Khoa Công nghệ thông tin

Môn: Công nghệ Web và Ứng dụng

**BÁO CÁO BÀI TẬP LẦN 2**

Sinh viên thực hiện:

* MSSV: 0512175
* Họ tên: Nguyễn Đăng Khoa

**Nội dung báo cáo:**

1. **Cấu trúc chung của từng ứng dụng:** các đoạn javascript không nhúng trực tiếp vào file HTML mà được đặt trong các file .js. Gồm một file sau:
   * + File .html: chứa giao diện của ứng dụng, không chứa mã xử lý.
     + Events.js: chứa các hàm dùng để xử lý các sự kiện phát sinh từ các thẻ, như sự kiện: onclick, onmousemove, …
     + Functions.js: chứa các hàm dùng để xử lý các công việc cụ thể như kiểm tra lỗi, đọc file XML, … Các hàm ở file Events.js trong quá trình xử lý sẽ gọi các hàm ở file Functions.js.
     + Các file có tên bắt đầu bằng “\_”: chứa các hàm dùng để xử lý lớp, sẽ được dùng lại cho các ứng dụng khác. Ví dụ: file \_XML.js chứa hàm đọc file XML, \_Number.js chứa các hàm kiểm tra số dương, kiểm tra một chuỗi nhập vào có phải là số hay không, …
     + Thư mục images: chứa các file ảnh.
2. **Bài tập 1:** Xây dựng trang web “Đăng ký học phần”
   1. Chức năng 1:
      * **Yêu cầu:** Khi một TextBox nhận focus, màu nền của nó sẽ được thay đổi. Khi mất focus, màu nền của TextBox đó trở lại bình thường. Phải có ToolTip cho từng TextBox.
      * **Cách giải quyết:** thêm sự kiện onfocus và onblur vào các TextBox bằng câu lệnh sau: onfocus="FocusEvent(this.id)" onblur="LostFocusEvent(this.id)". Trong đó hàm FocusEvent, LostFocusEvent (file Events.js), hai hàm này làm thay đổi màu nền của TextBox. Thêm thuộc tính title cho các textbox để có ToolTip.
   2. Chức năng 2:
      * **Yêu cầu:** Kiểm tra tính hợp lệ của dữ liệu khi người dung bấm vào nút Đăng ký.
      * **Cách giải quyết:** hàm kiểm tra lỗi được viết trong hàm CheckError trong file Functions.js. Hàm này sẽ gọi:
        1. Hàm Trim (file \_String.js) cắt khoảng trống ở 2 đầu của chuỗi rồi mới kiểm tra xem chuỗi rỗng hay không, áp dụng cho chuỗi strID, strName, strAddress.
        2. Hàm IsPhoneNumber (file Functions.js) để kiểm tra chuỗi strPhone có phải là số điện thoại hay không.
        3. Hàm CheckDate (file \_Date.js) để kiểm tra chuỗi ngày tháng có hợp lệ hay không.
        4. Hàm IsGoogleorYahooMail (file Functions.js) để kiểm tra chuỗi strEmail có phải là mail của google hay yahoo hay không.

Nếu tất cả đều hợp lệ thì hàm CheckError sẽ trả về chuỗi rỗng cho biết kết quả hợp lệ, ngược lại sẽ trả về một chuỗi mô tả chi tiết các lỗi phát sinh.

* 1. Chức năng 3:
     + **Yêu cầu:** Thiết kế giao diện chọn môn học như trên, xử lý 4 nút button. Khi nhấn button OK, hiển thị MessageBox chứa danh sách tên các môn học được chọn.
     + **Cách giải quyết:** 
       1. Trong hàm Load (file Events.js) (là hàm onload của document) gọi hàm lstSubjects\_Init (file Events.js), là hàm nạp các môn học vào lstSubjects, các môn học được lấy từ file Subjects.xml, việc đọc nội dung file xml do hàm ReadXMLFile (file Functions.js) thực hiện, trong hàm ReadXMLFile sẽ gọi hàm LoadXMLDoc (file \_XML.js) để nạp file xml lên bộ nhớ.
       2. Khi nhấp vào  thì gọi hàm MoveItem(lstSubjects, lstSelectedSubjects) (file Events.js) để di chuyển các item được chọn từ lstSubjects sang lstSelectedSubjects. lstSubjects là listbox bên trái, lstSelectedSubjects là listbox bên phải.
       3. Khi nhấp vào  thì gọi hàm MoveItem (lstSelectedSubjects, lstSubjects) (file Events.js) để di chuyển các item được chọn từ lstSelectedSubjects sang lstSubjects.
       4. Khi nhấp vào  gọi hàm MoveItemAll (lstSubjects, lstSelectedSubjects) (file Events.js) để di chuyển toàn bộ item từ lstSubjects sang lstSelectedSubjects. Hàm MoveItemAll thực hiện chọn tất cả các item rồi gọi hàm MoveItem.
       5. Khi nhấp vào  gọi hàm MoveItemAll (lstSelectedSubjects, lstSubjects) để di chuyển toàn bộ item từ lstSelectedSubjects sang lstSubjects.
       6. Khi nhấp vào gọi hàm butOK\_Click (file Events.js), lấy text của tất cả các item ở lstSelectedSubjects rồi xuất ra thông báo.
  2. Chức năng 4:
     + **Yêu cầu:** Bấm đăng ký, nếu dữ liệu hợp lệ sẽ được thêm vào danh sách sinh viên bên dưới
     + **Cách giải quyết:** khi nhấp vào  gọi hàm butRegister\_Click (file Events.js), hàm này gọi hàm CheckError để kiểm tra lỗi, nếu không có lỗi thì thêm một dòng mới vào table thông qua hàm InsertRow (file Functions.js), và xóa các ô đã nhập, để chuẩn bị nhập dữ liệu mới. Ngược lại, xuất thông báo lỗi.
  3. Chức năng 5:
     + **Yêu cầu:** Khi người dùng di chuyển chuột đến dòng dữ liệu nào, dòng đó sẽ được highlight. Khi người dùng nhấn vào Header ở cột nào thì dữ liệu sẽ được sort theo cột đó(một lần tăng rồi đến một lần giảm).
     + **Cách giải quyết:** 
       1. Trong hàm InsertRow, khi thêm một hàng, thêm sự kiện onmouseover và onmouseout cho hàng vừa thêm bằng câu lệnh: x.onmouseover = function() {FocusEvent(x.id);}; x.onmouseout = function() {LostFocusEvent(x.id);}; để thay đổi màu nền của các hàng khi có sự kiện mouseout và mouseover.
       2. Xây dựng hàm SortTable (file Functions.js) để sắp xếp một cột trong table, cách thực hiện của hàm là: nạp tất cả các dòng của table (trừ dòng đầu tiên) vào một mảng, mỗi phần tử của mảng là một mảng chứa dữ liệu của các cột trong dòng tương ứng. Sắp xếp mảng đó theo cột ColIndex (tham số hàm), nếu cột đó là kiểu Date thì sắp xếp theo kiểu Date, sắp xếp theo thuật toán nổi bọt. Sau khi sắp xếp xong sẽ nạp tất cả dữ liệu từ mảng trở về table. Nếu hàm này được gọi nhiều hơn 1 lần thì một lần tăng và một lần giảm, dựa vào biến Col[ColIndex] (Col là mảng kiểu boolean, khai báo ở file DangKyHocPhan.html), Col[ColIndex] == true thì sắp xếp tăng, ngược lại sắp xếp giảm. Thêm vào sự kiện onclick cho các cột nằm ở dòng đầu của table, onclick = SortTable.
  4. Để chỉ chọn một checkbox thì thêm sự kiện onclick cho các checkbox bằng lệnh okônclick="chkSex\_Click(this.id, true)", chkSex\_Click (file Events.js) đánh dấu checkbox được chọn, bỏ check đối với checkbox còn lại.
  5. Ứng dụng chạy tốt trên IE 6.0 và FireFox, những trình duyệt khác chưa kiểm tra.

1. **Bài tập 2:** Tạo trang web “Trình diễn hình ảnh”
   1. Chức năng 1:
      * **Yêu cầu:** Có hệ thống menu chiều ngang, các menuitem giống như của Notepad. Khi click vào menu item thì xuất ra câu thông báo cho biết đã chọn menu item nào**.**
      * **Cách giải quyết:**
        1. Dùng kết hợp với CSS để tạo menu chiều ngang (Phần này tham khảo cách dùng css tại trang http://www.cssplay.co.uk/menus/drop\_definition.html).
        2. Thêm sự kiện onclick vào các menuitem (thực chất là các thẻ <dd>) bằng lệnh onclick="MenuItem\_Click(this.id)". MenuItem\_Click (file Events.js) sẽ lấy element tương ứng rồi lấy chuỗi trong element đó xuất ra màn hình.
   2. Chức năng 2:
      * **Yêu cầu:** Khi bấm vào menu File – New sẽ tạo ra cửa sổ trên trang web (tham khảo meebo.com) (có thanh tiêu đề với nút Minimize, Maximize, Close), bấm vào nút Close thì cửa sổ đóng lại**.**
      * **Cách giải quyết:**
        1. Xây dựng lớp \_Window (file \_Window.js) để tạo ra window, và các hàm liên quan đến window như: Close, Minimize, Maximize, … Các bước tạo ra một window là: tạo một thẻ div chính có name = window (Thêm thẻ này vào body), tạo 3 thẻ div nữa với tên lần lượt là title, client, status, 3 thẻ mới này được thêm vào window thông qua hàm appendChild. Trong title có thêm 3 thẻ img, tượng trưng cho nút Close, Minimize, Maximize. 3 thẻ img này có thêm sự kiện onclick để xử lý các sự kiện tương ứng. Trong client có thêm 1 thẻ img để hiện thị ảnh cứ sau 5s là thay đổi ảnh. Trong status có thêm 1 thẻ div có name=”handle” nằm ở phía bên phải status, là nơi để thay đổi kích thước window.
        2. Xây dựng lớp \_ListWindow (file \_ListWindow.js) để quản lý các window được tạo ra từ lớp \_Window. Gồm 2 hàm: AddWindow để thêm một window, Get để lấy đối tượng \_Window thông qua tham số id.
        3. Thêm sự kiện onclick vào menuitem New bằng lệnh: onclick="New\_Click()". New\_Click (file Events.js) gọi hàm AddWindow của lớp \_ListWindow để tạo một window mới. Trong hàm AddWindow sẽ tạo mới một đối tượng thuộc \_Window.
        4. Khi nhấp vào nút Close (thực chất là một image) thì gọi hàm Close (file \_Window.js), hàm này xóa thẻ div có name=”window” bằng hàm removeChild.
   3. Chức năng 3:
      * **Yêu cầu:** Window vừa tạo ra có khả năng di chuyển (kéo thả), thay đổi kích thước, xử lý được nút Minimize, Maximize.
      * **Cách giải quyết:**
2. Để di chuyển window, thêm 2 sự kiện onmouseup và onmousedown cho document thông qua lệnh document.onmouseup = MouseUp; document.onmousedown = MouseDown; Hàm MouseDown (file Events.js) ghi nhận xem đối tượng nào xảy ra sự kiện nhấp chuột, nếu đó là đối tượng dùng để di chuyển window (ở đây là thẻ div có name = “title”) thì ghi nhận có sự kiện di chuyển window và kích hoạt sự kiện onmousemove của document thông qua lệnh document.onmousemove = DrapDrop; Hàm DrapDrop (file Functions.js) sẽ kiểm tra xem có phải đang thực hiện việc di chuyển window hay không thông qua biến obj và kiểm tra xem window có phải đang ở trạng thái “normal” hay không thông qua biến mCurrentState của đối tượng \_Window, nếu đúng thì thay đổi vị trí của window thông qua hàm MoveWindow (file \_Window.js), hàm này cập nhật lại thuộc tính top, left của thẻ div có name=”window”. Hàm MouseUp (file Events.js) ghi nhận rằng không có sự kiện di chuyển window.
3. Để thay đổi kích thước window, cách làm tương tự như cách làm di chuyển window, chỉ khác là khi có sự nhấp chuột và đối tượng được nhấp là thẻ div có name=”handle” thì ghi nhận có sự kiện thay đổi kích thước window và kích hoạt sự kiện onmousemove của document thông qua lệnh document.onmousemove = ChangeSizeWindow; Hàm ChangeSizeWindow (file Functions.js) sẽ kiểm tra xem có phải đang thực hiện việc thay đổi kích thước window hay không thông qua biến obj và kiểm tra xem window có phải đang ở trạng thái “normal” hay không thông qua biến mCurrentState của đối tượng \_Window, nếu đúng thì thay đổi kích thước của window thông qua hàm ChangeSize (file \_Window.js), hàm này cập nhật lại thuộc tính width, height của thẻ div có name=”window”.
4. Khi nhấp vào nút Minimize (thực chất là một image) thì gọi hàm Minimize (file \_Window.js), nếu trạng thái hiện tại của window không phải là minimize thì hàm này sẽ gọi hàm MoveWindow, ChangeSize (file \_Window.js) để di chuyển window về vị trí dành cho các window đang ở trạng thái minimize (top: 500px, left: chèn vào vị trí chưa có window nào minimize thông qua biến Contain) và thu nhỏ kích thước của window (height: 20px, width:100px). Còn nếu trạng thái hiện tại của nó là minimize thì trả lại trạng thái trước đó của window, nếu trạng thái trước đó là maximize thì cho window là maximize, nếu normal thì cho window là normal.
5. Khi nhấp vào nút Maximize (thực chất là một image) thì gọi hàm Maximize (file \_Window.js), nếu trạng thái hiện tại của window không phải là maximize thì hàm này sẽ gọi hàm MoveWindow, ChangeSize (file \_Window.js) để di chuyển window về vị trí dành cho các window ở trạng thái maximize (left: 1px, top: 1px) và phóng to kích thước của window (height: 600px, width: 800px). Còn nếu trạng thái hiện tại của nó là maximize thì trả lại trạng thái normal cho window.
   1. Chức năng 4:
      * **Yêu cầu:** Trên mỗi cửa sổ sẽ chứa 1 hình ảnh, khoảng 5s sẽ được thay đổi hiển thị một hình mới.
      * **Cách giải quyết:** dùng hàm setTimeout để cứ 5s thì thay đổi hình ảnh, và hàm này được đặt trong hàm ChangeImage (file Functions.js), hàm này sẽ lấy tên của hình ảnh tiếp theo trong ListImage để load hình đó lên thẻ img có name=”image”, biến mIndex cho biết hình hiện tại trong ListImage có số thứ tự là bao nhiêu. Tên của các hình được lấy từ file Images.xml, được nạp lên bởi hàm ListImage\_Init (file \_Window.js), hàm này sẽ nạp file xml lên bộ nhớ thông qua hàm ReadXMLFile (file \_XML.js) rồi đọc nội dung rồi đưa vào biến ListItem.
   2. Chức năng 5:
      * **Yêu cầu:** Khi click phải vào vị trí bất kỳ sẽ xuất hiện một ContextMenu(các menuitem do SV tự đặt ra). Khi click vào menu item thì xuất ra câu thông báo cho biết đã chọn menu item nào.
      * **Cách giải quyết:** 
        1. dùng CSS để tạo menupopup, thêm sự kiện onclick vào các menu item để khi nhấp vào thì hiện thông báo, được xử lý trong hàm pMenuItem\_Click (file Events.js),thêm sự kiện oncontextmenu cho document bằng lệnh document.oncontextmenu = ShowMenu; ShowMenu (file Functions.js) để hiển thị menupopup tại vị trí nhấp chuột phải, để hiển thị dùng lệnh p.style.display = "block"; trong đó p là thẻ div.
        2. Để khi nhấp chuột trái ra khỏi menupopup thì tắt menupopup thì trong hàm MouseDown (file Events.js) (là hàm xử lý onmousedown của document) có gọi HideMenu (file Functions.js) để ẩn menupopup, bằng lệnh p.style.display = "none"; p là thẻ div.
   3. Chưa xử lý được chuyện window tạo sau luôn nằm trên window tạo trước.
   4. Ứng dụng chạy tốt trên IE 6.0 và FireFox, những trình duyệt khác chưa kiểm tra.
6. **Tham khảo:** 
   1. Tham khảo cách gán sự kiện bằng hàm có chứa tham số tại trang <http://p2p.wrox.com/topic.asp?TOPIC_ID=20989>
   2. Tham khảo cách nạp file xml lên bộ nhớ tại <http://www.w3schools.com/dom/dom_parser.asp>
   3. Tham khảo cách drap & drop tại <http://javascript.internet.com/page-details/drag-n-drop.html>