I. Responsive Web Design (RWD)

- Định nghĩa: áp dụng nhiều bố cục trang web cho các loại kích cỡ màn hình khác nhau chứ không chỉ thiết kế một giao diện cố định như vẫn thường làm trước đó. Cộng với sự phát triển của các chuẩn HTML5 và CSS3, RWD đã trở thành một thứ quan trọng mà quản trị viên hay chủ sỡ hữu website cần phải nghĩ tới trong bối cảnh ngày càng nhiều thiết bị di động với đủ các kích cỡ, đủ loại độ phân giải màn hình khác nhau được tung ra thị trường.

- Lợi ích:

+ RWD dùng để bố cục lại giao diện trang web cho tương thích với nhiều loại kích cỡ màn hình khác nhau.



+ Trang web có thể hiển thị một cách đầy đủ trên màn hình di động. Tuy nhiên, nếu không áp dụng RWD, trang web khi xem trên thiết bị di động sẽ trở nên nhỏ xíu, bắt buộc phải dùng kính lúp, còn không thì phải zoom vào và kéo qua lại, lên xuống liên tục để đọc nội dung. Đây là trải nghiệm tiêu cực và nó khiến bạn nghĩ xấu về website, và điều tất nhiên là bạn chẳng thèm quay lại web đó nữa.

+ Nói tóm lại, RWD là một xu hướng thiết kế hoàn toàn có lợi bởi nó đảm bảo bạn sẽ luôn luôn có những trải nghiệm tốt nhất, đẹp nhất khi xem trang web dù bạn có đang dùng thiết bị nào đi nữa. Nó giúp nhà lập trình web tận dụng tối đa không gian để trình diễn những nội dung cho chúng ta xem theo cách thoải mái và thích thú nhất có thể.

- Hiệu năng

+ Năm 2009, Guy Podjarny, trưởng nhóm kiến trúc sản phẩm của công ty Akamai (một đơn vị chuyên nghiên cứu và tối ưu hóa tốc độ cho các giải pháp điện toán) đã thử nghiệm 347 trang web RWD được trình diễn trên http://mediaqueri.es bằng cách dùng Chrome trên nhiều thiết bị khác nhau, cộng với công cụ WebPageTest. Podjarny nhận thấy rằng kích thước tải về cũng như tốc độ load của các trang web này không khác mấy khi sử dụng trên nhiều thiết bị khác nhau và màn hình với độ phân giải khác nhau. Sau đó Podjarny đã trình bày phát hiện của mình tại một hội nghị chuyên về thiết kế.

+ Vậy thử nghiệm trên có ý nghĩa gì? Nó cho thấy rằng mặc dù trang web đã phản hồi lại với kích thước màn hình bằng cách thu gọn nội dung, ẩn bớt những thành phần không cần thiết, thu nhỏ cỡ ảnh... nhưng điều đó không đồng nghĩa với chuyện web sẽ tải nhanh hơn. Nó cũng không đảm bảo rằng dung lượng tải về sẽ nhỏ hơn, ít chiếm băng thông hơn. Việc tối ưu hóa này hoàn toàn nằm trong tay lập trình viên và nhà thiết kế web.



*Ghi chú: HTTP Request là những yêu cầu mà trình duyệt gửi đến máy chủ để tải hoặc làm một thứ gì đó. Máy chủ sẽ trả lời (response) lại tương ứng với request. Ví dụ: máy tính của bạn request để tải một trang web, server sẽ trả lời lại: "Đây, đây là file mà anh cần". Ngoài ra còn có request để tải ảnh, tải file XML,... Có thể thấy ở kết quả trên, số HTTP khác nhau giữa hai loại giao diện chỉ là 1 request, không đáng kể.*

+ Còn bên dưới là biểu đồ so sánh về kích thước trang của các web RWD trên hai loại độ phân giải do Akamai thực hiện. Đến 86% trang web khi tải giữa hai loại màn hình không cho thấy sự khác biệt rõ rệt về dung lượng trang, tức là chúng ta chỉ tiết kiệm được một khoảng dung lượng không đáng kể khi duyệt web bằng mobile.



- Độ phức tạp:

+ Thật ra trước đây các lập trình viên cũng có nghĩ đến một biện pháp khác khi mà RWD chưa phổ biến, đó là xây dựng một phiên bản di động dành cho web (ví dụ: m.tinhte.vn). Cách này cũng tốt, tác dụng gần như tương đương với RWD. Tuy nhiên, nó là một trang tập tin HTML riêng, một file CSS riêng được viết riêng cho thiết bị di động, hình ảnh cũng được thiết lập với kích thước nhỏ hơn. Và mặc dù tốn công hơn nhưng các trang web m. đơn giản hơn việc tích hợp RWD vào chung với trang web chính. Một số cuộc thảo luận trên mạng cũng có nói rằng ngay cả một website tối ưu hóa cực kì tốt với RWD cũng tải chưa nhanh bằng trang web mobile riêng.

+ Ngoài ra, một số dịch vụ online cũng có hỗ trợ chuyển RSS thành một trang web riêng biệt. Lập trình viên có thể tận dụng điều này để thiết kế web cho thiết bị di động mà không phải đầu tư quá nhiều công sức. Họ chỉ cần làm cho RSS của mình đầy đủ nhất có thể là xong. Người dùng truy cập từ các smartphone, tablet sẽ không thấy giao diện chính mà chỉ thấy các dòng cập nhật mới nhất, tin tức mới nhất. Như vậy cũng đã đủ đối với một số trang web rồi.

+ Responsive web design thì ngược lại, nó vốn dĩ là phức tạp hơn bởi nhà thiết kế web đang cố gắng nhiều trải nghiệm xem khác nhau chứ không nhắm đến một loại thiết bị cụ thể nào cả. Điều đó có nghĩa là trình duyệt trên máy mobile phải đảm đương một file HTML lớn, một tập tin CSS cũng lớn không kém. Nếu không được tích hợp tốt, RWD có thể làm cho việc duyệt web di động trở nên chậm chạp hơn mặc dù bố cục rất tốt.

- Tham khảo: <https://www.tinhte.vn/threads/responsive-web-design-la-gi-va-no-giup-ich-nhu-the-nao-cho-viec-duyet-web-tren-thiet-bi-di-dong.2101375/>

II. CSS Framework

- Định nghĩa: là bộ công cụ giúp design trang web bằng css nhanh hơn. Nghĩa là nó được trừu tượng hóa lên một mức cao hơn. Thay vì phải hiểu rõ về các bộ chọn, các thuộc tính và giá trị trong CSS để style cho trang web của mình, thì chỉ cần biết các thành phần có trên trang web như form, navbar, tooltip, dropdown-menu, modal, button,….. và thêm nó vào trang html một cách thích hợp. Công việc còn lại là của CSS Framework.

- Phiên bản: Hiện tại có rất nhiều CSS Framework được phát triển. Trong đó, có 2 loại được phát triển mạnh nhất đến thời điểm này là:

+ Bootstrap (v2 và v3)

+ Foundation (v3, v4 và v5)

- Tính năng:

+ Hỗ trợ khả năng Responsive: tức là trang web sẽ tự động co giãn theo kích thước của cửa sổ trình duyệt.

+ Tương thích tốt với thiết bị cỡ nhỏ: với sự phổ biến của smartphone hiện nay, đây là một yếu tố quan trọng. Không cần phải design một bản riêng cho mobile, với bootstrap bạn chỉ cần thiết kế một lần cho mọi thiết bị.

+ Được tích hợp với thư viện jQuery và tương tác tốt với chuẩn HTML5 và CSS3.

- Tham khảo: <http://thachpham.com/web-development/html-css/hoc-bootstrap3-grid-system.html>

III. Twitter Bootstrap (gọi tắt là Bootstrap)

- Định nghĩa:

+ Bootstrap là một CSS Framework phổ biến nhất hiện nay do Twitter phát triển.

+ Bootstrap bao gồm các mã CSS + HTML cơ bản cho typography, forms, buttons, tables, grids, navigation, và nhiều thành phần khác của website.

+ Twitter Bootstrap cung cấp lưới cố định (fixed) rộng 940px và 12 cột. Tất nhiên là cũng có giải pháp cho việc dùng layout dạng động (fluid).

+ Style của các phần tử HTML trong Twitter Bootstrap khá đơn giản và thanh lịch. Ví dụ như phần đổ bóng trong input, highlight của bảng biểu, các mã CSS hiển thị cảnh báo, tabs, phân trang…

- Phiên bản: Hiện tại Bootstrap bao gồm 2 phiên bản v2 và v3

- Tính năng: Bootstrap giúp chúng ta giảm thiểu thời gian thiết kết html và css. Bootstrap định nghĩa sẳn các class css. Công việc của chúng ta chỉ là sử dụng các class đó vào mục đích của mình. Bootstrap còn hổ trợ Responsive design, một cách làm giao diện đa thiết bị rất được ưu chuộng trong thời gian gần đây.

- Tham khảo: <http://thuctapcungdoanhnghiep.vn/hoc-lap-trinh-website/gioi-thieu-va-huong-dan-cach-su-dung-bootstrap/>

- So sánh Bootrap với các CSS Framework khác: http://responsive.vermilion.com/compare.php

|  |  |  |
| --- | --- | --- |
|  | **Bootstrap 3** | **Foundation v5** |
| **Summary** | Grids: Fluid  UI tools: Lots of widgets; good for rapid prototyping  History: Mobile-first update to the style guide for internal tools developed by Twitter | Grids: Fluid. Best in class grids across all viewports.  UI tools: Powerful and modular set of tools. More style agnostic than Bootstrap.  History: Performance and efficiency improvements to v4 |
| **Version** | 3.2 | 5.2.2 |
| **Last Released** | June 26, 2014 | April 4, 2014 |
| **Website** | [getbootstrap.com](http://getbootstrap.com/) | [foundation.zurb.com](http://foundation.zurb.com/) |
| **GitHub** | [github.com/twbs/bootstrap](https://github.com/twbs/bootstrap) | [github.com/zurb/foundation](https://github.com/zurb/foundation) |
| **Creator** | Twitter | ZURB |
| **License** | MIT License | MIT License |
| **Browser Support** | Chrome (Mac, Windows, iOS, and Android) Safari (Mac and iOS only) Firefox (Mac, Windows) Opera (Mac and Windows) IE8+ | Desktop: Chrome, Firefox, Safari, IE9+ Mobile: iOS (iPhone), iOS (iPad), Android 2, 4 (Phone), Android 2, 4 (Tablet), Windows Phone 7+, Surface |
| **Vermilion Commentary** | Unlike the release from 2 to 3, Bootstrap 3.2 has no major changes from 3.1. Changelog can be found [here](https://github.com/twbs/bootstrap/releases/tag/v3.2.0). New features include: 1.) Responsive Embeds - classes to proportionately scale down iframes or embeds, 2.) Responsive Utility Classes - now greater control over visibility and hidden classes for block, inline-block, and inline, so now the old .visible-xs can be .visible-xs-block, and 3.) Copy Docs Snippets - usage of 3rd party service ZeroClipboard, to allow for faster copying and pasting for developers of code snippets. Additional backend/administrative changes of bootstrap were made, including 1.)"LMVTFY" - bootstrap now validates code automatically before adding it in the GitHub repo, 2.) What Bootstrap is calling the "Wall of Browser Bugs" - see more about it at <http://getbootstrap.com/browser-bugs/>, and 3.) Docs have been updated to be easier to follow. Many bug fixes were made, among them include: improved progress bar for flexibility, css repaint performance (most notably through scrolling), keyboard navigation added to carousel, modal fix from shifting left upon open. | Foundation 5 is similar to version 4, with some performance improvements and developer tools. In conjunction with v5, ZURB also announced an [official enterprise consulting service](http://foundation.zurb.com/business/services.html). Version 5 switches to jQuery 2, removing support for Zepto, which was added in v4. This change, along with the addition of fastclick.js, should make performance better for both desktop and mobile users.  New [templates](http://foundation.zurb.com/templates.html) were also released with v5 for faster prototyping.  On the Sass side, ZURB added[Libsass](http://libsass.org/) to improve compilation performance.  Finally, the [developer docs](http://foundation.zurb.com/docs/) have been significantly improved. You can [read more here](http://zurb.com/article/1280/foundation-5-blasts-off--2) |

**IV. Devexpress WebForm Asp.net**

- Định nghĩa: Dùng các công cụ của Devexpress để hỗ trợ cho WebForm mà mặc định Visual Studio không hỗ trợ

- Phiên bản: Devexpress 13.1

- Tính năng: Do sử dụng các thư viện được viết sẵn nên tiết kiệm thời gian và giảm khả năng gây ra lỗi.

- Tham khảo: https://demos.devexpress.com/ASP/

- Các thư viện mà website sử dụng từ Devexpress

1. Grid View

+ Định nghĩa: Cho phép hiển thị và tùy biến dữ liệu từ List

+ Phiên bản: Devexpress 13.1

+ Tính năng: Grid cho phép Group theo cột, kéo thả để thay đổi vị trí giữa các cột trong bảng, cho phép tùy biến từng cột, thay đổi giao diện thông qua giao diện sẵn có hoặc sử dụng giao diện bên ngoài.

+ Tham khảo: documentation.devexpress.com/#AspNet/CustomDocument5823

2. Image and Data Browsing

+ Định nghĩa: Cho phép hiển thị và tùy biến dữ liệu từ List theo dạng cây phân hệ.

+ Phiên bản: Devexpress 13.1

+ Tính năng: Tree List hiển thị dữ liệu theo dạng cây chỉ cần biết được thông số IdUnique và Idparent, kéo thả để thay đổi vị trí giữa các cột trong cây, cho phép tùy biến từng cột, thay đổi giao diện thông qua giao diện sẵn có hoặc sử dụng giao diện bên ngoài.

+ Tham khảo: documentation.devexpress.com/#AspNet/CustomDocument14828

3. Docking and Popups

4. Loading Panel

5. Image Slider

6. Kết hợp Popup Control + Navigation

IV. Devexpress WinForm

- Định nghĩa: Dùng các công cụ của Devexpress để hỗ trợ cho WinForm mà mặc định Visual Studio không hỗ trợ

- Phiên bản: Devexpress 13.1

- Tính năng: Do sử dụng các thư viện được viết sẵn nên tiết kiệm thời gian và giảm khả năng gây ra lỗi.

- Tham khảo: https://www.devexpress.com/Products/NET/Controls/WinForms/

- Các thư viện mà website sử dụng từ Devexpress

1. Reporting

+ Định nghĩa: Cho phép xuất báo cáo và tùy biến dữ liệu từ Grid hoặc datasource

+ Phiên bản: Devexpress 13.1

+ Tính năng:

+ Tham khảo: documentation.devexpress.com/WindowsForms/CustomDocument8118

2. Data Grid

+ Định nghĩa:

+ Phiên bản: Devexpress 13.1

+ Tính năng:

+ Tham khảo: documentation.devexpress.com/#WindowsForms/CustomDocument3455

3. Ribbon, Menu and Docking

+ Định nghĩa:

+ Phiên bản: Devexpress 13.1

+ Tính năng:

+ Tham khảo: documentation.devexpress.com/WindowsForms/CustomDocument1199

4. Tree List

+ Định nghĩa:

+ Phiên bản: Devexpress 13.1

+ Tính năng:

+ Tham khảo: documentation.devexpress.com/#WindowsForms/CustomDocument2434

5. Printing-Exporting

+ Định nghĩa:

+ Phiên bản: Devexpress 13.1

+ Tính năng:

+ Tham khảo: documentation.devexpress.com/#WindowsForms/CustomDocument2079

6. Navigation Bar

+ Định nghĩa:

+ Phiên bản: Devexpress 13.1

+ Tính năng:

+ Tham khảo: documentation.devexpress.com/#WindowsForms/CustomDocument4870

7. Look & Feel

- Định nghĩa: Là một chức năng của Devexpress cho phép thay đổi giao diện phần mềm với kho giao diện có sẵn hoặc tự thiết kế.

- Phiên bản: Devexpress 13.1

- Tính năng: Cho phép thay đổi giao diện một cách trực quan mà không cần phải restart lại phần mềm.

- Tham khảo: documentation.devexpress.com/#WindowsForms/CustomDocument1092

V. Tính năng Static Load Website trong WebForm

- Định nghĩa: Code behide postback dùng UpdatePanel để load tĩnh

- Phiên bản:

- Tính năng:

- Tham khảo:

V. Mã hóa dữ liệu giữa Client và Server

- Định nghĩa: Dùng thuật toán mã hóa SHA-1 và Chuỗi ngẫu nhiên để mã hóa dữ liệu trước khi đưa lên server.

- Phiên bản:

- Tính năng: Đảm bảo an toàn dữ liệu

- Tham khảo:

Responsive

Bootrap

UserControl (chỉ dùng cho WebForm, MVC ko có)

Code behide postback dùng UpdatePanel để load tĩnh

Dùng các công cụ Devexpress cho web

Giao diện look and feel cho winform

Vì host dùng phiên bản 4.0 nên phải downgrade, trong quá trình downgrade gặp những khó khăn thách thức

Các phiên bản web có thể chạy

1 số thứ boottrap ko hỗ trợ phải dùng devexpress

Công nghệ đổ dữ liệu vào cây của devexpress

Chỉnh sửa router URL đường dẫn