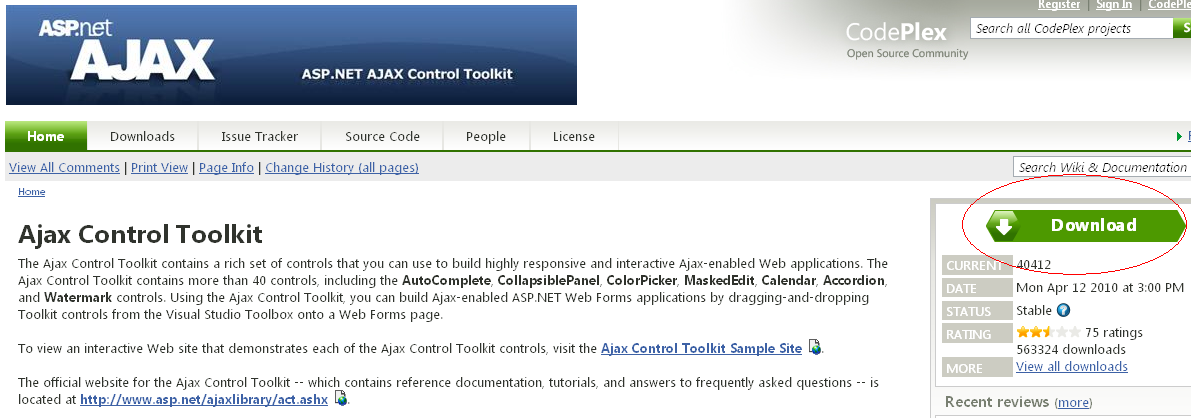
1. AJAX Control Toolkit?

Là một tập hợp các control được xây dựng sẵn để hỗ trỡ cho việc lập trình ASP.NET và Ajax được phát triển bởi Microsoft.

1. Hướng dẫn cài đặt
   1. Môi trường:

* Microsoft Visual Studio 2008
* .NET Framework 3.5
  1. Cài đặt

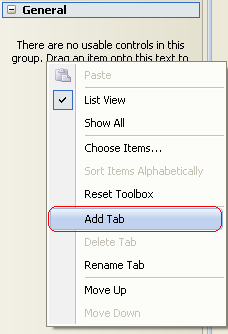
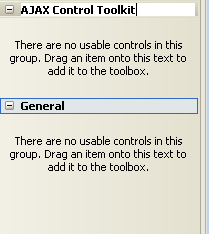
Download AJAX Control Toolkit tại <http://ajaxcontroltoolkit.codeplex.com/>



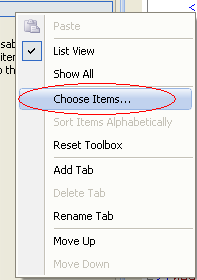
Giải nén thư mục vừa download.

Add thư viện vào VS:

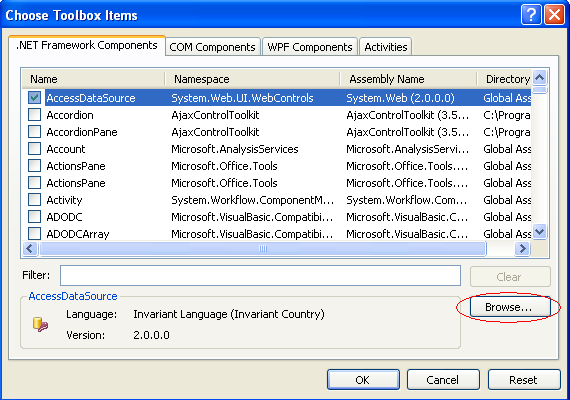
Click phải vào hộp thoại Toolbox của Visual Studio, chọn Add Tab, điền tên tab mới cho toolbox: AJAX Control Toolkit

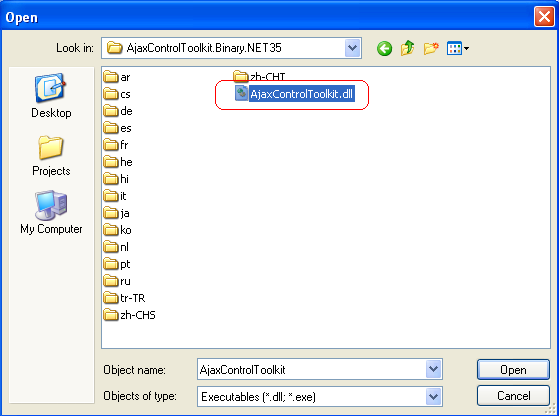
 

Click phải vào tab mới tạo, chọn Choose Items…

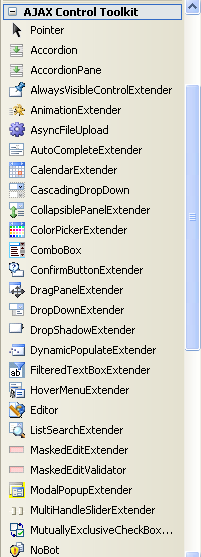


Chọn đường dẫn tới dll





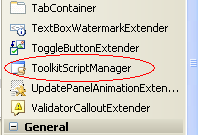
Các control sau khi được thêm vào:



1. Hướng dẫn sử dụng một số control thông dụng
   1. ToolkitScriptManager

Là công cụ quản lý, phân phối và điều khiển tất cả các script được sinh ra bởi việc sử dụng Ajax Control Toolkit.

Muốn sử dụng được tất cả các control của Ajax Control Toolkit đều phải sử dụng ToolkitScriptManager trước.



* 1. Accordion

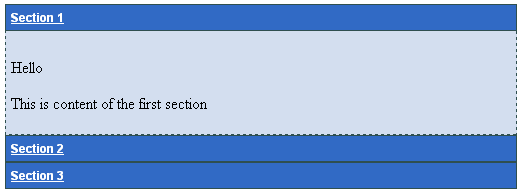
Accordion là một control cho phép cung cấp nhiều vùng hay ô và hiển thị chúng ở cùng một thời điểm. Control này giống dạng như một Panel có thể co giãn nhưng chỉ có một vùng có thể được mở rộng trong một thời điểm.

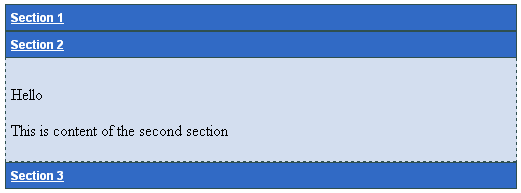
Control Accordion chứa nhiều control AccordionPane, mỗi AccordionPane có tiêu đề (Header) và nội dung (Content) của nó.

Các thuộc tính của Accordion:

* HeaderCssClass: tên của lớp Css được dùng cho tiêu đề.
* HeaderSelectedCssClass: tên của lớp Css được dùng cho tiêu đề sau khi được chọn.
* ContentCssClass: tên của lớp Css được dùng cho phần nội dung.
* FadeTransitions: nếu giá trị true thì dùng để làm mờ dần trong các hiệu ứng chuyển tiếp, nếu là false chuyển tiếp bình thường.
* TransitionDuration: số mili giây chuyển tiếp.
* FramesPerSecond: số khung hình trên giây được sử dụng trong chuyển tiếp hình ảnh.
* Panes: tập hợp chứa các control AccordionPane.
* ….

|  |
| --- |
| <form id="form1" runat="server">  <div>  <asp:ToolkitScriptManager ID="ToolkitScriptManager2" runat="server">  </asp:ToolkitScriptManager>  <asp:Accordion ID="Accordion1" runat="server" HeaderCssClass="accordionHeader"  ContentCssClass="accordionContent">  <Panes>  <asp:AccordionPane ID="AccordionPane1" runat="server" >  <Header>  <a href = "" class = "accordionLink"> Section 1</a>  </Header>  <Content>  <p>Hello</p>  <p>This is content of the first section</p>  </Content>  </asp:AccordionPane>  <asp:AccordionPane ID="AccordionPane2" runat="server">  <Header>  <a href = "" class = "accordionLink"> Section 2</a>  </Header>  <Content>  <p>Hello</p>  <p>This is content of the second section</p>  </Content>  </asp:AccordionPane>  <asp:AccordionPane ID="AccordionPane3" runat="server">  <Header>  <a href = "" class = "accordionLink"> Section 3</a>  </Header>  <Content>  <p>Hello</p>  <p>This is content of the third section</p>  </Content>  </asp:AccordionPane>  </Panes>  </asp:Accordion>  </div>  </form> |





* 1. Animation

Lớp AnimationExtender cho phép sử dụng một framework hoạt hình mạnh mẽ với trang hiện tại một cách dễ dàng. Nó phát hình ảnh động bất cứ khi nào có một sự kiện đặc biệt xảy ra như Onload, OnClick, OnMouseOver, hoặc OnMouseOut được nâng lên bởi control đích.

Framework này cung cấp những hình ảnh động để xử lý chuyển động, thay đổi kích thước, mờ dần, …

|  |
| --- |
| <asp:AnimationExtender ID="OpenAnimation" TargetControlID="btnInfo" runat="server">  <Animations>  <OnClick>  <Sequence>  //Xử lý sự kiện onclick  </Sequence>  </OnClick>  </Animations>  </asp:AnimationExtender> |

Thuộc tính TargetControlID chứa id của control phát sinh animation khi xảy ra sự kiện.



* 1. AlwaysVisibleControlExtender

Bất kể control nào được extend bởi control AlwaysVisibleControlExtender sẽ luôn luôn được hiển thị.

|  |
| --- |
| <asp:ScriptManager ID="ScriptManager1" runat="server">  </asp:ScriptManager>  <ajaxToolkit:AlwaysVisibleControlExtender runat="server"  ID="AlwaysVisibleControlExtender1"  TargetControlID="allwaysOnMessage"  </ajaxToolkit:AlwaysVisibleControlExtender> |

Thuộc tính TargetControlID là id của control sẽ luôn luôn được hiển thị.

* 1. CalendarExtender

Control CalendarExtender cho phép hiển thị một hộp thoại chọn ngày. Calendar là một ASP AJAX mở rộng, có thể được gắn vào bất kỳ control Textbox.

Có thể tương tác với control bằng cách bấm vào một ngày để thiết lập ngày mặc định.

Các mũi tên trái, phải được sử dụng để di chuyển qua tháng trước đó hoặc tháng tiếp theo.

|  |
| --- |
| <div>  <asp:ToolkitScriptManager ID="ToolkitScriptManager1" runat="server">  </asp:ToolkitScriptManager>  <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>  <img src=Calendar\_scheduleHS.png runat="server" id="Image"/>  <asp:CalendarExtender ID="CalendarExtender1" runat="server" TargetControlID="TextBox1" PopupButtonID="Image">  </asp:CalendarExtender>  </div> |

* 1. ColorPickerExtender

Control ColorPickerExtender cho phép hiển thị hộp thoại chọn màu sắc. Có thể đính kèm control này với bất cừ control textbox nào của ASP.NET.



Các thuộc tính

* TargetControlID: id của control texbox để mở rộng.
* PopupButtonID: id của control sử dụng để hiển thị bảng chọn màu. Nếu giá trị này không được thiết lập, các bảng chọn màu sẽ xuất hiện khi control Textbox được đại diện bởi TargetControlID nhận được focus.
* SampleControlID: id của control sử dụng để hiển thị màu được chọn, màu nền của control này hiển thị màu mà chuột chọn.
* PopupPosition: chỉ ra nơi popup chọn màu sẽ xuất hiện liên quan tới control textbox đang được mở rộng. Các giá trị có thể: BottomLeft, BottonRight, TopLeft, TopRight, Left hoặc Right. Mặc định là BottomLeft.
* SelectedColor: Giá trị màu được chọn.
* OnClientColorSelectionChanged: một hàm Javascript sẽ được gọi khi các sự kiện colorSelectionChanged được nâng lên.
  1. ConfirmButonExtender

ConfirmButton là một phần mở rộng bắt sự kiện click vào button, sẽ hiển thị thông báo cho người dùng.

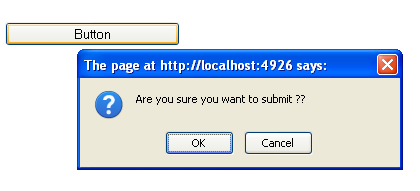
* Nếu người dùng click OK, các chức năng button và liên kết bình thường.
* Nếu người dùng không nhấp OK, sự kiện click bị mắc kẹt và các nút không thực hiện hành vi mặc định của chương trình.
* Người dùng có thể tùy chọn xác định kịch bản client để thực hiện khi các nút trong hộp thoại được xác nhận.

ConfirmButton thường được sử dụng để yêu cầu người dùng xác nhận một hành động nào đó.

Các thuộc tính:

* TargetControlID: Các ID của button để mở rộng.
* ConfirmText: Đoạn văn bản xác nhận để hiển thị. Các thực thể HTML có thể được sử dụng như cho 1 ký tự xuống dòng.
* OnClientCancel: script trên máy client khi người dùng click vào nút Cancel trong hộp thoại xác nhận.
* ConfirmOnFormSubmit: true nếu hộp thoại xác nhận không được hiển thị ngay trước khi form được submit. Điều này hữu ích khi trang ASP.Net chứa các control validator và hộp thoại xác nhận sẽ chỉ được hiển thị sau khi các kiểm tra xác nhận đã thành công.
* DisplayModalPopupID: id của control ModalPopup sử dụng để hiển thị hộp thoại xác nhận thay vì mặc định của window: confirmwindow. Khi sử dụng thuộc tính DisplayModalPopupID, các điều kiện sau phải được đáp ứng:
  + ModalPopup phải được cấu hình với với giá trị TargetControlID như ConfirmButtonExtender..
  + Điều khiển ModalPopup phải ghi rõ OkControlID hoặc CancelControlID để xác nhận các nút tương ứng với các nút OK, Cancel trong hộp thoại xác nhận.
  + ModalPopup không phải chỉ định một thuộc tính OnOkScript hoặc OnCancelScript.

|  |
| --- |
| <asp:ScriptManager ID="ScriptManager1" runat="server" />  <div>  <br />  <asp:Label ID="Label1" runat="server" Width="273px"></asp:Label><br />  <br />  <asp:Button ID="Button1" runat="server" OnClick="Button1\_Click" Text="Button" Width="175px" />  <ajaxToolkit:ConfirmButtonExtender ID="ConfirmButtonExtender1" runat="server" TargetControlID="Button1" ConfirmText="Are you sure you want to submit ??">  </ajaxToolkit:ConfirmButtonExtender>  </div> |



* 1. FilteredTextboxExtender

FilteredTextboxExtender là một control cho phép người dùng chỉ được nhập ký tự mà bạn xác định vào textbox hoặc ngăn cản người dùng nhập ký tự mà bạn chỉ định.

Vì control thực hiện dựa trên Javascript, không nên cho rằng các dữ liệu được gửi tới server chỉ gồm những ký tự bạn cho phép, luôn thực hiện kiểm tra xác nhận ở phía server trên dữ liệu được gửi từ client.

Các thuộc tính:

* TargetControlID: các ID của control cần kiểm tra.
* FilterType: Các loại bộ lọc được áp dụng, quy trình như một sự kết hợp cách nhau bằng dấu phẩy các giá trị: Numbers, LowercaseLetters, UppercaseLetters, và Custom.
* FilterMode: các chế độ lọc được áp dụng, những giá trị được hỗ trợ là ValidChars và InValidChars. Nếu thuộc tính được thiết lập InvalidChars, FilterType phải được thiết lập dạng Custom.
* ValidChars: một chuỗi bao gồm tất cả các ký tự được xem là hợp lệ cho textbox khi kiểu trường là Custom. Nếu kiểu trường không ở dạng Custom, giá trị thuộc tính được bỏ qua.
* InvalidChars: một chuỗi bao gồm tất cả các ký tự được xem là không hợp lệ cho textbox khi kiểu trường là Custom. Nếu kiểu trường không ở dạng Custom, giá trị thuộc tính được bỏ qua.
* FilterInterval: một số nguyên chỉ định khoảng thời gian tính bằng mili giây mà nội dung của các trường được lọc, mặc định là 250.

|  |
| --- |
| <cc1:FilteredTextBoxExtender ID="FilteredTextBoxExtender1"  runat="server" TargetControlID="TextBox1" FilterType="Numbers">  </cc1:FilteredTextBoxExtender>  <cc1:FilteredTextBoxExtender ID="FilteredTextBoxExtender2"  runat="server" TargetControlID="TextBox2" FilterType="LowercaseLetters">  </cc1:FilteredTextBoxExtender>  <cc1:FilteredTextBoxExtender ID="FilteredTextBoxExtender3"  runat="server" TargetControlID="TextBox3" FilterType="Custom, Numbers" ValidChars="+-=/\*().">  </cc1:FilteredTextBoxExtender> |

* 1. HoverMenuExtender

HoverMenuExtender là một control mở rộng mà có thể gắn vào bất kỳ một WebControl Asp.Net và kết hợp control đó với một bảng nội dung hiển thị thêm.

* Các bảng popup được hiển thị ở một vị trí xác định bởi các lập trình viên (trái, phải, trên hoặc phía dưới của control chính).

Các thuộc tính:

* TargetControlID: id của control mà khi xảy ra một sự kiện ở control này thì các popup menu được hiển thị.
* PopupControlID: id của control hiển thị khi chuột được rê lên trên control target.
* HoverCssClass: lớp CSS để áp dụng cho các popup menu khi rê chuột có thể nhìn thấy.
* PopupPosition: Nơi mà popup được bố trí so với TargetControl: trái, phải, trên, dưới, trung tâm.
* OffsetX/OffsetY: số lượng điểm ảnh cho khoảng trống của Popup so với vị trí mặc định của nó.
* HoverDelay: thời gian tính bằng mili giây, trước khi popup hiển thị sau khi lơ lửng trên control target. Mặc định là 0.
* PopDelay: Thời gian tính bằng mili giây, cho popup giữ nguyên sau khi con chuột di chuyển ra khỏi control target. Mặc định là 100.
* Animations: Những animation chung cho HoverMenuExtender.
* OnShow: Những animation hiển thị mỗi khi hovermenu được hiển thị. Các hình ảnh động có thể sử dụng để hiển thị hover menu cùng với các hình ảnh khác.
* OnHide: Các animation OnHide sẽ được phát mỗi lần di chuyển là ẩn menu.

|  |
| --- |
| <cc1:HoverMenuExtender ID="HoverMenuExtender1"  runat="server"  TargetControlID="btnProduct"  PopupControlID="PanelPopUp"  PopupPosition="bottom"  OffsetX="6"  PopDelay="25" HoverCssClass="popupHover">  </cc1:HoverMenuExtender>  <asp:Panel ID="PanelPopUp" runat="server" Height="50px" Width="125px" CssClass="popupMenu">  <asp:LinkButton ID="BtnUpDate" runat="server" CausesValidation="true" CommandName="Update" Text="Update Record"></asp:LinkButton>  <asp:LinkButton ID="BtnCancel" runat="server" CausesValidation="false" CommandName="Cancel" Text="Cancel"></asp:LinkButton>  </asp:Panel>  <asp:LinkButton ID="btnProduct" runat="server">Click here</asp:LinkButton> |

* 1. PasswordStrength

PasswordStrength là một control mở rộng của ASP.NET AJAX được gắn vào một control Textbox được sử dụng cho các mục nhập của mật khẩu. Control PasswordStrength cho biết sức mạnh của mật khẩu trong các textbox và tự cập nhật khi người dùng gõ mật khẩu.

Các chỉ số có thể hiển thị độ mạnh của mật khẩu dưới dạng tin nhắn văn bản hoặc với một thanh hiển thị tiến trình.

Kiểu và vị trí của cả 2 loại chỉ số có thể cấu hình.

Độ mạnh của mật khẩu cũng có thể cấu hình, cho phép lập trình viên cấu hình độ mạnh của mật khẩu.

Một số thuộc tính:

* TargetControlID: id của textbox để đính kèm vào.
* DisplayPosition: vị trí hiển thị chỉ số độ mạnh của mật khẩu.
* StrengthIndicatorType: loại chỉ số độ mạnh (văn bản hay thanh chỉ thị).
* PreferredPasswordLength: Độ dài mật khẩu được ưu tiên.
* PrefixText: đoạn văn bản tiền tố để hiển thị khi StrengthIndicatorType=text.
* TextCssClass: lớp Css áp dụng khi đoạn văn bản hiển thị khi StrengthIndicatorType=text.
* MinimumNumericCharacters: tối thiểu số ký tự số.
* MinimumSymbolCharacters: tối thiểu ký tự biểu tượng ($ ^ \* …).
* RequiresUpperAndLowerCaseCharacters: nếu trường hợp pha trộn các ký tự được yêu cầu.
* MinimumLowerCaseCharacters: chỉ có hiệu lực nếu có thuộc tính RequiresUpperAndLowerCaseCharacters. Chỉ định số lượng tối thiểu của các ký tự là ký tự thường yêu cầu khi cần trường hợp hỗn hợp như một phần độ mạnh của mật khẩu.
* MinimumUpperCaseCharacters: chỉ có hiệu lực nếu có thuộc tính RequiresUpperAndLowerCaseCharacters. Chỉ định số lượng của các ký tự là chữ hoa cần thiết khicaafn trường hợp hỗn hợp như như một phần độ mạnh của mật khẩu.
* TextStrengthDescriptions: danh sách những mô tả sử dụng khi StrengthIndicatorType=text (nhỏ nhất:2 lớn nhất: 10, trật tự từ yếu tới mạnh nhất).
* CalculationWeightings: danh sách những giá trị có dấu chấm phẩy ngăn cách số được sử dụng để xác định trọng số của một điểm mạnh.
* …

