộc tính z-index qui định layer hiển thị phần tử : =0 là layer mặc định, layer chỉ số dương nằm phía trên, layer chỉ số âm nằm phía dưới layer 0.

Thí dụ trang Web ở slide kế sẽ hiển thị con cá chạy từ trái sang phải rồi quay lại sang trái,... Con cá nằm ở layer dưới layer văn bản (mặc định là 0).



Thay đổi vị trí

III. Thay đổi vị trí động



Đây là chuỗi văn bản tham khảo

Thêm văn bản



Thay đổi vị trí (mã nguồn trang Web)

```
<html> <head> <script language=javascript>
function insertText() {
document.all.MyText.insertAdjacentHTML("BeforeBegin", "<p><b><font face='Helvetica'
color=#0000FF size=4>BIỂN CÁ MÊNH MÔNG</font></b></p>");
}
var pleft = 0; var ptop = 60; var pstep = 5;
function ChangePos() {
    pleft = pleft + pstep;
    if (pleft >=640) {
        pstep = -5; document.all.Image1.src = "FishLeft.gif";
    } else if (pleft <=0) {
        pstep = 5; document.all.Image1.src = "FishRght.gif";
    }
    document.all.Image1.style.left = pleft;
}
</script> </head>
<body onload=window.setInterval("ChangePos()",40);>
<h3>Demo việc thay đổi vị trí động </h3>
<p id=MyText>Đây là chuỗi văn bản tham khảo</p>
<input type=button value="Thêm văn bản" onclick=insertText()>

</body> </html>
```



5.3.5 Style động : Filters and Transitions

Ta có thể tạo/thay đổi hiệu ứng hiển thị đặc biệt cho từng phần tử trong trang Web. DHTML cung cấp khá nhiều hiệu ứng filter & transition :

- **Các hiệu ứng transition** : cho phép hiển thị phần tử A rồi từ từ chuyển sang phần tử B. Có 3 hiệu ứng transition là Blend, Reveal & Compositor.
- **Các hiệu ứng filter** : cho phép lọc phần tử theo hàm lọc xác định trước khi hiển thị kết quả. Có 13 hiệu ứng Filter : Alpha, Chroma, Shadow, DropShadow, Invert, Xray, MaskFilter, Blur, Glow, FlipH, FlipV, Light, Wave.

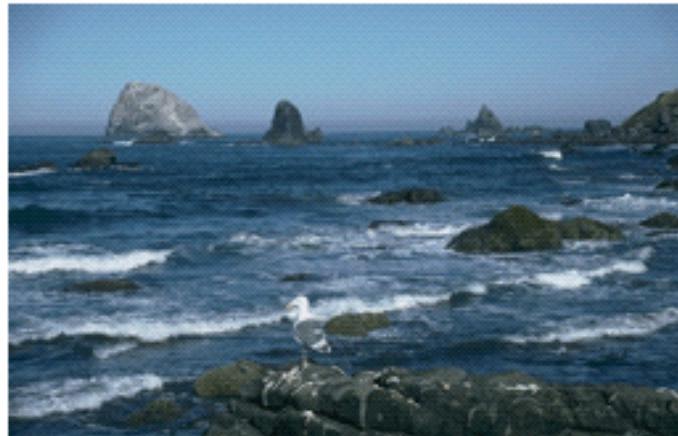
Các slide kế sẽ lần lượt trình bày từng hiệu ứng một.



5.3.5.1 Hiệu ứng Blend

Hiệu ứng Blend cho phép chuyển đổi phần tử A sang phần tử B 1 cách từ từ theo cơ chế "đồng hóa", từng pixel lúc đầu chứa 100% A và 0% B, từ từ tỉ lệ A giảm dần và tỉ lệ B tăng dần đến khi kết thúc thì mỗi pixel chứa 0% A và 100% B. Ảnh dưới cho thấy trạng thái đầu và cuối cùng của hiệu ứng Blend chuyển ảnh bình minh sang hoàng hôn :

Demo hiệu ứng Blend



Tạo hiệu ứng Blend

Demo hiệu ứng Blend



Tạo hiệu ứng Blend

Hiệu ứng Blend (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function BlendIt() {
    BlendImage.filters.blendTrans.Apply();
    BlendImage.src = "sunset.gif";
    BlendImage.filters.blendTrans.Play(10);
}
</script>
</head>

<body>
<h3>Demo hiệu ứng Blend</h3>

<p>
<input type=button value="Tạo hiệu ứng Blend" onclick=BlendIt()>
</body> </html>
```



5.3.5.2 Hiệu ứng Reveal

Hiệu ứng Reveal cho phép chuyển đổi phần tử A sang phần tử B 1 cách từ từ theo cơ chế "xâm thực", lúc đầu hiển thị 100% A và 0% B, sau đó phần tử B xâm chiếm A từng vùng theo quỹ đạo xác định đến khi kết thúc thì chỉ còn phần tử B. Có 23 cách xâm thực khác nhau có thể được xác định trong hiệu ứng reveal :

Transition name	Value	Đặc tả
Box in	0	hộp từ ngoài vào
Box out	1	hộp từ trong ra
Circle in	2	vòng tròn từ ngoài vào
Circle out	3	vòng tròn từ trong ra
Wipe up	4	kéo từ dưới lên
Wipe down	5	kéo từ trên xuống
Wipe right	6	kéo từ trái sang
Wipe left	7	kéo từ phải sang trái
Vertical blinds	8	kéo nhiều màn từ trái sang
Horizontal blinds	9	kéo nhiều màn từ trên xuống



5.3.5.2 Hiệu ứng Reveal

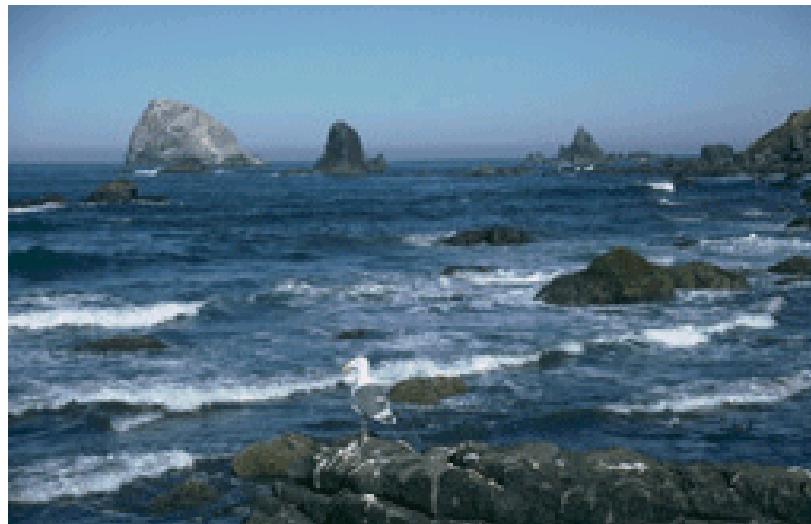
Transition name	Value	Đặc tả
Checkerboard across	10	kéo nhiều ô từ trái sang
Checkerboard down	11	kéo nhiều ô từ trên xuống
Random dissolve	12	thẩm thấu từ từ
Split vertical in	13	kéo 2 bên vào
Split vertical out	14	kéo từ giữa ra 2 bên
Split horizontal in	15	kéo 2 bên trên dưới vào giữa
Split horizontal out	16	kéo từ giữa ra 2 bên trên dưới
Strips left down	17	kéo từ góc trên phải xuống dưới trái
Strips left up	18	kéo từ góc dưới phải lên trên trái
Strips right down	19	kéo từ góc trên trái xuống dưới phải
Strips right up	20	kéo từ góc dưới trái lên trên phải
Random bars horizontal	21	tạo thanh ngang ngẫu nhiên
Random bars vertical	22	tạo thanh dọc ngẫu nhiên
Random	23	ngẫu nhiên 1 trong các cách trên



Thí dụ về hiệu ứng Reveal

Hãy quan sát trang Web dưới đây, mỗi lần người dùng ấn chuột vào button tạo hiệu ứng, IE sẽ thực hiện hiệu ứng Reveal với chỉ số tương ứng giữa 2 ảnh A (bình minh) và B (hoàng hôn).

Demo hiệu ứng Reveal



Tạo hiệu ứng Reveal số 0



Hiệu ứng Reveal (mã nguồn trang Web)

```
<html> <head> <script language=javascript>
var i = 0;
function Reveallt() {
    ReveallImage.filters.revealTrans.transition = i;
    ReveallImage.filters.revealTrans.Apply();
    if (i % 2) ReveallImage.src = "beach.gif";
    else ReveallImage.src = "sunset.gif";
    ReveallImage.filters.revealTrans.Play(5);
    if(++i>22) i=0;
    btn1.value = "Tạo hiệu ứng Reveal số " + i;
}
</script> </head> <body> <center>
<h3>Demo hiệu ứng Reveal</h3>
 <p>
<input id=btn1 type=button value="Tạo hiệu ứng Reveal số 0" onclick=Reveallt()>
</center> </body> </html>
```



5.3.5.3 Hiệu ứng Compositor

Hiệu ứng Compositor cho phép hiển thị 2 đối tượng được định vị tuyệt đối và có phần giao nhau theo 1 hàm xử lý nào đó. Có 18 hàm xử lý khác nhau, thí dụ hàm 25 hiển thị vùng ảnh B có trộn với từng pixel cùng vị trí của ảnh A. Hãy quan sát ảnh dưới đây :

Ảnh A

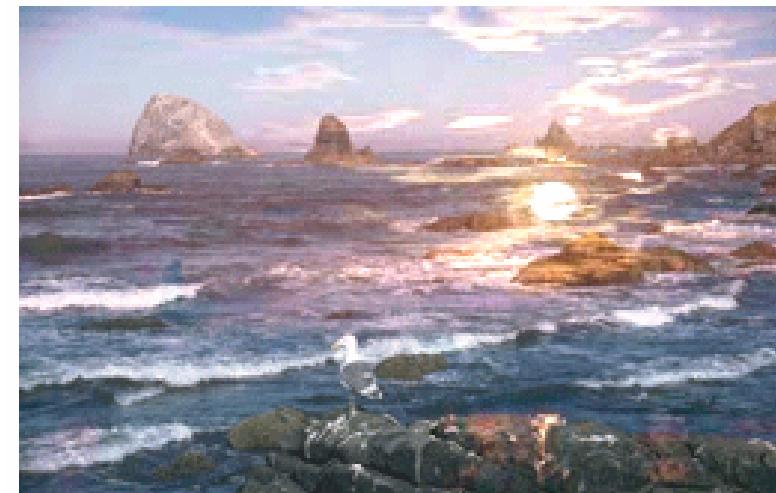


Ảnh B



Kết quả của hàm 25

Tạo hiệu ứng Compositor số 0



Hiệu ứng Compositor (mã nguồn trang Web)

```
<html>
<head>
<script language=javascript>
function OnLoad() {
    div1.filters.item(0).Apply();
    div1.innerHTML = "<img src=sunset.gif style='position:absolute; top:0; left:0;height:165; width:256'>"
    div1.filters.item(0).Play();
}

var i = 0;
function Compolt() {
    div1.filters.item(0).Function = i++;
    if(10<i && i <19) i=19;
    if (i >25) i = 0;
    btn1.value = "Tạo hiệu ứng Compositor số " + i;
}
```



Hiệu ứng Compositor (mã nguồn trang Web)

```
</script>  
</head>
```

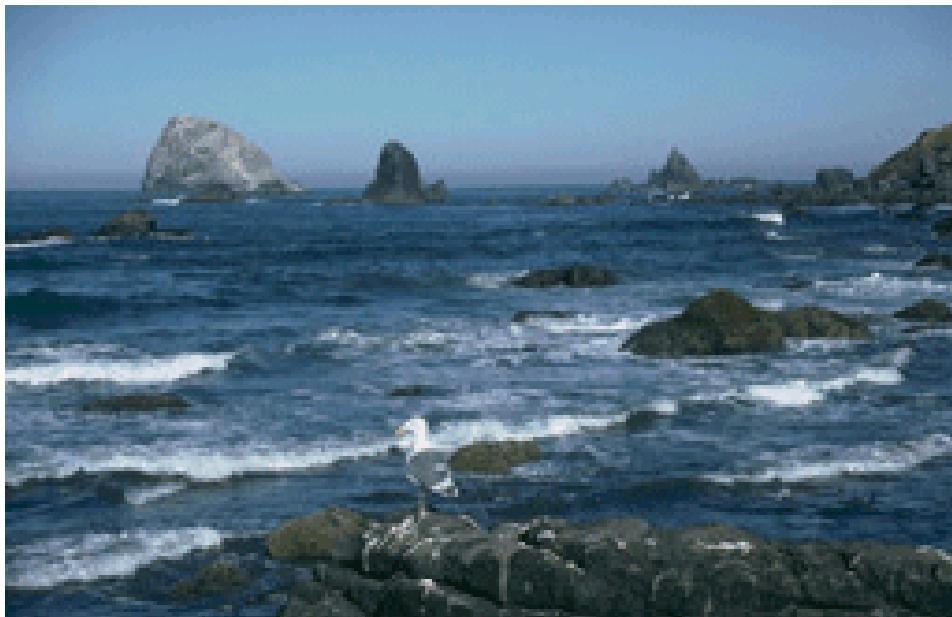
```
<body Onload = OnLoad() >  
<input id =btn1 type=button value="Tạo hiệu ứng Compositor số 0"  
onclick=Compolt()> </p>  
<div id=div1  
style="filter:progid:DXImageTransform.Microsoft.Compositor(function=25,  
duration=5);position:absolute; top:52; left:10; height:200; width:600;">  
  
</div>  
</body>  
</html>
```



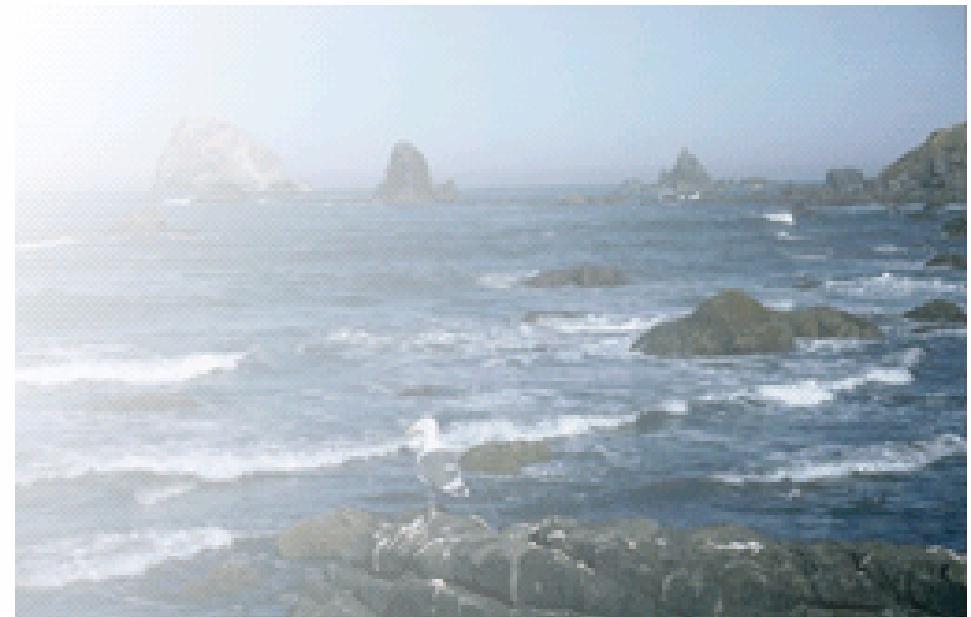
5.3.5.4 Hiệu ứng Alpha

Hiệu ứng Alpha cho phép thay đổi độ mờ (opacity) của từng pixel của đối tượng. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Alpha



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Alpha

Hiệu ứng Alpha (mã nguồn trang Web)

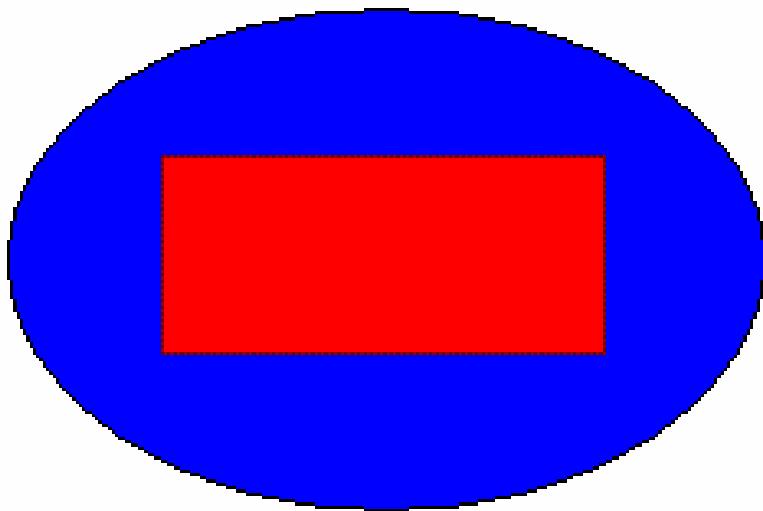
```
<html> <head> <script language=javascript>
function Alphalt() { //lọc tuyến tính từ trên trái xuống dưới phải
    Image1.filters.Alpha.enabled = true; Image1.filters.Alpha.Style = 1;
    Image1.filters.Alpha.Opacity = 0; Image1.filters.Alpha.FinishOpacity = 100;
    Image1.filters.Alpha.StartX = 0; Image1.filters.Alpha.StartY = 0;
    Image1.filters.Alpha.FinishX = 300; Image1.filters.Alpha.FinishY = 200;
}
</script> </head> <body>
<h3 align=center>Demo hiệu ứng Alpha</h3>
<table> <tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Alpha</td></tr>
</table> </body> </html>
```



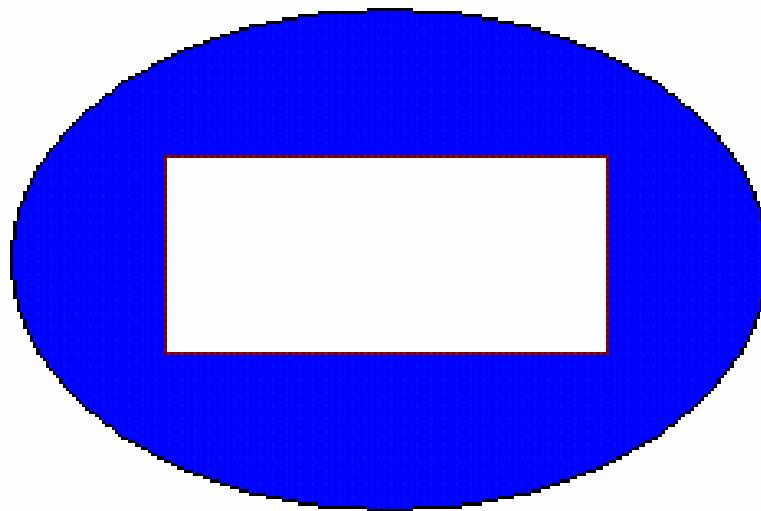
5.3.5.5 Hiệu ứng Chroma

Hiệu ứng Chroma cho phép lọc các vùng có màu xác định của đối tượng.
Hãy quan sát trang Web dưới đây, ảnh bên phải đã bị lọc màu đỏ :

Demo hiệu ứng Chroma



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng chroma lọc bỏ màu đỏ

Hiệu ứng Chroma (mã nguồn trang Web)

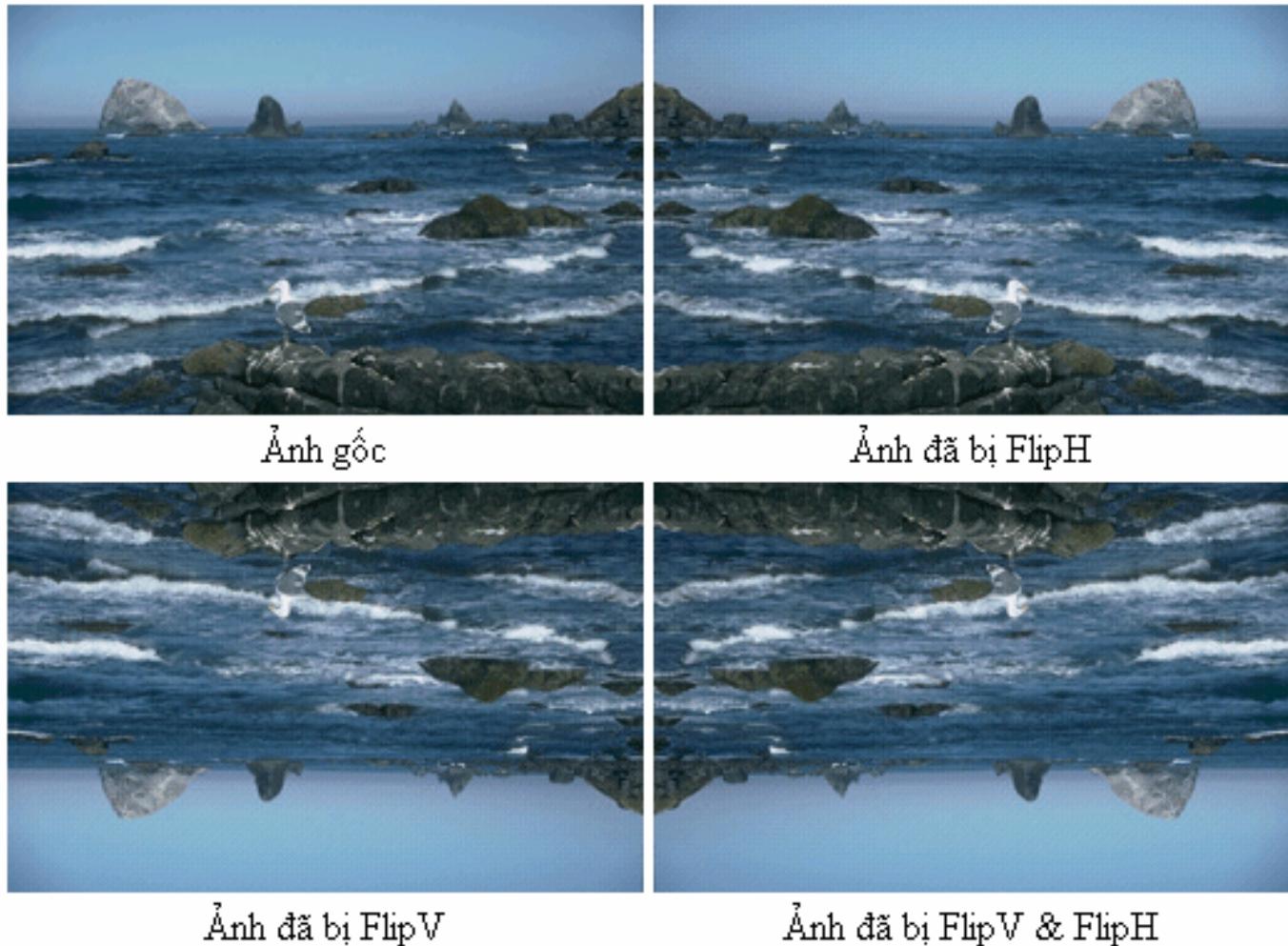
```
<html> <head>
<script language=javascript>
function Chromalt(){
    Image1.filters.chroma.enabled = true; Image1.filters.chroma.color = "#ff0000";
}
</script> </head>
<body>
<h3 align=center>Demo hiệu ứng Chroma</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td align=center>Ảnh đã có hiệu ứng chroma lọc bỏ màu đỏ </td></tr>
</table>
</body> </html>
```



5.3.5.6 Hiệu ứng FlipH & FlipV

Hiệu ứng FlipH cho phép lật ngang đối tượng trước khi hiển thị, còn hiệu ứng FlipV cho phép lật dọc đối tượng. Hãy quan sát trang Web hình bên :

Demo hiệu ứng FlipH, FlipV



Hiệu ứng FlipV, FlipH (mã nguồn trang Web)

```
<html> <head>
</head>

<body>
<h3 align=center>Demo hiệu ứng FlipH, FlipV</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc</td><td align=center>Ảnh đã bị FlipH</td></tr>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh đã bị FlipV</td><td align=center>Ảnh đã bị FlipV &
FlipH</td></tr>
</table>
</body> </html>
```



5.3.5.7 Hiệu ứng Gray

Hiệu ứng Gray cho phép tạo ảnh trắng đen theo n mức grayscale của đối tượng gốc (thường là ảnh màu). Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Gray



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Gray

Hiệu ứng Gray (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function GrayIt(){
    Image1.filters.Gray.enabled = true;
}
</script>
</head>

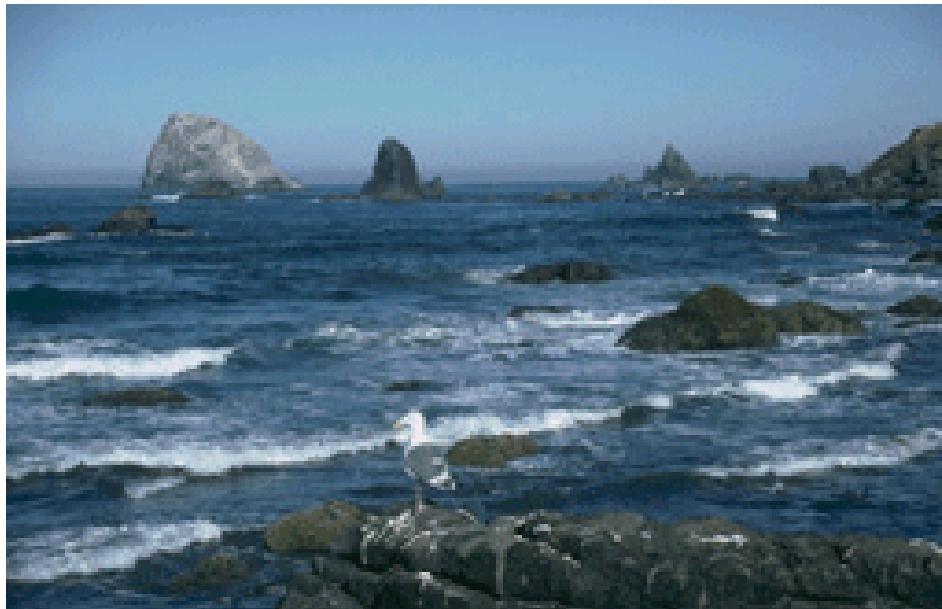
<body>
<h3 align=center>Demo hiệu ứng Gray</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Gray</td></tr>
</table> </body> </html>
```



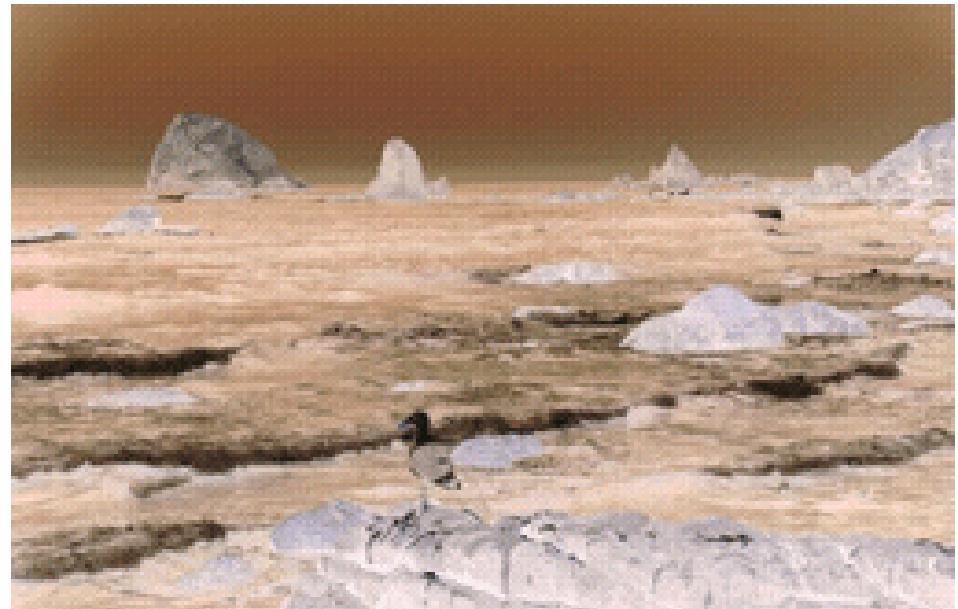
5.3.5.8 Hiệu ứng Invert

Hiệu ứng Invert cho phép tạo âm bản của đối tượng. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Invert



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Invert



Hiệu ứng Invert (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function InvertIt(){
    Image1.filters.Invert.enabled = true;
}
</script> </head>

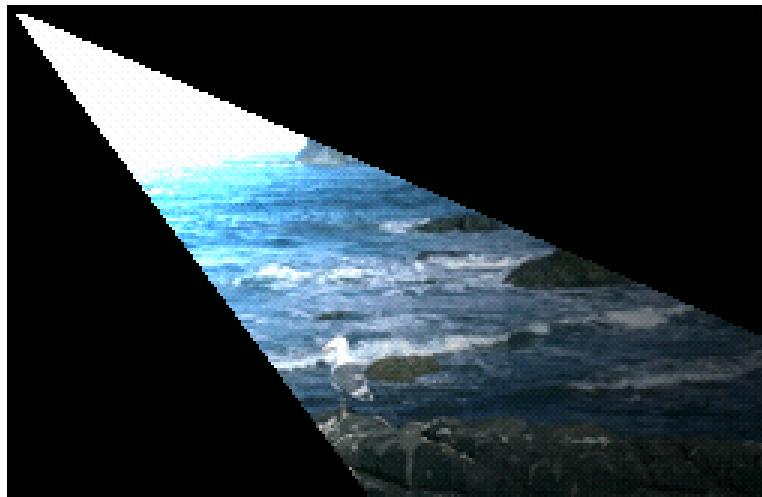
<body>
<h3 align=center>Demo hiệu ứng Invert</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Invert</td></tr>
</table> </body> </html>
```



5.3.5.9 Hiệu ứng Light

Hiệu ứng Light cho phép tạo nguồn sáng xác định để hiển thị đối tượng. DHTML hỗ trợ 3 loại nguồn sáng khác nhau : từ 1 điểm trên bề mặt đối tượng, nguồn sáng // chiếu vào đối tượng, nguồn sáng từ 1 điểm ngoài không gian chiếu vào đối tượng. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Light : tia quét bề mặt ảnh



Tạo hiệu ứng light trên bề mặt ảnh



Hiệu ứng Light (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function Setlight1(){
    Image1.filters.Light.enabled = true;
    Image1.filters.Light.clear();
    Image1.filters.Light.addcone(0,0,0,250,200,255,255,255,30,15);
}
</script>
</head>
<body>
<center>
<h3>Demo hiệu ứng Light : tia quét bề mặt ảnh</h3>

<p>
<input type=button value="Tạo hiệu ứng light trên bề mặt ảnh" onclick=Setlight1()>
</center>
</body> </html>
```



Hiệu ứng Light

Hãy quan sát trang Web dưới đây, nó thể hiện kết quả của việc dùng nguồn sáng là 1 điểm ngoài không gian chiếu thẳng gốc vào đối tượng :

Demo hiệu ứng Light : tia quét từ 1 điểm ngoài chiếu thẳng gốc vào ảnh



Tạo hiệu ứng light từ 1 điểm ngoài chiếu thẳng gốc vào ảnh



Hiệu ứng Light (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function Setlight2() {
    Image2.filters.Light.enabled = true;
    Image2.filters.Light.clear();
    Image2.filters.Light.addPoint(60,60,50,255,255,255,100);
}
</script> </head>
<body>
<center>
<h3>Demo hiệu ứng Light : tia quét từ 1 điểm ngoài chiếu thẳng gốc vào ảnh</h3>

<p>
<input type=button value="Tạo hiệu ứng light từ 1 điểm ngoài chiếu thẳng gốc vào ảnh" onclick=Setlight2()>
</center>
</body> </html>
```



Hiệu ứng Light

Hãy quan sát trang Web dưới đây, nó thể hiện kết quả của việc dùng nguồn sáng song song ngoài gian chiếu thẳng gốc vào đối tượng :

Demo hiệu ứng Light

Tia sáng từ nguồn sáng // ngoài chiếu thẳng gốc vào ảnh



Tạo hiệu ứng light // chiếu thẳng gốc vào ảnh



Hiệu ứng Light (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function Setlight3() {
    Image3.filters.Light.enabled = true;
    Image3.filters.Light.clear();
    Image3.filters.Light.AddAmbient(255,0,255,100);
}
</script> </head>
<body>
<center>
<h3>Demo hiệu ứng Light</h3>
<h3>Tia sáng từ nguồn sáng // ngoài chiếu thẳng gốc vào ảnh</h3>

<p>
<input type=button value="Tạo hiệu ứng light // chiếu thẳng gốc vào ảnh"
onclick=Setlight3()
</center> </body> </html>
```



5.3.5.10 Hiệu ứng MaskFilter

Hiệu ứng MaksFilter cho phép tạo "mặt nạ" của đối tượng. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng MaskFilter



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng MaskFilter

Hiệu ứng MaskFilter (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function WaveIt() {
    Image1.filters.item(0).enabled = true; Image1.filters.item(0).color = "#0000ff";
}
</script>
</head>
<body>
<h3 align=center>Demo hiệu ứng MaskFilter</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng MaskFilter</td></tr>
</table> </body> </html>
```



5.3.5.11 Hiệu ứng Wave

Hiệu ứng Wave cho phép biến đổi đối tượng theo hàm sin. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Wave



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Wave

Hiệu ứng Wave (mã nguồn trang Web)

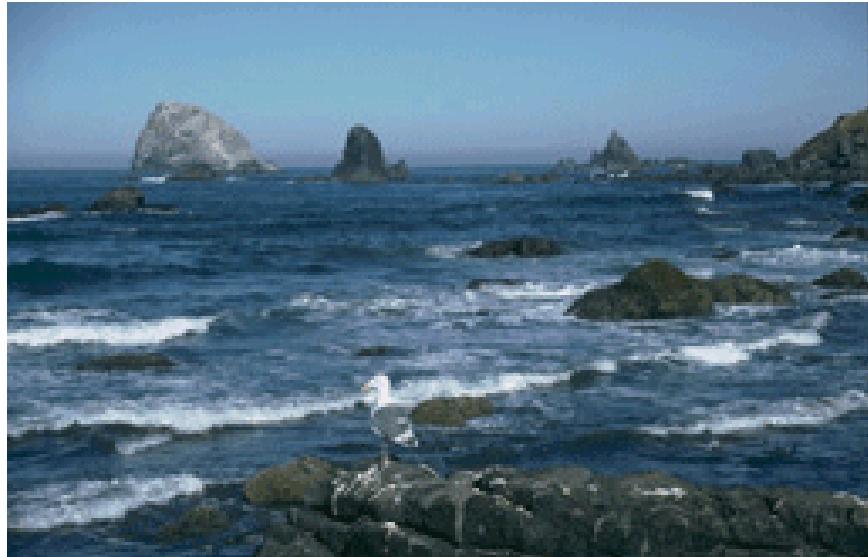
```
<html> <head> <script language=javascript>
function WaveIt(){
    Image1.filters.item(0).enabled = true; Image1.filters.item(0).add = "false";
    Image1.filters.item(0).freq =2; Image1.filters.item(0).phase =90;
    Image1.filters.item(0).strength=15; Image1.filters.item(0).lightstrength =40;
}
</script> </head> <body>
<h3 align=center>Demo hiệu ứng Wave</h3>
<table> <tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Wave</td></tr>
</table> </body> </html>
```



5.3.5.12 Hiệu ứng X-ray

Hiệu ứng X-ray cho phép tạo kết quả chụp X-quang của đối tượng. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng X-ray



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng X-ray

Hiệu ứng X-ray (mã nguồn trang Web)

```
<html> <head>
<script language=javascript>
function XrayIt(){
    Image1.filters.xray.enabled = true;
}
</script>
</head>

<body>
<h3 align=center>Demo hiệu ứng X-ray</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng X-ray</td></tr>
</table> </body> </html>
```



Các hiệu ứng cần đối tượng có nền trong suốt

Các hiệu ứng Shadow, DropShadow, Glow, Blur, Compositor mà chúng ta sẽ trình bày trong các slide kế tiếp cho kết quả đúng khi đối tượng bị tác động có nền trong suốt (transparent background). Một trong các dạng file ảnh cho phép khai báo nền trong suốt là GIF89a, bạn có thể dùng các phần mềm xử lý ảnh như PhotoShop, ImageReady, Photo Elements, CorelDraw,... để tạo ảnh có nền trong suốt. Thí dụ ảnh sau có nền trong suốt được soạn bởi PhotoShop.



5.3.5.13 Hiệu ứng Shadow

Hiệu ứng Shadow cho phép tạo vệt chiếu của đối tượng, kết quả của việc dùng nguồn sáng song song chiếu vào đối tượng theo 1 góc tới và cường độ chiếu sáng xác định. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Shadow



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Shadow

Hiệu ứng Shadow (mã nguồn trang Web)

```
<html> <head>
<SCRIPT language=javascript>
function ShadowIt(){
    Image1.filters.Shadow.enabled = true; Image1.filters.Shadow.color = "#ff0000";
    Image1.filters.Shadow.direction =45; Image1.filters.Shadow.strength =15;
}
</SCRIPT> </head>
<body>
<h3 align=center>Demo hiệu ứng Shadow</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Shadow</td></tr>
</table> </body> </html>
```



5.3.5.14 Hiệu ứng DropShadow

Hiệu ứng Shadow cho phép tạo bóng của đối tượng, kết quả của việc dùng nguồn sáng song song chiếu vào đối tượng với 1 độ dời (offsetX, offsetY). Hãy quan sát trang Web dưới đây :

Demo hiệu ứng DropShadow



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng DropShadow

Hiệu ứng DropShadow (mã nguồn trang Web)

```
<html> <head>
<SCRIPT language=javascript>
function DropShadowIt(){
    Image1.filters.DropShadow.enabled = true; Image1.filters.DropShadow.color =
"#ff0000";
    Image1.filters.DropShadow.offX = 10; Image1.filters.DropShadow.offY = -10;
}
</SCRIPT> </head>
<body>
<h3 align=center>Demo hiệu ứng DropShadow</h3>
<table>
<tr><td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng DropShadow</td></tr>
</table> </body> </html>
```



5.3.5.15 Hiệu ứng Blur IE4.0

Hiệu ứng Blur (IE 4.0) cho phép tạo hiệu ứng đối tượng chạy theo 1 góc tới và cường độ xác định. Hãy quan sát trang Web dưới đây :

Demo hiệu ứng Blur IE 4.0



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Blur IE 4.0

Hiệu ứng Blur IE4.0 (mã nguồn trang Web)

```
<html> <head>
<SCRIPT language=javascript>
function BlurIt(){
    Image1.filters.Blur.enabled = true;
    Image1.filters.Blur.direction = 45; Image1.filters.Blur.strength = 15;
}
</SCRIPT> </head>
<body>
<h3 align=center>Demo hiệu ứng Blur IE 4.0</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Blur IE 4.0</td></tr>
</table> </body> </html>
```



5.3.5.16 Hiệu ứng Blur IE 5.5

Hiệu ứng Blur IE 5.5 cho phép làm nhòe đối tượng. Hãy quan sát ảnh dưới đây :

Demo hiệu ứng Blur của IE 5.5



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Blur IE 5.5

Hiệu ứng Blur IE 5.5 (mã nguồn trang Web)

```
<html> <head>
<SCRIPT language=javascript>
function BlurIt(){
    Image1.filters.item(0).enabled = true; Image1.filters.item(0).PixelRadius=15;
    Image1.filters.item(0).MakeShadow=false;
}
</SCRIPT> </head>
<body>
<h3 align=center>Demo hiệu ứng Blur của IE 5.5 </h3>
<table> <tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Blur IE 5.5</td></tr>
</table> </body> </html>
```



5.3.5.17 Hiệu ứng Glow

Hiệu ứng Glow cho phép làm nhòe xung quanh đối tượng. Hãy quan sát ảnh dưới đây :

Demo hiệu ứng Glow



Ảnh gốc (ấn chuột để tạo hiệu ứng động)



Ảnh đã có hiệu ứng Glow

Hiệu ứng Glow (mã nguồn trang Web)

```
<html> <head> <script language=javascript>
function GlowIt() {
    Image1.filters.Glow.enabled = true;
    Image1.filters.Glow.color = "#ff0000";
    Image1.filters.Glow.strength = 10;
}
</script> </head>
<body>
<h3 align=center>Demo hiệu ứng Glow</h3>
<table>
<tr><td></td>
<td></td></tr>
<tr><td align=center>Ảnh gốc (ấn chuột để tạo hiệu ứng động)</td><td
align=center>Ảnh đã có hiệu ứng Glow</td></tr>
</table> </body> </html>
```



5.4 Dynamic font

Nếu trang Web nào muốn sử dụng font động (được download động từ server về chứ không dùng font trên máy client), người xây dựng trang web cần dùng tag "@font-face" để định nghĩa từng font động trong trang Web đó. Thí dụ về việc định nghĩa font động :

```
@font-face {  
    font-family: Helvetica;  
    font-style: normal;  
    font-weight: normal;  
    src: url ("Fonts/Helv.eot");  
}
```

Để giúp tạo các trang Web dùng font động được dễ dàng, Microsoft cung cấp tiện ích WEFT cho phép chuyển Website (gồm nhiều trang Web theo cấu trúc cây phân cấp) từ dạng bình thường thành dạng sử dụng font động, nhờ đó người xây dựng Website không cần phải định nghĩa thủ công từng tag "@font-face".



5.5 Data Binding

Tính chất này cho phép trang Web xử lý database mà không cần viết code, ta chỉ cần đặc tả các tag HTML và các đối số của chúng. Qui trình cụ thể là :

1. đặc tả đối tượng DSO (Data Source Object) miêu tả database cần truy xuất. Microsoft hỗ trợ 3 đối tượng DSO : TDC (Tabular Data Control), RDS (Remote Data Service) và MSHTML DSO. Thí dụ sau là tag HTML đặc tả đối tượng TDC truy xuất các record dữ liệu được lưu trên file danhbadienthoai.csv. File *.csv thường được tạo bởi chức năng Export của Excel, Access,...

```
<OBJECT classid="clsid:333C7BC4-460F-11D0-BC04-0080C7055A83"  
ID=tdcComposer HEIGHT=0 WIDTH=0>  
    <PARAM NAME="DataURL" VALUE="danhbadienthoai.csv">  
    <PARAM NAME="UseHeader" VALUE="True">  
</OBJECT>
```



Data Binding

2. đặc tả các tag HTML sử dụng dữ liệu của đối tượng DSO và hiển thị nó, ta gọi các tag HTML này là các đối tượng "Data Consumer". Có 2 loại tag Data consumer : xử lý dữ liệu đơn giản như tag INPUT & xử lý dữ liệu bảng như tag TABLE. Thí dụ tag TABLE sau cho phép hiển thị các record được chứa trong đối tượng DSO của slide trước :

```
<table datasrc=#tdcComposer border=1>
<thead><tr>
  <th>Tên thuê bao</th><th>Số điện thoại</th><th>Địa chỉ</th></tr>
</thead>
<tbody> <tr>
  <td><div dataId="Tenthuebao"></div></td>
  <td><div dataId="Sodienthoai"></div></td>
  <td><div dataId="Diachi"></div></td>
</tr></tbody>
</table>
```



Data Binding (mã nguồn trang Web)

```
<html><head><body> <div style="display:none">
<object classid="clsid:333C7BC4-460F-11D0-BC04-0080C7055A83"
ID=tdcComposer HEIGHT=0 WIDTH=0>
    <PARAM NAME="DataURL" VALUE="danhbadienthoai.csv">
    <PARAM NAME="UseHeader" VALUE="True">
</object> </div>
<h3>Demo Data Binding : Hiển thị bảng dữ liệu từ file data có sẵn</h3>
<table datasrc=#tdcComposer border=1> <thead>
<tr><th>Tên thuê bao</th><th>Số điện thoại</th><th>Địa chỉ</th></tr>
</thead>
<tbody> <tr>
    <td><div dataId="Tenthuebao"></div></td>
    <td><div dataId="Sodienthoai"></div></td>
    <td><div dataId="Diachi"></div></td>
</tr></tbody>
</table> </body> </html>
```



5.6 Thí dụ về Client-script

- Chúng ta hãy xây dựng 1 Website phục vụ trò chơi dò mìn đơn giản có trang homepage như sau (trạng thái đang chơi) :

Trang web demo tro choi do min

Nguoi viet : Nguyen Van Hiep

A 10x10 grid puzzle with colored numbers. The grid has a dotted border and contains the following values:

1	1	1							
1		1							
1	2	3	3	3	2	1			
	2							1	
	2							3	1
1	2	z	4		2				
	1	1	1					1	

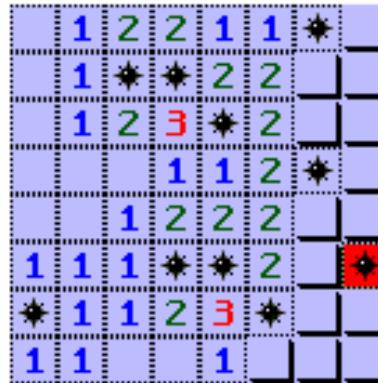
Hay do tung o min

Thí dụ về Client-script

- Trang homepage ở trạng thái thua như sau :

Trang web demo tro choi do min

Nguoi viet : Nguyen Van Hiep



Choi lai van khac

Ba da chet

Trò chơi dò mìn

Phần tử chính của trang web dò mìn là ma trận mìn, thí dụ ta chọn kích thước cố định là 8*8. Để thể hiện ma trận mìn, ta dùng 64 tag , chia làm 8 nhóm, mỗi nhóm 8 tag miêu tả 1 hàng mìn được kết thúc bởi tag
 để xuống hàng. Mỗi tag thể hiện 1 ô mìn, mỗi ô mìn có các trạng thái sau :

- chưa đap (thể hiện bằng ảnh wm_nul.gif)
- đã đap (thể hiện bằng 1 trong 11 ảnh bitmap : 0-8 mìn xung quanh nó, mìn nổ, mìn chưa kịp nổ).

Dữ liệu chính của trò chơi gồm 2 biến array :

- status[8][8], phần tử status[i][j] chứa giá trị luận lý miêu tả cell tương ứng đã bị đap chưa.
- min[8][8], phần tử min[i][j] chứa giá trị luận lý miêu tả cell tương ứng có mìn hay không.



Trò chơi dò mìn

Về việc xử lý sự kiện, ta có 2 phương án khác nhau :

- kết hợp toàn trang web với hàm xử lý sự kiện onclick, hàm này phải kiểm tra vị trí ấn chuột có nằm trong bàn mìn không, nếu có thì xác định vị trí i,j rồi kiểm tra có mìn không để đáp ứng phù hợp.
- kết hợp với từng cell mìn 1 hàm xử lý sự kiện, hàm này xử lý ngay vị trí i,j để đáp ứng phù hợp theo trạng thái của vị trí đó.

Các hàm chức năng :

- domin(i,j) kiểm tra việc đạp vào vị trí i,j.
- doquanh(i,j) đi dùm người dùng các cell quanh cell (i,j) vì biết chắc không có mìn.
- count(i,j) đếm số mìn xung quanh vị trí (i,j).



Trò chơi dò mìn

- Ta có thể dùng trình soạn thảo bitmap như Paint để soạn thảo từng hình theo yêu cầu. Lưu ý tất cả hình bitmap đều có cùng kích thước (thí dụ 17*17).
- Tạo thư mục miêu tả website, copy tất cả các file hình vừa soạn vào thư mục này, rồi dùng trình soạn thảo văn bản bất kỳ để soạn nội dung trang web trò chơi như các slide kế.
- Lưu ý nội dung chính của trang Web gồm 2 phần : phần HTML miêu tả ma trận mìn dùng 64 tag `` miêu tả ma trận 8*8 ô, phần code java script miêu tả các hàm xử lý ấn chuột và các hàm dịch vụ kèm theo



Trò chơi dò mìn

```
<html>
<head>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<title></title>
<script LANGUAGE=javascript>
// Các biến toàn cục
var odado=0;
var dangchoi=0;
var data_tbl;
var stat_tbl;

// Khởi tạo mảng
function MakeArray(size) {
    this.length = size;
    return this;
}
```



Trò chơi dò mìn

```
// Khoi dong tro choi
function wm_init(){
var i,j;
odado = 0;
dangchoi = 1;
// Hien thi ban min trong
for (i =0; i<64; i++)
    document.images[i].src = "wm_nul.gif";
// An button choi tiep
document.all.Mesg.innerText = "Hay do tung o min";
document.all.IDSTART.style.display = "none";
// Tao cac bang va khai dong trang thai
data_tbl= new MakeArray(64);
stat_tbl= new MakeArray(64);
for (i=0; i<64; i++) {
    data_tbl[i] = stat_tbl[i] = 0;
}
```



Thí dụ về Client-script

```
// Xep ngau nhien 10 trai min  
i = 0;  
while (i <10) {  
    j = Math.round(Math.random()*62);  
    if (data_tbl[j]==0) { data_tbl[j]=1; i++; }  
}  
}  
// Tinh so min xung quanh vi tri (h,c)  
function winecnt (h, c) {  
var cnt =0;  
if (h-1>=0 && c-1>=0 && data_tbl[(h-1)*8+(c-1)]) cnt++;  
if (h-1>=0 && data_tbl[(h-1)*8+c]) cnt++;  
if (h-1>=0 && c+1<8 && data_tbl[(h-1)*8+(c+1)]) cnt++;  
if (c-1>=0 && data_tbl[h*8+(c-1)]) cnt++;  
if (c+1<8 && data_tbl[h*8+(c+1)]) cnt++;  
if (h+1<8 && c-1>=0 && data_tbl[(h+1)*8+(c-1)]) cnt++;  
if (h+1<8 && data_tbl[(h+1)*8+c]) cnt++;  
if (h+1<8 && c+1<8 && data_tbl[(h+1)*8+(c+1)]) cnt++;  
return cnt;  
}
```



Thí dụ về Client-script

```
// Do tu dong cac o bao quanh o h,c
function doquanh(h,c) {
    if (h-1>=0 && c-1>=0 && stat_tbl[(h-1)*8+(c-1)]==0) domin(h-1,c-1);
    if (h-1>=0 && stat_tbl[(h-1)*8+c]==0) domin(h-1,c);
    if (h-1>=0 && c+1<8 && stat_tbl[(h-1)*8+(c+1)]==0) domin(h-1,c+1);
    if (c-1>=0 && stat_tbl[h*8+(c-1)]==0) domin(h,c-1);
    if (c+1<8 && stat_tbl[h*8+(c+1)]==0) domin(h,c+1);
    if (h+1<8 && c-1>=0 && stat_tbl[(h+1)*8+(c-1)]==0) domin(h+1,c-1);
    if (h+1<8 && stat_tbl[(h+1)*8+c]==0) domin(h+1,c);
    if (h+1<8 && c+1<8 && stat_tbl[(h+1)*8+(c+1)]==0) domin(h+1,c+1);
}

// Do min o vi tri h,c
function domin(h,c){
var i,cnt;
    i = h*8+c;
    if (stat_tbl[i]) return 0; //đã đập rồi ô này
```



Thí dụ về Client-script

```
if (data_tbl[i] ==1) { // Dap trung min
    for (h=0;h<64;h++){
        if (data_tbl[h]==1)
            document.images[h].src = "wm_mini.gif";
    }
    document.images[i].src = "wm_mina.gif";
    document.all.Mesg.innerText = "Ba da chet";
    document.all.IDSTART.style.display = "";
    dangchoi = 0;
} else { // do trung cho khong co min
    cnt = winecnt(h,c);
    stat_tbl[i] = 1;
    if (cnt ==0) {
        document.images[i].src = "wm_nul1.gif";
        doquanh(h,c);
    }
    if (cnt ==1) document.images[i].src = "wm_1.gif";
    if (cnt ==2) document.images[i].src = "wm_2.gif";
    if (cnt ==3) document.images[i].src = "wm_3.gif";
```



Thí dụ về Client-script

```
if (cnt ==4) document.images[i].src = "wm_4.gif";
if (cnt ==5) document.images[i].src = "wm_5.gif";
if (cnt ==6) document.images[i].src = "wm_6.gif";
if (cnt ==7) document.images[i].src = "wm_7.gif";
if (cnt ==8) document.images[i].src = "wm_8.gif";
odado++;
}
if (odado ==54) {
    document.all.Mesg.innerText = "Ba da thang";
    document.all.IDSTART.style.display = "";
    dangchoi = 0;
}
}
```



Thí dụ về Client-script

```
// Xu ly khi an mouse
function document_onmousedown() {
var i,h;
if (dangchoi == 0) return 0;
i = window.event.srcElement.sourceIndex-12;
if (i<0) return 0;
c = i%9;
h = (i-c) /9;
if (h >=8 || c == 8) return 0;
domin(h,c);
}
```

```
//Choi lai khi an button nay
function btnStart_onclick() {
wm_init();
}
</script>
</head>
```



Thí dụ về Client-script

Thí dụ về Client-script

- Sau khi soạn xong trang Web, cất nó lên file WinMine.html trong thư mục Website.
- Chạy IE và thử truy xuất trang Web dò mìn. Lưu ý bạn có thể truy xuất Webiste theo cấu hình cục bộ (file:///) hay thông qua Web Server (http://....).



Môn học : Lập trình mạng

Chương 6

LẬP TRÌNH WEB CHẠY TRÊN CLIENT DÙNG APPLET JAVA

6.1 Giới thiệu Applet Java

6.2 Qui trình xây dựng Applet Java dùng JBuilder

6.3 Qui trình xây dựng Website dùng Apllet Java bằng InterDev



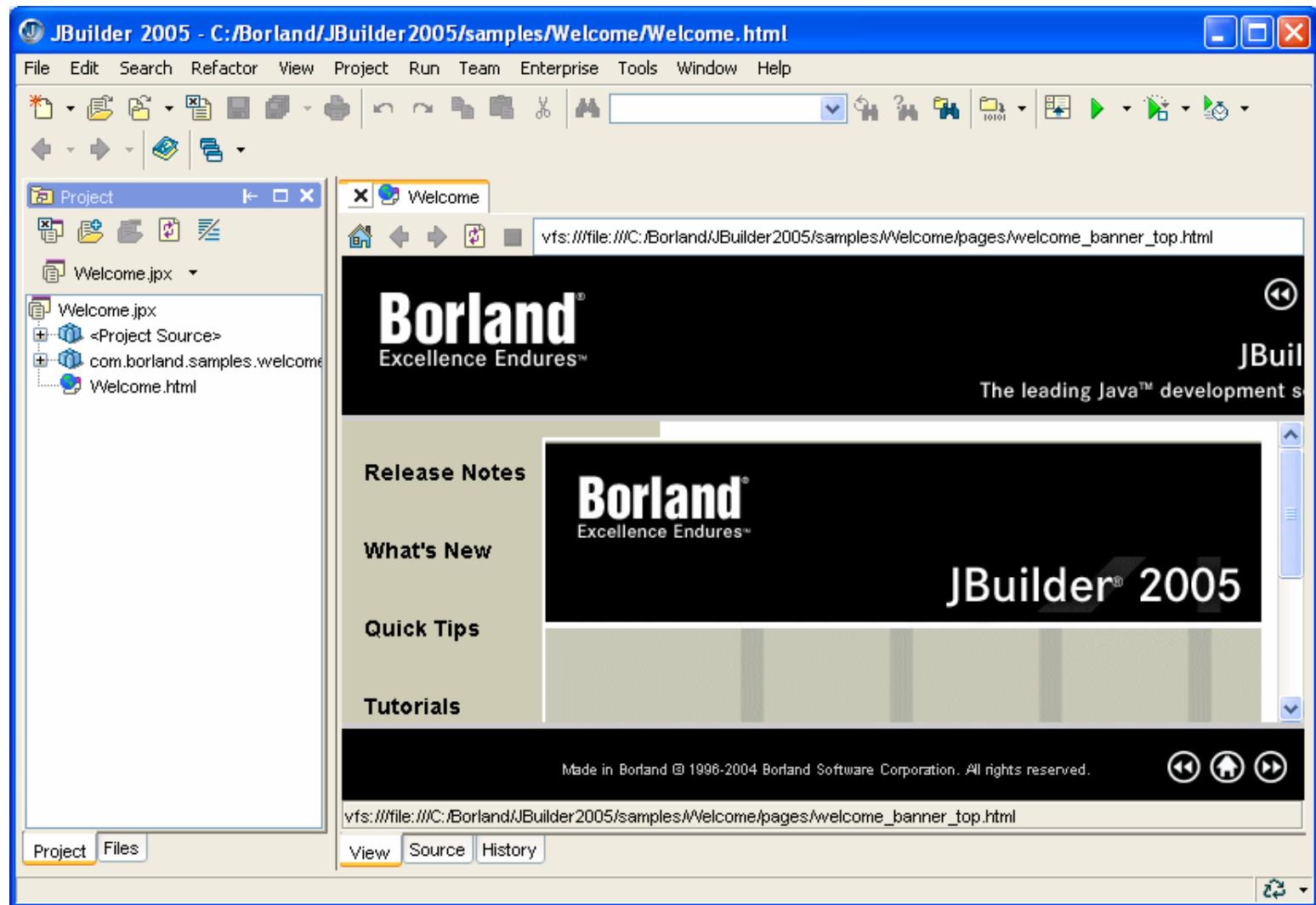
6.1 Giới thiệu Applet Java

- Chúng ta đã quen với các điều khiển (control) được dùng trong các môi trường thiết kế trực quan giao diện phần mềm như TextBox, Button, ListBox, ComboBox,...
- Microsoft đưa ra công nghệ ActiveX Control để giúp người lập trình tự tạo thêm các điều khiển theo nhu cầu riêng của mình. ActiveX là linh kiện phần mềm cấp hệ thống, nghĩa là mỗi khi nó được đăng ký vào Windows thì bất kỳ ứng dụng nào cũng có thể dùng nó, ứng dụng đó có thể là ứng dụng Windows truyền thống hay 1 trang Web. Cách thức sử dụng 1 ActiveX giống y như cách dùng điều khiển có sẵn.
- Xây dựng ActiveX gồm 2 bước chính : định nghĩa giao tiếp sử dụng và hiện thực chi tiết bên trong.
- Giao tiếp sử dụng ActiveX gồm 4 loại chân (pin, entry) : thuộc tính, tác vụ, sự kiện nhập, sự kiện xuất.
- Trong phần này, chúng ta sẽ giới thiệu qui trình điển hình để xây dựng 1 Applet bằng môi trường JBuilder



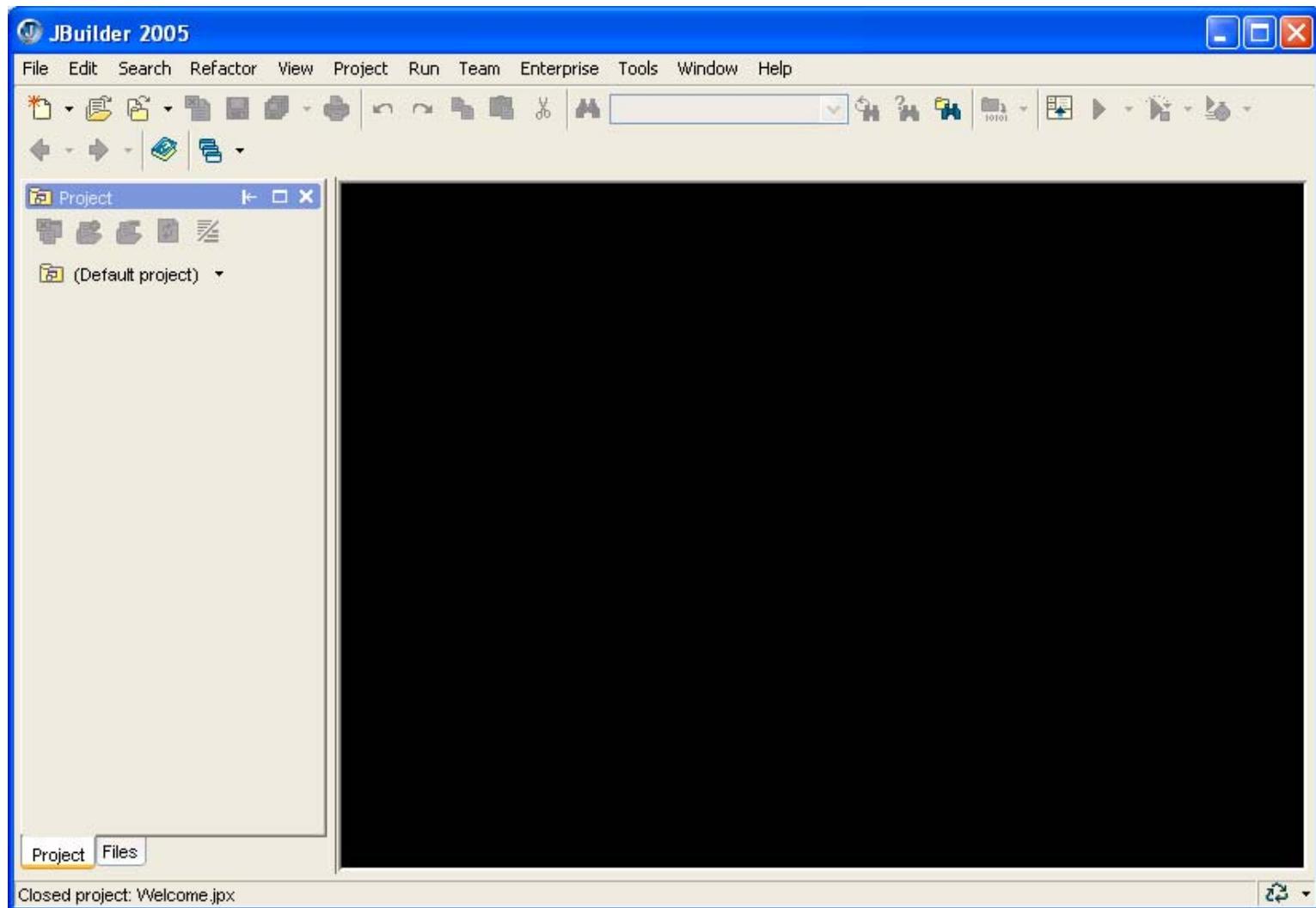
6.2 Qui trình xây dựng Applet bằng JBuilder

1. Chạy JBuilder (hoặc bằng icon shortcut trên desktop hoặc bằng menu Start Programs.. .), cửa sổ đầu tiên có dạng :



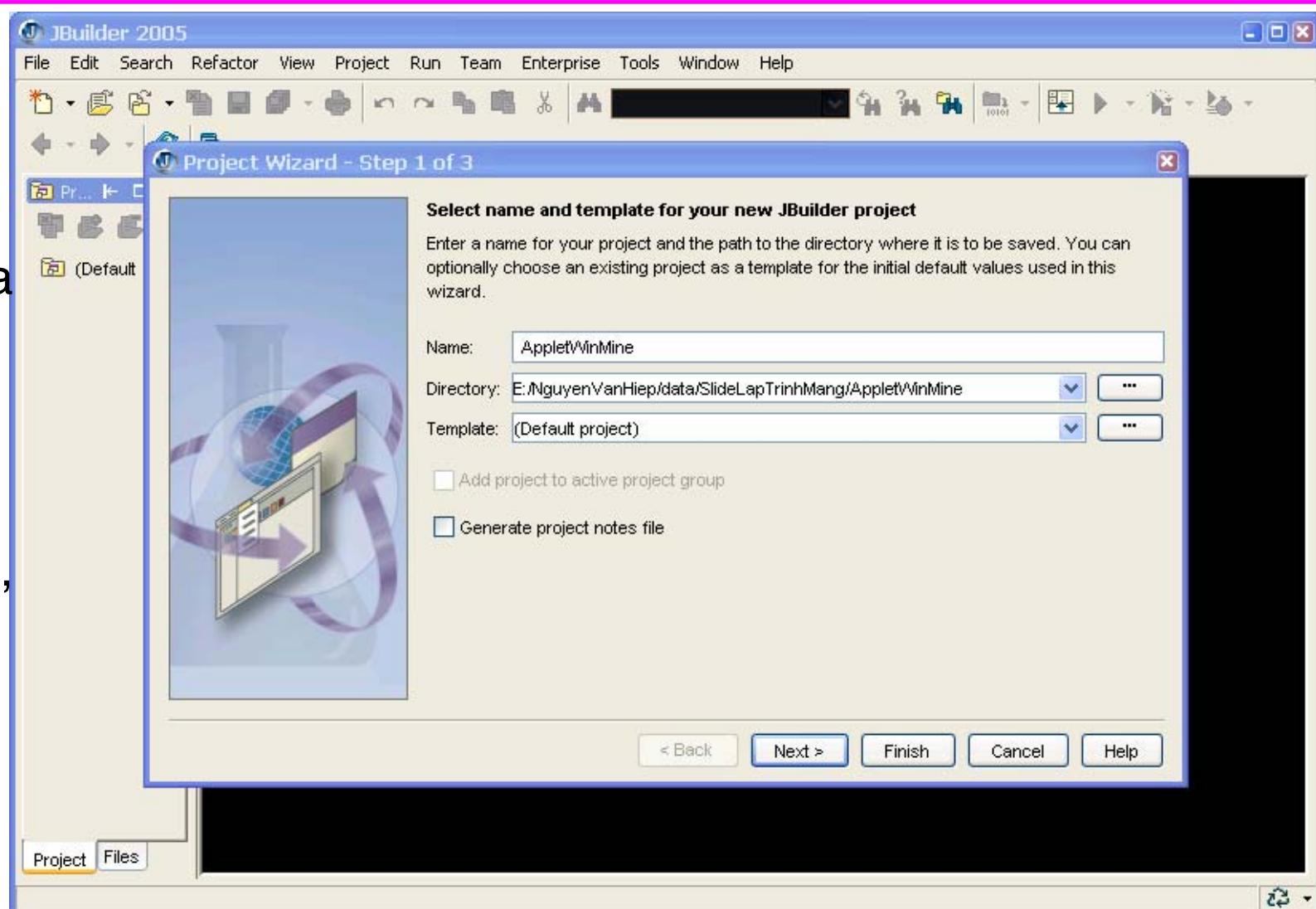
Qui trình xây dựng Applet bằng JBuilder

2. Nếu thấy JBuilder hiển thị Project nào đó (của lần cuối cùng trước đó), chọn menu File.Close Projects để hiển thị cửa sổ Close Projects rồi chọn button All và OK để đóng các Project lại.



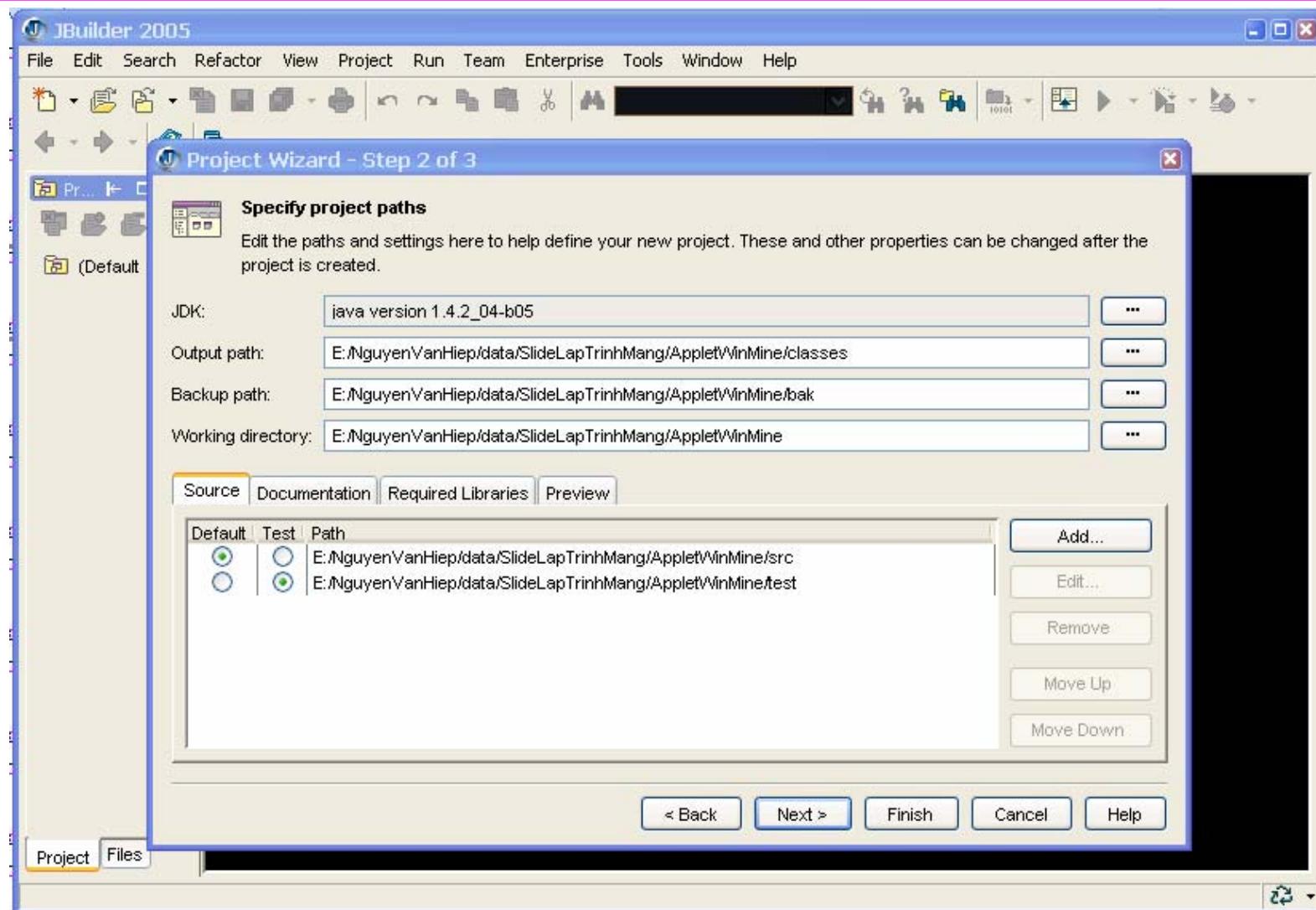
Qui trình xây dựng Applet bằng JBuilder

3. Tạo Project mới bằng cách vào menu File.New Project. Khi cửa sổ Step 1 dưới đây hiển thị, nhập tên Project là AppletWinMine, chọn vị trí thư mục rồi ấn button Next.



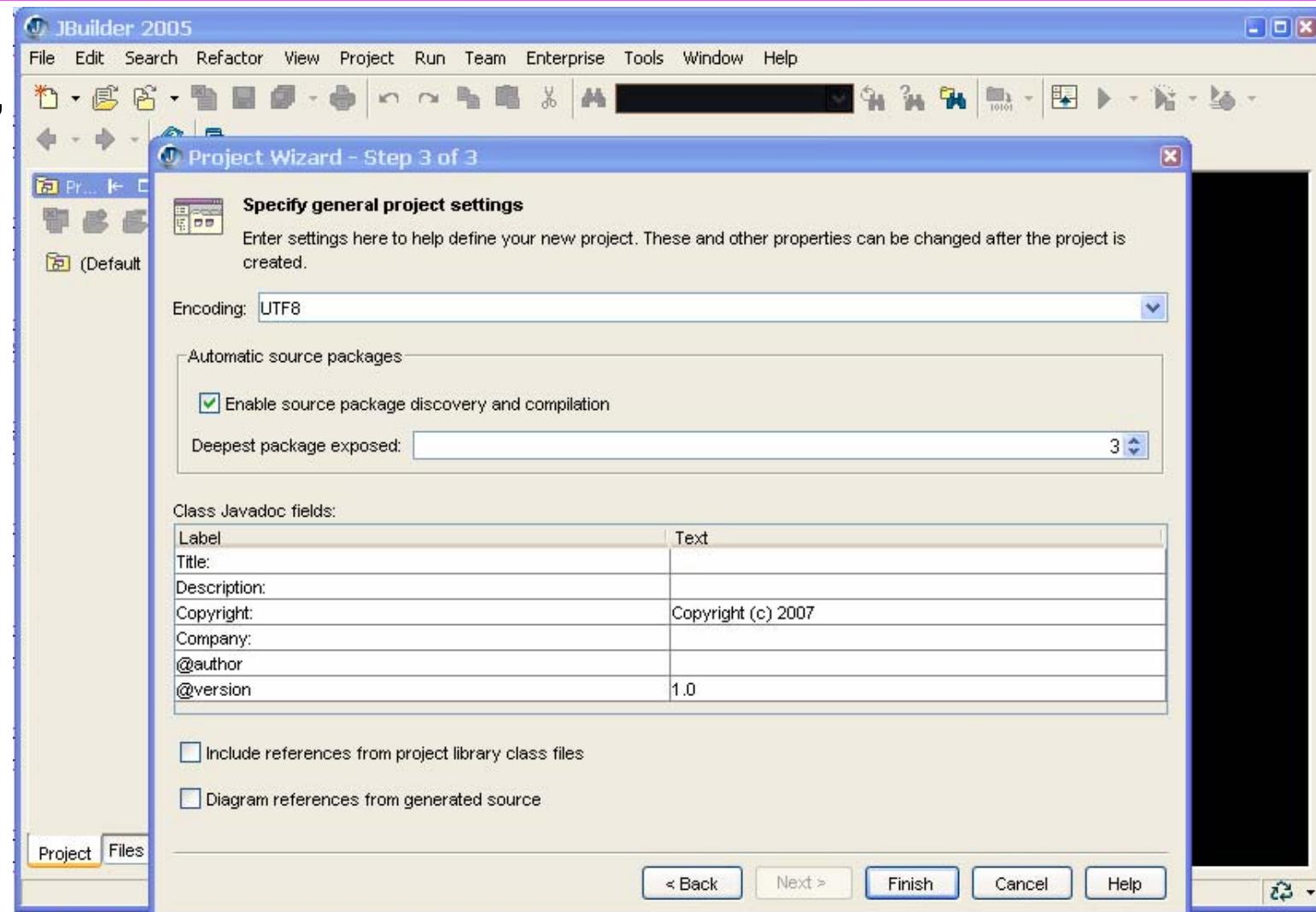
Qui trình xây dựng Applet bằng JBuilder

4. Khi cửa sổ Step 2 dưới đây hiển thị, hiệu chỉnh các thông số cần thiết (nên dùng giá trị mặc định) rồi ấn button Next.



Qui trình xây dựng Applet bằng JBuilder

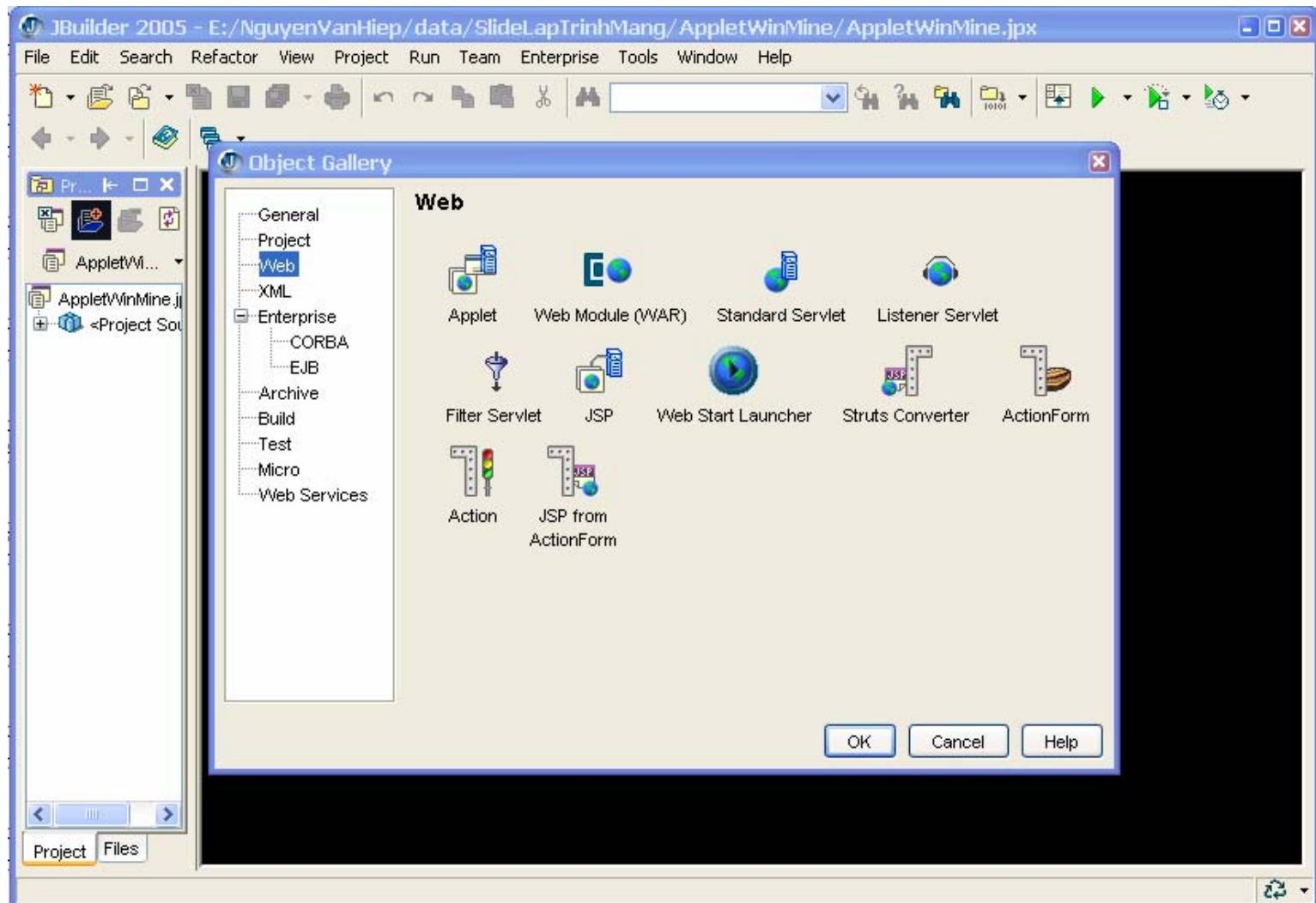
5. Khi cửa sổ Step 3 dưới đây hiển thị, hiệu chỉnh các thông số cần thiết (nên hiệu chỉnh nội dung của mục Encoding thành UTF8 để hỗ trợ tốt các chuỗi tiếng Việt Unicode trong các phần tử giao diện) rồi ấn button Finish để hoàn tất việc thiết lập các tham số của Project.



Qui trình xây dựng Applet bằng JBuilder

6. Chọn menu

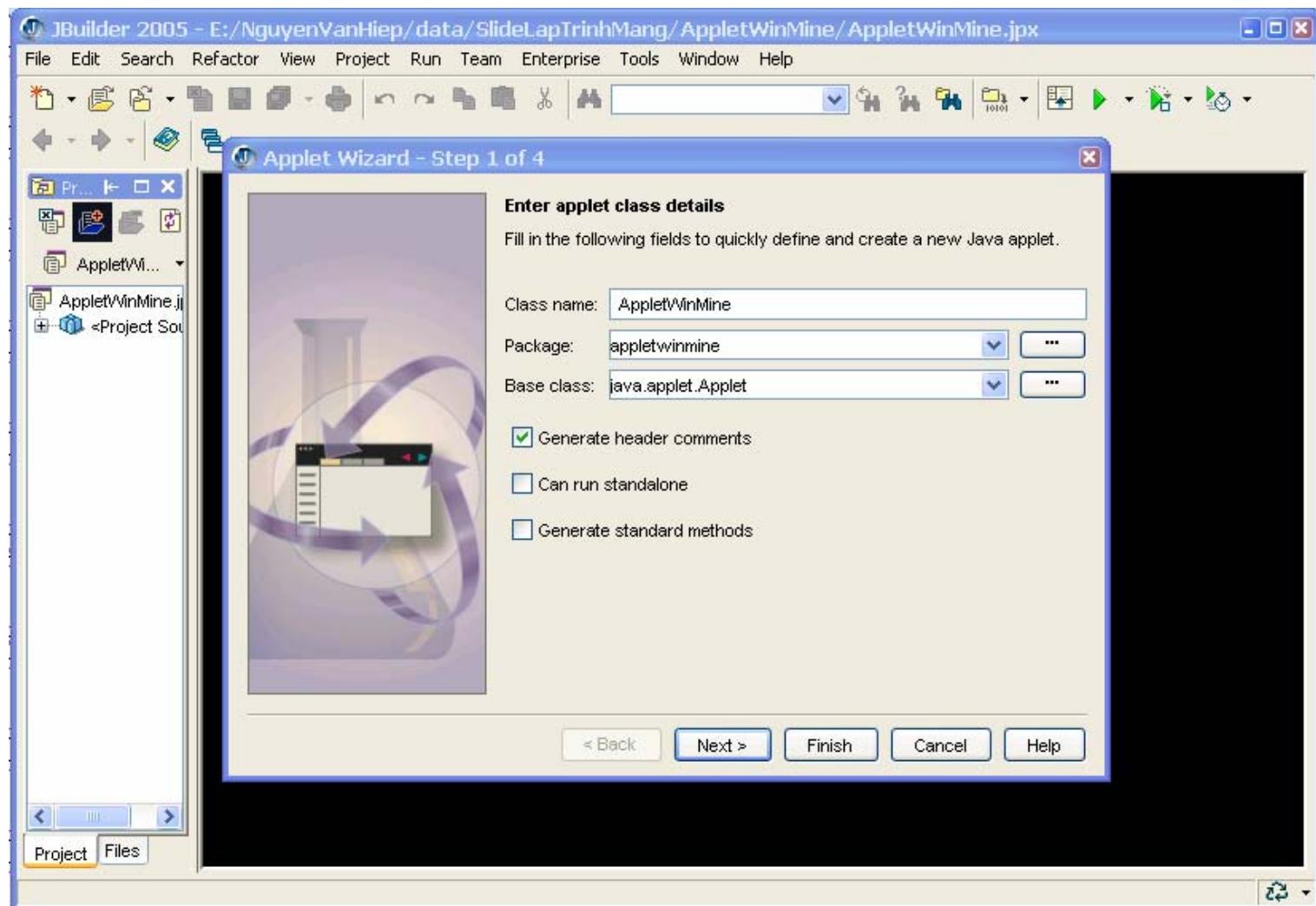
File.New để mở lại
của sổ Object
Gallery, chọn mục
Web, chọn icon
Applet và OK để
tạo 1 Applet
Java mới.



Qui trình xây dựng Applet bằng JBuilder

7. Khi cửa sổ

Step 1 dưới đây hiển thị, nhập tên Applet là AppletWinMine rồi chọn button Finish để hoàn thành qui trình Wizard tạo Applet với các tham số mặc định còn lại.



Qui trình xây dựng Applet bằng JBuilder

8. Cửa sổ soạn code cho Applet ban đầu có dạng bên. Hiệu chỉnh lại phát biểu class, thân hàm jbinit() và viết thêm 1 số hàm xử lý sự kiện và hàm dịch vụ cho trò chơi dò mìn như các slide kế.

The screenshot shows the JBuilder 2005 IDE interface. The title bar reads "JBuilder 2005 - E:/NguyenVanHiep/data/SlideLapTrinhMang/AppletWinMine/src/appletwinmine/AppletWinMine....". The menu bar includes File, Edit, Search, Refactor, View, Project, Run, Team, Enterprise, Tools, Window, Help. The toolbar has various icons for file operations like Open, Save, Print, and Run. The left sidebar contains a Project Explorer with a tree view of files: AppletWinMine.java, AppletWinMine.jpr, and AppletWinMine.wtp. Below it is a Structure view showing imports and a detailed tree of class members. The main editor window displays the Java code for "AppletWinMine.java". The code includes package declarations, imports for awt, event, and applet, a multi-line comment with copyright and company details, and a public class definition that extends Applet and sets isStandalone to false.

```
package appletwinmine;  
import java.awt.*;  
import java.awt.event.*;  
import java.applet.*;  
  
/**  
 * <p>Title: </p>  
 *  
 * <p>Description: </p>  
 *  
 * <p>Copyright: Copyright (c) 2007</p>  
 *  
 * <p>Company: </p>  
 *  
 * @author not attributable  
 * @version 1.0  
 */  
  
public class AppletWinMine extends Applet {  
    boolean isStandalone = false;  
}
```



Qui trình xây dựng Applet bằng JBuilder

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import javax.swing.*;

public class AppWinMine extends Applet implements MouseListener {
    private boolean isStandalone = false;
    private final int WM_NULL = 9;
    private final int WM_MINI = 10;
    private final int WM_MINA = 11;
    private int odado;
    private Image wm_null, wm_mini, wm_mina,wm_buttonI,wm_buttonA;
    private Image wm_imglist[] = new Image[12];
    private boolean data_tbl[][]= new boolean[8][8];
    private boolean stat_tbl[][] = new boolean[8][8];
    private int disp_tbl[][] = new int[8][8];
    private boolean dangchoi;
    private int xbt=20, ybt=190, xmsg=20, ymsg=170;
    private String mesg;
    private Image jbt;
```



Qui trình xây dựng Applet bằng JBuilder

```
...  
//Component initialization  
private void jblInit() throws Exception {  
    wm_imglist[0] = getImage(getCodeBase(), "wm_nul1.gif");  
    wm_imglist[1] = getImage(getCodeBase(), "wm_1.gif");  
    wm_imglist[2] = getImage(getCodeBase(), "wm_2.gif");  
    wm_imglist[3] = getImage(getCodeBase(), "wm_3.gif");  
    wm_imglist[4] = getImage(getCodeBase(), "wm_4.gif");  
    wm_imglist[5] = getImage(getCodeBase(), "wm_5.gif");  
    wm_imglist[6] = getImage(getCodeBase(), "wm_6.gif");  
    wm_imglist[7] = getImage(getCodeBase(), "wm_7.gif");  
    wm_imglist[8] = getImage(getCodeBase(), "wm_8.gif");  
    wm_imglist[WM_NULL] = getImage(getCodeBase(), "wm_nul.gif");  
    wm_imglist[WM_MINI] = getImage(getCodeBase(), "wm_mini.gif");  
    wm_imglist[WM_MINA] = getImage(getCodeBase(), "wm_mina.gif");  
    wm_buttonI = getImage(getCodeBase(), "wm_buttonI.gif");  
    wm_buttonA = getImage(getCodeBase(), "wm_buttonA.gif");  
    addMouseListener(this);  
    wm_init();  
}
```



Qui trình xây dựng Applet bằng JBuilder

```
private void wm_init() {  
    int h,c,mcnt;  
    odado = 0;  
    dangchoi = true;  
    jbt = wm_button1;  
    mesg = "Hay do tung o min";  
    // Tao cac bang va khai dong trang thai  
    for (h=0;h<8;h++)  
        for (c=0;c<8;c++) {  
            disp_tbl[h][c] = WM_NULL;  
            data_tbl[h][c] = false;  
            stat_tbl[h][c] = false;  
        }  
    // An button choi tiep  
    //jbtStart.disable();  
    // Xep ngau nhien 10 trai min  
    mcnt = 0;
```



Qui trình xây dựng Applet bằng JBuilder

```
while (mcnt <10) {  
    h = (int)Math.round(Math.random()*7);  
    c = (int)Math.round(Math.random()*7);  
    if (data_tbl[h][c]==false) {  
        data_tbl[h][c]= true;  
        mcnt++;  
    }  
}  
public void paint(Graphics g) {  
int h,c;  
    for (h= 0; h<8; h++)  
        for (c=0;c<8;c++)  
            g.drawImage(wm_imglist[disp_tbl[h][c]], c*17, h*17, this);  
    g.drawString(mesg,xmsg,ymsg);  
    g.drawImage(jbt,xbt,ybt,this);  
}
```



Qui trình xây dựng Applet bằng JBuilder

```
// Tinh so min xung quanh vi tri (h,c)
private int winecnt (int h, int c) {
int cnt=0;
    if (h-1>=0 && c-1>=0 && data_tbl[h-1][c-1]) cnt++;
    if (h-1>=0 && data_tbl[h-1][c]) cnt++;
    if (h-1>=0 && c+1<8 && data_tbl[h-1][c+1]) cnt++;
    if (c-1>=0 && data_tbl[h][c-1]) cnt++;
    if (c+1<8 && data_tbl[h][c+1]) cnt++;
    if (h+1<8 && c-1>=0 && data_tbl[h+1][c-1]) cnt++;
    if (h+1<8 && data_tbl[h+1][c]) cnt++;
    if (h+1<8 && c+1<8 && data_tbl[h+1][c+1]) cnt++;
    return cnt;
}
// Do tu dong cac o bao quanh o h,c
private void doquanh(int h, int c) {
    if (h-1>=0 && c-1>=0 && stat_tbl[h-1][c-1]==false) domin(h-1,c-1);
    if (h-1>=0 && stat_tbl[h-1][c]==false) domin(h-1,c);
    if (h-1>=0 && c+1<8 && stat_tbl[h-1][c+1]==false) domin(h-1,c+1);
    if (c-1>=0 && stat_tbl[h][c-1]==false) domin(h,c-1);
```



Qui trình xây dựng Applet bằng JBuilder

```
if (c+1<8 && stat_tbl[h][c+1]==false) domin(h,c+1);
if (h+1<8 && c-1>=0 && stat_tbl[h+1][c-1]==false) domin(h+1,c-1);
if (h+1<8 && stat_tbl[h+1][c]==false) domin(h+1,c);
if (h+1<8 && c+1<8 && stat_tbl[h+1][c+1]==false) domin(h+1,c+1);
}
// Do min o vi tri h,c
private int domin(int h,int c){
int i,j,cnt;
if (stat_tbl[h][c]) return 0;
if (data_tbl[h][c]) { // Dap trung min
for (i=0;i<8;i++)
    for (j=0;j<8;j++)
        if (data_tbl[i][j])
            disp_tbl[i][j] = WM_MINI;
disp_tbl[h][c] = WM_MINA;
mesg ="Ba da chet";
//jbtStart.enable();
dangchoi = false;
} else { // do trung cho khong co min
```



Qui trình xây dựng Applet bằng JBuilder

```
cnt = winecnt(h,c);
stat_tbl[h][c] = true;
disp_tbl[h][c] = cnt;
odado++;
if (cnt==0) doquanh(h,c);
}
if (odado ==54) {
mesg = "Ba da thang";
//jbtStart.enable();
    dangchoi = false;
}
return 1;
}
public void mouseReleased(MouseEvent e){
int x = e.getX();
int y = e.getY();
if (x > 8*17 || y > 8*17)
    if (xbt<=x && x <=xbt+60 && ybt<=y && y<=y+20) {
        jbt = wm_button1;
```



Qui trình xây dựng Applet bằng JBuilder

```
if (!dangchoi){  
    wm_init();  
}  
repaint();  
}  
}  
public void mousePressed(MouseEvent e) {  
    int h,c;  
    int x = e.getX(), y = e.getY();  
    if (x > 8*17 || y > 8*17) {  
        if (xbt<=x && x <=xbt+60 && ybt<=y && y<=y+20)  
            jbt = wm_buttonA;  
    } else { // an chout vao ban min  
        if (!dangchoi) return;  
        h = y/17; c = x/17;  
        domin(h,c);  
    }  
    repaint();  
}
```



Qui trình xây dựng Applet bằng JBuilder

```
public void mouseClicked(MouseEvent e) { }  
public void mouseEntered(MouseEvent e) { }  
public void mouseExited(MouseEvent e) { }  
// kết thúc code Java
```

- Sau khi soạn xong code, thử dịch và chạy Applet.
- Mặc định JBuilder có tạo sẵn 1 trang Web chứa Applet nhờ tag lệnh <applet>. Nội dung của trang Web được liệt kê trong slide kế.



Qui trình xây dựng Applet bằng JBuilder

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>HTML Test Page</title>
</head>
<body>
appletwinmine.AppletWinMine will appear below in a Java enabled browser.<br>
<applet
    codebase = "."
    code    = "appletwinmine.AppletWinMine.class"
    name    = "TestApplet"
    width   = "400"
    height  = "300"
    hspace  = "0"
    vspace  = "0"
    align   = "middle"
    >
</applet>
</body>
</html>
```



Qui trình xây dựng Applet bằng JBuilder

Lưu ý tham số thiết yếu của tag <Applet> là :

- codebase : miêu tả thư mục tham khảo
- code : miêu tả đường dẫn tương đối của file applet so với thư mục tham khảo.



Môn học : Lập trình mạng

Chương 7

LẬP TRÌNH WEB CHẠY TRÊN CLIENT DÙNG ACTIVEX CONTROL

7.1 Giới thiệu ActiveX Control

7.2 Qui trình xây dựng ActiveX dùng VC++

7.3 Qui trình xây dựng Website dùng ActiveX bằng InterDev



7.1 Giới thiệu ActiveX Control

- Chúng ta đã quen với các điều khiển (control) được dùng trong các môi trường thiết kế trực quan giao diện phần mềm như TextBox, Button, ListBox, ComboBox,...
- Microsoft đưa ra công nghệ ActiveX Control để giúp người lập trình tự tạo thêm các điều khiển theo nhu cầu riêng của mình. ActiveX là linh kiện phần mềm cấp hệ thống, nghĩa là mỗi khi nó được đăng ký vào Windows thì bất kỳ ứng dụng nào cũng có thể dùng nó, ứng dụng đó có thể là ứng dụng Windows truyền thống hay 1 trang Web. Cách thức sử dụng 1 ActiveX giống y như cách dùng điều khiển có sẵn.
- Xây dựng ActiveX gồm 2 bước chính : định nghĩa giao tiếp sử dụng và hiện thực chi tiết bên trong.
- Giao tiếp sử dụng ActiveX gồm 4 loại chân (pin, entry) : thuộc tính, tác vụ, sự kiện nhập, sự kiện xuất.

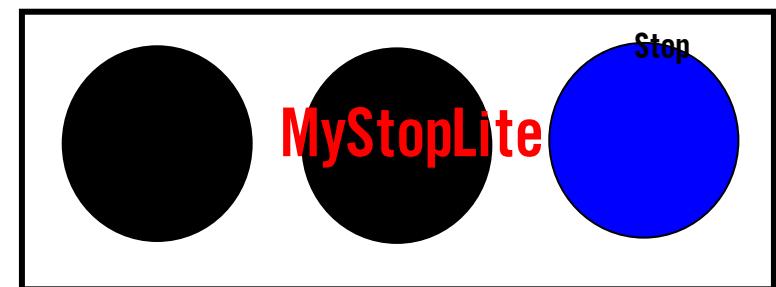


7.2 Tạo ActiveX bằng VC++

Để thấy rõ qui trình xây dựng 1 ActiveX, chúng ta hãy thử dùng VC++ để xây dựng ActiveX có tên là MyStopLite, nó giả lập dàn đèn điều khiển giao lộ với các thông số cụ thể như sau :

- Hiển thị : đang ở trạng thái nào thì đèn tương ứng sẽ sáng, đèn còn lại tắt (dùng màu đen).
- tác vụ next() : cho phép dàn đèn chuyển về trạng thái kế (đỏ → xanh → vàng)
- thuộc tính Color : cho phép thiết lập dàn đèn về màu tương ứng.
- các events xuất : sẽ được kích hoạt khi dàn đèn hoàn thành việc chuyển về trạng thái mới (Go, Caution, Stop, Testing, Off).
- các events nhập được xử lý : LBUTTONDOWN, khi nhận được events này, dàn đèn sẽ chuyển về trạng thái kế (đỏ → xanh → vàng).

Các events nhập
Caution Go

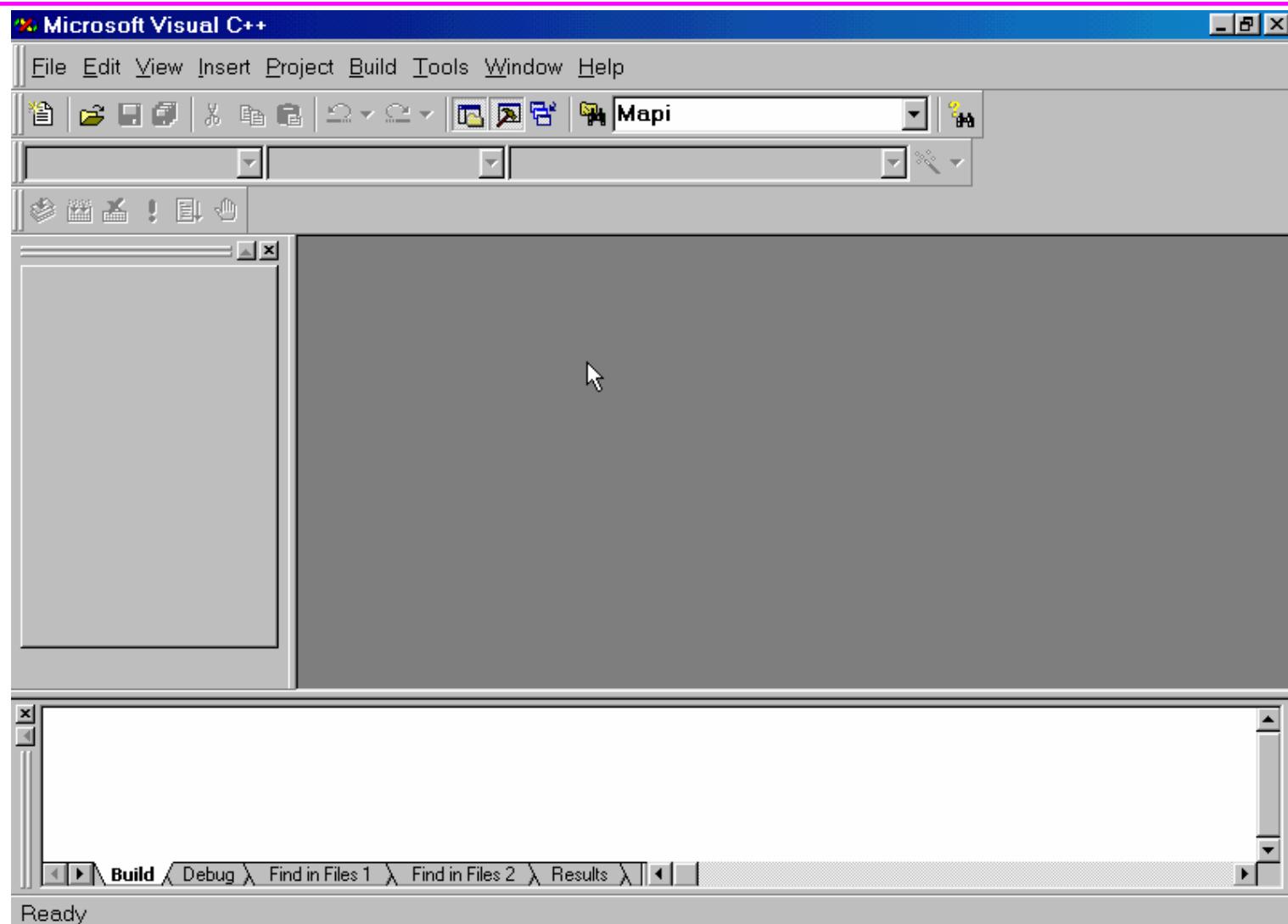


Next
Các method
Color
Các thuộc tính



7.2 Tạo ActiveX bằng VC++

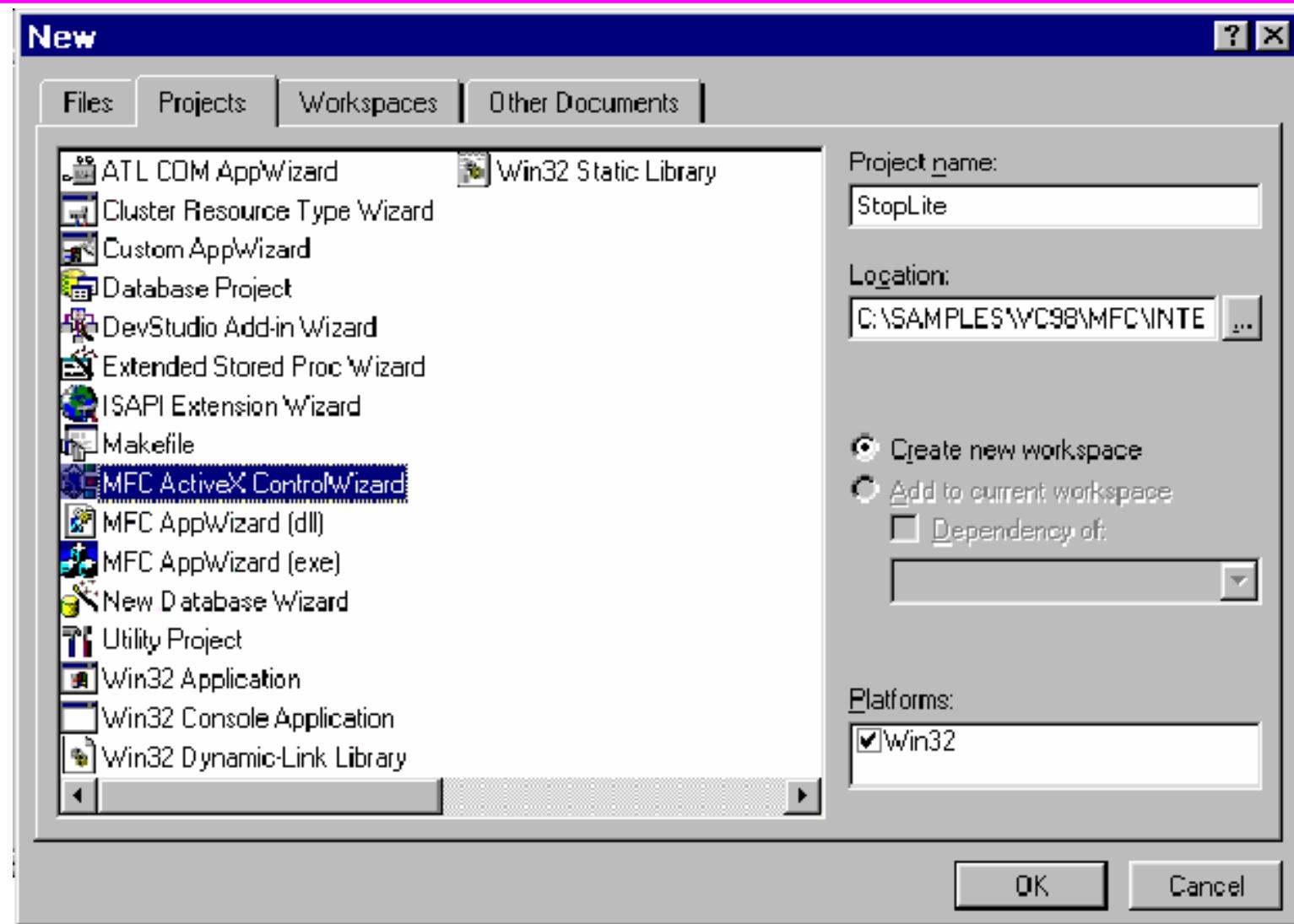
1. Để tạo 1 ActiveX Control bằng VC++, trước hết chạy Visual C++ 6.0 từ Windows (thí dụ chọn mục Start Programs Microsoft Visual Studio 6.0/Microsoft Visual C++ 6.0). Màn hình của VC++ được hiển thị như sau :



7.2 Tạo ActiveX bằng VC++

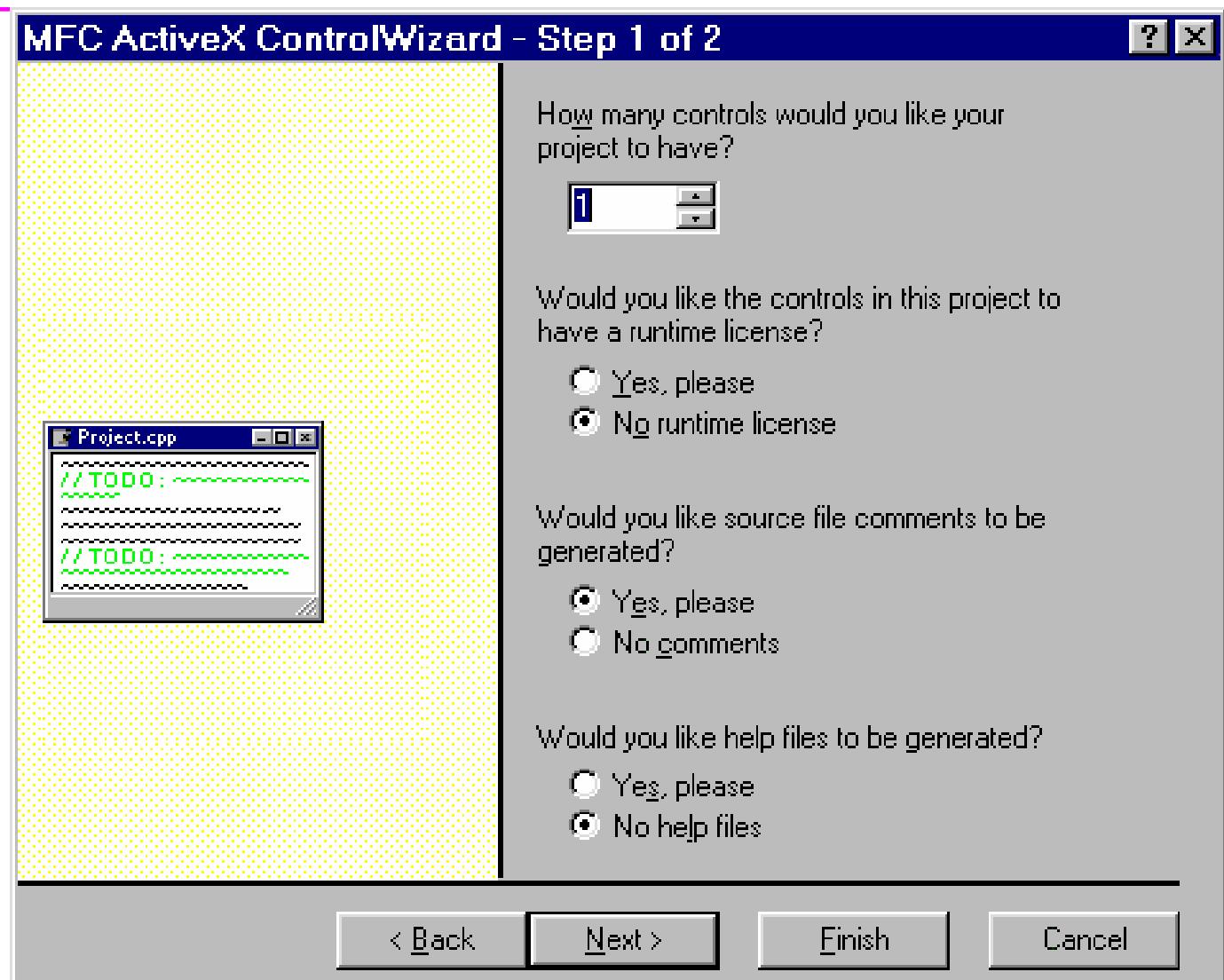
2. Bước 1 : tạo project phần mềm bằng Wizard.

Chọn menu File.New để tạo một Project VC++ chứa ActiveX Control, cửa sổ sau sẽ hiển thị :



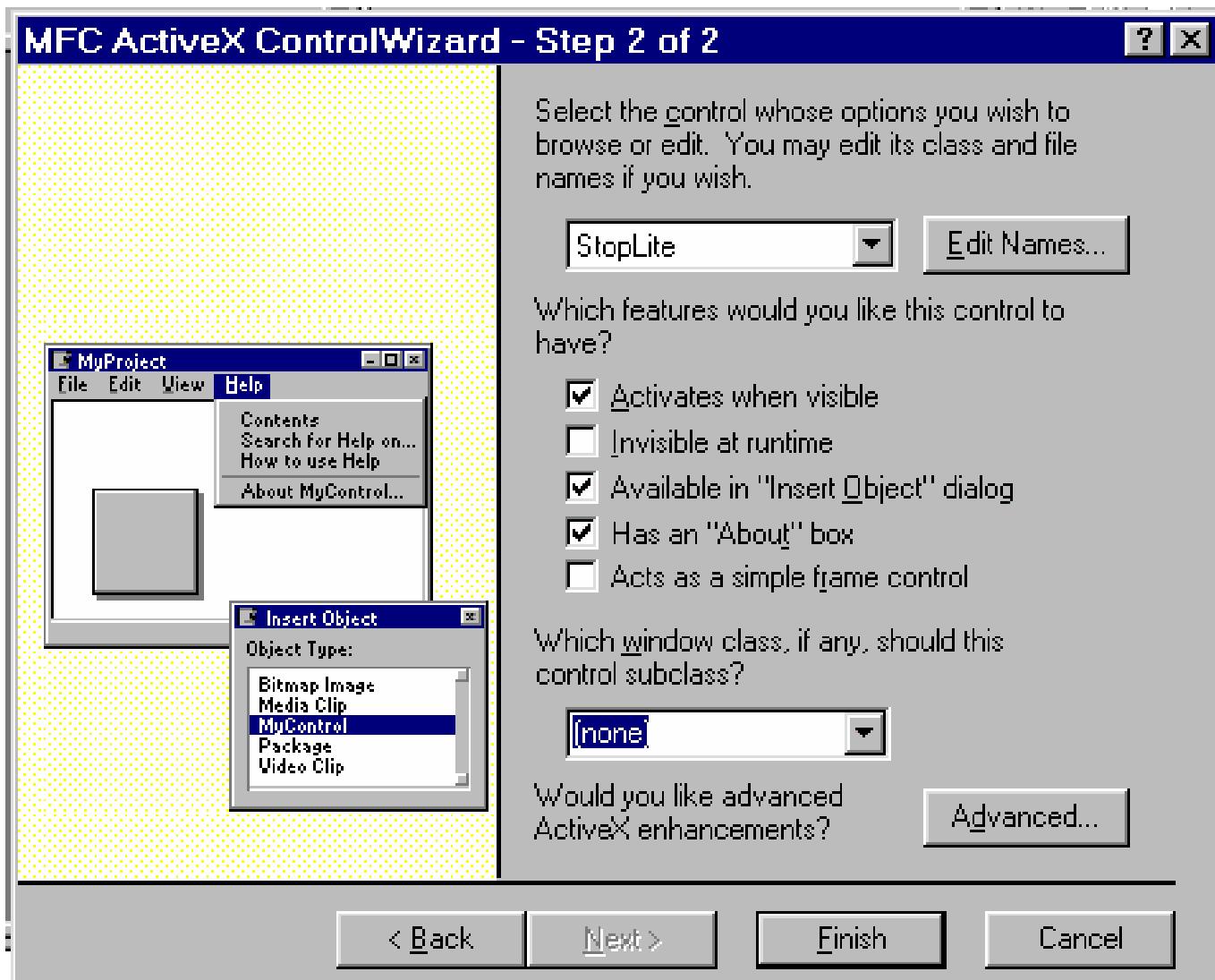
7.2 Tạo ActiveX bằng VC++

3. Chọn mục MFC ActiveX ControlWizard, chọn vị trí thư mục chứa Project, nhập tên project. Tên project trở thành thư mục chứa các file của project. Cuối cùng chọn button Ok để bắt đầu các bước khai báo thông số cho Project.



7.2 Tạo ActiveX bằng VC++

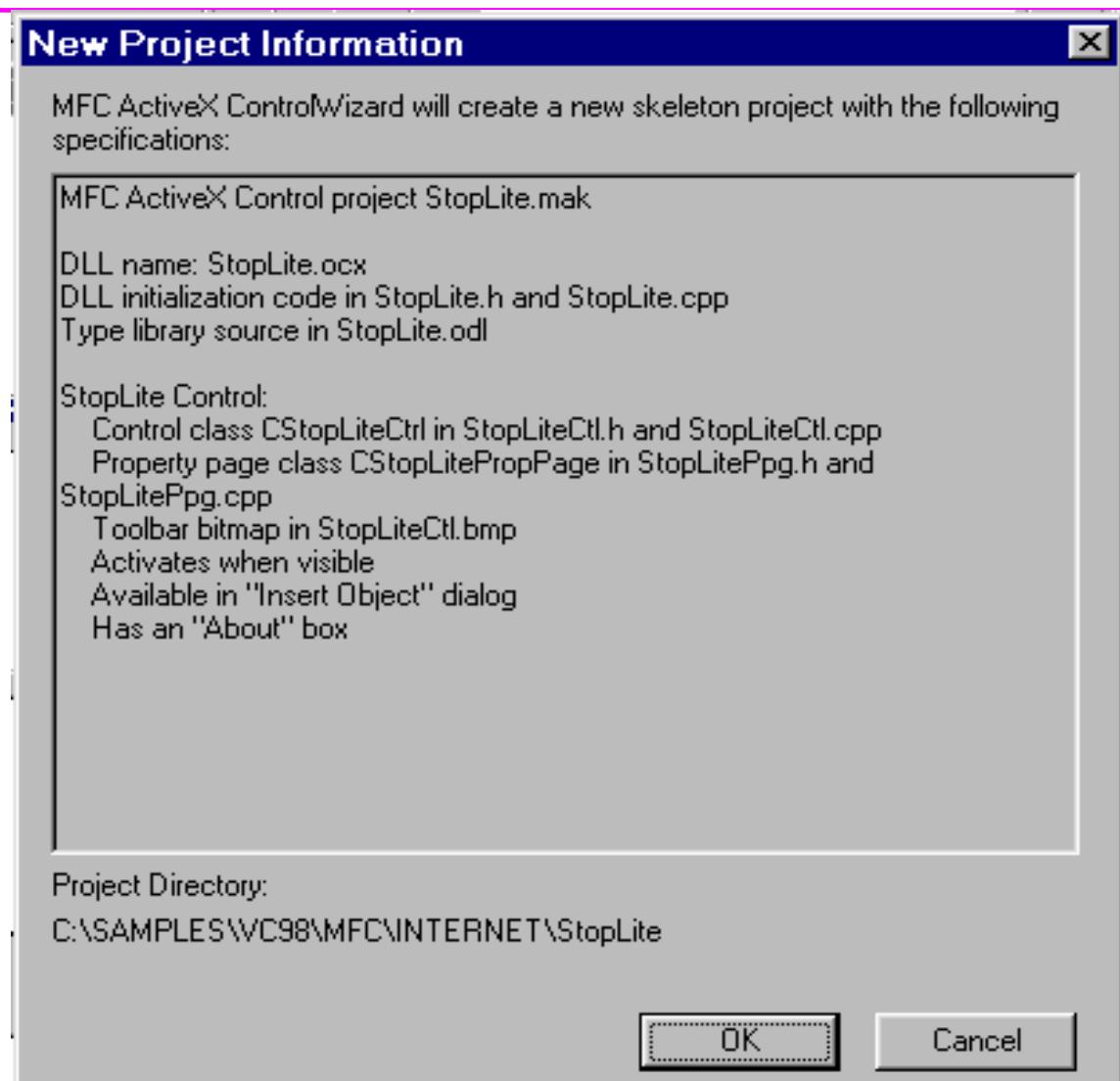
4. Wizard cho Activex Control chỉ có 2 step, hãy trả lời cho từng step rồi chọn button Finish ở step 2



7.2 Tạo ActiveX bằng VC++

5. Cửa sổ tổng kết các thông số vừa xác lập sẽ hiện thị, kiểm tra chúng và quyết định Ok hay Cancel. Lưu ý nếu Cancel thì phải tạo Project lại từ đầu vì các thông số của Project bị Cancel sẽ bị xóa hẳn.

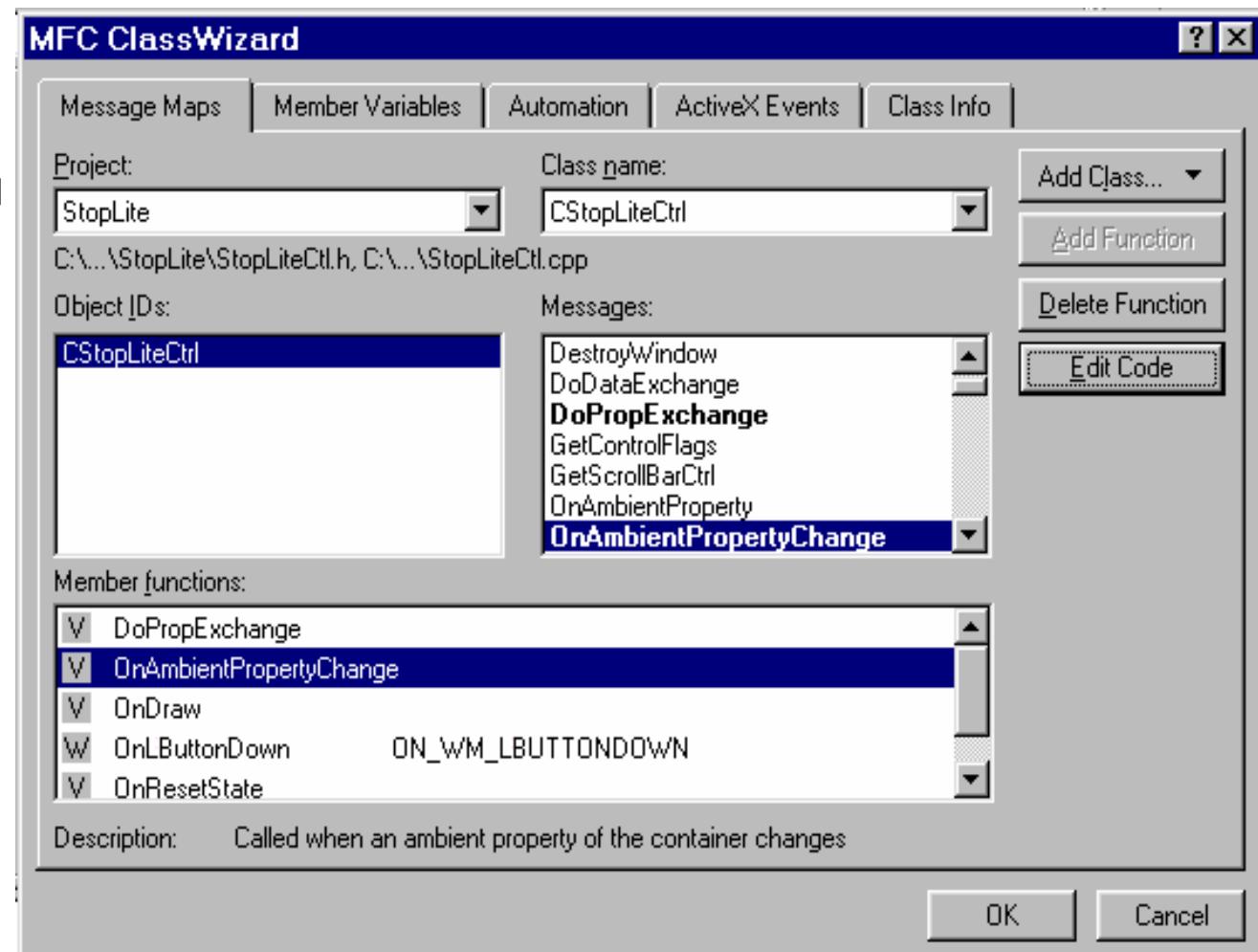
6. Bước 2+3 : thiết kế giao diện cho chương trình và thiết lập thuộc tính cho các phần tử giao diện (không cần làm trong ActiveX Control cụ thể này vì giao diện sẽ do phần mềm tự vẽ lấy khi chạy)



7.2 Tạo ActiveX bằng VC++

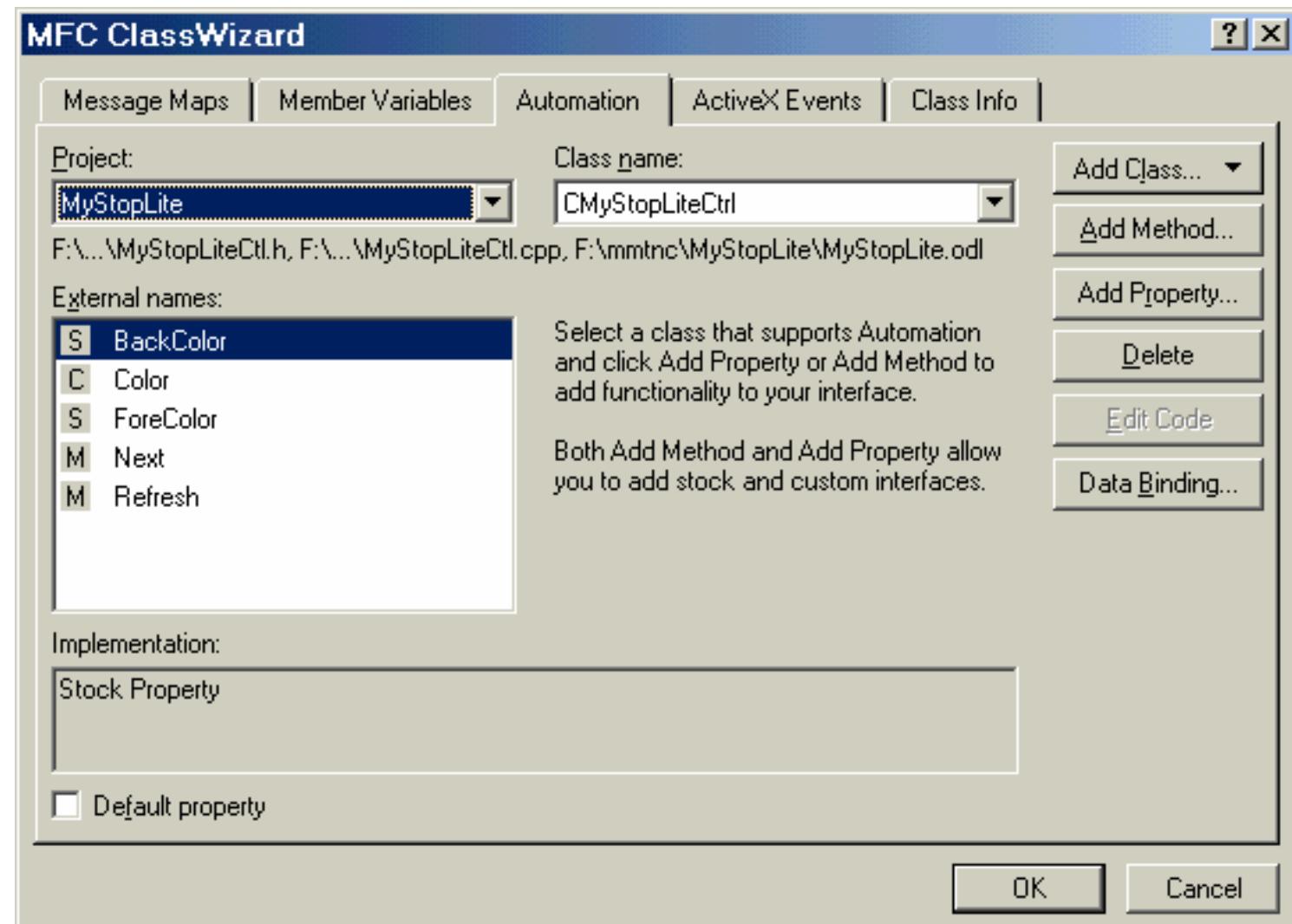
7. Bước 4 : định nghĩa interface của Activex Control, dùng menu

View.Classwizard, cửa sổ sau sẽ hiển thị. Trước hết định nghĩa các chân input events (các hàm xử lý các sự kiện nhập mà Activex Control quan tâm). Để định nghĩa các hàm xử lý sự kiện nhập, chọn page "Message Maps", chọn Project và class name tương ứng Activex control, danh sách các sự kiện có thể xử lý sẽ hiện thị trong Message, chọn từng thông báo cần xử lý rồi ấn button "Add Function".



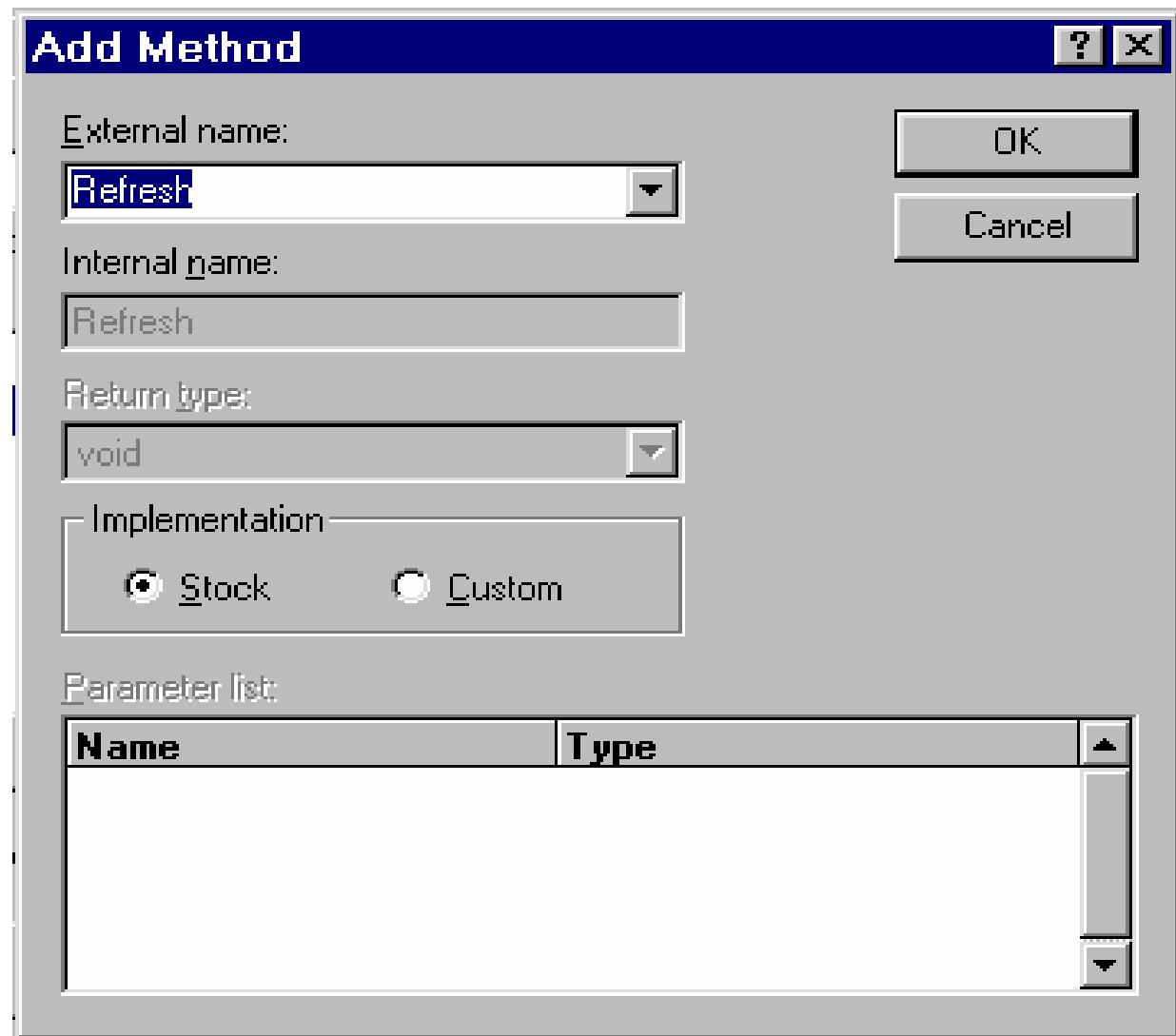
7.2 Tạo ActiveX bằng VC++

8. Để định nghĩa các method và các thuộc tính data, chọn page "Automation", chọn button "Add Method" để tạo từng method, chọn button "Add Propertiy" để tạo từng property (trúâu tượng).



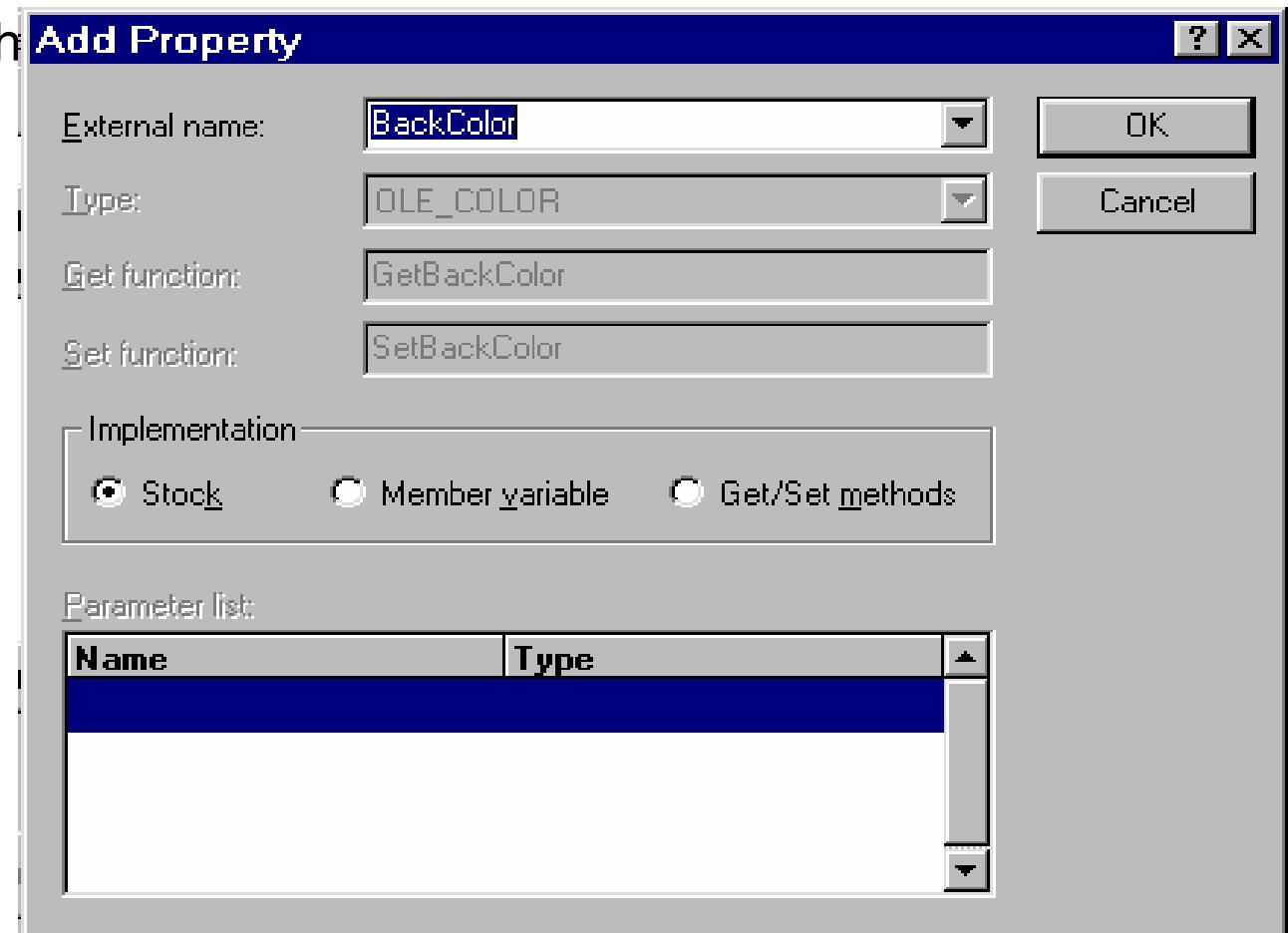
7.2 Tạo ActiveX bằng VC++

9. Ứng với mỗi method được tạo, cửa sổ "Add Method" sẽ hiển thị để ta thiết lập các tính chất của nó : tên bên ngoài, tên bên trong, kiểu trả về, danh sách thông số hình thức... Có 2 loại method : stock và customer. Method stock là method có sẵn của môi trường hay của class cha.



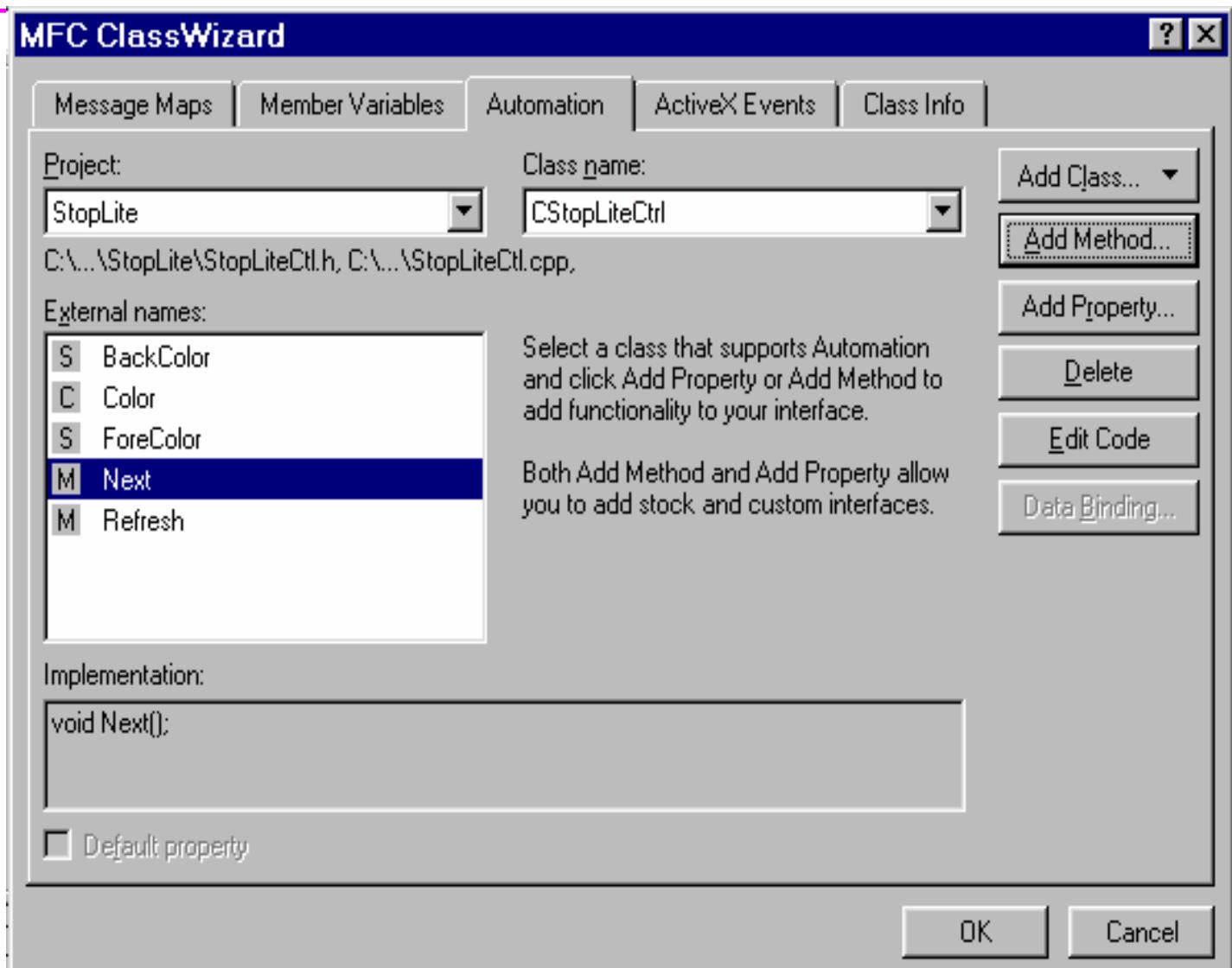
7.2 Tạo ActiveX bằng VC++

10. Ứng với mỗi property được tạo, cửa sổ "Add Property" sẽ hiển thị để ta thiết lập các tính chất của nó : tên bên ngoài, kiểu dữ liệu, tên hàm get và set tương ứng, danh sách thông số hình thức của từng hàm... Có 3 loại property : stock, member variable và get/set. Property stock là property có sẵn của môi trường hay của class cha. Property "member variable" được hiện thực bằng 1 biến thành viên tương ứng của class. Property get/set tương ứng với 2 method get/set.



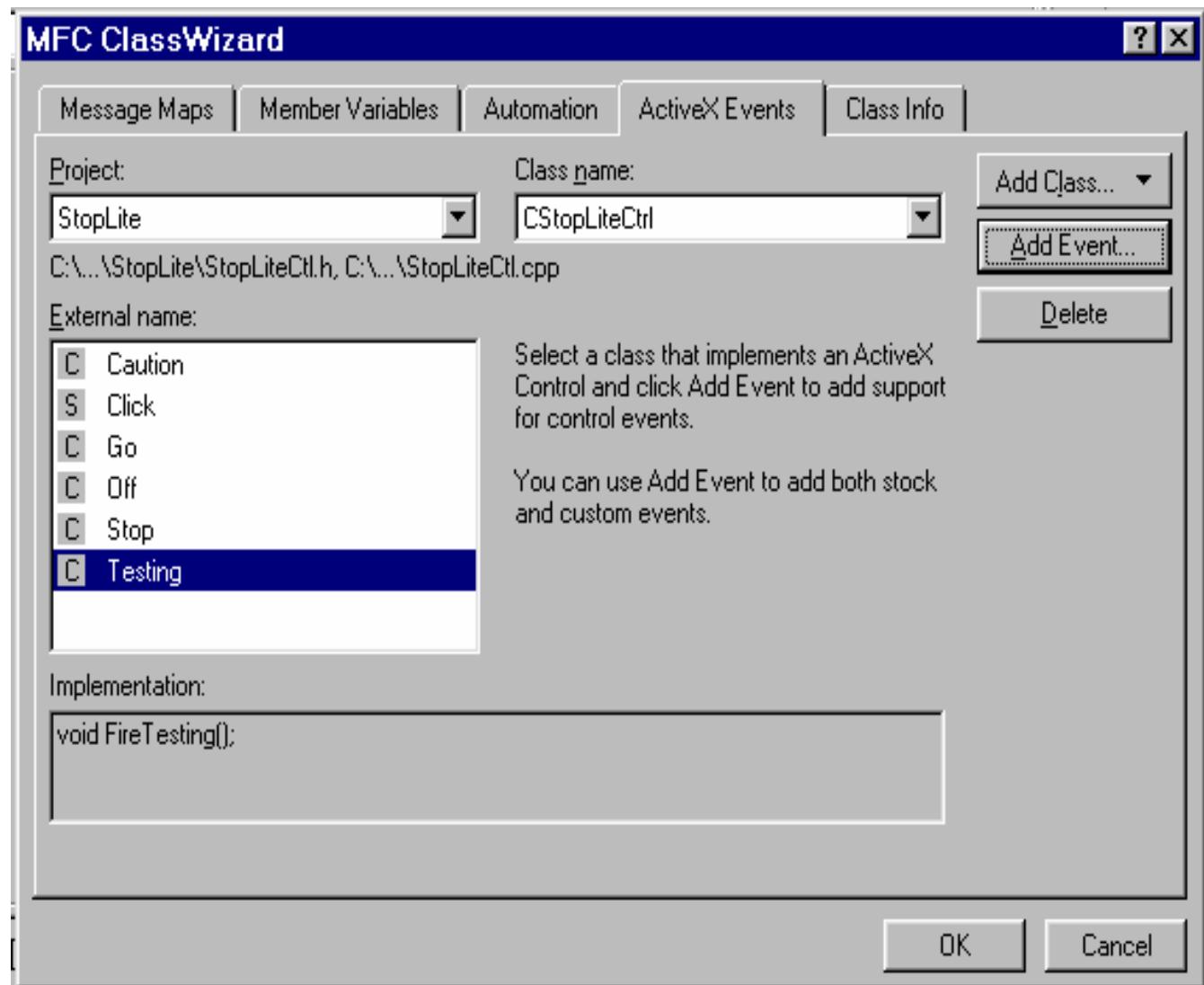
7.2 Tạo ActiveX bằng VC++

11. Sau khi tạo các method và property, chúng sẽ được hiện thị trong danh sách "External names" của cửa sổ Class Wizard.



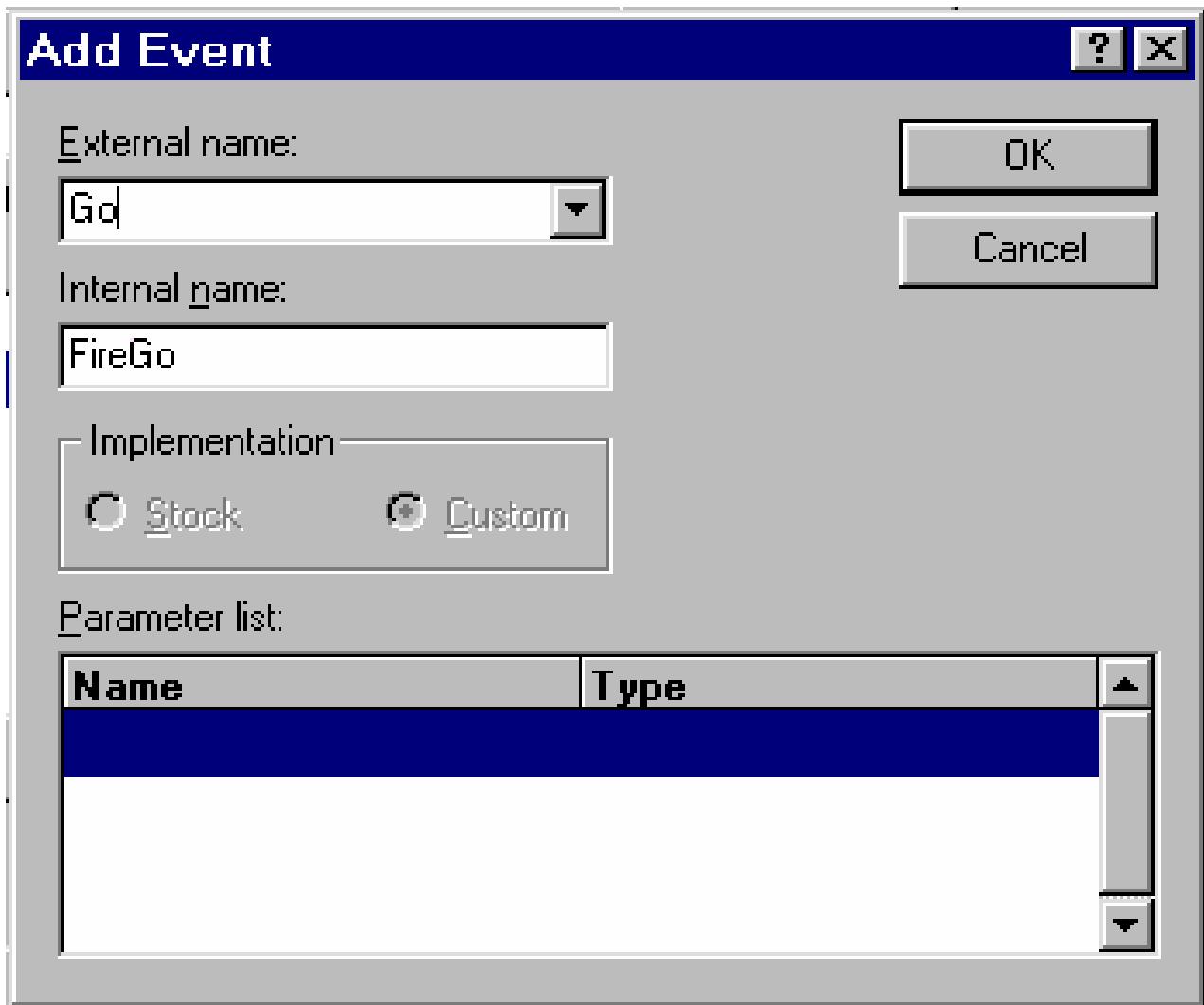
7.2 Tạo ActiveX bằng VC++

12. Để tạo các output event, chọn page "ActiveX Event", chọn button "Add Event" để tạo từng event



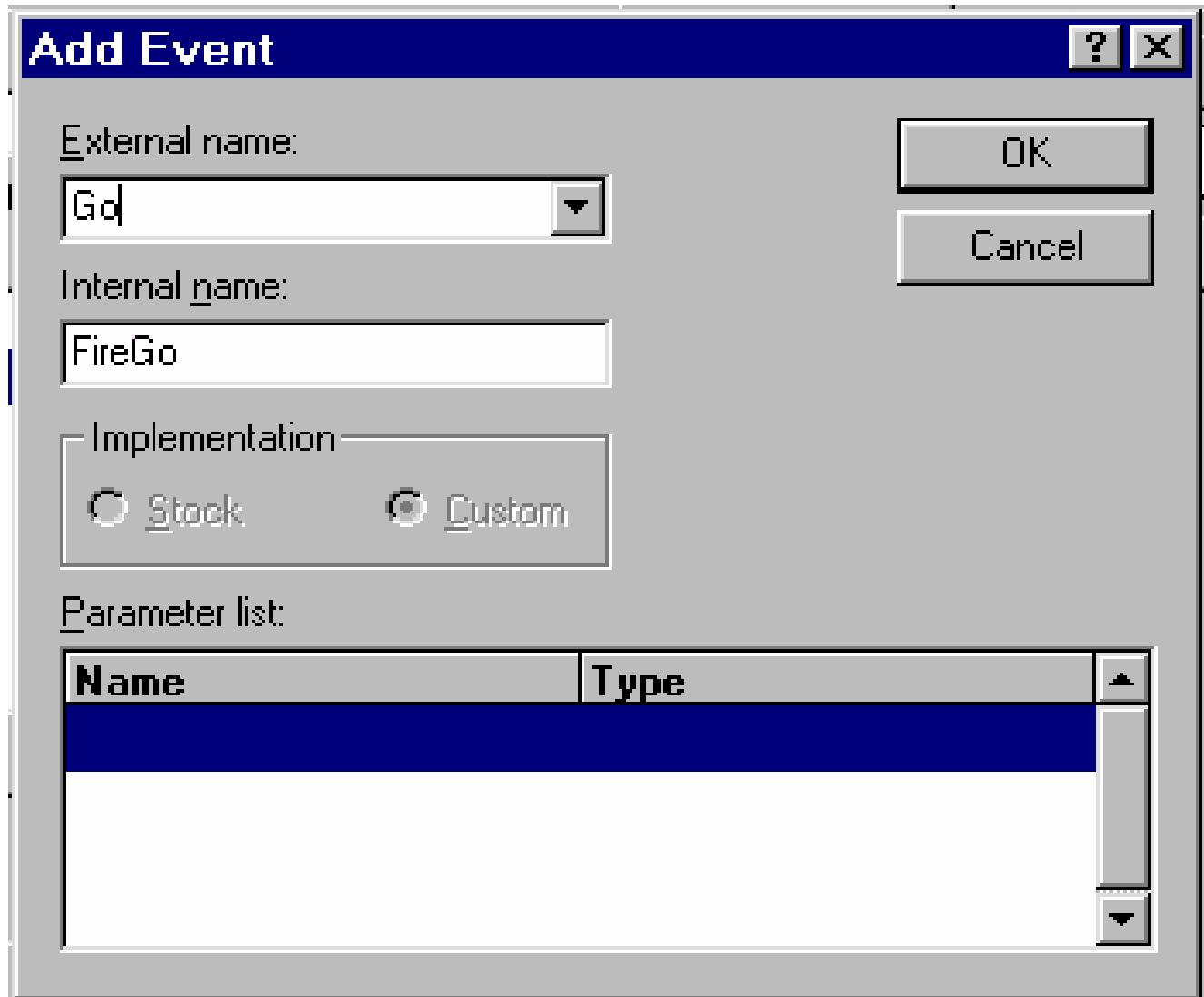
7.2 Tạo ActiveX bằng VC++

13. Cửa sổ "Add Event" cho phép thiết lập các thuộc tính của event : tên bên ngoài, tên bên trong (hàm tạo event), danh sách đối số, loại event : stock hay custom.



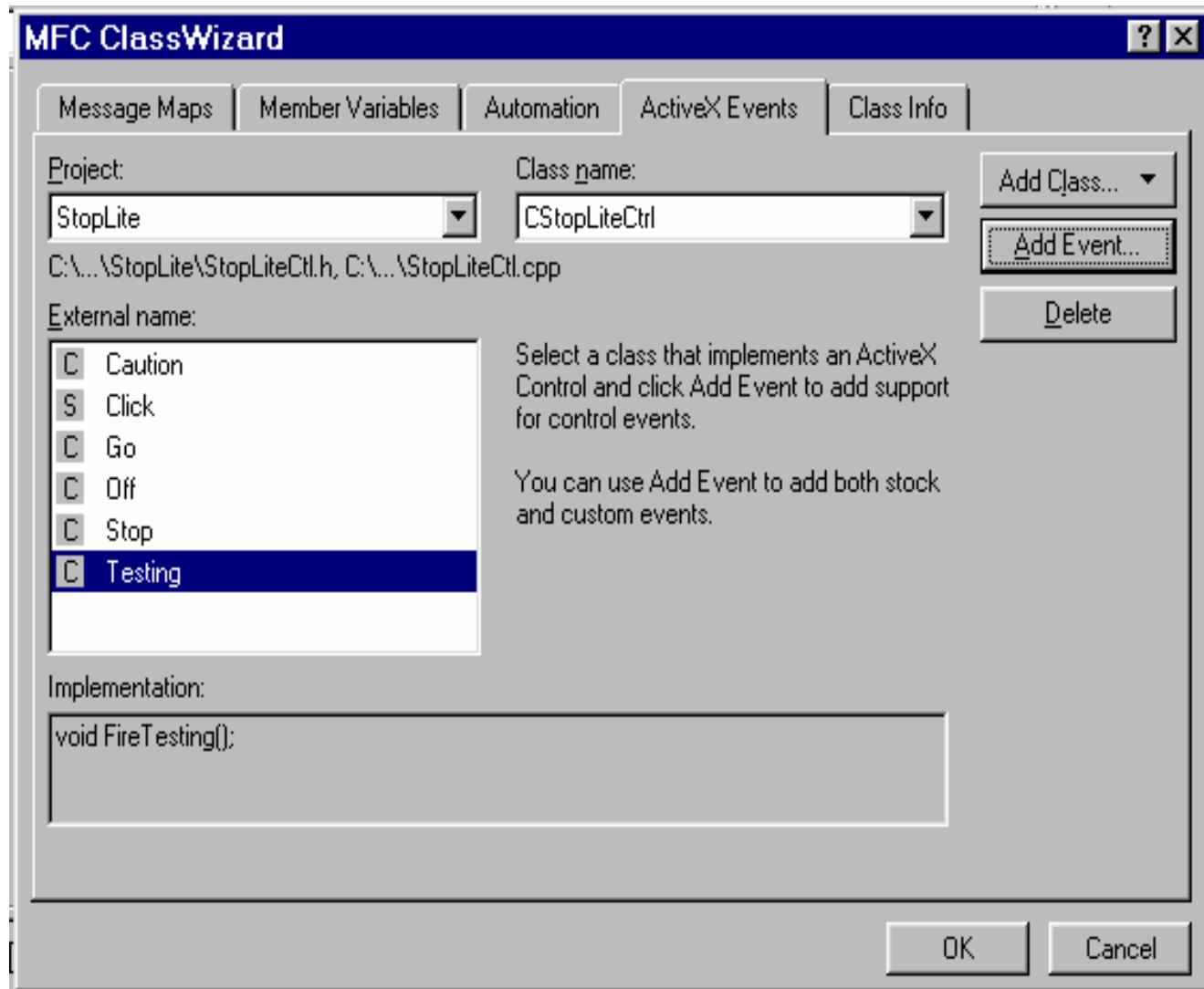
7.2 Tạo ActiveX bằng VC++

13. Cửa sổ "Add Event" cho phép thiết lập các thuộc tính của event : tên bên ngoài, tên bên trong (hàm tạo event), danh sách đối số, loại event : stock hay custom.



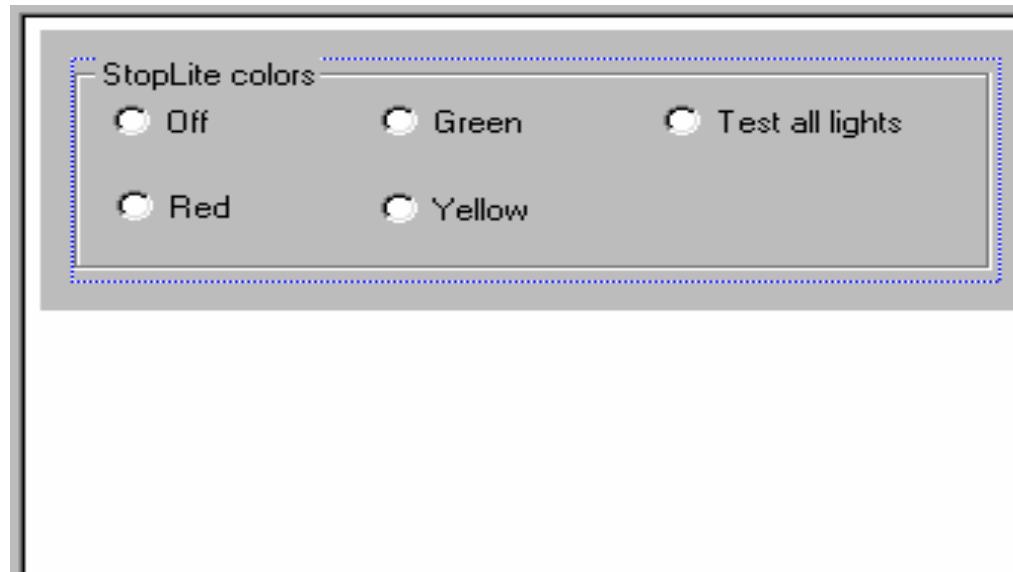
7.2 Tạo ActiveX bằng VC++

14. Sau khi đã tạo các event, chúng sẽ hiển thị trong danh sách "External name" trong cửa sổ Classwizard.



7.2 Tạo ActiveX bằng VC++

15. Tùy theo yêu cầu
thiết lập giá trị đầu của
các thuộc tính dữ liệu,
thiết kế trang Property
phù hợp cho ActiveX
Control..



7.2 Tạo ActiveX bằng VC++

16. Bước 5 : viết code cho các hàm xử lý biến cố và các method.

// Hiệu chỉnh lại hàm OnDraw của Activex Control để vẽ nó theo yêu cầu riêng của nó.

```
void CStopLiteCtrl::OnDraw( CDC* pdc, const CRect& rcBounds, const CRect& rcInvalid) {
```

// 1. xóa background của ActiveX Control dùng màu background của container.

```
CBrush brAmbientBack(TranslateColor(AmbientBackColor()));
```

```
pdc->FillRect(rcBounds, &brAmbientBack);
```

// 2. vẽ mép ActiveX Control dùng thuộc tính stock của control : BackColor + ForeColor

// tính kích thước khoảng 40% độ cao

```
CRect rcBezel(rcBounds);
```

```
int nheight = rcBounds.height();
```

```
int nwidth = rcBounds.width();
```



7.2 Tạo ActiveX bằng VC++

```
if (nheight >= nwidth) {  
int nBezelwidth = nheight * 40 / 100;  
if (nBezelwidth > nwidth)  
nBezelwidth = nwidth; // not more than width!  
int nDeflateBezel = (nwidth - nBezelwidth) / 2;  
    rcBezel.DeflateRect(nDeflateBezel, 0);  
} else {  
int nBezelheight = nwidth * 40 / 100;  
    if (nBezelheight > nheight)  
nBezelheight = nheight; // not more than width!  
int nDeflateBezel = (nheight - nBezelheight) / 2;  
rcBezel.DeflateRect(0,nDeflateBezel);  
}  
// create and select brush and pen  
CBrush brBack(TranslateColor(GetBackColor()));  
CBrush * pbrOld = pdc->SelectObject(&brBack);  
CPen pnFore(PS_SOLID, 2, TranslateColor(GetForeColor()));
```

7.2 Tạo ActiveX bằng VC++

```
CPen * ppnOld = pdc->SelectObject(&pnFore);
pdc->Rectangle(rcBezel); // draw
// select old brush, but not old pen
pdc->SelectObject(pbrOld);
// 3. vẽ 3 đèn dùng thuộc tính stock ForeColor.
// already selected translate enum code to bits for red, green, yellow
int nLights = TranslateLights();
// percentages are percentage of height
// draw red light on top, 6% down, 27% diameter
if (nheight >= nwidth) {
    DrawLight(pdc, rcBounds, 6, 27,
(nLights & SLBIT_RED) ? SLCOLOR_RED : SLCOLOR_OFF);
    // yellow light in middle, 37% down, 27% diameter
    DrawLight(pdc, rcBounds, 37, 27,
(nLights & SLBIT_YELLOW) ? SLCOLOR_YELLOW : SLCOLOR_OFF);
    // green light on bottom, 68% down, 27% diameter
    DrawLight(pdc, rcBounds, 68, 27,
(nLights & SLBIT_GREEN) ? SLCOLOR_GREEN : SLCOLOR_OFF);
```

7.2 Tạo ActiveX bằng VC++

```
} else { // Hien ngang
    DrawLight(pdc, rcBounds, 6, 27,
(nLights & SLBIT_RED) ? SLCOLOR_RED : SLCOLOR_OFF);
    // yellow light in middle, 37% down, 27% diameter
DrawLight(pdc, rcBounds, 37, 27,
    (nLights & SLBIT_YELLOW) ? SLCOLOR_YELLOW : SLCOLOR_OFF);
// green light on bottom, 68% down, 27% diameter
DrawLight(pdc, rcBounds, 68, 27,
(nLights & SLBIT_GREEN) ? SLCOLOR_GREEN : SLCOLOR_OFF);
}
pdc->SelectObject(ppnOld);
}
```



7.2 Tạo ActiveX bằng VC++

```
void CStopLiteCtrl::FireRightEvent() {  
// called whenever the stoplight state changes to fire the  
// appropriate event—must call AFTER m_color set to new value!  
// Use the source browser to make sure you call each time  
// m_color changed!  
    switch (m_color) {  
        case SL_RED: FireStop();  
        break;  
        case SL_YELLOW: FireCaution();  
        break;  
        case SL_GREEN: FireGo();  
        break;  
        case SL_NONE: FireOff();  
        break;  
        case SL_TEST: FireTesting();  
        break;  
    }  
}
```



7.2 Tạo ActiveX bằng VC++

```
int CStopLiteCtrl::TranslateLights() {  
// sets appropriate bits for stoplight state  
int nLights = SLBIT_RED; // safe default  
switch (m_color) {  
case SL_NONE: nLights = 0;  
break;  
case SL_RED: nLights = SLBIT_RED;  
break;  
case SL_GREEN: nLights = SLBIT_GREEN;  
break;  
case SL_YELLOW: nLights = SLBIT_YELLOW;  
break;  
case SL_TEST: nLights = SLBIT_RED | SLBIT_YELLOW | SLBIT_GREEN;  
break;  
}  
return nLights;  
}
```



7.2 Tạo ActiveX bằng VC++

```
void CStopLiteCtrl::DrawLight (
// draws an individual light centered in the control at
// vertical position specified
CDC* pdc, // DC in which to draw
const CRect& rcBounds, // control's rectangle
int nPercentDown, // top position as % of height
int nPercentDiameter, // diameter as % of height
COLORREF crColor // color to fill light) {
// calculate diameter in drawing units
int nheight = rcBounds.height();
int nwidth = rcBounds.width();
if (nwidth <nheight) { // doc
    int nDiameter = nheight * nPercentDiameter / 100;
    if (nDiameter > nwidth)
        nDiameter = nwidth; // but not greater than width!
    // create light's bounding rect
    int nLeftEdge = (rcBounds.left + rcBounds.right - nDiameter) / 2;
    int nTopEdge = rcBounds.top + nheight * nPercentDown / 100;
```

7.2 Tạo ActiveX bằng VC++

```
CRect rcLight( nLeftEdge, nTopEdge, nLeftEdge + nDiameter,  
nTopEdge + nDiameter);  
    // make absolutely sure we're within bounds—  
    // distort circle if necessary!  
    rcLight.IntersectRect(rcLight, rcBounds);  
    // create brush, draw, select old brush  
    CBrush brColor(crColor);  
    CBrush * brOld = pdc->SelectObject(&brColor);  
    pdc->Ellipse(rcLight);  
    pdc->SelectObject(brOld);  
} else { // ngang  
    int nDiameter = nwidth * nPercentDiameter / 100;  
    if (nDiameter > nheight)  
        nDiameter = nheight; // but not greater than height!  
    // create light's bounding rect  
    int nTopEdge = (rcBounds.top + rcBounds.bottom - nDiameter) / 2;  
    int nLeftEdge = rcBounds.left + nwidth * nPercentDown / 100;
```

7.2 Tạo ActiveX bằng VC++

```
CRect rcLight( nLeftEdge, nTopEdge, nLeftEdge + nDiameter,  
nTopEdge + nDiameter);
```

```
    // make absolutely sure we're within bounds—
```

```
    // distort circle if necessary!
```

```
    rcLight.IntersectRect(rcLight, rcBounds);
```

```
    // create brush, draw, select old brush
```

```
    CBrush brColor(crColor);
```

```
    CBrush * brOld = pdc->SelectObject(&brColor);
```

```
    pdc->Ellipse(rcLight);
```

```
    pdc->SelectObject(brOld);
```

```
}
```

```
}
```

```
////////////////////////////////////////////////////////////////////////
```

```
// CStopLiteCtrl message handlers
```

```
void CStopLiteCtrl::OnLButtonDown(UINT nFlags, CPoint point) {
```

```
    // TODO: Add your message handler code here and/or call default  
    Next();
```

```
    COleControl::OnLButtonDown(nFlags, point);
```

```
}
```



7.2 Tạo ActiveX bằng VC++

```
short CStopLiteCtrl::GetColor() {
    // TODO: Add your property handler here
    return m_color;
}

void CStopLiteCtrl::SetColor(short nnewValue) {
    // TODO: Add your property handler here
    if (nnewValue >= SL_NONE && nnewValue <= SL_TEST) {
        m_color = nnewValue;
        InvalidateControl();
        FireRightEvent();
        SetModifiedFlag();
    } else {
        ThrowError(CTL_E_ILLEGALFUNCTIONCALL,
        "Color parameter out of range");
    }
}
```



7.2 Tạo ActiveX bằng VC++

```
void CStopLiteCtrl::Next() {  
    // TODO: Add your dispatch handler code here  
    if (m_color >= SL_LAST || m_color < SL_FIRST) {  
        m_color = SL_FIRST;  
    } else m_color++;  
    InvalidateControl();  
    FireRightEvent();  
    SetModifiedFlag();  
}  
  
void CStopLiteCtrl::OnAmbientPropertyChange(DISPID dispid) {  
    // TODO: Add your specialized code here and/or call the base class  
    // Repaint if ambient background changed or if several changed  
    if (dispid == DISPID_AMBIENT_BACKCOLOR ||  
        dispid == DISPID_UNKNOWN)  
    {  
        InvalidateControl();  
    }  
    // pass on to base class  
    COleControl::OnAmbientPropertyChange(dispid);  
}
```

7.2 Tạo ActiveX bằng VC++

17. Bước 6 : dịch và debug. Sau khi viết xong dùng menu Build.Rebuild all để dịch và tạo ActiveX Control, nếu có lỗi thì sửa. Cuối cùng dùng menu Tools.ActiveX Control Test Container để debug ActiveX Control.

Lưu ý sau khi dịch thành công ActiveX, VC++ đăng ký tự động ActiveX vào Windows, do đó ActiveX sẵn sàng được dùng bởi bất kỳ ứng dụng nào trên Windows.

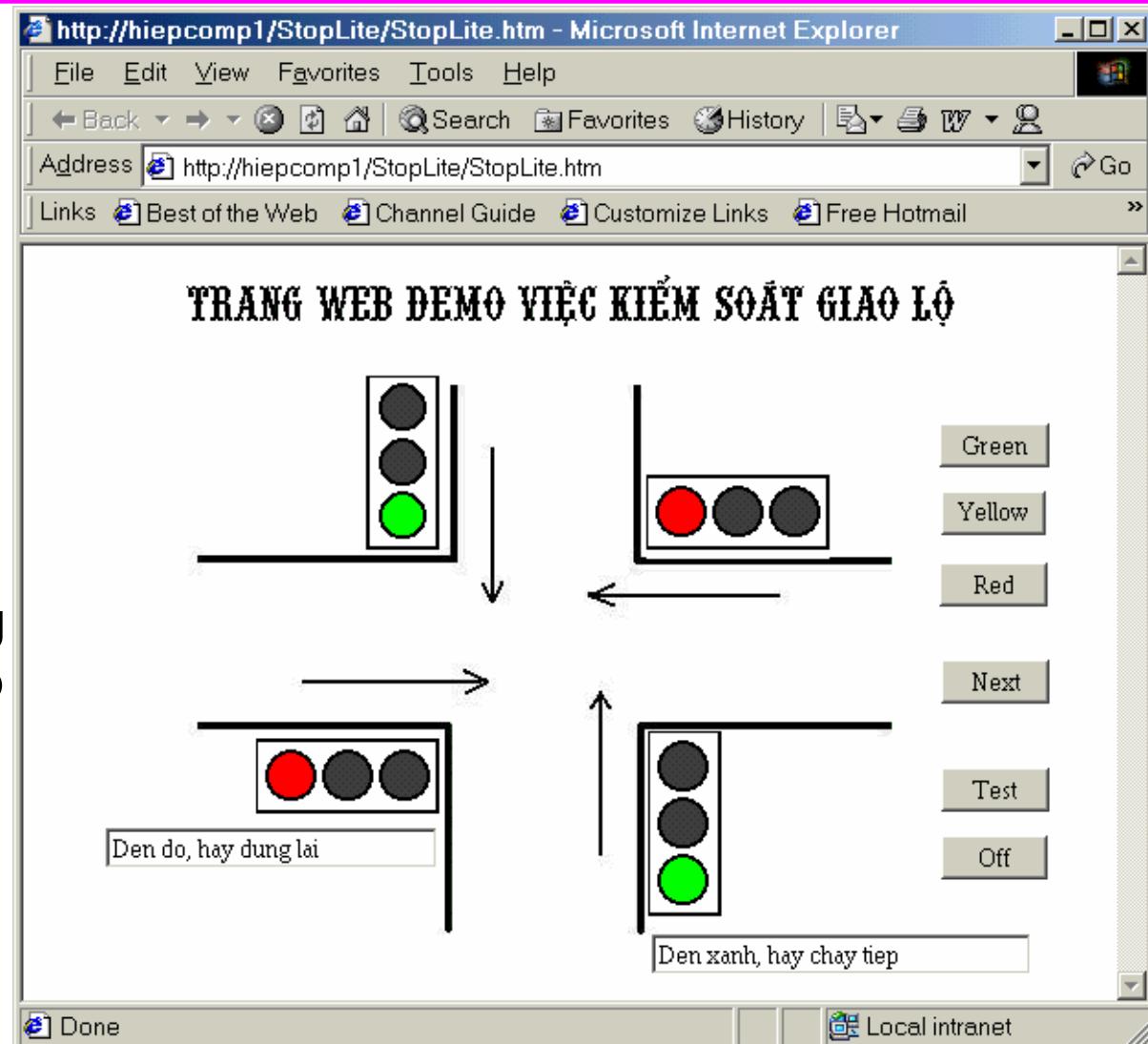
Khi copy ActiveX qua máy khác, bạn phải đăng ký ActiveX vào Windows nhờ tiện ích regsvr32.exe (trong thư mục windows\system32)



7.3 Tạo Website dùng ActiveX

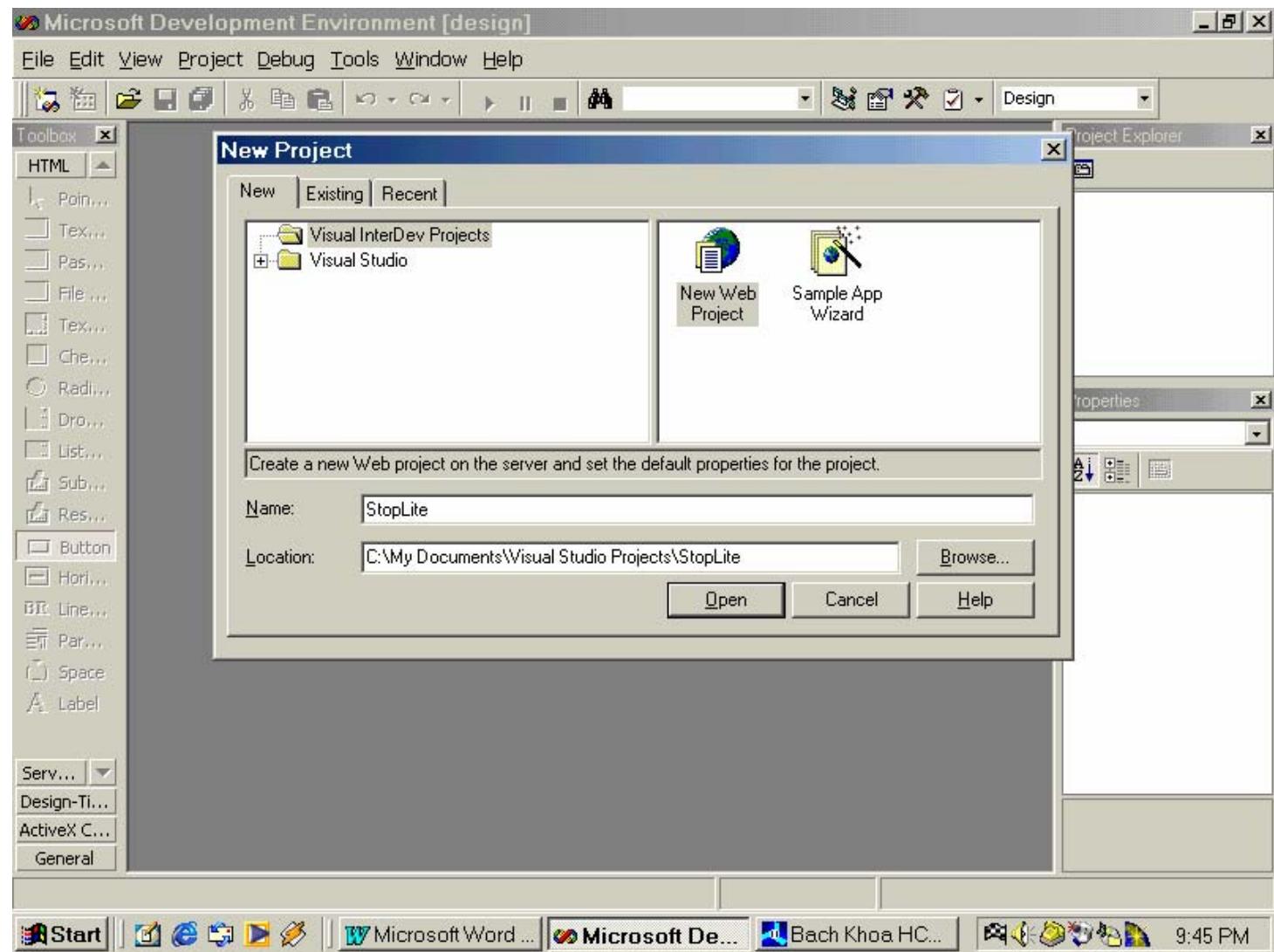
Chúng ta sẽ xây dựng 1 trang Web demo việc điều khiển ở 1 ngã tư (giao lộ của con đường), trang Web này sẽ dùng 4 instance MyStopLite để làm 4 dàn đèn điều khiển. Giao diện trang Web có dạng bên :

Các dàn đèn tự hiệu chỉnh trạng thái theo timer 5s, user (đóng vai trò công an) có thể thay đổi trạng thái điều khiển bằng cách ấn vào các button Green, Yellow, Red, Next, Test, Off. Khi các đèn chuyển trạng thái, 2 textbox kết hợp với 2 đèn L1, L2 sẽ hiển thị cảnh báo.



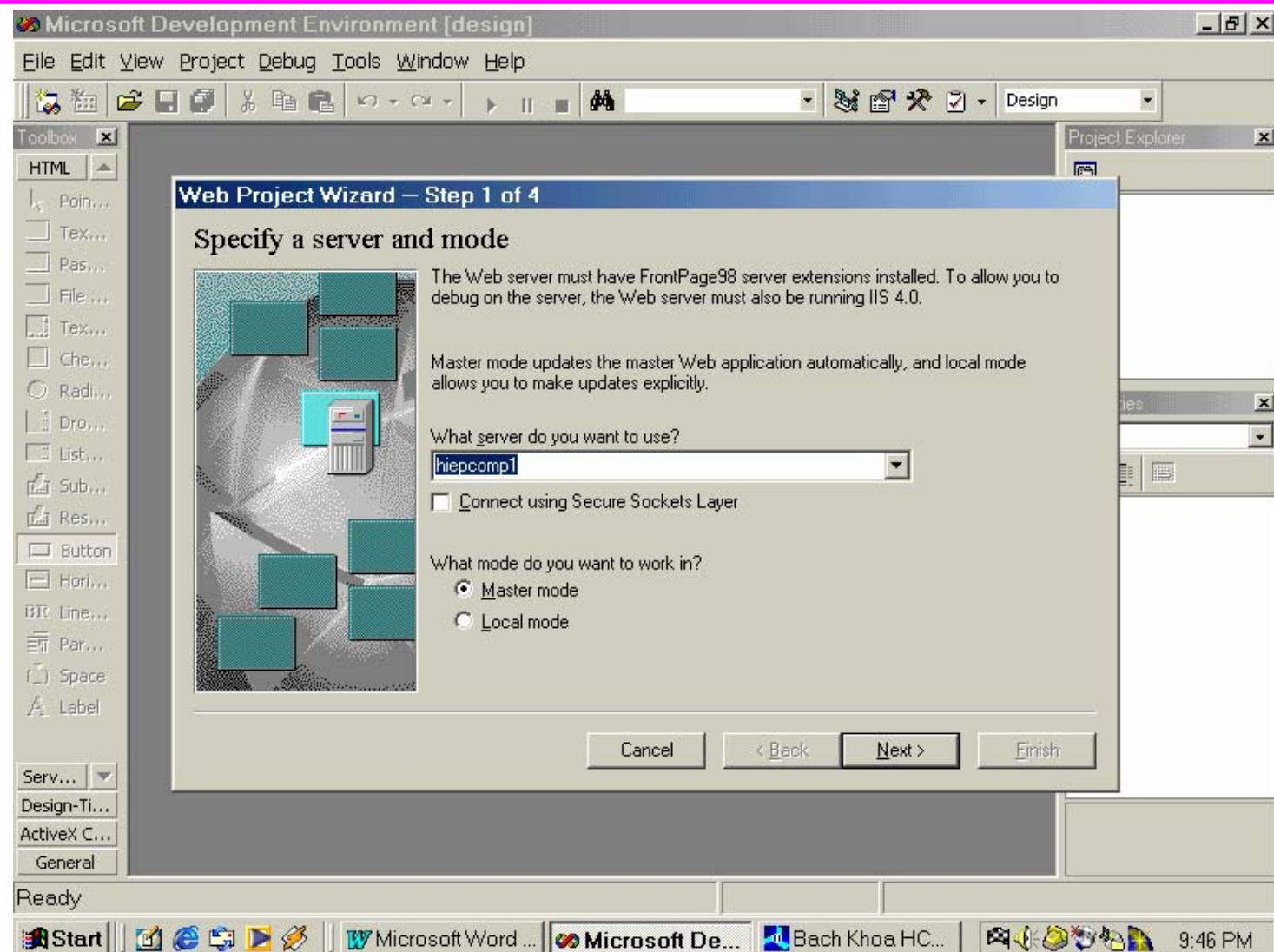
7.3 Tạo Website dùng ActiveX

1. Chạy InterDev, cửa sổ sau sẽ hiển thị.
Chọn page New, mục "Visual InterDev Projects", chọn icon "New Web project", chọn vị trí thư mục chứa Project, nhập tên project (trở thành thư mục chứa các file của project), chọn Ok.



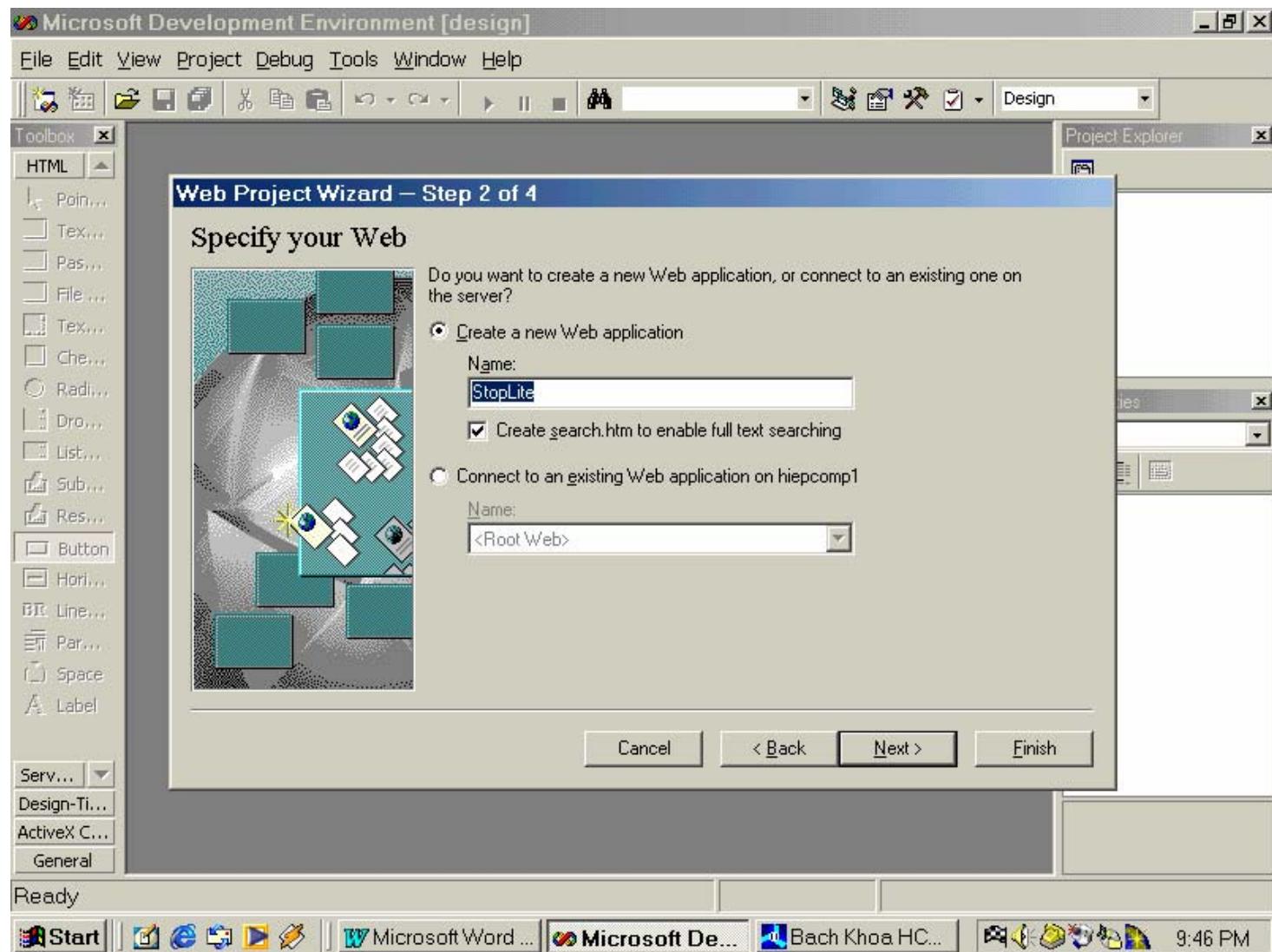
7.3 Tạo Website dùng ActiveX

2. Trong bước 1 của Wizard, chọn hay nhập tên Web server mà project Web sẽ được cài đặt lên rồi chọn Next.



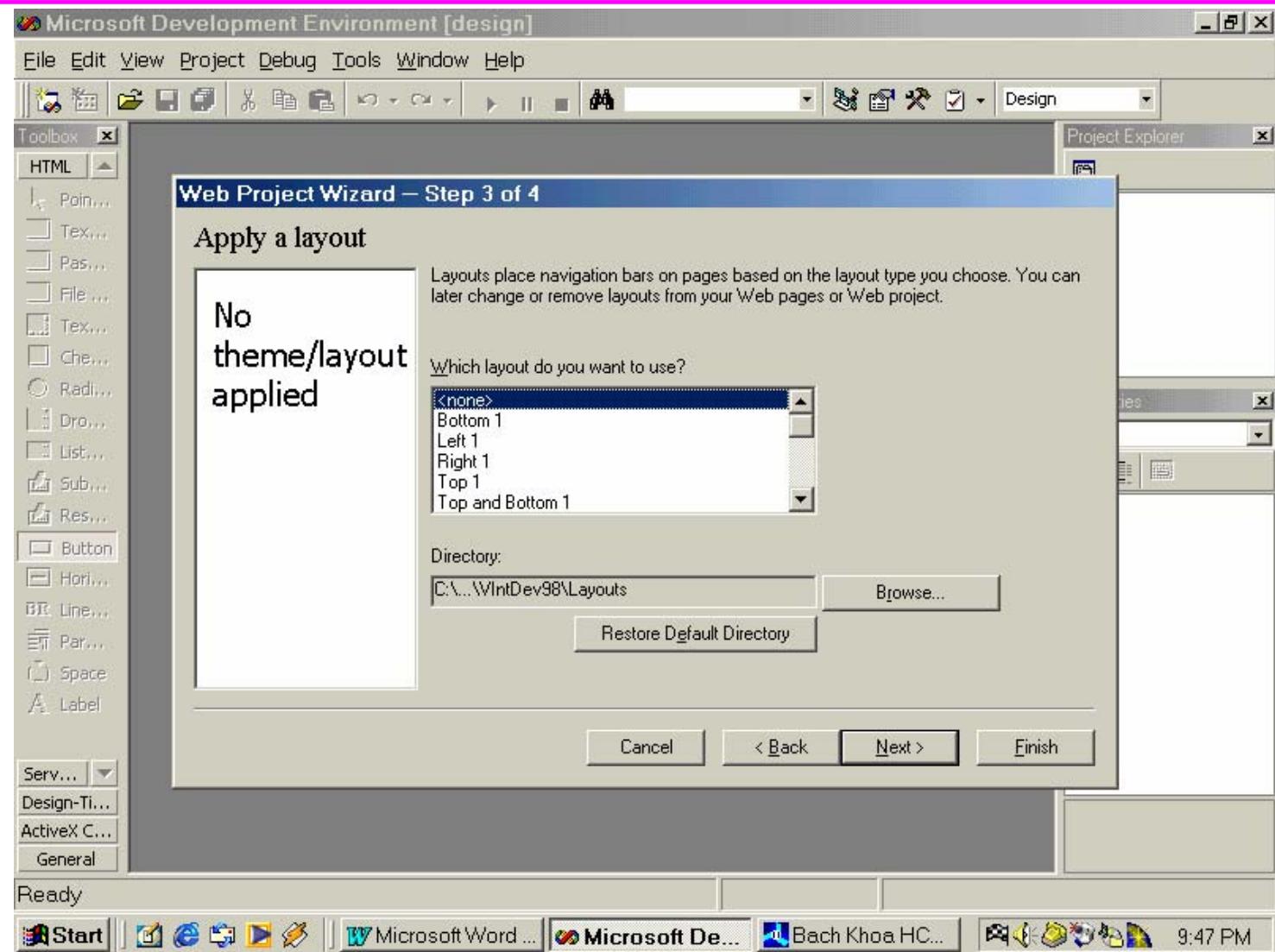
7.3 Tạo Website dùng ActiveX

3. Trong bước 2 của Wizard, nhập tên ứng dụng rồi chọn Next.



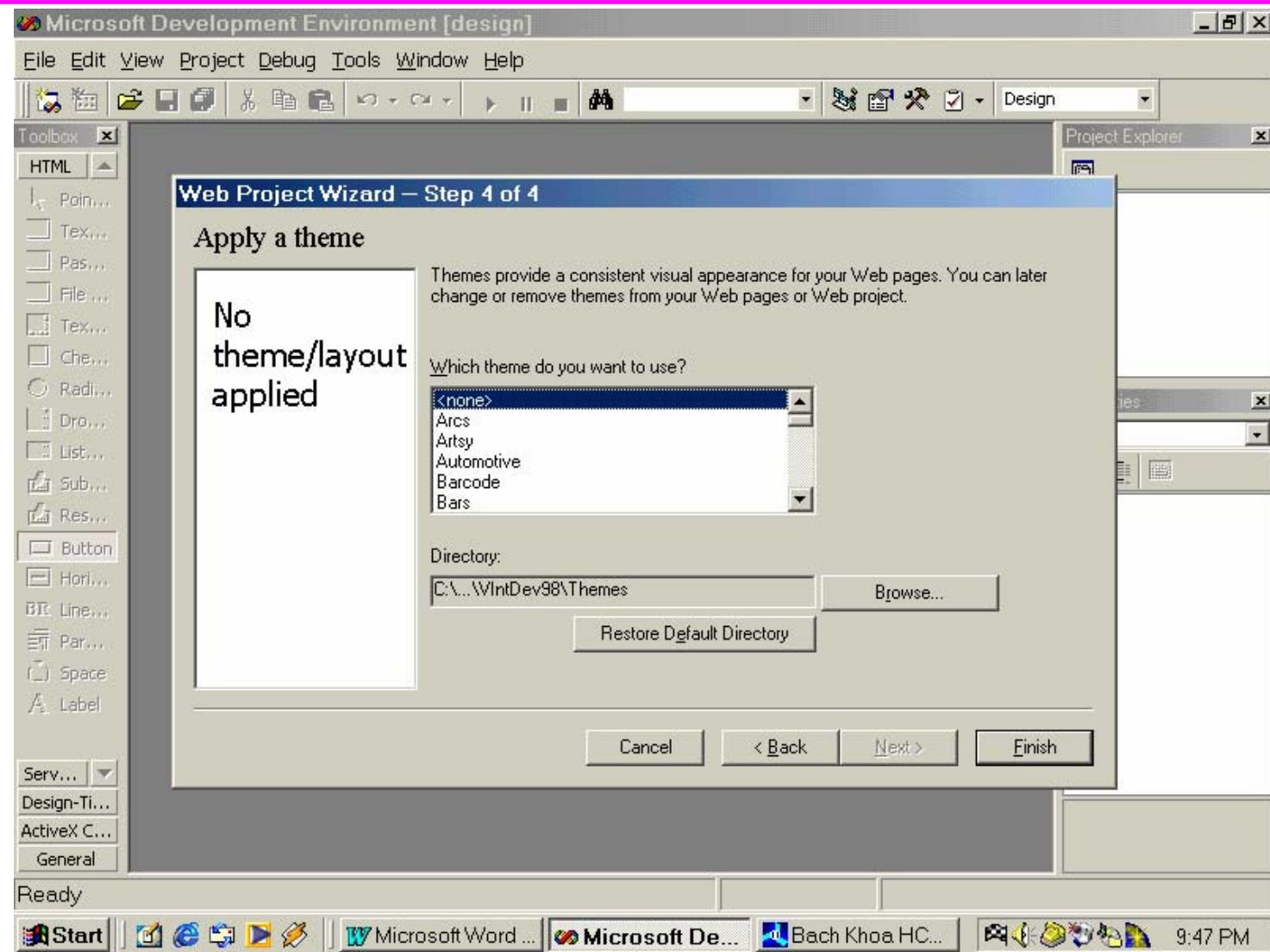
7.3 Tạo Website dùng ActiveX

4. Trong bước 3 của Wizard, chọn mẫu layout (td. <none>) cho project rồi chọn Next.



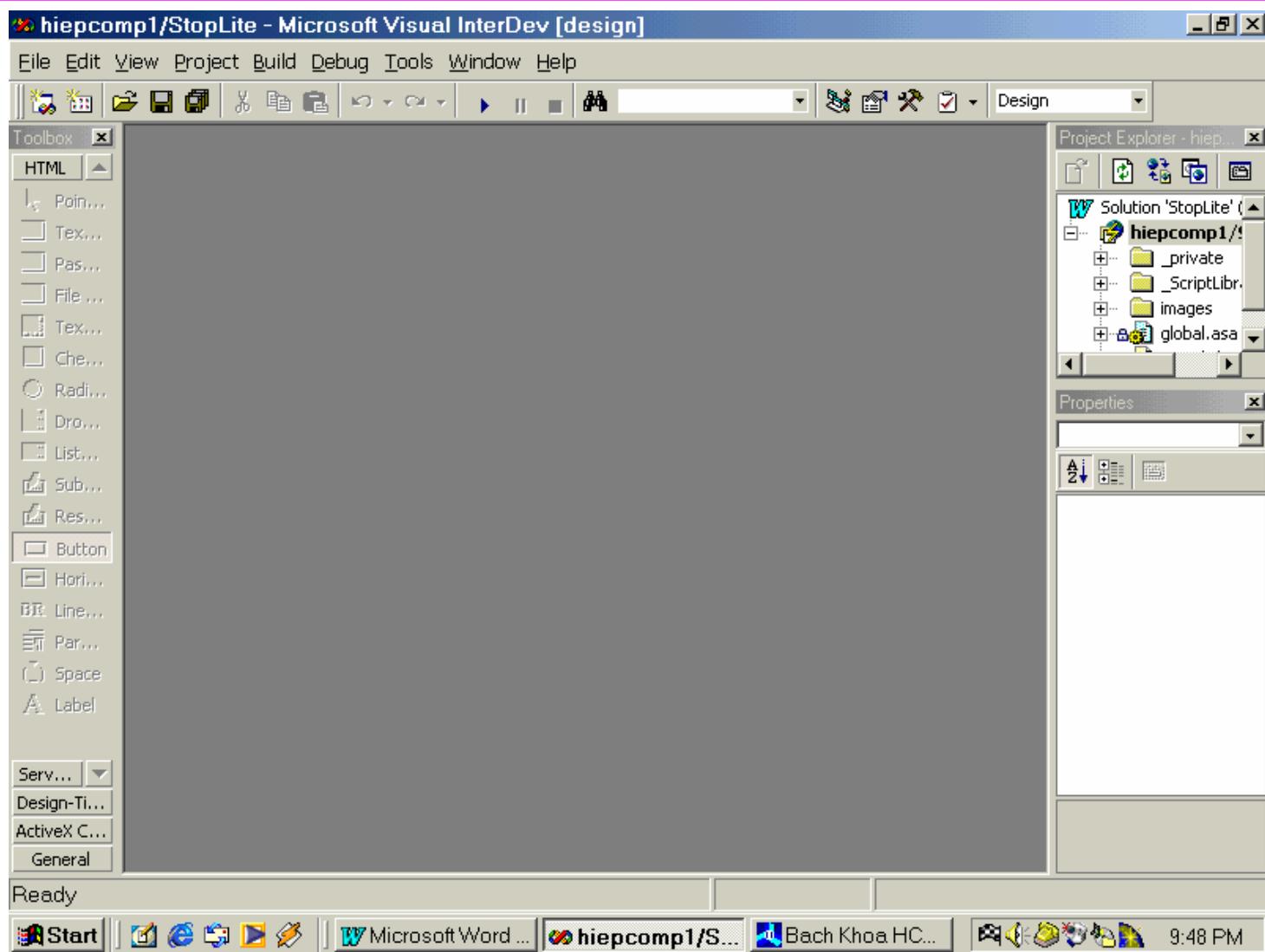
7.3 Tạo Website dùng ActiveX

5. Trong bước 4 của Wizard, chọn mẫu theme (td. <none>) cho project rồi chọnFinish.



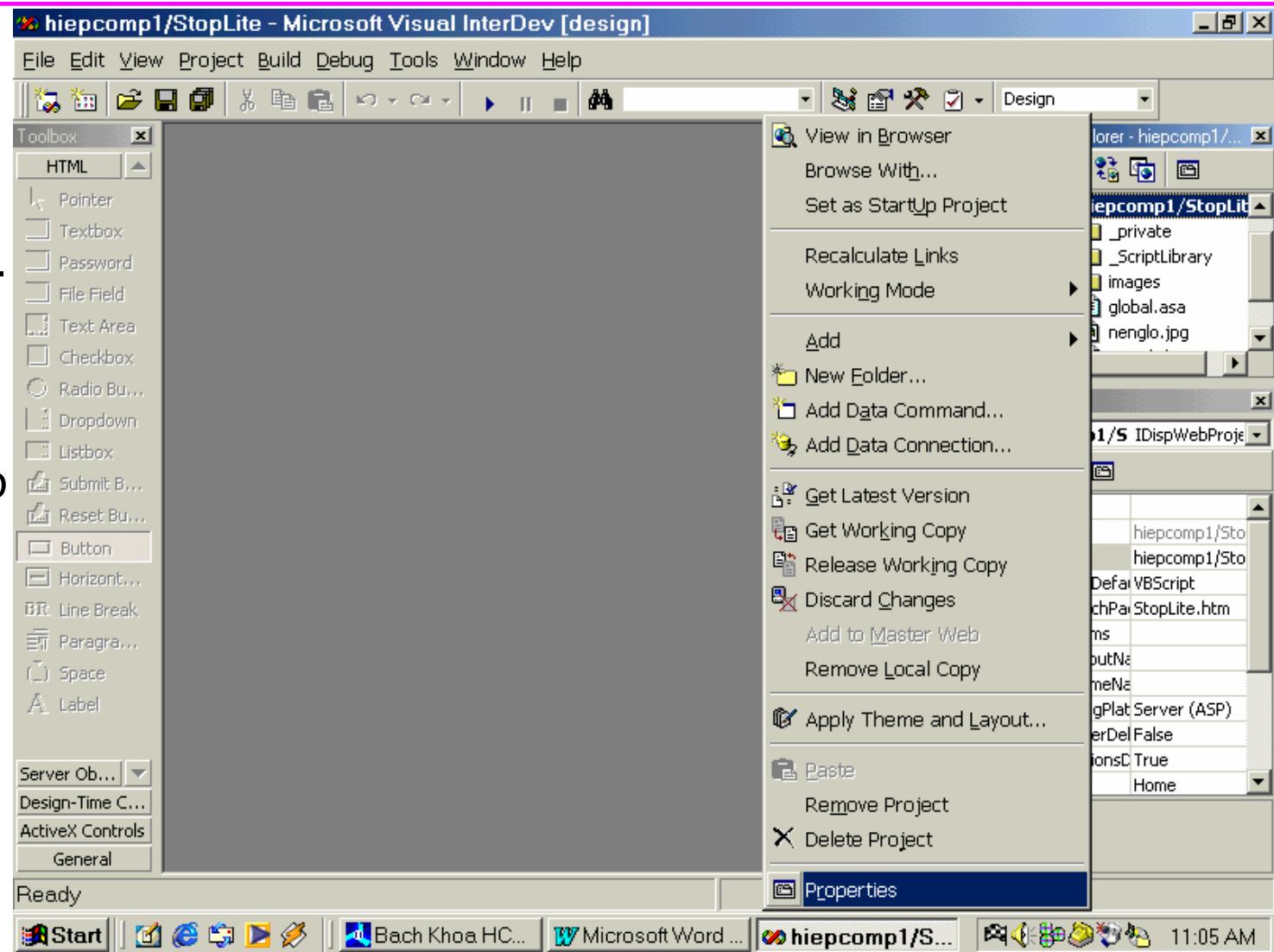
7.3 Tạo Website dùng ActiveX

6. Cửa sổ ban đầu
của project Web vừa
được tạo ra.



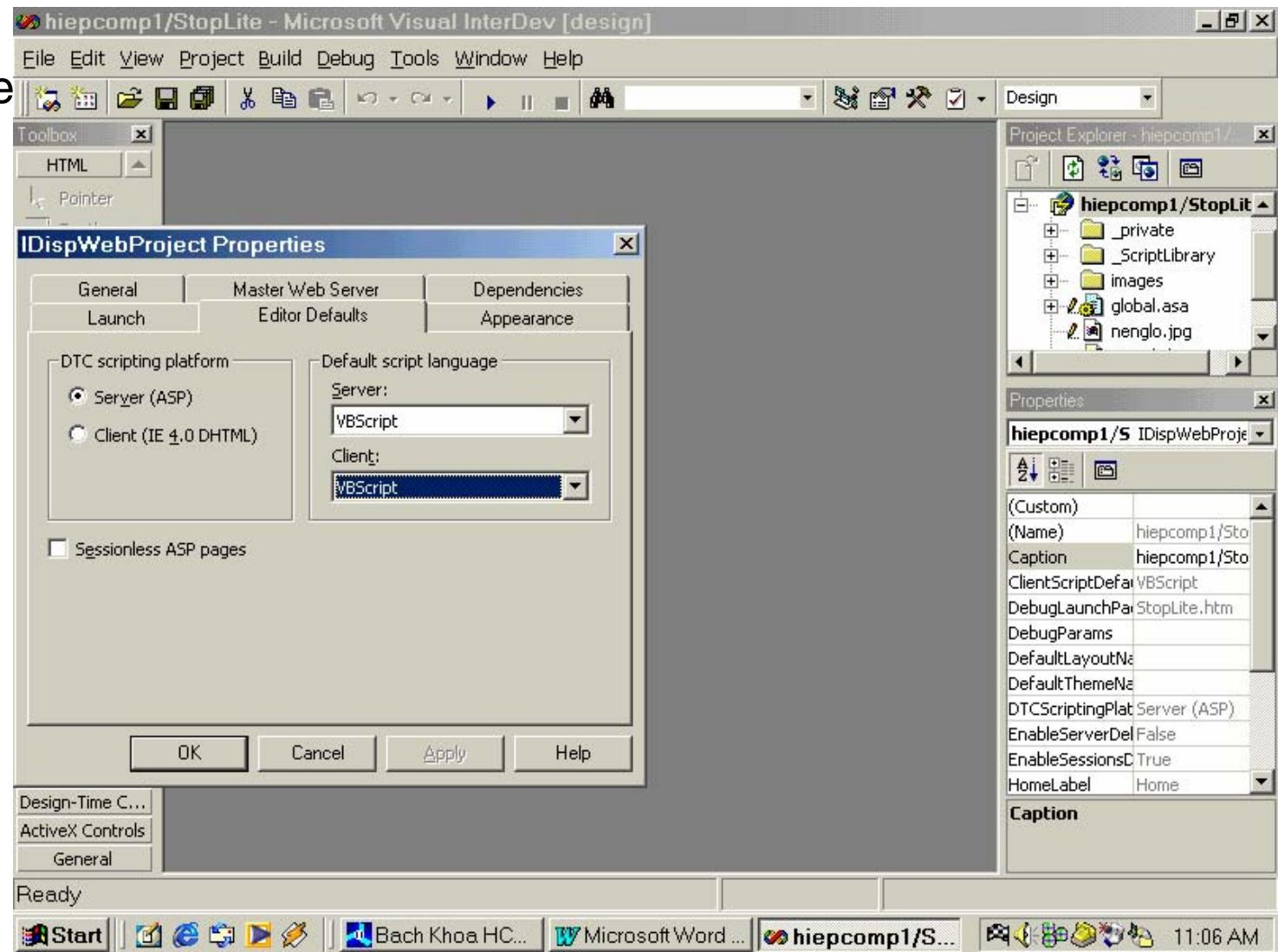
7.3 Tạo Website dùng ActiveX

7. Giả sử ta muốn dùng VBscript cho các trang Web trong Project, ta phải khai báo cho InterDev biết. Ấn phải chuột trên mục Project (cửa sổ chứa cây project trên phải) để menu pop-up hiển thị rồi chọn mục Properties :



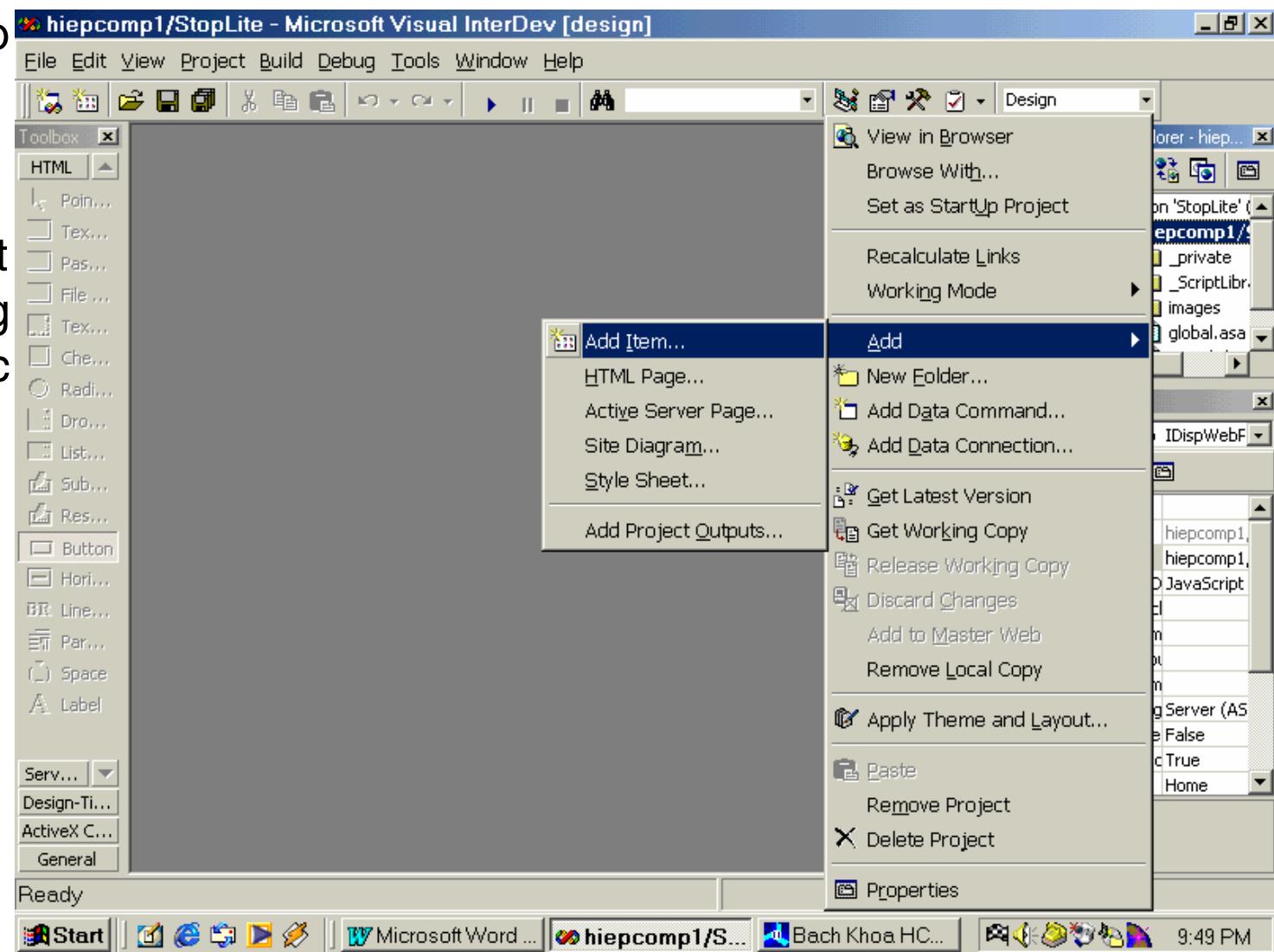
7.3 Tạo Website dùng ActiveX

8. Trong cửa sổ Properties, chọn page "Editor Defaults", chọn mục VBScript trong listbox "Client", chọn button Apply rồi Ok :



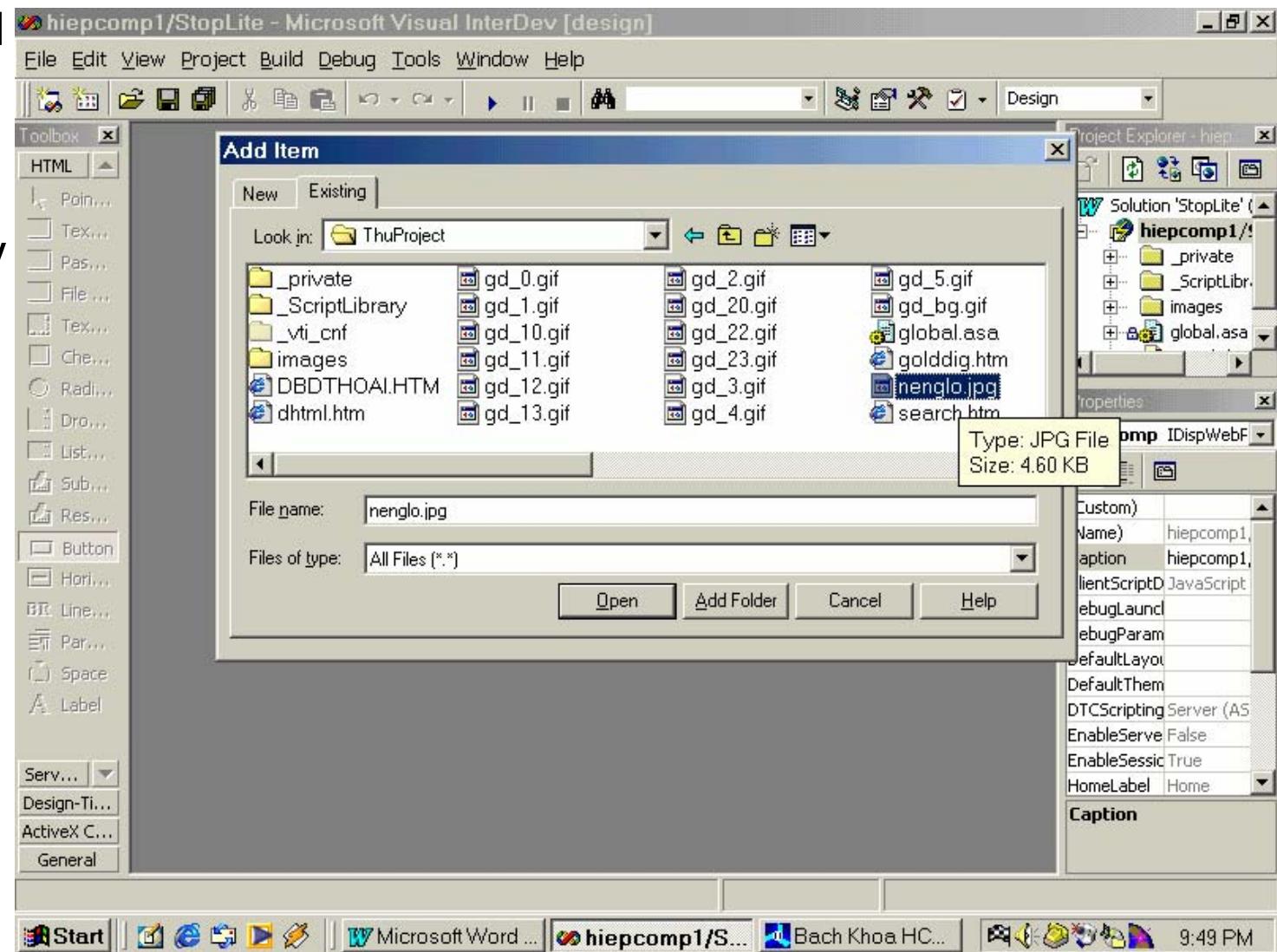
7.3 Tạo Website dùng ActiveX

9. Vì trang Web demo có dùng 1 đồ họa giao lô, ta nên 'add' file đồ họa này vào project. Ấn phải chuột vào mục Project trong cây project, chọn mục "Add" rồi "Add Item" :



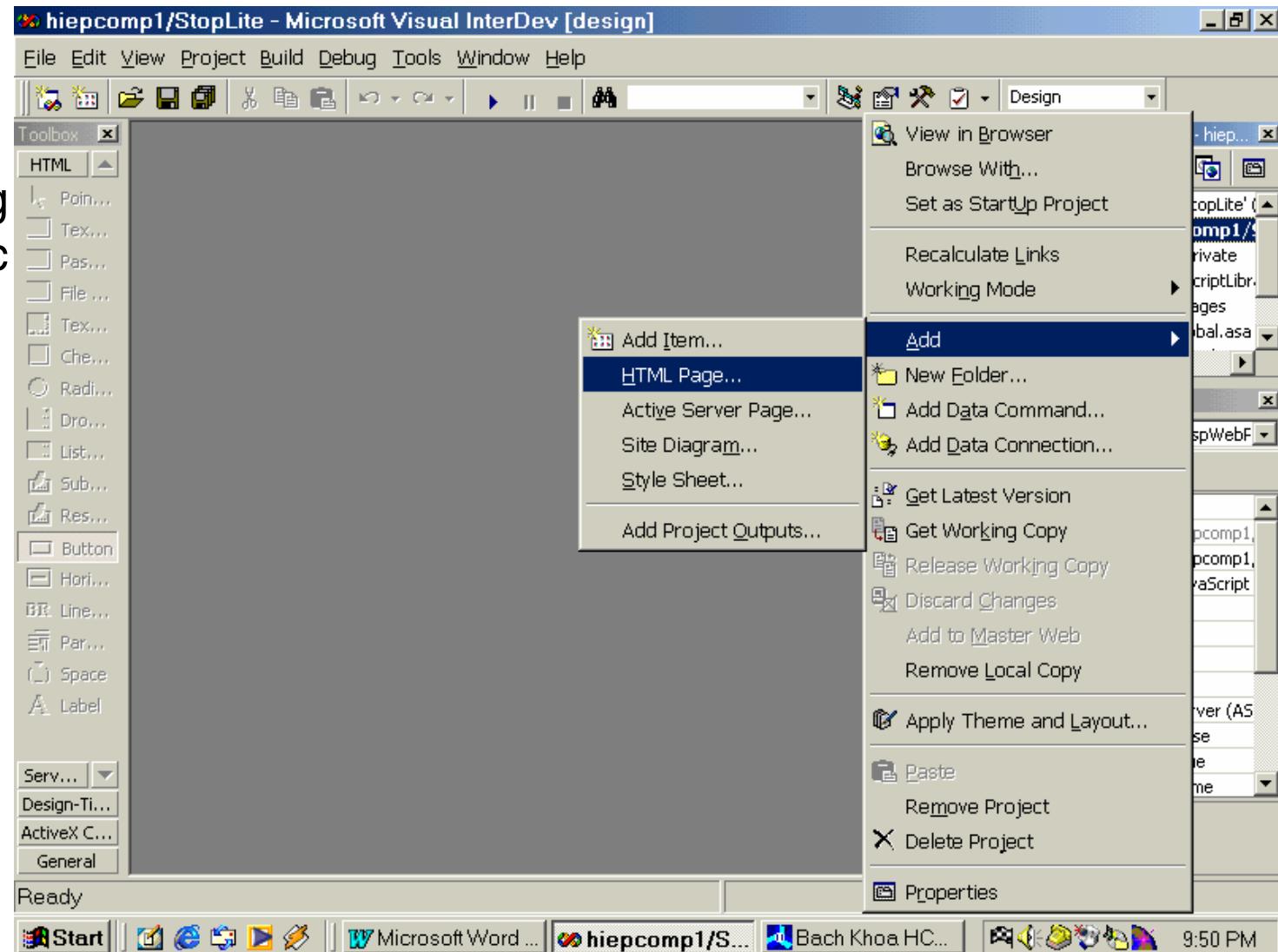
7.3 Tạo Website dùng ActiveX

10. Trong cửa sổ Add Item, duyệt cây thư mục và chọn file đồ họa rồi ấn button Open để 'add' file này vào projectWeb :



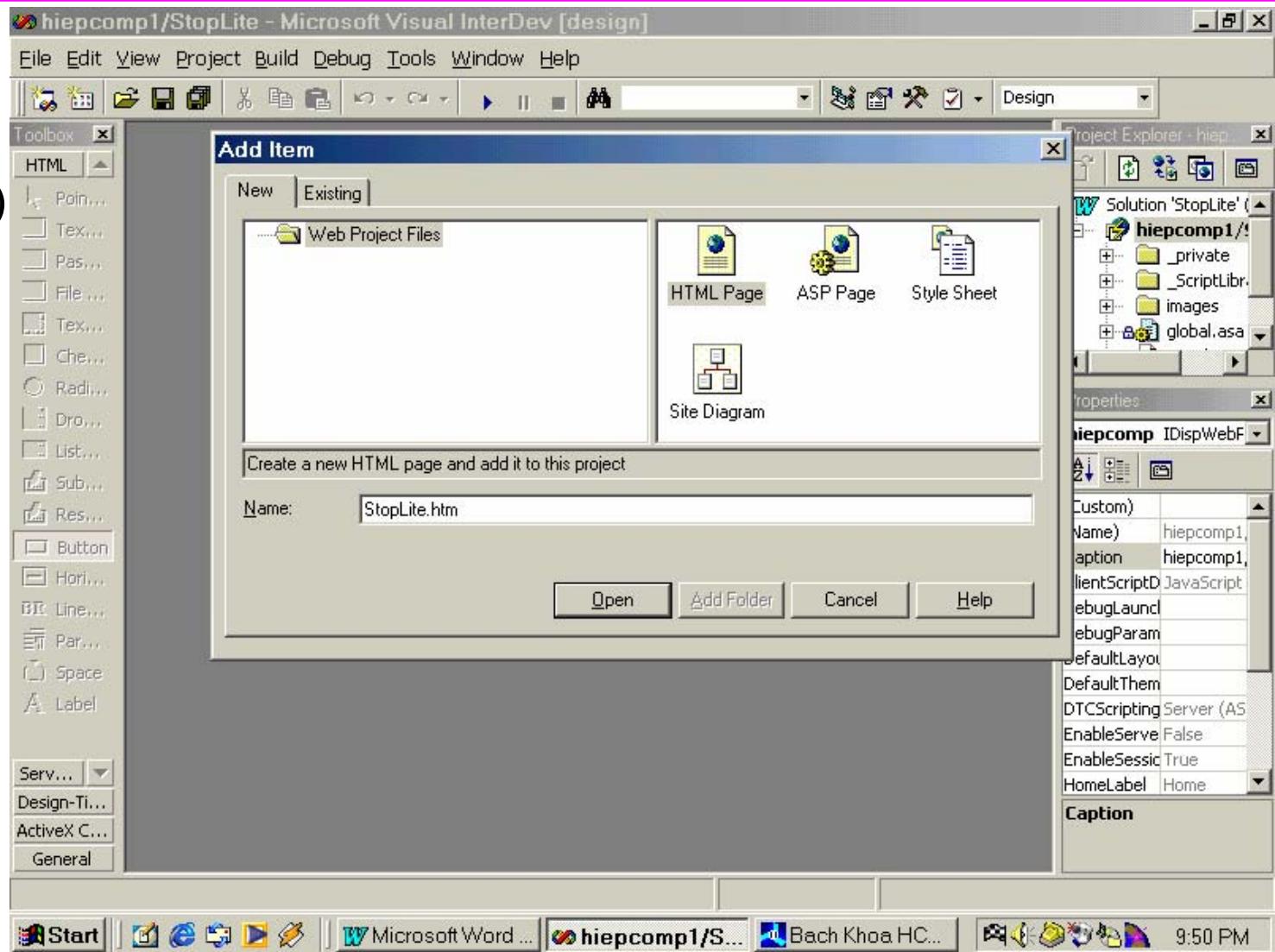
7.3 Tạo Website dùng ActiveX

11. Bây giờ tạo 1 trang Web mới bằng cách ấn phải chuột vào mục Project trong cây project, chọn mục "Add" rồi "HTML Page" :



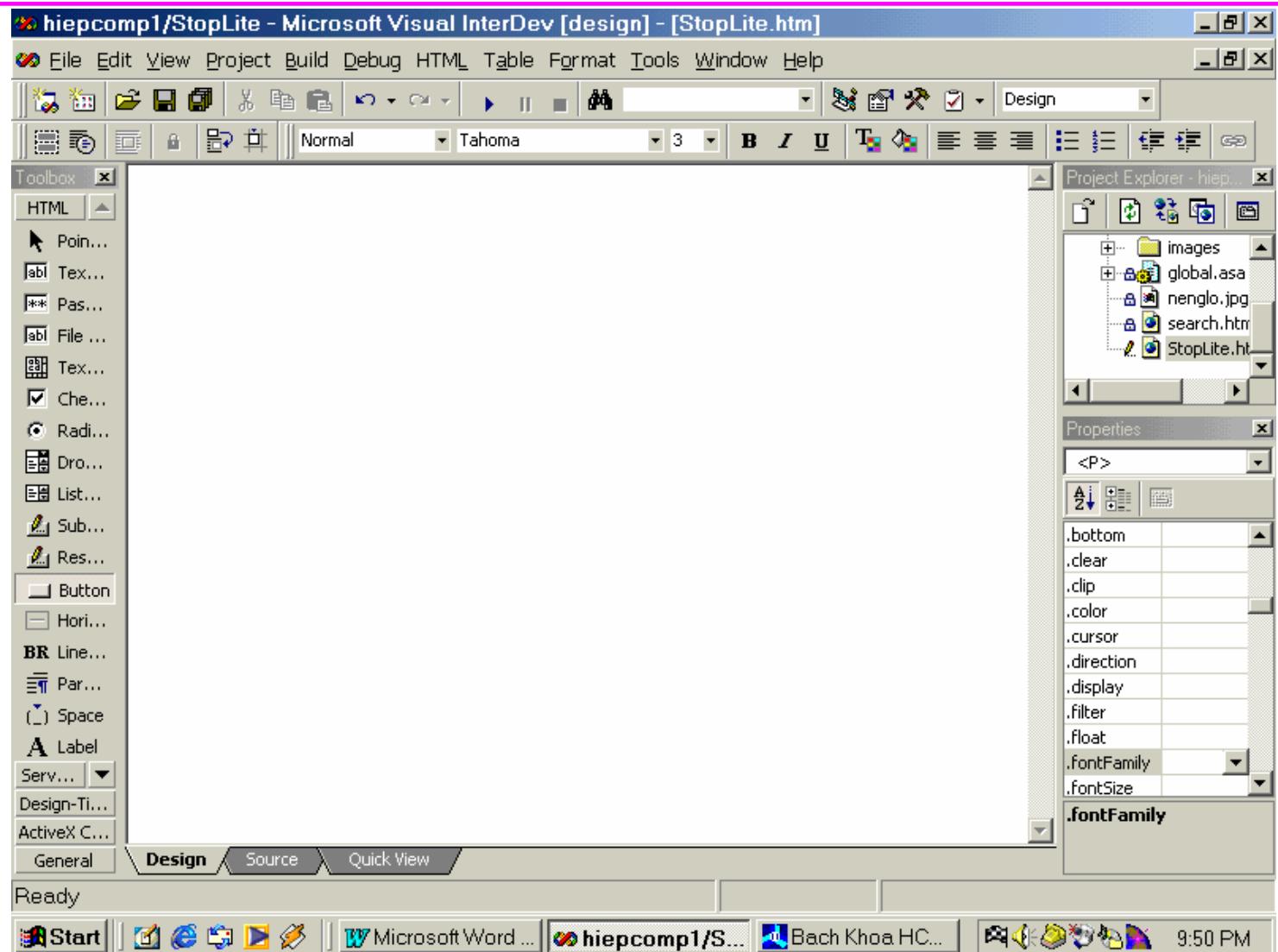
7.3 Tạo Website dùng ActiveX

12. Trong cửa sổ "Add Item", nhập tên trang Web cần tạo (thí dụ : StopLite.htm) rồi chọn button Ok:



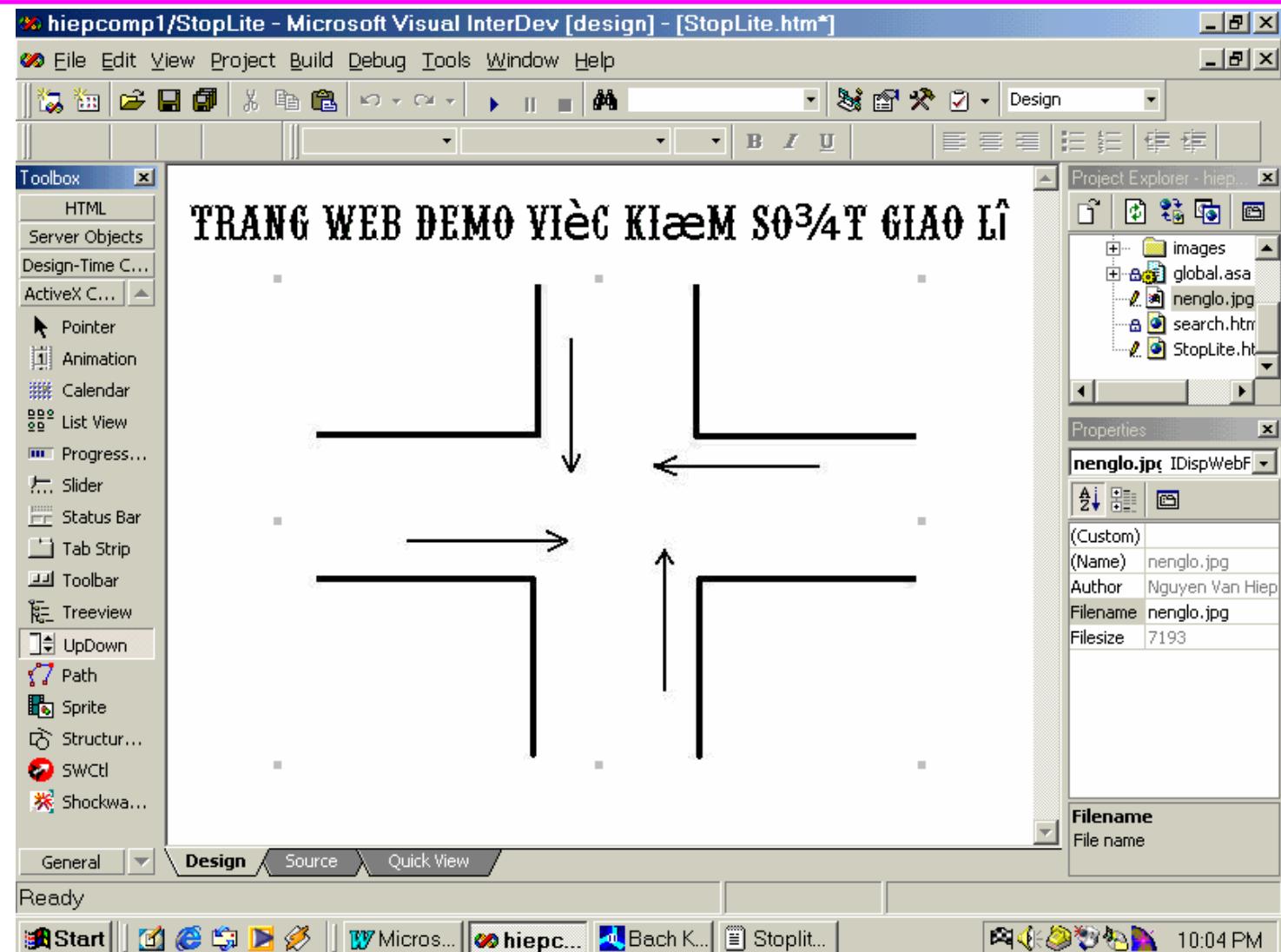
7.3 Tạo Website dùng ActiveX

13. Cửa sổ làm việc của trang web mới hoàn toàn trống. Hãy nhập dòng tiêu đề chính của trang Web (và những dòng text nội dung khác nếu muốn), format văn bản giống như word :



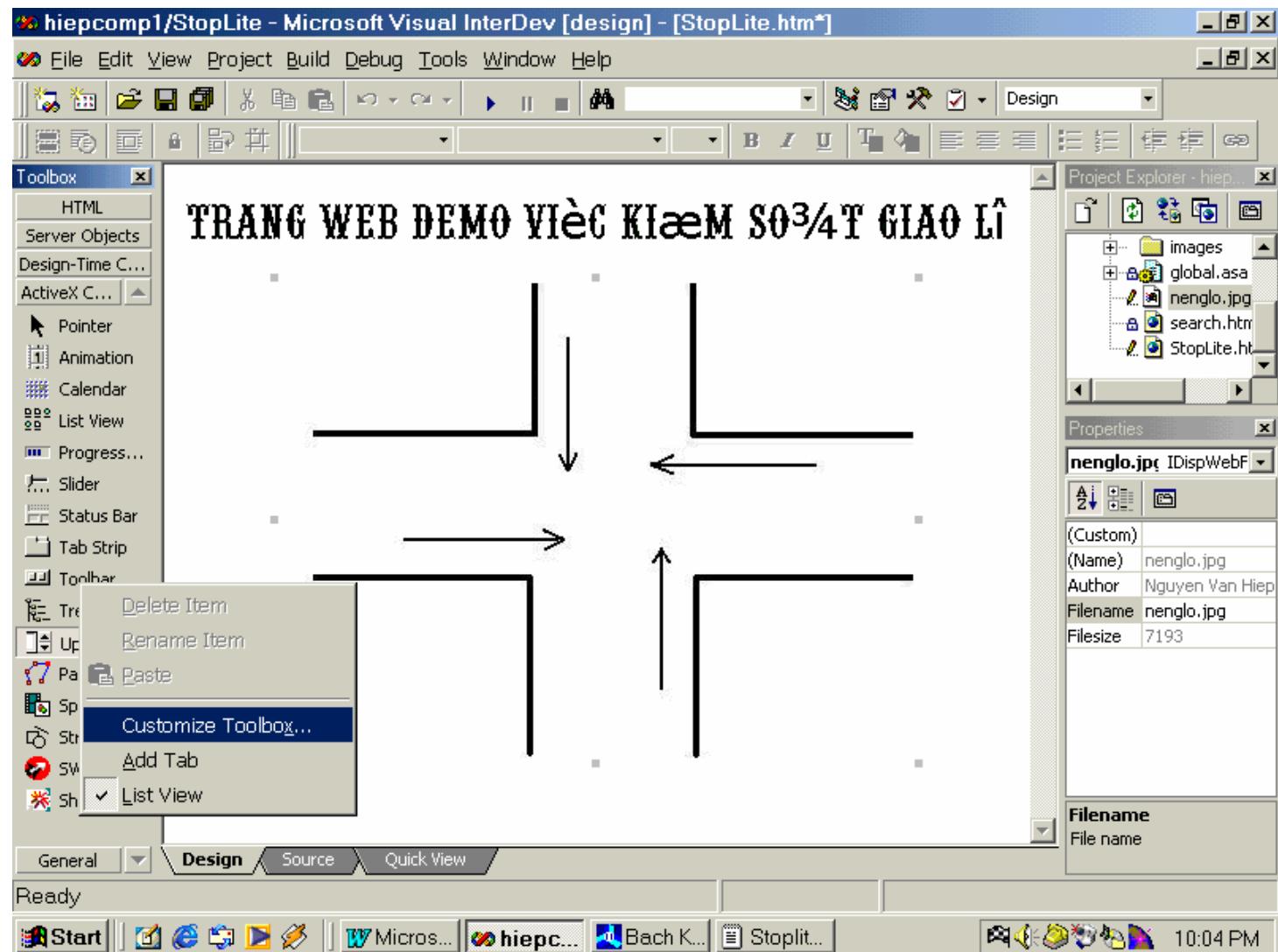
7.3 Tạo Website dùng ActiveX

14. Dùng chuột chọn tên file đồ họa trong cây project bên phải, drag nó vào vị trí thích hợp trong trang Web :



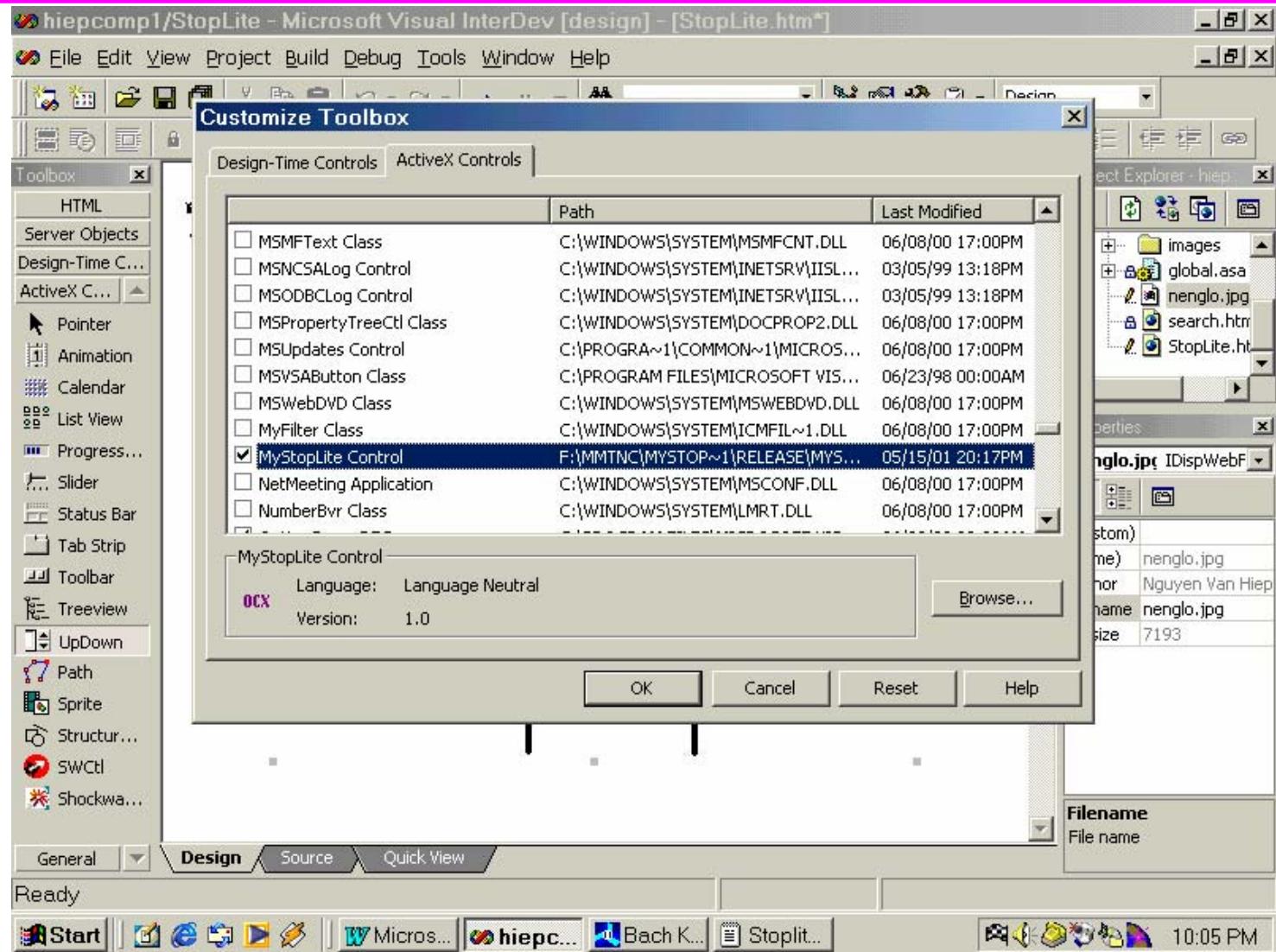
7.3 Tạo Website dùng ActiveX

15. Để project Web dùng được ActiveX Control, đầu tiên phải 'add' nó vào thanh Toolbox của project. Hãy dời chuột vào Toolbox, ấn chuột phải rồi chọn mục "Customize Toolbox" trong pop-up menu :



7.3 Tạo Website dùng ActiveX

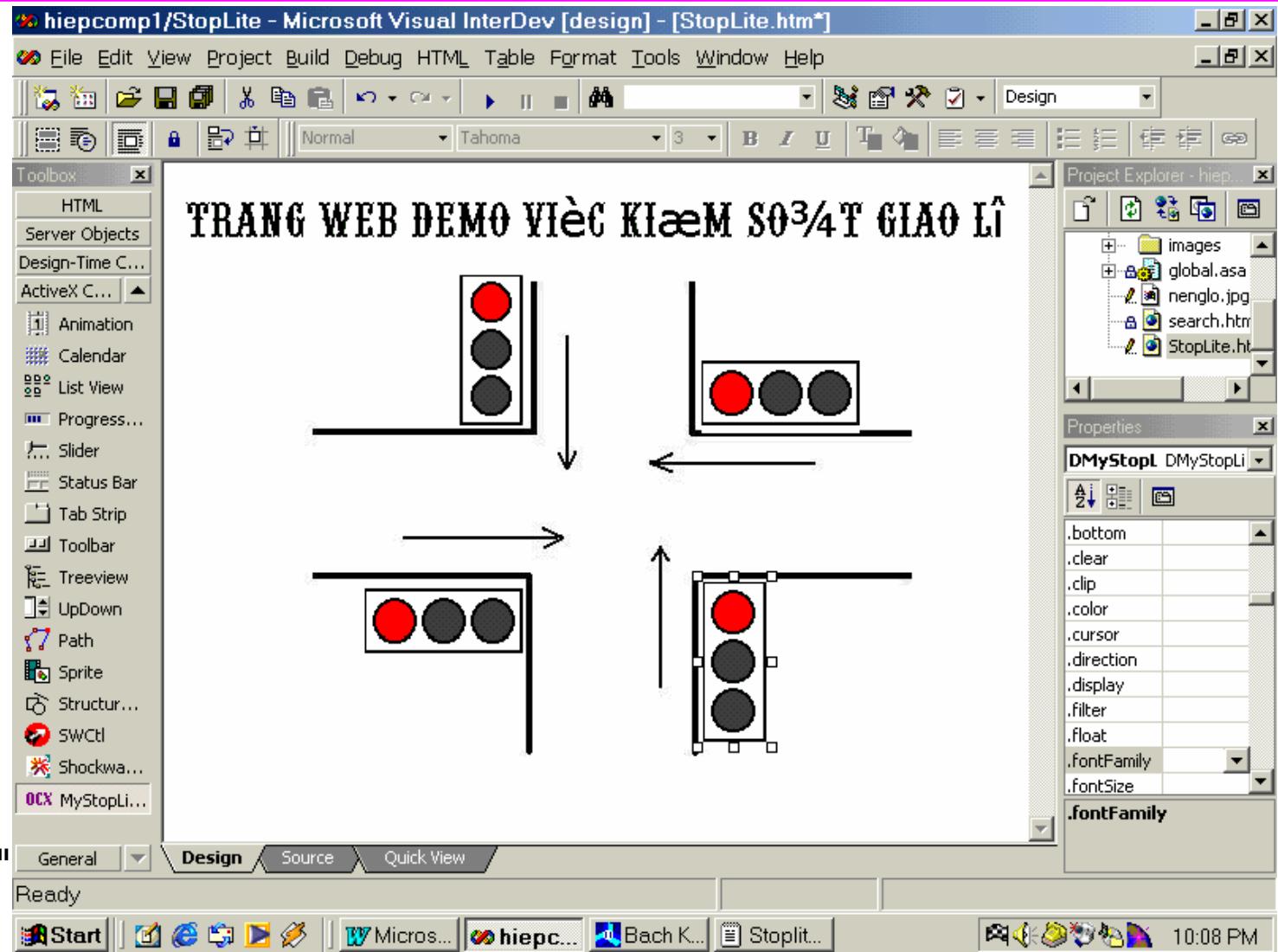
16. Chọn page "ActiveX Control", tìm mục tên Activex Control cần dùng trong danh sách, đánh dấu chọn vào checkbox tương ứng (lặp lại nhiều lần cho nhiều ActiveX Control cần dùng) rồi chọn button Ok.



7.3 Tạo Website dùng ActiveX

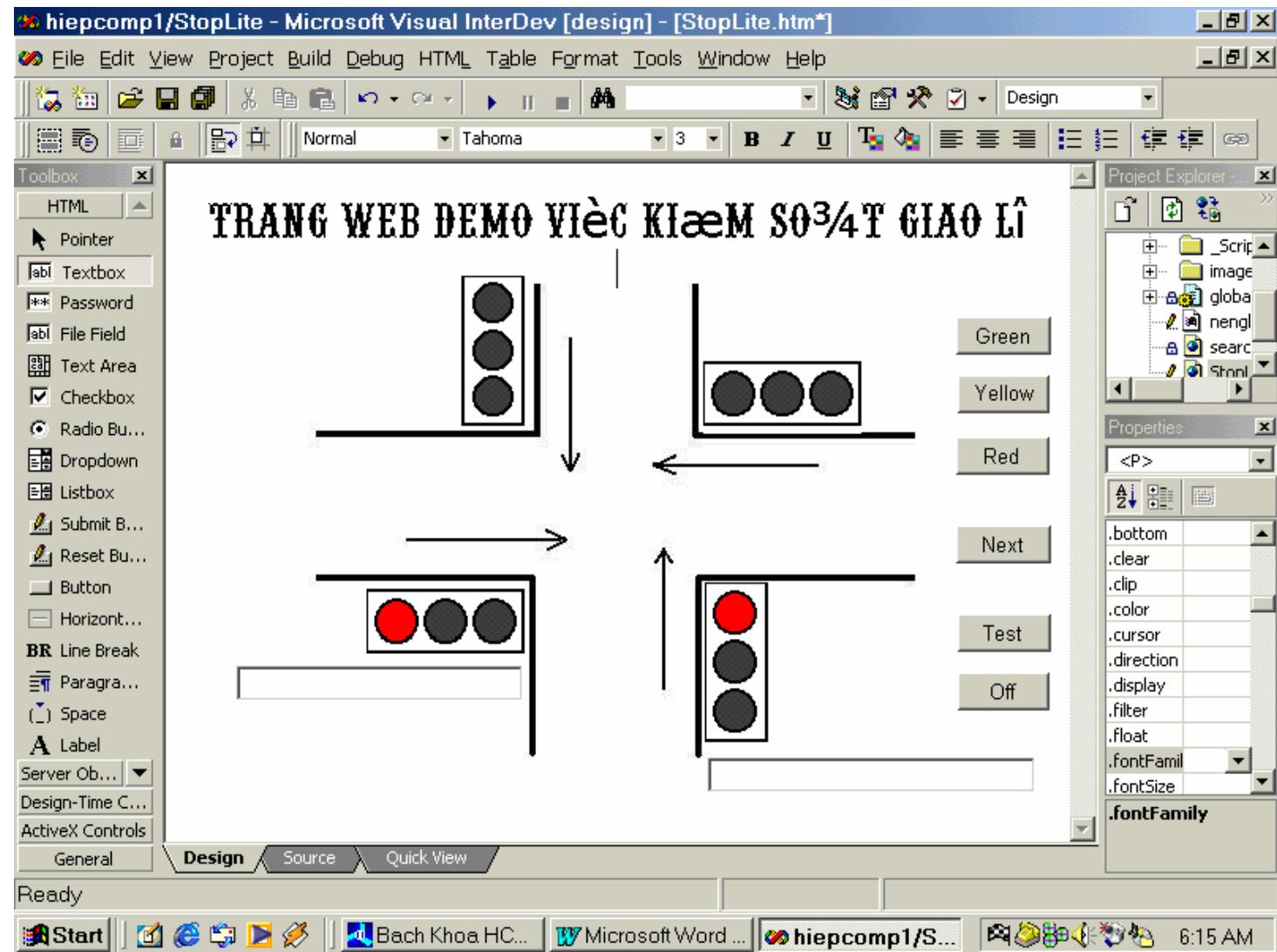
17. Chọn icon

"Absolute Mode" để có thể định vị chính xác vị trí các điều khiển trong trang Web. Dùng chuột drag ActiveX Control từ Toolbox vào vị trí mong muốn 4 lần, mỗi lần cho 1 dàn đèn, nếu vị trí và kích thước chưa đúng thì dời và thay đổi kích thước đến lúc mong muốn rồi ấn vào icon "Absolute Positioning" để fix nó.



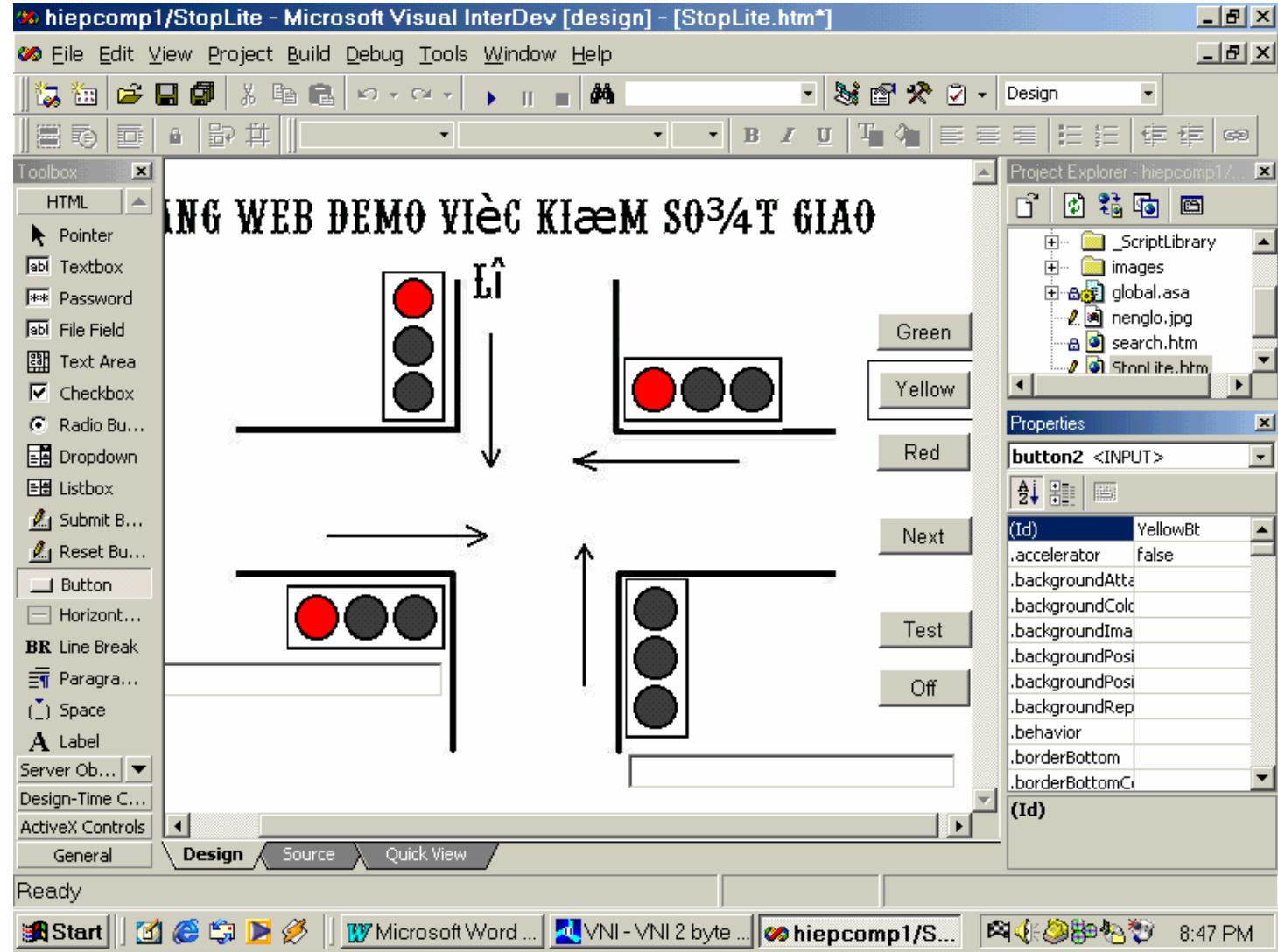
7.3 Tạo Website dùng ActiveX

18. Tiếp tục cách làm này cho 6 button và 2 textbox như hình sau (lưu ý các button và textbox nên được chọn từ page HTML trong Toolbox).



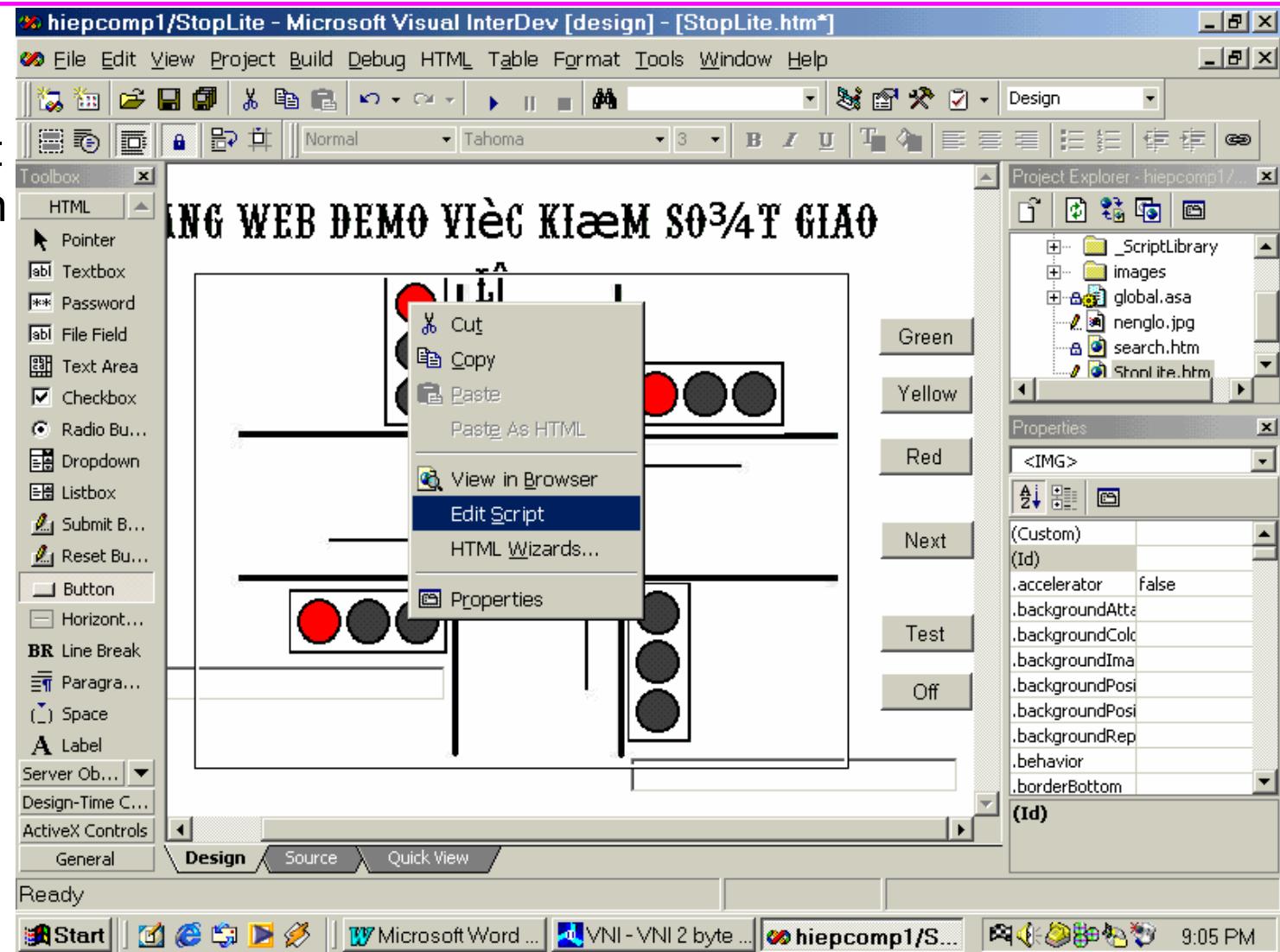
7.3 Tạo Website dùng ActiveX

19. Bây giờ hãy thiếp lập ID cho từng control. Thí dụ ấn phải chuột vào button Yellow rồi chọn mục Properties, tìm mục Id trong cửa sổ properties rồi hiệu chỉnh lại giá trị Id theo muốn muốn (tên các button là GreenBt, YellowBt, RedBt, NextBt, TestBt, OffBt, tên các textbox là warning1, warning2, tên các Activex là DMyStopLite1, DMyStopLite2, DMyStopLite3, DMyStopLite4).



7.3 Tạo Website dùng ActiveX

20. Để lập trình hàm xử lý biến cố cho các control, cần phải chuột trên 1 control rồi chọn mục Edit Script :



7.3 Tạo Website dùng ActiveX

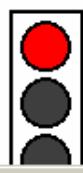
21. Để viết code xử lý biến cố Caution của ActiveX DMyStopLite1, nói rộng mục DMyStopLite1 trong cửa sổ Script Outline, ấn kép vào mục Caution, hàm xử lý sẽ được tạo ra tự động như sau :

The screenshot shows the Microsoft Visual InterDev interface. The title bar reads "hiepc1/StopLite - Microsoft Visual InterDev [design] - [StopLite.htm*]". The menu bar includes File, Edit, View, Project, Build, Debug, HTML, Table, Format, Tools, Window, Help. The left pane is the "Script Outline" window, which lists various ActiveX controls and their events. The "Caution" event under "DMyStopLite1" is selected. The main pane displays the HTML source code for "StopLite.htm". The code includes a VBScript block for the "Caution" event:

```
<html>
<head>
<META name="VI60_DefaultClientScript" Content="VBScript"
<meta NAME="GENERATOR" Content="Microsoft Visual Studio
<title></title>
<SCRIPT ID=clientEventHandlersVBS LANGUAGE=vbscript>
<!--

Sub DMyStopLite1_Caution
|
End Sub

-->
</SCRIPT>
</head>
<body onload="StartTimer () ">
<p align="center"><font face="VnWoodType-Demi" size="5">
giao li</font></p>
<p align="center">



```

The right pane contains the "Project Explorer" and "Properties" windows. The "Properties" window shows the "DOCUMENT" properties for the current file, including "bgColor" set to "Custom". The status bar at the bottom shows "Ready", "Ln 10 Col 1 Ch 1 STREAM INS", and the date/time "10:04 PM".



7.3 Tạo Website dùng ActiveX

22. Sau khi viết code
xử lý biến cố

Caution, Go, Stop cho
2 đèn kết hợp với 2
warning ta có kết quả
sau:

The screenshot shows the Microsoft Visual InterDev interface with the following details:

- Title Bar:** hiepcomp1/StopLite - Microsoft Visual InterDev [design] - [StopLite.htm*]
- Menu Bar:** File, Edit, View, Project, Build, Debug, HTML, Table, Format, Tools, Window, Help
- Toolbars:** Standard, Edit, View, Insert, Tools, Window, Help
- Script Outline:** Shows a tree view of client objects and scripts. Key nodes include Client Objects, DMyStopLite, and various methods like Caution, Click, Go, Off, Stop, Testing, and document.
- Code Editor:** Displays VBScript code for handling client events. The code defines several subroutines for DMyStopLite1 and DMyStopLite2 objects, setting the value of warning1 and warning2 variables based on button presses (Caution, Go, Stop).
- Project Explorer:** Lists files in the project: images, global.asa, nenglo.jpg, search.htm, StopLite.htm, and StopLite1.htm.
- Properties:** Shows properties for the DOCUMENT object, including bgColor, which is set to (Custom).
- Status Bar:** Ready, Ln 22, Col 13, Ch 10, STREAM, INS, 10:06 PM



7.3 Tạo Website dùng ActiveX

23. Code xử lý biến
cố cho các button
như sau :

The screenshot shows the Microsoft Visual InterDev interface with the following details:

- Title Bar:** hiepcomp1/StopLite - Microsoft Visual InterDev [design] - [StopLite.htm*]
- Menu Bar:** File, Edit, View, Project, Build, Debug, Tools, Window, Help
- Toolbars:** Standard, Edit, View, Insert, Tools, Window, Help
- Script Outline:** Shows event handlers for a button named "GreenBt". The selected event is "onclick". Other listed events include onactive, onafter, onbefo, onbefo, onbefo, onbefo, onbefo, onbefo, onbefo, onblur, oncellch, oncont, oncont, oncopy, oncut, ondata, ondata, ondata, ondblcli, ondeac, ondrag, ondrag, ondrag, ondrag, ondrag, ondrop, onerror.
- Code Editor:** Displays VBScript code for various button click events:

```
Sub GreenBt_onclick
    stat = 3
    NextBt_onclick()
End Sub

Sub RedBt_onclick
    stat = 1
    NextBt_onclick()
End Sub

Sub YellowBt_onclick
    stat = 0
    NextBt_onclick()
End Sub

Sub TestBt_onclick
    DMyStopLite1.Color(4)
    DMyStopLite3.Color(4)
    DMyStopLite2.Color(4)
    DMyStopLite4.Color(4)
End Sub

Sub OffBt_onclick
    DMyStopLite1.Color(0)
    DMyStopLite3.Color(0)
    DMyStopLite2.Color(0)
    DMyStopLite4.Color(0)
End Sub
```
- Project Explorer:** Lists files in the project: images, global.asa, nenglo.jpg, search.htm, StopLite.htm, and StopLite1.htm.
- Properties:** Shows properties for the DOCUMENT item, including:

(Custom)
aLink
background
bgColor
bgProperties
bottomMargin
defaultClientScri
defaultServerScri
DTCScriptingPlat
Client (IE 4.0 C)
enableScripting
False
enableSessions
True
leftMargin
Link
- Status Bar:** Ready, Design, Source, Quick View, Ln 45, Col 21, Ch 18, STREAM, INS, 10:09 PM



7.3 Tạo Website dùng ActiveX

24. Code xử lý
button Next :

The screenshot shows the Microsoft Visual InterDev interface with the following details:

- Title Bar:** hiepcomp1/StopLite - Microsoft Visual InterDev [design] - [StopLite.htm*]
- Menu Bar:** File Edit View Project Build Debug Tools Window Help
- Script Outline:** Shows event handlers for a button named "NextBt". The "onclick" event is selected.
- Code Editor:** Displays the following VBA code:

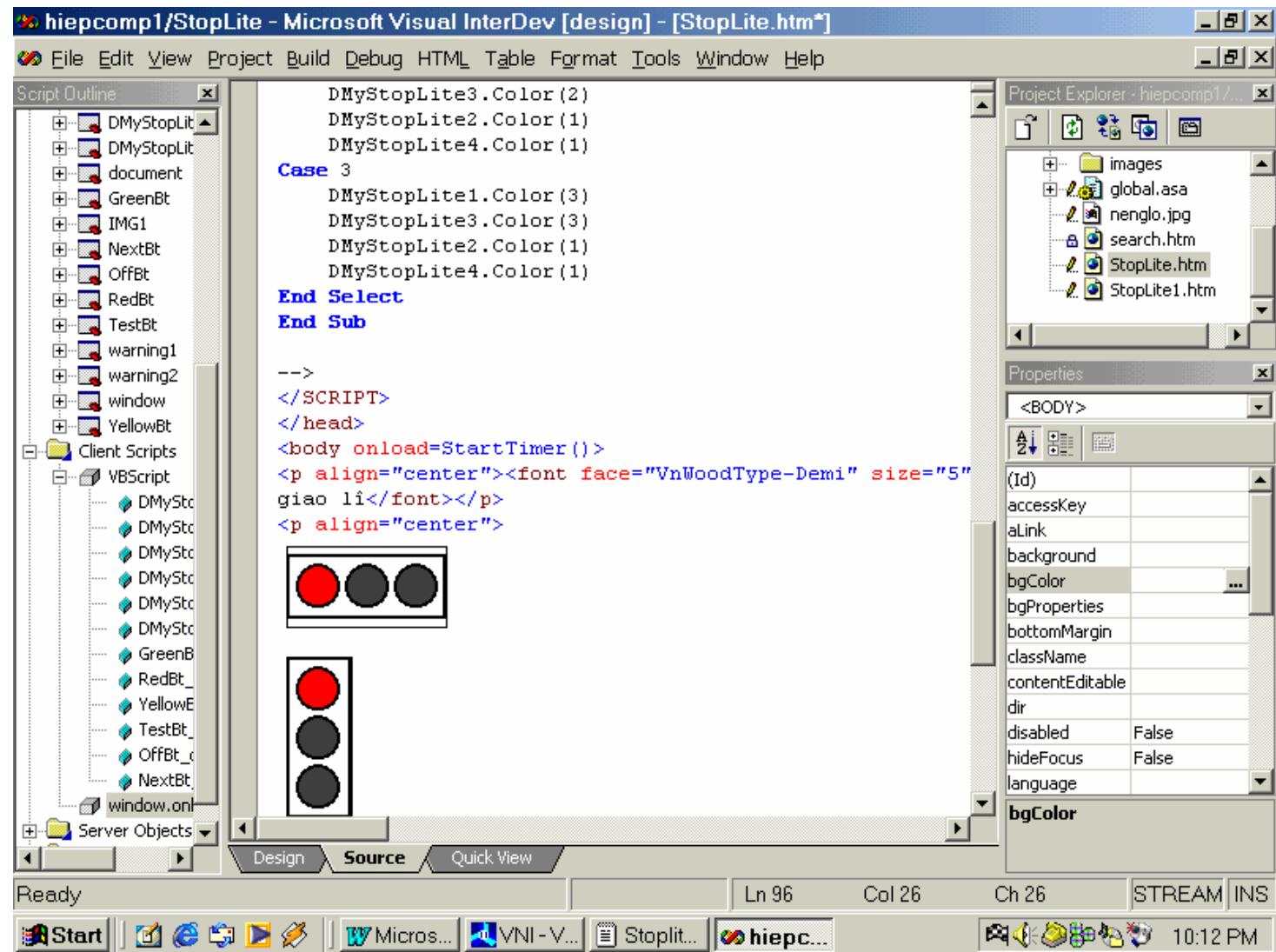
```
End Sub

Sub NextBt_onclick
stat = stat + 1
if stat=4 then
    stat = 0
end if
Select Case stat
Case 0
    DMyStopLite1.Color(1)
    DMyStopLite3.Color(1)
    DMyStopLite2.Color(2)
    DMyStopLite4.Color(2)
Case 1
    DMyStopLite1.Color(1)
    DMyStopLite3.Color(1)
    DMyStopLite2.Color(3)
    DMyStopLite4.Color(3)
Case 2
    DMyStopLite1.Color(2)
    DMyStopLite3.Color(2)
    DMyStopLite2.Color(1)
    DMyStopLite4.Color(1)
Case 3
    DMyStopLite1.Color(3)
    DMyStopLite3.Color(3)
    DMyStopLite2.Color(1)
    DMyStopLite4.Color(1)
End Select
```
- Project Explorer:** Shows files like images, global.asa, nenglo.jpg, search.htm, StopLite.htm, and StopLite1.htm.
- Properties:** Shows the "DOCUMENT" properties, including "bgColor" set to "bgColor".
- Status Bar:** Ready, Design, Source, Quick View, Ln 63, Col 1, Ch 1, STREAM, INS, 10:10 PM.



7.3 Tạo Website dùng ActiveX

25. Thêm thuộc tính
onload vào tag body
của trang Web để
khai báo hàm khởi
động thông số cho
ứng dụng Web :



7.3 Tạo Website dùng ActiveX

26. Viết code cho hàm khởi động trang Web :

The screenshot shows the Microsoft Visual InterDev interface with the following details:

- Title Bar:** hiepcomp1/StopLite - Microsoft Visual InterDev [design] - [StopLite.htm*]
- Menu Bar:** File Edit View Project Build Debug HTML Table Format Tools Window Help
- Toolbars:** Standard, Edit, View, Insert, Tools, Window, Help.
- Script Outline:** Shows objects like DMyStopLit, document, GreenBt, IMG1, NextBt, OffBt, RedBt, TestBt, warning1, warning2, window, YellowBt, and Client Scripts (VBScript).
- Code Editor:** Displays VBScript code for a web page:

```
<html>
<head>
<META name="VI60_DefaultClientScript" Content="VBScript"
<meta NAME="GENERATOR" Content="Microsoft Visual Studio
<title></title>
<SCRIPT ID=clientEventHandlersVBS LANGUAGE=vbscript>
<!--
' Khoi dong bien trang thai
stat = 3

Sub StartTimer()
NextBt_onclick()
id = window.setInterval("StatTrans()",5000)
End Sub

Sub StatTrans()
NextBt_onclick()
End Sub

Sub DMyStopLite1_Caution
warning1.value = "Den vang, hay dung lai"
End Sub

Sub DMyStopLite1_Go
warning1.value = "Den xanh, hay chay tiep"
End Sub

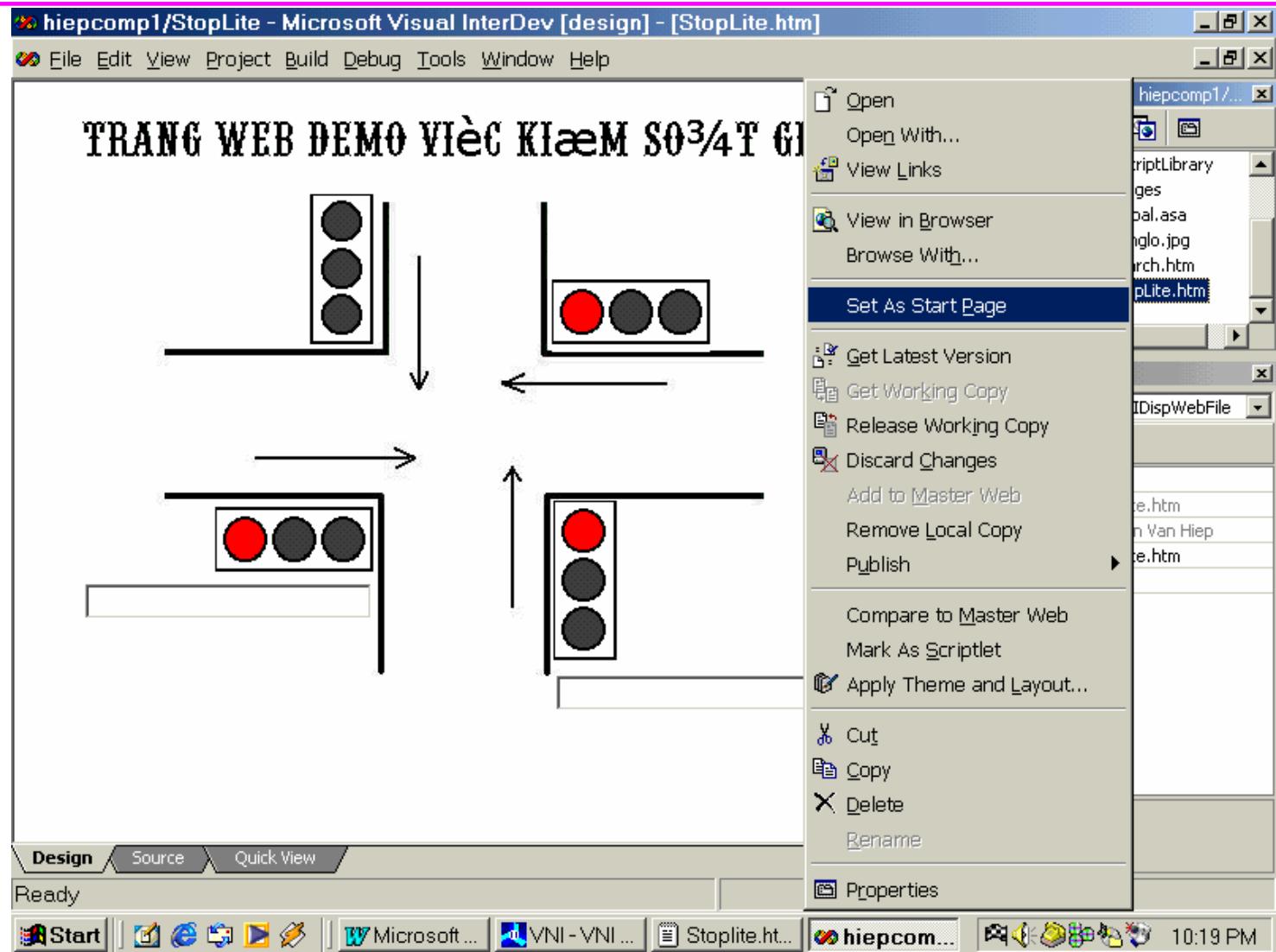
Sub DMyStopLite1_Stop
warning1.value = "Den do, hay dung lai"
End Sub
```
- Project Explorer:** Shows files like images, global.asa, nenglo.jpg, search.htm, StopLite.htm, and StopLite1.htm.
- Properties:** Shows DOCUMENT properties including aLink, background, bgColor, bgProperties, bottomMargin, defaultClientScri, defaultServerScri, DTCScriptingPlat Client (IE 4.0 C), enableScriptingC, enableSessions, leftMargin, Link, and bgColor.
- Status Bar:** Ready, Design, Source, Quick View, Ln 19, Col 1, Ch 1, STREAM, INS.
- Taskbar:** Start, Microsoft Internet Explorer, Stoplite, hiepc...
- System Tray:** Icons for speaker volume, network, and time (10:13 PM).



7.3 Tạo Website dùng ActiveX

27. Chọn mục tên trang Web trong cây Project bên phải, ấn phải chuột trên nó và chọn option "Set As Start Page" để khai báo trang Web này là trang bắt đầu chạy của project :

28. Thủ chạy trang web bằng cách chọn menu Debug.Start. Nếu có lỗi thì debug và sửa lỗi.



Môn học : Lập trình mạng

Chương 8

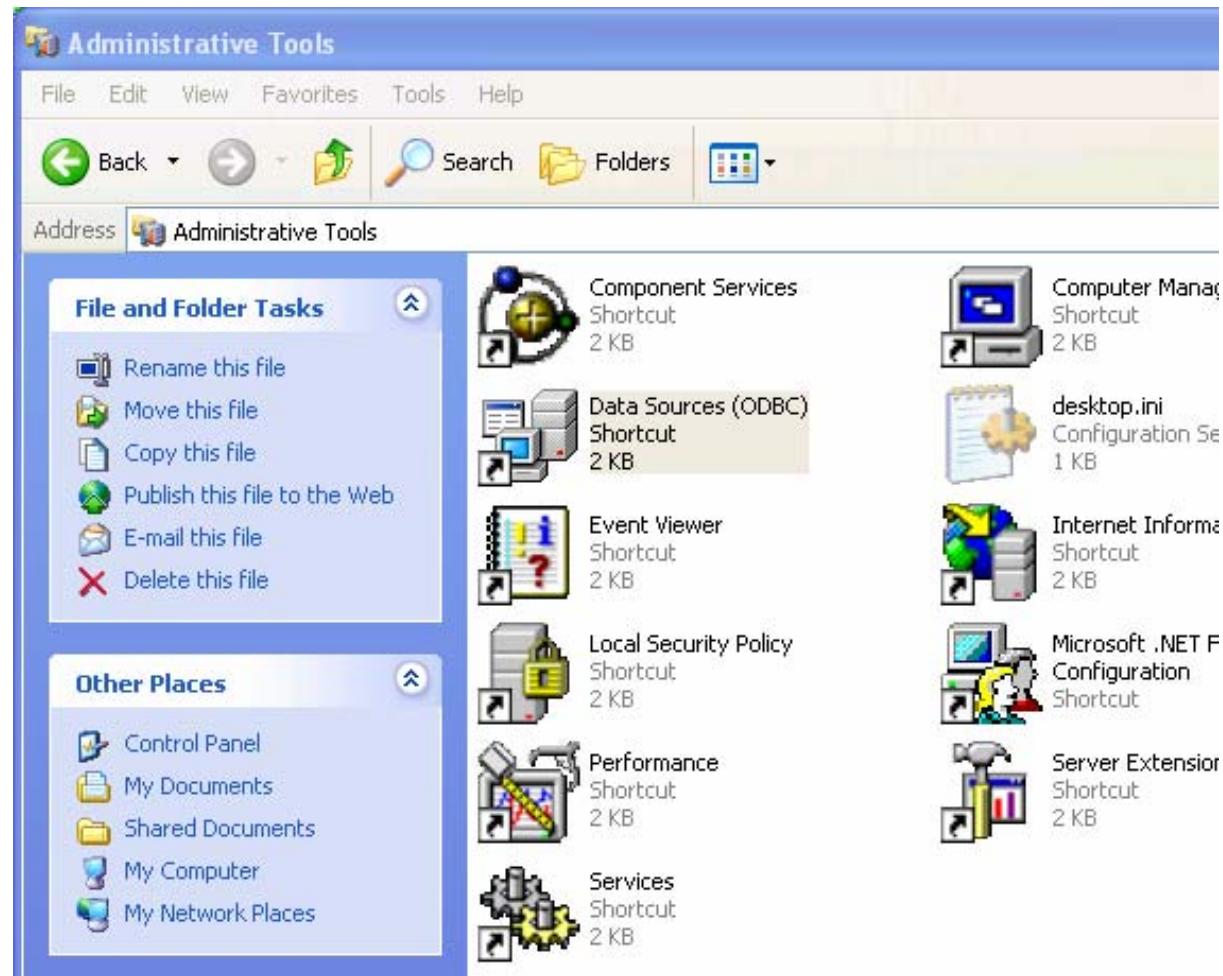
LẬP TRÌNH WEB CHẠY Ở SERVER BẰNG CÔNG NGHỆ MICROSOFT

- 8.1 Định nghĩa DSN (Data Source Name)
- 8.2 CGI (Common Gateway Interface)
- 8.3 ISAPI Extension
- 8.4 ASP (Active Server Page)



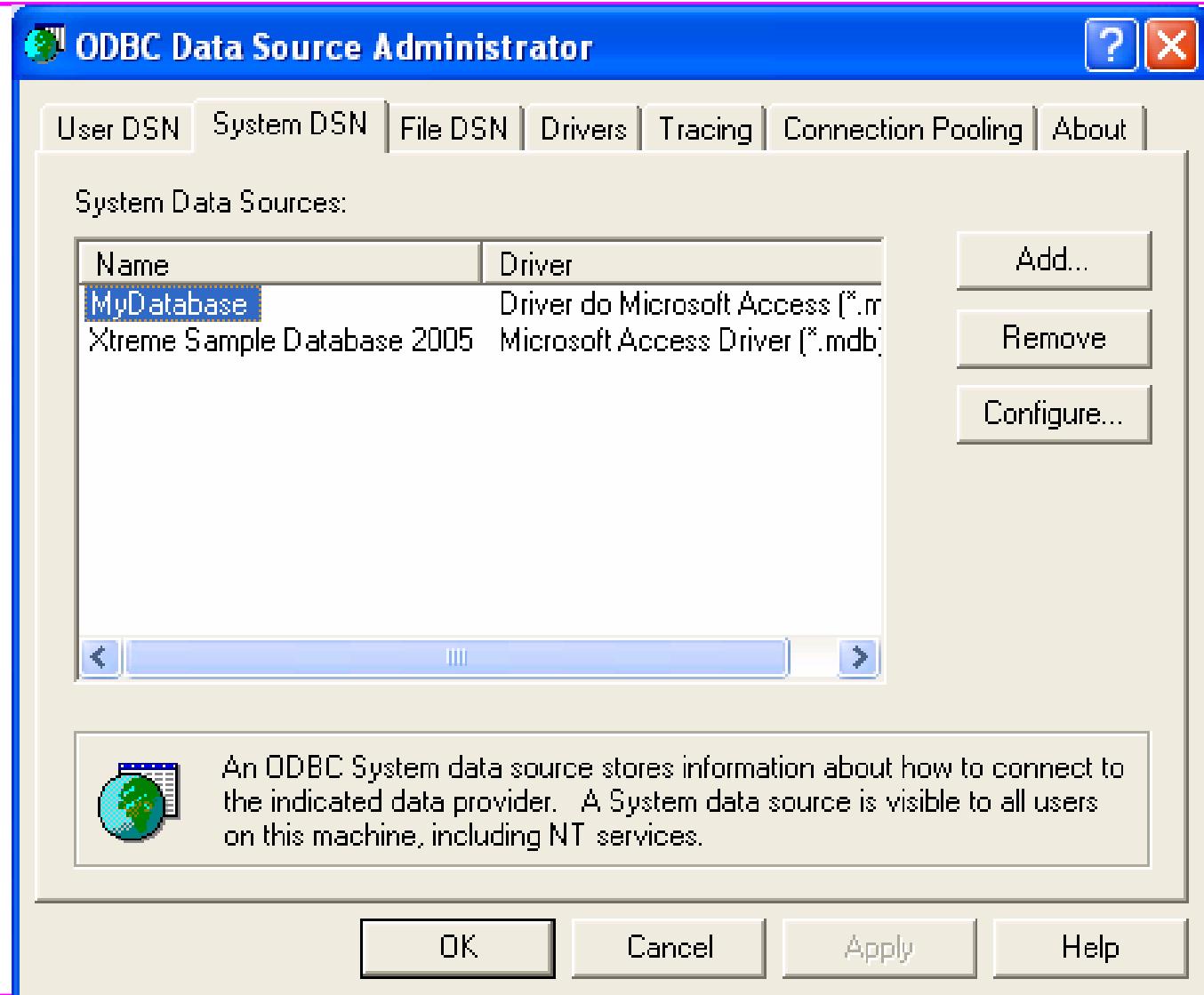
8.1 Qui trình định nghĩa 1 DSN

Trong các website của chương này, chúng ta sẽ truy xuất database. Để code truy xuất database có tính tổng quát cao, không phụ thuộc vào công nghệ quản lý database và vị trí database, chúng ta sẽ dùng DSN (Data Source Name) để nhận dạng database cần truy xuất. Để định nghĩa 1 DSN trên Windows, ta vào Control Panel. Administrative Tools, cửa sổ bên sẽ hiển thị :



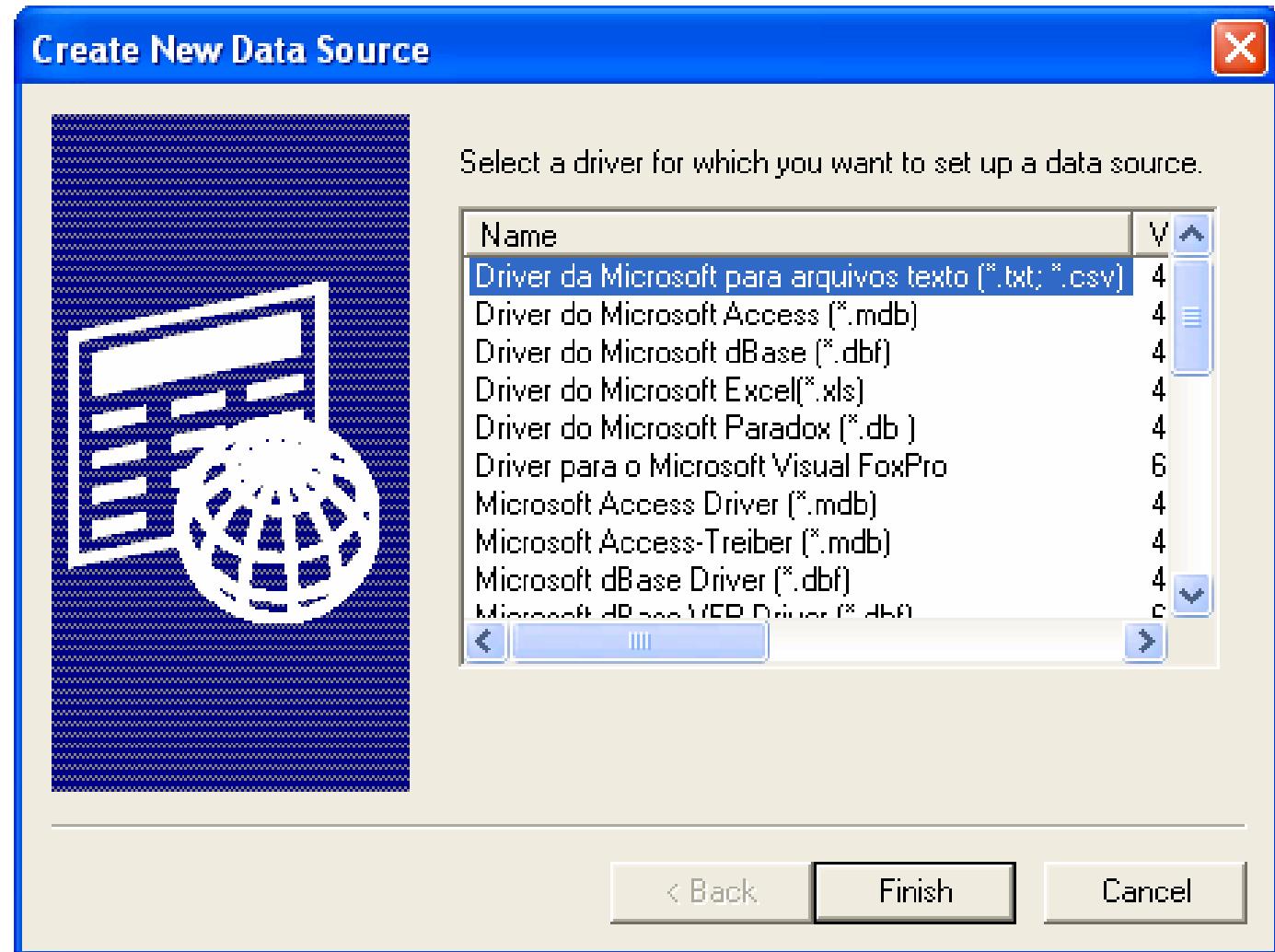
Qui trình định nghĩa 1 DSN

Ấn kép chuột vào mục Data Source (ODBC) để mở cửa sổ của tiện ích tương ứng như hình bên. Chọn tab “System DSN” để định nghĩa DSN có tầm vực truy xuất cấp hệ thống (bất kỳ ứng dụng nào cũng truy xuất được). Chọn button Add để thêm 1 DSN mới, cửa sổ Add sẽ hiển thị như slide kế.



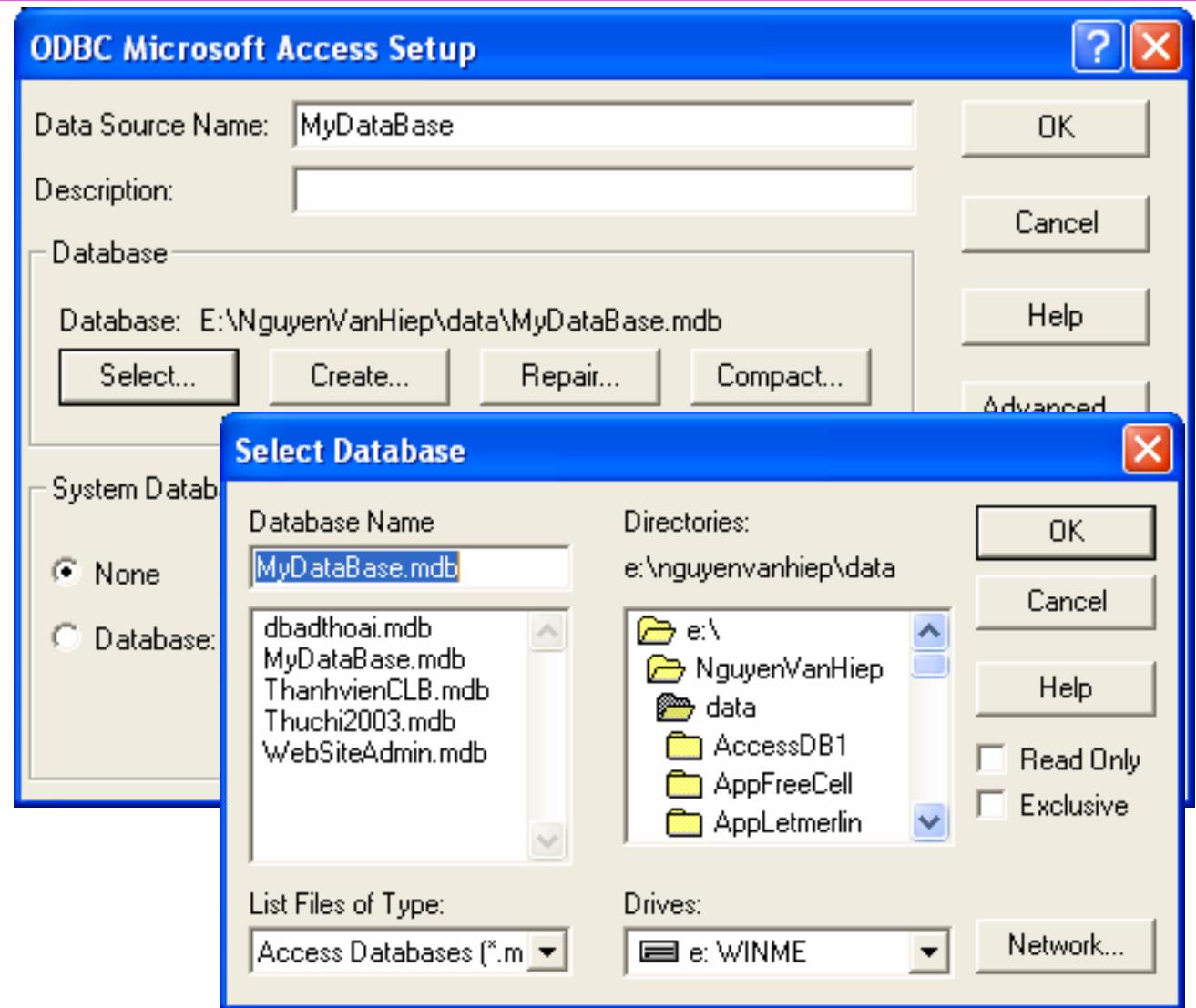
Qui trình định nghĩa 1 DSN

Duyệt tìm driver tương thích với hệ quản trị database và chọn nó. Cuối cùng chọn button Finish, cửa sổ của slide kế sẽ hiển thị.



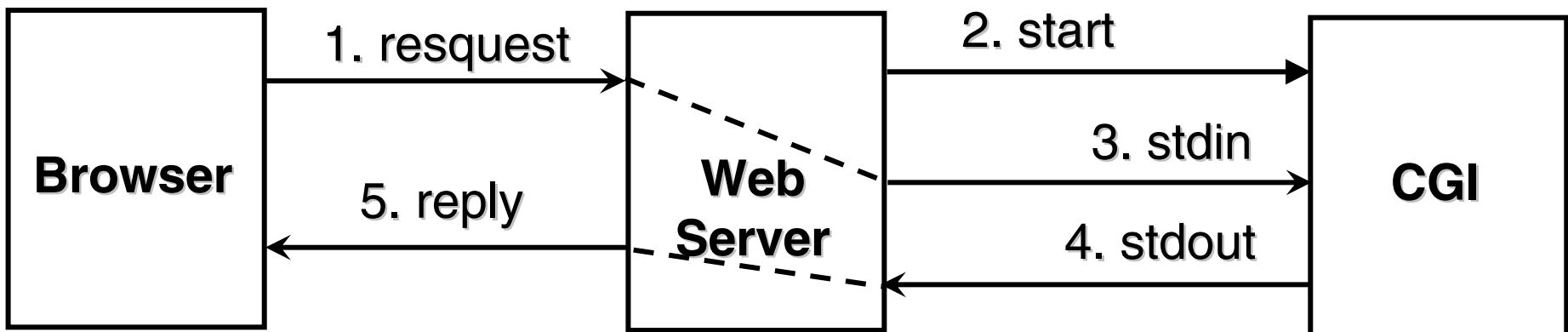
Qui trình định nghĩa 1 DSN

Nhập tên nhận dạng DSN, chọn button Select để hiển thị cửa sổ Select Database, trên cửa sổ Select, duyệt tìm file database, ấn button OK để kết thúc qui trình Add DSN. Từ đây, bất kỳ ứng dụng nào cũng có thể truy xuất database với tên nhận dạng là MyDatabase.



8.2 Tổng quát về CGI

- CGI là 1 phần mềm xử lý hàng lệnh truyền thống cho phép chuyển hướng (redirection) thiết bị nhập/xuất chuẩn. Mỗi lần Web server nhận yêu cầu từ client, nó chuẩn bị dữ liệu trong 1 file nhập, thiết lập file nhập và file xuất thành 2 thiết bị I/O chuẩn của module CGI rồi kích hoạt CGI chạy. Trong quá trình chạy, CGI nhập dữ liệu từ file nhập và xuất dữ liệu ra file xuất do Server thiết lập. Sau khi CGI chạy xong, Web server sẽ lấy dữ liệu từ file xuất và gửi về client.



Tổng quát về CGI

- Viết 1 CGI cũng giống như viết 1 ứng dụng xử lý hàng lệnh, ứng dụng này chỉ được đọc dữ liệu từ thiết bị stdin và xuất dữ liệu ra stdout. Có nhiều môi trường cho phép bạn viết ứng dụng CGI, trong đó môi trường VC++ là chuyên nghiệp nhất. Bạn có thể dùng loại Project “Win32 Console Application” để tạo ra ứng dụng CGI.



Thí dụ xây dựng 1 website bằng CGI

Yêu cầu : tạo 1 website dùng công nghệ CGI cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

Phân tích : cần 2 module chức năng :

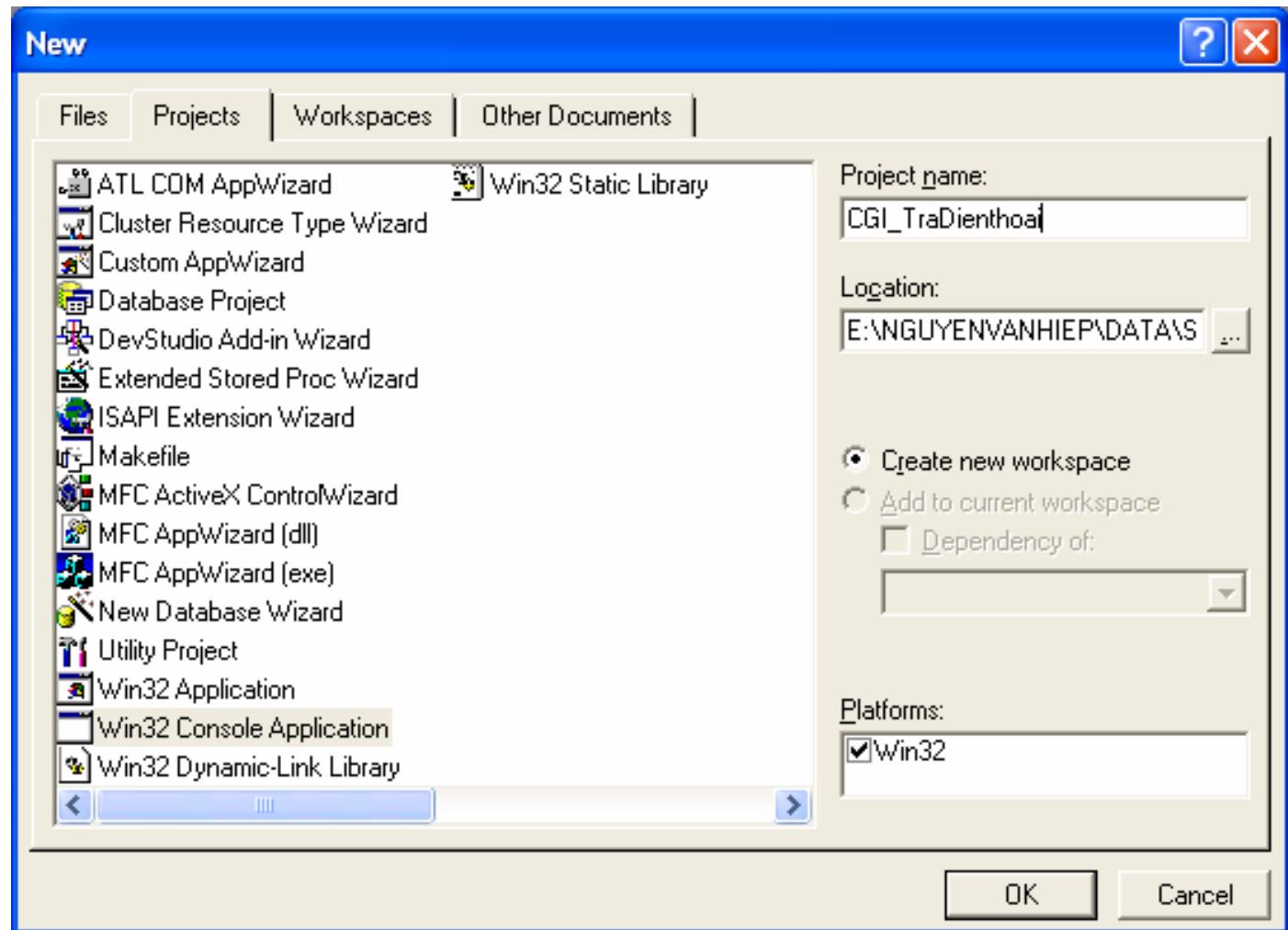
- module chứa form nhập thông tin về các account cần tra. module này có thể là 1 trang HTML thuần túy.
- và module truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm, module này là 1 ứng dụng CGI.

Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình VC++ 6.0. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



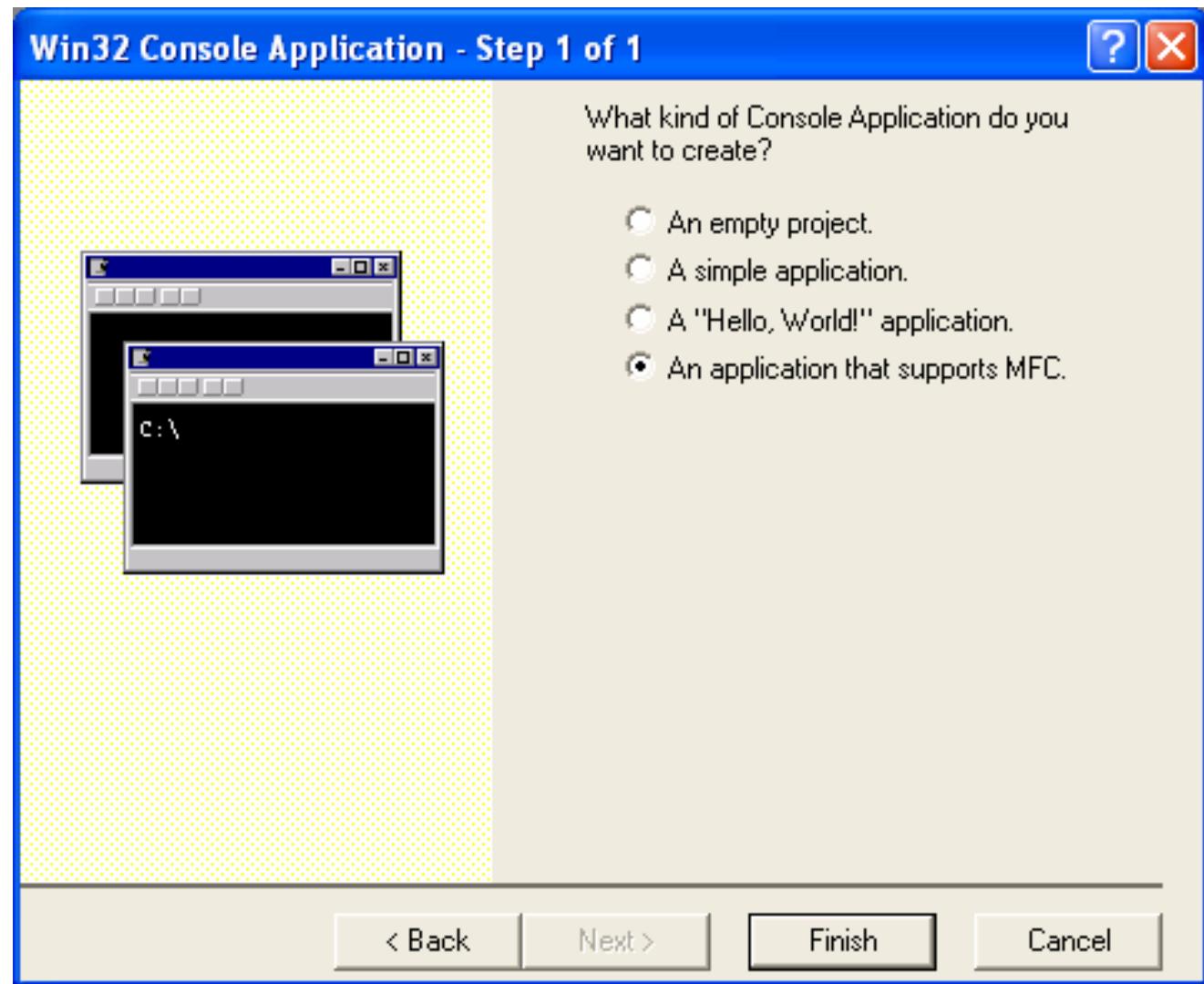
Tạo ứng dụng CGI tra điện thoại

1. Chạy VC++ 6.0, chọn menu File.New.Project, chọn loại Project là “Win32 Console Application”, chọn vị trí chứa Project “Location”, nhập tên Project là CGI_TraDienthoai, ấn button Ok.



Tạo ứng dụng CGI tra điện thoại

2. Trong cửa sổ Step 1, chọn mục “An application that supports MFC” để tạo ứng dụng hỗ trợ việc dùng các class MFC, chọn button Finish để hoàn tất việc tạo Project ứng dụng.



Tạo ứng dụng CGI tra điện thoại

3. Copy 3 file đặc tả class Parse (parse.h, parse.c, t99_type.h) từ vị trí ban đầu vào thư mục chứa Project, class này cho phép ta duyệt tìm từng tham số trong chuỗi request do client gửi đến để xử lý.
4. “Add” 3 file vừa copy vào Project bằng cách chọn menu Project.Add to Project.Files... rồi duyệt tìm và chọn 3 file trong thư mục chứa Project.
5. Chọn tab “Files View” trong cửa sổ Project để hiển thị cây các file cấu thành Project, tìm và mở file CGI_TraDienthoai.cpp rồi viết lại hàm _tmain() và hàm dịch vụ Search_Print() với nội dung như các slide kế tiếp.



Tạo ứng dụng CGI tra điện thoại

```
int _tmain(int argc, TCHAR* argv[], TCHAR* envp[]) {  
    char *query_str;  
    int len;  
    char *meth_str = getenv("REQUEST_METHOD");  
    int nRetCode = 0;  
    // initialize MFC and print an error on failure  
    if (!AfxWinInit(::GetModuleHandle(NULL), NULL, ::GetCommandLine(), 0)) {  
        // TODO: change error code to suit your needs  
        cerr << _T("Fatal Error: MFC initialization failed") << endl;  
        nRetCode = 1;  
    } else {  
        // TODO: code your application's behavior here.  
        if (strcmp(meth_str,"POST")==0) { // method POST  
            len = atoi(getenv("CONTENT_LENGTH"));  
            if (len) {  
                query_str = (char*)malloc(len+1);  
                fread(query_str,1,len,stdin);  
                query_str[len] = 0;  
            }  
        } else // method GET  
            query_str = getenv("QUERY_STRING");  
    }
```



Tạo ứng dụng CGI tra điện thoại

```
//xuất các tag lệnh ở đầu trang Web kết quả
printf("Content-Type: text/html\r\n\r\n");
printf("<HTML><HEAD>\n");
printf("<TITLE>Danh sach cac thue bao tim duoc</TITLE>\n");
printf("</HEAD><BODY>\n");
printf("<h2>Ket qua tra danh ba dien thoai cua ban :</h2>\n");
//tìm kiếm và xuất kết quả tìm kiếm
if(query_str && strlen(query_str)>0) {
    //tạo đối tượng list để duyệt tìm tham số
    Parse list( query_str );
    Search_Print(&cout, list.get_item_n("tentbao"),
        list.get_item_n("sodthoai"),list.get_item_n("diachi"));
} else {
    printf("<P>Noi dung goi ve co %d ky tu.\n",len);
}
//xuất các tag lệnh ở cuối trang Web kết quả
printf("<a href=CGI_TraDienthoai.htm>Quay ve trang chu</a>\n");
printf("</BODY></HTML>\n");
}
return nRetCode;
}
```



Tạo ứng dụng CGI tra điện thoại

```
// Hàm truy xuất database và xuất kết quả dạng *.htm
void Search_Print (ostream* pCtx, LPCTSTR tentbao, LPCTSTR sodthoai, LPCTSTR
diachi) {
char fsqlstart = 1;
    // Xây dựng lệnh SQL tìm kiếm
    CString sql(_T("Select * From danhbadienthoai"));
    if (tentbao && strlen(tentbao)>0) {
        sql += _T(" where tenthuebao like '%");
        sql += _T(tentbao);
        sql += _T("%'");
        fsqlstart = 0;
    }
    if (sodthoai && strlen(sodthoai) > 0) {
        if (fsqlstart) sql += _T(" where ");
        else sql += _T(" and ");
        sql += _T("sodienthoai like '%");
        sql += sodthoai;
        sql += _T("%'");
        fsqlstart = 0;
    }
}
```



Tạo ứng dụng CGI tra điện thoại

```
if (diachi && strlen(diachi) > 0) {
    if (fsqlstart) sql += _T(" where ");
    else sql += _T(" and ");
    sql += _T("diachi like '%");
    sql += diachi;
    sql += _T("%'");
    fsqlstart = 0;
}
_RecordsetPtr Rs;
char buf[256];
int nItem = 0;
try {
    ::CoInitialize(NULL);
    Rs.CreateInstance (__uuidof(Recordset));
    Rs->Open (_bstr_t(sql),
//_bstr_t("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=c:\\Source\\danhbadthoai.mdb"),
adOpenForwardOnly, adLockReadOnly, adCmdUnknown);
    _bstr_t("DSN=MyDatabase"),           adOpenForwardOnly,           adLockReadOnly,
adCmdUnknown);
    VARIANT vTentbao,vSodthoai,vDiachi;
    CString m_Tentbao,m_Sodthoai, m_Diachi;
```



Tạo ứng dụng CGI tra điện thoại

```
if (VARIANT_FALSE == Rs->GetadoEOF()) {  
    *pCtxt << "<table border=1>\n";  
    *pCtxt << "<tr><th>So thu tu</th><th>Ten thue bao</th><th>So dien  
thoai</th><th>Dia chi</th></tr>\n";  
    while (VARIANT_FALSE == Rs->GetadoEOF()) {  
        vTentbao = Rs->Fields->GetItem(_variant_t((long)1))->Value;  
        vSodthoai = Rs->Fields->GetItem(_variant_t((long)2))->Value;  
        vDiachi = Rs->Fields->GetItem(_variant_t((long)3))->Value;  
        m_Tentbao = vTentbao.bstrVal; m_Sodthoai = vSodthoai.bstrVal;  
        m_Diachi = vDiachi.bstrVal;  
        sprintf(buf, "++nItem, m_Tentbao, m_Sodthoai, m_Diachi);  
        *pCtxt << buf; Rs->MoveNext();  
    }  
    *pCtxt << "</table>\n";  
}  
if (nItem==0) *pCtxt << "<p> Khong co thue bao thoai dieu kien tim kiem cua ban.\n";  
}  
catch(_com_error &e) { // Exception handling. }  
Rs->Close(); Rs.Release();  
}
```



Tạo ứng dụng CGI tra điện thoại

6. Thêm lệnh import sau vào đầu file mã nguồn CGI_TraDienthoai.cpp để có thể dùng các đối tượng ADO truy xuất database :

```
#import "C:\Program Files\Common  
Files\SYSTEM\ADO\msado15.dll" no_namespace rename( "EOF",  
"adoEOF" )
```

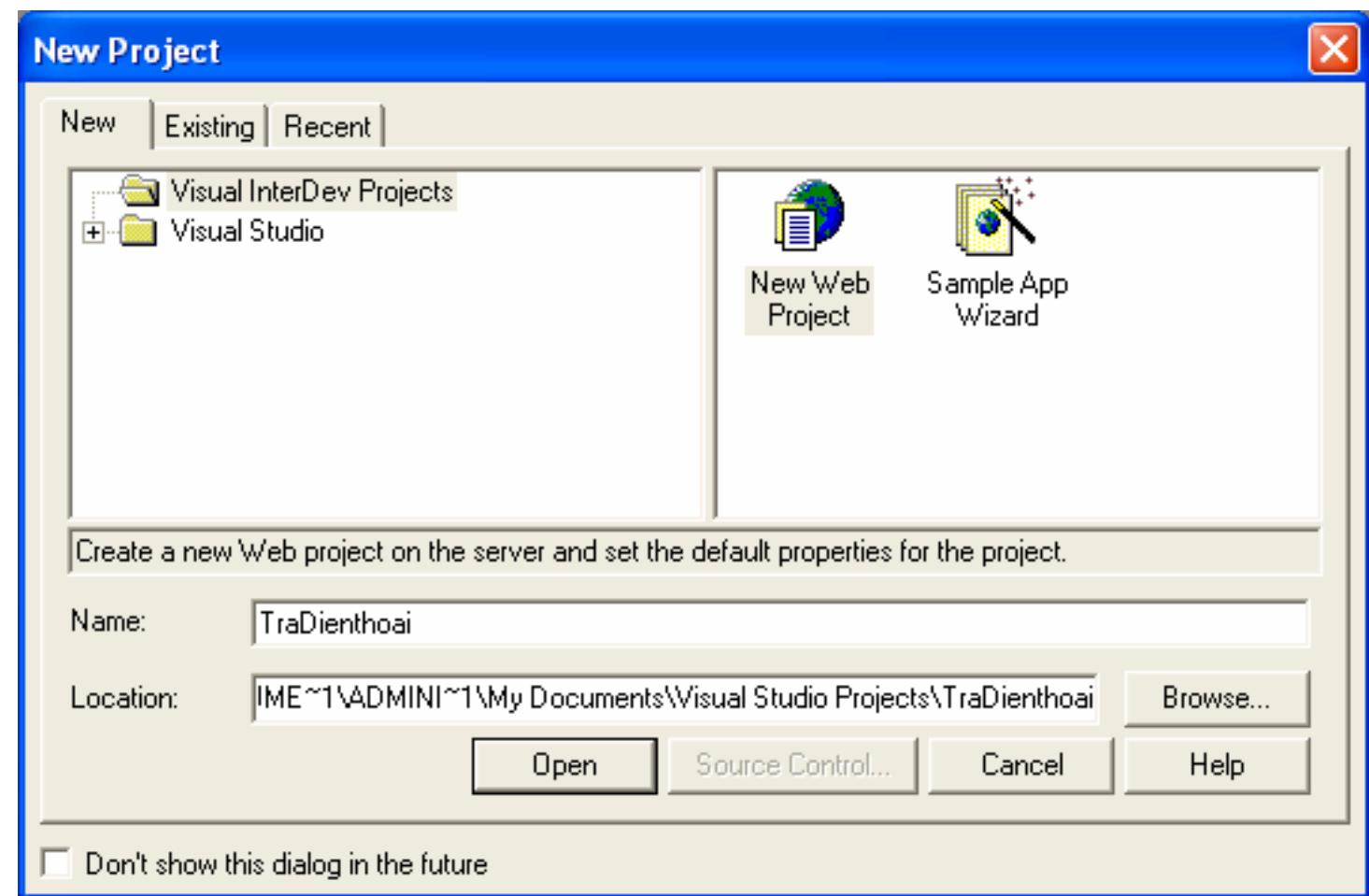
7. Chọn menu Build.Rebuild All để dịch các file mã nguồn và tạo file khả thi tương ứng. Nếu có lỗi từ vựng và cú pháp thì sửa. Sau khi tạo được file khả thi, hãy debug để tìm lỗi run-time nếu có.

Sau khi hoàn thành, ta có được file CGI khả thi tên là CGI_TraDienthoai.exe trong thư mục Debug hay Release tùy thuộc vào chế độ dịch (chọn menu Build.Set Active Configuration...). File CGI sẽ được “publish” lên website tra danh bạ mà ta sẽ tạo ra theo đặc tả của các slide kế tiếp.



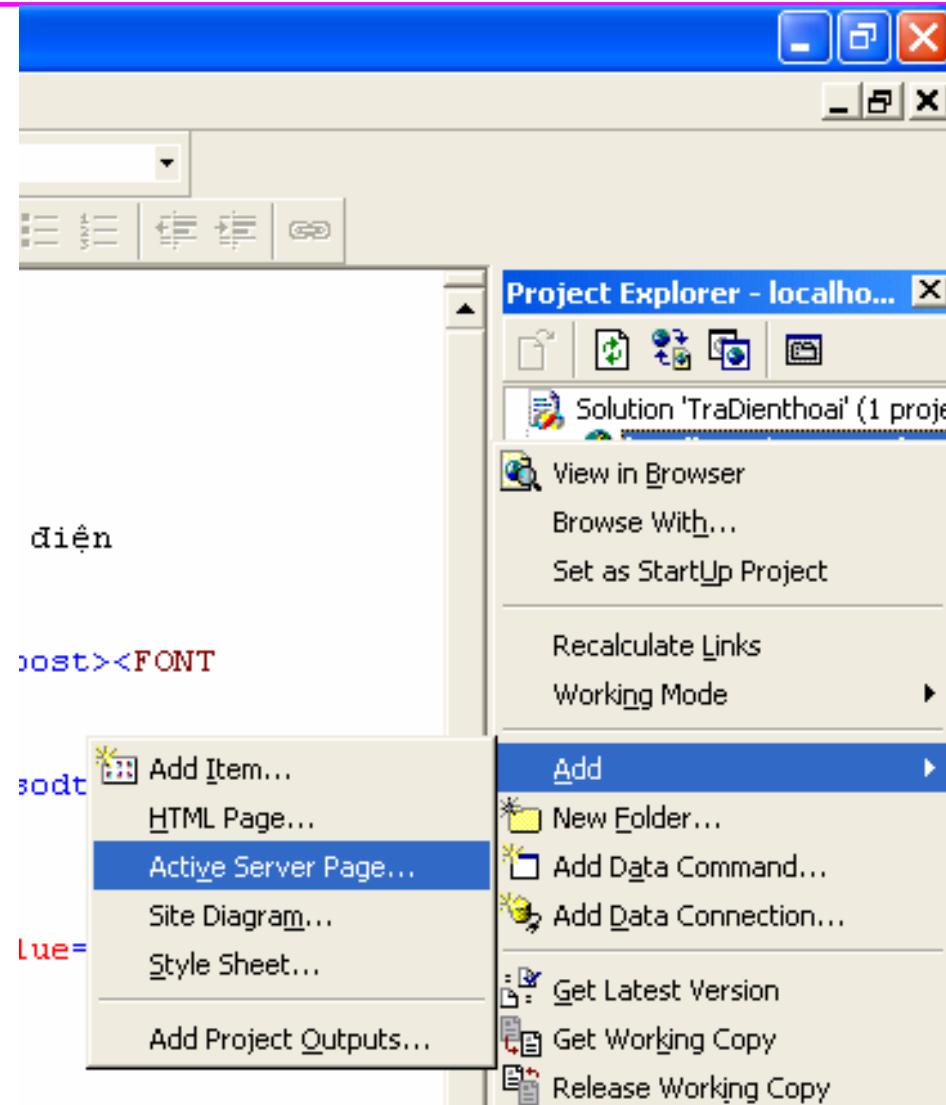
Tạo website tra điện thoại bằng InterDev

1. Chạy InterDev, khi cửa sổ New Project hiển thị, chọn Location chưa Project, nhập tên Project là TraDienthoai, ấn button Open rồi trả lời các yêu cầu trên các cửa sổ Wizard để hoàn tất việc tạo Project.



Tạo website tra điện thoại bằng InterDev

2. Dời chuột về cửa sổ Project, ấn phải chuột vào nút gốc để hiển thị menu pop-up chứa các lệnh, chọn mục Add.HTML Page..., nhập tên file là CGI_TraDienthoai để tạo home page cho Website tra danh bạ điện thoại.



Tạo website tra điện thoại bằng InterDev

3. Thực hiện các thao tác thiết kế trực quan để tạo trang Web có dạng như hình dưới (chứa tiêu đề chính và 1 form. Form có 3 textbox nhập liệu và 1 button “Bắt đầu tra”).

Demo việc tra điện thoại

Tên thuê bao :

Số điện thoại :

Địa chỉ :

Bắt đầu tra



Mã nguồn HTML của trang CGI_TraDienthoai.htm

```
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<TITLE></TITLE>
</HEAD>
<BODY>
<P align=center><FONT face=WoodType-Demi size=5>Demo việc tra điện thoại</FONT></P>
<P><FONT face=VnWoodType-Demi size=5>
<FORM action="CGI_TraDienthoai.exe" method=post>
<FONT face=VnTimes size=3>Tên thuê bao :&nbsp;
<INPUT id=tentbao name=tentbao style="width: 269px; height: 22px"></FONT>
<P><FONT face=VnTimes size=3>Số điện thoại :&nbsp;
<INPUT id=sodthoai name=sodthoai></FONT></P>
<P><FONT face=VnTimes size=3> Địa chỉ :&nbsp;
<INPUT id=diachi name=diachi style="width: 373px; height: 22px"></FONT></P>
<P><FONT face=VnTimes size=3>
<INPUT type=submit value="Bắt đầu tra"></FONT></P></FORM></FONT>
<P></P>
</BODY> </HTML>
```

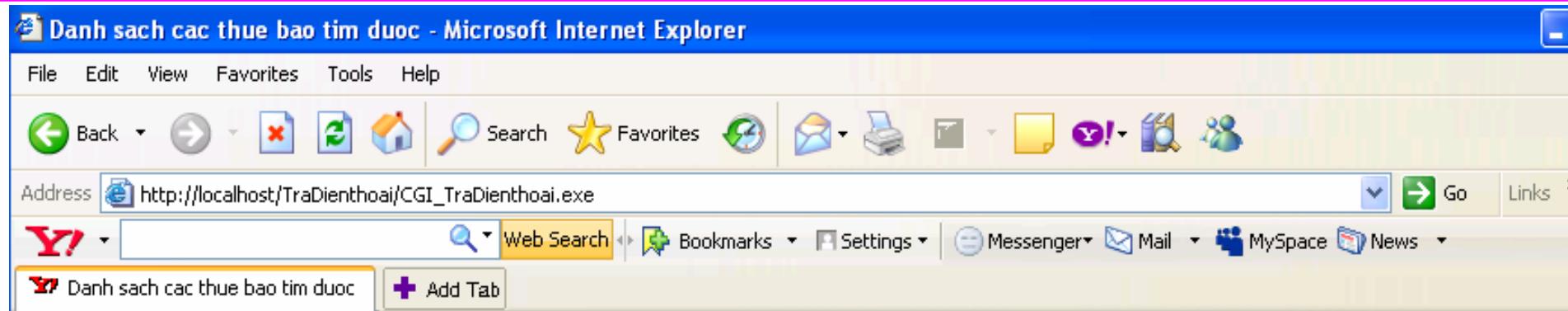


Tạo website tra điện thoại bằng InterDev

4. Copy file khả thi CGI_TraDienthoai.exe vào thư mục chứa Website.
5. Mở rộng nội dung của cây Project, chọn mục CGI_TraDienthoai.htm, ấn phải chuột trên nó rồi chọn mục “Set As Start Page”
6. thử chạy Website bằng cách chọn menu Debug.Start. Khi trang Web chứa form nhập liệu hiển thị, người dùng sẽ nhập các thông tin về account mà mình muốn tìm kiếm rồi ấn button “Bắt đầu tra”. Kết quả tra cứu sẽ có dạng như hình ở slide kế tiếp.



Tạo website tra điện thoại bằng InterDev



Kết quả tra danh bạ điện thoại của bạn :

So thu tu	Ten thue bao	So dien thoai	Địa chỉ
1	Nguyen Van Hiep	8644717	8A Lu Gia, P15, Q11, Tp.HCM
2	Huynh Thoai Loan	9070437	245/4 Ly Thuong Kiet, P15, Q11, Tp.HCM
3	Nguyen Luong Hoa	9024437	240 Tran Binh Trong, P4, Q5, Tp.HCM
4	Nguyen Lac Han	2030478	157 Hoang Dieu, P1, Q1, Tp.HCM

[Quay về trang chủ](#)

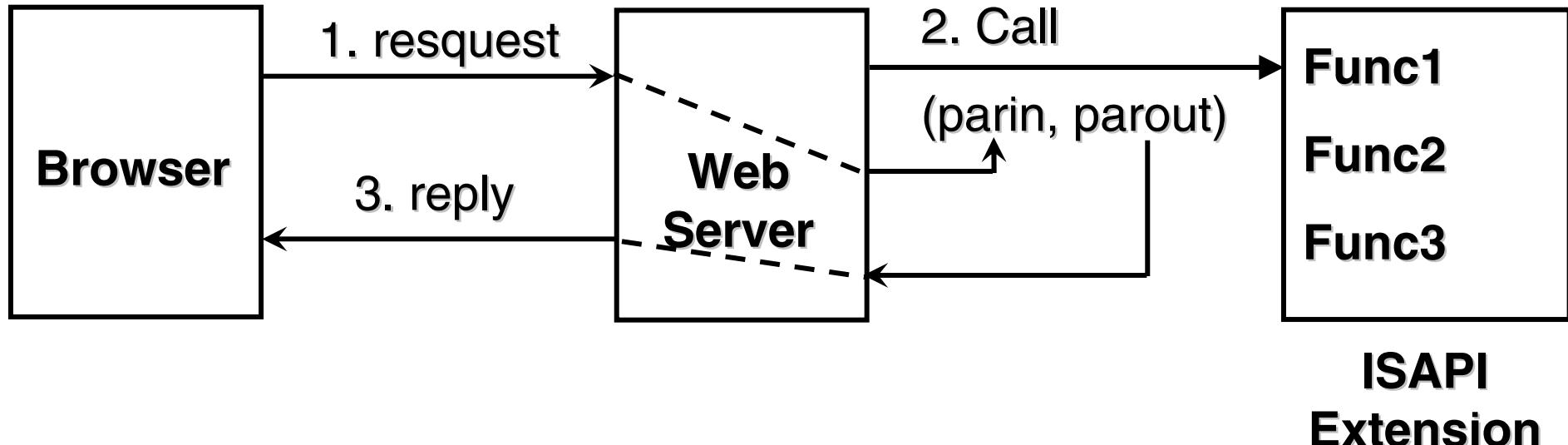


8.3 Tổng quát về ISAPI Extension

- Công nghệ CGI không hiệu quả vì nếu có nhiều khách hàng cùng truy xuất đồng thời thì server sẽ kích hoạt chạy đồng thời nhiều “instance” của ứng dụng CGI, mỗi “instance” là 1 process độc lập. Chi phí quản lý process của HĐH là khá cao.
- Để khắc phục khuyết điểm trên, Microsoft đề nghị công nghệ ISAPI Extension. Ý tưởng chính là thay vì phải viết 1 ứng dụng độc lập để xử lý 1 form từ khách hàng, ta sẽ đặt đoạn code xử lý 1 form vào trong 1 tác vụ (hàm), đặt hàm này (và nhiều hàm xử lý form khác) vào trong cùng 1 thư viện *.dll (Microsoft dùng thuật ngữ ISAPI Extension để miêu tả thư viện này).
- Mỗi khi có yêu cầu từ client, Web server sẽ liên kết động với thư viện và gọi hàm xử lý form thích hợp, như vậy không cần phải tạo thêm process mới. Kết quả là độ hiệu quả được nâng cao rất nhiều so với công nghệ CGI.



Tổng quát về ISAPI Extension



- Hiện môi trường VC++ 6.0 hỗ trợ việc xây dựng ISAPI Extension rất tốt. Bạn có thể dùng loại Project “ISAPI Extension Wizard” để tạo ra 1 ISAPI Extension/ISAPI Filter.

Nhập/xuất thông tin của ISAPI Extension

- Mỗi hàm xử lý form trong ISAPI Extension có nhiều tham số hình thức, trong đó tham số đầu tiên pCtxt miêu tả đối tượng xuất thông tin, còn các tham số còn lại miêu tả các dữ liệu nhập từ Form.

//đặc tả hàm TraDthoai() phục vụ việc tra điện thoại

ON_PARSE_COMMAND(TraDthoai, CMyISAPIExtension,

ITS_PSTR | TS_PSTR | TS_PSTR)

ON_PARSE_COMMAND_PARAMS("tentbao sodthoai diachi")

- Việc truy xuất dữ liệu nhập luôn thông qua tham số hình thức tương ứng.
- Mỗi lần cần xuất kết quả, ta dùng lệnh có dạng :
*pCxt << “chuỗi cần xuất”;



Thí dụ xây dựng 1 website bằng ISAPI

Yêu cầu : tạo 1 website dùng công nghệ ISAPI Extension cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

Phân tích : cần 2 module chức năng :

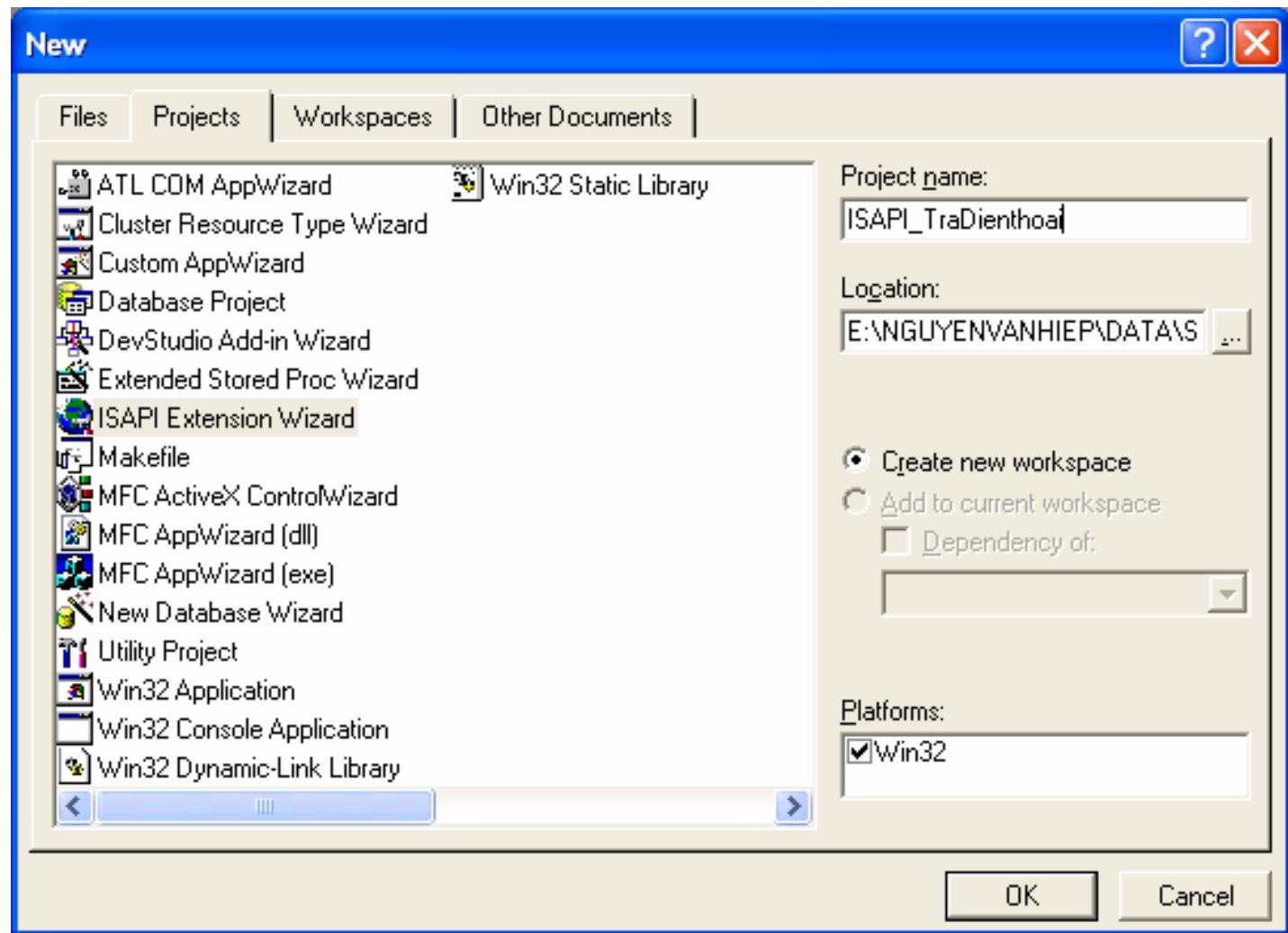
- module chứa form nhập thông tin về các account cần tra, module này có thể là 1 trang HTML thuần túy.
- và module ISAPI Extension, module này chứa ít nhất 1 hàm TraDthoai() có chức năng truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm.

Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình VC++ 6.0. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



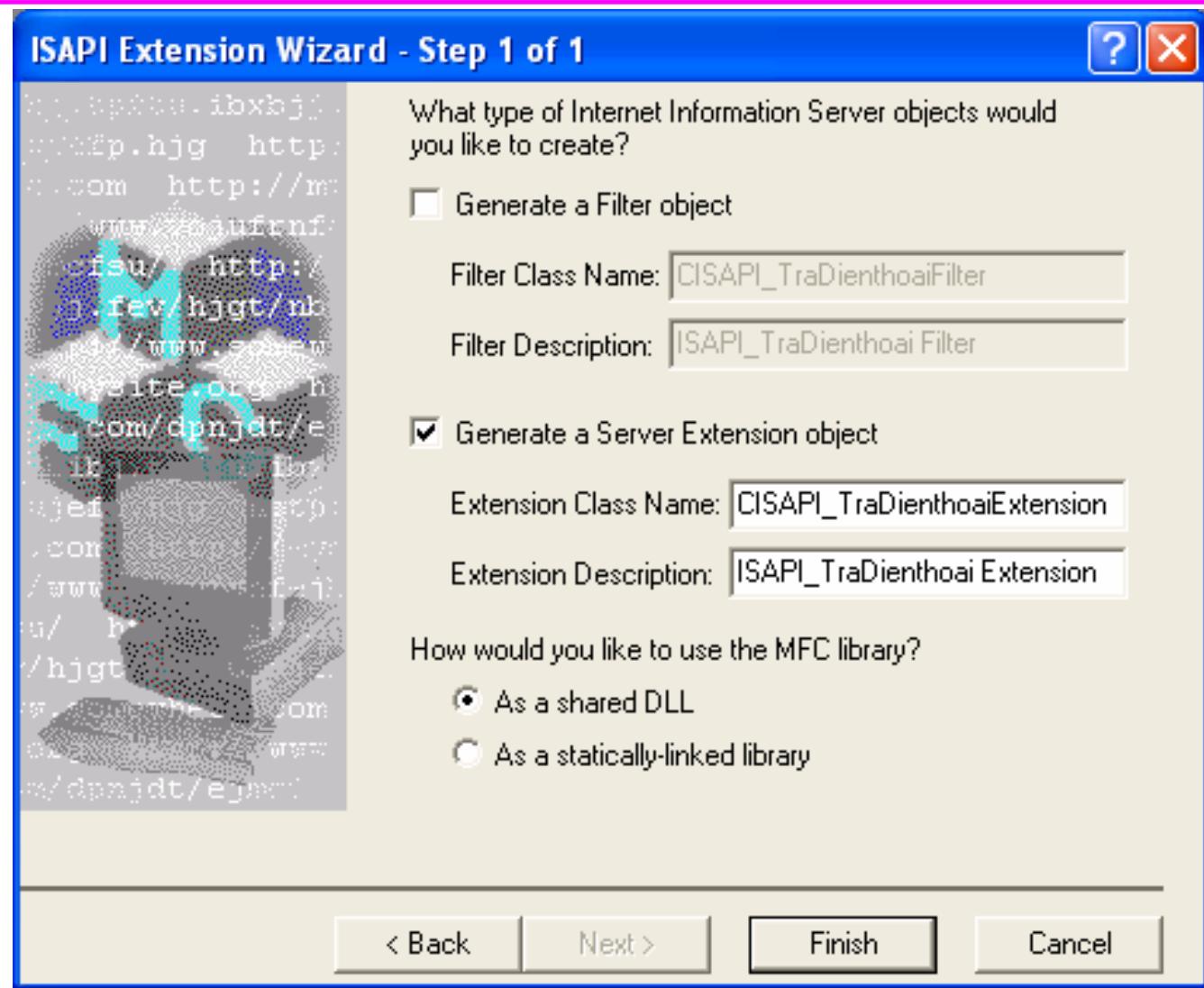
Tạo ISAPI Extension tra điện thoại

1. Chạy VC++ 6.0, chọn menu File.New.Project, chọn loại Project là “ISAPI Extension Wizard”, chọn vị trí chứa Project “Location”, nhập tên Project là ISAPI_TraDienthoai, ấn button Ok.



Tạo ISAPI Extension tra điện thoại

2. Trong cửa sổ Step 1, chọn mục “Generate a Server Extension object” để tạo module ISAPI Extension, chọn button Finish để hoàn tất việc tạo Project.



Tạo ISAPI Extension tra điện thoại

3. Chọn tab Class View trong cửa sổ Project để hiển thị các class cấu thành ISAPI, mở rộng cây Project, ấn kép chuột vào mục ISAPI_TraDienthoaiExtension để mở cửa sổ soạn code của ISAPI như hình bên.

The screenshot shows the Microsoft Visual Studio interface. The title bar reads "ISAPI_TraDienthoai - Microsoft Visual C++ - [ISAPI_TraDienthoai.cpp]". The menu bar includes File, Edit, View, Insert, Project, Build, Tools, Window, Help. The toolbar has various icons for file operations. The Class View window on the left shows the class hierarchy under "ISAPI_TraDienthoai cl." It includes the main class "CISAPI_TraDienthoaiE" and its members: ~CISAPI_TraDienthoaiE, DefaultCHttpServer, GetExtensionVersion, and TerminateExtension. A "Globals" folder is also listed. The code editor on the right contains the implementation for this class. The code includes comments explaining the structure and how to handle commands. It defines a CWinApp object named "theApp", a command-parsing map, and an instance of CISAPI_TraDienthoaiExtension named "theExtension". The implementation includes constructors and destructors for the extension class and a method for getting extension version information.

```
// The one and only CWinApp object
// NOTE: You may remove this object if you alter your project to no
// longer use MFC in a DLL.

CWinApp theApp;

///////////////////////////////
// command-parsing map

BEGIN_PARSE_MAP(CISAPI_TraDienthoaiExtension, CHttpServer)
// TODO: insert your ON_PARSE_COMMAND() and
// ON_PARSE_COMMAND_PARAMS() here to hook up your commands.
// For example:

ON_PARSE_COMMAND(Default, CISAPI_TraDienthoaiExtension, ITS_EMPTY)
DEFAULT_PARSE_COMMAND(Default, CISAPI_TraDienthoaiExtension)
END_PARSE_MAP(CISAPI_TraDienthoaiExtension)

///////////////////////////////
// The one and only CISAPI_TraDienthoaiExtension object

CISAPI_TraDienthoaiExtension theExtension;

///////////////////////////////
// CISAPI_TraDienthoaiExtension implementation

CISAPI_TraDienthoaiExtension::CISAPI_TraDienthoaiExtension()
{
}

CISAPI_TraDienthoaiExtension::~CISAPI_TraDienthoaiExtension()
{
}

BOOL CISAPI_TraDienthoaiExtension::GetExtensionVersion(HSE_VERSION_INFO
```



Tạo ISAPI Extension tra điện thoại

4. ISAPI là 1 thư viện chứa nhiều tác vụ, mỗi tác vụ được khai báo trong macro BEGIN_PARSE_MAP :

```
BEGIN_PARSE_MAP(CISAPI_TraDienthoaiExtension, CHttpServer)
    // TODO: insert your ON_PARSE_COMMAND() and
    // ON_PARSE_COMMAND_PARAMS() here to hook up your commands.
    // For example:
    ON_PARSE_COMMAND(Default, CISAPI_TraDienthoaiExtension,
ITS_EMPTY)
    DEFAULT_PARSE_COMMAND(Default, CISAPI_TraDienthoaiExtension)
END_PARSE_MAP(CISAPI_TraDienthoaiExtension)
//đặc tả hàm TraDthoai() phục vụ việc tra điện thoại
    ON_PARSE_COMMAND(TraDthoai,CISAPI_TraDienthoaiExtension,ITS_PSTR
ITS_PSTR ITS_PSTR)
    ON_PARSE_COMMAND_PARAMS("tentbao sodthoai diachi")
END_PARSE_MAP(CISAPI_TraDienthoaiExtensio)
```

Tạo ISAPI Extension tra điện thoại

5. Viết code cho từng tác vụ vừa đặc tả :

```
void CISAPI_TraDienthoaiExtension::TraDthoai(CHttpServerContext* pCtx,  
LPCTSTR tentbao,LPCTSTR sodthoai,LPCTSTR diachi) {  
StartContent(pCtx);  
WriteTitle(pCtx);  
*pCtx << "<h2>Kết quả tra danh bạ điện thoại của bạn :</h2>\n";  
//gọi hàm Search_Print tìm và xuất kết quả tìm kiếm  
Search_Print(pCtx, tentbao, sodthoai, diachi);  
*pCtx << "<a href=ISAPI_TraDienthoai.htm>Quay về trang chủ</a>\n";  
EndContent(pCtx);  
}
```

Thêm lệnh import sau vào đầu file mã nguồn ISAPI_TraDienthoai.cpp để có thể dùng các đối tượng ADO truy xuất database :

```
#import "C:\Program Files\Common Files\SYSTEM\ADO\msado15.dll"  
no_namespace rename( "EOF", "adoEOF" )
```



Tạo ISAPI Extension tra điện thoại

Viết code cho hàm dịch vụ Search_Print() :

```
// Truy xuất database  
void CISAPI_TraDienthoaiExtension::Search_Print (CHttpServerContext* pCtxt,  
LPCTSTR tentbao, LPCTSTR sodthoai, LPCTSTR diachi) {  
    //nội dung y như hàm Search_Print() trong ứng dụng CGI ở phần trước.  
}
```

6. Mở file ISAPI_TraDienthoai.h, rồi thêm 2 lệnh khai báo hàm vừa viết vào class CISAPI_TraDienthoaiExtension :

```
class CISAPI_TraDienthoaiExtension : public CHttpServer {  
public:  
    CISAPI_TraDienthoaiExtension();  
    ~CISAPI_TraDienthoaiExtension();  
    void CISAPI_TraDienthoaiExtension::TraDthoai(CHttpServerContext* pCtxt,  
        LPCTSTR tentbao, LPCTSTR sodthoai, LPCTSTR diachi);  
    void CISAPI_TraDienthoaiExtension::Search_Print (CHttpServerContext* pCtxt,  
        LPCTSTR tentbao, LPCTSTR sodthoai, LPCTSTR diachi);  
    ...  
}
```



Tạo ISAPI Extension tra điện thoại

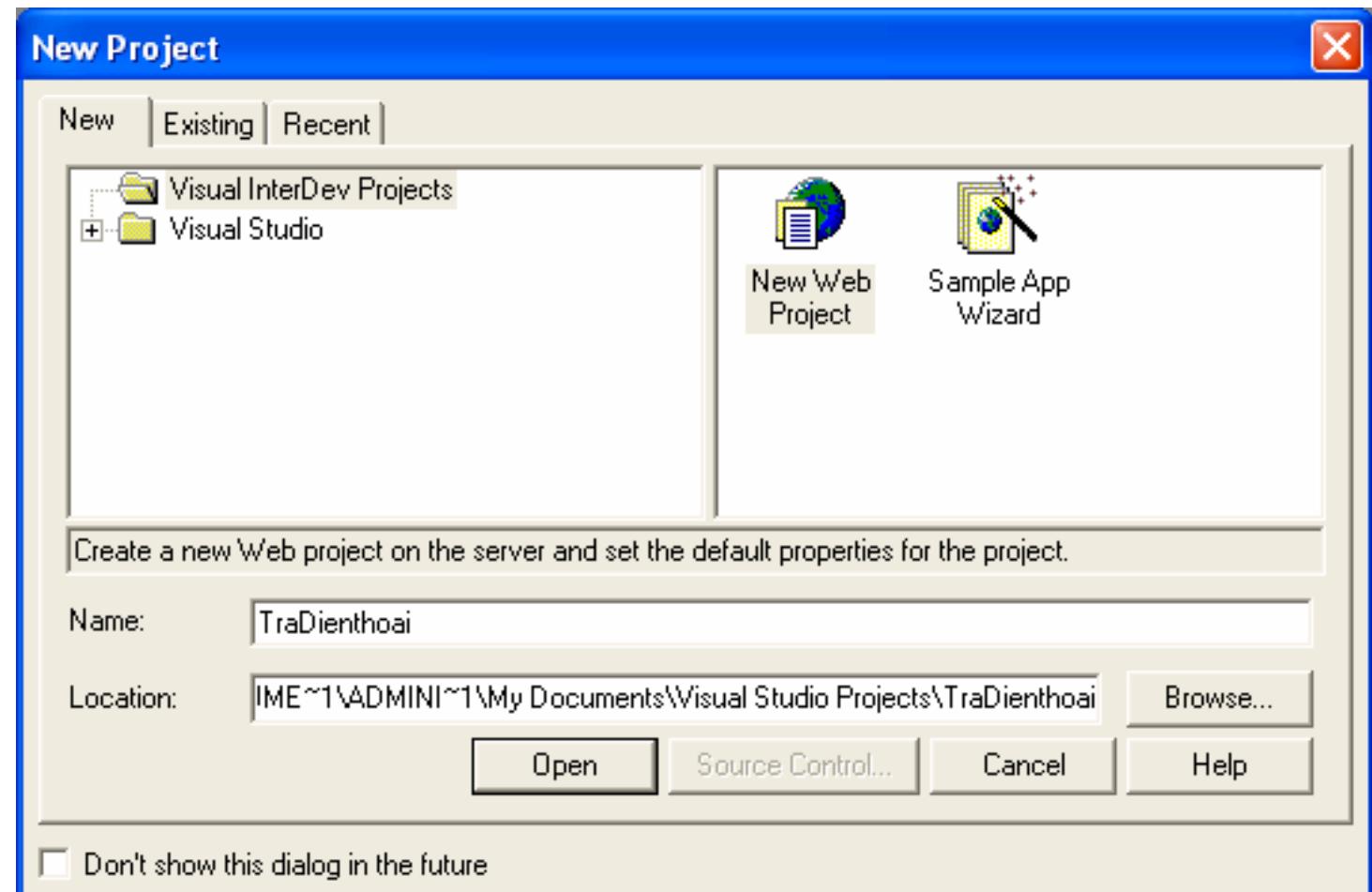
7. Chọn menu Build.Rebuild All để dịch các file mã nguồn và tạo file khả thi tương ứng. Nếu có lỗi từ vựng và cú pháp thì sửa. Sau khi tạo được file khả thi, hãy debug để tìm lỗi run-time nếu có.

Sau khi hoàn thành, ta có được file ISAPI khả thi tên là ISAPI_TraDienthoai.dll trong thư mục Debug hay Release tùy thuộc vào chế độ dịch (chọn menu Build.Set Active Configuration...). File ISAPI sẽ được “publish” lên website tra danh bạ mà ta sẽ tạo ra theo đặc tả của các slide kế tiếp.



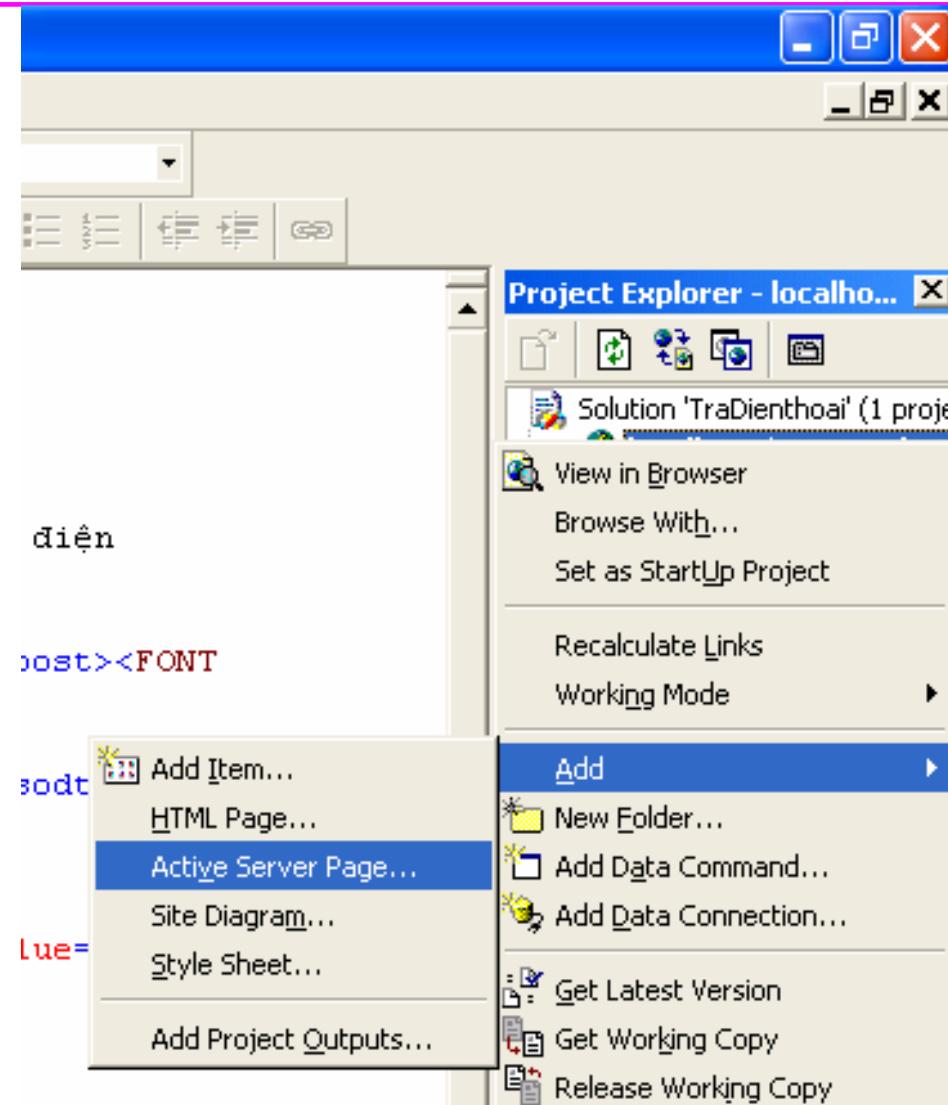
Tạo website tra điện thoại bằng InterDev

1. Chạy InterDev, khi cửa sổ New Project hiển thị, chọn Location chưa Project, nhập tên Project là TraDienthoai, ấn button Open rồi trả lời các yêu cầu trên các cửa sổ Wizard để hoàn tất việc tạo Project (hoặc mở lại Project đã tạo trong phần CGI).



Tạo website tra điện thoại bằng InterDev

2. Dời chuột về cửa sổ Project, ấn phải chuột vào nút gốc để hiển thị menu pop-up chứa các lệnh, chọn mục Add.HTML Page..., nhập tên file là ISAPI_TraDienthoai để tạo home page cho Website tra danh bạ điện thoại.



Tạo website tra điện thoại bằng InterDev

3. Thực hiện các thao tác thiết kế trực quan để tạo trang Web có dạng như hình dưới (chứa tiêu đề chính và 1 form. Form có 3 textbox nhập liệu và 1 button “Bắt đầu tra”).

Demo việc tra điện thoại

Tên thuê bao :

Số điện thoại :

Địa chỉ :

Bắt đầu tra



Mã nguồn HTML của trang ISAPI_TraDienthoai.htm

```
<HTML> <HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<TITLE></TITLE>
</HEAD>
<BODY>
<P align=center><FONT face=WoodType-Demi size=5>Demo việc tra điện thoại</FONT></P>
<P><FONT face=VnWoodType-Demi size=5>
<FORM ACTION="ISAPI_TraDienthoai.DLL?" METHOD="POST">
<INPUT TYPE=HIDDEN NAME="MfcISAPICommand" VALUE="TraDthoai">
<FONT face=VnTimes size=3>Tên thuê bao :&nbsp;
<INPUT id=tentbao name=tentbao style="width: 269px; height: 22px"></FONT>
<P><FONT face=VnTimes size=3>Số điện thoại :&nbsp;
<INPUT id=sodthoai name=sodthoai></FONT></P>
<P><FONT face=VnTimes size=3> Địa chỉ :&nbsp;
<INPUT id=diachi name=diachi style="width: 373px; height: 22px"></FONT></P>
<P><FONT face=VnTimes size=3>
<INPUT type=submit value="Bắt đầu tra"></FONT></P></FORM></FONT>
<P></P>
</BODY> </HTML>
```

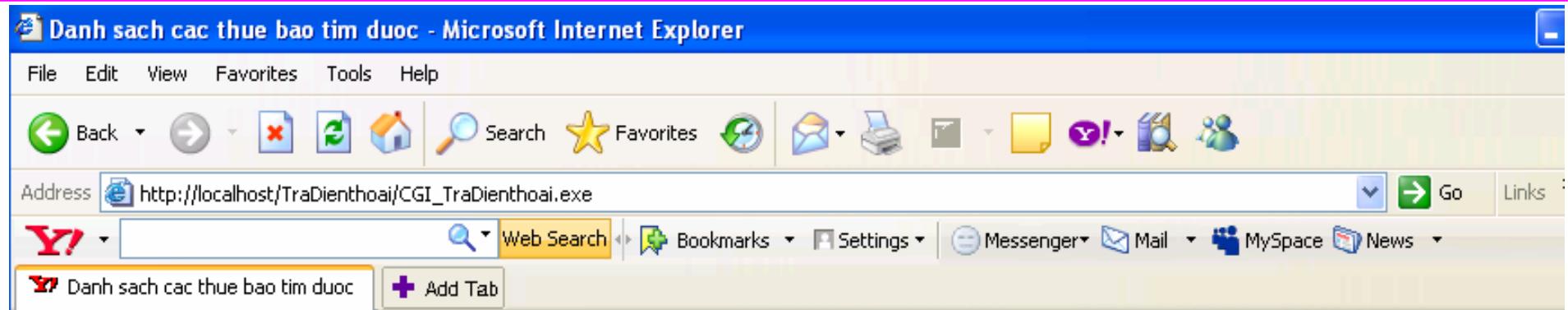


Tạo website tra điện thoại bằng InterDev

4. Copy module ISAPI_TraDienthoai.dll vào thư mục chứa Website.
5. Mở rộng nội dung của cây Project, chọn mục ISAPI_TraDienthoai.htm, ấn phải chuột trên nó rồi chọn mục “Set As Start Page”
6. thử chạy Website bằng cách chọn menu Debug.Start. Khi trang Web chứa form nhập liệu hiển thị, người dùng sẽ nhập các thông tin về account mà mình muốn tìm kiếm rồi ấn button “Bắt đầu tra”. Kết quả tra cứu sẽ có dạng như hình ở slide kế tiếp.



Tạo website tra điện thoại bằng InterDev



Kết quả tra danh bạ điện thoại của bạn :

So thu tu	Ten thue bao	So dien thoai	Địa chỉ
1	Nguyen Van Hiep	8644717	8A Lu Gia, P15, Q11, Tp.HCM
2	Huynh Thoai Loan	9070437	245/4 Ly Thuong Kiet, P15, Q11, Tp.HCM
3	Nguyen Luong Hoa	9024437	240 Tran Binh Trong, P4, Q5, Tp.HCM
4	Nguyen Lac Han	2030478	157 Hoang Dieu, P1, Q1, Tp.HCM

[Quay về trang chủ](#)



8.4 Tổng quát về ASP

- ASP (Active Server Page) là công nghệ script chạy ở máy server của Microsoft tương tự với JSP của SUN hay PHP.
- Hiện ASP có thể chạy trên IIS của Microsoft.
- Một trang ASP là trang HTML có tăng cường 1 số đoạn script viết bằng VBScript hay JavaScript.
- Để server nhận biết được file nào là ASP, ta luôn dùng extension đã qui định theo cấu hình (mặc định là *.asp).
- Đoạn script ASP được đóng trong tag `<% %>` hay tag `<SCRIPT ... LANGUAGE=vbscript RUNAT=Server>`
.....
`</SCRIPT>`
- ASP thường dùng các đối tượng COM chính quy để xử lý các giải thuật phức tạp.



Nhập/xuất thông tin của ASP

- Đối tượng định sẵn có tên là **Request** sẽ hỗ trợ việc nhập thông tin từ client gửi về từ URL yêu cầu hay từ Form nhập liệu. Thí dụ tham số client gửi về có tên là txtTenthuebao, muốn ASP đọc giá trị của tham số này, ta viết lệnh như sau :

```
tentbao = Request.Form("txtTenthuebao")
```

- Đối tượng định sẵn có tên là **Response** sẽ hỗ trợ việc xuất thông tin từ đoạn code ASP về client. Thí dụ đoạn script ASP sau sẽ xuất hàng chữ “Đây là kết quả thực hiện đoạn script ASP” :

```
<%
```

```
Response.Write "<h3 align=center>Đây là kết quả thực hiện đoạn  
script ASP.</h3>"
```

```
%>
```



Thí dụ xây dựng 1 website bằng ASP

Yêu cầu : tạo 1 website dùng công nghệ ASP cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

Phân tích : cần 2 module chức năng :

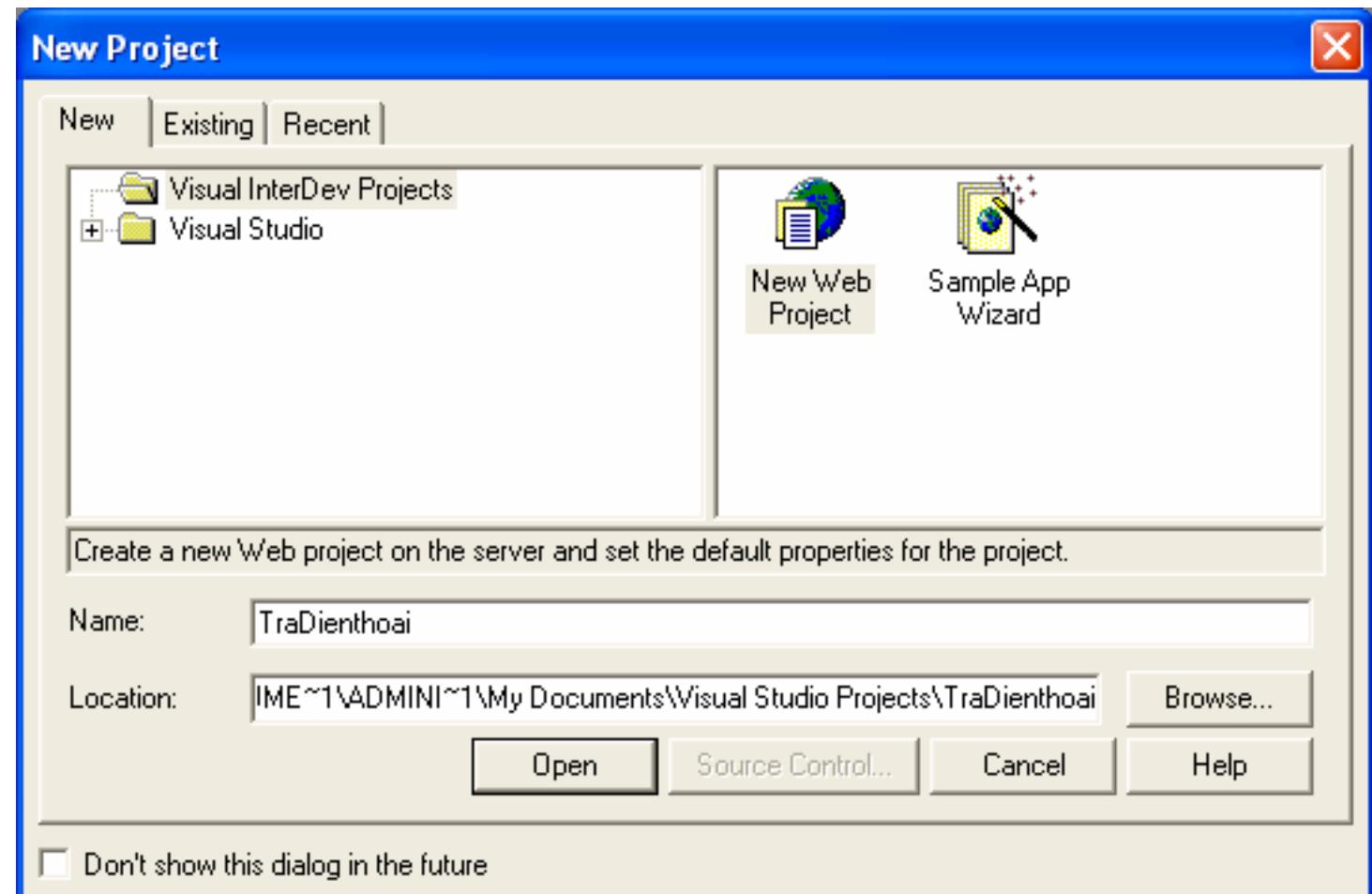
- module chứa form nhập thông tin về các account cần tra, module này có thể là 1 trang HTML thuần túy.
- và module truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm, module này là 1 trang ASP theo yêu cầu.

Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình InterDev của Microsoft. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



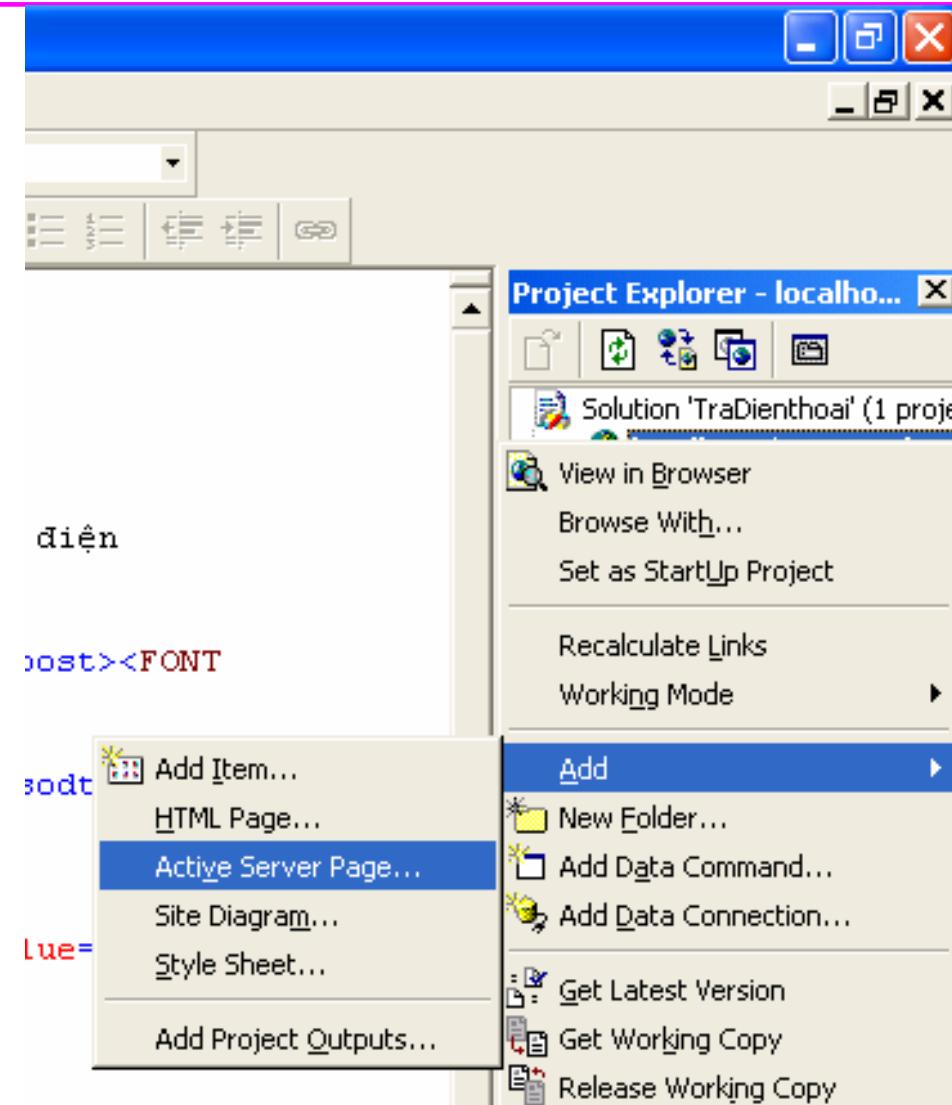
Tạo website tra điện thoại bằng InterDev

1. Chạy InterDev, khi cửa sổ New Project hiển thị, chọn Location chưa Project, nhập tên Project là TraDienthoai, ấn button Open rồi trả lời các yêu cầu trên các cửa sổ Wizard để hoàn tất việc tạo Project (hoặc mở lại Project đã tạo trong phần CGI).



Tạo website tra điện thoại bằng InterDev

2. Dời chuột về cửa sổ Project, ấn phải chuột vào nút gốc để hiển thị menu pop-up chứa các lệnh, chọn mục Add.HTML Page..., nhập tên file là ASP_TraDienthoai để tạo home page cho Website tra danh bạ điện thoại.



Tạo website tra điện thoại bằng InterDev

3. Thực hiện các thao tác thiết kế trực quan để tạo trang Web có dạng như hình dưới (chứa tiêu đề chính và 1 form. Form có 3 textbox nhập liệu và 1 button “Bắt đầu tra”).

Demo việc tra điện thoại

Tên thuê bao :

Số điện thoại :

Địa chỉ :

Bắt đầu tra



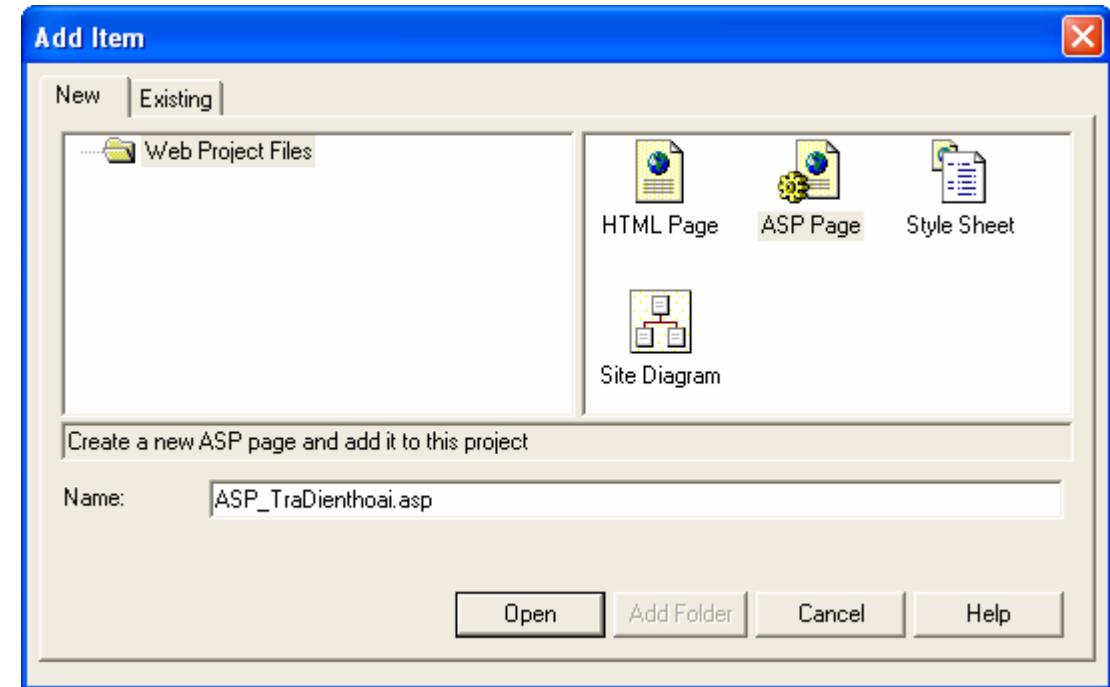
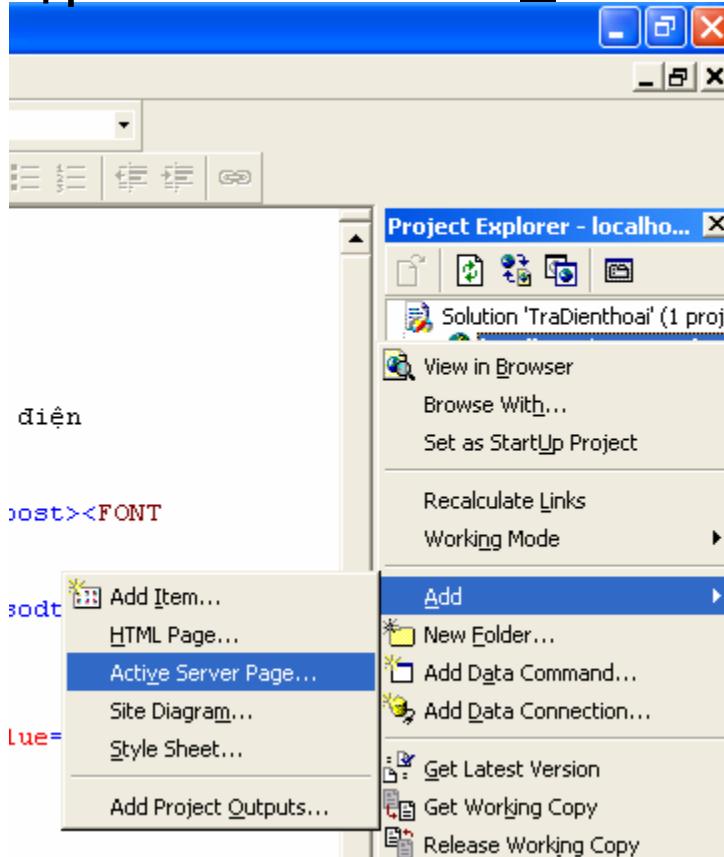
Mã nguồn HTML của trang ASP_TraDienthoai.htm

```
<HTML> <HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<TITLE></TITLE>
</HEAD>
<BODY>
<P align=center><FONT face=WoodType-Demi size=5>Demo việc tra điện thoại</FONT></P>
<P><FONT face=VnWoodType-Demi size=5>
<FORM name=FORM1 action=ASP_TraDienthoai.asp method=post>
<FONT face=VnTimes size=3>Tên thuê bao :&nbsp;
<INPUT id=tentbao name=tentbao style="width: 269px; height: 22px"></FONT>
<P><FONT face=VnTimes size=3>Số điện thoại :&nbsp;
<INPUT id=sodthoai name=sodthoai></FONT></P>
<P><FONT face=VnTimes size=3> Địa chỉ :&nbsp;
<INPUT id=diachi name=diachi style="width: 373px; height: 22px"></FONT></P>
<P><FONT face=VnTimes size=3>
<INPUT type=submit value="Bắt đầu tra"></FONT></P></FORM></FONT>
<P></P>
</BODY> </HTML>
```



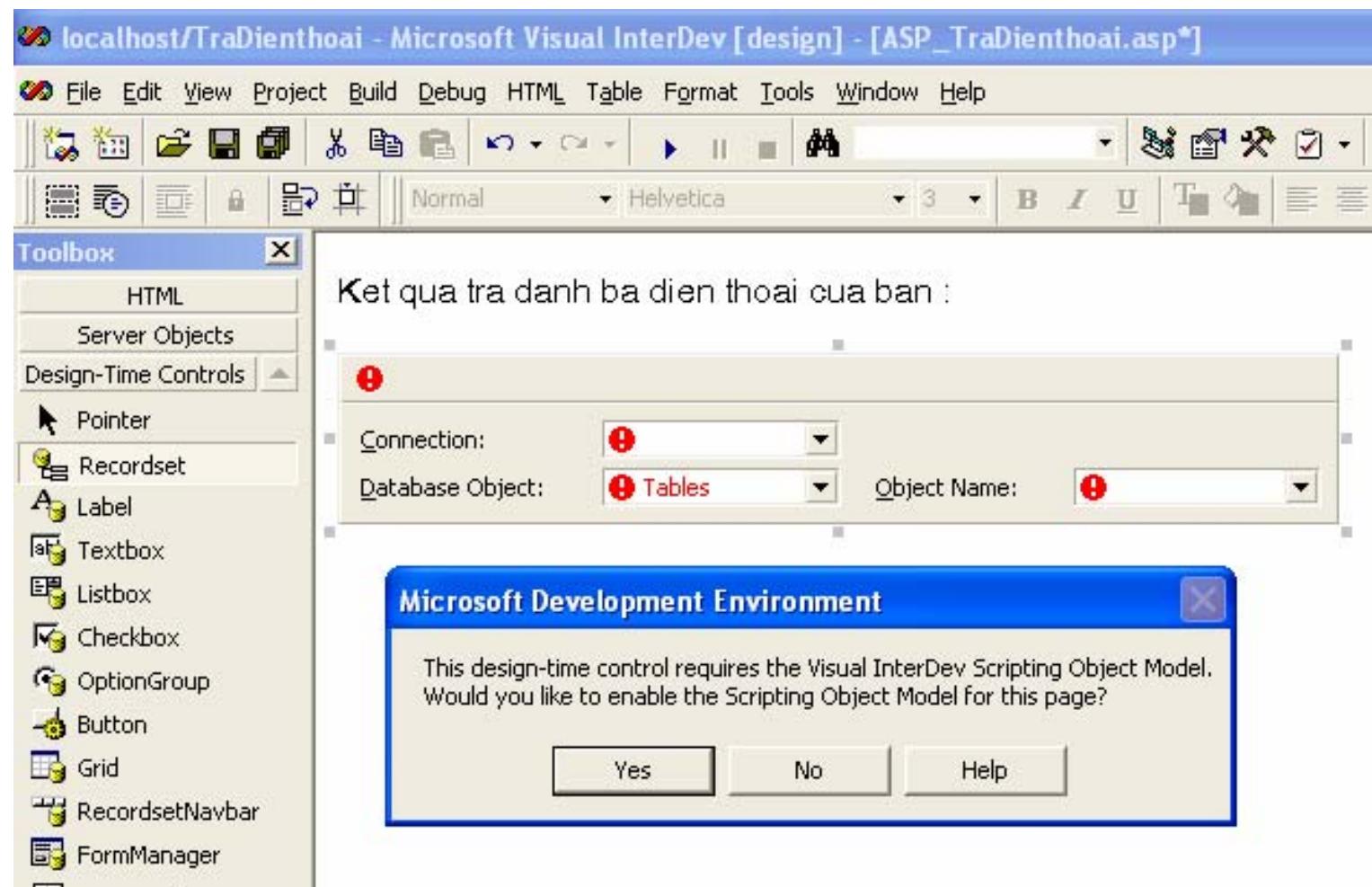
Tạo website tra điện thoại bằng InterDev

4. Dời chuột về cửa sổ Project, ấn phải chuột vào nút gốc để hiển thị menu pop-up chứa các lệnh, chọn mục Add.Active Server Page..., nhập tên file là ASP_TraDienthoai.asp :



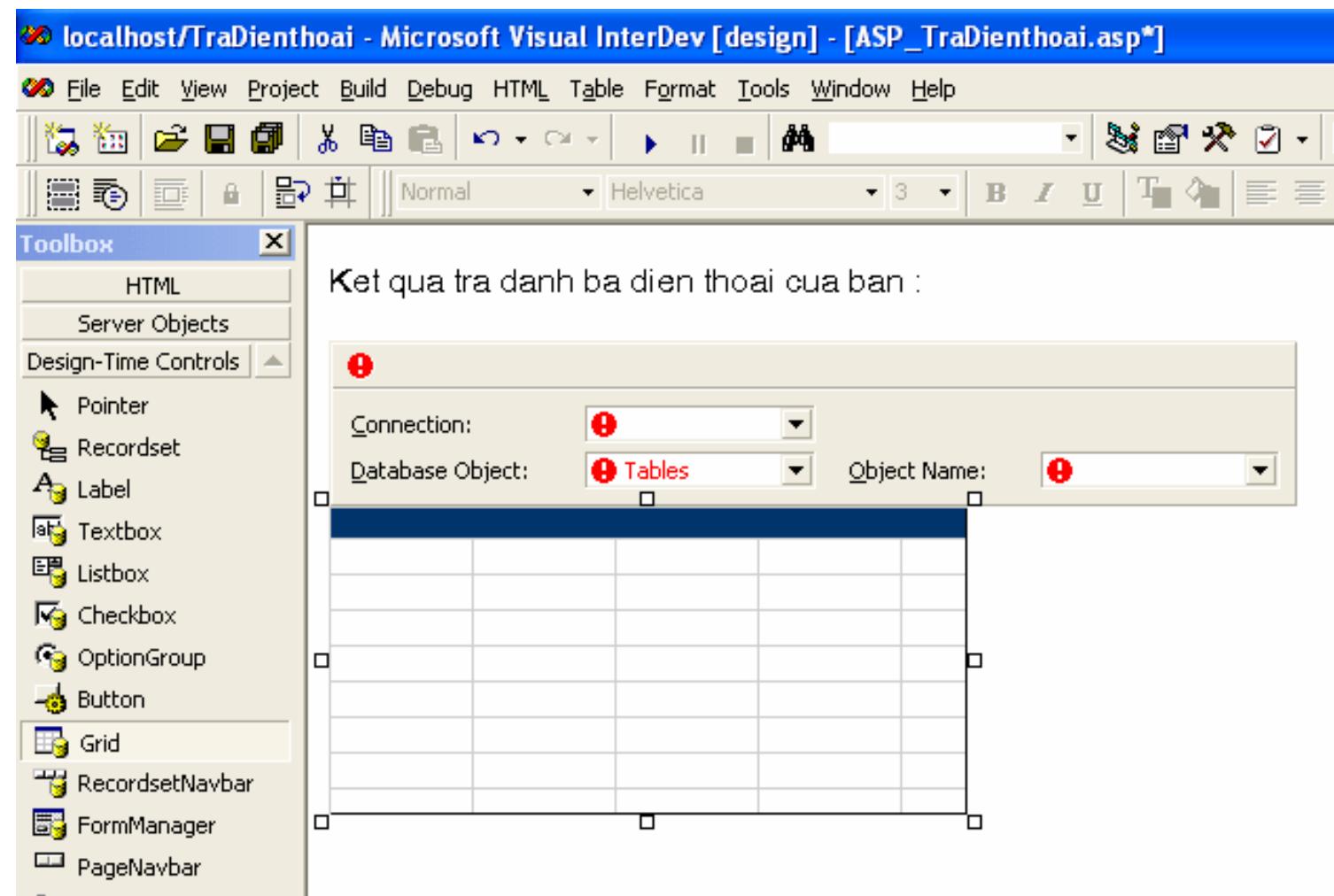
Tạo website tra điện thoại bằng InterDev

5. Soạn nội dung cho trang ASP như hình bên, để tạo đối tượng Recordset, ta hiển thị của sổ ToolBox, chọn và drag phần tử Recordset từ Toolbox vào vị trí thích hợp trong trang ASP, sau đó ấn button Yes trong của sổ Warning :



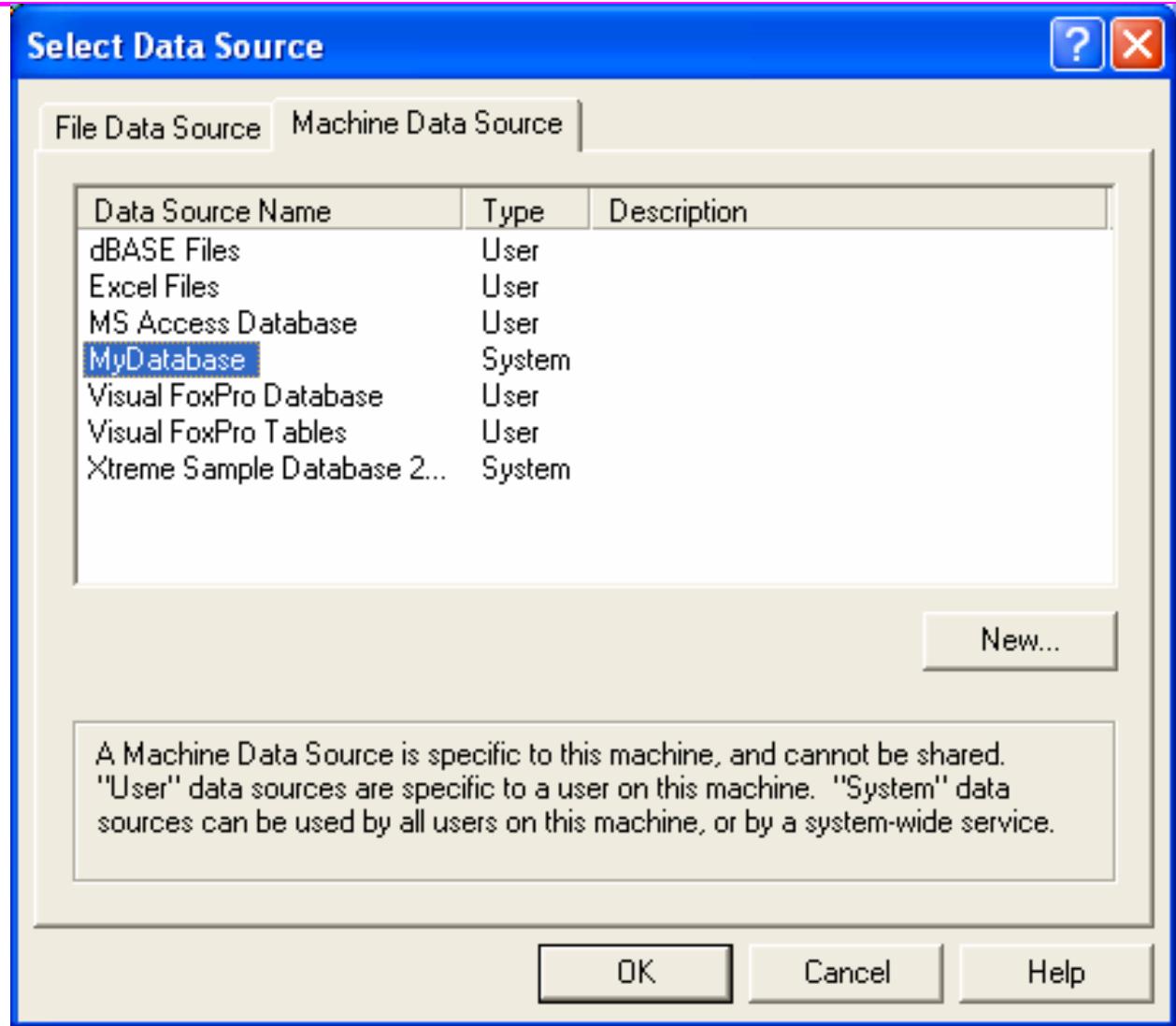
Tạo website tra điện thoại bằng InterDev

6. Tiếp tục tạo đối tượng DataGrid để hiển thị kết quả tìm kiếm, ta hiển thị cửa sổ ToolBox, chọn và drag phần tử Grid từ Toolbox vào vị trí thích hợp trong trang ASP.



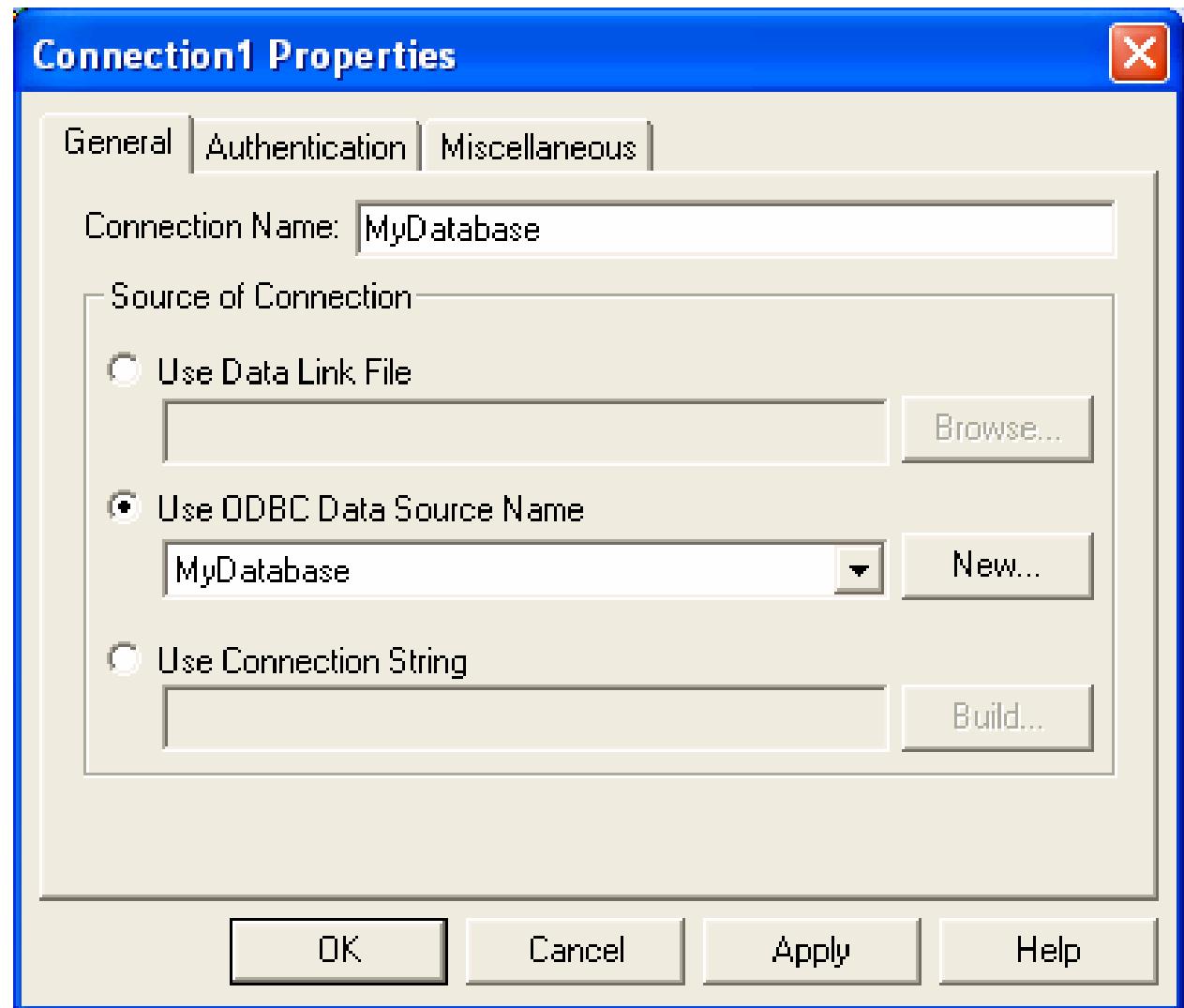
Tạo website tra điện thoại bằng InterDev

7. Tạo đối tượng Connection nối kết với database cần truy xuất cho Website bằng cách dời chuột về cửa sổ Project, ấn phải chuột vào nút gốc để hiển thị menu pop-up chứa các lệnh, chọn mục Add Data Connection. Cửa sổ hình bên sẽ hiển thị, chọn tab “Machine Data Source”, chọn DSN MyDatabase rồi ấn OK.



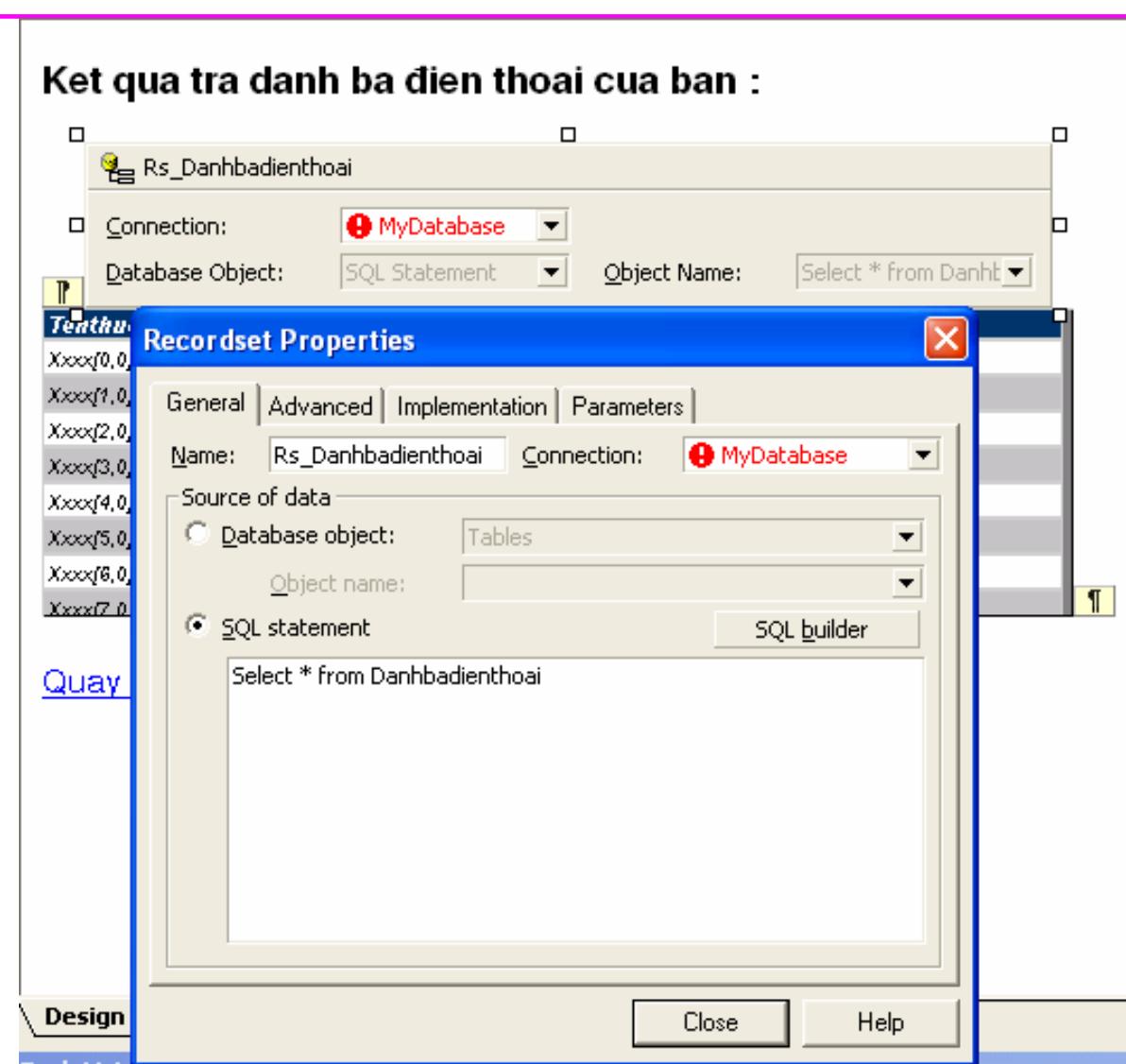
Tạo website tra điện thoại bằng InterDev

8. Khi cửa sổ hình bên hiển thị, hãy tham khảo và hiệu chỉnh các thuộc tính cần thiết, thí dụ hãy đặt tên nhận dạng cho Connection là MyDatabase rồi ấn button Apply, OK.



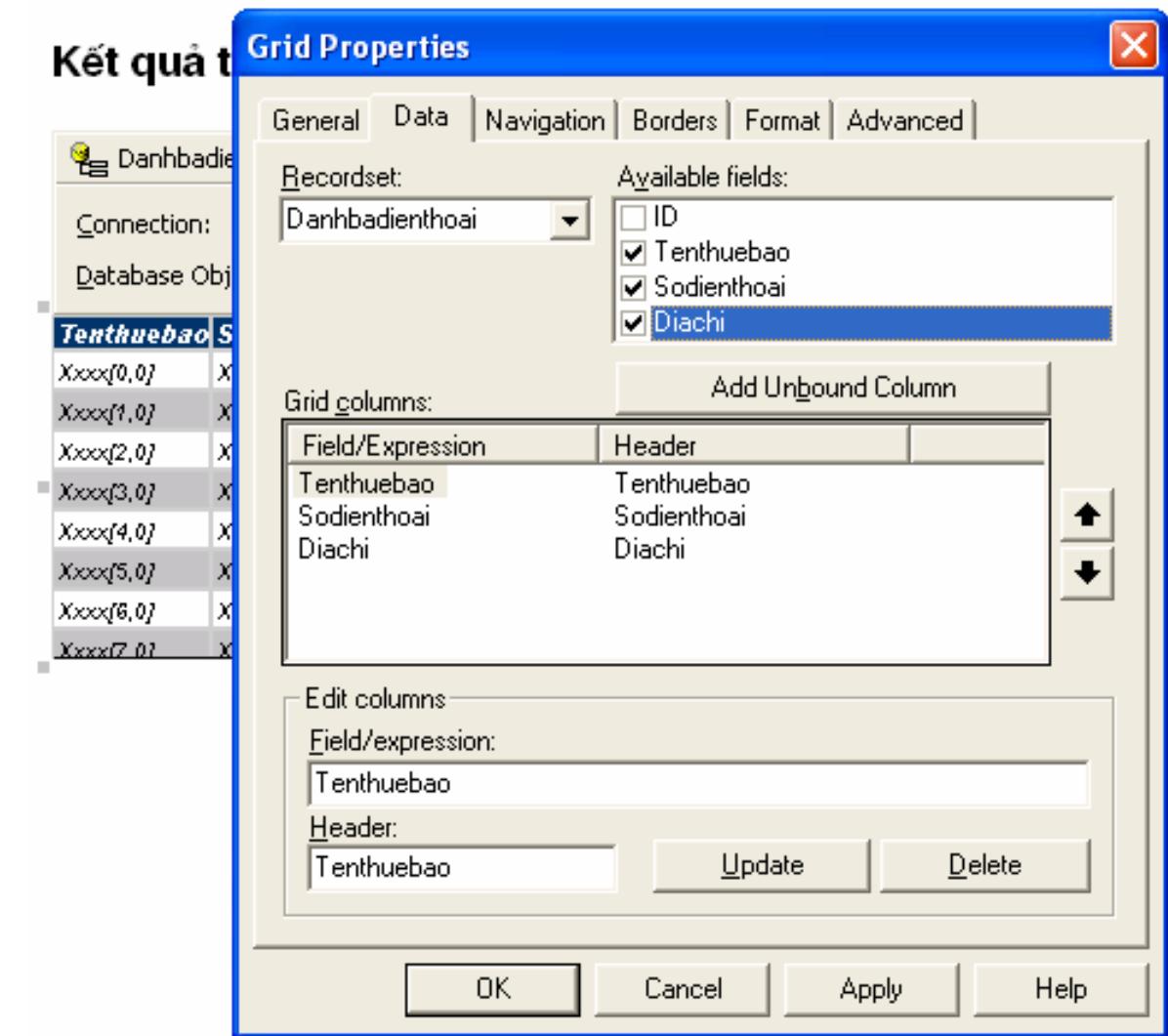
Tạo website tra điện thoại bằng InterDev

9. Quay lại trang ASP đang soạn thảo, chọn đối tượng Recordset, chọn menu View.Properties Page để hiển thị cửa sổ thuộc tính của Recordset. Hãy xem và thay đổi 1 số thuộc tính cần thiết, thí dụ đặt tên nhận dạng là Rs_Danhbdienthoai, chọn chế độ “SQL statement”, nhập lệnh truy vấn như hình bên.



Tạo website tra điện thoại bằng InterDev

10. Chọn đối tượng DataGrid, chọn menu View.Properties Page để hiển thị cửa sổ thuộc tính của DataGrid. Hãy xem và thay đổi 1 số thuộc tính cần thiết, thí dụ đặt tên nhận dạng là DbGrid, chọn Recordset kết hợp, chọn các field cần hiển thị, rồi ấn button Apply, OK.



Tạo website tra điện thoại bằng InterDev

11. Thiết lập lại

1 cách trực quan **Kết quả tra danh bạ điện thoại của bạn :**

độ rộng của từng cột trong datagrid để hiển thị nội dung của cột đó cho phù hợp (thí dụ cho cột Địa chỉ rộng càng nhiều càng tốt. Tạo 1 liên kết về trang chủ ở cuối trang [Quay về trang chủ](#) ASP.

Danh bạ điện thoại		
Connection:	Database Object:	Object Name:
Tên thuê bao	Số điện thoại	Địa chỉ
Xxxx[0,0]	Xxxx[0,1]	Xxxx[0,2]
Xxxx[1,0]	Xxxx[1,1]	Xxxx[1,2]
Xxxx[2,0]	Xxxx[2,1]	Xxxx[2,2]
Xxxx[3,0]	Xxxx[3,1]	Xxxx[3,2]
Xxxx[4,0]	Xxxx[4,1]	Xxxx[4,2]
Xxxx[5,0]	Xxxx[5,1]	Xxxx[5,2]
Xxxx[6,0]	Xxxx[6,1]	Xxxx[6,2]
Xxxx[7,0]	Xxxx[7,1]	Xxxx[7,2]



Tạo website tra điện thoại bằng InterDev

12. Chọn tab Source để hiển thị cửa sổ soạn code trang ASP, chọn tab Toolbox để hiển thị cửa sổ Toolbox. Trong cửa sổ Toolbox, mở rộng, duyệt tìm trong cây đối tượng và ấn kép chuột vào mục onbeforeopen của đối tượng

Rs_Danhbadienthoai để tạo thủ tục xử lý sự kiện tương ứng. Sau đó viết code cho thủ tục như các slide kế tiếp.

The screenshot shows the Microsoft Development Environment (InterDev) interface. The title bar reads "Miscellaneous Files - Microsoft Development Environment [design] - [ASP_TraDienthoai.asp*]". The menu bar includes File, Edit, View, Project, Build, Debug, Tools, Window, Help. The toolbar has various icons for file operations. The "Design" tab is selected in the top right corner.

The left side features the "Script Outline" window, which displays a tree structure of script objects. Under "Server Objects & Events", the "Danhbadienthoai" object is expanded, showing events like onafterupc, onbeforeopen (which is selected), onbeforeupc, onbeforeupl, ondatasetc, ondatasetc, onrowenter, and onrowexit. Other objects listed include DbGrid, Request, Response, ScriptingConte, Server, and Session.

The main code editor area contains the following VBScript code:

```
<%@ Language=VBScript %>
<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file=_ScriptLibrary/pm.asp-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<H3>Ket qua tra danh ba dien thoai cua ban :</H3>
<P>
```

Below the code editor is a preview pane titled "Danhbadienthoai" showing a table with three columns: Tenthuebao, Sodienthoai, and Diachi. The table contains 8 rows of placeholder data:

Tenthuebao	Sodienthoai	Diachi
Xxxx[0,0]	Xxxx[0,1]	Xxxx[0,2]
Xxxx[1,0]	Xxxx[1,1]	Xxxx[1,2]
Xxxx[2,0]	Xxxx[2,1]	Xxxx[2,2]
Xxxx[3,0]	Xxxx[3,1]	Xxxx[3,2]
Xxxx[4,0]	Xxxx[4,1]	Xxxx[4,2]
Xxxx[5,0]	Xxxx[5,1]	Xxxx[5,2]
Xxxx[6,0]	Xxxx[6,1]	Xxxx[6,2]
Xxxx[7,0]	Yyyy[7,1]	Yyyy[7,2]

At the bottom of the preview pane are buttons for "Design", "Source" (which is selected), and "Quick View".



Tạo website tra điện thoại bằng InterDev

```
Sub Rs_Danhbadienthoai_onbeforeopen()
fsqlstart = 1
tentbao = Request.Form("tentbao")
sodthoai = Request.Form("sodthoai")
diachi = Request.Form("diachi")
newSQL="SELECT * FROM DanhbadienThoai"
if tentbao <> "" then
    newSQL = newSQL + " where tenthuebao like '%" + tentbao + "%'"
    fsqlstart = 0
end if
if sodthoai <> "" then
    if fsqlstart = 1 then
        newSQL = newSQL + " where sodienthoai like '%" + sodthoai + "%'"
    else
        newSQL = newSQL + " and sodtienhoai like '%" + sodthoai + "%'"
    end if
    fsqlstart = 0
end if
```

Tạo website tra điện thoại bằng InterDev

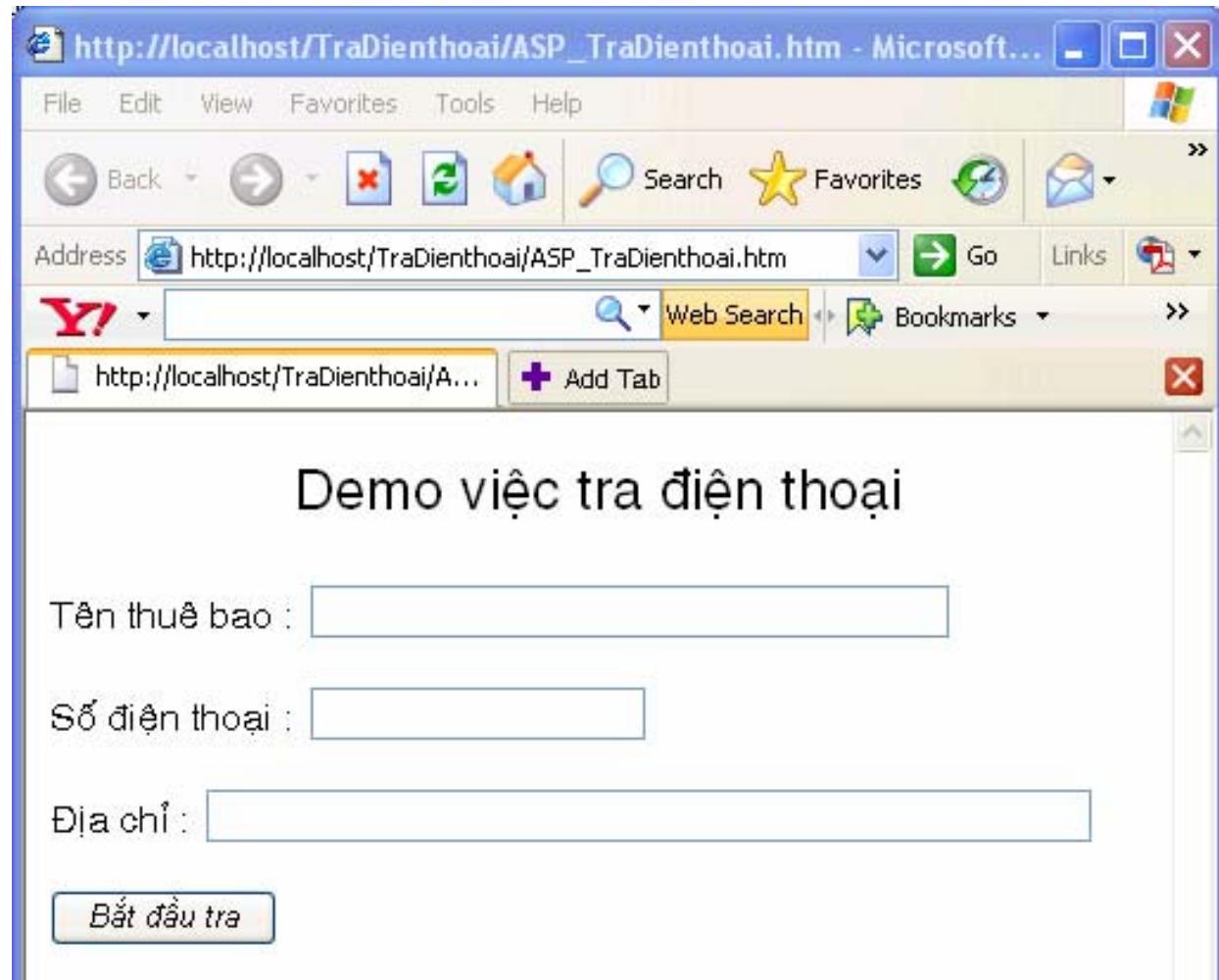
```
if diachi <> "" then
    if fsqlstart = 1 then
        newSQL = newSQL + " where diachi like '%" + diachi + "%'"
    else
        newSQL = newSQL + " and diachi like '%" + diachi + "%'"
    end if
    fsqlstart = 0
end if
Rs_Danhbadienthoai.setSQLText(newSQL)
End Sub
```



Tạo website tra điện thoại bằng InterDev

13. Vào cửa sổ Project, chọn mục ASP_TraDienthoai.htm, ấn phải chuột trên nó rồi chọn mục “Set As Start Page”

14. thử chạy Website bằng cách chọn menu Debug.Start. Khi trang Web chưa form nhập liệu hiển thị, người dùng sẽ nhập các thông tin về account mà mình muốn tìm kiếm rồi ấn button “Bắt đầu tra”.



Tạo website tra điện thoại bằng InterDev

15. Kết quả tra cứu có dạng như hình bên.

The screenshot shows a Microsoft Internet Explorer window with the title bar "http://localhost/TraDienthoai/ASP_TraDienthoai.asp - Microsoft Internet Explorer". The address bar contains the same URL. Below the toolbar, there is a search bar with "Web Search" and a tab bar showing the current page. The main content area displays a table titled "Ket qua tra danh ba dien thoai cua ban :". The table has three columns: "Tenhuebao", "Sodienthoai", and "Diachi". The data is as follows:

Tenhuebao	Sodienthoai	Diachi
Nguyen Van Hiep	8644717	8A Lu Gia, P15, Q11, Tp.HCM
Huynh Thoai Loan	9070437	245/4 Ly Thuong Kiet, P15, Q11, Tp.HCM
Nguyen Luong Hoa	9024437	240 Tran Binh Trong, P4, Q5, Tp.HCM
Nguyen Lac Han	2030478	157 Hoang Dieu, P1, Q1, Tp.HCM

Below the table are navigation buttons: '<', '<<', '>', and '>'. To the right, it says "Page : 1 of 1". At the bottom, there is a link "Quay ve trang chu".



Môn học : Lập trình mạng

Chương 9

LẬP TRÌNH WEB CHẠY Ở SERVER BẰNG PHP

9.1 Tổng quát về PHP

9.2 Cách viết script PHP

9.3 Thí dụ xây dựng 1 website bằng PHP



9.1 Tổng quát về PHP

- PHP (Pre HyperText Processor) là ngôn ngữ script chạy ở máy server.
- PHP dùng cú pháp giống như C++.
- Hiện nay PHP có thể chạy trên nhiều HĐH khác nhau từ Windows, Linux, OS2, MacOS,...
- Ý tưởng là viết module thông dịch source PHP và "hook" vào Webserver.
- PHP cung cấp rất nhiều nhóm hàm API để tiện lập trình trên nhiều lĩnh vực khác nhau, trong đó nhóm các hàm API "odbc_x" cho phép truy xuất đến database bất kỳ theo chuẩn giao tiếp ODBC.
- Quá trình cài đặt và cấu hình chế độ làm việc cho PHP phụ thuộc vào HĐH mà ta muốn cài PHP. Trên Windows (IIS, PWS), bạn chỉ cần chạy trình installer.exe rồi trả lời trực quan 1 số cửa sổ Wizard.



9.2 Cách viết script PHP

- Một trang PHP là trang HTML có tăng cường 1 số đoạn script viết bằng PHP. Để module xử lý PHP nhận biết được file nào mình xử lý, ta phải dùng extension đã qui định theo cấu hình (mặc định là *.php).
- Đoạn script PHP được đóng trong tag `<?php ?>`.
- Cú pháp PHP rất giống với C++, PHP phân biệt chữ HOA và thường.
- Biến PHP luôn bắt đầu bằng ký tự \$, thí dụ \$name, \$database,...
- Giống như nhiều ngôn ngữ script khác, PHP không kiểm tra kiểu dữ liệu, những kiểu cơ bản là Boolean, Integer, floating point, String, Array, Object...
- PHP cho phép ép (casting) kiểu dữ liệu : `$var = (int) "123abc";`



Các toán tử, quyền ưu tiên, tính kết hợp

Operators

!, ~, ++, --, @, (the casting operators)

*, /, %

+, - .

<<, >>

<, <=, >=, >

==, !=

&

^

|

&&

||

? : (conditional operator)

=, +=, -=, *=, /=, %=, ^=, .=, &=, |=

And

Xor

Or

P

16

15

14

13

12

11

10

9

8

7

6

5

4

3

2

1

A

Right

Left

Left

Left

Non-associative

Non-associative

Left



Các lệnh thực thi PHP

Lệnh gán :

```
$i = expr;
```

Lệnh if :

```
if (condition) statement1 else statement2
```

Lệnh switch :

```
switch(expr) {
```

```
case expr:
```

```
    statements
```

```
    break;
```

...

```
default:
```

```
    statements
```

```
    break;
```

```
}
```



Các lệnh thực thi PHP (tt)

Lệnh while :

```
while (condition) {  
    statements  
}
```

Lệnh do ... while :

```
do {  
    statements  
} while (condition);
```

Lệnh for :

```
for (start_expr; cond_expr; iter_expr) {  
    statements  
}
```

Định nghĩa function :

```
function soundcheck($a, $b, $c) {  
    return "Testing, $a, $b, $c";  
}
```



Các biến định sẵn của PHP

\$GLOBALS : dãy các tham khảo đến các biến toàn cục, mỗi phần tử được nhận dạng bằng chỉ số, chỉ số thường là tên biến tương ứng.

\$_SERVER : dãy các biến được thiết lập bởi Web server hay ứng dụng nào đó khởi động môi trường thực thi cho đoạn script hiện hành, biến này tương tự với biến dãy \$HTTP_SERVER_VARS.

\$_GET : dãy các tham số được gửi tới script từ chuỗi query URL (từ Web browser), biến này tương tự với biến \$HTTP_GET_VARS. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên tham số đó.

Thí dụ : \$_GET["txtDiachi"] sẽ truy xuất tham số trong chuỗi URL yêu cầu có tên là txtDiachi được gửi về dùng method GET.

\$_POST : dãy các tham số được gửi tới script từ lệnh POST, biến này tương tự với biến \$HTTP_POST_VARS. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên tham số đó, thí dụ : \$_POST["txtDiachi"].

Thí dụ : \$_POST["txtDiachi"] sẽ truy xuất tham số trong form yêu cầu có tên là txtDiachi được gửi về dùng method POST.



Các biến định sẵn của PHP (tt)

\$_COOKIE : dãy các biến được gửi tới script từ cơ chế cookie của giao thức HTTP, biến này tương tự với biến **\$HTTP_COOKIE_VARS**. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên của cookie đó.

\$_FILES : dãy các biến được gửi tới script từ lệnh " HTTP post file uploads", biến này tương tự với biến **\$HTTP_POST_FILES**. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên của file đó.

\$_ENV : dãy các biến được gửi tới script từ môi trường, biến này tương tự với biến **\$HTTP_ENV_VARS**. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên của biến môi trường đó.

\$_REQUEST : dãy các tham số được gửi tới script từ các cơ chế gửi tham số GET, POST và COOKIE. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên tham số đó.

\$_SESSION : dãy các biến được đăng ký cho session script hiện hành. Truy xuất tham số thông qua chỉ số, chỉ số thường là tên biến đó. Biến này tương tự với biến **\$HTTP_SESSION_VARS**.



Các hàm API của PHP

Do khả năng của ngôn ngữ script PHP không đủ mạnh như ngôn ngữ lập trình chính quy nên nó cần 1 "đại gia" phía sau hỗ trợ. "Đại gia" phía sau PHP là các hàm API PHP, rất nhiều và đa dạng về chức năng. Hiện có trên 100 nhóm hàm API PHP khác nhau, mỗi nhóm gồm nhiều hàm cung cấp các chức năng liên quan đến nhóm tương ứng. Thí dụ để truy xuất database thông qua giao tiếp ODBC, ta sẽ dùng các hàm trong nhóm ODBC, các hàm trong nhóm này đều có tên bắt đầu bởi tiếp đầu ngữ "obdc_" : odbc_autocommit, odbc_binmode, odbc_close, odbc_close_all, odbc_commit, odbc_connect, odbc_cursor, odbc_do, odbc_error, odbc_errormsg, odbc_exec, odbc_execute, odbc_fetch_into, odbc_fetch_row, odbc_field_name, odbc_field_num, odbc_field_type, odbc_field_len, odbc_field_precision, odbc_field_scale, odbc_free_result, odbc_longreadlen, odbc_num_fields, odbc_pconnect, odbc_prepare, odbc_num_rows, odbc_result, odbc_result_all, odbc_rollback, odbc_setoption, odbc_tables, odbc_tableprivileges, odbc_columns, odbc_columnprivileges, odbc_gettypeinfo, odbc_primarykeys, odbc_foreignkey, odbc_procedures, odbc_procedurecolumns, odbc_specialcolumns, odbc_statistics



9.3 Thí dụ xây dựng 1 website bằng PHP

Yêu cầu : tạo 1 website dùng công nghệ PHP cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

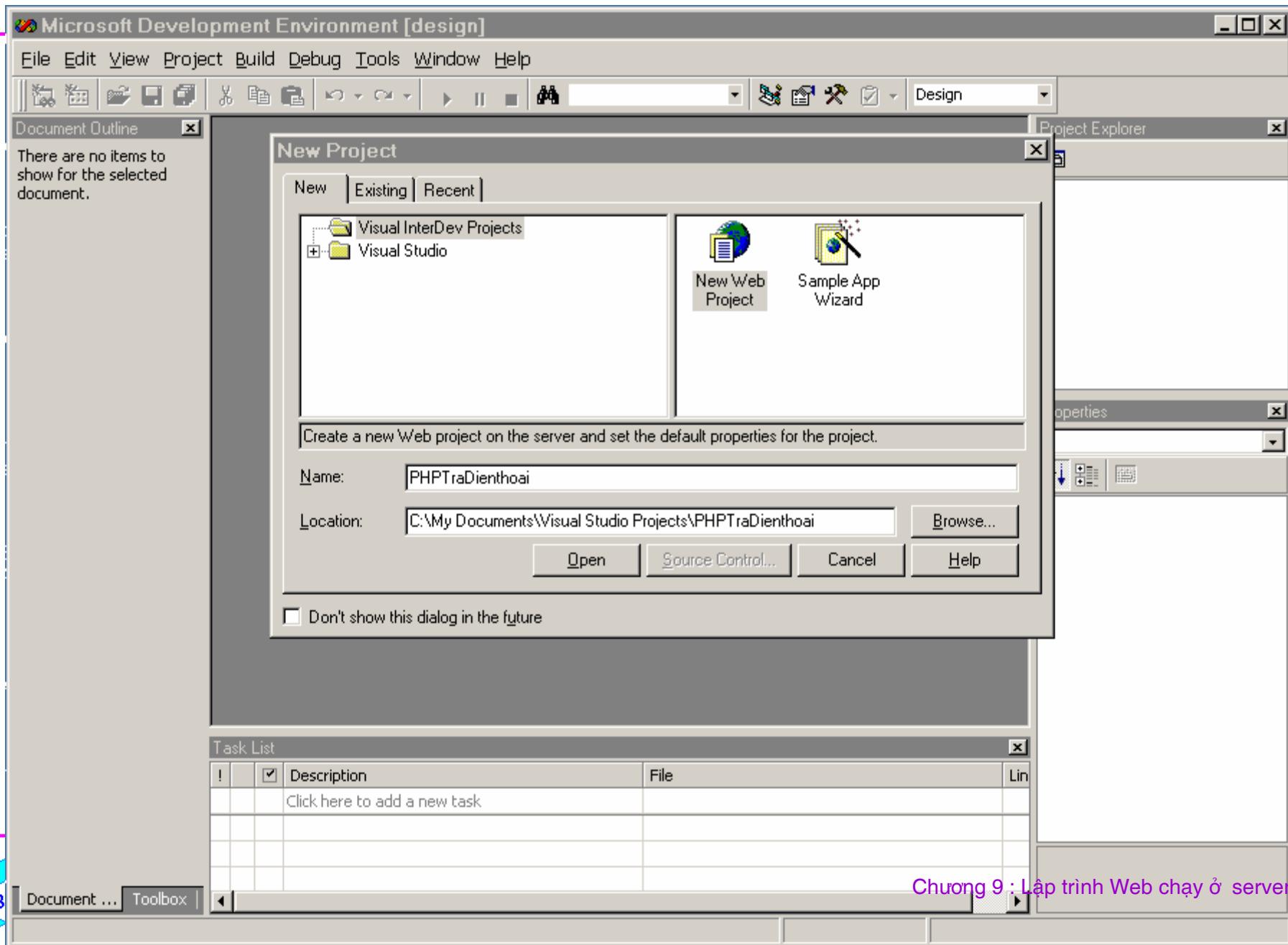
Phân tích : cần 2 module chức năng :

- module chứa form nhập thông tin về các account cần tra. module này có thể là 1 trang HTML thuần túy.
- và module truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm, module này là 1 trang PHP theo yêu cầu.

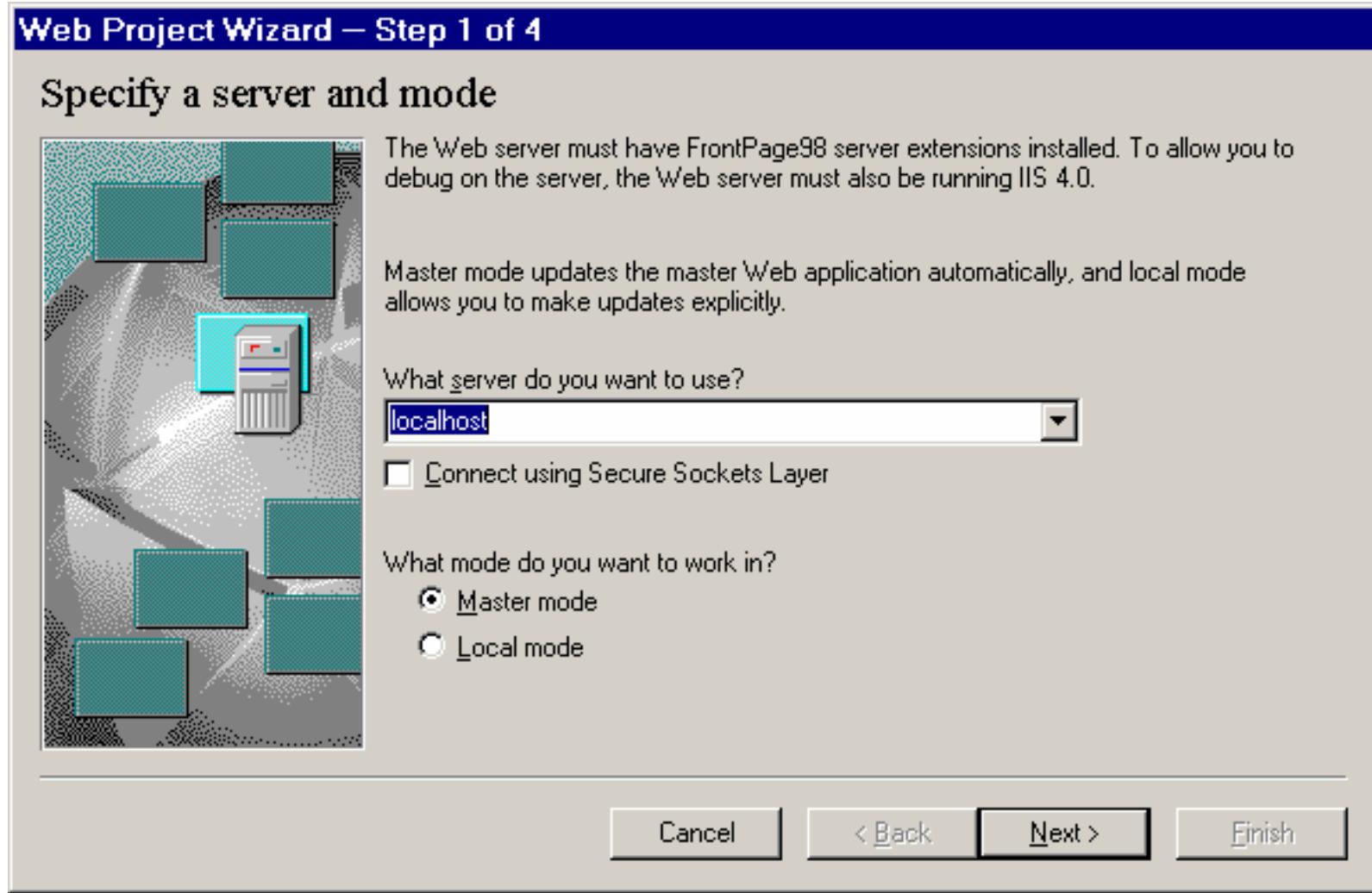
Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình Web của Microsoft tên là Visual InterDev. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



1. Chạy InterDev, chọn Location và nhập tên của Project quản lý, rồi Open



2. Nhập tên Web server, trả lời các cửa sổ Wizard



3. Tạo trang HTML tên PHPInForm.htm, tạo form nhập liệu, tạo các phần tử trong form

The screenshot shows the Microsoft Development Environment (IDE) interface for creating an HTML form. The title bar reads "Miscellaneous Files - Microsoft Development Environment [design] - [DBDTHOAI.HTM*]". The menu bar includes File, Edit, View, Project, Debug, HTML, Table, Format, Tools, Window, and Help. The toolbar contains various icons for file operations and design tools. The main workspace displays an HTML page titled "Demo việc tra danh b\u00e1nh \u00e1ienn tho\u00e1i". The page content includes:

```
<BODY>
<FORM>
    <input type="text" name="tentbao"/> T\u00e1n thu\u00e1 bao : 
    <input type="text" name="sodthoai"/> S\u00e1 \u00e1ienn tho\u00e1i : 
    <input type="text" name="diachi"/> )\u00e1a ch\u00e1 : 
    <input type="button" value="Bat dau tra" name="submit1"/>
</FORM>
```

The "HTML Outline" panel on the left shows the structure of the form elements. The "Project Explorer" panel on the right shows the solution and the current file "DBDTHOAI.HTM". The "Properties" panel on the right lists CSS properties for the selected element, with ".fontFamily" highlighted. The bottom navigation bar has tabs for Design, Source, and Quick View, with Design selected. A Task List window is open at the bottom.

4. Nội dung tag Form nhập liệu trong trang Web

```
<FORM action="Tradienthoai.php" method=post>
```

Tên thuê bao :

```
<INPUT name=txtTenthuebao style="width: 269px; height: 22px">
<P>Số điện thoại : <INPUT name=txtSodienthoai></P>
<P>Địa chỉ : <INPUT name=txtDiachi style="width: 373px; height: 22px"></P>
<P><INPUT type=submit value="Bắt đầu tra"></P>
</FORM>
```

Lưu ý : 3 textbox trong form có tên lần lượt là txtTenthuebao, txtSodienthoai và txtDiachi. Trang PHP xử lý form có tên là Tradienthoai.php.



5. tạo trang PHP tên là Tradienthoai.php, nhập code PHP tra danh bạ điện thoại.

The screenshot shows the Microsoft Development Environment (IDE) interface. The main window displays the following PHP code:

```
<html><head><title></title></head>
<body>
<h1>Kết quả tra danh bạ theo yêu cầu của bạn :</h1>
<table border=1 cellPading=1 cellSpacing=1>
<tr><th>Số thuê bao</th><th>Tên thuê bao</th><th>Số điện thoại</th><th>Địa chỉ</th></tr>
<?
# connect to a DSN "Danhbadienthoai" with a user and password ""
$connect = odbc_connect("Danhbadienthoai", "", "");
# query the users table for name and surname
# $query = "SELECT * FROM dbadthoai";
$fsqlstart = 1;
$newSQL="SELECT * FROM dbadthoai";
if ($tentbao != "") {
    $newSQL = "$newSQL where tentbao like '%$tentbao%'";
    $fsqlstart = 0;
}
if ($sodthoai != "") {
    if ($fsqlstart == 1)
        $newSQL = "$newSQL where sodthoai like '%$sodthoai%'";
    else
        $newSQL = "$newSQL and sodthoai like '%$sodthoai%'";
    $fsqlstart = 0;
}
if ($diachi != "") {
    if ($fsqlstart == 1)
        $newSQL = "$newSQL where diachi like '%$diachi%'";
    else
        $newSQL = "$newSQL and diachi like '%$diachi%'";
    $fsqlstart = 0;
}
$query = $newSQL;
```

The Project Explorer on the right shows a solution named 'Solution1' containing a 'Miscellaneous Files' folder with two files: 'DBDTHOAI.HTM' and 'Tradienthoai.php'. The Properties window is also visible.

At the bottom, the Task List window shows a single entry: 'Click here to add a new task'.

On the left, the Document Outline window indicates there are no items to show for the selected document.

The status bar at the bottom right shows: 'Chuong 9 : Lập trình Web chạy ở server using PHP' and 'Slide 973'.

6. Source code cụ thể của trang PHP Tradienthoai.php

```
<html><head><title></title></head>
<body>
<h3>Ket qua tra danh ba theo yeu cau cua ban :</h3>
<?php
# xac dinh cac du lieu nhap tu form
$ttbao = $_POST["txtTenthuebao"];
$sodthoai = $_POST["txtSodienthoai"];
$diachi = $_POST["txtDiachi"];
# connect to a DSN "Danhbadienthoai" with a user and password ""
$connect = odbc_connect("MyDatabase", "", "");
# xay dung lenh SQL tim kiem
$fsqlstart = 1;
$newSQL="SELECT * FROM danhbadienthoai";
```



6. Source code cù thê của trang PHP Tradienthoai.php (tt)

```
if ($ttbao != "") {  
    $newSQL = "$newSQL where tenthuebao like '%$ttbao%'";  
    $fsqlstart = 0;  
}  
if ($sodthoai != "") {  
    if ($fsqlstart == 1)  
        $newSQL = "$newSQL and sodienthoai like '%$sodthoai%'";  
    else  
        $newSQL = "$newSQL and sodienthoai like '%$sodthoai%'";  
    $fsqlstart = 0;  
}  
if ($diachi != "") {  
    if ($fsqlstart == 1)  
        $newSQL = "$newSQL and diachi like '%$diachi%'";  
    else  
        $newSQL = "$newSQL and diachi like '%$diachi%'";  
    $fsqlstart = 0;  
}
```



6. Source code cù thê của trang PHP Tradienthoai.php (tt)

```
# perform the query
$result = odbc_exec($connect, $newSQL);
# fetch the data from the database
$i = 1;
if (odbc_fetch_row($result)) {
    # tao table va xuat hang dau tien
    printf("<table border=1 cellPading=1 cellSpacing=1>");
    printf("<tr><th>So thu tu</th><th>Ten thue bao</th><th>So dien
thoai</th><th>Dia chi</th></tr>");
    $tentbao = odbc_result($result, 2);
    $sodthoai = odbc_result($result, 3);
    $diachi = odbc_result($result, 4);
    print("<tr><td> $i </td><td> $tentbao </td><td> $sodthoai </td><td> $diachi
</td><tr>\n");
    $i = $i+1;
}
```



6. Source code cụ thể của trang PHP TradienThoai.php (tt)

```
while(odbc_fetch_row($result)) {
    $tentbao = odbc_result($result, 2);
    $sodthoai = odbc_result($result, 3);
    $diachi = odbc_result($result, 4);
    print("<tr><td> $i </td><td> $tentbao </td><td> $sodthoai </td><td> $diachi
</td><tr>\n");
    $i = $i+1;
}
printf("</table><p></p>\n");
} else printf("<p>Khong tim duoc account nao thoai tieu chuan cua ban ca!!<br>");
# close the connection
odbc_close($connect);
?>
<p>
<a href="PHPInForm.htm">Quay ve trang chu</a>
</body>
</html>
```



Môn học : Lập trình mạng

Chương 10

LẬP TRÌNH WEB CHẠY Ở SERVER BẰNG JAVA

- 10.1 Servlet Java
- 10.2 JSP
- 10.3 JSP & JavaBean



10.1 Tổng quát về Servlet Java

- Servlet là đối tượng Java (giống như Applet), nhưng không có giao diện người dùng, nó chạy trên máy server, giao tiếp với client của nó thông qua việc gửi nhận request/response.
- Viết 1 servlet cũng giống như 1 class Java bình thường khác, nhưng Servlet phải là class con của class HttpServlet và thường phải override 1 hay cả 2 method doGet() và doPost(). Template của 1 servlet :
public class MyServlet extends HttpServlet {

```
...
// tác vụ xử lý HTTP Get request
public void doGet(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException { ... }
// tác vụ xử lý HTTP Post request
public void doPost(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException { ... }
...
}
```



Nhập/xuất thông tin của Servlet Java

- tham số request của 2 tác vụ doGet() & doPost() cho phép Servlet truy xuất thông tin được gửi về từ URL yêu cầu hay từ Form nhập liệu. Cách thức truy xuất tham số như sau :

```
String tentbao = request.getParameter("txtTenthuebao");
```

- tham số response của 2 tác vụ doGet() & doPost() cho phép Servlet xuất thông tin gửi về ứng dụng client. Cách thức xuất kết quả như sau :

'tìm đối tượng xuất

```
PrintWriter out = response.getWriter();
```

'xuất kết quả

```
out.println("chuỗi kết quả");
```

...



Thí dụ xây dựng 1 website bằng Servlet

Yêu cầu : tạo 1 website dùng công nghệ Servlet Java cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

Phân tích : cần 2 module chức năng :

- module chứa form nhập thông tin về các account cần tra. module này có thể là 1 trang HTML thuần túy.
- và module truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm, module này là 1 Servlet theo yêu cầu.

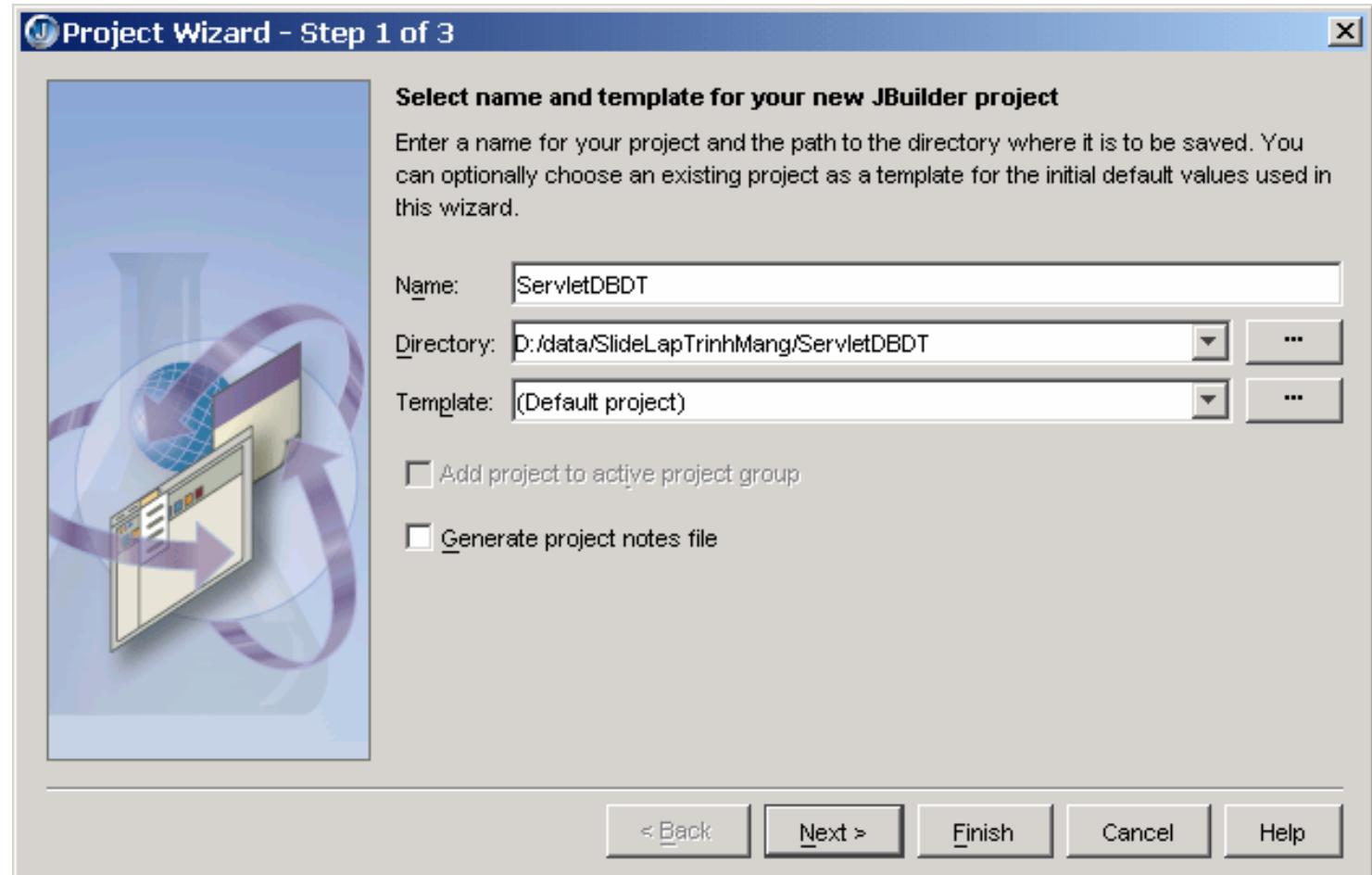
Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình JBuilder 2005. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



Tạo website tra điện thoại bằng Servlet trên JBuilder

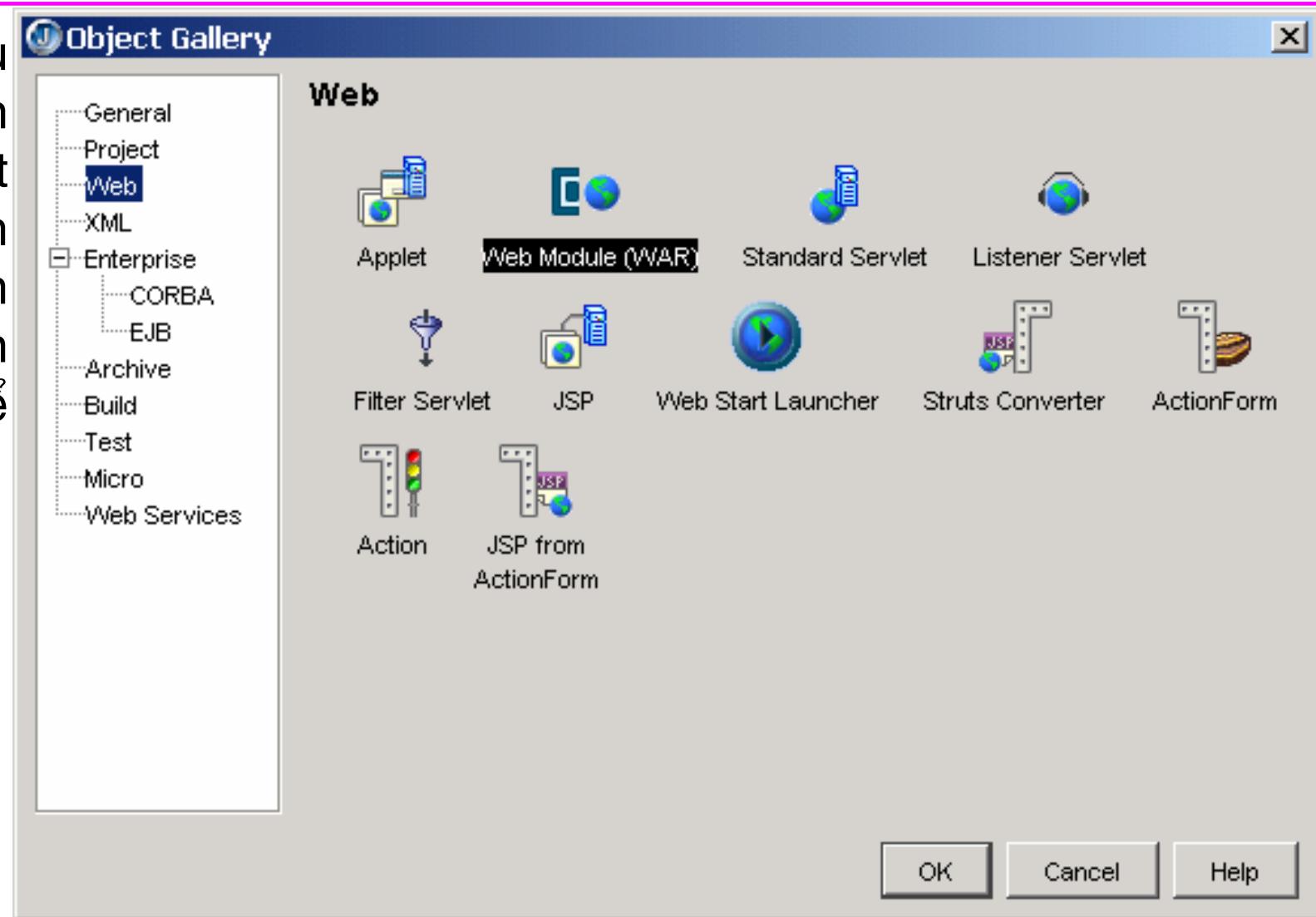
- Chạy JBuilder 2005, chọn menu File.New Project rồi trả lời các cửa sổ Wizard để tạo 1 project mới quản lý Website.

Lưu ý trong cửa sổ Step 1, chọn thư mục và nhập tên Project quản lý Website là ServletDBDT.



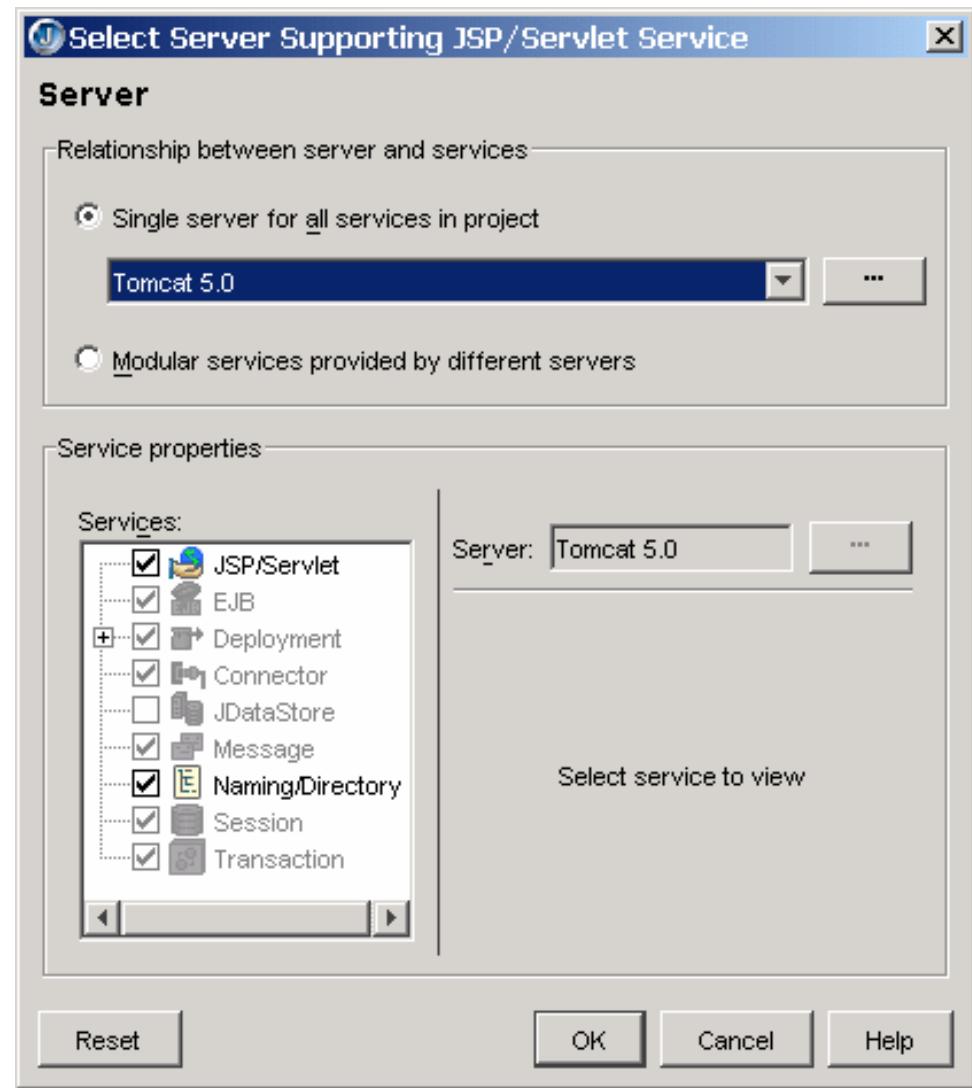
Tạo website tra điện thoại bằng Servlet trên JBuilder

2. chọn menu File.New để hiển thị cửa sổ Object Gallery như hình bên. chọn nhóm Web, chọn icon Web Module (WAR) để tạo Website.



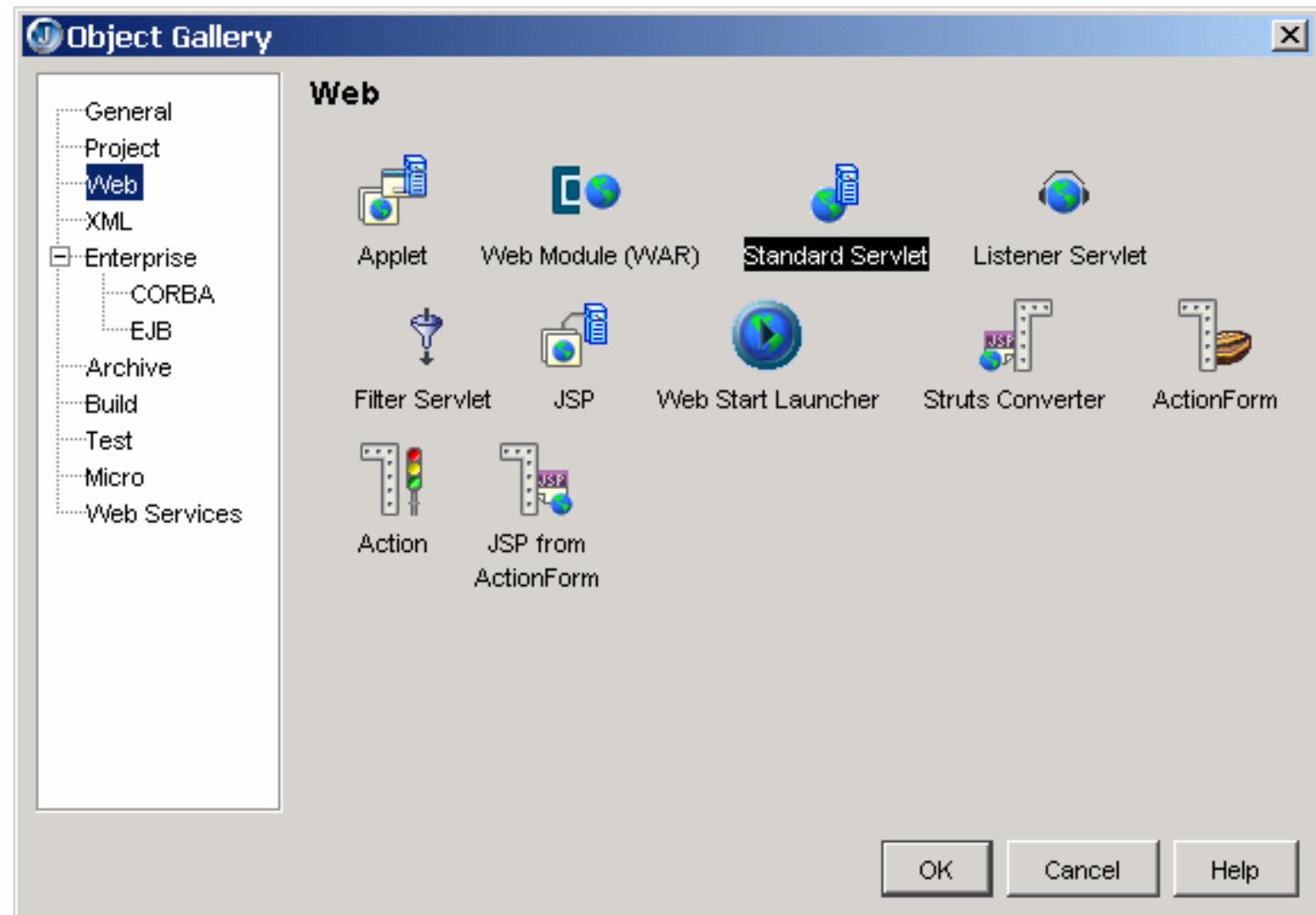
Tạo website tra điện thoại bằng Servlet trên JBuilder

2.1 Khi cửa sổ "Select Server..." hiển thị như hình bên, bạn ấn dấu chọn vào checkbox "Single server...", chọn server "Tomcat 5.0" rồi ấn button OK.



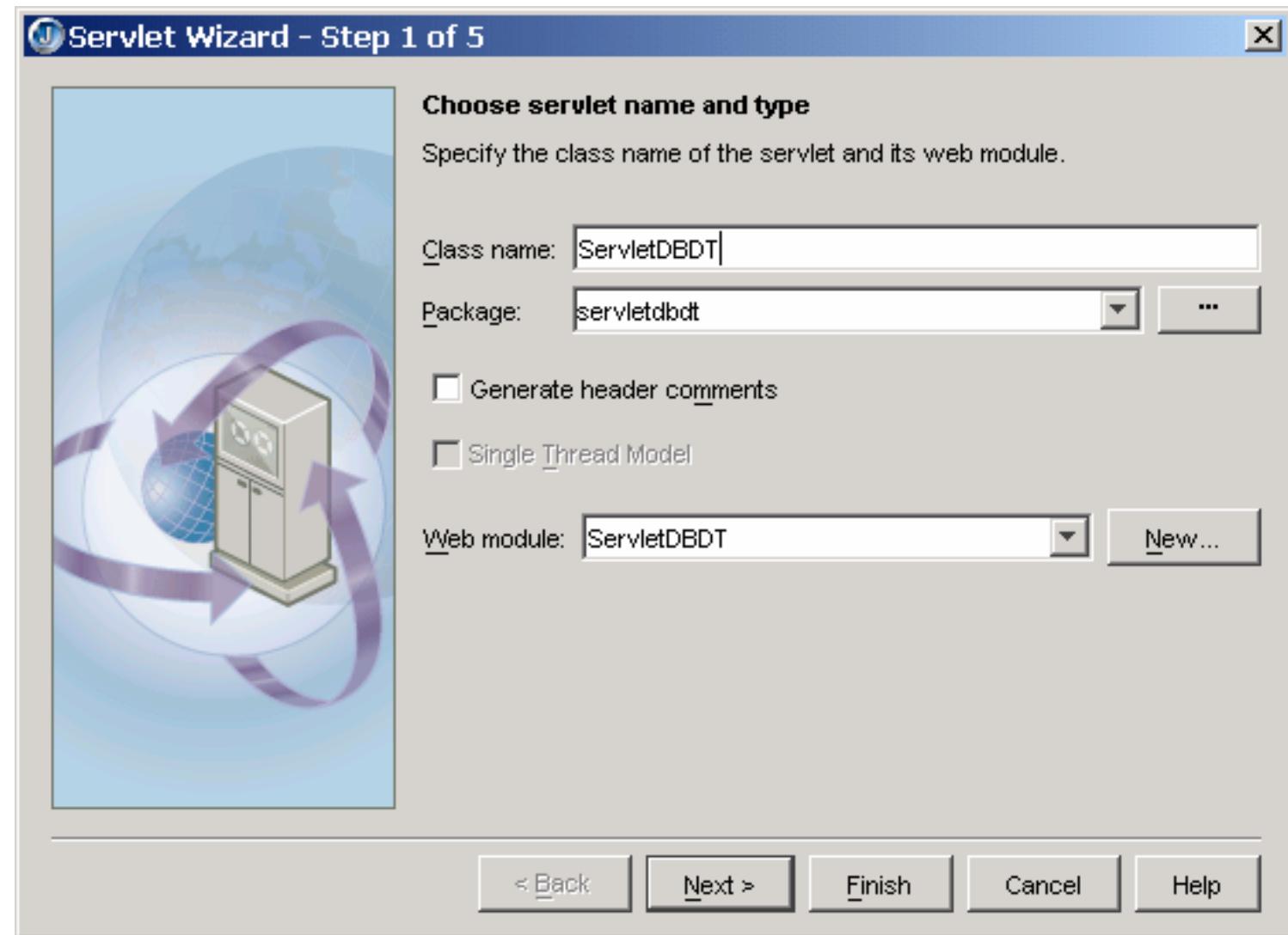
Tạo website tra điện thoại bằng Servlet trên JBuilder

3. chọn menu File.New để hiển thị cửa sổ Object Gallery như hình bên. Chọn nhóm Web, chọn icon Standard Servlet để tạo Servlet tra account danh bạ điện thoại, rồi ấn button OK.



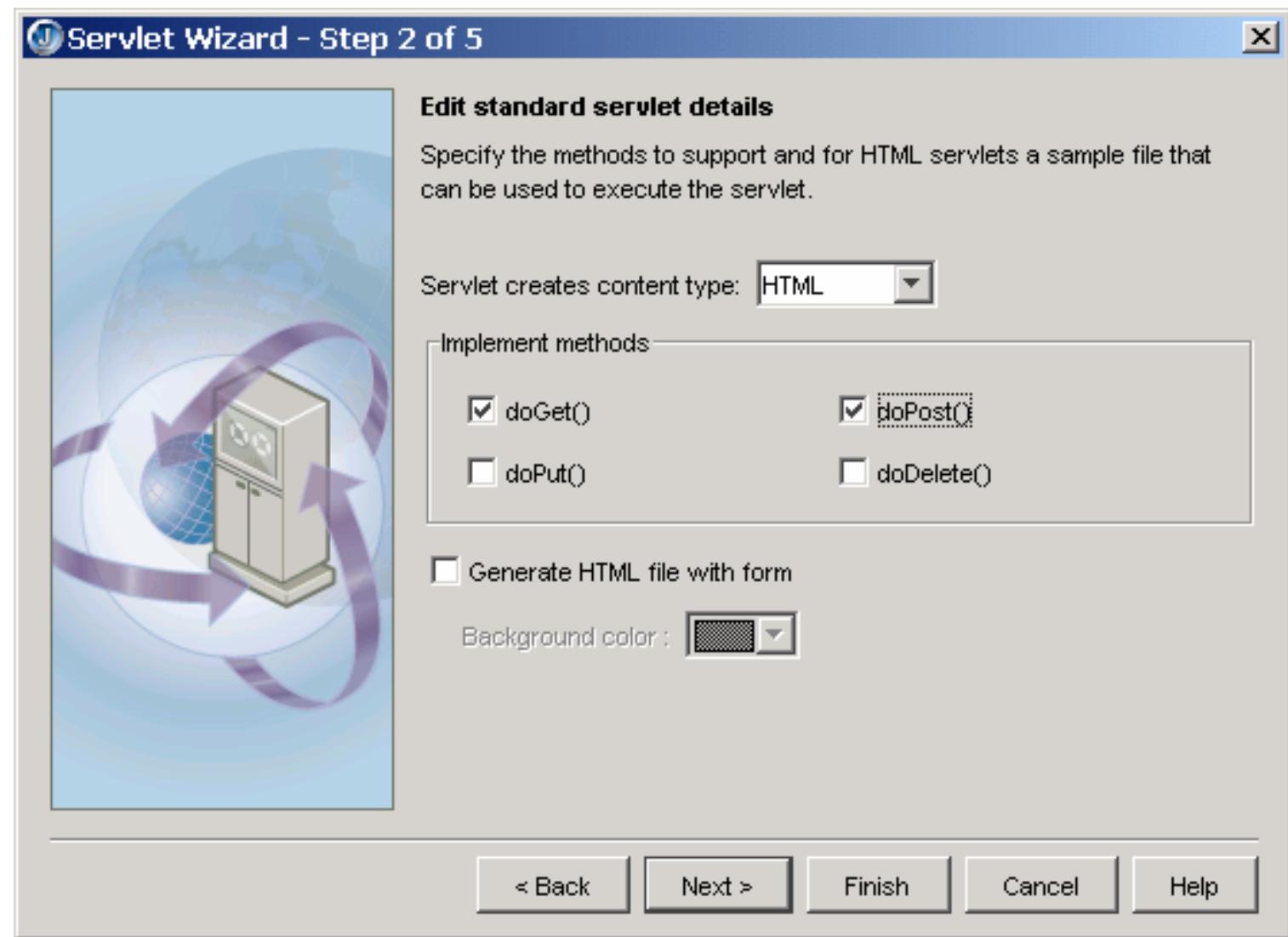
Tạo website tra điện thoại bằng Servlet trên JBuilder

3.1 trong cửa sổ Step 1 của qui trình Wizard tạo Servlet, bạn nhập tên servlet là ServletDBDT, rồi ấn button Next để tiếp tục Step 2.



Tạo website tra điện thoại bằng Servlet trên JBuilder

3.2 trong cửa sổ Step 2 của qui trình Wizard tạo Servlet, bạn ấn dấu chọn vào 2 checkbox doGet() và doPost() để tạo 2 tác vụ xử lý tương ứng, rồi ấn button Finish để hoàn tất việc tạo Servlet.



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
//code đầy đủ của servlet
package servlettradbdt;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
import java.util.*;
import java.sql.*;
public class ServletTraDBDT extends HttpServlet {
    public ServletTraDBDT() {
        try { jblInit(); } catch (Exception ex) { ex.printStackTrace(); }
    }

    //định nghĩa các thuộc tính cần dùng cho Servlet
    private static final String CONTENT_TYPE = "text/html; charset=UTF-8";
    private Connection con;
```



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
//tác vụ khởi động các thuộc tính  
public void init() throws ServletException {  
}  
//tác vụ xử lý yêu cầu HTTP Post  
public void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
    //gọi tác vụ doGet() trực hiện  
    doGet(request,response);  
}  
//tác vụ dọn dẹp các thuộc tính trước khi servlet bị xóa  
public void destroy() {  
}  
//tác vụ khởi động được gọi từ init()  
private void jbInit() throws Exception {  
}
```



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
//tác vụ xử lý yêu cầu HTTP Get
public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    response.setContentType(CONTENT_TYPE);
    PrintWriter out = response.getWriter();
    //tạo các tag HTML ở đầu trang web kết quả
    out.println("<html>");
    out.println("<head><title>SearchAccount</title></head>");
    out.println("<body>");
    out.println("<h2>Kết quả tra điện thoại theo yêu cầu của bạn :</h2>");
    //tìm nội dung các tham số được truyền qua Servlet
    String tentbao = request.getParameter("txtTenthuebao");
    String sodthoai = request.getParameter("txtSodienthoai");
    String diachi = request.getParameter("txtDiachi");
    // xây dựng lệnh SQL tìm kiếm
    boolean fsqlstart = true;
    String newSQL = "SELECT * FROM danhbadienthoai";
```



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
if (tentbao != null) {  
    if (tentbao.length() != 0) {  
        newSQL = newSQL + " where tenthuebao like '%" + tentbao + "%'";  
        fsqlstart = false;  
    }  
}  
if (sodthoai != null) {  
    if (sodthoai.length() != 0) {  
        if (fsqlstart) {  
            newSQL = newSQL + " where sodienthoai like '%" + sodthoai + "%'";  
        }  
        else {  
            newSQL = newSQL + " and sodienthoai like '%" + sodthoai + "%'";  
        }  
        fsqlstart = false;  
    }  
}
```



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
if (diachi != null) {  
    if (diachi.length() != 0) {  
        if (fsqlstart) {  
            newSQL = newSQL + " where diachi like '%" + diachi + "%';  
        } else {  
            newSQL = newSQL + " and diachi like '%" + diachi + "%';  
        }  
        fsqlstart = false;  
    }  
}  
//xây dựng chuỗi ConnectionString truy xuất database dùng DSN  
String conStr = "jdbc:odbc:MyDatabase";  
//định nghĩa các biến cần dùng  
Statement stmt = null;  
ResultSet rs = null;
```



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
try { //kết nối database, thực thi lệnh SQL, kết quả chứa trong rs
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    con = DriverManager.getConnection(conStr, "", "");
    stmt = con.createStatement();
    rs = stmt.executeQuery(newSQL);
    int i = 1;
    if (rs != null && rs.next()) {
        //nếu có ít nhất 1 account, tạo bảng dữ liệu, hàng tiêu đề và hàng dữ
        //liệu đầu tiên
        out.println("<table border=1 cellPadding=1 cellSpacing=1>");
        out.println("<tr><th>Số  thự  tự</th><th>Tên  thuê  bao</th><th>Số
        điện thoại</th><th>Địa chỉ</th></tr>\n");
        out.println("<tr><td>" + i++ + "</td><td>" + rs.getString(2) +
                    "</td><td>" + rs.getString(3) + "</td><td>" +
                    rs.getString("diachi") + "</td></tr>\n");
    }
}
```



Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
while (rs.next()) { //lập xuất từng hàng kết quả còn lại
    out.println("<tr><td>" + i++ + "</td><td>" + rs.getString(2) +
                "</td><td>" + rs.getString(3) + "</td><td>" +
                rs.getString("diachi") + "</td></tr>\n");
}
out.println("</table>");
} else out.println("<p>Khong tim duoc account nao thoac yeu cau cua
ban<br>");
}
catch (Exception e) {
    System.out.println("Error : " + e);
}
```



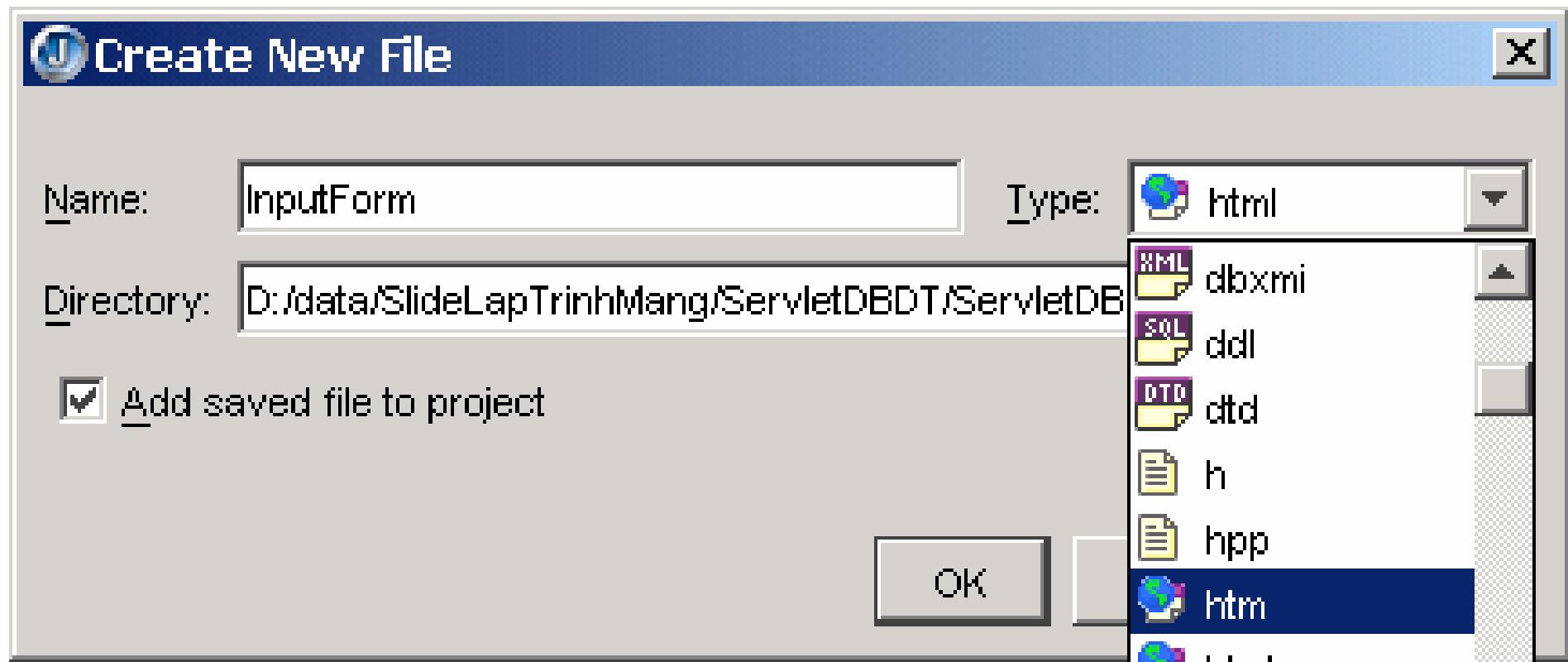
Viết code cho 2 tác vụ doGet() & doPost() của Servlet

```
//đóng các đối tượng đã dùng lại
finally {
    if (rs != null) {
        try { rs.close(); }
        catch (Exception _ex) { System.out.println("Exception: " + _ex); }
    }
    if (stmt != null) {
        try { stmt.close(); }
        catch (Exception _ex) { System.out.println("Exception: " + _ex); }
    }
}
//xuất tag liên kết về trang chủ chứa form nhập liệu
out.println("<a href='InputForm.html'>Quay về trang chủ</a>");
out.println("</body></html>");
}
```



Tạo trang chủ chứa form nhập liệu cần tra

4. chọn menu File.New File... để hiển thị cửa sổ "Create New File" như hình dưới đây. Chọn thư mục chứa file (lưu ý để server Tomcat quản lý đúng Website, bạn phải chọn thư mục có chứa thư mục con là WEB-INF), nhập tên file HTML là "InputForm", duyệt tìm và chọn Type file là html.



Viết code cho trang chủ chứa form

```
<HTML><HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=utf8">
<TITLE></TITLE></HEAD>
<BODY>
<font face=Helvetica size="4">
<H2 align=center>Trang Web demo việc tra danh bạ điện thoại</H2>
<FORM id=FORM1 name=FORM1 action=servletdbdt method=post>
<P>&nbsp;</P>
<P>Nhập thông tin về tên thuê bao : <INPUT style="width: 244px; height: 22px" size=34 name=txtTenthuebao></P>
<P>Nhập thông tin về số điện thoại : <INPUT name=txtSodienthoai></P>
<P>Nhập thông tin về địa chỉ : <INPUT style="width: 284px; height: 22px" size=40 name=txtDiachi></P>
<P><INPUT style="width: 144px; height: 24px" type=submit size=45 value="Bắt đầu tra" name=submit1></P></FORM>
</font>
</BODY></HTML>
```



Kết quả chạy Servlet ServletDBDT (chọn menu Run.Run Project)

The screenshot shows the JBuilder 2005 IDE interface with the following details:

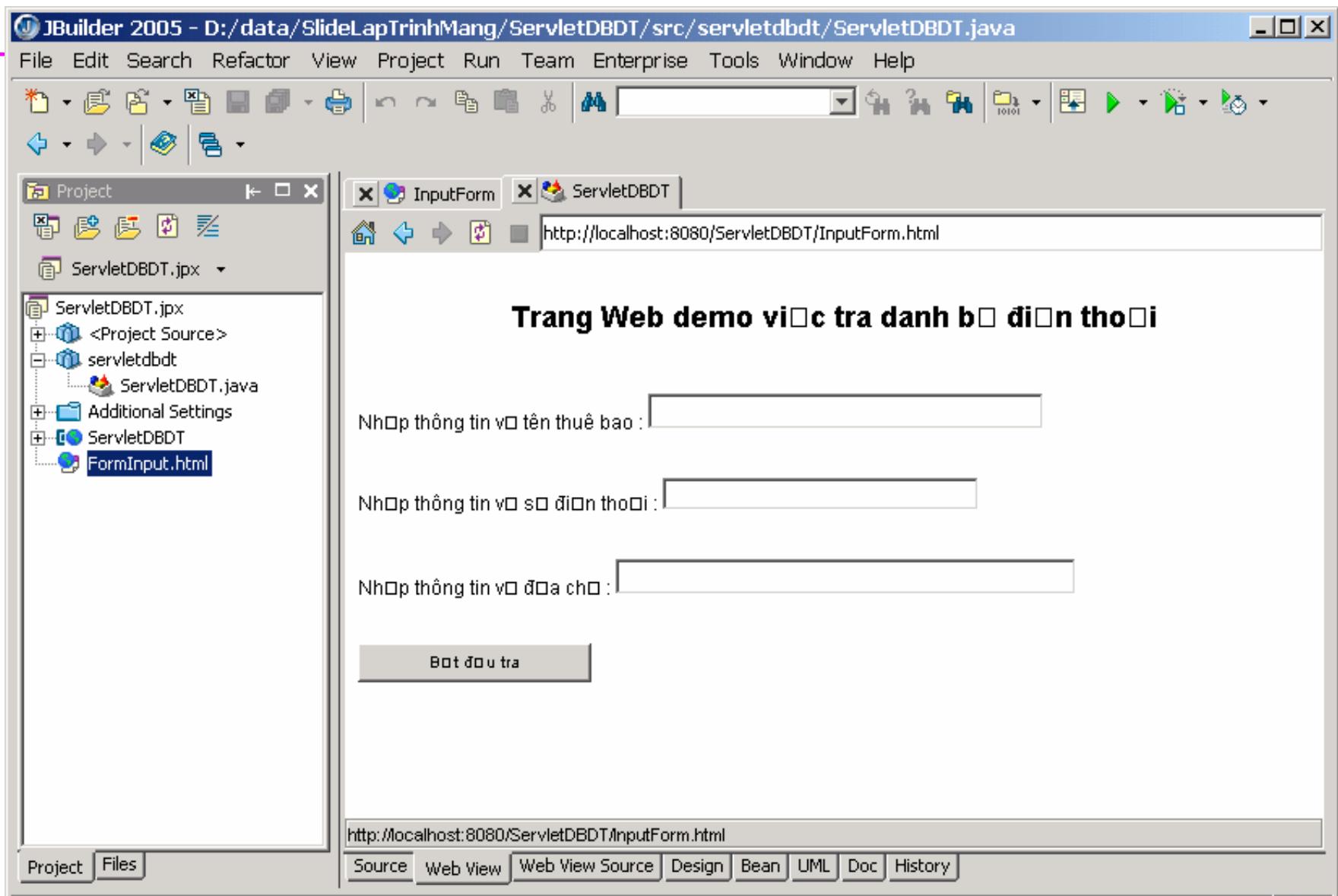
- Title Bar:** JBuilder 2005 - D:/data/SlideLapTrinhMang/ServletDBDT/src/servletdbdt/ServletDBDT.java
- Menu Bar:** File, Edit, Search, Refactor, View, Project, Run, Team, Enterprise, Tools, Window, Help
- Toolbar:** Includes icons for file operations, project management, and deployment.
- Project Explorer:** Shows the project structure with files like ServletDBDT.java, InputForm.html, and additional settings.
- Web Browser View:** Displays the output of the application at <http://localhost:8080/ServletDBDT/servletdbdt>. The page title is "Ket qua tra dien thoai theo yeu cau cua ban :". It contains a table with the following data:

So thu tu	Ten thue bao	So dien thoai	Địa chỉ
1	Nguyen Van Hiep	8644717	8A Lu Gia, P15, Q11, Tp.HCM
2	Huynh Thoai Loan	9070437	245/4 Ly Thuong Kiet, P15, Q11, Tp.HCM
3	Nguyen Luong Hoa	9024437	240 Tran Binh Trong, P4, Q5, Tp.HCM
4	Nguyen Lac Han	2030478	157 Hoang Dieu, P1, Q1, Tp.HCM

- Message View:** Shows logs from Tomcat 5.0. The log entries include:

```
INFO: Starting Coyote HTTP/1.1 on http-8080
Nov 28, 2006 10:17:36 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 1250 ms
NewSQL = SELECT * FROM danhbaidienthoai
NewSQL = SELECT * FROM danhbaidienthoai
```
- Bottom Status Bar:** Shows "Truong DTN Bach Khoa TP.HCM" and "Tomcat 5.0 http:8080".

Kết quả chạy file InputForm (chọn "Quay ve trang chu")



10.2 Tổng quát về JSP

- JSP là công nghệ script chạy ở máy server của SUN tương tự với ASP của Microsoft hay PHP.
- Hiện JSP có thể chạy trên nhiều Web server khác nhau từ Tomcat, Apache, Jrun, JavaWebServer,...
- Một trang JSP là trang HTML có tăng cường 1 số đoạn script viết bằng Java.
- Để server nhận biết được file nào là JSP, ta luôn dùng extension đã qui định theo cấu hình (mặc định là *.jsp).
- Đoạn script JSP được đóng trong tag <% %>.
- JSP thường dùng các đối tượng Java chính quy để xử lý các giải thuật phức tạp. Các đối tượng java này sẽ được đóng gói trong module JavaBean.



Cách viết trang JSP

Trước hết cần lưu ý rằng trang JSP sẽ được dịch ra thành 1 Servlet Java trước khi được thực thi.

Trang JSP là trang HTML có tăng cường 1 số đoạn mã Java, mỗi đoạn mã java thuộc 1 trong 3 loại sau :

1. Biểu thức có dạng `<%= expression %>`, khi chạy biểu thức được tính và kết quả sẽ được thay thế vào vị trí của mã nguồn.
2. Đoạn script có dạng `<% code %>`, sẽ được thêm vào tác vụ `_jspService` của Servlet kết quả, tác vụ này sẽ được gọi bởi tác vụ service (là tác vụ đóng vai trò như tác vụ `doGet` hay `doPut`).`_`
3. Khai báo có dạng `<%! code %>`, sẽ được thêm vào thân của Servlet nhưng nằm ngoài bất kỳ tác vụ nào.



Nhập/xuất thông tin của Servlet Java

- tham số request của 2 tác vụ doGet() & doPost() cho phép Servlet truy xuất thông tin được gửi về từ URL yêu cầu hay từ Form nhập liệu. Cách thức truy xuất tham số như sau :

```
String tentbao = request.getParameter("txtTenthuebao");
```

- tham số response của 2 tác vụ doGet() & doPost() cho phép Servlet xuất thông tin gửi về ứng dụng client. Cách thức xuất kết quả như sau :

'tìm đối tượng xuất

```
PrintWriter out = response.getWriter();
```

'xuất kết quả

```
out.println("chuỗi kết quả");
```

...



Thí dụ xây dựng 1 website bằng JSP

Yêu cầu : tạo 1 website dùng công nghệ JSP cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

Phân tích : cần 2 module chức năng :

- module chứa form nhập thông tin về các account cần tra, module này có thể là 1 trang HTML thuần túy.
- và module truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm, module này là 1 trang JSP theo yêu cầu.

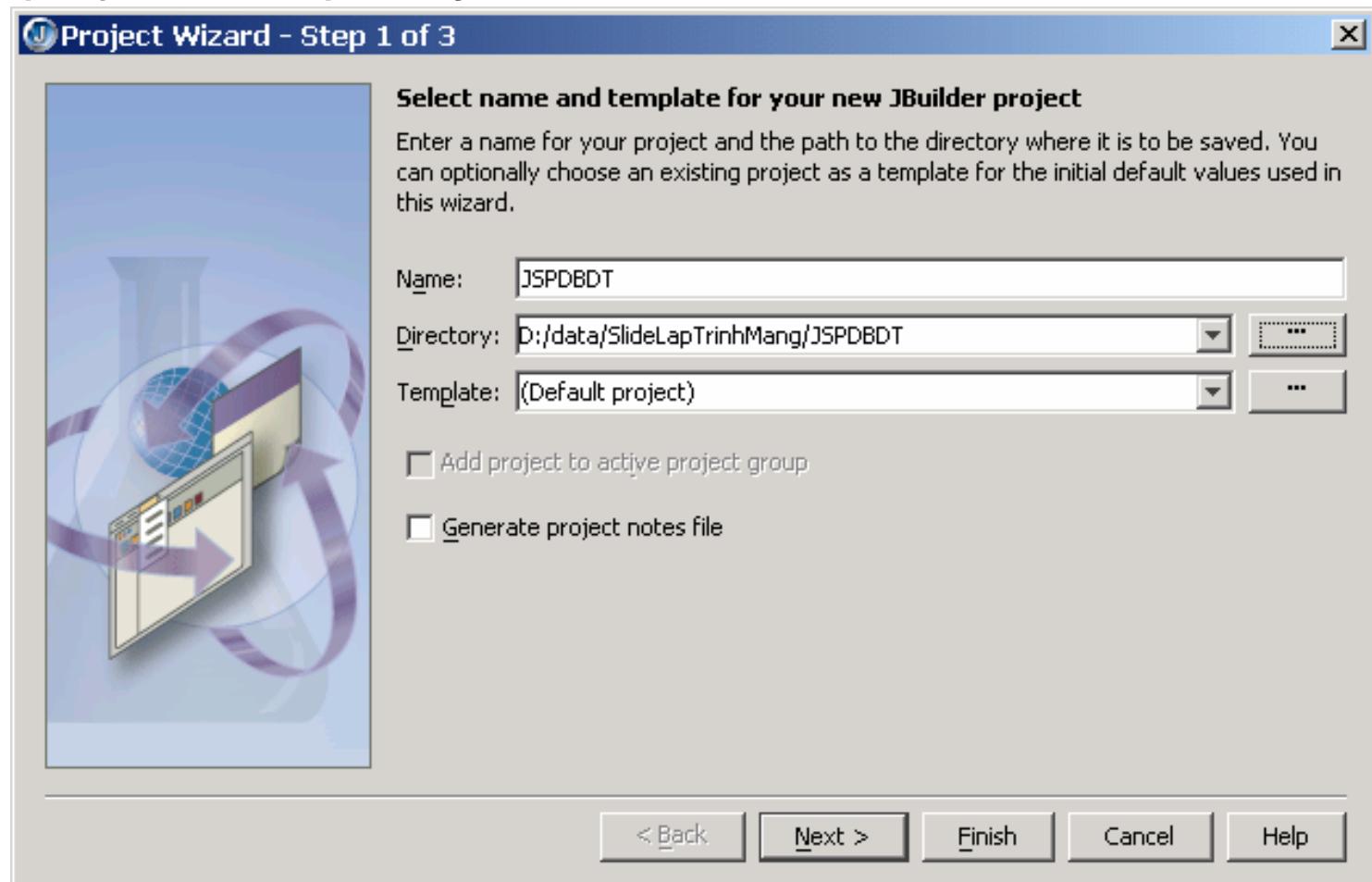
Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình JBuilder 2005. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



Tạo website tra điện thoại bằng JSP trên JBuilder

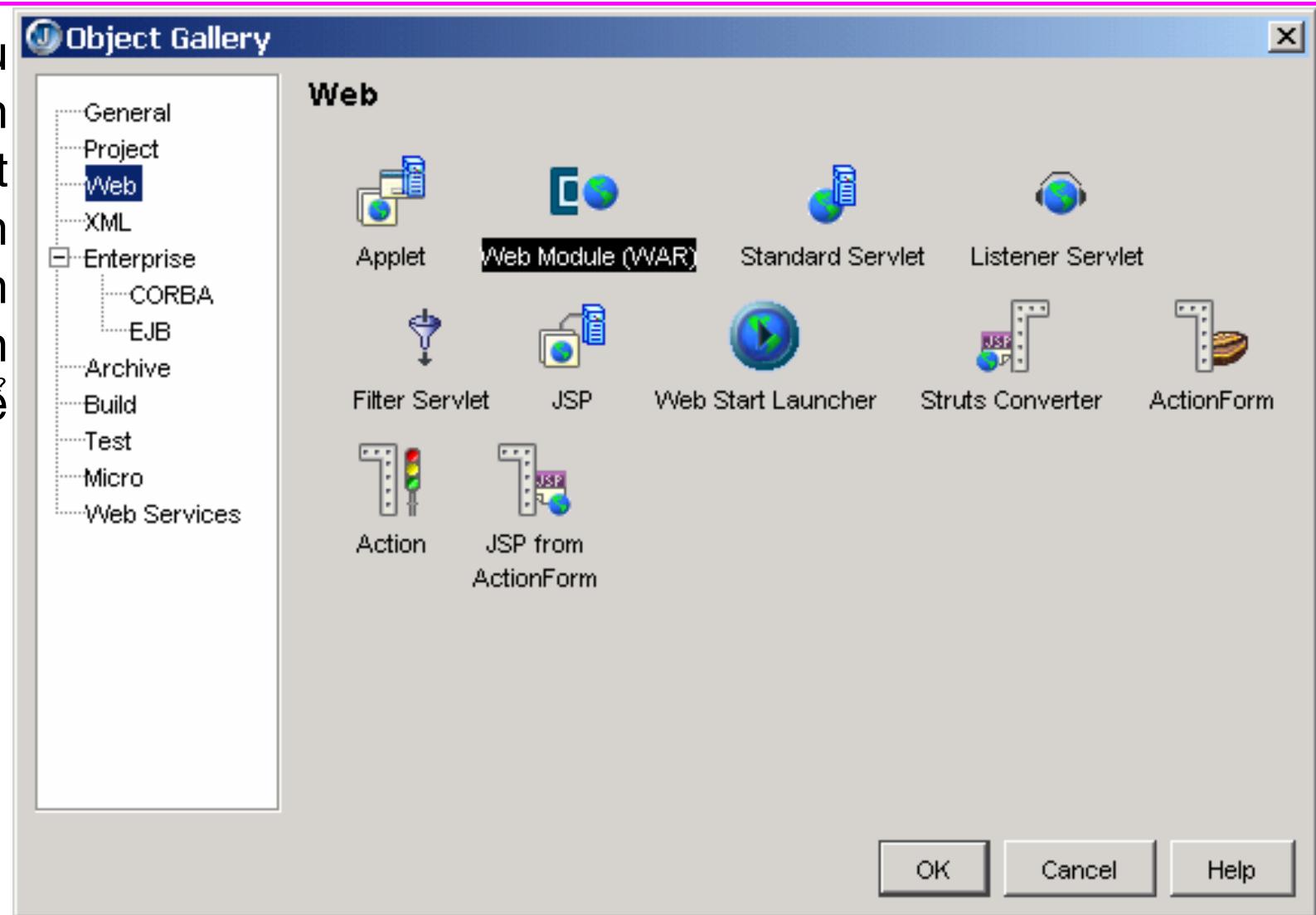
- Chạy JBuilder 2005, chọn menu File.New Project rồi trả lời các cửa sổ Wizard để tạo 1 project mới quản lý Website.

Lưu ý trong cửa sổ Step 1, chọn thư mục và nhập tên Project quản lý Website là JSPDBDT.



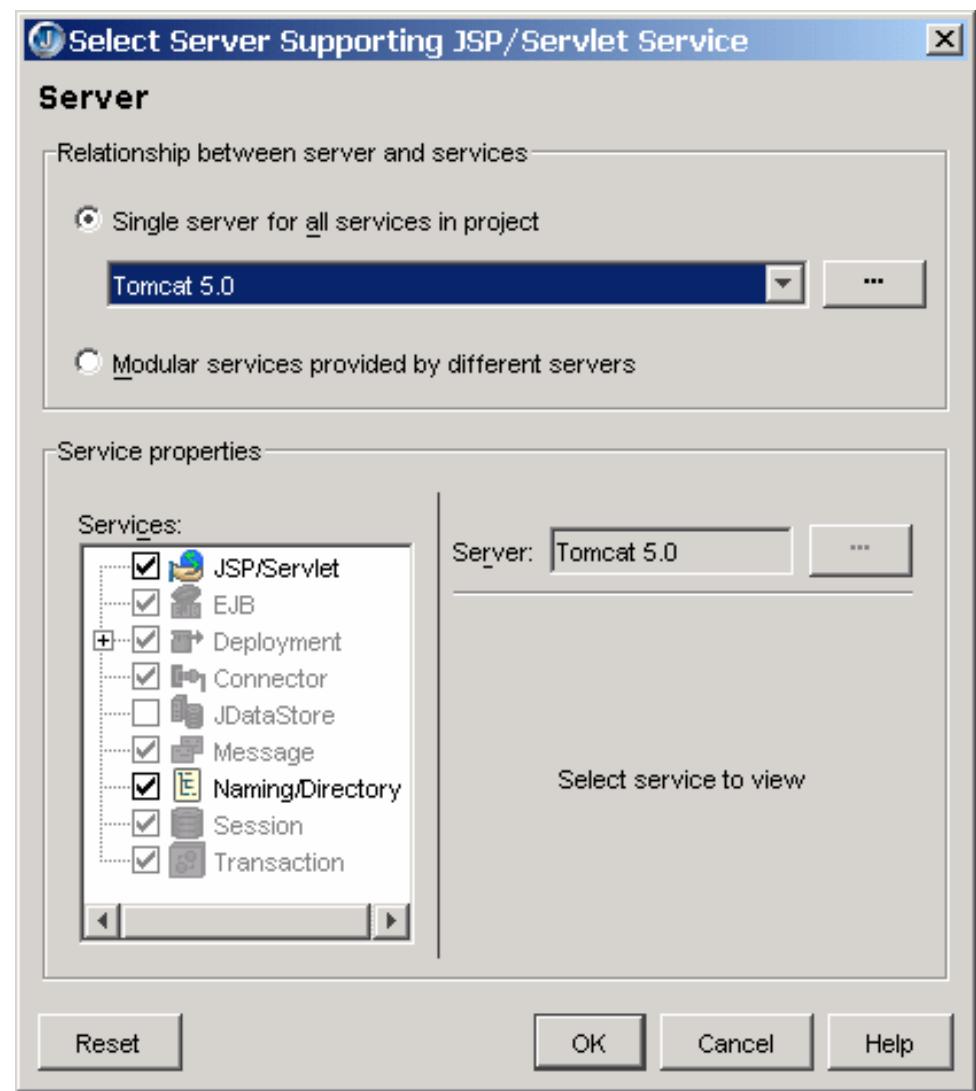
Tạo website tra điện thoại bằng JSP trên JBuilder

2. chọn menu File.New để hiển thị cửa sổ Object Gallery như hình bên. chọn nhóm Web, chọn icon Web Module (WAR) để tạo Website.



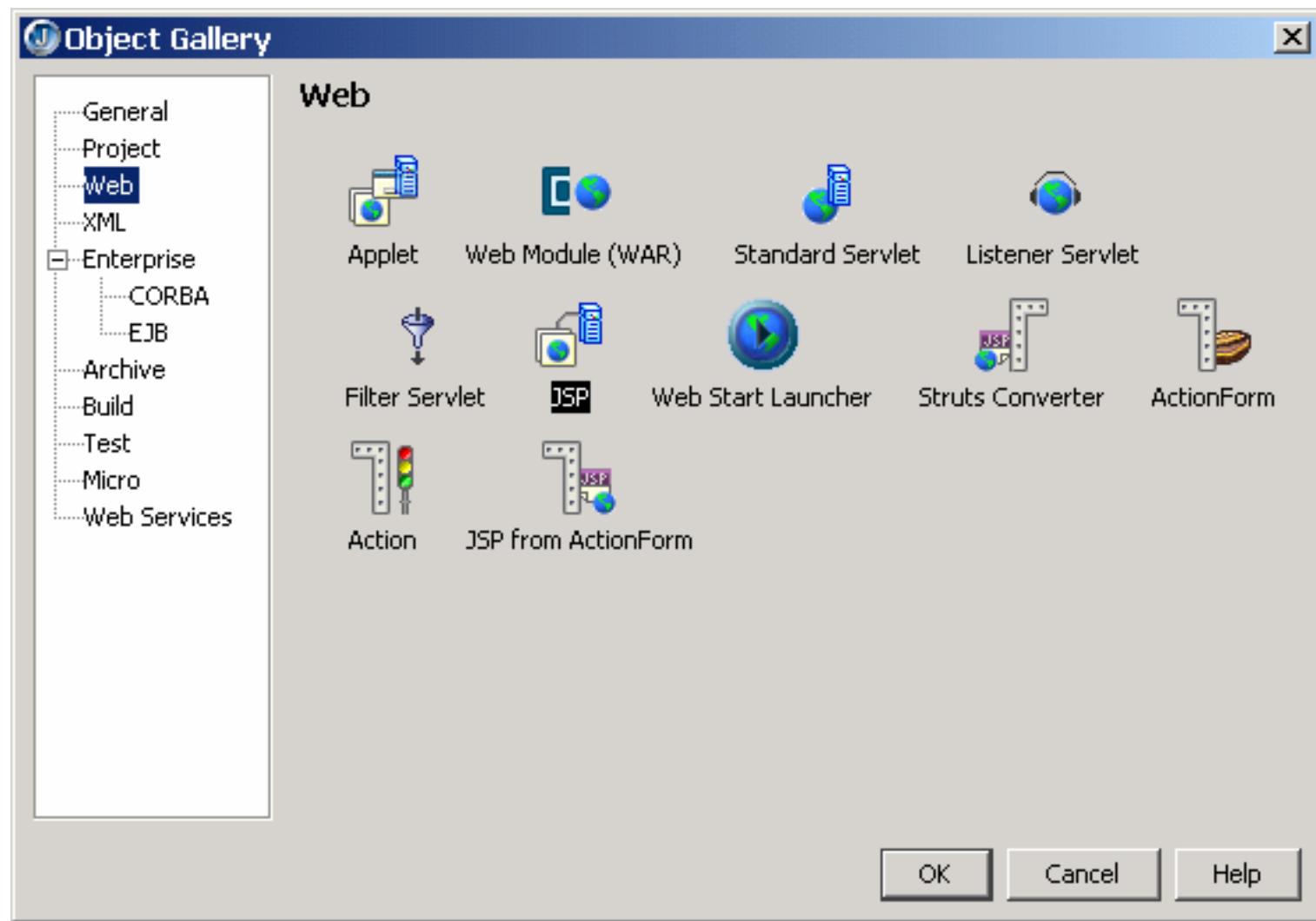
Tạo website tra điện thoại bằng JSP trên JBuilder

2.1 Khi cửa sổ "Select Server..." hiển thị như hình bên, bạn ấn dấu chọn vào checkbox "Single server...", chọn server "Tomcat 5.0" rồi ấn button OK.



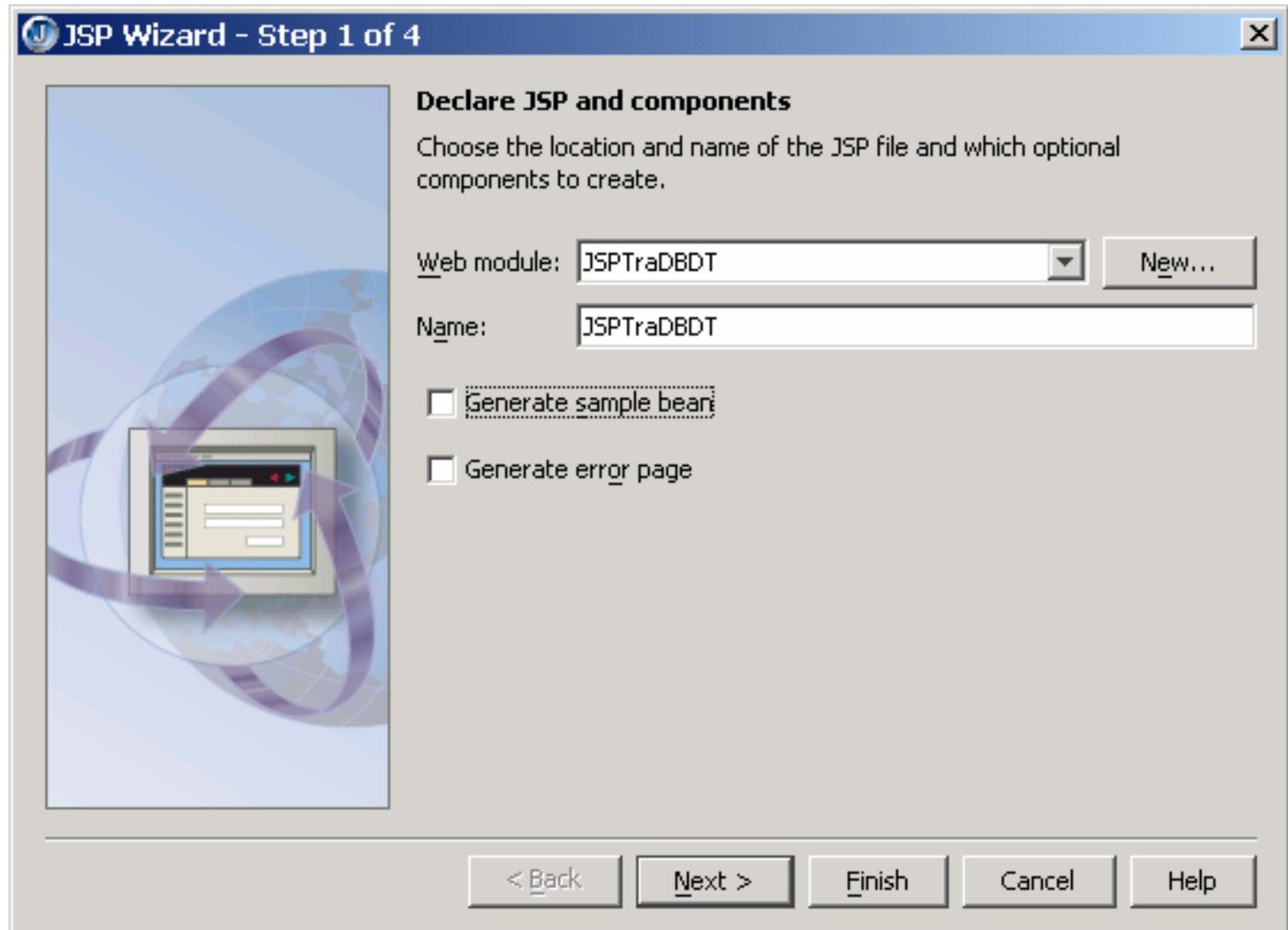
Tạo website tra điện thoại bằng JSP trên JBuilder

3. chọn menu File.New để hiển thị cửa sổ Object Gallery như hình bên. Chọn nhóm Web, chọn icon JSP để tạo 1 trang JSP tra account danh bạ điện thoại, rồi ấn button OK.



Tạo website tra điện thoại bằng JSP trên JBuilder

3.1 trong cửa sổ Step 1 của qui trình JSP Wizard, bạn nhập tên cho trang JSP là JSPTraDBDT, rồi ấn button Next để tiếp tục Step 2 (hay button Finish để chấp nhận các thông số mặc định cho các bước còn lại).



Viết code cho trang JSP như sau

```
<%@ page contentType="text/html; charset=UTF-8" %>
<html>
<head>
<%@ page import="java.sql.*" %>
<%!
public String SearchAccount (String tentbao, String sodthoai, String diachi) {
    StringBuffer table = new StringBuffer();
    Connection con;
    // xây dựng lệnh SQL tìm kiếm theo yêu cầu người dùng
    boolean fsqlstart = true;
    String newSQL = "SELECT * FROM danhbadienthoai";
    if (tentbao != null)
        if (tentbao.length() != 0) {
            newSQL = newSQL + " where tenthuebao like '%" + tentbao + "%'";
            fsqlstart = false;
        }
}
```



Code của trang JSP (tt)

```
if (sodthoai != null)
    if (sodthoai.length() != 0) {
        if (fsqlstart) newSQL = newSQL + " where sodienthoai like '%" +
            sodthoai + "%'";
        else newSQL = newSQL + " and sodienthoai like '%" + sodthoai +
            "%'";
        fsqlstart = false;
    }
if (diachi != null)if (diachi.length() != 0) {
    if (fsqlstart) newSQL = newSQL + " where diachi like '%" + diachi +
        "%'";
    else newSQL = newSQL + " and diachi like '%" + diachi + "%'";
    fsqlstart = false;
}
//xây dựng chuỗi ConnectionString truy xuất database dùng DSN
String conStr = "jdbc:odbc:MyDatabase";
```



Code của trang JSP (tt)

```
try {
    // tạo connection đến database,
    // thực hiện lệnh SQL, kết quả trả về trong recordset rs
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    con = DriverManager.getConnection(conStr,"","");
    Statement stmt = con.createStatement();
    ResultSet rs =stmt.executeQuery(newSQL);
    int i = 1;
    if (rs != null && rs.next()) {
        //nếu có ít nhất 1 account, tạo bảng, tạo tiêu đề và hàng miêu tả
        account đầu tiên
        table.append("<table border=1 cellPadding=1 cellSpacing=1>");
        table.append("<tr><th>So thu tu</th><th>Ten thue bao</th><th>So
dien thoai</th><th>Dia chi</th></tr>\n");
        table.append("<tr><td>" + i++ + "</td><td>" + rs.getString(2) +
                    "</td><td>" + rs.getString(3) + "</td><td>" +
                    rs.getString("diachi") + "</td></tr>\n");
    }
}
```



Code của trang JSP (tt)

```
//lặp xuất từng hàng thông tin account còn lại
while (rs.next()) {
    table.append("<tr><td>" + i++ + "</td><td>" + rs.getString(2) +
    "</td><td>" + rs.getString(3) + "</td><td>" +
    rs.getString("diachi") + "</td></tr>\n");
}
table.append("</table>");
} else
    table.append("<p>Không có account nào thỏa yêu cầu tìm
kiếm!</p>\n");
//đóng các đối tượng đã dùng lại
rs.close(); stmt.close(); con.close();
} catch(Exception e){System.out.println("Error : "+e);}
return table.toString();
}
```

%>



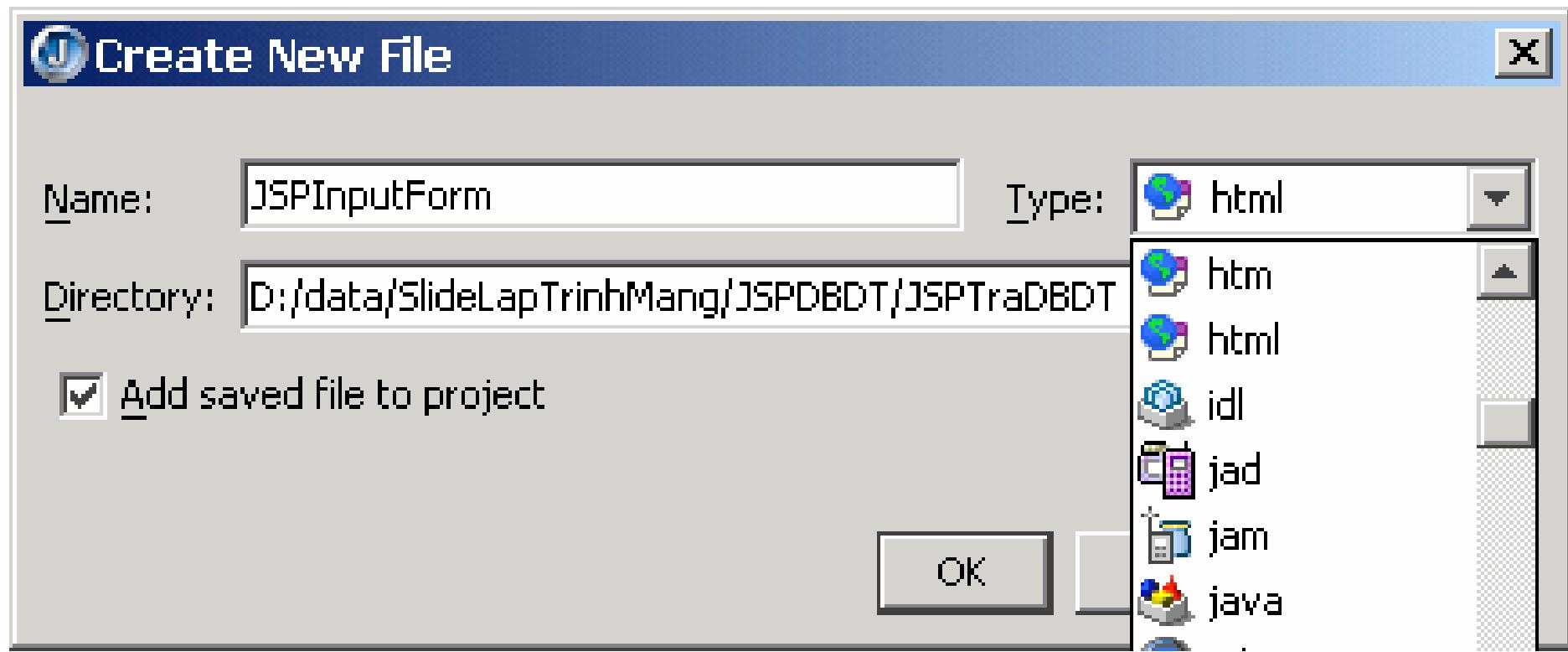
Code của trang JSP (tt)

```
<title>
JSPTraDBDT
</title>
</head>
<body>
<h3>Kết quả tra danh bạ theo yêu cầu của bạn :</h3>
<%=SearchAccount(request.getParameter("txtTenthuebao"),request.getParameter("txtSodienthoai"),request.getParameter("txtDiachi"))%>
<p><a href="JSPInputForm.html">Quay về trang chủ</a></p>
</body>
</html>
```



Tạo trang chủ chứa form nhập liệu cần tra

4. chọn menu File.New File... để hiển thị cửa sổ "Create New File" như hình dưới đây. Chọn thư mục chứa file (lưu ý để server Tomcat quản lý đúng Website, bạn phải chọn thư mục có chứa thư mục con là WEB-INF), nhập tên file HTML là "JSPIInputForm", duyệt tìm và chọn Type file là html.



Viết code cho trang chủ chứa form

```
<HTML><HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=utf8">
<TITLE></TITLE></HEAD>
<BODY>
<font face=Helvetica size="4">
<H2 align=center>Trang Web demo việc tra danh bạ điện thoại</H2>
<FORM id=FORM1 name=FORM1 action=JSPTraDBDT.jsp method=post>
<P>Nhập thông tin về tên thuê bao : <INPUT style="width: 244px; height: 22px" size=34 name=txtTenthuebao></P>
<P>Nhập thông tin về số điện thoại : <INPUT name=txtSodienthoai></P>
<P>Nhập thông tin về địa chỉ : <INPUT style="width: 284px; height: 22px" size=40 name=txtDiachi></P>
<P><INPUT style="width: 144px; height: 24px" type=submit size=45 value="Bắt đầu tra" name=submit1></P></FORM>
</font>
</BODY></HTML>
```



Kết quả chạy trang JSP (chọn menu Run.Run Project)

The screenshot shows the JBuilder 2005 IDE interface. The title bar reads "JBuilder 2005 - D:/data/SlideLapTrinhMang/JSPDBDT/JSPTraDBDT/JSPTraDBDT.jsp". The menu bar includes File, Edit, Search, Refactor, View, Project, Run, Team, Enterprise, Tools, Window, and Help. The toolbar has various icons for file operations like Open, Save, Print, and search. The Project Explorer on the left shows a project named "JSPTraDBDT" with files like "JSPTraDBDT.jsp", "JSPTraDBDT.jpx", and "JSPTraDBDT.html". The JSPTraDBDT folder contains "Deployment descriptors" and "Module directory" which includes "WEB-INF" with "JSPTraDBDT.jsp" and "JSPTraDBDT.html". The main workspace displays a JSP page titled "Kết quả tra danh bù theo yêu cầu của bạn :". Below it is a table showing a list of tenants. At the bottom, there is a link "Quay về trang chủ" and a status bar with the URL "http://Mocalhost:8080/JSPTraDBDT/JSPTraDBDT.jsp" and tabs for Source, View, Web View, Web View Source, and History.

So thu tu	Ten thue bao	So dien thoai	Địa chỉ
1	Nguyen Van Hiep	8644717	8A Lu Gia, P15, Q11, Tp.HCM
2	Huynh Thoai Loan	9070437	245/4 Ly Thuong Kiet, P15, Q11, Tp.HCM
3	Nguyen Luong Hoa	9024437	240 Tran Binh Trong, P4, Q5, Tp.HCM
4	Nguyen Lac Han	2030478	157 Hoang Dieu, P1, Q1, Tp.HCM

[Quay về trang chủ](#)



Kết quả chạy file JSPTraDBDT.jsp

The screenshot shows the JBuilder 2005 IDE interface. The title bar reads "JBuilder 2005 - D:/data/SlideLapTrinhMang/JSPDBDT/JSPTraDBDT/JSPTraDBDT.jsp". The menu bar includes File, Edit, Search, Refactor, View, Project, Run, Team, Enterprise, Tools, Window, Help. The toolbar has various icons for file operations. The Project Explorer on the left shows a project named "JSPDBDT.jpx" containing "JSPInputForm.jsp", "JSPTraDBDT", and "JSPTraDBDT.jsp". The JSPTraDBDT folder contains a deployment descriptor and a WEB-INF folder with "JSPInputForm.html" and "JSPTraDBDT.jsp". The main workspace displays a web page titled "Trang Web demo việc tra danh bùa điền thoại". It contains three input fields labeled "Nhập thông tin về tên thuê bao:", "Nhập thông tin về số điện thoại:", and "Nhập thông tin về địa chỉ:". Below these is a button labeled "Bắt đầu tra". At the bottom, there is a status bar with the URL "http://localhost:8080/JSPTraDBDT/JSPInputForm.html" and tabs for Source, View, Web View, Web View Source, and History.



Thí dụ xây dựng 1 website bằng JSP

Yêu cầu : tạo 1 website dùng công nghệ JSP cho phép người dùng tra danh bạ điện thoại bằng Web Browser.

Phân tích : cần 2 module chức năng :

- module chứa form nhập thông tin về các account cần tra, module này có thể là 1 trang HTML thuần túy.
- và module truy tìm trên database các account thỏa mãn tiêu chuẩn tìm kiếm, module này là 1 trang JSP theo yêu cầu.

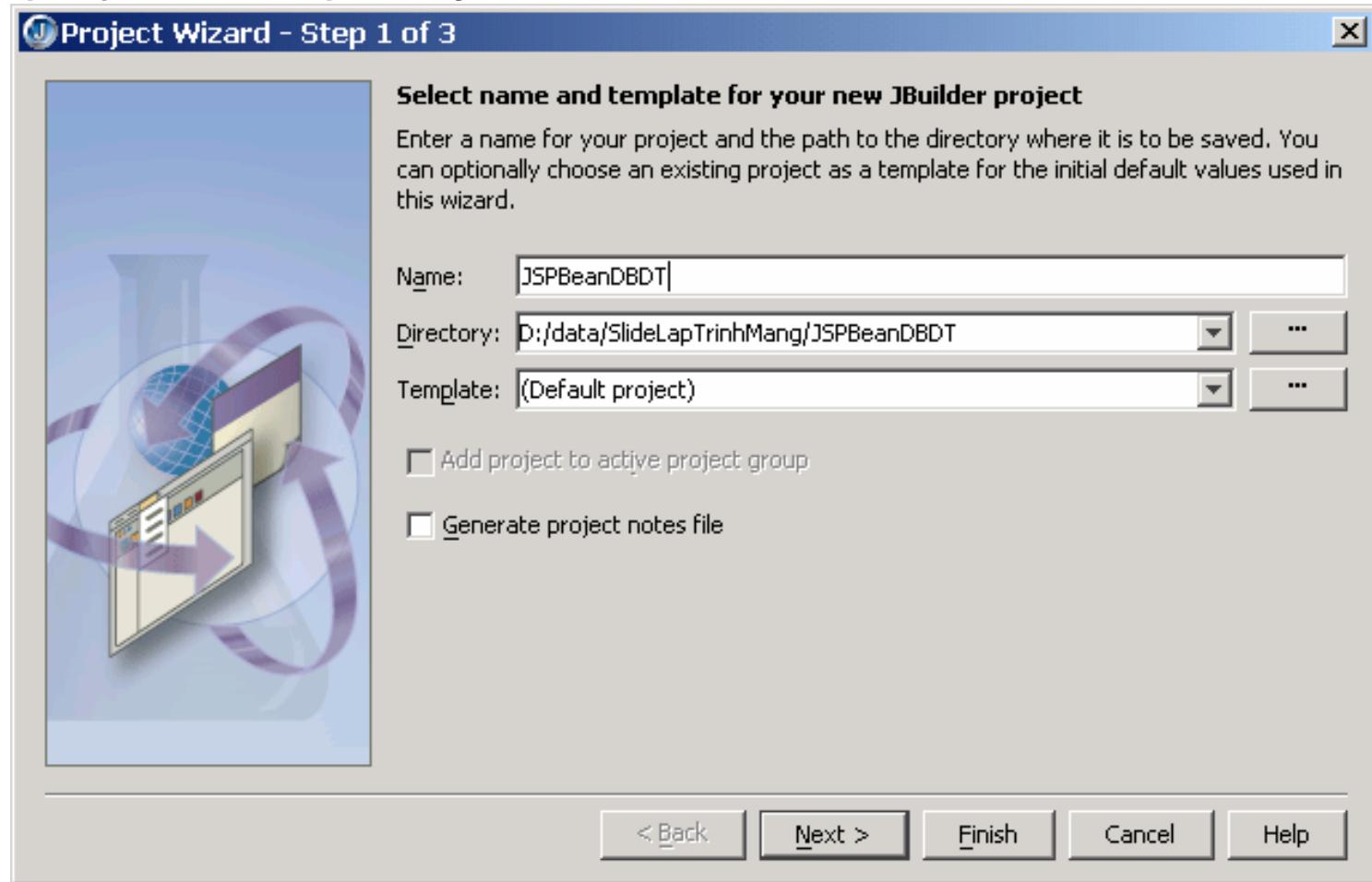
Qui trình thực hiện cụ thể : ở đây ta dùng môi trường lập trình JBuilder 2005. Qui trình điển hình cụ thể được liệt kê trong các slide kế tiếp.



Tạo website tra điện thoại bằng JSP trên JBuilder

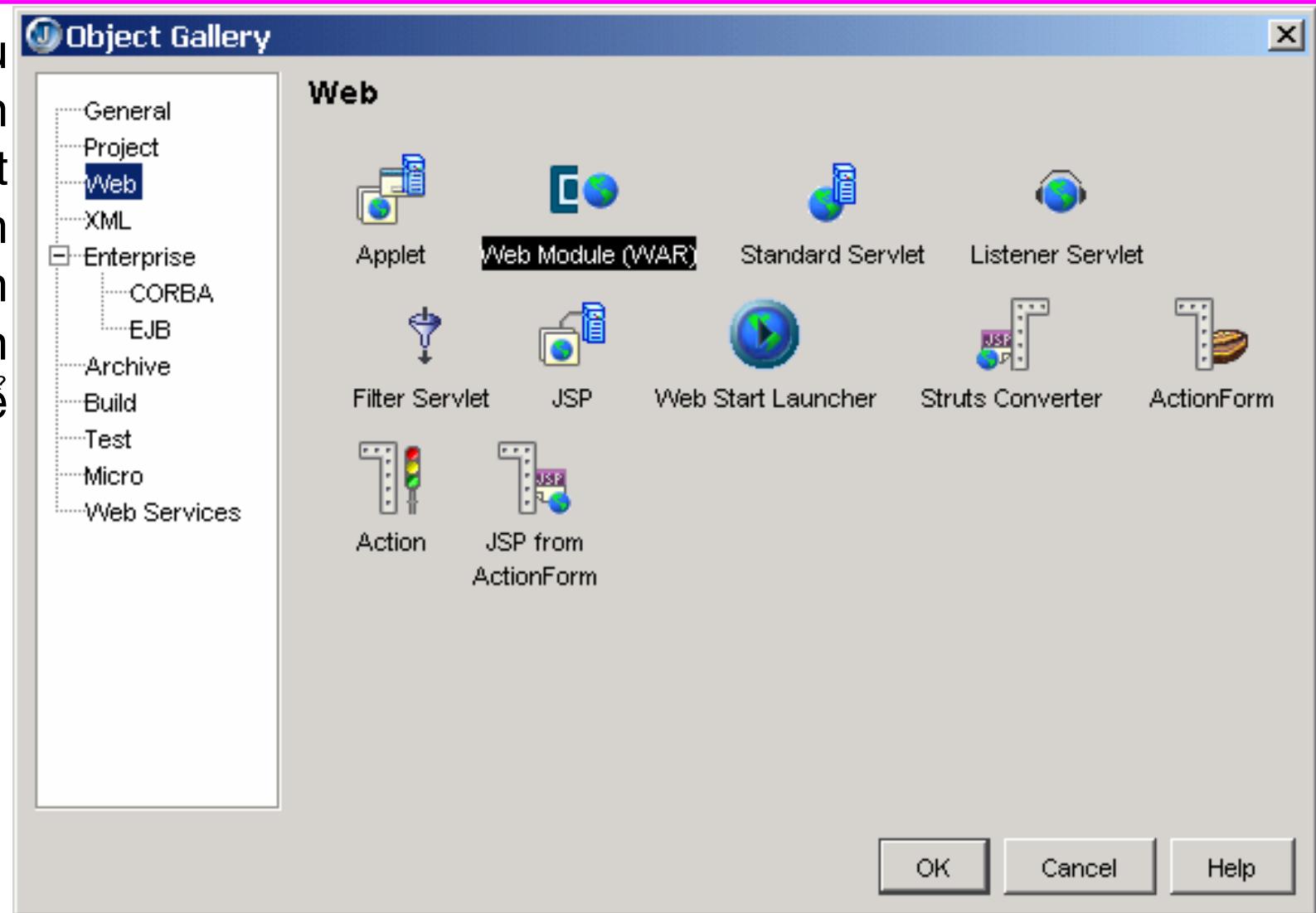
- Chạy JBuilder 2005, chọn menu File.New Project rồi trả lời các cửa sổ Wizard để tạo 1 project mới quản lý Website.

Lưu ý trong cửa sổ Step 1, chọn thư mục và nhập tên Project quản lý Website là JSPDBDT.



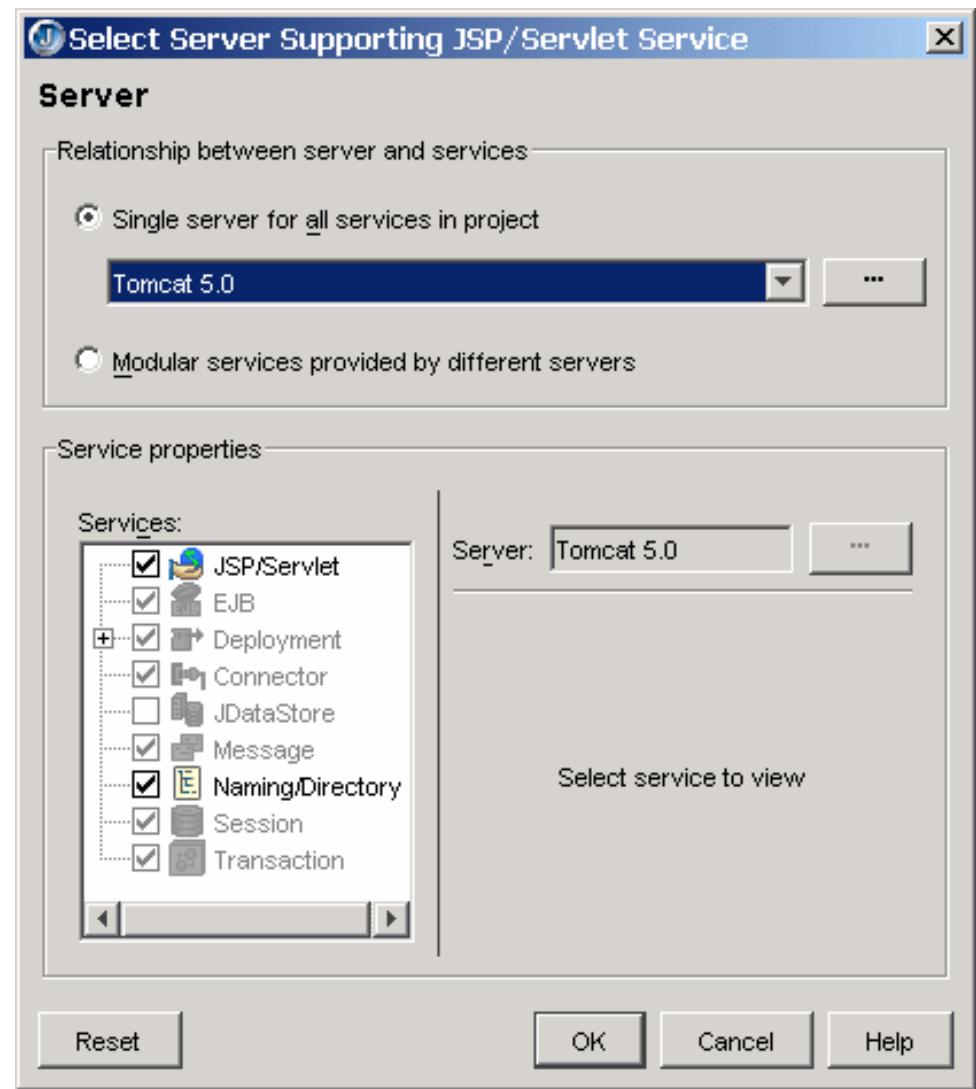
Tạo website tra điện thoại bằng JSP trên JBuilder

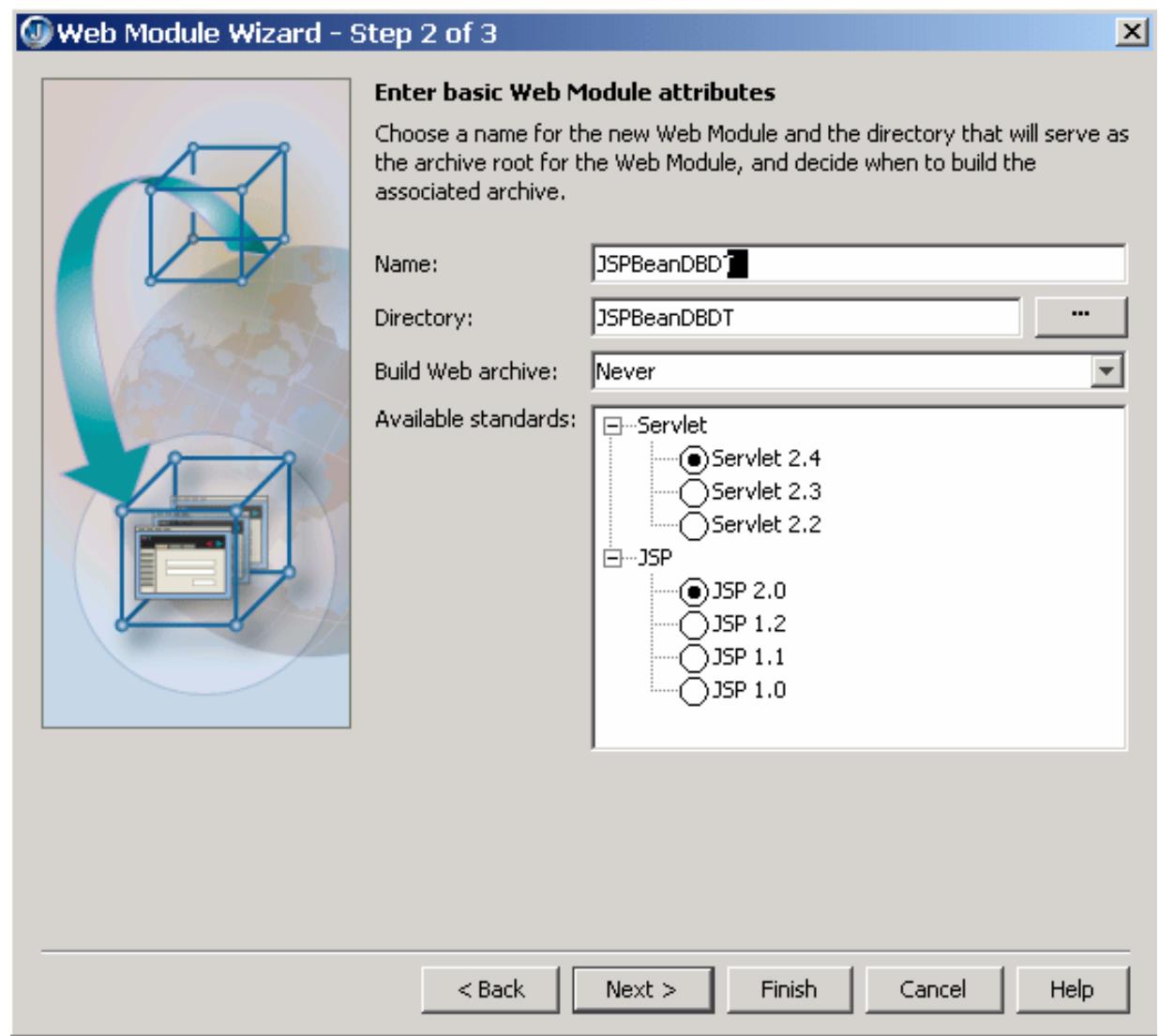
2. chọn menu File.New để hiển thị cửa sổ Object Gallery như hình bên. chọn nhóm Web, chọn icon Web Module (WAR) để tạo Website.



Tạo website tra điện thoại bằng JSP trên JBuilder

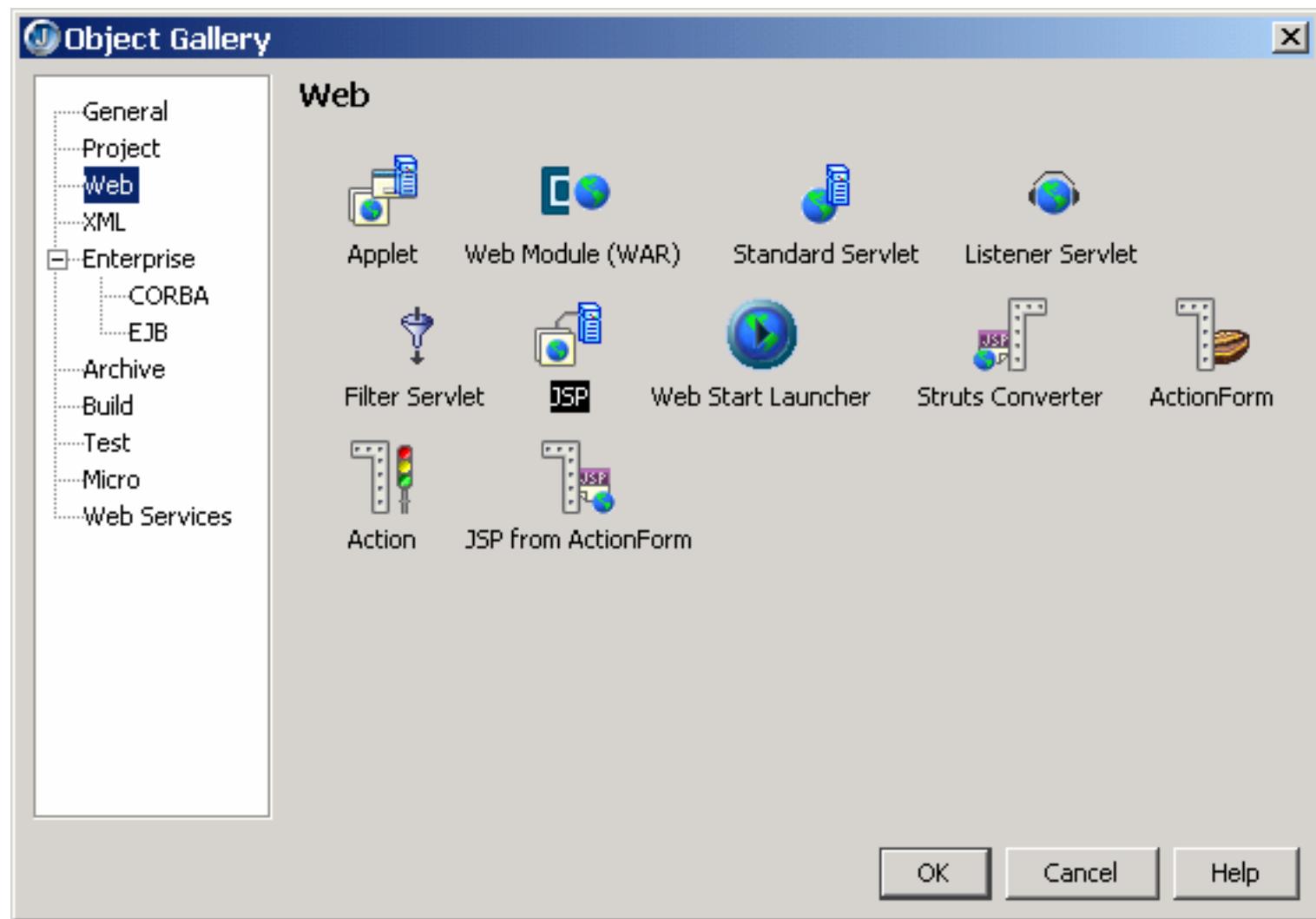
2.1 Khi cửa sổ "Select Server..." hiển thị như hình bên, bạn ấn dấu chọn vào checkbox "Single server...", chọn server "Tomcat 5.0" rồi ấn button OK.





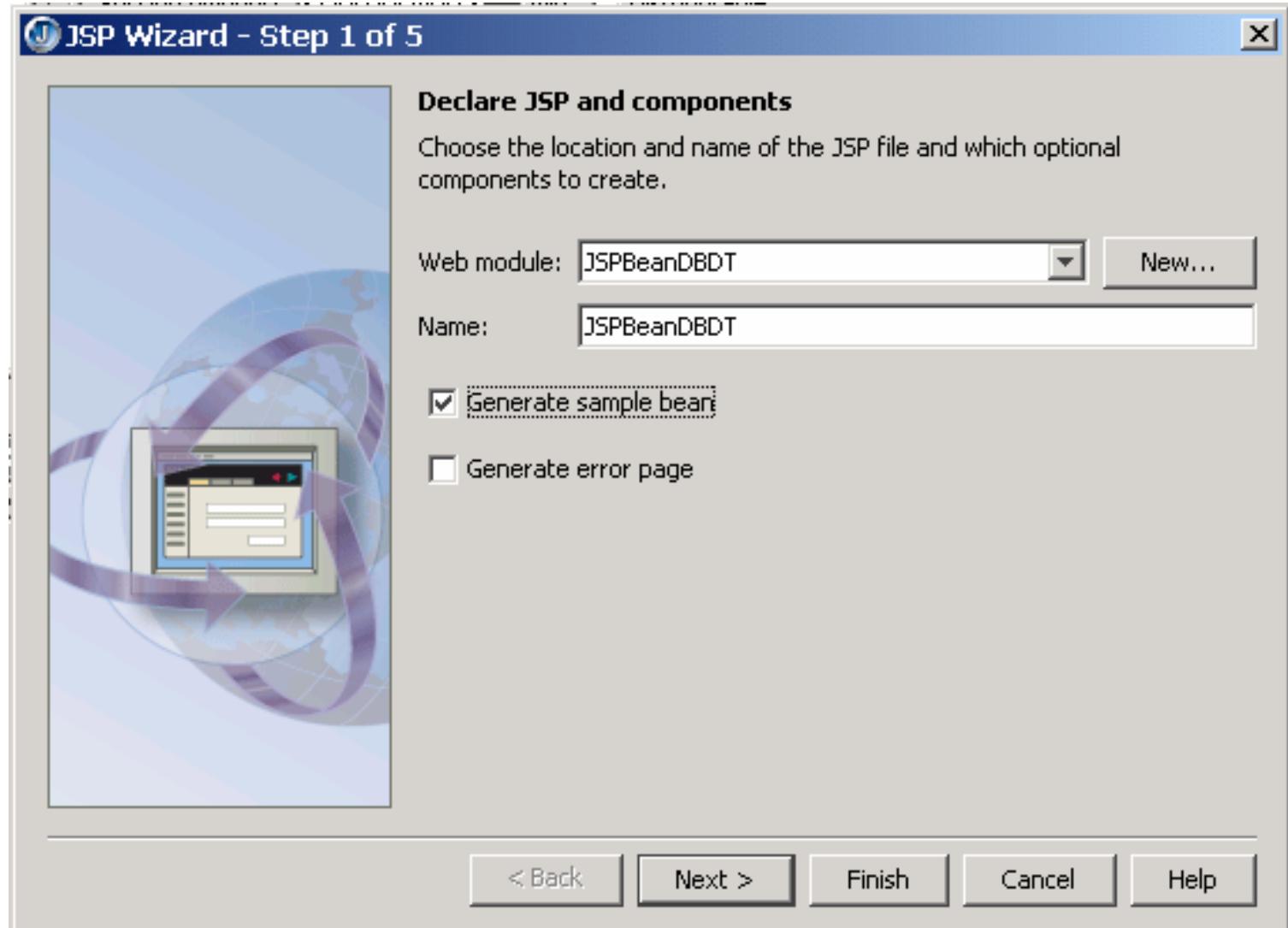
Tạo website tra điện thoại bằng JSP trên JBuilder

3. chọn menu File.New để hiển thị cửa sổ Object Gallery như hình bên. Chọn nhóm Web, chọn icon JSP để tạo 1 trang JSP tra account danh bạ điện thoại, rồi ấn button OK.



Tạo website tra điện thoại bằng JSP trên JBuilder

3.1 trong cửa sổ Step 1 của qui trình JSP Wizard, bạn nhập tên cho trang JSP là JSPTraDBDT, rồi ấn button Next để tiếp tục Step 2 (hay button Finish để chấp nhận các thông số mặc định cho các bước còn lại).



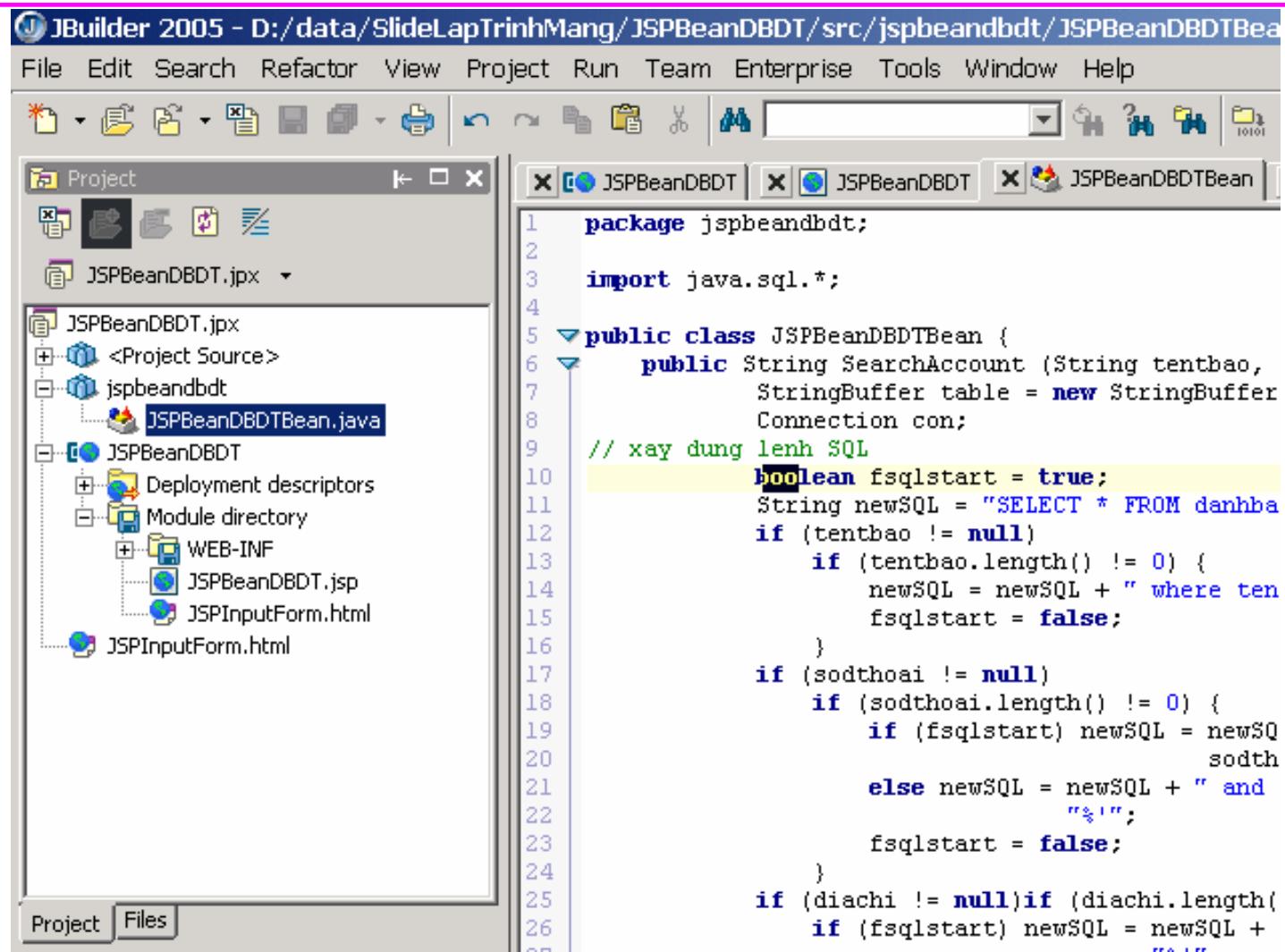
Viết code cho trang JSP như sau

```
<%@ page contentType="text/html; charset=UTF-8" %>
<html><head>
<title>JSPBeanDBDT</title>
</head>
<jsp:useBean id="jSPBeanDBDTBeanId" scope="session"
class="jspbeandbdt.JSPBeanDBDTBean" />
<jsp:setProperty name="jSPBeanDBDTBeanId" property="*" />
<body bgcolor="#ffffff">
<h3>Kết quả tra danh bạ theo yêu cầu của bạn :</h3>
<%=jSPBeanDBDTBeanId.SearchAccount(request.getParameter("txtTenthue
bao"),request.getParameter("txtSodienthoai"),request.getParameter("txtDiachi
"))%>
<p><a href="JSPInputForm.html">Quay về trang chủ</a>
</body>
</html>
```



Viết code JavaBean phục vụ trang JSP

Dời chuột về cửa sổ Project, tìm mục JSPBeanDBDTBean.java, ấn kép chuột vào nó để hiển thị cửa sổ soạn code của JavaBean, rồi soạn code cho nó theo nội dung các slide tiếp theo.



The screenshot shows the JBuilder 2005 IDE interface. The left pane is the Project Explorer, displaying a project named 'JSPBeanDBDT' with files like 'JSPBeanDBDT.java', 'JSPInputForm.html', and 'JSPInputForm.jsp'. The right pane is the Code Editor, showing Java code for a JavaBean named 'JSPBeanDBDTBean'. The code implements a method 'String SearchAccount' which constructs a SQL query based on search criteria. A specific line of code, 'boolean fsqqlstart = true;', is highlighted in yellow.

```
package jspbeandbdt;
import java.sql.*;
public class JSPBeanDBDTBean {
    public String SearchAccount (String tentbao,
        StringBuffer table = new StringBuffer
        Connection con;
    // xay dung lenh SQL
    boolean fsqqlstart = true;
    String newSQL = "SELECT * FROM danhba
    if (tentbao != null)
        if (tentbao.length() != 0)
            newSQL = newSQL + " where ten =
    fsqqlstart = false;
    }
    if (sodthoai != null)
        if (sodthoai.length() != 0) {
            if (fsqqlstart) newSQL = newSQ
                sodth
            else newSQL = newSQL + " and
                "%";
            fsqqlstart = false;
        }
    if (diachi != null)if (diachi.length(
        if (fsqqlstart) newSQL = newSQL +
            "%"
    }
}
```



Viết code JavaBean phục vụ trang JSP (tt)

```
package jspbeandbdt;
import java.sql.*;
//class đặc tả JavaBean
public class JSPBeanDBDTBean {
    public String SearchAccount (String tentbao, String sodthoai, String diachi) {
        StringBuffer table = new StringBuffer();
        Connection con;
        boolean fsqlstart = true;
        // xây dựng lệnh SQL tìm kiếm theo yêu cầu người dùng
        String newSQL = "SELECT * FROM danhbadienthoai";
        if (tentbao != null)
            if (tentbao.length() != 0) {
                newSQL = newSQL + " where tenthuebao like '%" + tentbao + "%'";
                fsqlstart = false;
            }
    }
}
```



Code JavaBean phục vụ trang JSP (tt)

```
if (sodthoai != null)
    if (sodthoai.length() != 0) {
        if (fsqlstart) newSQL = newSQL + " where sodienthoai like '%" +
            sodthoai + "%'";
        else newSQL = newSQL + " and sodienthoai like '%" + sodthoai +
            "%'";
        fsqlstart = false;
    }
if (diachi != null)if (diachi.length() != 0) {
    if (fsqlstart) newSQL = newSQL + " where diachi like '%" + diachi +
        "%'";
    else newSQL = newSQL + " and diachi like '%" + diachi + "%'";
    fsqlstart = false;
}
//xây dựng chuỗi ConnectionString truy xuất database dùng DSN
String conStr = "jdbc:odbc:MyDatabase";
```



Code JavaBean phục vụ trang JSP (tt)

```
try {
    // tạo connection đến database,
    // thực hiện lệnh SQL, kết quả trả về trong recordset rs
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    con = DriverManager.getConnection(conStr,"","");
    Statement stmt = con.createStatement();
    ResultSet rs =stmt.executeQuery(newSQL);
    int i = 1;
    if (rs != null && rs.next()) {
        //nếu có ít nhất 1 account, tạo bảng, tạo tiêu đề và hàng miêu tả
        account đầu tiên
        table.append("<table border=1 cellPadding=1 cellSpacing=1>");
        table.append("<tr><th>So thu tu</th><th>Ten thue bao</th><th>So
dien thoai</th><th>Dia chi</th></tr>\n");
        table.append("<tr><td>" + i++ + "</td><td>" + rs.getString(2) +
                    "</td><td>" + rs.getString(3) + "</td><td>" +
                    rs.getString("diachi") + "</td></tr>\n");
    }
}
```



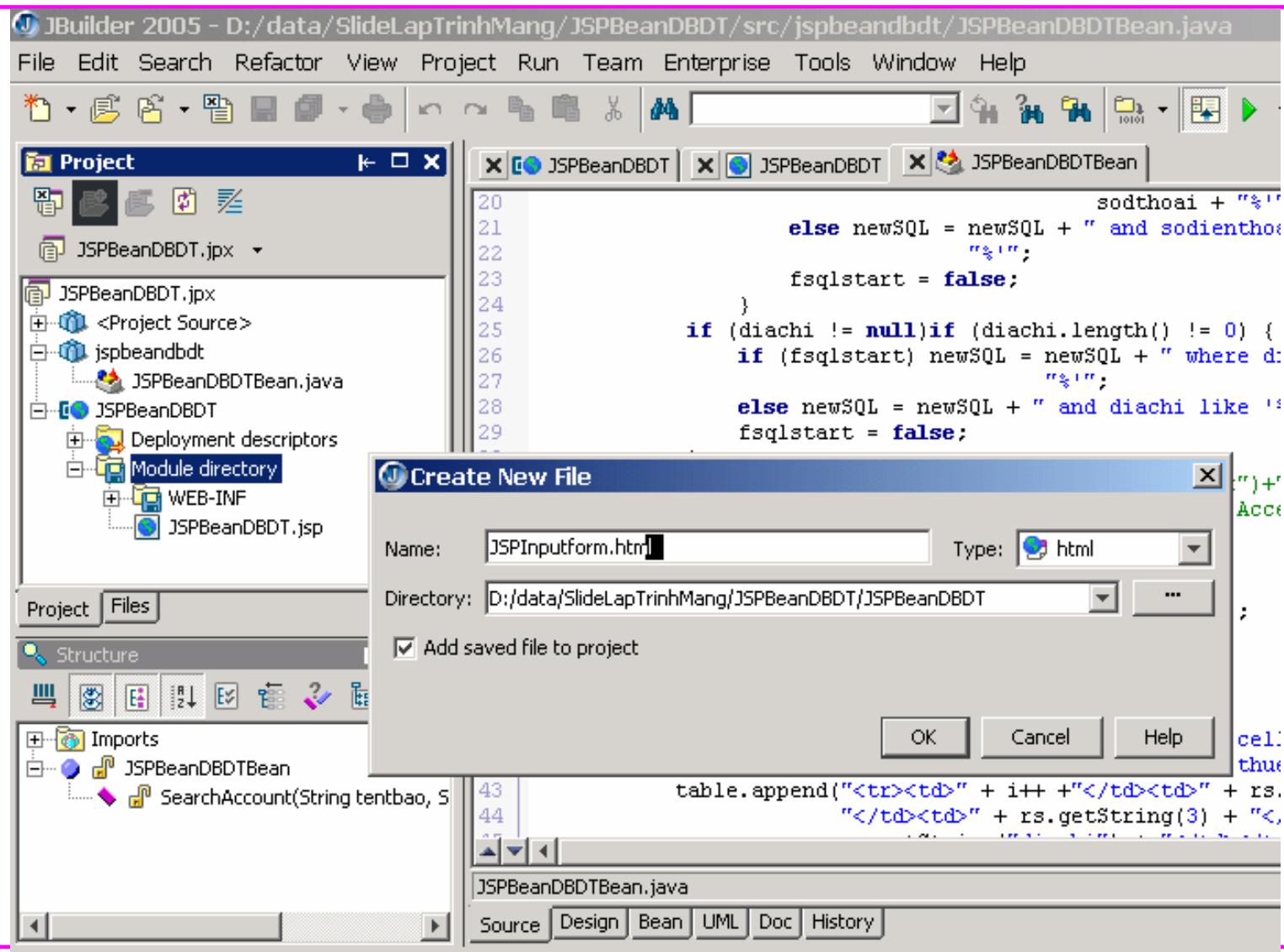
Code JavaBean phục vụ trang JSP (tt)

```
//lặp xuất từng hàng thông tin account còn lại
while (rs.next()) {
    table.append("<tr><td>" + i++ + "</td><td>" + rs.getString(2) +
    "</td><td>" + rs.getString(3) + "</td><td>" +
    rs.getString("diachi") + "</td></tr>\n");
}
table.append("</table>");
} else
    table.append("<p>Không có account nào thỏa yêu cầu tìm
kiếm!</p>\n");
//đóng các đối tượng đã dùng lại
rs.close(); stmt.close(); con.close();
} catch(Exception e){System.out.println("Error : "+e);}
return table.toString();
}//end of SearchAccount
}//end of JavaBean
```



Tạo trang chủ chứa form nhập liệu cần tra

4. chọn menu File.New File... để hiển thị cửa sổ "Create New File" như hình dưới đây. Chọn thư mục chứa file (lưu ý để server Tomcat quản lý đúng Website, bạn phải chọn thư mục có chứa thư mục con là WEB-INF), nhập tên file HTML là "JSPInputForm", duyệt tìm và chọn Type file là html.



Viết code cho trang chủ chứa form

```
<HTML><HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=utf8">
<TITLE></TITLE></HEAD>
<BODY>
<font face=Helvetica size="4">
<H2 align=center>Trang Web demo việc tra danh bạ điện thoại</H2>
<FORM id=FORM1 name=FORM1 action=JSPTraDBDT.jsp method=post>
<P>Nhập thông tin về tên thuê bao : <INPUT style="width: 244px; height: 22px" size=34 name=txtTenthuebao></P>
<P>Nhập thông tin về số điện thoại : <INPUT name=txtSodienthoai></P>
<P>Nhập thông tin về địa chỉ : <INPUT style="width: 284px; height: 22px" size=40 name=txtDiachi></P>
<P><INPUT style="width: 144px; height: 24px" type=submit size=45 value="Bắt đầu tra" name=submit1></P></FORM>
</font>
</BODY></HTML>
```



Kết quả chạy trang JSP (chọn menu Run.Run Project)

The screenshot shows the JBuilder 2005 IDE interface with the following details:

- Title Bar:** JBuilder 2005 - D:/data/SlideLapTrinhMang/JSPBeanDBDT/JSPBeanDBDT/JSPBeanDBDT.jsp
- Menu Bar:** File, Edit, Search, Refactor, View, Project, Run, Team, Enterprise, Tools, Window, Help
- Toolbar:** Standard icons for file operations like Open, Save, Print, etc.
- Project Explorer:** Shows the project structure:
 - JSPBeanDBDT.jpx
 - <Project Source>
 - jspbeandbdt
 - JSPBeanDBDTBean.java
 - JSPBeanDBDT
 - Deployment descriptor
 - Module directory
 - WEB-INF
 - JSPBeanDBDT.jsp
 - JSPInputForm.htm
 - JSPInputForm.html
- Toolbars:** Standard Java development toolbars.
- Central Area:** Displays the output of the JSP page.

Kết quả tra danh bù theo yêu cầu của bạn :

So thu tu	Ten thue bao	Số điện thoại	Địa chỉ
1	Nguyen Van Hiep	8644717	8A Lu Gia, P15, Q11, Tp.HCM
2	Huynh Thoai Loan	9070437	245/4 Ly Thuong Kiet, P15, Q11, Tp.HCM
3	Nguyen Luong Hoa	9024437	240 Tran Binh Trong, P4, Q5, Tp.HCM
4	Nguyen Lac Han	2030478	157 Hoang Dieu, P1, Q1, Tp.HCM

[Quay về trang chủ](#)

http://localhost:8080/JSPBeanDBDT/JSPBeanDBDT.jsp

Source View Web View Web View Source History
- Messages Window:** Shows deployment logs:

```
INFO: Create Host deployer for direct deployment ( non-jmx )
Nov 30, 2006 10:30:44 AM org.apache.coyote.http11.Http11Protocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
```
- Task List:** Shows a task for Tomcat 5.0 http://8080.

Kết quả chạy file JSPIInputForm (chọn "Quay ve trang chu")

The screenshot shows the JBuilder 2005 IDE interface. The title bar reads "JBuilder 2005 - D:/data/SlideLapTrinhMang/JSPBeanDBDT/JSPBeanDBDT/JSPBeanDBDT.jsp". The menu bar includes File, Edit, Search, Refactor, View, Project, Run, Team, Enterprise, Tools, Window, Help. The toolbar has various icons for project management and file operations. The left pane is the "Project" view, showing the project structure: "JSPBeanDBDT.jpx" containing "Project Source", "jspbeandbdt" (with "JSPBeanDBDTBean"), "JSPBeanDBDT" (with "Deployment descriptor" and "Module directory" containing "WEB-INF" and "JSPBeanDBDT" and "JSPIInputForm"), and "JSPIInputForm.html". The right pane displays the "JSPIInputForm" page. The page title is "Trang Web demo vi c tra danh b d n tho i". It contains three input fields labeled "Nh p th ng tin v t tnh thu bao:", "Nh p th ng tin v s d n tho i:", and "Nh p th ng tin v d a ch:". Below these is a button labeled "B t đ u tra". At the bottom, the URL "http://Mocalhost:8080/JSPBeanDBDT/JSPIInputForm.html" is shown in the browser's address bar, along with tabs for Source, View, Web View, Web View Source, and History.

