



My Env

(My Environment)

By: Joshua Daniel

Project Advisor: Alex Liu



Overview

MyEnv (My Environment) is a virtual desktop environment in which users can create multiple browser windows to browse the internet, create bookmarks and save their session so that they may retrieve it at a later time



Motivation

The motivation of this project is to:

- Learn how much can be accomplished by using JavaScript/Ajax to provide a desktop-like environment
- Find out what the limitations of web browsing will be when using such an environment



Objectives

The objectives of this master's project are:

- Provide a virtual desktop environment in which users have the ability to save their session
- Design the system using technologies available to the modern browser (i.e., DHTML/AJAX)
- Make the system portable so that it may be moved/copied to other servers (windows, mac, linux)



Requirements

- User must be able to create a browser desktop environment within a single webpage - multiple browsers on a single webpage
- User must be able to open multiple web browsers and position them according to personal preference
- User may end their session at any time and log in later with their browser environment still in intact



Result

The result will be a mobile environment that a user may access from any modern JavaScript enabled browser





Related Works

Yahoo! User Interface Library:

- The interface, more commonly known as YUI, is an API based on a set of JavaScript controls built to provide developers a tool to create a rich web environment
- Abstracts JavaScript - The developer does not need to worry about recording mouse position based on event or manually inserting or removing objects from the HTML DOM
- It provides the end user with a website that feels more like a desktop environment rather than a traditional web application



Related Works (continued)

YUI Controls

Some examples of YUI controls are:

- Modal dialogs
- Document Object Model (DOM) utilities
- Movable onscreen windows
- JavaScript Event Handlers
- YUI Logger (used in MyEnv project)

<http://developer.yahoo.com/yui/>



Related Works (continued)

YouOS:

- Web-based operating system
- Written in JavaScript
- Includes word processor, spreadsheet, **Web Browser**, etc.
- <http://www.youos.com/>

Why is YouOS important to the MyEnv experiment?

- It includes a single web browser composed of an iframe
- It does not support multiple windows or book marking of URLs, Why?
- JavaScript cross domain security prevents YouOS from accessing the url it's iframe is pointing to (*More on this later*)



Technologies used in MyEnv

- JavaScript/DHTML
- Python
- AJAX/JSON



JavaScript and DHTML

- DHTML stands for Dynamic HTML
- Modern JavaScript has the ability to access the HTML DOM (document object model) and manipulate it in real time
- The DOM is accessed client side, which means that it is altered on the user's browser not the server



Python

- Scripting Language
- Used in industry (Google for application testing)
- Freely downloadable packages and APIs:
 - Webpy (web server)
 - Beautiful Soup (HTML DOM manipulation)
- Portable across OS - MyEnv is currently installed on this Mac and a remote Windows machine
- Alternative server side languages: classic ASP, ASP.net, PHP, Java



Ajax and JSON

- “X” in Ajax stands for XML
- Ajax communication traditionally handled by passing XML back and forth between client side JavaScript and server side script
- XML has to be compiled and parsed on each side
- JSON is a JavaScript object represented by a string
- XML easier to read, JSON easier to develop
- Both JavaScript and Python can parse JSON string into an object so MyEnv uses JSON for communication



Implementation

User Profile:

- Each user logging into the MyEnv system will require a user profile. This will allow for the system to save the user’s information in the database.
- Currently active browser windows, x/y position, and URL
- Saved bookmark



Implementation (continued)

Coordinate System Using CSS:

- CSS stands for cascading style sheet
- CSS attributes can be modified by JavaScript

Example:

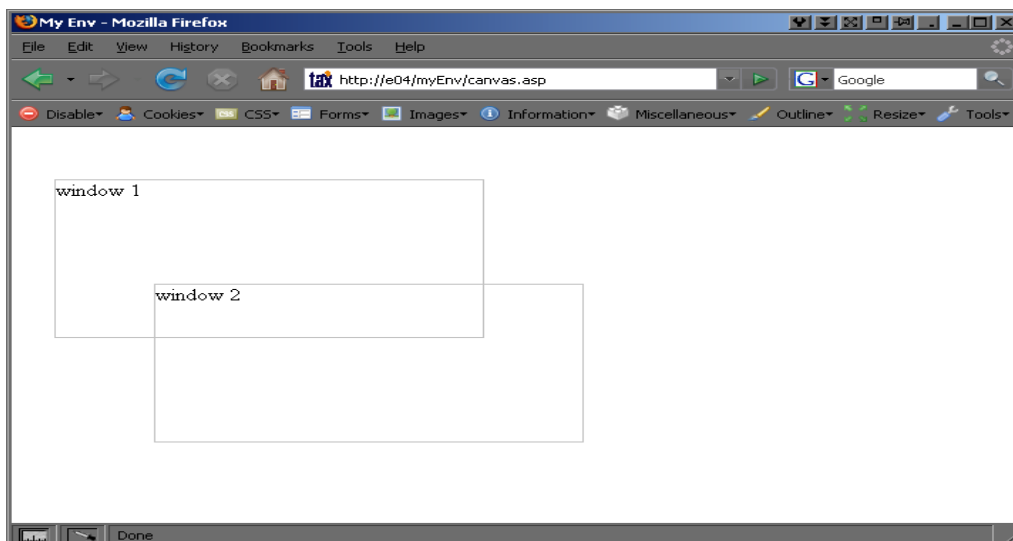
```
<div class="canvas" style="width:100%;height:100%">
  <div class="browser"
    style="left:30px;top:50px;width:300px;height:150px">window
  1</div>
  <div class="browser"
    style="left:100px;top:150px;width:300px;height:150px">window
  2</div>
</div>
```



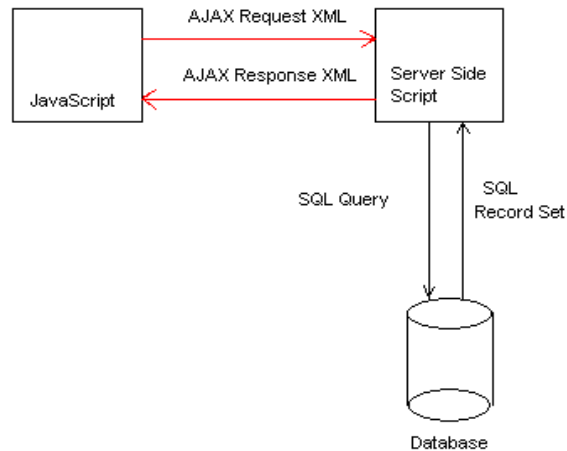
Implementation (continued)

Coordinate System Using CSS:

Example:



System Diagram



Ajax Request/Response Examples

- Add a bookmark
- Add a video bookmark
- Retrieve bookmark URL
- Retrieve video bookmark embed code
- Save browser location
- Retrieve last environment



Add a Bookmark Example

Request:

```
<myENV>
  <request>
    <addBookmark>
      <userID> userID </userID>
      <title> bookmark Title </title>
      <url> bookmark URL </url>
    </addBookmark>
  </request>
</myENV>
```



Add a Bookmark (continued)

Client side JavaScript XML Request:

```
<myENV>
  <request>
    <addBookmark>
      <userID> userID </userID>
      <title> bookmark Title </title>
      <url> bookmark URL </url>
    </addBookmark>
  </request>
</myENV>
```

SQL translation:

insert into bookmarks (userid, title, url) values (*userID*, *bookmark Title*, *URL*)



Add a Bookmark (continued)

Response:

```
<myENV>
  <response>
    <addBookmark>
      <bookmarkID> bookmarkID </bookmarkID>
      <title> bookmarkTitle </title>
    </addBookmark>
  </response>
</myEnv>
```



Project Challenges

JavaScript Security

- One of the major problems of this experiment
- JavaScript will not allow us to reference iframe URLs
- 3 ways to reference however, JavaScript will throw an exception if the URL is cross domain:
 - `frames['myiframe'].location.href`
 - `frames['myiframe'].location`
 - `frames['myiframe'].src`

Error: Permission denied to get property Location.href
Source File: <http://0.0.0.0:8080/static/test.html>

Error: Permission denied to call method Location.toString



JavaScript Security

- Since the system must be able to emulate a desktop of multiple browsers, it will need to know the current URLs of all browsers
- It must also know and record the current location and previous location if they exist (browser history)
- We cannot rely on the back and forward buttons of the parent window since this will change state for all browser windows



JavaScript Security (continued)

Why does JavaScript throw an exception?

- It is a privacy issue
- An unauthorized website could potentially track your clicks or steal your information without your knowledge
- They would only have to create an iframe their site which points to a different site (online banking). They can then have access to your information, via URL variables.



JavaScript Security (continued)

- What would cause an exception?
 - Attempting to access the URL of an iframe pointing to <http://www.amazon.com/> when the parent window is pointing to <http://www.myenv.com>
- What would not cause an exception?
 - Attempting to access the URL of an iframe pointing to <http://www.myenv.com/subpage.html> when parent window is pointing to <http://www.myenv.com>



JavaScript Security Solution

Hyperlink Injection

- The solution would be to make every link in the iframe point to our local domain (in our example www.myenv.com)
- This can be done by pre-pending a relative URL to the beginning of any clickable link on the page
- Example:
 - <http://www.amazon.com/page1.html>
 - `/pagegetter/http://www.amazon.com/page1.html`
 - `/pagegetter.php?url=http://www.amazon.com/page1.html`
- PageGetter is a script which takes a single argument that is the URL to retrieve



Hyperlink Injection

By using a server side script (pagegetter) to intercept the user's click we can load the HTML on our local server and modify tags on the page.

Tags that must be modified:

- Image
- CSS include <link>
- Anchor Tag <a>
- JavaScript Include <script>
- Form Tag <form>



Tag Modification

- A Python Package called Beautiful Soup will be used to read the HTML text into a DOM object in order to loop through the tags and modify them
- Regular Expression could have also be used but would have been more difficult to read, maintain and make changes to

Example of anchor Tag modification:

for a in soup.findAll('a'):

try:

if a['href'].startswith("http://") or a['href'].startswith("https://"):

a['href'] = "/pagegetter/" + a['href']

else:

addSlash = "/"

if a['href'].startswith("/"):

addSlash = ""

a['href'] = "/pagegetter/" + scheme + "://" + domain + addSlash + a['href']

except KeyError, e:



Tag Modification (continued)

```
for a in soup.findAll('a'):
    try:
        if a['href'].startswith("http://") or a['href'].startswith("https://"):
            a['href'] = "/pagegetter/" + a['href']
        else:
            addSlash = "/"
            if a['href'].startswith("/"):
                addSlash = ""
            a['href'] = "/pagegetter/" + scheme + "://" + domain + addSlash + a['href']
    except KeyError, e:
```



Browser History Problem

- Back and forward button functionality
- Iframe URL access solved by Hyperlink Injection
- Each browser will have to have its own browser history
- How to store this object
 - JavaScript Browser History Object (doubly linked list)



Conclusion

- Able to mimic a Desktop type environment in JavaScript
- Able to work around the JavaScript cross domain violation
- Where would MyEnv be useful? - School Environment Library
 - Controlled Environment
- Reliability and Stability of JavaScript.
 - New Browser releases
 - Browser Support



Demo



Any Questions?