**CHƯƠNG 2: NỀN TẢNG CÔNG NGHỆ**

|  |
| --- |
| **Nội dung chương này trình bày những vấn đề sau:**   * Giới thiệu về Angular 2 * Ứng dụng đa nền tảng và Ionic 2 * Firebase |

**2.1 Giới thiệu về Angular 2**

**2.1.1 Angular JS**

AngularJS được bắt đầu từ năm 2009, do lập trình viên Misko Hevery tại Google viết ra như là một dự án kiểu “viết cho vui”. Misko và nhóm lúc này đang tham gia vào 1 dự án của Google tên là Google Feedback. Với AngularJS, Misko đã rút ngắn số dòng code front-end từ 17000 dòng còn chỉ khoảng 1500. Với sự thành công đó, đội ngũ của dự án Google Feedback quyết định phát triển AngularJS theo hướng mã nguồn mở. Hiện tại dự án AngularJS đang có gần 11000 người theo dõi và hơn 28000 lượt fork

AngularJS là một framework có cấu trúc cho các ứng dụng web động. Nó cho phép sử dụng HTML như là ngôn ngữ mẫu và cho phép mở rộng cú pháp của HTML để diễn đạt các thành phần ứng dụng của bạn một cách rõ ràng và súc tích. Hai tính năng cốt lõi: Data binding và Dependency injection của AngularJS loại bỏ phần lớn code thường phải viết.

Ưu điểm

* Cung cấp khả năng tạo ra các Single Page Aplication dễ dàng.
* Cung cấp khả năng data binding tới HTML, khiến cho người dùng cảm giác linh hoạt, thân thiện.
* Dễ dàng tái sử dụng component
* Giúp lập trình viên viết code ít hơn với nhiều chức năng hơn.
* Chạy được trên các loại trình duyệt, trên cả PC lẫn mobile.

Nhược điểm

* Không an toàn : được phát triển từ javascript cho nên ứng dụng được viết bởi AngularJS không an toàn. Nên có sự bảo mật và xác thực phía server sẽ giúp ứng dụng trở nên an toàn hơn.
* Nếu người sử dụng ứng dụng của vô hiệu hóa JavaScript thì sẽ chỉ nhìn thấy trang cơ bản.

**2.1.2 TypeScript**

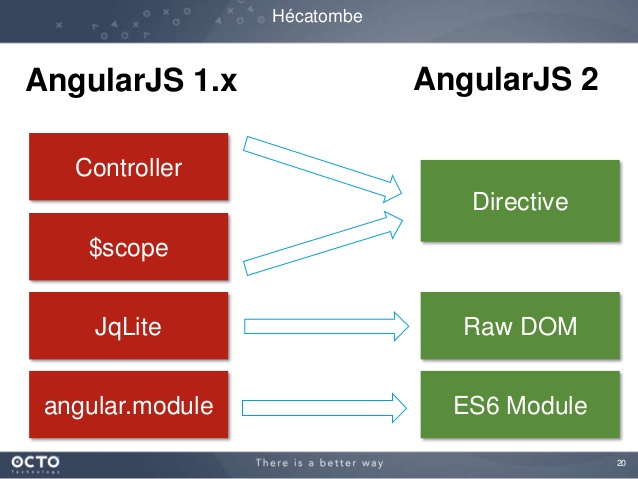
TypeScript là một dự án mã nguồn mở được phát triển bởi Microsoft, nó có thể được coi là một phiên bản nâng cao của Javascript bởi việc bổ sung tùy chọn kiểu tĩnh và lớp hướng đối tượng mà điều này không có ở Javascript. TypeScript có thể sử dụng để phát triển các ứng dụng chạy ở client-side (Angular2) và server-side (NodeJS).

TypeScript sử dụng tất cả các tính năng của của ECMAScript 2015 (ES6) như classes, modules. Không dừng lại ở đó nếu như ECMAScript 2017 ra đời nhiều khả năng TypeScript cũng sẽ nâng cấp phiên bản của mình lên để sử dụng mọi kỹ thuật mới nhất từ ECMAScript.

Các ưu điểm nổi bật của TypeScript

* **Dễ phát triển dự án lớn**: Với việc sử dụng các kỹ thuật mới nhất và lập trình hướng đối tượng nên TypeScript giúp chúng ta phát triển các dự án lớn một cách dễ dàng.
* **Nhiều Framework lựa chọn**: Hiện nay các Javascript Framework đã dần khuyến khích nên sử dụng TypeScript để phát triển, ví dụ như AngularJS 2.0 và Ionic 2.0.
* **Hỗ trợ các tính năng của Javascript phiên bản mới nhất**: TypeScript luôn đảm bảo việc sử dụng đầy đủ các kỹ thuật mới nhất của Javascript, ví dụ như version hiện tại là ECMAScript 2015 (ES6).
* **Là mã nguồn mở**: TypeScript là một mã nguồn mở nên bạn hoàn toàn có thể sử dụng mà không mất phí, bên cạnh đó còn được cộng đồng hỗ trợ.
* **TypeScript là Javscript**: Bản chất của TypeScript là biên dịch tạo ra các đoạn mã javascript nên ban có thê chạy bất kì ở đâu miễn ở đó có hỗ trợ biên dịch Javascript. Ngoài ra bạn có thể sử dụng trộn lẫn cú pháp của Javascript vào bên trong TypeScript, điều này giúp các lập trình viên tiếp cận TypeScript dễ dàng hơn.

**2.1.3 Angular 2 và những điểm khác biệt so với Angular 1**



Những thay đổi trong Angular 2

#### Module

Khi sử dụng Angular 2.0, các nhà phát triển xây dựng ứng dụng bao gồm các module có chứa các thành phần, dịch vụ,… Không giống như các module trong AngularJS phiên bản trước, các module trong angular 2 được sử dụng định dạng ES6. Các câu lệnh nhập cho phép người dùng truy cập các phần cốt lõi trong khuôn khổ cũng như các thành phần tùy chỉnh (AngularJS filter).

* Components

Trong Angular 2, việc sử dụng các components đã loại bỏ sự cần thiết của một số bộ điều khiển và những dòng lệnh gọi nó, do đó giải quyết được một số vấn đề mà các nhà phát triển phải đối mặt trong AngularJS phiên bản trước. Ngoài ra, các components đã được đóng gói lại, có nghĩa là chúng ta không phải lo lắng về các mã lệnh trong những thành phần khi ta can thiệp vào những thành phần khác có trên trang, miễn là ta đang sử dụng chúng một cách chính xác.

#### Template Syntax & Data-binding Binding

Khi làm việc với Angular 2, các nhà phát triển sử dụng luồng dữ liệu đơn hướng để chuyển đổi dữ liệu trong suốt quá trình của ứng dụng. Trường hợp ngoại lệ đối với trường hợp này là **ngModel** trong Angular 2, ta có thể cung cấp hai cách ràng buộc dữ liệu khi cần thiết bằng cách sử dụng cú pháp: **[{ngModel}]**. Bạn có thể thấy cả dấu ngoặc đơn “( )” và dấu ngoặc vuông “[ ]” trong cú pháp này, cả hai đều là sự cải tiến mới của cú pháp trong Angular 2

Những ưu điểm của Angular 2 so với Angular 1

* Thứ nhất, angular 2 có thể build trên bất kì nền tảng nào, **cross-flatform**
* Tiếp theo Angular js 2 còn khắc phục những thiếu sót, hạn chế của version trước đó, như cải thiện về performance, tối ưu hóa cho SEO, có tính Productivity cao
* Angular 2 đã lược bỏ controller, $scope thay bằng các Component
* Các Service có thể build được bằng cách tạo các class dựa trên nền tảng ES6 so với kiểu function của Angular 1
* Không những vậy, Angular 2 còn được xây dựng trên nền tảng Typescript - hỗ trợ OOP rất tốt cho Javascript
* Và còn nhiều nét ưu việt nữa. Mình chỉ tóm gọn các đặc điểm vượt trội như thế thôi. Còn những nét khác các bạn có thể google thêm nhé

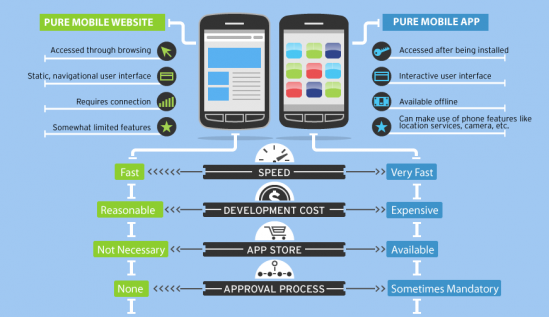
**2.2 Ứng dụng đa nền tảng và Ionic 2**

**2.2.1 Hybrid Mobile App**

Hybrid mobile apps giống bất kỳ những app khác trên điện thoại di động. Chúng có thể cài đặt được trên điện thoại và có thể tìm chúng trên các chợ ứng dụng. Với những ứng dụng này, chúng ta có thể chơi game, kết nối bạn bè qua các mạng xã hội, chụp ảnh, theo dõi sức khoẻ, và nhiều hơn nữa.

Giống như những trang web khác trên internet, hybrid mobile apps cũng được xây dựng bằng sự kết hợp của các công nghệ web như HTML, CSS và Javascript. Chỉ khác là các ứng dụng hybrid mobile được lưu trữ bên trong một ứng dụng native sử dụng WebView. (Có thể tưởng tượng rằng một WebView như là một cửa sổ trình duyệt được cấu hình chạy fullscreen bên trong một ứng dụng) Chính nhờ điều này, nên chúng có thể truy cập được vào các chức năng dựa trên phần cứng của điện thoại như là cảm biến gia tốc, camera, contacts, và nhiều hơn nữa. Đây là những khả năng thường bị hạn chế khi truy cập trên các trình duyệt di động. Hơn nữa, những ứng dụng hybrid mobile có thể còn bao gồm các thành phần Native UI trong những trường hợp cần thiết.

Các ứng dụng hybrid mobile được xây dựng tương tự như các trang web. Chúng đều là sự kết hợp của các công nghệ thiết kế web như HTML, CSS và Javascript. Tuy nhiên, thay vì hướng tới các trình duyệt di động, những ứng dụng hybrid mobile hướng tới các WebView nằm bên trong những ứng dụng native. Điều này giúp cho các ứng dụng có thể truy cập các chức năng dựa trên phần cứng của điện thoại.



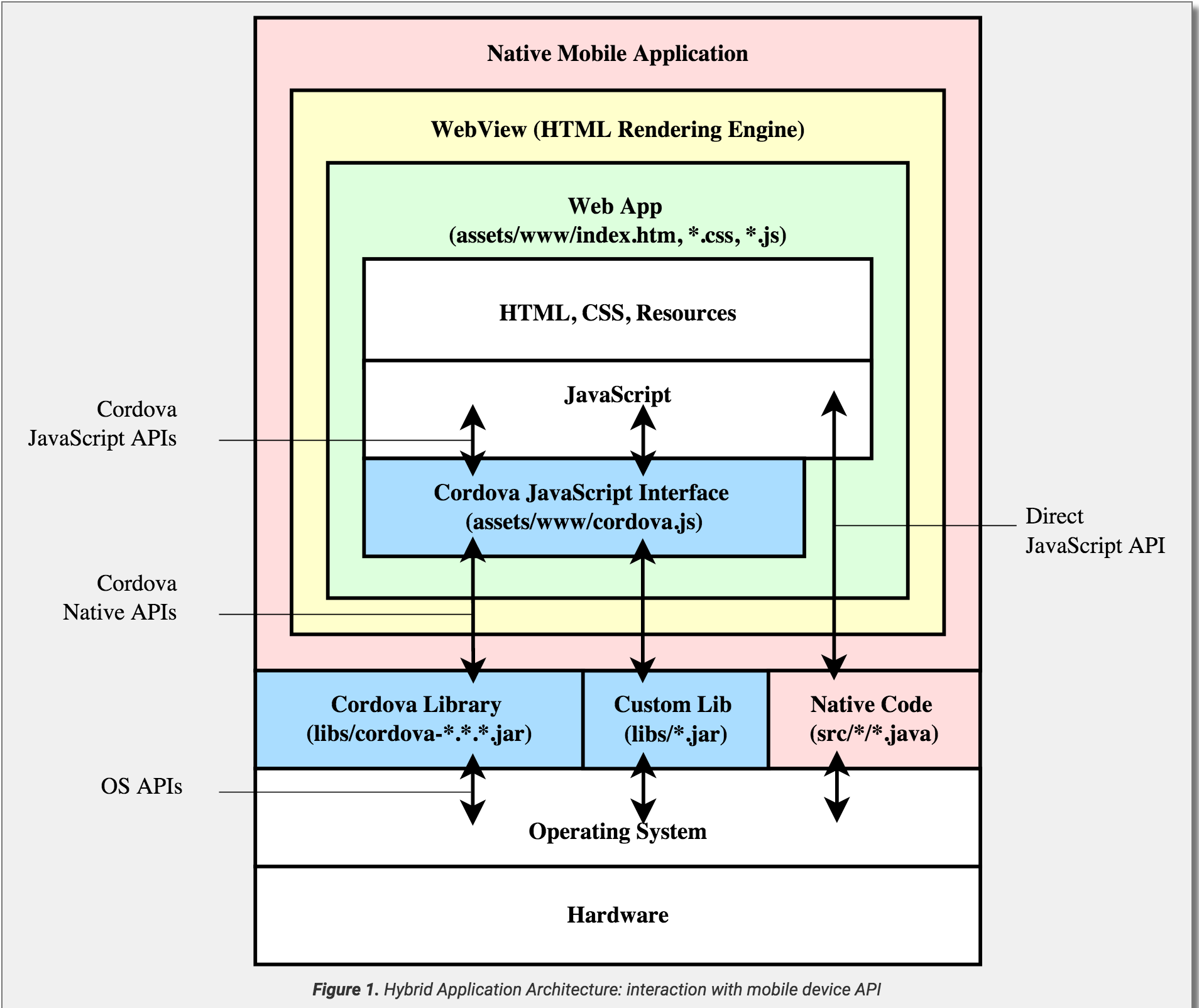
Có một số ứng dụng hybrid mobile nổi tiếng có sẵn trong các cửa hàng ứng dụng. như [Basecamp](https://itunes.apple.com/app/basecamp-for-iphone-official/id599139477), [Instagram](https://itunes.apple.com/app/instagram/id389801252), [Yelp](https://itunes.apple.com/app/yelp/id284910350), [Untappd](https://itunes.apple.com/app/untappd-discover-beer/id449141888), và [SydJS](https://itunes.apple.com/app/sydjs/id897396382) …

**2.2.2 Cordova**

Ngày nay hầu hết các ứng dụng hybrid mobile dựa trên [Apache Cordova](http://cordova.apache.org/), một nền tảng cung cấp một tập hợp các API Javascript để truy cập khả năng thiết bị thông qua plugins, được xây dựng bằng native code. Lưu ý, Apache Cordova có nguồn gốc bắt đầu từ một dự án có tên là [PhoneGap](http://phonegap.com/). Ngày nay PhoneGap tồn tại như một bản phân phối của Apache Cordova bao gồm các tiện ích bổ sung. Bạn có thể tìm hiểu thêm về điều này thông qua bài viết [PhoneGap, Cordova, and what’s in a name?](http://phonegap.com/blog/2012/03/19/phonegap-cordova-and-what-e2-80-99s-in-a-name).

Các plugins bao gồm các APIs để truy cập cảm biến gia tốc của thiết bị, camera, contacts, và nhiều hơn nữa. Ngoài ra còn có một số các plugins được xây dựng và duy trì bởi các cộng đồng phát triển tương đối lớn. Nhưng plugins này có thể được tìm thấy tại [Apache Cordova Plugins Registry](http://plugins.cordova.io/). Một lượng lớn các plugins đã được kiểm tra kỹ lưỡng, đầy đủ tài liệu và được mở rộng có thể tìm thấy tại [Telerik Verified Plugins Marketplace](http://plugins.telerik.com/).

Các assets như HTML, CSS, JavaScript được đóng gói thông qua các công cụ có sẵn thông qua Apache Cordova để nhắm tới mục tiêu nền tảng SDK. Sau khi xây dựng, bạn có một ứng dụng có thể chạy như bất kỳ loại ứng dụng khác trên thiết bị. Các công cụ được cung cấp bởi Apache Cordova phần lớn được giao tiếp thông qua giao diện dòng lệnh. Tuy, các nhà phát triển vẫn có thể tận dụng các IDEs như Visual Studio và các giải pháp như Platform Telerik để nâng cao productivity.



**2.2.3 Ionic 2**

Ionic là một hybrid Framework được sử dụng để phát triển các ứng dụng di động dựa trên nền tảng công nghệ web HTML ( sự kết hợp giữa Angular và Cordova), được tạo bởi Max Lynch, Ben Sperry, và Adam Bradley vào 2013. Đây là một framework rất mạnh để viết các ứng dụng hybrid. Nó khắc phục các nhược điểm của Native app và Mobile webapp và kết hợp được sức mạnh của 2 nền tảng đó.

Do được xây dựng trên khung phát triển AngularJS từ Google, Ionic sử dụng AngularJS để cung cấp các cấu trúc ứng dụng, trong khi bản thân Ionic tập trung chính vào giao diện người dùng. Nói cách khác, chúng ta thấy được sự phối hợp ăn ý giữa sức mạnh của AngularJS và vẻ đẹp của Ionic UI.

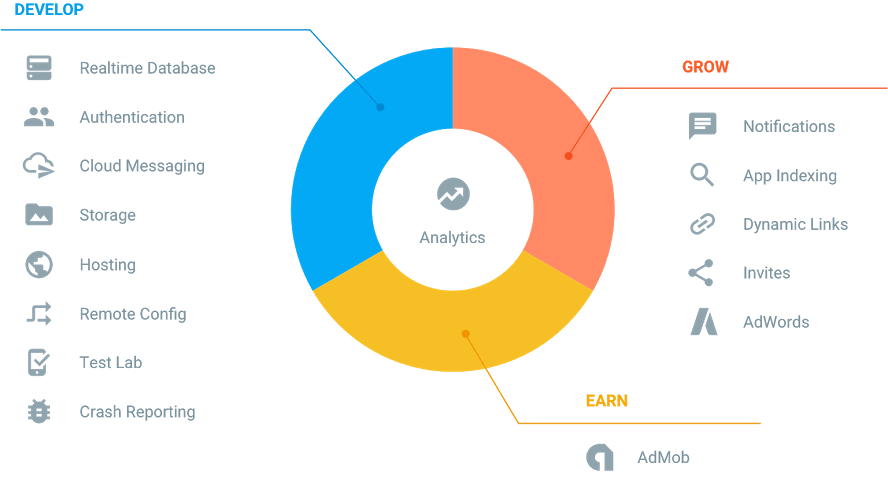
Ionic cung cấp một tập các Angular directives (nghĩa là các phần tử HTML tùy biến) để làm các thành phần của nó, tạo ra sự dễ dàng để sử dụng các tiện ích gọn để viết mã HTML. Ngoài các directives, Ionic còn sử dụng và thêm vào các thành phần khác như: Angular touch recognizers, view animation logic, HTML sanitation, và asynchronous communication.

**2.2.4 So sánh, đánh giá Ionic2 với các công nghệ khác**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Ionic** | **React Native** | **Native Language** |
| Đa nền tảng | Hỗ trợ | Hỗ trợ | Không hỗ trợ |
| Hiệu năng | Trung bình | Khá | Cao |
| Giao diện | Đơn giản, tự chuyển đổi giữa các platform | Kém | Đẹp |
| Kích thước file build | Trung bình(>= 2MB) | Tương đối lớn (>= 4MB) | Nhỏ |
| Thời gian xây dựng app | Nhanh | Nhanh | Chậm vì phải xây dựng cho từng nền tảng |
| Cộng đồng | Đông đảo | Trung bình nhưng tăng nhanh | Đông đảo |

**2.3 Firebase**

Google firebase là một dịch vụ cơ sở dữ liệu thời gian thực hoạt động trên nền tảng đám mây được cung cấp bởi Google nhằm giúp các lập trình phát triển nhanh các ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.



Các tính năng chính

* Dữ liệu thời gian thực – Firebase Realtime Database

+ Bạn đăng ký 1 tài khoản trên firebase, tạo một ứng dụng và bạn đã có 1 cơ sở dữ liệu thời gian thực.

+ Dữ liệu sẽ được cung cấp cho bạn dưới dạng JSON và luôn đồng bộ thời gian thực đến mọi kết nối client.

+ Với các ứng dụng đa nền tảng tất cả các client sẽ đề sử dụng chung 1 DB và luôn được tự động cập nhật dữ liệu mới nhất.

+ Tất cả dữ liệu được truyền qua một kết nối an toàn SSL với một chứng nhận 2048-bit.

+ Làm việc offline: Đừng lo nếu mất mạng bởi dữ liệu luôn được lưu trữ trước ở local mỗi khi có sự thay đổi nào sẽ được tự động cập nhật lên server của Firebase và ngược lại khi dữ liệu ở local là cũ hơn với server nó cũng sẽ được tự động cập nhật để dữ liệu luôn là mới nhất.

* Xác thực người dùng: Firebase đã xây dựng chức năng cho việc xác thực người dùng với Email, Facebook, Twitter, GitHub, Google, và xác thực nạc danh. Nó giúp bạn nhiều trong việc xác thực người dùng.
* Firebase Hosting: Firebase cung cấp các hosting và được phân phối qua SSL từ CDN sẽ giúp bạn tiết kiệm được rất nhiều thời gian trong việc xây dựng ứng dụng.

Lợi ích của Firebase

* Triển khai ứng dụng cực nhanh: Firebase cung cấp cho bạn khá nhiều các API, hỗ trợ đa nền tảng giúp bạn tiết kiệm thời gian quản lý cũng như đồng bộ dữ liệu cung cấp hosting, hỗ trợ xác thực người dùng thì việc triển khai ứng dụng sẽ giảm được rất nhiều thời gian phát triển.
* Bảo mật: Với việc sử dụng các kết nối thông qua giao thức bảo mật SSL hoạt động trên nền tảng cloud đồng thời cho phép phân quyền người dùng database bằng cú pháp javascipt cũng nâng cao hơn nhiều độ bảo mật cho ứng dụng của bạn.
* Sự ổn định: Firebase hoạt động dựa trên nền tảng cloud cung cấp bởi Google do đó hãy yên tập về việc một ngày đẹp trời nào đó server ngừng hoạt động hay như DDOS hoặc là tốc độ kết nối như rùa bò. Một điều đáng lưu ý nữa đó là do hoạt động trên nền tảng Cloud vì vậy việc nâng cấp hay bảo trì server cũng diễn ra rất đơn giản mà không cần phải dừng server.