Mục lục

[I. Giới thiệu 3](#_Toc23063698)

[1. Định nghĩa 3](#_Toc23063699)

[2. Tổng quan 3](#_Toc23063700)

[II. Framework types 4](#_Toc23063701)

[1. Multi-page applications 4](#_Toc23063702)

[2. Single-page applications 5](#_Toc23063703)

[3. Framework nào tốt hơn 5](#_Toc23063704)

[III. Ứnng dụng web và công nghệ khác 6](#_Toc23063705)

[1. Website 6](#_Toc23063706)

[2. Web application 6](#_Toc23063707)

[3. Ứng dụng thuần túy (Native app): 6](#_Toc23063708)

[4. Ứng dụng lai (Hybrid app): 7](#_Toc23063709)

[IV. Web application hoạt động như thế nào 8](#_Toc23063710)

[V. Ưu, khuyết điểm 8](#_Toc23063711)

[1. Ưu điểm 8](#_Toc23063712)

[a. Các ứng dụng web chạy trên nhiều nền tảng bất kể HĐH hay thiết bị miễn là trình duyệt tương thích 8](#_Toc23063713)

[b. Tất cả người dùng truy cập cùng một phiên bản: 8](#_Toc23063714)

[c. Chúng không được cài đặt trên ổ cứng, do đó loại bỏ giới hạn không gian: 8](#_Toc23063715)

[d. Ứng dụng web dựa trên đăng ký (tức là SaaS) giảm chi phí cho cả doanh nghiệp. 8](#_Toc23063716)

[2. Khuyết điểm 9](#_Toc23063717)

[a) Sự phụ thuộc Internet 9](#_Toc23063718)

[b) Bảo mật 9](#_Toc23063719)

[c) Giảm tốc độ 10](#_Toc23063720)

[d) Hỗ trợ trình duyệt 10](#_Toc23063721)

[VI. Phân loại 10](#_Toc23063722)

[1. ỨNG DỤNG WEB TĨNH (STATIC WEB APPLICATIONS) 10](#_Toc23063723)

[2. ỨNG DỤNG WEB NĂNG ĐỘNG(DYNAMIC WEB APPLICATIONS) 10](#_Toc23063724)

[3. MUA HÀNG TRỰC TUYẾN HOẶC THƯƠNG MẠI ĐIỆN TỬ(SHOP ONLINE OR E-COMMERCE) 11](#_Toc23063725)

[4. ỨNG DỤNG WEB PORTAL(PORTAL WEB APP) 11](#_Toc23063726)

[5. ANIMATED WEB APPLICATIONS 11](#_Toc23063727)

[6. ỨNG DỤNG WEB VỚI HỆ THỐNG QUẢN LÝ NỘI DUNG(WEB APPLICATIONS WITH A CONTENT 11](#_Toc23063728)

[VII. Progressive web apps: 12](#_Toc23063729)

**Web application**

# Giới thiệu

## Định nghĩa

Ứng dụng web chính là một phần mềm ứng dụng nền tảng web để chạy các phần mềm theo mong muốn và nhu cầu của người sử dụng. Thông qua ứng dụng web người dùng có thể thực hiện được một số công việc như sau: chia sẻ hình ảnh, mua sắm, **phần mềm tính tiền** thuận tiện cho người bán…vì web app có tính tương tác cao hơn website rất nhiều.

Đối với một số người dùng không rành về công nghệ thông tin, nên họ cứ nghĩ rằng những thứ online vào được bằng trình duyệt web thì đều được hiểu là website cả. Chính vì thế, người dùng thường yêu cầu: website bán hàng, website quản lý siêu thị,… nhưng thực chất thì chúng đều là [**web application**](https://mona.media/dich-vu-lap-trinh-web-app/)**.**

## Tổng quan

Các doanh nghiệp sử dụng Internet để phát sóng thương hiệu, sản phẩm và dịch vụ của mình ra thế giới. Họ liên tục khởi động và cập nhật sản phẩm web mới bởi vì Internet đang phát triển với một tốc độ đáng kinh ngạc trong năm này qua năm khác. Mặc dù quyết tâm theo kịp các xu hướng mới và sắp tới và cạnh tranh để giành được khách hàng, luôn có một nỗ lực để mở rộng các chức năng thương mại điện tử và dịch vụ đám mây của họ. Khi nhu cầu thay đổi thúc đẩy nhu cầu về các tính năng mới khiến khách hàng dễ dàng tận hưởng, các trang web đã phát triển phức tạp để đáp ứng những thách thức này. Những người đầu tiên như eBay và Amazon đã thành công không chỉ bởi vì họ có đội ngũ tiếp thị mạnh mẽ. Các công ty khác cũng có đội ngũ tiếp thị mạnh mẽ. Họ tận dụng thị trường phát triển nhanh chóng với các **ứng dụng web có tính ứng dụng cao(highly functional web applications)**.

Họ đã lấy các khái niệm đơn giản, chẳng hạn như mua, hàng tồn kho và giao dịch, và tạo ra các thị trường web có thể mở rộng đang chiếm ưu thế ngày nay. Các ứng dụng web phục vụ các nhu cầu khác nhau, chẳng hạn như dịch vụ email đám mây Hotmail hoặc Gmail, tận dụng các khả năng của trình duyệt web để tạo các ứng dụng email không yêu cầu cài đặt phần mềm. Tính di động này là một lý do khiến họ phát triển thành ngày nay. Xác thực người dùng, hồ sơ, theo dõi dữ liệu thời gian thực, nhắn tin, thanh toán trực tuyến, trạng thái ứng dụng - đây là một số tính năng phổ biến mà bạn thấy ngày nay. Việc phát hành một ứng dụng web thành công sẽ khó hơn nếu thiếu quá nhiều tính năng mà các ứng dụng web khác thường có.

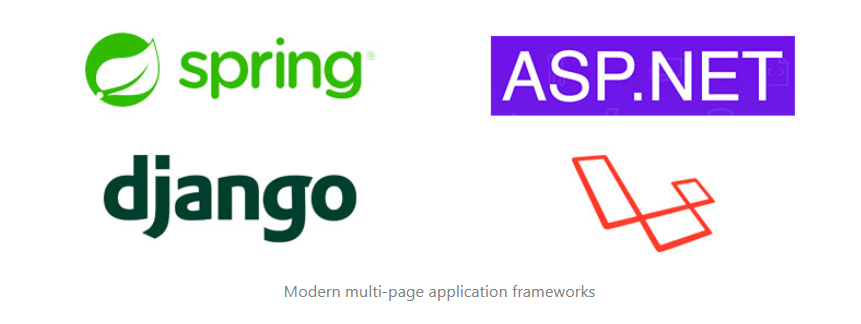
Điều quan trọng là tập trung vào các tính năng và chức năng cốt lõi khi phát triển ứng dụng web, nhưng chỉ tập trung vào những điều cốt lõi đó là không đủ. Một lượng nỗ lực tương đương nên được dành cho các tính năng khác để bổ sung cho chức năng cốt lõi để khiến người dùng yêu thích sản phẩm của bạn. Các khung ứng dụng web tiếp tục phổ biến khi các công ty mới tham gia và tạo các ứng dụng của riêng họ để giành thị phần. Các khung cung cấp một cơ sở cho hầu hết các chức năng cốt lõi mà mọi ứng dụng web yêu cầu. Hầu như tất cả các ứng dụng web mới đều sử dụng ít nhất một khung để đơn giản hóa sự phát triển và theo kịp sự thay đổi.

# Framework types

Các loại khung Có nhiều khung lập trình có sẵn để phát triển các ứng dụng web từ - một số được liệt kê trong các phần bên dưới. Chúng được chia thành 2 loại chính: ứng dụng nhiều trang (MPA) và ứng dụng một trang (SPA).

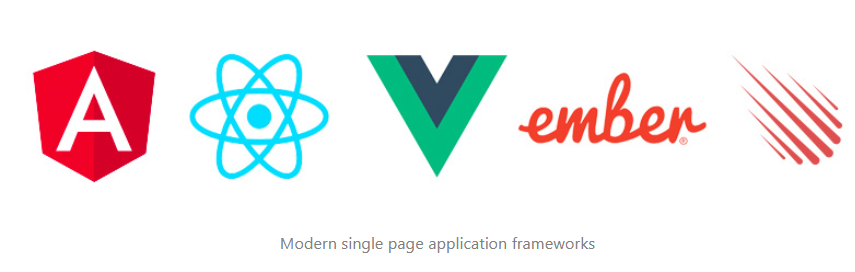
## Multi-page applications

Các ứng dụng nhiều trang(Multi-page applications) xuất hiện đầu tiên và sử dụng các ngôn ngữ lập trình truyền thống như PHP, .NET và Java. MPA là các ứng dụng chạy trên máy chủ và có thể giao tiếp trực tiếp với cơ sở dữ liệu.

Các khung MPA phổ biến bao gồm Java Spring, Python Django, ASP.NET MVC / Razor và PHP Laravel. Khi không phục vụ các trang, họ cũng có thể đóng vai trò là các back-end API để các khung công tác SPA tiêu thụ.

## Single-page applications

Các ứng dụng một trang (Single-page applications) đã đạt được lực kéo khi hỗ trợ JavaScript trên nhiều trình duyệt được cải thiện. Điều này cũng xảy ra xung quanh khi việc sử dụng một trình cắm phổ biến của trình duyệt, Adobe Flash, mất đi sự ưu ái và không còn nữa. Không giống như các MPA sống trên máy chủ, các SPA được tải xuống trình duyệt máy khách một lần và trang không cần phải tải lại. Với một SPA, bạn vẫn cần một máy chủ phụ trợ để cung cấp nội dung từ các API, nhưng có sự tách biệt lớn hơn giữa cả hai lớp. Mỗi lớp giao diện người dùng và lớp máy chủ phía sau có thể được hoán đổi mà không ảnh hưởng đến nhau miễn là cả hai đều sử dụng cùng một hợp đồng API.

Các khung SPA phổ biến nhất hiện nay là Angular, React và Vue. Các SPA khác bao gồm Ember và Meteor. Không có người chiến thắng rõ ràng về việc khung SPA nào tốt hơn, nhưng các yếu tố quan trọng bao gồm thiết kế hệ thống và sự thoải mái và trải nghiệm của nhà phát triển với khung.

## Framework nào tốt hơn

Chúng ta không có cái tốt nhất chỉ có cái nào phù hợp.

Nếu bạn muốn ứng dụng web của mình hoạt động tương tự như ứng dụng trên máy tính để bàn hoặc thiết bị di động, có điều hướng nhanh giữa các trang, chuyển tiếp liền mạch, thì đó là một ý tưởng tuyệt vời để triển khai SPA.

Mặt khác, nếu bạn yêu cầu cơ sở dữ liệu nhưng có một vài tương tác giữa người dùng và ứng dụng, thì việc phát triển và triển khai bằng MPA có thể dễ dàng hơn.

 Hầu hết các công ty đã áp dụng các SPA hoặc đang trong quá trình chuyển đổi ứng dụng web của họ sang một SPA vì các công ty muốn xây dựng sản phẩm tốt nhất và cung cấp trải nghiệm người dùng tốt nhất sẽ khiến khách hàng gắn bó với họ.

 Năm 2019, trải nghiệm chung cho người dùng là các ứng dụng web nhanh hơn đáng kể so với trước đây. Bây giờ rất hiếm khi thấy các doanh nghiệp lớn hoặc các công ty khởi nghiệp xây dựng các ứng dụng web của họ dưới dạng MPA.

# Ứnng dụng web và công nghệ khác

## Website

Một website thông thường chỉ bao gồm những trang riêng được thiết kế để hiển thị thông tin với người dùng(webpage), những thông tin này đa phần là văn bản và hình ảnh và không hỗ trợ tương tác với người dùng cao.

Ví dụ:

<http://www.visithumboldt.com/>

<http://pierrehermenicolasbuffe.com/en>

## Web application

Ngược lại Web app là một sự kết hợp những website và ứng dụng, nói dễ hiểu là một phần mềm thay vì cài đặt trên máy tính thì nay chúng lại hoạt động trên một giao diện website, nghĩa là khi bạn đăng nhập vào web, bạn có thể quản trị kho hàng, nhân viên, sản phẩm và nhiều tính năng tương tự như một phần mềm máy tính thông thường.

Ví dụ:

Google Apps for Work has Gmail, Google Docs, Google Sheets, Google Slides, GSuite,Tiki…

## Ứng dụng thuần túy (Native app):

Native app là ứng dụng được thiết kế riêng cho một hệ điều hành di động. App loại này không thể chạy trên các nền tảng khác, chẳng hạn như một ứng dụng native iOS sẽ được phát triển từ ngôn ngữ lập trình Swift và không thể chạy trên điện thoại Android.

* **Ưu điểm:**

Lợi thế lớn nhất của native app là chúng cho phép người dùng truy cập và sử dụng tất cả các tính năng một cách dễ dàng và sẽ ít khi gặp lỗi nếu được thiết kế một cách hoàn chỉnh. Native app cũng mang đến trải nghiệm tốc độ cao và hoàn chỉnh nhất nên hiện vẫn chiếm phần lớn tổng lượng ứng dụng hiện nay.

* **Khuyết điểm:**

Tuy nhiên, cũng chính vì không thể chạy trên các hệ điều hành khác nên nhà phát triển muốn phủ sóng ứng dụng của mình trên nhiều nền tảng buộc phải code lại nó rất nhiều lần. Ví dụ như việc các startup phải phát triển ứng dụng của mình tới 3 lần cho các nền tảng phổ biến như iOS, Android và Windows là không hiếm khi xảy ra. Quá trình không chỉ đòi hỏi nguồn nhân lực tương đối mà còn cả thời gian, công sức và chi phí. Do vậy, nhiều công ty chưa có nguồn lực dồi dào chọn cách phát triển app cho riêng 1 nền tảng trước. Nếu ứng dụng thành công trên nền tảng này, họ sẽ quay lại phát triển nó cho nền tảng khác.

## Ứng dụng lai (Hybrid app):

Đúng như tên gọi, hybrid app là loại ứng dụng được thiết kế để chạy được trên nhiều nền tảng khác nhau, nó là sự kết hợp của web app trong vỏ của native app. Chúng được phát triển dựa trên một ngôn ngữ lập trình tiêu chuẩn (chẳng hạn nhưu C# kết hợp với HTML5 và Javascript). Các tương tác của người dùng trên app loại này cũng thường nằm dưới sự quản lý của các plugin trong hệ điều hành và có thể truy cập vào những hardware features mà web app không có quyền.

* **Ưu điểm:**

Lợi thế lớn nhất của app Hybrid là chúng có thể hỗ trợ đa nền tảng và giúp các công ty tiết kiệm chi phí code nhiều lần. Về phía người dùng, miễn là ứng dụng được thiết kế tốt, thì họ cũng khó lòng phân biệt được đâu là native đâu là hybrid app. Với những framework hỗ trợ hư Ionic, Phonegap,… ngày càng được nâng cấp, rất nhiều startup nổi tiếng đã chuyển sang hybrid app, tiêu biểu là Uber, Evernote, Instagram, Gmail,…

* **Khuyết điểm:**

Tuy nhiên, hybrid cũng không phải là không có nhược điểm vì nó là kết hợp giữa 2 loại nên khó khăn sẽ khởi nguồn từ 2 platforms. Đối với ứng dụng đòi hỏi những tương tác phức tạp thì hybrid app sẽ là bài toán khó cho các nhà phát triển. Hybrid có nhiều hạn chế khi lập trình viên muốn phát triển app của mình lên mức cao và tinh tế hơn. Chi phí duy trì ứng dụng hybrid cũng thường lớn hơn chi phí cho các ứng dụng native.

# Web application hoạt động như thế nào

Các ứng dụng web thường được mã hóa bằng ngôn ngữ hỗ trợ trình duyệt như JavaScript và HTML vì các ngôn ngữ này dựa vào trình duyệt để hiển thị chương trình thực thi. Một số ứng dụng là động, yêu cầu xử lý phía máy chủ. Những người khác hoàn toàn tĩnh mà không cần xử lý tại máy chủ. Ứng dụng web yêu cầu máy chủ web quản lý các yêu cầu từ máy khách, máy chủ ứng dụng để thực hiện các tác vụ được yêu cầu và đôi khi là cơ sở dữ liệu để lưu trữ thông tin. Công nghệ máy chủ ứng dụng bao gồm từ ASP.NET, ASP và ColdFusion, đến PHP và JSP. Đây là một luồng ứng dụng web thông thường trông như thế nào:

1. Người dùng kích hoạt yêu cầu đến máy chủ web qua Internet, thông qua trình duyệt web hoặc giao diện người dùng của ứng dụng
2. Máy chủ web chuyển tiếp yêu cầu này đến máy chủ ứng dụng web thích hợp
3. Máy chủ ứng dụng web thực hiện tác vụ được yêu cầu - chẳng hạn như truy vấn cơ sở dữ liệu hoặc xử lý dữ liệu - sau đó tạo kết quả của dữ liệu được yêu cầu
4. Máy chủ ứng dụng web gửi kết quả đến máy chủ web với thông tin được yêu cầu hoặc dữ liệu được xử lý
5. Máy chủ web phản hồi lại máy khách với thông tin được yêu cầu xuất hiện trên đó màn hình của người dùng

# Ưu, khuyết điểm

## Ưu điểm

### Các ứng dụng web chạy trên nhiều nền tảng bất kể HĐH hay thiết bị miễn là trình duyệt tương thích

### Tất cả người dùng truy cập cùng một phiên bản:

Web app sẽ được cập nhật và đồng bộ (không như native app lúc tải về cần update) nên khi truy cập mọi người đều sử dụng được phiên bản mới nhất.

### Chúng không được cài đặt trên ổ cứng, do đó loại bỏ giới hạn không gian:

Chỉ cần trình duyệt để truy cập theo URL là đủ. Đồng thời dễ chia sẻ qua link và là lợi thế kinh doanh vì sự tiên lợi.

### Ứng dụng web dựa trên đăng ký (tức là SaaS) giảm chi phí cho cả doanh nghiệp.

SaaS(Software as a Service) cho phép người mua trả một khoản phí thuê bao hàng năm hoặc hàng tháng, chi phí đã bao gồm giấy phép sử dụng phần mềm, phí hỗ trợ và hầu hết các chi phí khác. Lợi ích chính của SaaS là có thể linh hoạt mở rộng quy mô dịch vụ để phù hợp với nhu cầu của doanh nghiệp theo thời gian, từ đó giúp tiết kiệm chi phí. Doanh nghiệp không phải chịu các trách nhiệm kỹ thuật như khắc phục sự cố và bảo trì hệ thống. Các nhà cung cấp dịch vụ SaaS sẽ lo việc đó.

## Khuyết điểm

### Sự phụ thuộc Internet

Trong khi truy cập internet 4G & Wi-Fi có sẵn ở nhiều địa điểm, nếu bạn mất kết nối, bạn sẽ không thể truy cập ứng dụng web của mình.

Ví dụ như các web app đặt hàng cần internet để gửi dữ liệu. Hiện giờ cũng đã có cách khắc phục để sử dụng offline nhưng vẫn còn hạn chế ở phạm vi những web app lưu trữ data văn phòng, ghi chú, cung cấp thông tin.

### Bảo mật

Mặc dù nhiều người kinh doanh có thể tin rằng dữ liệu kém an toàn hơn trong môi trường đám mây, chúng tôi xin phép khác biệt Có một số cách giúp bạn giảm rủi ro vi phạm dữ liệu, chẳng hạn như thực thi SSL để truy cập HTTPS an toàn vào ứng dụng của bạn. Web applications cũng gặp phải một số security risks theo top 10 của tiêu chuẩn OWASP (Open Web Application Security Project) như:

* Injection: là lỗ hổng xảy ra do sự thiếu sót trong việc lọc các dữ liệu đầu vào không đáng tin cậy kẻ tấn công có thể chèn các đoạn mã độc để gây ra lộ lọt dữ liệu và chiếm quyền kiểm soát trình duyệt của khách hàng.
* 2Cross-Site Scripting (XSS): attacker chèn đoạn mã Javascript vào ứng dụng web. Khi đầu vào này không được lọc, chúng sẽ được thực thi mã độc trên trình duyệt của người dùng. Kẻ tấn công có thể lấy được cookie của người dùng trên hệ thông hoặc lừa người dùng đến các trang web độc hại.
* Broken Authentication and Session Management: những rủi ro gặp phải trong quá trình xác thực.
* Insecure Direct Object References: Lỗ hổng này xảy ra khi chương trình cho phép người dùng truy cập các tài nguyên mà không thực hiện quá trình kiểm soát quyền hạn (hoặc quá trình này không hoàn chỉnh) kẻ tấn công có thể truy cập một cách bất hợp pháp vào các dữ liệu nhạy cảm, quan trọng trên máy chủ.
* Cross-Site Request Forgery (CSRF): Trình duyệt bị đánh lừa bởi một số bên thứ ba lạm dụng quyền hạn.
* Security Misconfiguration: máy chủ website và các ứng dụng bị cấu hình sai.
* Insecure Cryptographic Storage
* Failure to Restrict URL Access: sai sót và bị lạm dụng trong vấn đề phân quyền.
* Using Components with Known Vulnerabilities: sử dụng thư viện bị lỗi sẵn có thể dẫn đến security breaches.
* Unvalidated Redirects and Forwards: vấn đề về lọc đầu vào và chuyển hướng, người dùng có thể bị đánh lừa chuyển hướng sang web độc hại.

### Giảm tốc độ

Có thể một ứng dụng web sẽ hoạt động với tốc độ chậm hơn một chút so với tốc độ được lưu trữ trên máy chủ cục bộ. Một số nguyên nhân gây ra vấn đề về performance của web apps là:

* Vấn đề DNS và connection.
* Server chậm.
* Code không hiệu quả.
* Thiếu sự phân chia workload khi có nhiều requests => thời gian response chậm.
* Traffic spikes.
* HTML title tags thiếu chi tiết.
* Failing to optimize bandwidth usage.

### Hỗ trợ trình duyệt

* Khi không sử dụng cùng một trình duyệt. Điều này có nghĩa là trong quá trình phát triển, bạn sẽ cần đảm bảo ứng dụng của mình được hỗ trợ trên nhiều trình duyệt khác nhau.

# Phân loại

## ỨNG DỤNG WEB TĨNH (STATIC WEB APPLICATIONS)

Nếu bạn chọn tạo một ứng dụng web tĩnh, điều đầu tiên cần biết là loại ứng dụng web này hiển thị rất ít nội dung và không đặc biệt linh hoạt. Các ứng dụng web tĩnh thường được phát triển bằng HTML và CSS nhưng đây không phải là nền tảng duy nhất để phát triển ứng dụng tĩnh; bạn có thể sử dụng jQuery và Ajax theo sự thuận tiện của riêng bạn. Bạn cũng có thể bao gồm hoặc hiển thị các đối tượng hoạt hình, chẳng hạn như biểu ngữ, GIF, video, v.v. trong ứng dụng tĩnh. Thật không may, sửa đổi nội dung của các ứng dụng web tĩnh là không dễ dàng. Để làm điều này, trước tiên bạn phải tải xuống mã HTML, sau đó sửa đổi nó và cuối cùng gửi lại cho máy chủ. Những thay đổi này chỉ có thể được thực hiện bởi quản trị trang web hoặc công ty phát triển đã lên kế hoạch và thiết kế Ứng dụng đầu tiên. Ví dụ về sự phát triển của các ứng dụng web tĩnh bao gồm danh mục đầu tư chuyên nghiệp hoặc sơ yếu lý lịch kỹ thuật số. Tương tự, một trang giới thiệu một công ty cũng có thể sử dụng loại ứng dụng web này để hiển thị thông tin liên hệ của họ.

## ỨNG DỤNG WEB NĂNG ĐỘNG(DYNAMIC WEB APPLICATIONS)

Các ứng dụng web động phức tạp hơn nhiều ở cấp độ kỹ thuật. Họ sử dụng cơ sở dữ liệu để tải dữ liệu và nội dung của nó được cập nhật mỗi khi người dùng truy cập chúng. Họ thường có một bảng quản trị (được gọi là CMS), nơi quản trị viên có thể sửa hoặc sửa đổi nội dung ứng dụng, cho dù là văn bản hoặc hình ảnh. Các ngôn ngữ lập trình khác nhau có thể được sử dụng để phát triển các ứng dụng web động. PHP và ASP là các ngôn ngữ phổ biến nhất được sử dụng cho mục đích này vì chúng cho phép cấu trúc nội dung. Trong loại ứng dụng này, việc cập nhật nội dung rất đơn giản và máy chủ thậm chí không phải truy cập để sửa đổi được thực hiện. Ngoài ra, nó cho phép bạn triển khai rất nhiều tính năng, chẳng hạn như diễn đàn hoặc cơ sở dữ liệu. Dự án - ngoài nội dung - có thể được sửa đổi để phù hợp với sở thích của quản trị viên.

## MUA HÀNG TRỰC TUYẾN HOẶC THƯƠNG MẠI ĐIỆN TỬ(SHOP ONLINE OR E-COMMERCE)

Nếu ứng dụng web là một cửa hàng trực tuyến, sự phát triển của nó có thể giống với trang web thương mại điện tử hoặc thương mại điện tử. Loại quy trình phát triển ứng dụng này phức tạp hơn vì nó phải cho phép thanh toán điện tử có thể được thực hiện từ thẻ tín dụng, PayPal hoặc các phương thức thanh toán khác. Nhà phát triển cũng phải tạo một bảng quản lý cho quản trị viên; sẽ được sử dụng để liệt kê các sản phẩm mới, cập nhật chúng, xóa các mục và để quản lý các ứng dụng và thanh toán. Ứng dụng web của bạn phù hợp với các thiết bị di động giống như một ứng dụng di động, giúp nó có thể tương tác với nó như thể nó là một ứng dụng gốc.

## ỨNG DỤNG WEB PORTAL(PORTAL WEB APP)

Thông qua cổng thông tin, chúng tôi đề cập đến một loại ứng dụng truy cập vào các phần hoặc danh mục khác nhau thông qua một trang chủ. Các ứng dụng này có thể bao gồm nhiều thứ: diễn đàn, trò chuyện, e-mail, trình duyệt, các khu vực được truy cập thông qua đăng ký, nội dung mới nhất, v.v.

## ANIMATED WEB APPLICATIONS

Hoạt hình chắc chắn được liên kết với công nghệ Flash. Phương pháp lập trình này cho phép bạn hiển thị nội dung với các hiệu ứng hoạt hình. Loại ứng dụng này cho phép các thiết kế sáng tạo và hiện đại hơn và là một trong những công nghệ chính được sử dụng bởi các nhà thiết kế và giám đốc sáng tạo. Nhược điểm cố hữu trong việc phát triển các ứng dụng web hoạt hình là loại công nghệ này không phù hợp với mục đích định vị web và tối ưu hóa SEO vì các công cụ tìm kiếm không thể đọc đúng thông tin mà chúng chứa.

1. ỨNG DỤNG WEB VỚI HỆ THỐNG QUẢN LÝ NỘI DUNG(WEB APPLICATIONS WITH A CONTENT MANAGEMENT SYSTEM)

Nội dung cần được cập nhật liên tục khi phát triển ứng dụng web để việc cài đặt hệ thống quản lý nội dung (CMS) là một lựa chọn nghiêm túc cần xem xét. Quản trị viên có thể sử dụng CMS này để thực hiện các thay đổi và cập nhật một mình. Những người quản lý nội dung là trực quan và rất dễ xử lý. Một số ví dụ về hệ thống quản lý nội dung là: WordPress: chắc chắn là phổ biến nhất xung quanh việc quản lý nội dung. Có rất nhiều thông tin, hướng dẫn và hướng dẫn có sẵn trên internet sẽ giúp bạn tùy chỉnh nó và hiểu cách thức hoạt động của nó. Ngoài ra, nó còn miễn phí. Joomla: CMS này chỉ đứng thứ hai sau WordPress. Nó không có nhiều người dùng như cuối cùng, nhưng có một cộng đồng mạnh mẽ và cũng rất trực quan. Drupal: nó là một phần mềm CSM miễn phí - rất dễ thích nghi và đặc biệt được khuyên dùng để xây dựng cộng đồng. Loại ứng dụng web này rất phổ biến trong số các trang nội dung: blog cá nhân, blog công ty, blog chuyên nghiệp, trang tin tức, bài viết, phương tiện truyền thông, v.v.

# Progressive web apps:

PWAs là web applications sử dụng công nghệ web hiện đại để cho tối ưu hóa user experience cho ngườ dùng mặc dù chạy trên browser. Đây là lựa chọn thích hợp cho những doanh nghiệp không đủ kinh phí cho native mobile apps.

Theo thống kê thì 87% thị trường digital phụ thuộc vào web app và PWA là một phiên bản tốt hơn của web app nên nó có tiềm năng lợi nhuận đáng chú ý.

* **Ưu điểm:**
* Native Experience: PWAs sử dụng web technologies (HTML, CSS, Javascript) như web app thông thường nhưng nó cho user experience mượt mà và tiện lợi như một native app. Đồng thời, PWAs hiện giờ cũng có access tới một số features như push notifications, camera... để hỗ trợ tiện ích sử dụng.
* Ease of access: Không cần download về máy vì PWAs chạy trên browser, có thể dễ dàng share và truy cập bằng link. Sử dụng PWA khi add vào homescreen thì không khác gì native app mà gọn nhẹ, đông thời khuyến khích tăng interaction của người dùng với web app hơn.
* Faster service: PWA cache data và load text, images và những thành phần nội dung trước khi load hoàn thiện-> giảm thời gian chờ đợi vì theo số liệu 40% người dùng sẽ tắt một website khi nó load >3s => tăng engagement với user.
* Updated data access: một điểm cộng của PWAs là không cần updates vì nó tự động update chính nó thông qua bản mới nhất trên server => tiện lợi dẫn đến thêm user engagement.
* Low development cost: không cần submission ở appstore => cho những brands có thể gia nhập thị trường và nhận được ROI(return on investment) cao hơn.
* Works offline: sử dụng ServiceWorkers với kết nối HTTPS cho PWAs hoạt động được offline và secure.
* Lower memory consumption than native app: các elements nằm trên server và không phải trên user device (trừ cache) nên không đi quá giới hạn của localstorage của một web (5MB) trên device.
* **Khuyết điểm:**
* Còn là công nghệ mới nên không được supported trên tất cả browsers
* Hạn chế về hardware access so với native app (alarms, telephony features…).
* **Ví dụ của PWAs:**
* **Trivago:**

Trivago là app book và so sánh giá khách sạn có bao gồm native app cho android và ios. Nhưng nhờ vào PWAs thì tăng được lượng clickthrough.

* **Pinterest PWA:**

Pinterest là cộng đồng tìm kiếm và bookmark hình ảnh, ý tưởng... với PWA, pinterest đã tăng được conversion rates và những lợi ích:

* Time spend by the users within the app has increased by 40%
* Ad revenue has increase by 44%
* Core engagement has increased by 60%
* **Starbucks PWA:**

Starbucks, chuỗi café nổi tiếng là dùng PWA để cho khách hàng trải nghiệm offline. Tính năng như xem menu, order customization, and cart management đều available offline.

Starbucks tạo nên user experience nhanh và đáng tin cậy và có kết quả gần như double lượng users ở cả websites và trên mobile app.

* **Twitter Lite:**

Với mục đích load nhanh nên Twitter áp dụng PWA cho ra Twitter Lite, có thể dễ dàng add vào homescreen , chức năng tương tự như twitter app với trải nghiệm nhanh hơn, chứng kiến thay đổi:

* 65% increase in pages per session
* 75% increase in Tweets sent
* 20% decrease in bounce rate



Nhiều brands trên thế giới sử dụng PWAs…