1. **Introduction/ Giới thiệu**

**//////////////////////**

Ionic là một framework dùng để phát triển ứng dụng hybrid dựa trên HTML5. Hybrid có thể được hiểu là con lai giữa ứng dụng native và web mobile.Một ứng dụng hybrid là một ứng dụng di động được xây dựng bằng các công nghệ phát triển web như HTML5, CSS, JavaScript và sau đó được đóng gói trong một môi trường hiển thị nội dung có thể hoạt động trên hầu hết các thiết bị di động giống như trình duyệt web (Native container). Các ứng dụng này sử dụng các thành phần hiển thị nội dung website (các trình duyệt ẩn như UIWebView trên iOS, hay Webview trên Android và các thiết bị khác) để hiển thị các đoạn mã HTML.

Ứng dụng hybrid đem lại nhiều lợi thế như khả năng hiển thị nội dung trên tất cả các thiết bị di động, cũng như tận dụng tối đa các tính năng khác của thiết bị di động như GPS, camera, danh sách liên lạc,… Bên cạnh đó, thời gian và chi phí dùng để tạo nên một ứng dụng hybrid cũng thấp hơn so với các ứng dụng di động thông thường.

Bạn có thể coi Ionic như là một bộ khung front-end giúp bạn kiểm soát hình ảnh và trải nghiệm trên ứng dụng của bạn. Giống như “Bootstrap for Native”, nhưng với sự hỗ trợ của một lượng lớn các thành phần di động, hiệu ứng chuyển động mượt mà và thiết kế đẹp.

Không như các framework khác, Ionic mang đến cho chúng ta những thành phần giao diện và cách bố trí mang phong cách rất giống với như phong cách thiết kế mặc định trên thiết bị di động.

Vì Ionic là một framework HTML5 nên nó cần bộ đóng gói (wrapper) như Cordova hay PhoneGap để có thể chạy được như một ứng dụng di động. Cordova là bộ đóng gói mặc định trong Ionic framework.

**//////////////////////**

Tìm hiểu về ionic framework

Giới thiệu Ionic là một framework dùng để phát triển một ứng dụng Hybrid cho mobile. Hybrid có thể được hiểu là con lai giữa ứng dụng native và web mobile. Bản chất của hybrid là một giao diện được viết bởi html, css, javascript nhưng nó cho phép gọi các api native của hệ thống nên có thể thao tác với hệ điều hành của mobile như các ứng dụng native khác. Ionic được phát triển dựa trên nền của Apache Cordova(trước kia còn gọi là phonegap) và dùng angular1, angular2 để làm core của hệ thống. Các tính năng của Cordova

* + Command Line Interface (Cordova CLI): đây là một tool dùng để khởi tạo một project , xây dựng ứng dụng trên nhiều nên tảng khác nhau cũng như là thêm nhiều plugin có ích để giúp việc phát triển dễ dàng hơn.
  + Cordova Core Components: Cordova đưa ra một tập các thành phần mà mọi ứng dụng di động cần đến.
  + Cordova Plugins: Cordova đưa ra các API để sử dụng các chức năng của thiết bị di động như cảm biến, camera, GPS…

Ưu điểm: Các ứng dụng hybrid có nhiều ưu điểm như có thể hiển thị cho nhiều nền tảng Tận dụng các tính năng khác của thiết bị di động như GPS, camera.... Thời gian và chi phí dùng để phát triển ứng dung cũng thấp hơn nhiều so với native. Chỉ cần biết ngôn ngữ javascriot mà ko cần phải biết từng ngôn ngữ lập trình Nhược điểm: Tốc độ chậm với một số tính năng chuyển trang trên di động Các plugin có thể không tương thích với một số thiết bị và nền tảng. Một số API cũng chưa được hỗ trợ để giao tiếp với thiết bị. Ionic có thể được coi là một khung front-end giúp bạn kiểm soát hình ảnh và trải nghiệm trên ứng dụng di động, hiệu ứng chuyển động của nó mượt và thiết kế đẹp. Cộng đồng Ionic cũng lớn, có hẳn 1 trang thiết kế theme riêng có thể lựa chọn để phát triển cho nhanh

\***) Ưu điểm và nhược điểm**

**a. Ưu điểm**

– Dễ học, thời gian phát triển nhanh, có thể sử dụng các kỹ năng từ lập trình web

– Đa nền tảng

– Khả năng truy cập đến các tính năng của thiết bị và hệ điều hành như bluetooth, camera,..

– Dễ dàng thiết kế giao diện cho các thiết bị có kích cỡ khác nhau

– Việc sử dụng AngularJS làm core giúp phần xử lý UI linh động hơn so với javasript hay thư viện Jquery.

– Việc sử dụng AngularJS làm core cũng mang lại lợi thế lớn so với các framework cho ứng dụng hybrid khác.

– Ionic cung cấp đầy đủ các thành phần trong giao diện người dùng như Pull-to-Refresh, Infinite-loader, tabs, ..

**b. Nhược điểm**

– Vẫn còn trong giai đoạn phát triển

– Hiệu năng vẫn chưa cao và ổn định

– Cộng đồng phát triển ứng dụng vẫn còn chưa đông



**2. Features/ Tính năng**

**2.1 Cordova**



Apache Cordova là một bộ khung để xây dựng ứng dụng di động sử dụng HTML, CSS và Javascript. Apache Cordova bao gồm một tập hợp các API thiết bị cho phép người lập trình di động truy cập, sử dụng các chức năng native của thiết bị như là camera hay cảm biến gia tốc bằng Javascript. Kết hợp với một bộ khung phát triển giao diện như jQuery Mobile or Dojo Mobile hoặc Ionic, cho phép ứng dụng di động có thể được phát triển chỉ dựa trên HTML, CSS và Javascript.

Khi sử dụng Cordova API, một ứng dụng có thể được xây dựng mà không phải sử dụng bất kỳ một đoạn mã native code nào. Thay vào đó, công nghệ web sẽ được sử dụng, và chúng sẽ được tổ chức trên chính ứng dụng đấy chứ không cần thông qua một server nào.

Và bởi vì những API Javascript này là thống nhất trên tất cả các nền tảng thiết bị và được xây dựng trên chuẩn web nên những ứng dụng được viết trên nền tảng này có thể được sử dụng trên các nền tảng khác mà có thể không cần có bất cứ sự thay đổi nào.

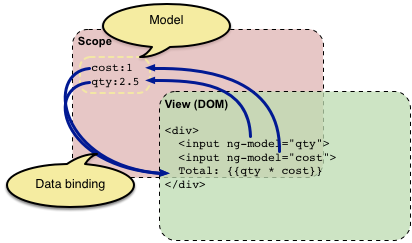
Cordova cung cấp một tập hợp các thư viện Javascript đã được chuẩn hóa để có thể sử dụng. Cordova hiện có thể sử dụng cho các nền tảng như iOS, Android, Blackberry, Windows Phone, Palm WebOS, Bada và Symbian.

Kể từ phiên bản 3.0, bạn có thể chọn một trong hai quy trình để phát triển một ứng dụng di động.

* Cross-platform (CLI) workflow: được sử dụng khi bạn muốn xây dựng ứng dụng trên càng nhiều nền tảng càng tốt, mà ít cần quan tâm đến đặc trưng của nền tảng. Trung tâm của quy trình này là CLI (Command-Line Interface), được giới thiệu từ phiên bản Cordova 3.0. CLI là một công cụ ở mức cao cho phép bạn xây dựng dự án trên nhiều nền tảng một lần duy nhất, trừu tượng hóa đi nhiều chức năng ở mức thấp. CLI cũng cung cấp một giao diện chung để sử dụng các plug-in vào ứng dụng của bạn.
* Platform-centered workflow: được sử dụng khi bạn muốn tập trung phát triển ứng dụng trên một nền tảng cụ thể và muốn can thiệp sâu vào mức thấp ví dụ như khi bạn muốn kết hợp một thành phần native đã được tùy biến vào một thành phần web của Cordova. Quy trình này phụ thuộc vào một tập các đoạn mã ở mức thấp được thiết kế riêng cho mỗi nền tảng được hỗ trợ. Vì vậy khi sử dụng cách phát triển này sẽ có thể gặp nhiều khó khăn hơn bởi vì sự thiếu hụt của những công cụ ở mức cao nhưng bạn sẽ có được sự tự do hơn trong việc phát triển ứng dụng.

**2.2 Mô hình phát triển ứng dụng MV Pattern**

Angular JS được sử dụng trong Ionic là một MVC framework, nhưng theo thời gian thì nó trở nên thân thuộc hơn với khái niệm MVVM trong đó đối tượng $scope hoạt động như một ViewModel, sử dụng và thao tác trong các Controller.



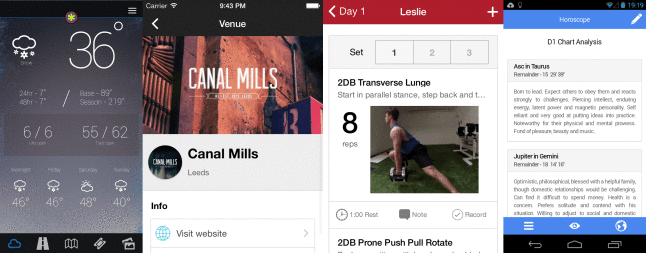
Các developer có thể linh hoạt sử dụng MVC hoặc là MVVM theo ý thích của họ, miễn là đạt được mục tiêu: Việc tách phần trình bày giao diện khỏi tầng nghiệp vụ logic thúc đẩy hiệu suất phát triển sản phẩm và bảo trì.

**Thao tác với DOM**

Về khả năng tương tác với các thành phần DOM của webView. Angular nhúng jqLite, là một phần nhỏ gọn của thư viện jQuery cho phép thao tác DOM trong các trình duyệt theo cách tương thích, và không phụ thuộc vào trình duyệt.

**2.3 Giao diện người dùng UI**

Ionic thể hiện được tiềm năng của nó. là sự đơn giản. Giống như phong cách của Google, nó dựa trên sự tồn tại của HTML5 và CSS3 để cung cấp những trải nghiệm nhanh chóng. Chính xác là tốc độ có trong sự đơn giản của nó – không có bóng đổ hay góc tròn, chỉ đơn giản là phẳng – Như vậy, Ionic không hứa hẹn có được giao diện người dùng bóng bẩy, nhưng nó cung cấp giao diện một cách nhanh chóng và nhất quán, thậm chí là chúng ta sử dụng trên các thiết bị với khả năng thông dịch HTML5 chậm trong các ứng dụng Ionic.



Sass là một ngôn ngữ mở rộng của CSS – Cascading Style Sheets – cho phép Ionic thêm các biến số và khả năng lồng cú pháp để mở rộng sự xuất hiện của giao diện ứng dụng. Ngoài ra, Ionic còn được đóng gói thêm các thư viện icon nguồn mở, khoảng 440 icons.

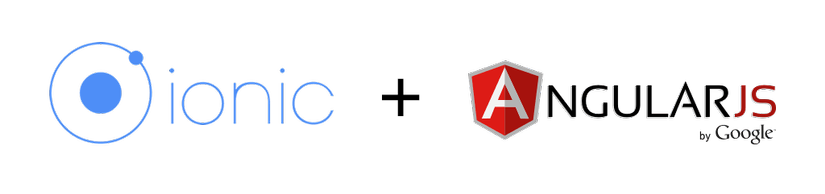
**2.4 Giao diện dựng sẵn Widgets**

Các thành phần của Ionic cực kỳ đơn giản và mạnh mẽ. Chúng là các phần tử HTML phức hợp, được gọi là các directives, Ionic cũng cung cấp các Controller để bổ sung cho cấu hình và tương tác. Ionic cung cấp các khối xây dựng có phần đơn giản mà có thể được kết hợp để cung cấp giao diện người dùng phong phú.



Ở phiên bản hiện tại v1.2.8, Ionic cung cấp một loạt các thành phần giao diện thiết kế sẵn bao gồm: form elements, header and footer bars, buttons, simple list, grid elements và các thành phần phổ biến khác nữa.

**2.5 Pairing with AngularJS**



AngularJS là một dự án mã nguồn mở của Google đã trở nên khá phổ biến đối với những nhà phát triển ứng dụng web trong thời gian gần đây. AngularJS cung cấp cho người phát triển khả năng hoàn thiện ứng dụng web một cách nhanh chóng và tạo cấu trúc hợp lí cho ứng dụng. Các framework cho ứng dụng web dùng Javascript như AngularJS cho phép bạn xây dựng các ứng dụng phức tạp ngay trong trình duyệt mà không cần dựa trên server. Đây chắc chắn là một lợi thế cho các nhà phát triển ứng dụng hybrid, bởi trình duyệt là nền tảng để tạo nên những ứng dụng này. Nếu bạn quen thuộc với AngularJS hoặc các framework Javascript khác như Ember, bạn sẽ dễ dàng làm quen với việc phát triển ứng dụng di động bằng Ionic.

Ionic sử dụng AngularJS để tạo ra một framework tốt nhất cho việc xây dựng các ứng dụng di động đa nền tảng, nó cung cấp một bộ giao diện người dùng (UI) mã nguồn mở miễn phí đi cùng với các tính năng của AngularJS.

Việc xây dựng ứng dụng dựa trên AngularJS đòi hỏi mã nguồn phải có khả năng mở rộng cao để bổ sung các tính năng mới. Với Ionic, người ta có thể tái sử dụng các chức năng trong ứng dụng trên các nền tảng khác nhau đồng thời vẫn có thể tùy chỉnh giao diện người dùng cho mỗi nền tảng riêng biệt. Các thành phần trong Ionic như danh sách, slide,.. chính là các directive(các thuộc tính của thẻ HTML dùng trong Angular) của AngularJS. Đó là lí do khiến cho Ionic và AngularJS kết hợp rất tốt với nhau.

**2.6 Performance obsessed**

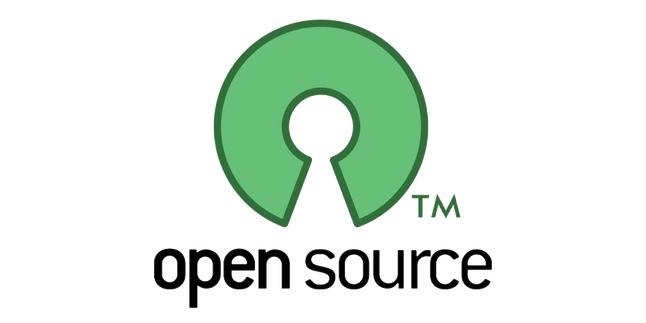
Ionic rất chú trọng đến hiệu suất, mặc dù nó ra đời chưa lâu. Ionic có một giao diện ổn định và tốc độ tốt, với các hiệu ứng chuyển động được áp dụng kỹ thuật tăng tốc phần cứng (hardware accelerating) và tối giản các thao tác với DOM. Ionic cũng không cần sử dụng đến jQuery, mặc dù người sử dụng có thể thêm vào nếu muốn. Ionic có thể tạo ra các ứng dụng phức tạp chạy trên cả iOS và Android.

Một điểm thú vị của Ionic đồng thời giúp tăng hiệu suất hoạt động của nó, đó là Ionic không cố gắng tự mình thực hiện tất cả các công việc. Để đóng gói ứng dụng, Ionic sử dụng Cordova và tận dụng cấu trúc thư mục mặc định của nó.

**2.7 Application scripting**

Ionic không những mang đến cho chúng ta CSS và markup tùy biến cao mà còn những mẫu thiết kế (design pattern) Javascript để giúp chúng ta xây dựng những ứng dụng giống nhất với những ứng dụng native trên Android và iOS. Ionic muốn giải phóng những ứng dụng web ra khỏi thanh địa chỉ và hướng đến những tương tác giao diện mạnh mẽ như mà side menu hay view controllers.

**2.8 Open-source**



Mục đích của Ionic framework không chỉ là xây dựng một bộ khung phát triển ứng dụng di động đa nền tảng mà còn muốn xây dựng một nền tảng để chia sẻ các kiến thức cho các nhà phát triển, tạo ra một cách để đưa ra những design pattern tốt nhất để xây dựng các ứng dụng di động. Vì vậy, Ionic là một framework mở hoàn toàn cho phép tất cả mọi người có thể chia sẻ những kiến thức, kinh nghiệm để xây dựng những ứng dụng di động tuyệt vời trên nền tảng HTML5.

Ionic sẽ đặt nền tảng cho ứng dụng của bạn dựa trên HTML5, CSS và Javascript. Thay vì mỗi lập trình viên phải tự tìm tòi những khuyết điểm của việc phát triển ứng dụng di động trên HTML5 thì Ionic sẽ cho bạn CSS cơ bản và tạo cho bạn 1 kiến trúc tốt để phát triển phía trên nó.

**2.9 Pros and Cons**

Sử dụng Ionic mang lại nhiều lợi ích, trong đó có các lợi thế từ ứng dụng hybrid như:

– Người phát triển có thể sử dụng các kỹ năng từ lập trình web, giúp giảm thiểu thời gian cũng như chi phí đào tạo nói riêng và thời gian cũng như chi phí dành cho dự án nói chung.

– Chỉ một bộ mã lệnh có thể sử dụng trên nhiều hệ điều hành khác nhau.

– Khả năng truy cập đến các tính năng của thiết bị và hệ điều hành.

– Khả năng sử dụng trong chế độ offline.

– Việc thiết kế giao diện cho các thiết bị có kích cỡ khác nhau trở nên dễ dàng với thiết kế tùy ứng (responsive design) của web.

– Việc sử dụng AngularJS làm core cũng mang lại lợi thế lớn so với các framework cho ứng dụng hybrid khác.

– Ionic cung cấp đầy đủ các thành phần trong giao diện người dùng như Pull-to-Refresh, Infinite-loader, tabs, ..

– Một lợi thế khác khi sử dụng Ionic là nó có một cộng đồng người dùng lớn mạnh, giúp cho việc tìm hiểu và hỗ trợ dễ dàng hơn.

Tuy vậy Ionic cũng có các mặt bất lợi của nó:

– Vẫn còn trong giai đoạn beta, vẫn còn có thể xuất hiện bugs.

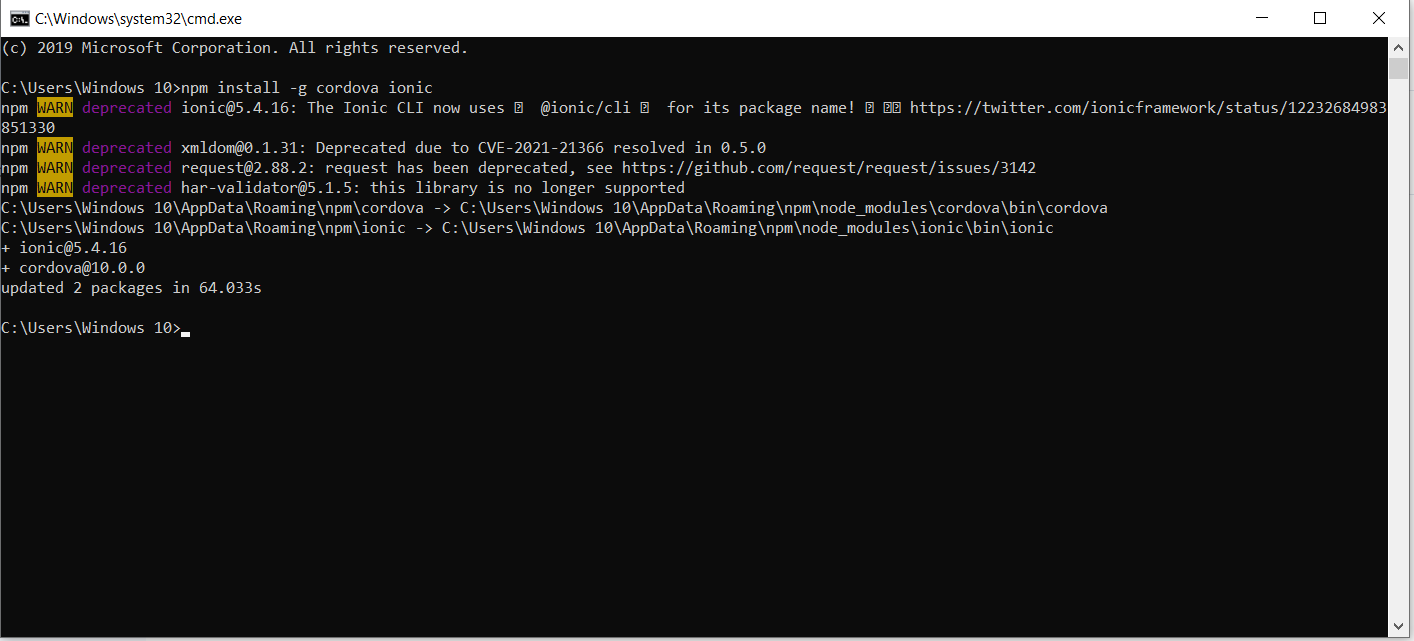
– Các hiệu ứng và khả năng chuyển đổi giữa các trang có hiệu suất chưa cao.

**3. Cài Đặt và Demo**

Trên đây mình đã giới thiệu qua cho các bạn về ionic framework, trong bài tiếp theo mình sẽ hưỡng dẫn các bạn cài đặt và tạo 1 project với ionic.

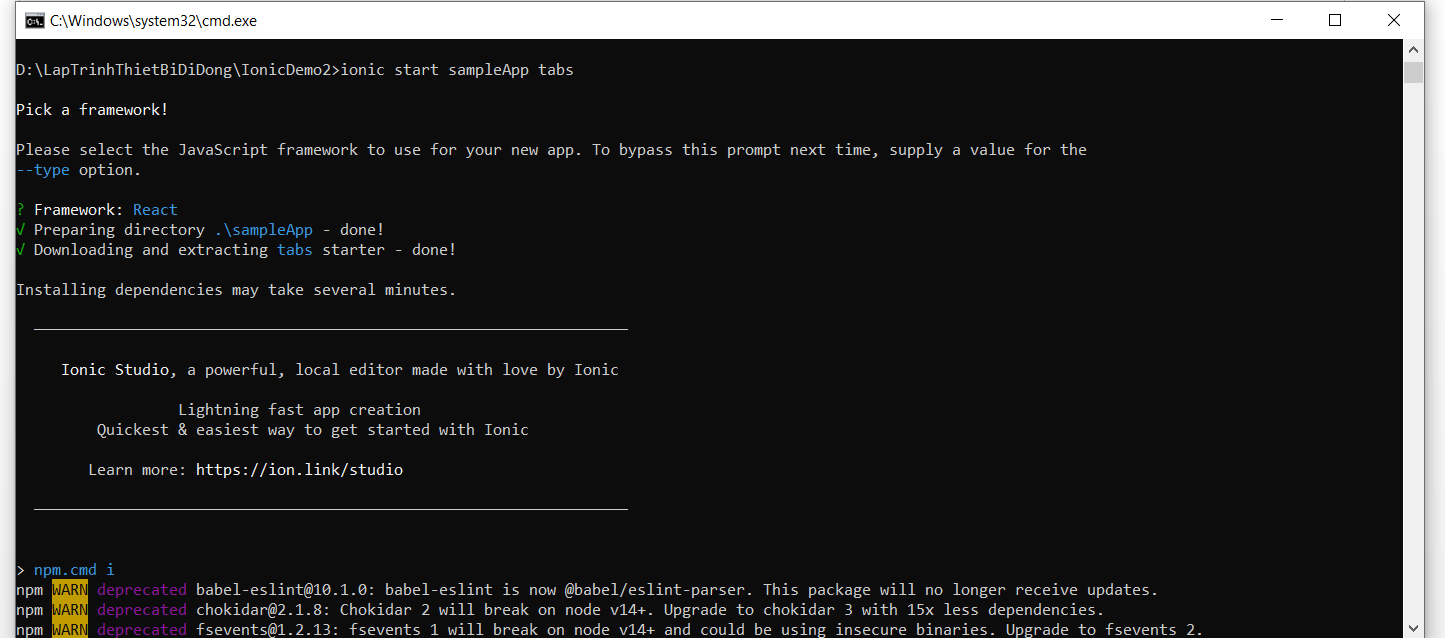


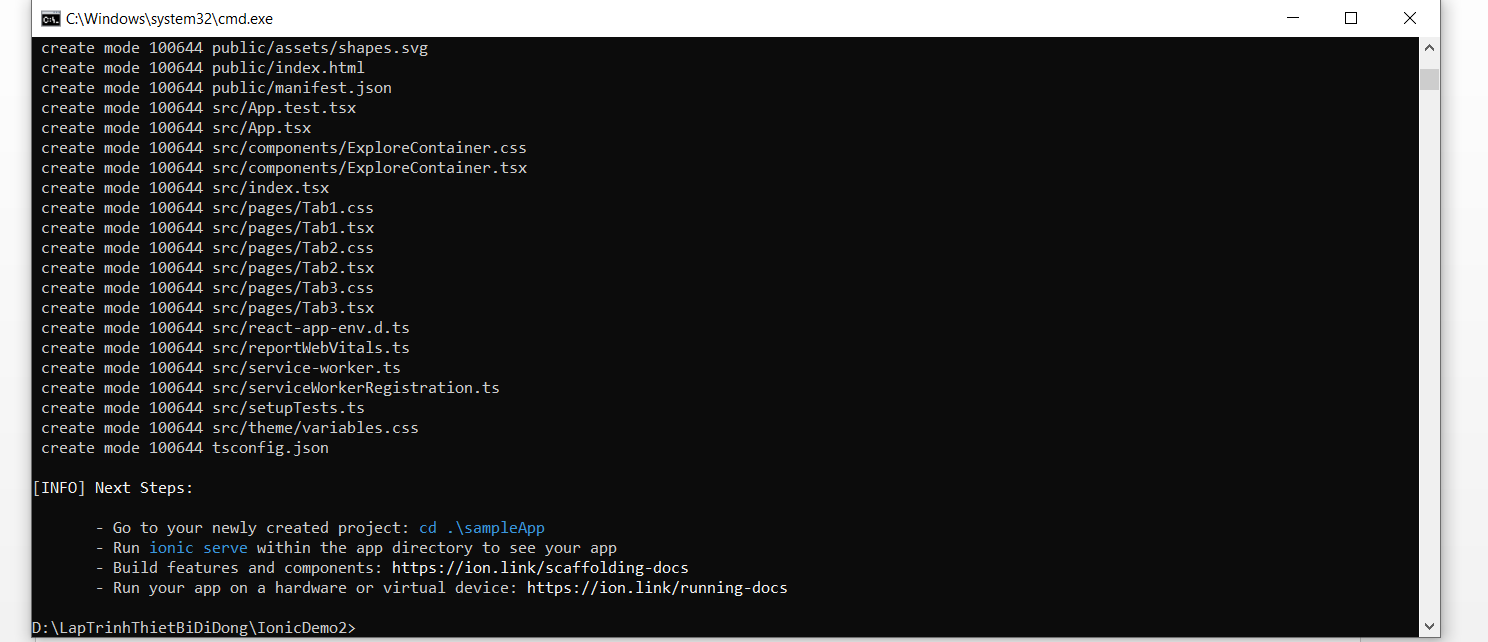
Mở terminal nhập vào lệnh *npm install -g cordova ionic*



Tạo 1 ứng dụng có tên DemoIonic

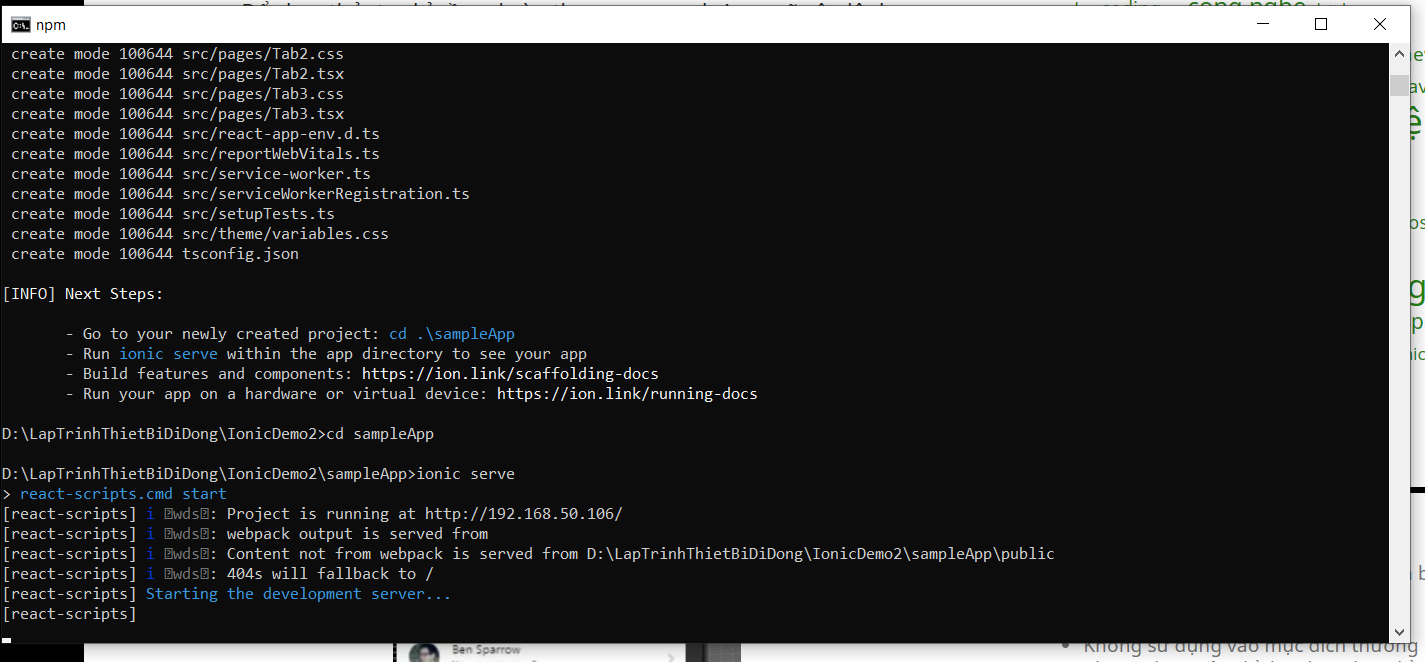
CD vào thư mục rồi gõ lệnh *ionic start sampleApp tabs*

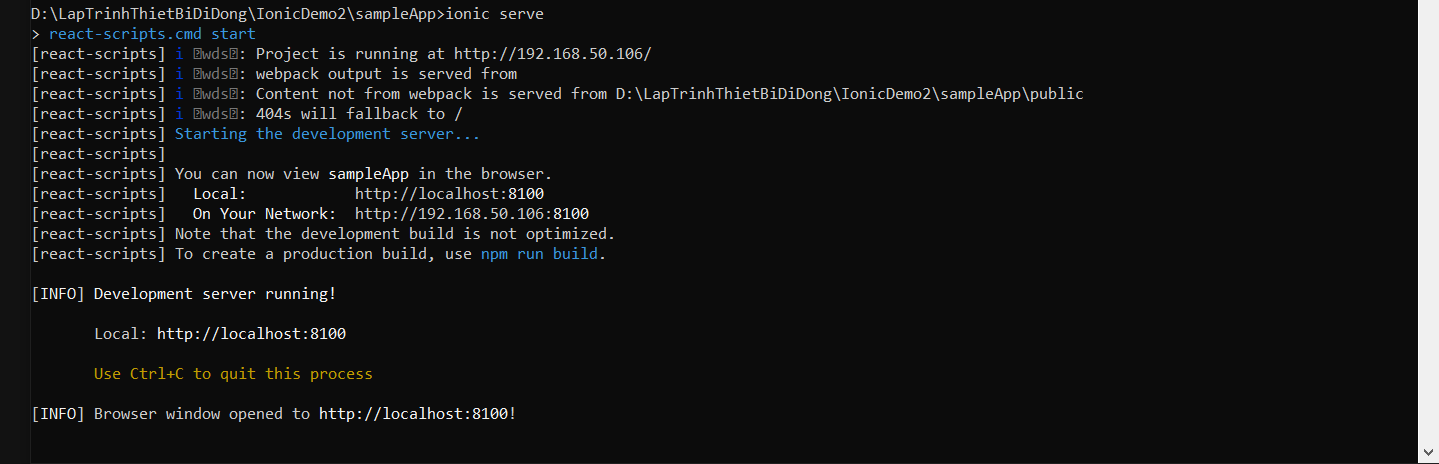


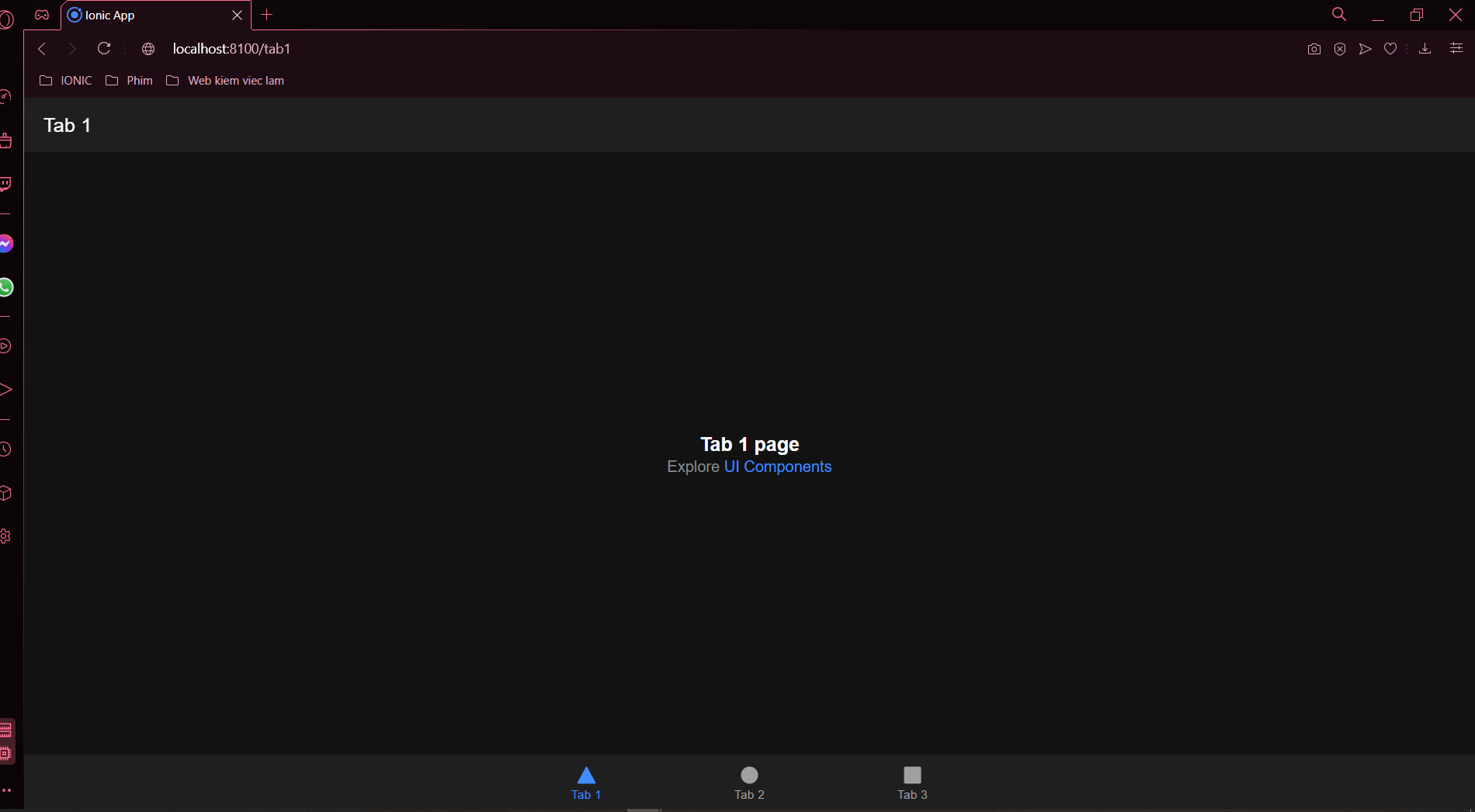


Cd vào sampleApp

Gõ ionic serve







### ****4. Chạy ứng dụng trên di động****

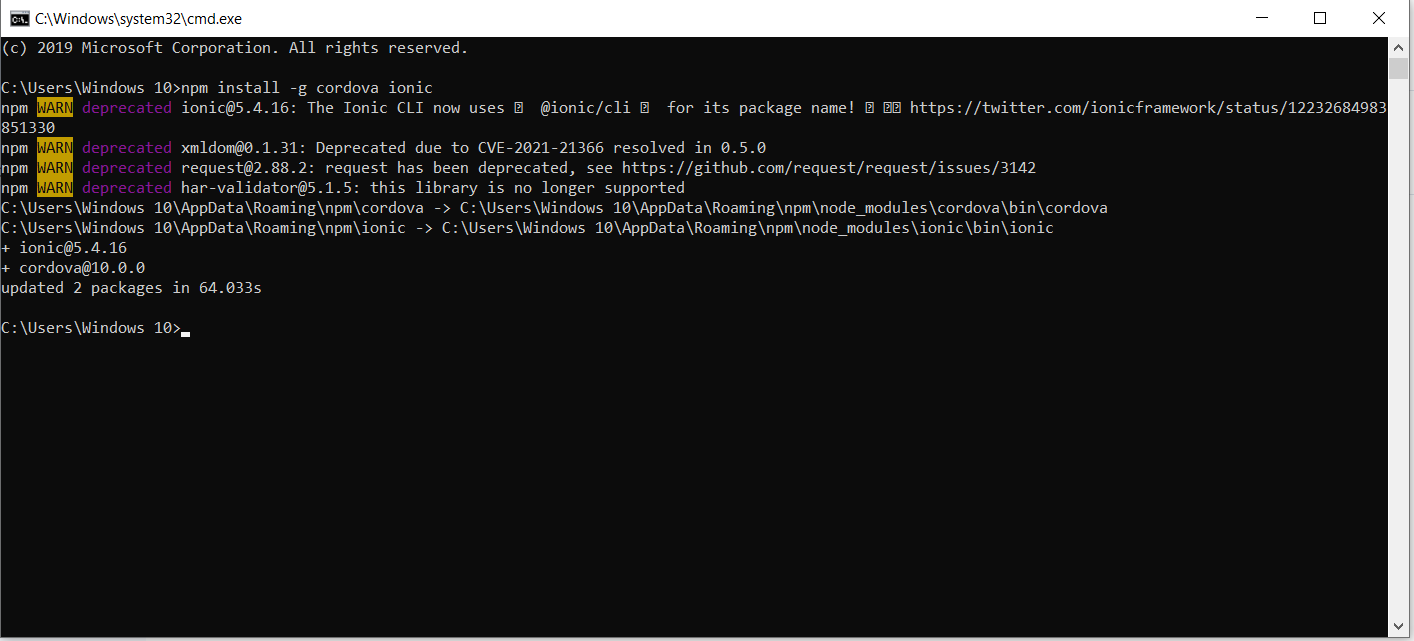
Upload ứng dụng lên máy ảo di động và chạy

* Cài đặt ionic 2 upload lên android studio

Cài Đặt Ionic Framework

Trước hết chúng ta phải cài đặt Cordova (Có giải thích trong slide thuyết trình )

$ npm install -g ionic cordova

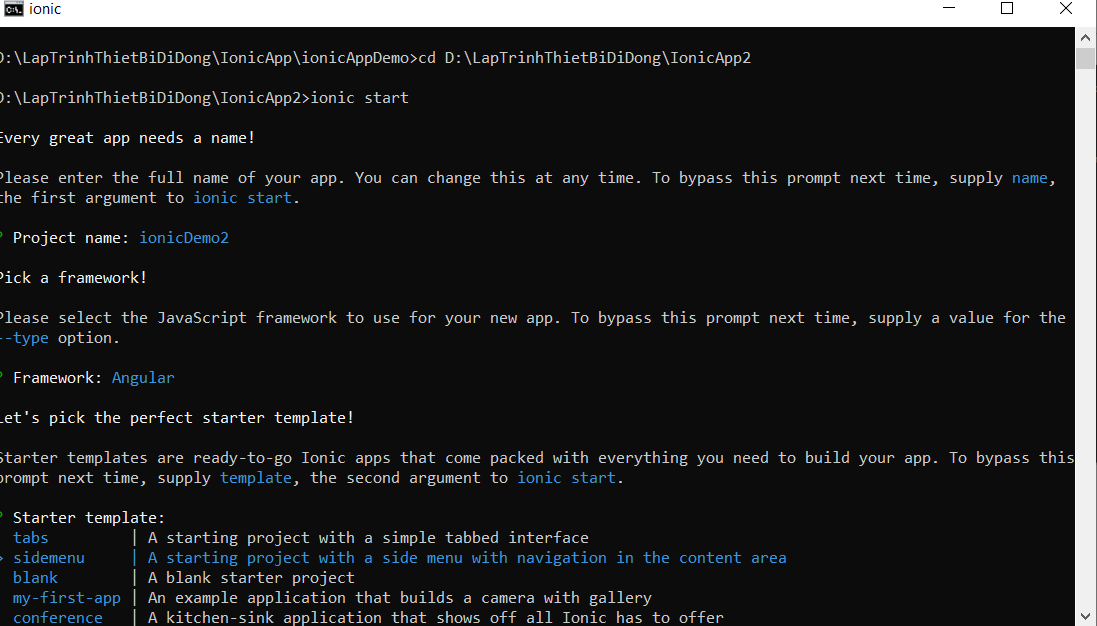


Ở đây thì mình đã cài cordova rồi

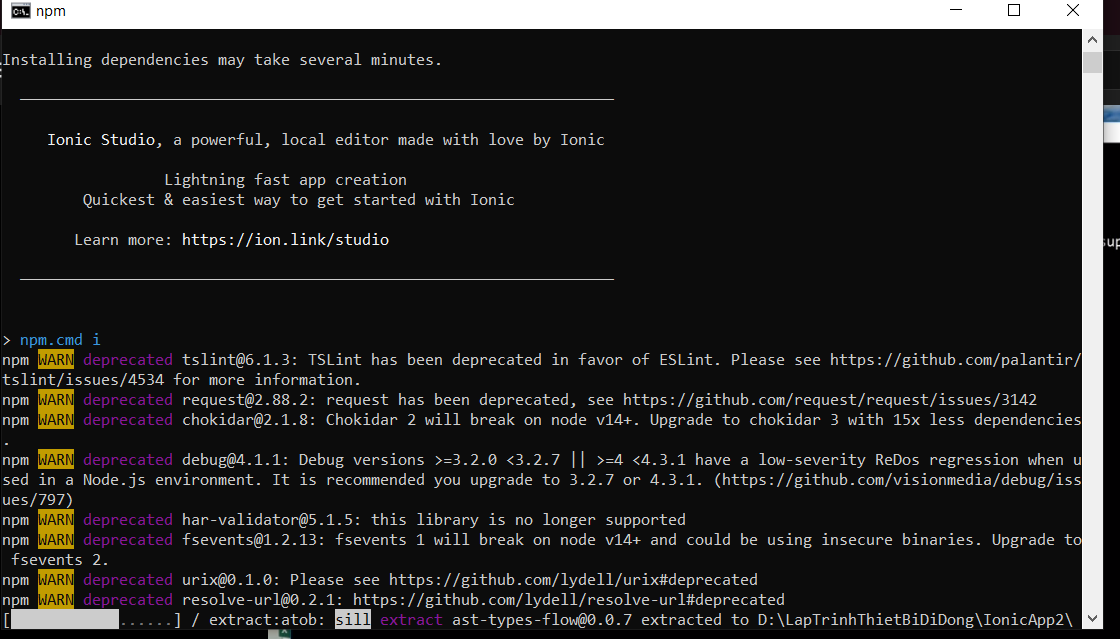
Nên là mình sẽ tạo 1 thư mục mới để chứa project ionic

CD vào thư mục cần cài đặt và gõ ionic start

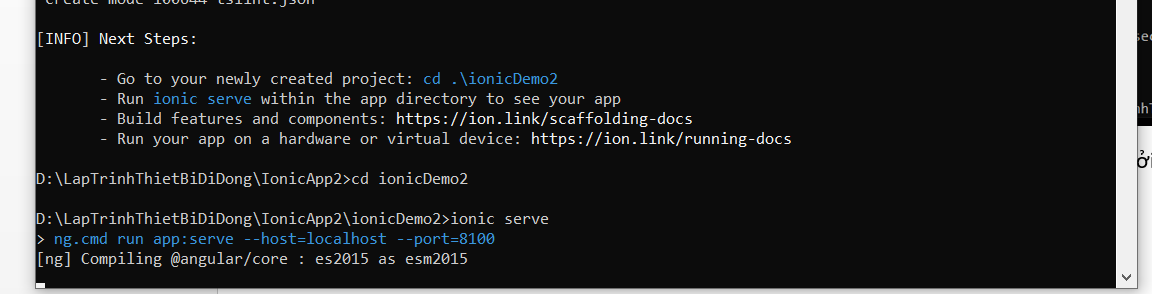
Và đặt tên là ionic Demo2 FrameWork là Angular và Dạng giao diện là side menu

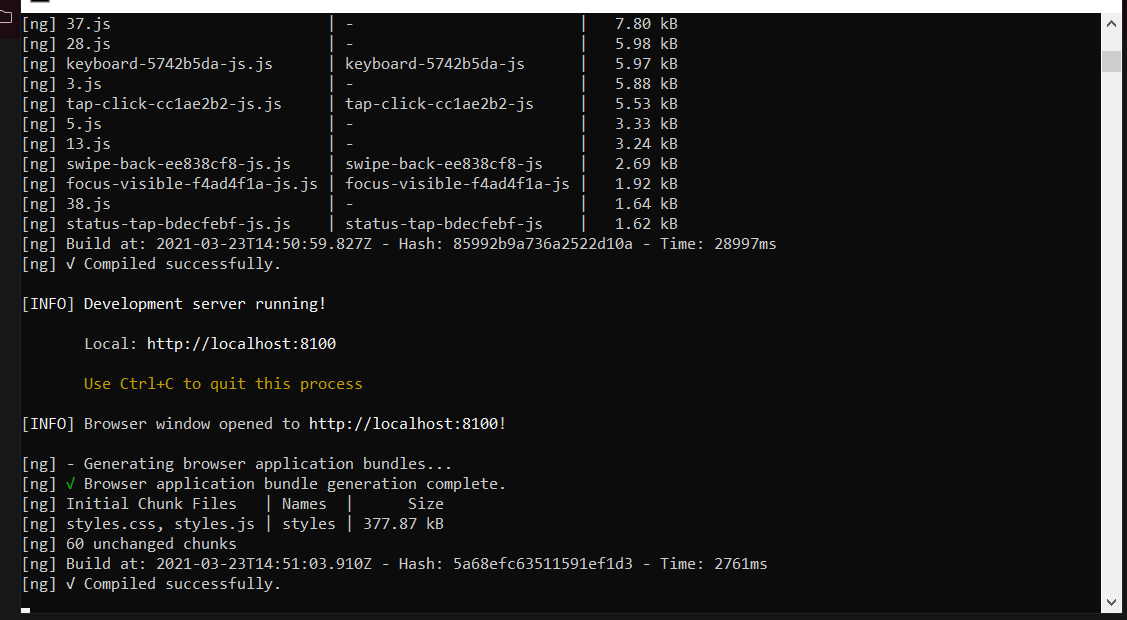


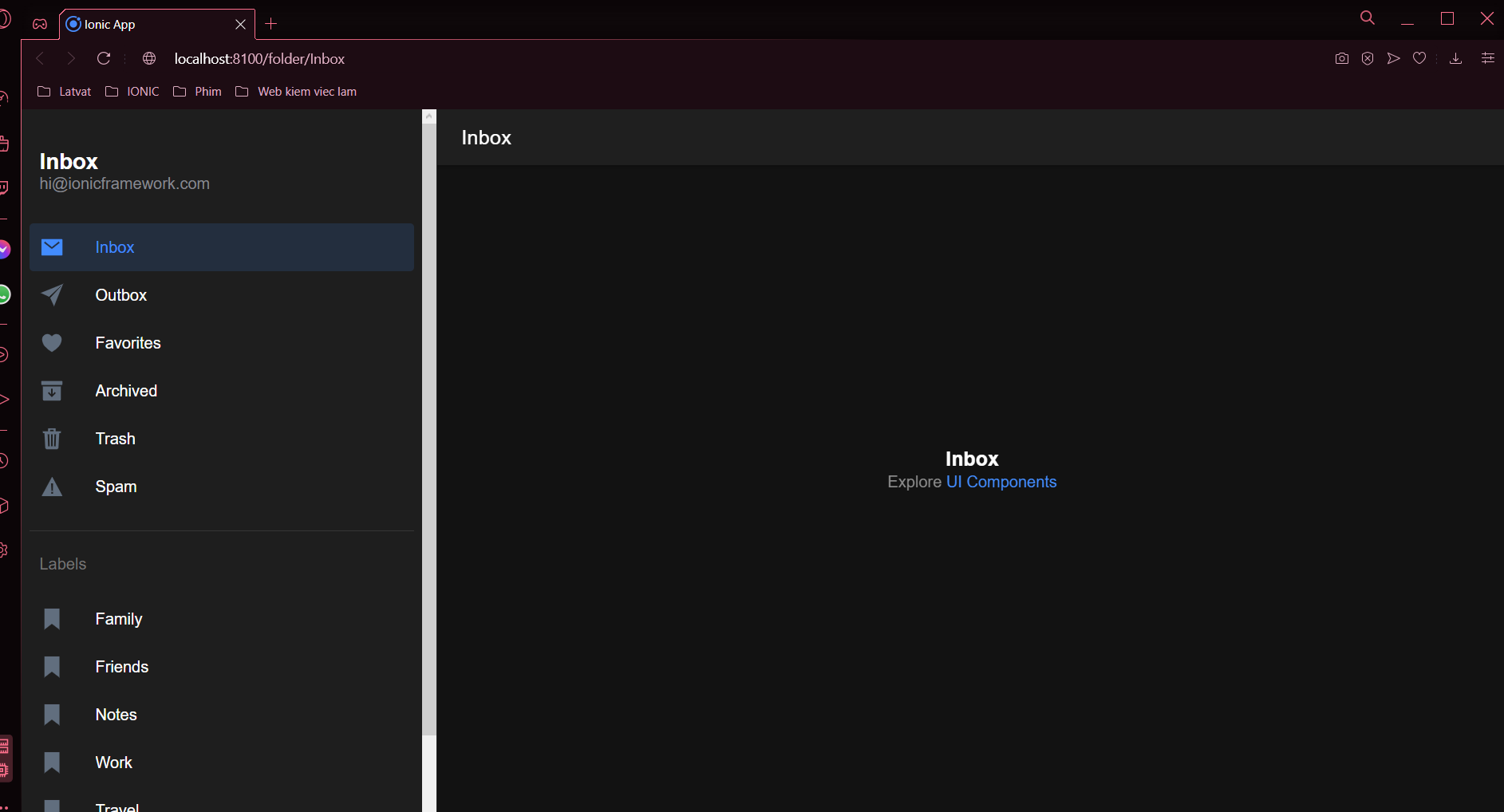
Hệ thống sẽ tải và cài đặt các file cần thiết trên thư mục quá trình mất từ 5 đến 10 phút .



Sau khi cài đặt xong ta cd vào thư mục project và gõ ionic serve để hệ thống khởi động máy ảo và chạy trên localhost .

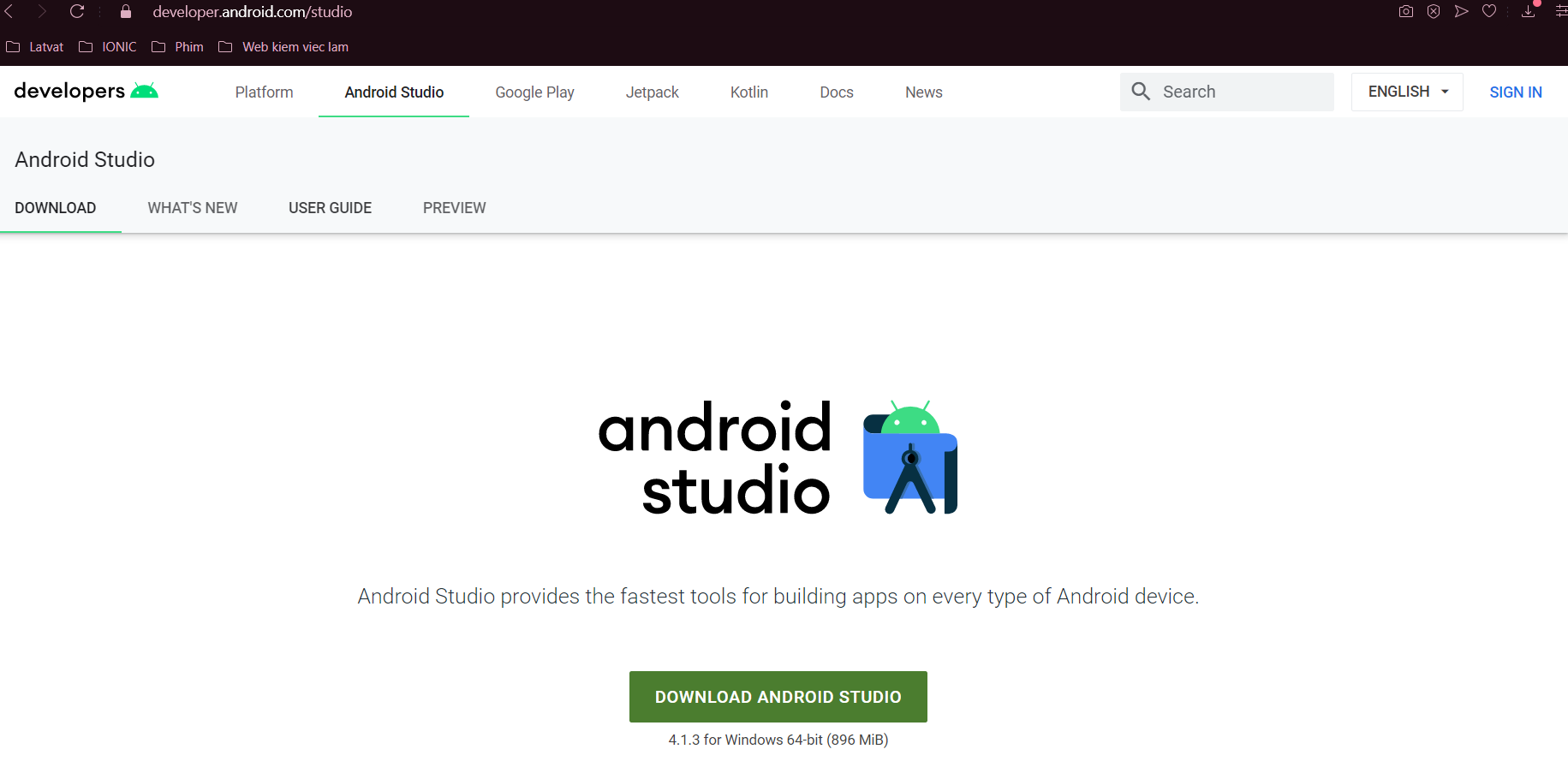




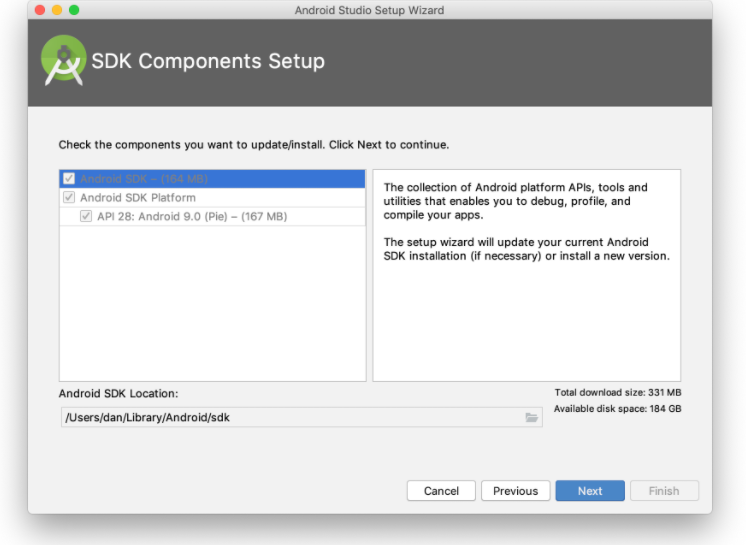


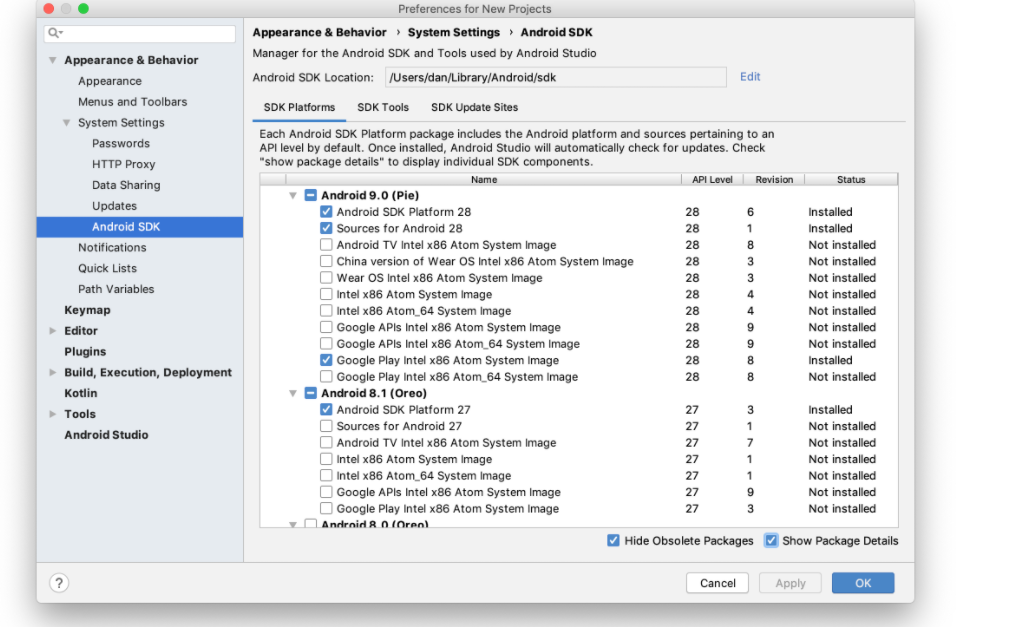
Chuyển Ứng Dụng Ionic App thành Ứng dụng Native APP chạy trên Android Studio

Đầu tiên ta cần phải cài đặt Android studio

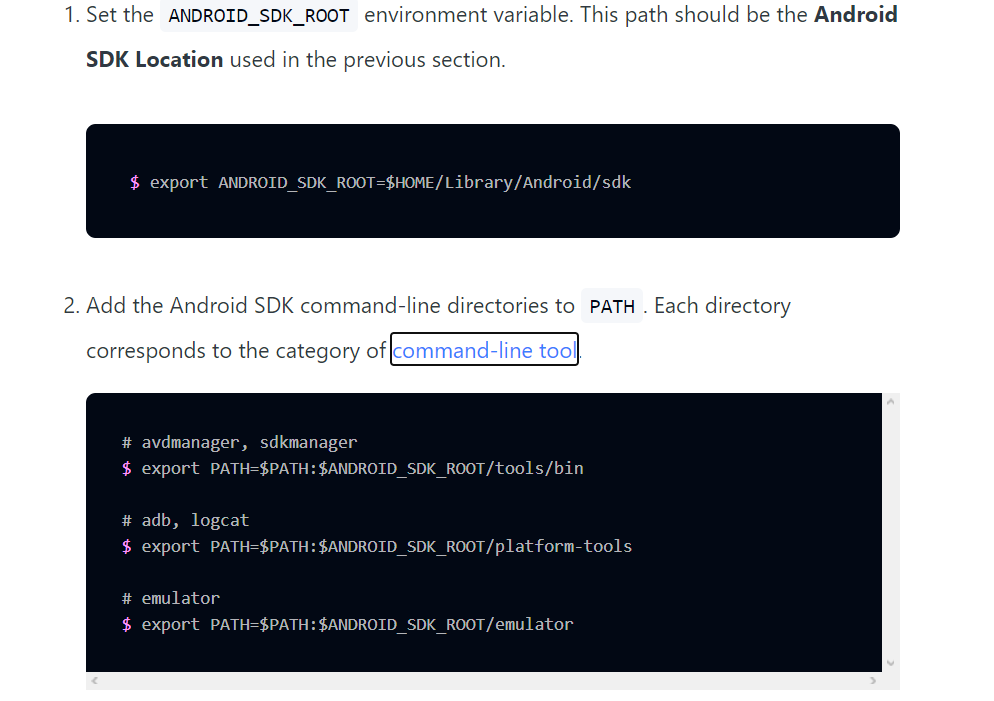


Và Android SDK



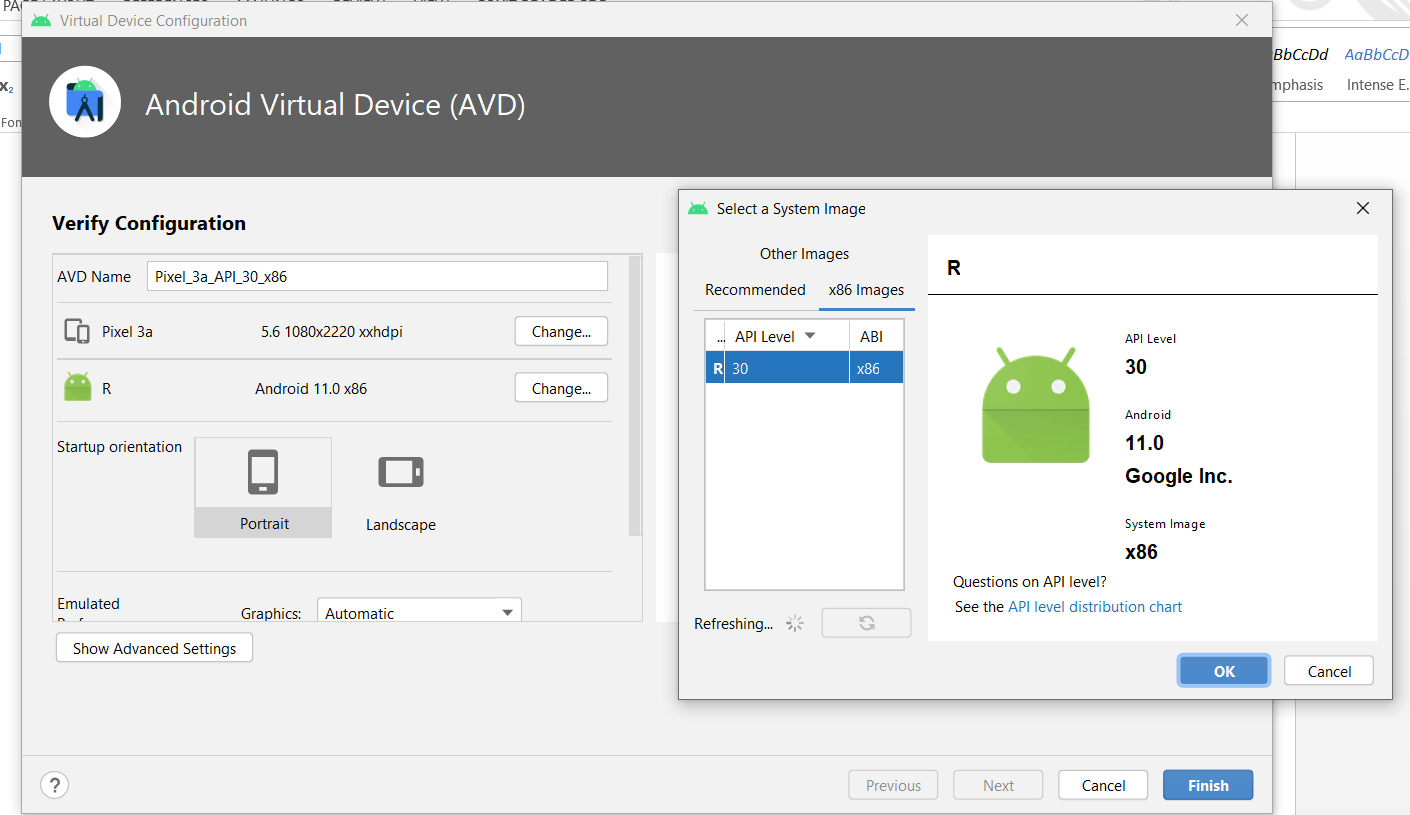


Nhớ cài đặt cả command line tool để add đường dẫn Android SDK vào environment variables trên máy tính của bạn nhé

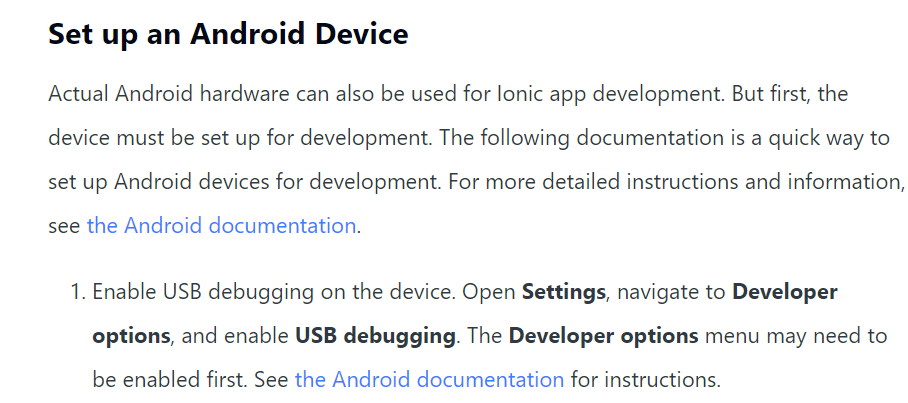


Tiếp theo là tạo 1 máy ảo

Để có thể chạy được Ứng dụng Ionic Native app trên android chúng ta cần API của máy ảo từ 24 trở lên

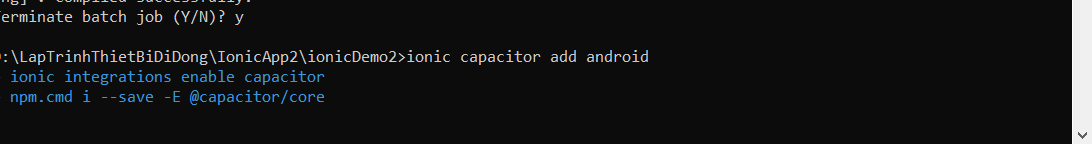


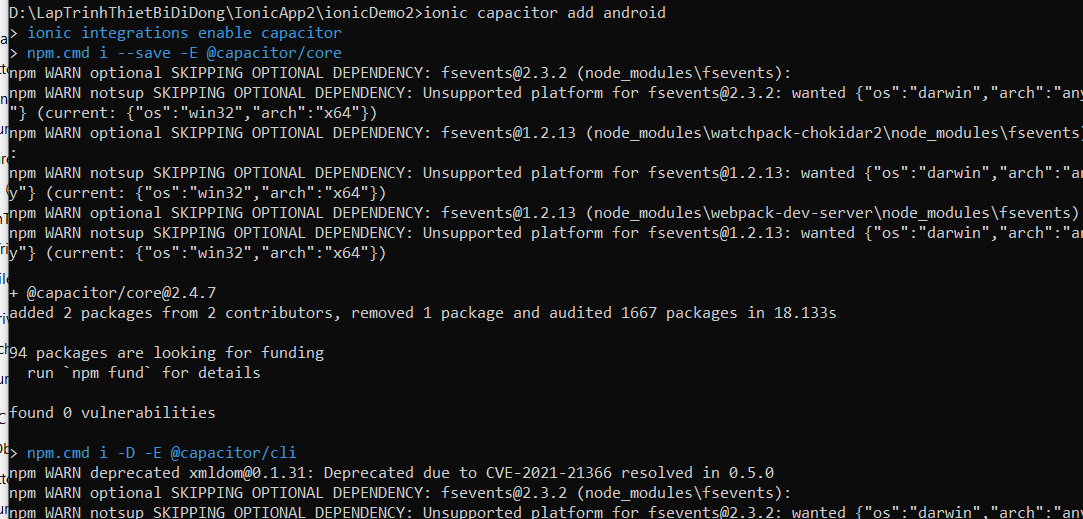
Sau khi tạo máy ảo nhớ chỉnh lại cài đặt của máy ảo 1 chút chuyển sang Developer Mode và cho phép kết nối USD debugging chúng ta cũng có thể làm điều này tương tự với máy thật khi kết nối với máy tính của mình thông qua cổng USB

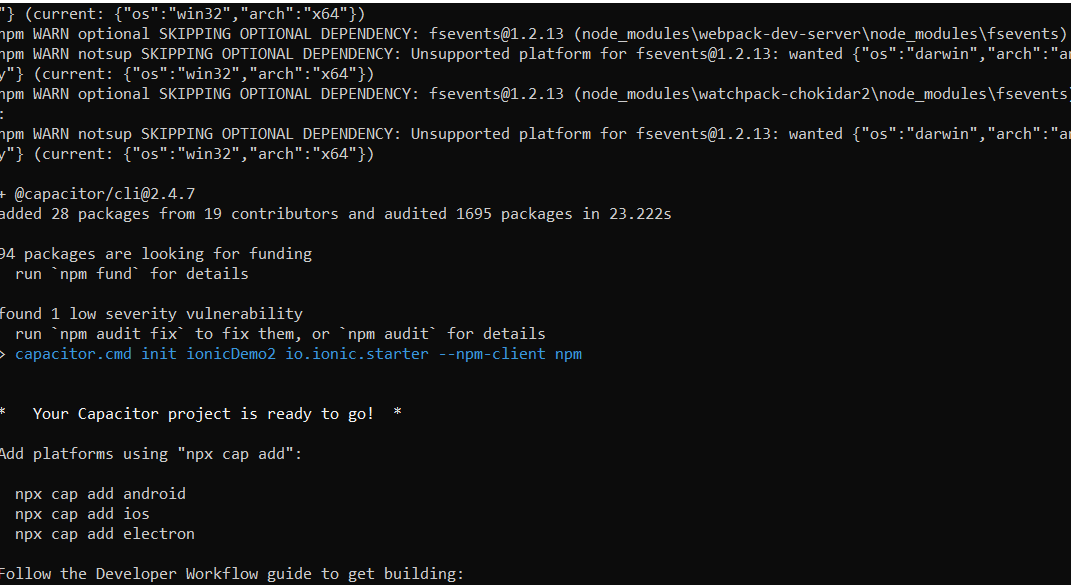


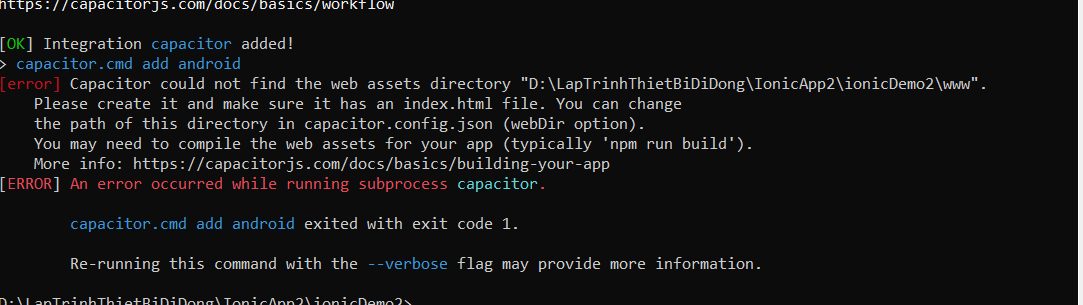
Tiếp theo trên cửa sổ Terminal cd vào thư mục project mình đã tạo và bắt đầu khởi tạo ứng dụng native app trên máy ảo của mình

ionic capacitor add android



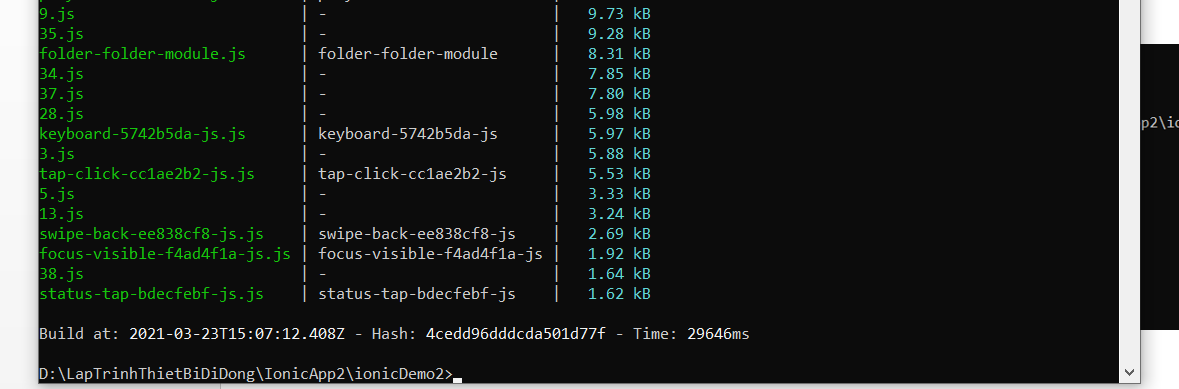




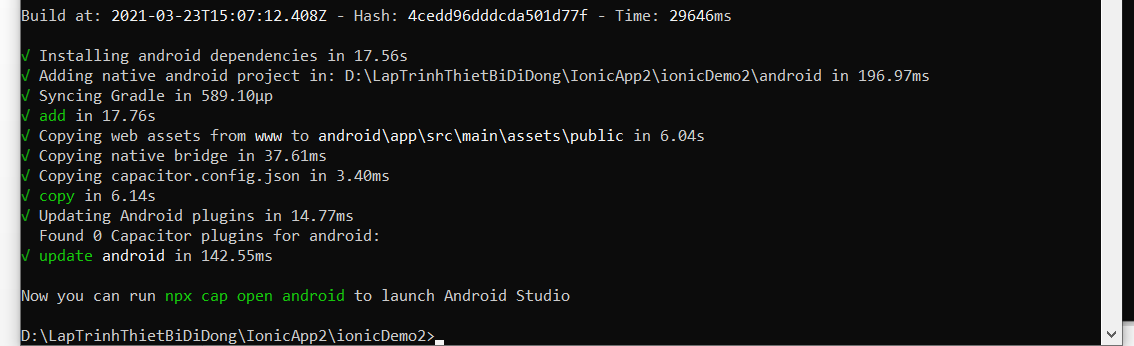


Chúng ta phải build tập tin www trên thư mục trước đã

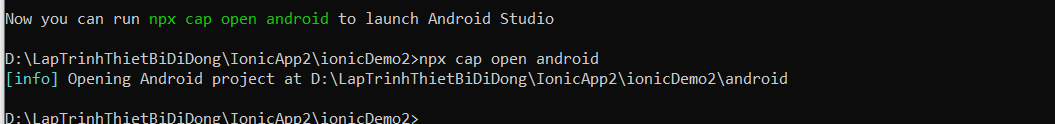
Gõ lệnh ionic build



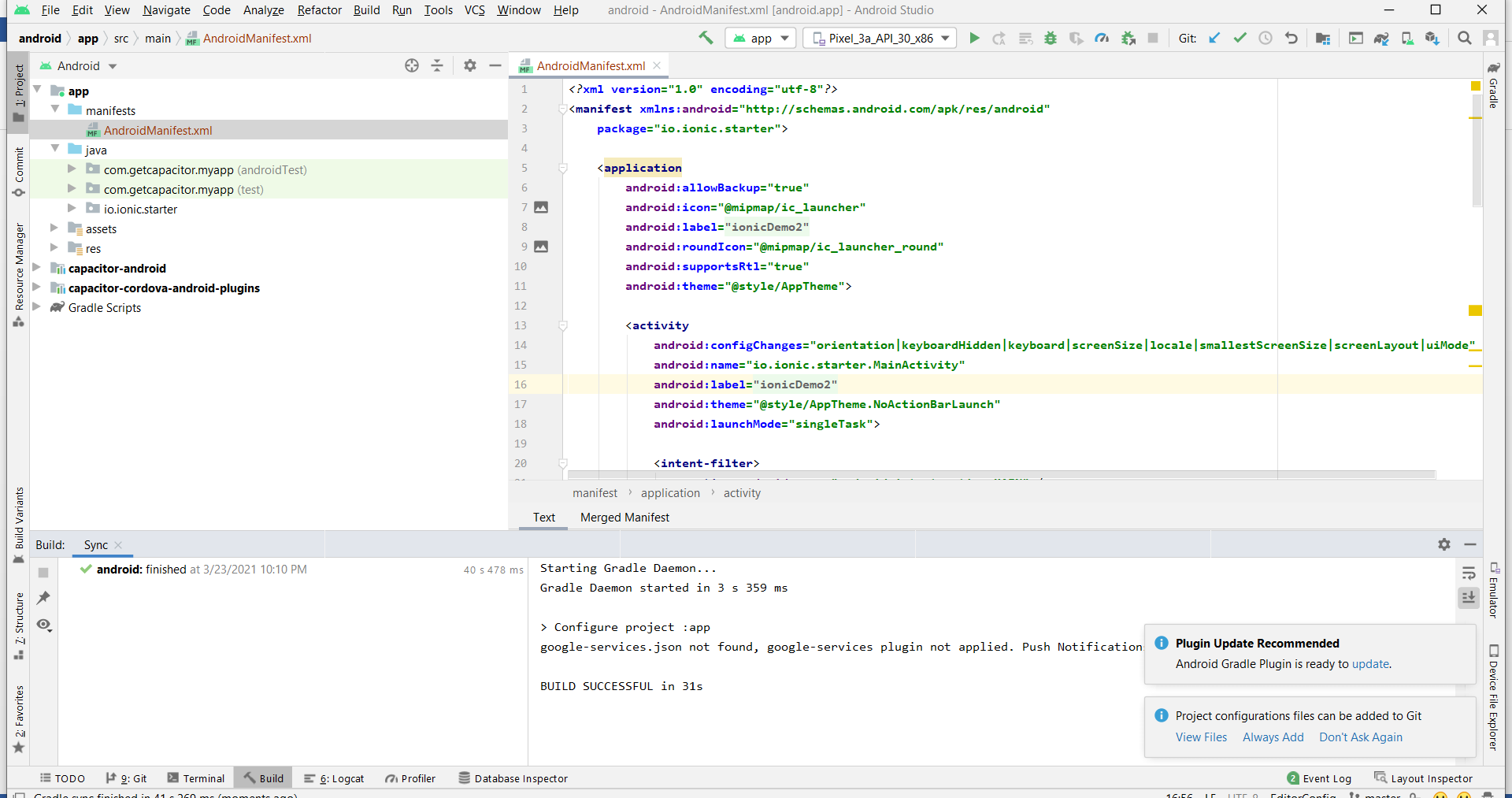
Gõ lệnh npx cap add android để bắt đầu tạo Ứng dụng Native



Gõ lệnh npx cap open android để chạy ứng dụng trên anroid studio



Để chạy được Project Ionic Native App trên android studio trước ta phải cài Đặt Android SDK 28.0.3 và 29 platform trên máy tính trước đã



Mình đã cài đặt rồi nên là nó diễn ra suôn sẻ

Bây giờ chỉ cần run app như mọi project Android bình thường thôi là xong

