



Nội dung thuyết trình - 123

Thương mại điện tử (Trường Đại học Công nghệ thông tin, Đại học Quốc gia Thành phố Hồ Chí Minh)



Scan to open on Studocu

4.5 CÁC CÔNG CỤ KHÁC CỦA TRANG WEB THƯƠNG MẠI ĐIỆN TỬ

Bây giờ bạn đã hiểu các yếu tố chính ảnh hưởng đến tốc độ, khả năng và khả năng mở rộng của trang web của mình, chúng ta có thể xem xét một số yêu cầu quan trọng khác. Bạn sẽ cần một thiết kế trang web hợp lý, có ý nghĩa kinh doanh—không nhất thiết phải là một trang web để làm du khách kinh ngạc hoặc phấn khích, mà là để bán một thứ gì đó cho họ. Bạn cũng sẽ cần biết cách xây dựng nội dung động và tính tương tác vào trang web của mình—không chỉ hiển thị các trang HTML tĩnh. Bạn phải có khả năng theo dõi khách hàng đến, rời đi và quay lại trang web của mình để có thể chào đón khách hàng quay lại (“Chào Sarah, chào mừng trở lại!”). Bạn cũng sẽ muốn theo dõi khách hàng trên toàn trang web của mình để có thể cá nhân hóa và tùy chỉnh trải nghiệm của họ. Bạn chắc chắn sẽ muốn khả năng cho khách hàng tạo nội dung và phản hồi trên trang web của mình để tăng sự gắn kết của họ với thương hiệu của bạn. Cuối cùng, bạn sẽ cần thiết lập một bộ chính sách thông tin cho trang web của mình—chính sách quyền riêng tư, khả năng truy cập và truy cập thông tin. Để đạt được các khả năng kinh doanh này, bạn sẽ cần lưu ý đến một số hướng dẫn thiết kế và các công cụ phần mềm bổ sung có thể đạt được chức năng kinh doanh cần thiết một cách hiệu quả về chi phí.

THIẾT KẾ WEBSITE: CÁC CÂN NHẮC KINH DOANH CƠ BẢN

Đây không phải là một bài viết về cách thiết kế website. (Trong Chương 6, chúng ta sẽ thảo luận về các vấn đề thiết kế website từ góc độ marketing.) Tuy nhiên, từ góc độ của một nhà quản lý kinh doanh, có những mục tiêu thiết kế nhất định mà bạn phải truyền đạt cho người thiết kế website để cho họ biết bạn sẽ đánh giá công việc của họ như thế nào. Tối thiểu, khách hàng của bạn sẽ cần có khả năng tìm thấy những gì họ cần trên trang web của bạn, thực hiện mua hàng và rời đi. Một trang web làm phiền lòng khách hàng có nguy cơ mất khách hàng mãi mãi. Xem Bảng 4.8 để biết danh sách những phàn nàn phổ biến nhất của người tiêu dùng về các trang web.

Một số nhà phê bình cho rằng thiết kế kém phổ biến hơn thiết kế tốt. Có vẻ như việc mô tả những gì làm khó chịu mọi người về các trang web dễ dàng hơn là mô tả cách thiết kế một trang web tốt. Những trang thương mại điện tử tệ nhất khiến việc tìm kiếm thông tin về sản phẩm của họ trở nên khó khăn và làm cho việc mua hàng trở nên phức tạp; chúng có các trang bị thiếu và các liên kết bị hỏng, cấu trúc điều hướng gây nhầm lẫn, và đồ họa hoặc âm thanh gây khó

chịu mà bạn không thể tắt đi. Bảng 4.9 trình bày lại những trải nghiệm tiêu cực này như là các mục tiêu tích cực cho thiết kế trang web.

CÁC TÍNH NĂNG TRANG WEB THƯƠNG MẠI ĐIỆN TỬ LÀM KHÓ CHỊU KHÁCH HÀNG (Bảng 4.8)

Yêu cầu người dùng xem quảng cáo hoặc trang giới thiệu trước khi truy cập nội dung trang web	Không thể sử dụng nút "Quay lại" của trình duyệt
Quảng cáo pop-up và cửa sổ bật lên từ dưới lên	Không có thông tin liên hệ (chỉ có mẫu web)
Quá nhiều cú nhấp chuột để truy cập nội dung	Màn hình splash, hoạt ảnh không cần thiết, v.v.
Các liên kết không hoạt động	Nhạc hoặc âm thanh khác tự động phát
Điều hướng gây nhầm lẫn; không có chức năng tìm kiếm	Các yếu tố thiết kế không chuyên nghiệp
Yêu cầu đăng ký và đăng nhập trước khi xem nội dung hoặc đặt hàng	Văn bản khó đọc do kích thước, màu sắc, định dạng
Trang tải chậm	Lỗi chính tả
Nội dung đã lỗi thời	Không có hoặc không rõ chính sách đổi trả

CÔNG CỤ TỐI ƯU HÓA CÔNG CỤ TÌM KIẾM

Một trang web chỉ có giá trị từ góc độ kinh doanh khi có nhiều người truy cập vào nó. Điểm dừng đầu tiên của hầu hết khách hàng khi tìm kiếm một sản phẩm hoặc dịch vụ là bắt đầu với một công cụ tìm kiếm, theo dõi danh sách trên trang (thường bắt đầu với ba đến năm kết quả đầu tiên), và sau đó liếc nhìn các quảng cáo được tài trợ ở bên phải. Bạn càng xuất hiện cao trên các trang công cụ tìm kiếm, bạn sẽ càng nhận được nhiều lưu lượng truy cập. Xuất hiện ở Trang 1 tốt hơn nhiều so với xuất hiện ở Trang 2. Vậy làm thế nào để bạn lên Trang 1 trong danh sách tìm kiếm tự nhiên (không trả phí)? Mặc dù mỗi công cụ tìm

kiếm đều khác nhau, và không công cụ nào công bố thuật toán xếp hạng trang của họ, nhưng có một số ý tưởng cơ bản hoạt động tốt:

- Metatag, từ khóa, tiêu đề, nội dung trang: Các công cụ tìm kiếm "crawl" trang web của bạn và xác định các từ khóa cũng như các trang tiêu đề và sau đó lập chỉ mục chúng để sử dụng trong tìm kiếm. Rải rác các từ khóa trên các trang của bạn mô tả chính xác những gì bạn nói bạn làm trong các phần "mô tả" và "từ khóa" của mã nguồn metatag. Mục tiêu là tìm sự cân bằng giữa các loại từ khóa khác nhau, bao gồm các từ khóa đầu ngắn hơn có thể chung chung hơn (như "xe hơi"), các từ khóa thân có thể cụ thể hơn một chút (như "xe thể thao Anh"), và các từ khóa đuôi dài chi tiết hơn nhiều (như "xe Jaguar XKE mui trần màu đỏ năm 1968").
- Cung cấp chuyên môn: Các báo cáo chuyên sâu, phân tích ngành, trang FAQ, hướng dẫn và lịch sử là những cách tuyệt vời để xây dựng sự tin tưởng của người dùng và khuyến khích họ xem trang web của bạn như một nơi để tìm kiếm sự giúp đỡ và hướng dẫn.
- Liên kết: Khuyến khích các trang web khác liên kết đến trang của bạn; tạo một blog thu hút mọi người và họ sẽ chia sẻ URL của bạn với người khác và đăng liên kết trong quá trình đó. Tạo một trang Facebook cho công ty của bạn, và cân nhắc sử dụng Instagram hoặc Pinterest để phát triển lượng người theo dõi hoặc fan base cho sản phẩm của bạn.
- Mua quảng cáo: Bổ sung nỗ lực tối ưu hóa tìm kiếm tự nhiên của bạn bằng các từ khóa và quảng cáo trả phí trên công cụ tìm kiếm. Chọn từ khóa của bạn và mua hiển thị trực tiếp trên các trang web. Bạn có thể đặt ngân sách và đặt trần để kiểm soát chi phí. Xem cái nào hiệu quả và quan sát số lượt truy cập vào trang web của bạn do mỗi chuỗi từ khóa tạo ra.
- Thương mại điện tử địa phương: Phát triển thị trường quốc gia có thể mất nhiều thời gian. Nếu trang web của bạn đặc biệt hấp dẫn đối với người dân địa phương, hoặc nếu nó liên quan đến các sản phẩm được bán tại địa phương, hãy sử dụng các từ khóa thể hiện vị trí của bạn để mọi người có thể tìm thấy bạn ở gần đó. Tên thị trấn, thành phố và khu vực trong từ khóa của bạn - chẳng hạn như "mật ong Hudson Valley" hoặc "nhạc blues San Francisco" - có thể hữu ích.

TÁM YẾU TỐ QUAN TRỌNG NHẤT TRONG THIẾT KẾ TRANG THƯƠNG MẠI ĐIỆN TỬ THÀNH CÔNG

Yếu tố	Mô tả
Chức năng	Các trang hoạt động tốt, tải nhanh và hướng khách hàng đến các sản phẩm của bạn
Thông tin	Các liên kết mà khách hàng có thể dễ dàng tìm thấy để tìm hiểu thêm về bạn và sản phẩm của bạn
Dễ sử dụng	Điều hướng đơn giản, không gây nhầm lẫn
Điều hướng dự phòng	Phương án điều hướng thay thế đến cùng một nội dung
Dễ dàng mua hàng	Chỉ cần một hoặc hai cú nhấp chuột để mua hàng
Tương thích đa trình duyệt	Trang web hoạt động với hầu hết các trình duyệt phổ biến
Đồ họa đơn giản	Tránh đồ họa và âm thanh gây phiền nhiễu, khó chịu mà người dùng không thể kiểm soát
Văn bản dễ đọc	Tránh nền làm biến dạng văn bản hoặc khiến nó khó đọc

CÔNG CỤ CHO TƯƠNG TÁC VÀ NỘI DUNG ĐỘNG (TOOLS FOR INTERACTIVITY AND ACTIVE CONTENT)

Một trang web càng tương tác, nó sẽ càng hiệu quả trong việc tạo ra doanh số và khuyến khích khách hàng quay lại. Mặc dù chức năng và dễ sử dụng là mục tiêu tối cao trong thiết kế trang web, bạn cũng sẽ muốn tương tác với người dùng và mang đến cho họ một trải nghiệm sôi động, "năng động". Bạn sẽ muốn cá nhân hóa trải nghiệm cho khách hàng bằng cách đáp ứng nhu cầu cá nhân của họ và tùy chỉnh nội dung của các đề xuất của bạn dựa trên hành vi hoặc mong muốn được bày tỏ của họ. Để đạt được các mục tiêu kinh doanh này, bạn sẽ cần cân nhắc kỹ lưỡng các công cụ cần thiết để xây dựng những khả năng này.

Các tương tác đơn giản như khách hàng gửi một cái tên, cùng với các tương tác phức tạp hơn liên quan đến thẻ tín dụng, tùy chọn người dùng và phản hồi của người dùng đối với các lời nhắc, tất cả đều yêu cầu các chương trình đặc biệt.

Các phần sau đây cung cấp mô tả ngắn gọn về một số công cụ phần mềm thường được sử dụng để đạt được mức độ tương tác cao của trang web.

Trong những ngày đầu của Web, các chương trình (scripts) Common Gateway Interface (CGI) là phương pháp chính để tạo ra tính tương tác. Common Gateway Interface (CGI) là bộ tiêu chuẩn được chấp nhận rộng rãi đầu tiên cho giao tiếp giữa trình duyệt và chương trình chạy trên máy chủ, cho phép tương tác giữa người dùng và máy chủ. CGI cho phép một chương trình thực thi truy cập tất cả thông tin trong các yêu cầu đến từ khách hàng. Sau đó, chương trình có thể tạo ra tất cả đầu ra cần thiết để tạo thành trang trả về (HTML, mã script, văn bản, v.v.) và gửi nó trở lại cho khách hàng thông qua máy chủ web. Tất cả việc tính toán diễn ra ở phía máy chủ (đây là lý do tại sao các chương trình CGI và những chương trình tương tự được gọi là chương trình "phía máy chủ"). Tuy nhiên, ngày nay, các script CGI đã trở nên lỗi thời do các vấn đề về bảo mật và các lo ngại khác, và đã được thay thế bằng các phương pháp hiện đại và an toàn hơn.

Java, Java Server Pages (JSP), and JavaScript

Java là một ngôn ngữ lập trình cho phép lập trình viên tạo ra tính tương tác và nội dung động trên máy tính khách hàng, từ đó tiết kiệm đáng kể tải cho máy chủ. Java ban đầu được phát triển bởi Sun Microsystems như một ngôn ngữ lập trình độc lập với nền tảng cho thiết bị điện tử tiêu dùng. Ý tưởng là tạo ra một ngôn ngữ mà các chương trình của nó (gọi là chương trình Write Once Run Anywhere [WORA]) có thể hoạt động trên bất kỳ máy tính nào bất kể hệ điều hành. Điều này có thể thực hiện được nếu mọi hệ điều hành tại thời điểm đó (Macintosh, Windows, Unix, DOS và hệ thống mainframe MVS) đều có cài đặt Java Virtual Machine (VM) để diễn giải các chương trình Java cho môi trường đó.

Tuy nhiên, đến năm 1995, khi Sun Microsystems phát hành Java 1.0, phiên bản công khai đầu tiên của ngôn ngữ này, đã trở nên rõ ràng rằng Java phù hợp hơn với Web so với thiết bị điện tử tiêu dùng. Các "applet" Java có thể được tải xuống máy khách qua Web và thực thi hoàn toàn trên máy tính của khách hàng. Các thẻ applet có thể được đưa vào trang HTML. Để làm được điều này, mỗi trình duyệt phải bao gồm một Java VM. Khi trình duyệt truy cập một trang có applet, một yêu cầu sẽ được gửi đến máy chủ để tải xuống và thực thi chương trình và phân bổ không gian trang để hiển thị kết quả của chương trình.

Ngày nay, Java vẫn là một trong những ngôn ngữ lập trình phổ biến nhất, với nhiều công nghệ quan trọng, chẳng hạn như nền tảng di động Google Android (mặc dù không phải iOS của Apple), tận dụng các khía cạnh của ngôn ngữ này. Java SE (Java Platform, Standard Edition) 18 và môi trường phát triển liên quan JDK (Java Development Kit) 18 là các phiên bản mới nhất của nền tảng Java tính đến tháng 2 năm 2022. Tuy nhiên, Java cũng phải đối mặt với một số thách thức. Trong những năm gần đây, nó đã bị ảnh hưởng bởi các lỗ hổng bảo mật mà Oracle (công ty đã mua lại Sun Microsystems) đã cố gắng khắc phục bằng cách thường xuyên phát hành các phiên bản mới và bản vá bảo mật. Do lo ngại về bảo mật, Oracle đã loại bỏ dần Java applet, và các phiên bản hiện tại của hầu hết các trình duyệt không còn hỗ trợ chúng nữa. Các nhà phát triển ứng dụng web dựa trên Java giờ đây đóng gói Java trong ứng dụng để không cần applet nữa (Oracle, 2022).

Java Server Pages (JSP) là một tiêu chuẩn mã hóa trang web cho phép các nhà phát triển sử dụng kết hợp HTML, script JSP và Java để tạo động các trang web để đáp ứng yêu cầu của người dùng. JSP sử dụng "servlet" Java, các chương trình Java nhỏ được chỉ định trong trang web và chạy trên máy chủ web để sửa đổi trang web trước khi nó được gửi đến người dùng đã yêu cầu nó. JSP được hỗ trợ bởi hầu hết các máy chủ ứng dụng phổ biến trên thị trường hiện nay.

JavaScript là một ngôn ngữ lập trình được phát minh bởi Netscape, được sử dụng để kiểm soát các đối tượng trên trang HTML và xử lý tương tác với trình duyệt. Nó thường được sử dụng nhiều nhất ở phía máy khách để xử lý việc xác minh và xác thực đầu vào của người dùng, cũng như để thực hiện logic kinh doanh. Ví dụ, JavaScript có thể được sử dụng trên các biểu mẫu đăng ký khách hàng để xác nhận rằng số điện thoại, mã zip hoặc địa chỉ email hợp lệ đã được cung cấp. Trước khi người dùng hoàn thành biểu mẫu, địa chỉ email được cung cấp có thể được kiểm tra tính hợp lệ. JavaScript dường như được chấp nhận nhiều hơn đối với các tập đoàn và môi trường khác phần lớn vì nó ổn định hơn và cũng bị giới hạn trong hoạt động của các trang HTML được yêu cầu. JavaScript cũng được sử dụng như một phần của Node.js, một môi trường đa nền tảng cho các ứng dụng phía máy chủ (bao gồm cả di động). Node.js là một trong những khung phát triển phía máy chủ phổ biến nhất, được sử dụng bởi các công ty như PayPal, Walmart và LinkedIn, khiến JavaScript vẫn là một ngôn ngữ quan trọng không chỉ cho phát triển web mà còn cho các ứng dụng nền tảng dưới dạng dịch vụ (PaaS).

Cũng có một số công cụ khác dựa trên JavaScript giúp tự động hóa quá trình tạo ứng dụng web. React và Vue là các thư viện JavaScript mã nguồn mở

để xây dựng giao diện người dùng. AngularJS (đôi khi còn được gọi là Angular.js) là một công cụ phổ biến khác. AngularJS là một khung ứng dụng web phía trước mã nguồn mở dựa trên JavaScript mở rộng chức năng của HTML. D3.js (viết tắt của Data Driven Documents) là một thư viện JavaScript để trực quan hóa dữ liệu bằng HTML, SVG và CSS. jQuery là một thư viện JavaScript đa nền tảng được thiết kế để đơn giản hóa việc lập trình phía máy khách của HTML. Ajax (JavaScript và XML không đồng bộ) sử dụng nhiều công cụ khác nhau, bao gồm cả JavaScript, để cho phép các trang web được cập nhật không đồng bộ (tức là chỉ cập nhật một phần của trang thay vì phải tải lại toàn bộ trang để thay đổi chỉ một phần của nội dung). TypeScript là một ngôn ngữ lập trình mã nguồn mở được phát triển và duy trì bởi Microsoft, được thiết kế để phát triển các ứng dụng lớn. Nó được bao gồm trong gói phát triển phần mềm Visual Studio của Microsoft. Typescript là một tập hợp mở rộng của JavaScript, xây dựng trên JavaScript nhưng có thêm các công cụ phát triển. Các chương trình JavaScript hiện có cũng là các chương trình TypeScript hợp lệ. TypeScript có thể được sử dụng để phát triển cả ứng dụng JavaScript phía máy khách và phía máy chủ. Việc sử dụng TypeScript đã tăng vọt trong vài năm qua (Krill, 2022).

Active Server Pages (ASP) and ASP.NET

Active Server Pages (ASP) được phát minh bởi Microsoft vào cuối năm 1996 và phát triển nhanh chóng để trở thành kỹ thuật chính cho lập trình web phía máy chủ trong môi trường Windows. ASP cho phép các nhà phát triển dễ dàng tạo và mở các bản ghi từ cơ sở dữ liệu và thực thi các chương trình trong một trang HTML cũng như xử lý tất cả các hình thức tương tác khác nhau được tìm thấy trên các trang thương mại điện tử. ASP cho phép tương tác diễn ra giữa trình duyệt và máy chủ. Tuy nhiên, các chương trình ASP bị giới hạn sử dụng trên các máy chủ Windows chạy phần mềm máy chủ web IIS của Microsoft.

ASP.NET, được phát hành lần đầu vào tháng 1 năm 2002 và là một phần của framework .NET của Microsoft, là phiên bản kế thừa của ASP. Phiên bản hiện tại của ASP.NET có một framework web hiện đại, đa nền tảng cho các máy chủ ứng dụng đám mây và thông thường.

ColdFusion

ColdFusion là một môi trường tích hợp phía máy chủ để phát triển các ứng dụng web và di động tương tác. Ban đầu được phát triển bởi Macromedia và hiện nay được cung cấp bởi Adobe, ColdFusion kết hợp một ngôn ngữ kịch bản dựa trên thẻ trực quan và một ngôn ngữ kịch bản máy chủ dựa trên thẻ (CFML) giúp giảm chi phí tạo ra các tính năng tương tác.

ColdFusion cung cấp một bộ công cụ mạnh mẽ cho thiết kế trực quan, lập trình, gỡ lỗi và triển khai, bao gồm khả năng tạo ứng dụng di động, các tính năng bảo mật mạnh mẽ và hỗ trợ khả năng tương tác.

Phiên bản mới nhất, Adobe ColdFusion 2021/2020, cung cấp:

- Giao diện người dùng quản trị mới
- Hỗ trợ lập trình hướng đối tượng được cải thiện
- Tính năng cài đặt nhanh chóng và mô-đun
- Công cụ đám mây và bảo mật nâng cao (Atteo, 2022)

PHP, Ruby on Rails (RoR), and Django

PHP là một ngôn ngữ lập trình mã nguồn mở, đa năng được sử dụng phổ biến nhất trong các ứng dụng web phía máy chủ để tạo ra nội dung trang web động, điều này rất quan trọng đối với các nền tảng TMĐT hiện đại. Nó cũng có thể được sử dụng cho các ứng dụng giao diện người dùng đồ họa phía máy khách, giúp tạo ra trải nghiệm mua sắm trực tuyến mượt mà. PHP là một phần của nhiều framework phát triển ứng dụng web như CakePHP và CodeIgniter, thường được sử dụng để xây dựng các cửa hàng trực tuyến và hệ thống quản lý đơn hàng. Nó cũng là thành phần trong mô hình phát triển web mã nguồn mở LAMP (Linux, Apache, MySQL, PHP), một stack công nghệ phổ biến cho việc xây dựng các trang TMĐT động và có khả năng mở rộng. Theo W3Techs, PHP là ngôn ngữ lập trình phía máy chủ được sử dụng phổ biến nhất (trên 75% các trang web), điều này cho thấy sự phổ biến của nó trong việc xây dựng các trang TMĐT. Các công nghệ khác như ASP.NET, Ruby on Rails, Java, Scala, JavaScript và Python cũng được sử dụng nhưng ít phổ biến hơn trong lĩnh vực này. Đáng chú ý, tin tặc thường cố gắng khai thác mã PHP để thực hiện các cuộc tấn công như SQL injection, chèn mã độc và lừa đảo. Đây là những mối đe dọa an ninh nghiêm trọng đối với các trang TMĐT, có thể dẫn đến rò rỉ dữ liệu khách hàng, mất thông tin thanh toán hoặc gián đoạn hoạt động kinh doanh. Do đó, việc bảo mật mã PHP là cực kỳ quan trọng đối với các doanh nghiệp TMĐT để bảo vệ khách hàng và duy trì uy tín của họ trong thị trường trực tuyến cạnh tranh.

Ruby on Rails là một framework ứng dụng web mã nguồn mở dựa trên ngôn ngữ lập trình Ruby. RoR dựa trên một triết lý được gọi là quy ước hơn cấu hình, hoặc lập trình theo quy ước (CoC), có nghĩa là framework cung cấp một bộ cục có cấu trúc giúp giảm thiểu số lượng quyết định mà lập trình viên cần phải đưa ra, từ đó đơn giản hóa và tăng tốc quá trình phát triển.

JavaScript và Ajax được tích hợp sâu vào RoR, giúp dễ dàng xử lý các yêu cầu Ajax để cập nhật trang. Ruby rất phổ biến vào đầu những năm 2000, nhưng trong vài năm gần đây, nó đã giảm sự ưa chuộng phần nào. Một số trang web nổi tiếng dựa trên RoR bao gồm Shopify, Groupon, Etsy, Kickstarter, Hulu và Airbnb (Rejman, 2022).

Django cũng là một framework ứng dụng web mã nguồn mở. Nó dựa trên ngôn ngữ lập trình Python. Django được tối ưu hóa cho việc tạo ra các trang web phức tạp, hướng cơ sở dữ liệu.

Đặc điểm của Django:

- Cho phép phát triển nhanh chóng
- Tập trung vào việc tự động hóa càng nhiều càng tốt
- Nhấn mạnh khả năng tái sử dụng của các thành phần khác nhau
- Tuân theo nguyên tắc lập trình DRY (Don't Repeat Yourself - Đừng Lặp Lại Chính Mình)

Một số trang web nổi tiếng dựa trên Django bao gồm Instagram, Spotify, Pinterest, Dropbox, NASA và Quora (Citrusbug.com, 2021).

Các yếu tố thiết kế khác (Other Design Elements)

Một cách dễ dàng để tăng năng lượng cho trang web của bạn là đưa vào một số widget phù hợp (đôi khi được gọi là gadget, plug-in, hoặc snippet). Widget là những đoạn mã nhỏ tự động thực thi trong trang web HTML của bạn. Chúng được xây dựng sẵn và nhiều widget miễn phí. Mạng xã hội và blog sử dụng widget để cung cấp cho người dùng nội dung từ khắp nơi trên Web (tiêu đề tin tức từ các nguồn tin cụ thể, thông báo, thông cáo báo chí và các nội dung thông thường khác), lịch, đồng hồ, thời tiết, trò chơi và các chức năng khác. Bạn có thể sao chép mã vào trang web HTML. Bạn có thể tìm thấy widget tại Apple's Dashboard Widgets, Wolfram|Alpha Widgets, và SIMILE Widgets. Cũng có các widget dành riêng cho các nền tảng cụ thể như WordPress, Amazon và Pinterest.

Mashup phức tạp hơn một chút và liên quan đến việc lấy chức năng và dữ liệu từ một chương trình và đưa vào một chương trình khác. Mashup phổ biến nhất liên quan đến việc sử dụng dữ liệu và phần mềm Google Maps và kết hợp với dữ liệu khác. Ví dụ, nếu bạn có một trang web bất động sản địa phương, bạn có thể tải ứng dụng Google Maps và hình ảnh vệ tinh vào trang web của mình để khách truy cập có thể có cảm nhận về khu vực lân cận. Có hàng nghìn mashup Google Map, từ bản đồ các cuộc biểu tình chính trị đến bản đồ các công ty

Fortune 500, tất cả đều kèm theo các câu chuyện tin tức và nội dung liên quan khác. Các mashup khác liên quan đến thể thao, mua sắm và tin tức.

CÔNG CỤ CÁ NHÂN HÓA/TÙY CHỈNH

Bạn chắc chắn sẽ muốn biết cách đối xử với từng khách hàng trên cơ sở cá nhân và mô phỏng một thị trường truyền thống, trực tiếp. Cá nhân hóa (khả năng đối xử với mọi người dựa trên đặc điểm cá nhân và lịch sử tương tác trước đó với trang web của bạn) và tùy chỉnh (khả năng thay đổi sản phẩm để phù hợp hơn với nhu cầu của khách hàng) là hai yếu tố then chốt của thương mại điện tử. Những yếu tố này có tiềm năng làm cho thương mại điện tử gần như mạnh mẽ như một thị trường truyền thống và thậm chí có thể còn mạnh hơn việc mua sắm tại một trung tâm thương mại vô danh ở ngoại ô. Giao tiếp trực tiếp với khách hàng trên cơ sở một-đối-một, và thậm chí điều chỉnh sản phẩm theo khách hàng, là điều khá khó khăn trong loại hình tiếp thị đại trà, giao dịch thương mại một-kích-cỡ-phù-hợp-tất-cả thông thường đặc trưng cho phần lớn hoạt động thương mại hiện đại.

Cá nhân hóa trang web liên quan đến việc thay đổi một trang web dựa trên người đang xem nó, bằng cách cung cấp nội dung động, thông điệp và ưu đãi phù hợp với từng cá nhân. Tùy chỉnh trong thương mại điện tử tập trung vào việc tạo ra các đề xuất sản phẩm được cá nhân hóa, bao gồm cả các mặt hàng tương tự và bổ sung.

Có nhiều phương pháp để thực hiện cá nhân hóa và tùy chỉnh trong thương mại điện tử. Một phương pháp cơ bản là sử dụng cookie - một tệp văn bản nhỏ được lưu trên thiết bị của người dùng. Cookie có thể chứa các thông tin như mã khách hàng, mã chiến dịch marketing, hoặc lịch sử mua hàng trên trang web. Khi khách hàng quay lại hoặc duyệt sâu hơn trong trang, hệ thống có thể truy xuất lịch sử của họ từ cơ sở dữ liệu. Những thông tin này giúp cá nhân hóa trải nghiệm mua sắm và tùy chỉnh sản phẩm cho phù hợp. Chúng tôi sẽ thảo luận chi tiết hơn về cookie và các công cụ theo dõi hành vi khác trong Chương 6.

Ngoài ra, thị trường còn có nhiều công cụ tiên tiến hơn, cung cấp khả năng phân khúc khách hàng thông minh, cá nhân hóa và đề xuất sản phẩm theo từng cá nhân. Ví dụ như Kibo Montetate và Barilliance. Hầu hết các nền tảng thương mại điện tử toàn diện như Salesforce Commerce Cloud đều tích hợp sẵn các chức năng tương tự. Ở phân khúc khác, Google Optimize là một công cụ miễn phí có thể tùy chỉnh trang web dựa trên vị trí, thiết bị và hành vi số của khách truy cập.

Bộ chính sách thông tin (THE INFORMATION POLICY SET)

Khi phát triển một trang thương mại điện tử, bạn cũng cần tập trung vào bộ chính sách thông tin sẽ quản lý trang web. Bạn cần xây dựng một chính sách bảo mật - một tập hợp các tuyên bố công khai cho khách hàng biết cách bạn xử lý thông tin cá nhân mà bạn thu thập trên trang web. Bạn cũng cần thiết lập các quy tắc về khả năng tiếp cận - một tập hợp các mục tiêu thiết kế đảm bảo rằng người dùng khuyết tật có thể truy cập trang web của bạn một cách hiệu quả. Có hơn 50 triệu người Mỹ bị khuyết tật, nhiều người trong số họ có thể cần sự hỗ trợ đặc biệt để có thể truy cập máy tính hoặc thiết bị di động (xem phần Insight on Society: Thiết kế cho Khả năng Tiếp cận). Các chính sách thông tin trong thương mại điện tử được mô tả chi tiết hơn trong Chương 8.

4.6. PHÁT TRIỂN TRANG WEB DI ĐỘNG VÀ XÂY DỰNG ỨNG DỤNG DI ĐỘNG (DEVELOPING A MOBILE WEBSITE AND BUILDING MOBILE APPLICATIONS)

Sự hiện diện đa nền tảng: Doanh nghiệp cần phát triển sự hiện diện không chỉ trên web truyền thống mà còn trên các nền tảng di động để đáp ứng xu hướng sử dụng thiết bị di động của người dùng.

Đa dạng nền tảng m-commerce: Có nhiều lựa chọn như trang web di động, ứng dụng web di động, ứng dụng native và ứng dụng hybrid. Mỗi loại có ưu điểm và chi phí riêng.

Tối ưu hóa trải nghiệm người dùng: Trang web di động được thiết kế đơn giản hóa, tập trung vào việc giúp người dùng tìm kiếm thông tin và ra quyết định mua hàng nhanh chóng.

Công nghệ phát triển: Trang web di động sử dụng các công nghệ web tiêu chuẩn như HTML, Linux, PHP và SQL, nhưng được tối ưu hóa cho thiết bị di động.

Thách thức về hiệu suất: Trang web di động thường hoạt động chậm hơn so với trang web truyền thống do giới hạn về băng thông của thiết bị di động.

Xu hướng thị trường: Hầu hết các công ty lớn đều đã phát triển trang web di động, cho thấy đây là một xu hướng quan trọng trong thương mại điện tử.

Native App

1. Định nghĩa ứng dụng native:

- a. Đây là ứng dụng được thiết kế đặc biệt để hoạt động trên phần cứng và hệ điều hành cụ thể của thiết bị di động.
- b. Các Native app được xây dựng trên các ngôn ngữ lập trình chỉ có thể được hỗ trợ bởi một nền tảng nhất định. Java, Kotlin thường là

các công nghệ được áp dụng để phát triển Android và Objective-C và Swift – cho iOS.

2. Khả năng hoạt động offline: Ứng dụng native có thể tải và xử lý dữ liệu ngay cả khi không có kết nối internet.
3. Tính đặc thù của nền tảng: Ứng dụng native cần được phát triển riêng cho từng nền tảng di động (ví dụ: iOS, Android) do sự khác biệt về phần cứng và hệ điều hành.
4. Quy trình phát triển: Ứng dụng native được xây dựng bằng ngôn ngữ lập trình phù hợp với từng nền tảng, sau đó được biên dịch thành mã nhị phân.
5. Hiệu suất cao: Ứng dụng native thực thi rất nhanh trên thiết bị di động.
6. Ứng dụng phù hợp: Ứng dụng native lý tưởng cho các trường hợp đòi hỏi hiệu suất cao như trò chơi, tương tác phức tạp, tính toán tức thời, xử lý đồ họa và quảng cáo đa phương tiện.

Mobile App

1. Định nghĩa ứng dụng web di động:
 - a. Là ứng dụng chạy trên trình duyệt web tích hợp của thiết bị di động.
 - b. nhà phát triển sử dụng một số công nghệ web như HTML, CSS và Javascript framework. Nói cách khác, chúng là nhiều trang web được đóng gói vào các container mô phỏng hành vi phần mềm cho mọi nền tảng hoạt động và theo cách tự nhiên.
2. Công nghệ phát triển: Sử dụng HTML5, CSS và JavaScript để mô phỏng tính năng của ứng dụng native.
3. Thiết kế tối ưu cho di động: Được thiết kế đặc biệt cho kích thước màn hình, điều hướng bằng ngón tay và giao diện đơn giản của thiết bị di động.
4. Khả năng ứng dụng: Có thể hỗ trợ tương tác phức tạp, tính toán thời gian thực và tích hợp GPS.
5. Hiệu suất: Nhanh hơn trang web di động nhưng chậm hơn ứng dụng native.

Hybrid App

- 1) 1. Định nghĩa ứng dụng hybrid: Kết hợp các yếu tố của ứng dụng native và ứng dụng web di động.
- 2) Đặc điểm của ứng dụng native:
 - a) Chạy trong container native trên thiết bị di động
 - b) Có quyền truy cập vào API của thiết bị

- c) Có thể tận dụng các tính năng đặc biệt của thiết bị (ví dụ: con quay hồi chuyển)
- d) Có thể được phân phối qua cửa hàng ứng dụng
- 3) Đặc điểm của ứng dụng web di động:
 - a) Sử dụng HTML5, CSS3 và JavaScript
 - b) Sử dụng công cụ trình duyệt của thiết bị để hiển thị và xử lý
- 4) Ưu điểm:
 - a) Kết hợp được ưu điểm của cả ứng dụng native và web di động
 - b) Có thể truy cập các tính năng của thiết bị mà ứng dụng web di động thông thường không thể
 - c) Linh hoạt trong phát triển và phân phối

Thiết kế sự hiện diện trên di động khác với thiết kế trang web truyền thống cho máy tính để bàn do sự khác biệt về phần cứng, phần mềm và kỳ vọng của người tiêu dùng. Bảng 4.11 mô tả một số khác biệt chính.

Các nhà thiết kế cần tính đến các hạn chế của nền tảng di động khi thiết kế. Tốc độ tải trang trên di động đã được chứng minh là một yếu tố quan trọng ảnh hưởng đến tỷ lệ chuyển đổi. Kích thước tệp nên nhỏ hơn và số lượng tệp gửi đến người dùng nên được giảm bớt. Tập trung vào một số đồ họa mạnh mẽ và giảm thiểu số lượng hình ảnh gửi đến người dùng. Ưu tiên tải nội dung quan trọng trước, và trong khi người dùng đang xử lý nội dung đó, bắt đầu tải lớp nội dung tiếp theo. Đơn giản hóa các hộp lựa chọn và danh sách để người dùng có thể dễ dàng cuộn và chọn bằng cảm ứng.

Sự hiện diện trên di động đã trở nên quan trọng đến mức nó đang thúc đẩy xu hướng ngày càng tăng là đảo ngược quy trình phát triển thương mại điện tử truyền thống và bắt đầu với việc phát triển sự hiện diện trên di động thay vì trang web máy tính để bàn (được gọi là thiết kế ưu tiên di động - mobile first design).

Thiết kế ưu tiên di động có một số ưu điểm. Thay vì tạo ra một thiết kế đầy đủ tính năng cho trang web máy tính để bàn rồi sau đó phải thu gọn lại, thiết kế ưu tiên di động tập trung vào việc tạo ra trải nghiệm tốt nhất có thể trong những hạn chế của nền tảng di động, sau đó mới bổ sung các yếu tố cho nền tảng máy tính để bàn, dần dần nâng cao chức năng của trang web.

Những người ủng hộ thiết kế ưu tiên di động cho rằng nó buộc các nhà thiết kế tập trung vào những gì quan trọng nhất, và điều này giúp tạo ra một thiết kế di động tinh gọn và hiệu quả, hoạt động tốt hơn nhiều so với một thiết kế bắt đầu với nền tảng truyền thống phải được cắt giảm để hoạt động trên thiết bị di động. Tuy nhiên, thiết kế ưu tiên di động cũng không phải không có thách thức. Nó đòi hỏi một cách tiếp cận khác có thể khó khăn đối với các nhà thiết kế quen thuộc hơn với quy trình truyền thống (The Stack Group, 2021).

Các công cụ và kỹ thuật thiết kế web đáp ứng (RWD) cho phép thiết kế một trang web tự động điều chỉnh bố cục và hiển thị theo độ phân giải màn hình của thiết bị đang xem, dù là máy tính để bàn, máy tính bảng hay smartphone. Công cụ RWD bao gồm HTML5 và CSS3, và ba nguyên tắc thiết kế chính liên quan đến việc sử dụng bố cục dựa trên lưới linh hoạt; hình ảnh và phương tiện linh hoạt; và truy vấn phương tiện. RWD sử dụng cùng một mã HTML và thiết kế cho mỗi thiết bị nhưng sử dụng CSS (xác định bố cục của trang web) để điều chỉnh bố cục và hiển thị theo hệ số hình dạng của màn hình. RWD thường hoạt động tốt cho các trang web có chức năng tương đối đơn giản (tức là các trang chủ yếu cung cấp nội dung) và người dùng tương tác theo cách tương tự bất kể thiết bị đang sử dụng. Tuy nhiên, việc sử dụng RWD có thể tốn kém, thường đòi hỏi thiết kế lại hoàn toàn giao diện của trang web. Một vấn đề khác với RWD, đặc biệt nếu không kết hợp với thiết kế ưu tiên di động, là trang web đáp ứng vẫn có kích thước và độ phức tạp của một trang web máy tính để bàn truyền thống, đôi khi làm cho nó chậm tải và hoạt động trên thiết bị di động. Một kỹ thuật khác, được gọi là thiết kế web thích ứng, đã được phát triển để giải quyết vấn đề này.

Với thiết kế web thích ứng (AWD) (đôi khi còn được gọi là phân phối thích ứng hoặc thiết kế web đáp ứng với các thành phần phía máy chủ [RESS]), máy chủ lưu trữ trang web phát hiện các thuộc tính của thiết bị đưa ra yêu cầu và, sử dụng các mẫu được định nghĩa trước dựa trên kích thước màn hình thiết bị cùng với CSS và JavaScript, tải một phiên bản của trang web được tối ưu hóa cho thiết bị đó. AWD có một số ưu điểm, bao gồm thời gian tải nhanh hơn, khả năng nâng cao hoặc loại bỏ chức năng một cách linh hoạt, và thường mang lại trải nghiệm người dùng tốt hơn, đặc biệt là đối với các doanh nghiệp mà mục đích sử dụng của người dùng khác nhau tùy thuộc vào nền tảng được sử dụng. Ví dụ, việc tạo trang web di động bằng AWD cho phép Lufthansa tập trung vào các hành động mà người dùng di động có khả năng thực hiện nhất, như check-in, lấy thông tin trạng thái chuyến bay và tra cứu lịch trình du lịch, đồng thời cung cấp trải nghiệm khác biệt so với trang web máy tính để bàn truyền thống (Vinney, 2021). Một biến thể của AWD sử dụng nền tảng dựa trên đám mây để cung cấp chức năng tương tự.

Công cụ phát triển ứng dụng đa nền tảng (CROSS-PLATFORM MOBILE APP DEVELOPMENT TOOLS)

Ngoài việc tạo ứng dụng native từ đầu bằng ngôn ngữ lập trình như Objective C hoặc Java (như đã mô tả trong Chương 3), có hàng trăm bộ công cụ phát triển ứng dụng mã nguồn mở hoặc chi phí thấp giúp việc tạo ứng dụng di động đa nền tảng trở nên tương đối dễ dàng và ít tốn kém, mà không cần sử dụng ngôn ngữ lập trình dành riêng cho thiết bị.

Ví dụ, Flutter là một bộ công cụ mã nguồn mở do Google phát triển có thể được sử dụng để tạo ứng dụng native cho thiết bị Android và iOS (cũng như ứng

dụng cho Windows, Mac và Web). React Native là một công cụ mã nguồn mở khác sử dụng React và JavaScript để xây dựng ứng dụng native cho iOS, Android, Windows và Mac. Appery.io là một nền tảng dựa trên đám mây cho phép bạn sử dụng công cụ xây dựng trực quan kéo và thả để tạo ứng dụng HTML5 sử dụng jQuery Mobile. Appery.io hỗ trợ ứng dụng Android và iOS. Codiqa là một công cụ tương tự thậm chí còn dễ sử dụng hơn. Nó cũng cung cấp giao diện kéo và thả và xây dựng ứng dụng với các thành phần HTML5 100%, không cần phải viết mã.

Đối với những người ít kỹ thuật hơn, Swiftic là một công cụ tạo ứng dụng di động miễn phí cho phép bạn thêm nhiều chức năng khác nhau, bao gồm thương mại điện tử, thông báo và nguồn cấp dữ liệu xã hội. Ở phía kỹ thuật hơn, Ionic là một framework phát triển di động mã nguồn mở khác cho phép xây dựng ứng dụng di động hybrid sử dụng HTML, CSS và JavaScript. Axway Appcelerator là một công cụ tương tự, ít kỹ thuật hơn để tạo và quản lý các ứng dụng di động hybrid.

Sự hiện di động: Cân nhắc về hiệu suất và chi phí (MOBILE PRESENCE: PERFORMANCE AND COST CONSIDERATIONS)

Nếu bạn chưa có website hiện có, quy trình hiệu quả nhất có thể là sử dụng triết lý thiết kế ưu tiên di động và tạo một trang web di động trước. Một cách khác, bạn có thể chọn xây dựng một trang web truyền thống sử dụng kỹ thuật RWD hoặc AWD. Nếu bạn đã có một trang web mà bạn không muốn phát triển lại hoàn toàn, con đường ít tốn kém nhất là điều chỉnh kích thước để tạo ra một trang web di động thân thiện với điện thoại thông minh. Làm như vậy thường sẽ không đòi hỏi một nỗ lực thiết kế lại hoàn toàn. Bạn sẽ cần giảm bớt đồ họa và văn bản, đơn giản hóa điều hướng, và tập trung vào việc cải thiện trải nghiệm khách hàng để không gây nhầm lẫn cho mọi người. Bởi vì khách hàng của bạn có thể vẫn cần sử dụng kết nối di động tương đối chậm đôi khi, bạn sẽ cần giảm bớt lượng dữ liệu bạn gửi đi. Ngoài ra, do khó khăn trong việc nhập dữ liệu khách hàng trên thiết bị di động, bạn không thể mong đợi khách hàng vui vẻ nhập các chuỗi số hoặc ký tự văn bản dài. Để rõ ràng trong tiếp thị, hãy đảm bảo hình ảnh thương hiệu được sử dụng trên trang web di động phù hợp với những hình ảnh trên trang web truyền thống. Chi phí phát triển một trang web di động có thể dao động rộng, từ hơn 1 triệu đô la cho một trang web được thiết kế tùy chỉnh cho một doanh nghiệp lớn toàn cầu đến dưới 1.000 đô la cho một doanh nghiệp nhỏ chọn một công ty như Wix hoặc MoFuse, cung cấp một mẫu hoặc công cụ tạo trang web di động.