

فاز سوم تحقیق درس مهندسی اینترنت

پروتکل‌ها و تکنولوژی‌های مورد استفاده در وب (نسخه دوم)

استاد درس :

جناب آقای دکتر حقیقی

ارائه شده به :

آقای نادری

ارائه شده توسط:

فاطمه دشتی - ۸۵۲۱۳۰۵۹

گروه آموزشی کامپیوتر

۹۰ خرداد

فهرست :

۳(Cascading Style Sheets) CSS	۱
۱۰HTML	۲
۱۳JavaScript	۳
۱۸Ajax	۴
۲۳(Active Server Page) ASP	۵
۲۵Web Browser	۶
۲۶PHP	۷
۲۸MySQL	۸

۱. Cascading Style Sheets یا برگه های سبک آبشاری

Cascading Style Sheet یا برگه های سبک آبشاری روشی ساده برای قالب بندی و

جلوه های تصویری (مانند نوع قلم، رنگ و اندازه ها) بر صفحه های وب اند. الگوهای آبشاری از جنس زبان های نشانه گذاری یا markup، با ساختار متن ساده هستند و درون هر کدام، دستورهای آبشارمانند و پی در پی، برای چگونگی نمایش هر صفحه وب افزوده می شود. به بیانی ساده تر، این دستورها روش نشان داده شدن قلم ها و اندازه شان، رنگ ها و پس زمینه ها، روش چیدمان موزاییک های دربرگیرنده داده ها (دیواره ها)، و بسیاری دیگر از عنصرهای ساختار هر صفحه وب را، درون خود جای می دهند. کاربرد معمول CSS چیدمان صفحه های وب نوشته شده در HTML و XHTML است ولی این زبان می تواند برای هر XML document از جمله SVG، plain XML و XUL به کار برود. می توان آن را در فایل جداگانه نوشت و به HTML افزود یا با استفاده از برچسب Style در داخل فایل HTML از آن استفاده نمود.

در بحث طراحی با استانداردهای وب، همه ساختار گرافیکی و تصویری صفحه های وب باید درون این الگوهای آبشاری قرار گیرد و لابلای صفحه های وب نباید دستورهایی از این جنس، برای درج چگونگی نمایش صفحه ها نوشته شود. به بیانی دیگر، ساختار سند و متن در هر صفحه، باید پاکیزه باشد و دستورهای چگونگی نمایش و زیبایی، همگی در الگوهای آبشاری افزوده شوند.

با اینکه در سال های نخست پیدایش طراحی وب، طراح ها از جدول ها و ترفند پنهان سازی آنها

برای چیدمان کمک می گرفتند، روش مدرن طراحی وب، این روند را منسوخ کرده است. چیدمان اکنون، تنها با دیواره (یا DIV)، که برتری های فراوانی نسبت به جدول ها (یا TABLE) دارند، انجام می شود. هر دیواره، یک موزاییک مجازی بخش کننده صفحه است که کار چیدمان را آسان تر انجام

می‌دهد. جدول‌ها به جایگاه نمایش داده‌های ستونی خود بازگشتند و دیگر از آنها برای چیدن داده‌های صفحه‌های وب استفاده نمی‌شود.

CSS برای جدا کردن محتوای document (نوشته شده در HTML یا هر زبان markup دیگر) از قسمت presentation آن از جمله layout، رنگ و فونت، طراحی شد. این جداسازی می‌تواند دسترس پذیری محتوا را بهبود بخشد، انعطاف پذیری و کنترل بیشتری در جزئیات presentation ایجاد کند، امکان استفاده چند صفحه از یک formatting و کاهش پیچیدگی و تکرار در ساختار محتوایی را فراهم می‌کند. CSS همچنین این اجازه را می‌دهد که یک markup page با سبک‌های مختلف برای متدهای مختلف ارائه، مثل پرینت، صدا، Braille و دستگاه‌های لمسی ارائه شود. در حالیکه نویسنده یک document به طور معمول آن را به یک CSS style sheet متصل می‌کند، خواننده می‌تواند از style sheet‌های دیگری برای تغییر چیزی که نویسنده مشخص کرده، استفاده کند.

الگوهای آبخاری، افزون بر سبک‌تر کردن هر صفحه وب و پاک‌تر کردن آن از داده‌های تصویری و نمایشی بی‌استاندارد، روش طراحی وب را به میزان فراوانی آسان‌تر، و دسترسایی و کارایی وب را بهتر کرده‌اند.

یک الگوی آبخاری معمولاً در حافظه پنهان مرورگر ذخیره می‌شود، بنابراین می‌تواند بدون نیاز به بازخوانی در چندین صفحه استفاده شود که این امر باعث کاهش تبادل اطلاعات در شبکه می‌شود.

برخی قابلیت‌های CSS به شرح زیر است:

سلکتور Class

با استفاده از کلاس می توانیم برای یک تگ HTML استایل‌های متفاوتی تعریف کنیم. برای نمونه فرض کنیم می خواهیم در صفحه دو نوع پاراگراف داشته باشیم: یکی با متن مشکی و دیگری با متن قرمز. برای مشخص کردن این دو نوع پاراگراف می توانیم از سلکتور کلاس استفاده کنیم. ابتدا استایل را به صورت زیر می نویسیم:

```
p.black{color: black}  
p.red{color: red}
```

در مرحله بعد باید برای هر پاراگرافی که می خواهیم به رنگ مشکی باشد از شناسه

`class="black"` در تگ `<p>` استفاده کنیم و برای پاراگراف قرمز از شناسه `class="red"` نام

کلاسها اختیاری است ولی بهتر است از نام دارای مفهوم استفاده کرد تا کد خوانایی داشته باشد. کد

مربوط به نوشتن دو پاراگراف بالا را در متن HTML :

```
<p class="black">این متنی است که به رنگ مشکی نمایش داده می شود</p>
```

```
<p class="red">این متن به رنگ قرمز نمایش داده می شود</p>
```

می توان بیش از یک کلاس برای یک تگ HTML تعریف کرد اما برای این کار نباید دو بار از

شناسه `class` استفاده کرد بلکه باید در یک شناسه `class` نام دو کلاس مورد نظر را با یک فاصله بین

آنها وارد کرد. برای مثال برای پاراگراف زیر از دو کلاس فرضی `red` و `center` استفاده شده است:

```
<p class="red center">این متن پاراگرافی است که تحت تأثیر دو کلاس red و center قرار
```

```
دارد</p>
```

شناسه style

ساده ترین راه برای افزودن استایل به یک صفحه استفاده از شناسه `style` در تگهای HTML

است. برای افزودن استایل به یک تگ باید عبارت `style=" "` را وارد تگ مربوطه کنیم و ویژگیهای

مورد نیاز خود را بین (" ") وارد کنیم. مثلاً اگر بخواهیم به یک تگ DIV استایل اضافه کنیم به صورت زیر در می آید:

```
<div style="property: value"
```

این امکان نیز وجود دارد که به یک تگ HTML بیش از یک ویژگی افزوده شود. تنها کافی است بین ویژگیهای مختلف از یک نقطه ویرگول (;) استفاده کنیم. برای مثال اگر بخواهیم متنی که در یک DIV قرار دارد به رنگ قرمز و به صورت ایتالیک باشد

```
<div style="color: red; font-style: italic">این متن قرمز رنگ و ایتالیک است</div>
```

نتیجه به صورت زیر خواهد بود:

این متن قرمز رنگ و ایتالیک است

تغییر رنگ و تصویر زمینه عناصر HTML با CSS

این ویژگی به ما اجازه می دهد تا تقریباً برای هر چیزی در صفحه رنگ زمینه مشخص کنیم. برای این کار باید این ویژگی را به همراه رنگ مورد نظر وارد تگ مربوط به قسمت مورد نظر کنیم. اگر بخواهیم زمینه قسمتی از یک متن را مشخص کنیم می توانیم از این ویژگی در یک تگ SPAN در اطراف متن مورد نظر استفاده کنیم.

```
<p" background-color: yellow">این متن زمینه زرد دارد</p>
```

نتیجه به این صورت است:

این متن زمینه آبی دارد

این ویژگی برای مشخص کردن تصویر زمینه به کار می رود. کاربرد background-image همانند background-color است با این تفاوت که به جای مشخص کردن رنگ باید یک تصویر را تعیین کنیم. این ویژگی به صورت زیر نوشته می شود.

```
background-image: url(آدرس تصویر مورد نظر)
```

بعد از نام ویژگی باید از URL برای مشخص کردن آدرس تصویر استفاده کنیم. آدرس تصویری که می خواهیم به عنوان زمینه استفاده کنیم نیازی نیست بین علامتهای نقل قول (") قرار گیرد ولی باید بین دو پرانتز قرار گیرد.

Spacing

این ویژگی به ما اجازه می دهد تا فاصله میان حروف یک کلمه را در یک متن تعیین کنیم. این ویژگی می تواند مقادیری را در مقیاس پیکسل به خود بگیرد یا از مقدار normal استفاده کند. گزینه پیش فرض برای این ویژگی همان normal است. در اینجا به ذکر یک مثال در مورد این ویژگی می پردازیم:

```
<div style="letter-spacing: 5px">  
فاصله ۵ پیکسل از یکدیگر نمایش داده خواهند شد حروف این متن با  
</div>
```

نتیجه را به صورت زیر است:

حروف این متن با فاصله ۵ پیکسل از یکدیگر نمایش داده خواهند شد.

Text-align

این ویژگی به ما اجازه می دهد تا نحوه قرارگیری قسمتی از متن را از لحاظ راست چین، چپ چین یا وسط چین بودن تنظیم کنیم.

این ویژگی می تواند مقادیر left, right, center را استفاده کند.

```
</div> <div style="text-align: left; border: 1px dashed #00CCFF">
```

نتیجه به صورت زیر خواهد بود:

محتویات این DIV در سمت چپ صفحه نمایش داده می شوند.

Text-decoration

این ویژگی اجازه می دهد تا برای قسمت خاصی از متن خاصیت‌های از قبیل زیرخط تعریف کنیم. مثلاً اگر بخواهیم یک متن را با خطی بالای آن به وجود بیاوریم می توان از این ویژگی به صورت زیر استفاده کرد:

```
<div style="text-decoration: underline">نمونه!</div>
```

نتیجه به صورت زیر خواهد بود:

نمونه!

این ویژگی مقادیر none، underline، overlain، line-through و blink را می تواند داشته باشد.

Text-indent

این ویژگی مقدار تو رفتگی متن را در سطر اول هر قسمت (مثلاً سطر اول هر پاراگراف) مشخص می کند. این مقدار می تواند بر حسب پیکسل یا در صد باشد.

در مثال زیر این مقدار تو رفتگی را ۳۰ پیکسل در نظر می گیریم:

```
<div style="text-indent: 30px">
```

```
<br>با ۳۰ پیکسل تو رفتگی نمایش داده می شود سطر اول این قسمت
```

```
<br>صورت عادی و در حالی که بقیه سطرها به
```

```
بدون تو رفتگی نمایش داده می شوند
```

```
</div>
```

نتیجه به صورت زیر است:

سطر اول این قسمت با ۳۰ پیکسل تو رفتگی نمایش داده می شود.
در حالی که بقیه سطرها به صورت عادی و
بدون تو رفتگی نمایش داده می شوند.

۲. HTML

HTML (Hyper Text Markup Language) سنگ بنای وب است؛ یک زبان برای

نشانه‌گذاری hypertext است که برای تدوین قالب و طراحی صفحه‌های وب به کار می‌روند.

دستورعمل‌های این زبان، برچسب (Tag) نام دارند که محتوای یک صفحه وب، با آنها نشانه‌گذاری

شده و بدین ترتیب، نحوه نمایش آن صفحه برای مرورگرهای وب، توصیف می‌شود.

یک سند HTML، یک سند مبتنی بر متن (Text-based) است که معمولاً با پسوند .htm

یا .html نام‌گذاری شده و محتویات آن از tagهای HTML تشکیل می‌شوند. هر یک از برچسب‌های

HTML، معنا و مفهوم خاصی دارند و تأثیر مشخصی بر محتوا می‌گذارند؛ مثلاً برچسب‌هایی برای

تغییر شکل ظاهری متن، نظیر درشت و ضخیم کردن یک کلمه یا برقراری پیوند به صفحات دیگر

تعریف شده‌اند. مرورگرهای وب، که قادر به درک و تفسیر برچسب‌های HTML هستند، تک‌تک آن‌ها

را از داخل سند خوانده و سپس محتوای آن صفحه را نمایان‌سازی (Render) می‌کنند.

HTML زبان برنامه‌نویسی نیست، بلکه زبانی برای نشانه‌گذاری ابرمتن است و اساساً برای

ساختنمند کردن اطلاعات، پویاسازی صفحه‌های وب و جداسازی اجزای منطقی یک نوشتار مانند

عناوین، تصاویر، فهرست‌ها، بندها و جداول به کار می‌رود. از سوی دیگر، HTML را نباید به عنوان

زبانی برای صفحه‌آرایی یا نقاشی صفحات وب پنداشت؛ این وظیفه اکنون بر دوش فناوری‌های دیگری

همچون CSS است.

HTML شکلی از زبان دیگری بنام SGML است و کنسرسیوم وب جهان‌شمول آن را به

عنوان استاندارد برای نشانه‌گذاری مستندات ابرمتنی برای عرضه در وب، تدوین کرده‌است.

هر HTML document از دو قسمت head و body تشکیل می‌شود. head شامل اطلاعاتی به صورت کلی مثل title می‌شود. body شامل اطلاعاتی می‌شود که در web browser کاربر نمایش داده می‌شود.

```
<HTML>
```

```
<Head>
```

```
<TITLE> .....</TITLE>
```

```
</Head>
```

```
<BODY>
```

```
...
```

```
...
```

```
</BODY>
```

```
</HTML>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Hello HTML</title>
```

```
  </head>
```

```
  <body>
```

```
    <p>Hello World!</p>
```

```
  </body>
```

```
</html>
```

Elements

HTML document از HTML elements تشکیل شده‌اند و فرم کلی آنها به صورت

```
<tag attribute1="value1" attribute2="value2">content</tag>
```

است.

بعضی از المان‌های HTML، empty، تعریف می‌شوند و بدون محتوا هستند.

عنوان در HTML document در head به صورت زیر تعریف می‌شود:

```
<head>
  <title>The title</title>
</head>
```

Heading ها با tag های <h1> تا <h6> تعریف می شوند:

```
<h1>Heading1</h1>
<h2>Heading2</h2>
<h3>Heading3</h3>
<h4>Heading4</h4>
<h5>Heading5</h5>
<h6>Heading6</h6>
```

:Attributes

Attribute ها صفات tag مربوطه را مشخص و توصیف می کنند. به طور مثال:

```
<font color="blue">...</font>
```

```
<a href=http://www.rpi.edu>...</a>
```

پاراگراف:

```
<p>Paragraph 1</p>
```

کامنت:

```
<!-- this is a comment -->
```

۳. JavaScript

JavaScript زبانی است که توسط NetScape تولید شده است و زبان برنامه نویسی اسکریپت مبتنی بر اشیاء است. علیرغم اشتباه عمومی، زبان جاوا اسکریپت با زبان جاوا ارتباطی ندارد، اگر چه ساختار این زبان به C++ و جاوا شباهت دارد که این امر برای سهولت یادگیری در نظر گرفته شده است. از همین رو دستورهای متداول مانند *if*، *catch*، *for* و ... در این زبان هم یافت می شوند. این زبان می تواند هم به صورت هم به صورت شیء گرا (object-oriented) و هم ساخت یافته (structural) مورد استفاده قرار گیرد. در این زبان اشیاء با اضافه شدن متدها و خصوصیات پویا به اشیای خالی ساخته می شوند. بعد از ساخته شدن یک شیء به روش فوق، این شیء می تواند به عنوان نمونه ای برای ساخته شدن اشیاء مشابه مورد استفاده قرار گیرد.

کاربرد گسترده این زبان در سایتها و صفحات اینترنتی می باشد و به کمک این زبان می توان به اشیاء داخل صفحات HTML دسترسی پیدا کرد و آنها را تغییر داد. به همین علت برای پویا نمایی در سمت کاربر (client-side)، از این زبان استفاده می شود.

امنیت : JavaScript و DOM به طور بالقوه امکان اجرای اسکریپت های نوشته شده توسط افراد با سوءنیت را در کامپیوتر کاربر به وسیله وب فراهم می کند. نویسندگان مرورگرها از دو محدودیت برای مقابله با این ریسک استفاده می کنند. اول اینکه اسکریپت ها در یک sandbox اجرا می شوند که تنها می توانند اعمال مرتبط با وب را اجرا کنند و نه کارهای همه منظوره مثل ایجاد فایل. دوم اینکه اسکریپت یک سایت به اطلاعاتی مانند password، username یا cookie های فرستاده شده به یک سایت دیگر دسترسی ندارد. بیشتر باگ های امنیتی مرتبط با JavaScript بر اثر نقض یکی از این دو مورد است.

ساخت یافته و امری

JavaScript syntax برنامه نویسی ساخت یافته در C مثل if، حلقه while، switch و ... را پشتیبانی می کند و استثنای جزئی آن این است که قسمت بندی در سطح بلوک (block-level) به سبک C را پشتیبانی نمی کند و در عوض از قسمت بندی در سطح تابع (function-level) استفاده می کند.

پویایی

تایپ پویا: مانند بیشتر زبان های اسکریپت نویسی typeها مرتبط با مقادیر هستند و نه متغیرها. برای مثال متغیر X می تواند عدد باشد و بعد به string تغییر یابد. JavaScript انواع مختلف تست کردن نوع یک شیء را مانند duck typing پشتیبانی می کند.

مبنتی بر شیء: JavaScript تقریباً به طور کامل مبتنی بر شیء (object-based) است. اشیاء آرایه های انجمنی تقویت شده با نمونه های اصلی (prototypes) هستند. نام تعلق اشیاء، رشته های کلیدی هستند: $obj.x = 10$ و $obj["x"] = 10$ مساوی هستند. خواص (properties) و مقادیرشان می توانند در زمان run-time تغییر، اضافه یا حذف شوند.

ارزیابی زمان اجرا: JavaScript شامل تابع "eval" هست که می تواند statement های رشته ای را در زمان run-time اجرا کند.

تابعی بودن

توابع **first-class** : توابع خودشان **object** هستند و در نتیجه دارای متد و **properties** مثل **length** و **call()** هستند و می توانند به متغیرها داده شوند یا به عنوان آرگومان استفاده شوند و مثل هر **object** دیگری تغییر داده شوند.

توابع تودرتو(داخلی) و بستارها : توابعی هستند که توسط دیگر توابع تعریف می شوند و هر بار تابع اصلی صدا شود ایجاد می شوند متغیرهای توابع بیرونی تا زمانی که تابع داخلی وجود داشته باشد، وجود خواهند داشت، حتی پس از اتمام آن فراخوانی از تابع بیرونی.

مبتنی بر prototype

نمونه ها یا **prototypes** به جای کلاس ها برای ارث دهی استفاده می شوند. امکان شبیه سازی ویژگی های مبتنی بر کلاس با **prototype** ها در **JavaScript** وجود دارد.

توابع به عنوان سازنده ها (constructor)

استفاده از **new** قبل از صدا کردن تابع، تابع جدیدی را ایجاد می کند و آن تابع به همراه دارای کلمه کلیدی **this** را صدا می کند. خواص **prototype** تابع سازنده، شیء مورد استفاده برای **prototype** داخلی شیء جدید را مشخص می کند.

توابع به عنوان متدها

بر خلاف بسیاری از زبان های **object-oriented** تفاوتی میان تعریف یک تابع و تعریف یک متد وجود ندارد و تفاوت زمان صدا کردن تابع اتفاق می افتد؛ یک تابع می تواند به عنوان یه متد صدا

زده شود. وقتی تابعی به عنوان متد یک شیء صدا زده می شود، **this** محلی تابع به آن شیء و برای صدا کردن آن، منتسب می شود.

چند برنامه ساده به جاوا اسکریپت

برنامه‌های ورودی و خروجی در مرورگر وب:

```
<script type="text/javascript">
  alert ("Hello world");
  var x = prompt("Enter Name:", "Default name");
  alert ( "Hello, " + x )
</script>
```

برای استفاده ساده از عناصر HTML و DOM:

```
<script type="text/javascript">
  document.write("Hello world");
  function handle() {
    document.write("You entered: " +
document.getElementById("myinput").value );
  }
</script>

<form>
<input type="text" id="myinput"/>
<input type="button" onclick="handle()" value="Click here" />
</form>
```

برنامه بازگشتی (فاکتوریل):

```
function factorial(n) {
  if (n == 0) {
    return 1;
  }
  return n * factorial(n - 1);
}
```

برنامه swap:

```
<script type="text/javascript">
  var a = 10, b = 40;
  [a,b] = [b,a]
  alert("" + a + " " + b);
</script>
```

```
<script type="text/javascript">
var myarray = [1,5,8,2,3,0,2,9,11];
var i = 0;
var j = 0;
for (i = 0; i < myarray.length ; ++i )
{
    for (j = 0; j < myarray.length; ++j)
    {
        if (myarray[i] < myarray[j])
        {
            var tmp = myarray[i];
            myarray[i] = myarray[j];
            myarray[j] = tmp;
        }
    }
}

for (i = 0; i < myarray.length; ++i)
{
    document.write (myarray[i] + " ");
}
</script>
```

۴ Ajax

Ajax (ایجکس) از ابتدای نام Asynchronous JavaScript (جاوااسکریپت غیر همگام) و XML می آید و به گروهی از متدهای مرتبط با web development، مورد استفاده در طرف کاربر، برای ایجاد application های تعاملی، گفته می شود.

به وسیله Ajax برنامه های تحت وب می توانند به صورت آسنکرون یا غیرهمزمان در پس زمینه و بدون ایجاد مزاحمت برای نمایش (display) یا رفتار و وضعیت صفحه موجود، برای server داده فرستاده یا از آن بازیابی کنند. داده ها معمولاً به وسیله شیء XMLHttpRequest دریافت می شوند. بر خلاف اسم آن نیازی به استفاده از XML نیست و غالباً به جای آن از JSON استفاده می شود و نیازی به آسنکرون بودن درخواست ها (request) نیست.

طرز کار: بجای اینکه مرورگر در آغاز ارتباط تنها یک صفحه HTML را بارگذاری کند، یک موتور Ajax را نیز همراه با صفحه بارگذاری می کند، که به زبان جاوااسکریپت نوشته شده و معمولاً مابین فریم‌هایی در صفحه پنهان شده است. این موتور مسئول به روز کردن رابط کاربری که کاربر می بیند و همچنین برقراری ارتباط با سرور در پشت پرده است. موتور Ajax امکان تعامل کاربر با برنامه را، مستقل از ارتباطات و تبادل اطلاعات بین برنامه و سرور می دهد. بنابراین، کاربر دیگر شاهد یک صفحه خالی و ساعت شنی و انتظار برای انجام عملیات سمت سرور نخواهد بود.

آن دسته از فعالیت‌های کاربر که باعث یک درخواست HTTP می شدند، اکنون با استفاده از جاوااسکریپت، موتور Ajax را فراخوانی می کنند. در مورد پاسخ به آن دسته از فعالیت‌های کاربر که نیاز به ارسال درخواست به سرور ندارند (مانند اعتبار سنجی‌های ساده داده‌ها، ویرایش داده‌های درون

حافظه سیستم کاربر و حتی هدایت کاربر به بخش‌های مختلف برنامه در برخی موارد) موتور Ajax خود وارد عمل می‌شود. اگر موتور Ajax برای پاسخ‌گویی نیاز به چیزی از سمت سرور داشته باشد، (مثلاً اطلاعاتی را از بانک اطلاعاتی بخواهد، کدهای جدیدی برای رابط کاربر لازم داشته باشد و...) بصورت غیر همروند و معمولاً در قالب XML یک درخواست به سرور ارسال می‌کند. ارسال این درخواست هیچ وقفه‌ای در کار کاربر با برنامه ایجاد نخواهد کرد.

Ajax مانند DHTML یا LAMP یک تکنولوژی نیست و گروهی از تکنولوژی‌هاست و از ترکیب HTML و CSS برای نشانه‌گذاری و اطلاعات style، استفاده می‌کند. دسترسی به مدل شیء سند (Document Object Model) یا DOM به وسیله JavaScript است؛ جهت نمایش پویا و برای اینکه کاربر بتواند با اطلاعات ارائه شده تعامل داشته باشد. JavaScript و شیء XMLHttpRequest: متدی برای تبادل آسنکرون داده بین مرورگر و سرور برای جلوگیری از بارگذاری مجدد و کامل صفحه، ارائه می‌کند.

نمونه ای از ایجاد request:

```
if (window.XMLHttpRequest) // Standard object
{
    xhr = new XMLHttpRequest(); // Firefox, Safari, ...
}
else
if (window.ActiveXObject) // Internet Explorer
{
    xhr = new ActiveXObject("Microsoft.XMLHTTP");
}
```

```
// waiting for response
if (xhr.readyState == 4)
{
    // Received, OK
} else
{
    // Wait...
}

//making the request
xhr.open('GET', 'http://www.xul.fr/somefile.xml', true);
xhr.send(null);
```

روش کار برنامه‌های کلاسیک وب چیزی شبیه این است: اکثر تعاملات کاربر با رابط کاربری باعث ارسال یک درخواست به سرور می‌شود. سرور پردازش‌های لازم را انجام داده و سپس یک صفحه HTML به کاربر باز می‌گرداند. این مدل بر اساس هدف اصلی وب، یعنی ایفای نقش یک رسانه برای ابرمتن است. اما آنچه وب را برای ابرمتن‌ها مناسب می‌کند، الزاماً آن را برای برنامه‌های نرم‌افزاری نیز مناسب نخواهد کرد. قسمت‌های مهم معماری پیشنهادی ای‌جکس به این قرارند:

رخدادهای کوچک سمت سرور : اجزای درون یک برنامه وب می‌توانند، درخواست‌های کوچکی برای سرور ارسال کنند، اطلاعات مورد نیازشان را از سرور دریافت کنند، و هر قسمت از صفحه را که لازم است بر اساس DOM به‌روز کنند، بدون اینکه تمام صفحه دوباره بارگذاری و ترسیم شود.

غیر همروندی : درخواست‌هایی که به سرور ارسال می‌شود، باعث مسدود شدن صفحه مرورگر کاربر نمی‌شوند. کاربر می‌تواند به کار با قسمت‌های دیگر برنامه ادامه دهد و هنگامی که پاسخی از سرور دریافت شد، با به‌روز کردن رابط کاربر، به او اطلاع داده می‌شود.

عکس‌العمل برای هر رخداد : تقریباً می‌توان برای تمام رخدادهایی که کاربر به وجود می‌آورد با سرور ارتباط برقرار کرد. مرورگرهای مدرن اکثر رخدادهایی را که سیستم‌عامل اجازه می‌دهد دریافت می‌کنند: کلیک کردن موس، حرکت موس روی شیء، زدن یک کلید و... هر رخدادی می‌تواند باعث ارسال یک درخواست به صورت غیر همروند برای سرور شود.

تکنولوژی‌ها:

- HTML یا XHTML و CSS برای ارائه یا presentation
- مدل شیء سند (Document Object Model) یا DOM برای تعامل و نمایش پویای داده
- شیء نشان‌گذاری جاوااسکریپت (JavaScript Object Notation) یا JSON برای تبادل داده و یا XML برای تبادل داده و XSLT برای تغییر و دستکاری آن
- XMLHttpRequest object برای ارتباط آسنکرون
- زبان‌های اسکریپت client-side مثل JavaScript و VBScript برای ادغام این تکنولوژی‌ها-

مشکلات Ajax :

اگرچه شکاف بین برنامه‌های دسکتاپ و برنامه‌های وب را بسیار کم‌تر کرده‌است، اما مشکلاتی نیز دارد که هنگام کار به این روش، باید به آن‌ها توجه کرد. یکی از مشکلاتی که در مورد Ajax عنوان می‌شود، این است که این روش کاربرد کلید Back مرورگرها را مختل کرده و این بر خلاف عادت کاربران در محیط وب است. البته برای حل این مسئله راه‌حلهایی پیشنهاد شده است.

تاخیر شبکه و نگرانی در مورد زمان پاسخگویی سیستم نیز مسأله دیگری است. مسأله تاخیر شبکه در یک برنامه Ajax باید با دقت در نظر گرفته شود. تأخیر در بارگذاری اولیه رابط کاربر برنامه، چیزیست که معمولاً برای کاربر قابل درک نیست و باید با دادن پیغامی مناسب، کاربر را از اینکه سیستم در حال بارگذاری، یا انجام فعالیتی و برقراری ارتباطی با سرور است، آگاه کرد.

مسأله دیگر استفاده Ajax از جاوااسکریپت است و اینکه کاربر می‌تواند اجازه اجرای جاوااسکریپت در مرورگر خود را ندهد. پس باید پیش از شروع بارگذاری موتور Ajax از اجازه داشتن اجرای جاوااسکریپت روی مرورگر کاربر اطمینان پیدا کرد.

همچنین کدهای جاوااسکریپت برنامه را هر کسی به راحتی می‌تواند ببیند. در نتیجه بهتر است قسمت‌هایی از برنامه را که مربوط به امنیت، یا منطق پردازشی برنامه می‌شود، سمت سرور نگه داریم.

۵. ASP Active Server Pages

ASP اولین موتور اسکریپت مایکروسافت سمت سرور برای ایجاد پویا و تعاملی صفحات وب است. ابتدا به عنوان یک add-on برای Internet Information Services(IIS) در ویندوز NT4 عرضه شد. ASP محیط اجرایی سمت سرور در Microsoft Internet Information Server (IIS) نسخه 3.0 است که اجازه اجرای اسکریپت های ActiveX و ActiveX server component در سرور را می دهد.

یک صفحه ASP، یک صفحه HTML حاوی اسکریپت سمت سرور است که به وسیله web server قبل از ارسال به مرورگر کاربر پردازش می شوند. به جز HTML می توان ASP را با XML و COM نیز ترکیب کرد.

ASP 2.0 شش شیء درون ساخته ارائه می دهد: برنامه (application)، خطای ASP

(ASPError)، درخواست (Request)، پاسخ (Response)، server و session. به عنوان مثال session شیئی بر اساس cookie وضعیت متغیرها را صفحه به صفحه نگه می دارد.

صفحات وب با فرمت فایل asp. از ASP استفاده می کنند، با این وجود بعضی وبسایت ها انتخاب زبان اسکریپتشان را به دلایل امنیتی پنهان می کنند(هنوز از فرمت htm و html استفاده می کنند). صفحات با فرمت aspx. از ASP.NET کامپایل شده(مبتنی بر .NET Framework

مایکروسافت) استفاده می کنند که آنها را سریعتر و قوی تر نسبت به scripting سمت سرور در ASP می کند و در زمان run-time ترجمه می شوند؛ با این حال هنوز صفحات ASP.NET شامل بعضی ASP scripting هستند. اکثر صفحه های ASP با VBScript نوشته می شوند ولی به کمک

نوشته‌های @language یا <script language="language" runat="server"> می‌توان به جای VBScript از سایر موتورهای پردازش‌نویسی فعال دیگر نیز استفاده کرد.

۶. Web Browser یا مرورگر وب

مرورگر وب به نرم‌افزارهای کاربردی گفته می‌شود که برای دریافت، نمایش و مرور اطلاعات از world wide web مورد استفاده قرار می‌گیرند. یک منبع اطلاعات با یک URL شناخته می‌شود و می‌تواند یک صفحه وب، تصویر، ویدئو و یا هر قطعه اطلاعاتی دیگری باشد. Hyperlinks کاربر را قادر می‌سازند که به راحتی میان مرورگر و منابع مرتبط حرکت کنند. هر چند مرورگرها در درجه اول برای دسترسی به world wide web طراحی شدند، اما می‌توانند برای دسترسی به اطلاعات سرورهای وب در شبکه‌های خصوصی و یا پرونده‌ها در سیستم‌های پرونده مورد استفاده قرار گیرند.

اصلی ترین مرورگرهای وب عبارتند از: Mozilla Firefox, Internet Explorer.

Opera, Apple Safari و Google Chrome.

تمام مرورگرهای اصلی اجازه باز کردن چند منبع همزمان را یا در پنجره دیگر و یا tab دیگر در همان پنجره به کاربر می‌دهند. همچنین شامل pop-up blockerها برای جلوگیری از باز شدن ناخواسته پنجره‌ها بدون اجازه کاربر، می‌شوند.

امنیت و حریم خصوصی : بیشتر مرورگرها HTTP Secure را پشتیبانی می‌کنند و راه‌های ساده برای پاک کردن cookie, web cache و browsing history ارائه می‌دهند.

مرورگرهای مدرن ترکیبی از HTML استاندارد و de facto و XHTML را پشتیبانی می‌کنند که باید به صورت یکسان توسط تمام مرورگرها ارائه شود.

PHP ۷

PHP مخفف PHP: Hypertext Preprocessor (پیش پردازنده فرامتن)، یکی از

پرطرفدارترین زبان‌های پردازنده نویسی یا اسکریپتینگ متن باز (open source) و تحت وب است که بیشتر برای طراحی برنامه‌های سمت سرور در وبگاه‌های پویا (Dynamic) مورد استفاده قرار می‌گیرد.

ساختار این زبان بسیار شبیه زبان C و Perl می‌باشد. شکل پذیری فوق العاده آن و نیز همگونی با اغلب بانک‌های اطلاعاتی از قبیل MySQL قدرت آن را افزایش داده است. مشهورترین نرم‌افزارهای ایجاد شده با PHP عبارت‌اند از: phpBB، SMF، Mambo، Joomla، PHPNuke، MediaWiki. همچنین PHP می‌تواند جایگزینی برای زبان‌های JSP، ASP.NET و Perl باشد. PHP امکان استفاده از انواع مختلف پایگاه‌های داده را از جمله: MySQL، SQLite، Oracle، PostgreSQL، IBM DB2 و Microsoft SQL Serve با دستورهای ساده فراهم می‌سازد. PHP روی بیشتر سیستم‌عامل‌های معروف از جمله لینوکس، یونیکس، ویندوز و Mac OSX و با اغلب Web Serverهای معروف، قابل اجراست.

PHP بیشتر بر روی سرور وب اجرا می‌شود و عمدتاً از آن در طراحی وب استفاده می‌شود. PHP معمولاً مانند یک فیلتر عمل می‌کند، به این صورت که کد برنامه را به عنوان ورودی گرفته و صفحات وب را به عنوان خروجی تولید می‌کند. خروجی آن غالباً از نوع HTML می‌باشد. همچنین می‌تواند برای اسکریپت کردن خط فرمان استفاده شود. PHP می‌تواند بر روی صفحات وب و سیستم‌های عامل و یا پایگاه‌های داده مستقر شده یا به عنوان سیستم مدیریتی پایگاه داده استفاده شود. PHP رایگان هست و گروه PHP منبع کاملی از کد آن را در اختیار کاربران قرار می‌دهد. همچنین می‌توان از آن برای کاربردهای شخصی استفاده کرد.

یک برنامه PHP می‌تواند قسمتی از یک برنامه HTML باشد. برای جداسازی برنامه PHP و

HTML باید قسمت PHP را بین `<?php?>` قرار داد به عنوان نمونه:

```
</code>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>PHP Test</title>
  </head>
  <body>
    <?php
    echo 'Hello World';
    ?>
  </body>
</html>
```

۸. MySQL

MySQL یک RDBMS است که به عنوان سروری که دسترسی چندکاربره به تعدادی از پایگاه های داده را فراهم می کند، اجرا می شود. SQL از Structured Query Language (زبان پرسشی ساخت یافته) می آید. به دلیل سادگی نصب و مدیریت، و متن باز بودن (open source) از محبوبیت زیادی برخوردار است SQL. محبوب ترین زبان کامپیوتری است که برای ایجاد، تغییر، بازیابی و عملیات بر روی داده ها در مدل رابطه ای می باشد. این زبان به سمت مدل شی گرا-رابطه ای پیشرفت کرده است.

برنامه هایی که از MySQL استفاده می کنند عبارتند از: Joomla، WordPress، MyBB، phpBB، Drupal و دیگر برنامه هایی که بر LAMP ساخته شده اند. MySQL همچنین در بسیاری از محصولات در مقیاس بزرگ و مهم World Wide Web از جمله Wikipedia، Facebook، Flickr و YouTube استفاده شده است.

MySQL به زبان C و C++ و تجزیه کننده (parser) SQL آن به yacc نوشته شده است. MySQL در platform های مختلفی از جمله HP-UX، FreeBSD، BSDi، AIX، Microsoft Windows، Mac OS X، Linux، IRIX، i5/OS، eComStation، QNX، OS/2 Warp، OpenSolaris، OpenBSD، Novell NetWare، NetBSD، SCO UnixWare، SCO OpenServer، SunOS، Symbian، Solaris کار می کند.

بسیاری از زبان های برنامه نویسی با API شامل کتابخانه هایی برای دسترسی به پایگاه داده MySQL هستند. این شامل MySQL connector برای ادغام با Visual Studio (زبان هایی مثل C# و VB معمولاً استفاده می شوند) و ODBC driver برای جاوا می شود. HTSQL، متد

query مبتنی بر URL نیز اجازه تعامل مستقیم یک پایگاه داده MySQL و هر کاربر وب به وسیله URL ساخت یافته را می دهد. MySQL server و کتابخانه های رسمی آن بیشتر در ANSI C++ پیاده سازی شده اند.

مزایای MySQL :

- کارایی بالا
- پشتیبانی cross-platform
- نماهای قابل به روز رسانی
- موتور ذخیره مستقل
- پشتیبانی SSL
- ذخیره query
- Indexing و جست و جوی full-text
- پایگاه داده کتابخانه جاسازی شده (embedded)
- Hot backup تحت شرایط خاص
- پشتیبانی جزئی Unicode
- پشتیبانی از X/Open XA distributed transaction processing
- هزینه پایین
- سهولت پیکربندی و یادگیری
- Triggers