

<b>Lời Mở Đầu .....</b>	<b>3</b>
<b>CHƯƠNG 1: TỔNG QUAN VỀ REACT NATIVE .....</b>	<b>4</b>
1.1. REACT NATIVE VÀ LỊCH SỬ PHÁT TRIỂN .....	4
1.2. KIẾN TRÚC REACT NATIVE .....	5
1.2.1. Native Code/Modules .....	5
1.2.2. Javascript Jvm .....	5
1.3. REACT NATIVE BRIDGE .....	6
1.4. LUỒNG TRONG REACT NATIVE .....	7
1.4.1. Main Thread (Native Queue) .....	7
1.4.2. Javascript Thread (Js Queue) .....	7
1.4.3. Custom Native Modules .....	7
1.5. VIEW MANAGER .....	7
1.6. VÒNG ĐỜI CỦA REACT COMPONENT .....	8
1.6.1. Constructor(Props) .....	8
1.6.2. Componentwillmount() .....	9
1.6.3. Render() .....	9
1.6.4. Componentdidmount() .....	9
1.6.5. Componentwillreceiveprops(Nextprops) .....	9
1.6.6. Shouldcomponentupdate(Nextprops, Nextstate) .....	9
1.6.7. Componentwillupdate(Nextprops, Nextstate) .....	10
1.6.8. Componentdidupdate(Prevprops, Prevstate) .....	10
1.6.9. Componentwillunmount() .....	10
1.7. ƯU NHƯỢC ĐIỂM CỦA REACT NATIVE .....	10
1.7.1. Ưu Điểm .....	10
1.7.2. Nhược Điểm .....	11
1.8. HƯỚNG DẪN CÀI ĐẶT REACT NATIVE .....	11
1.8.1. Cài Đặt Các Thành Phần Bắt Buộc Cho Môi Trường Phát Triển .....	11
1.8.2. Trên Môi Trường Phát Triển MacOS .....	12
1.8.3. Máy Ảo Phát Triển Android .....	13
1.8.4. Cài Đặt React Native Cli .....	16
1.8.5. Cài Đặt Môi Trường Android .....	16
1.9. CẤU TRÚC THƯ MỤC REACT NATIVE .....	19
<b>CHƯƠNG 2: XÂY DỰNG ỨNG DỤNG HỖ TRỢ XEM LỊCH CÁ NHÂN TRÊN NỀN TẢNG IOS .....</b>	<b>22</b>
2.1. MỤC ĐÍCH VÀ Ý NGHĨA CỦA ỨNG DỤNG HỖ TRỢ XEM LỊCH HỌC CÁ NHÂN .....	22
2.2. MÔ TẢ BÀI TOÁN .....	22
2.3. PHÂN TÍCH, THIẾT KẾ .....	23
2.3.1. Biểu Đồ Use Case .....	23
2.3.2. Xây Dựng Kịch Bản Cho Use Case .....	25
2.3.3. Biểu Đồ Trạng Thái .....	33
2.3.4. Biểu Đồ Trình Tự .....	34
2.3.5. Biểu Đồ Hoạt Động .....	36
<b>CHƯƠNG 3: CÀI ĐẶT ỨNG DỤNG VÀ CHẠY THỬ NGHIỆM .....</b>	<b>37</b>

3.1.	CÀI ĐẶT ỨNG DỤNG .....	37
3.2.	CHẠY THỬ ỨNG DỤNG .....	37
3.2.1.	Giao Diện Lấy Cập Nhập Lịch Học .....	37
3.2.2.	Giao Diện Xem Lịch Âm Lịch Và Dương Lịch.....	38
3.2.3.	Giao Diện Xem Lịch Theo Lịch Biểu .....	39
3.2.4.	Giao Diện Xem Thời Gian Ra Vào Lớp .....	40
3.2.5.	Giao Diện Chi Tiết Lịch Học .....	41
3.2.6.	Giao Diện Sửa Ghi Chú .....	42
<b>Kết Luận</b>	.....	<b>43</b>
<b>Nhận Xét Của Giáo Viên</b>	.....	<b>44</b>
<b>Tài Liệu Tham Khảo</b>	.....	<b>45</b>

## LỜI MỞ ĐẦU

Em xin cảm ơn sự hướng dẫn tận tình của các thầy cô giáo trong bộ môn Kỹ Thuật Phần Mềm và dưới sự hướng dẫn trực tiếp của thầy: **TS. NGUYỄN VĂN NÚI** đã cung cấp cho em những thông tin cần thiết cùng với việc khảo sát, phân tích các yêu cầu thực tế, em đã xây dựng hoàn thành đề tài “*Tìm hiểu về React Native và xây dựng ứng dụng hỗ trợ xem lịch học cá nhân trên nền tảng IOS*”.

Qua một thời gian thiết kế và xây dựng đến nay nội dung của đề tài cơ bản đã hoàn thành. Mặc dù đã có sự cố gắng và được sự giúp đỡ tận tình của thầy cô trong bộ môn Kỹ Thuật Phần Mềm, nhưng do thời gian và kinh nghiệm thực tế còn nhiều hạn chế nên em không tránh khỏi những thiếu sót và khuyết điểm.

Vì vậy em rất mong có được sự đóng góp của các thầy cô trong Khoa Công nghệ thông tin, cũng như ý kiến đóng góp của các bạn để chương trình của em được hoàn thiện hơn.

*Em xin chân thành cảm ơn!*

## CHƯƠNG 1: TỔNG QUAN VỀ REACT NATIVE

### 1.1. React native và lịch sử phát triển

Sự phát triển về CNTT ngày càng khủng khiếp. Những công nghệ mới đã và đang làm thay đổi thế giới một cách nhanh chóng. Từ lúc mới ra đời cho đến nay smartphone đã có những bước tiến mạnh mẽ vì vậy mà những công nghệ kèm theo cũng đòi hỏi những nhà phát triển phần mềm viết ra nhiều phần mềm hơn để có thể sử dụng được trên smartphone chứ không còn đơn thuần là những ứng dụng nhắn tin gọi điện thông thường.

Đồng thời giá smartphone đang ngày càng rẻ qua từng năm, vì các dòng smartphone giá rẻ đến từ các quốc gia phát triển và đông dân như Ấn Độ, Trung Quốc. Cùng với đó công nghệ sản xuất càng dần hoàn thiện hơn kéo giá smartphone xuống tới các tầng lớp phổ thông của xã hội. Từ đó smartphone trở thành một thiết bị có thể được sở hữu dễ dàng.

React Native là một framework do công ty công nghệ nổi tiếng Facebook phát triển nhằm mục đích giải quyết bài toán hiệu năng của Hybrid và bài toán chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động. Chúng ta sẽ build được ứng dụng Native, và chúng ta cũng có thể build ứng dụng đó một cách đa nền tảng (multi-platform) chứ không phải là một “mobile web app”, không phải là “HTML5 app”, và cũng không phải là một “hybrid app” hay cũng không chỉ build trên iOS hay Android mà chúng ta build và chạy được cả hai hệ sinh thái luôn, sợ chưa!!! Một điểm hay của React Native nữa là giảm chi phí recompile của Native bằng cách sử dụng Hot-Loading tức là bạn không cần phải build lại ứng dụng từ đầu nên việc chỉnh sửa diễn ra rất nhanh chóng. Giúp cho lập trình viên có thể thấy được những chỉnh sửa của họ một cách nhanh chóng trực quan, không còn phải bỏ quá nhiều thời gian trong việc build và run ứng dụng nữa.

Và điểm lợi hại kế tiếp của React Native đó chính là chúng ta chỉ cần sử dụng JS để phát triển được một ứng dụng di động hoàn chỉnh và rồi còn cả kết hợp với code native như Swift, Java, v.v...

Ngày nay, React Native đang dần trở nên phổ biến. Hàng nghìn ứng dụng được tạo ra có dính dáng đến React native. Những ứng dụng lớn sử dụng React Native như Facebook, Instagram, Skype, Pinterest, Baidu Mobile, Uber và nhiều công ty khác cũng đã chọn React native để xây dựng ứng dụng của họ.

React Native mở ra cơ hội việc làm rất nhiều cùng với cộng đồng hỗ trợ rất mạnh mẽ và đông đảo áp đảo so với những công cụ lập trình di động đa nền tảng khác hiện nay

Ứng dụng RN truy cập trực tiếp đến toàn bộ APIs của ngôn ngữ Native. Như vậy một ứng dụng được xây dựng bằng react native cho hiệu năng và cảm nhận tương tự như một ứng dụng native.

## **1.2. Kiến trúc React Native**

### **1.2.1. Native Code/Modules**

Đối với ios toàn bộ native code được viết bằng Objective C hoặc Swift, còn với Android sẽ được viết bằng Java. Nhưng để viết một ứng dụng React Native, chúng ta sẽ không bao giờ phải viết bằng native code của Android hay Ios.

### **1.2.2. Javascript JVM**

Máy ảo JS sẽ chạy toàn bộ JS code của chúng ta. Trên các máy ảo Ios hay Android và trên các thiết bị React Native sử dụng JavaScriptCore. Các công cụ JavaScript hỗ trợ Safari. JavaScriptCore là một công cụ mã nguồn mở xây dựng cho Webkit. Với IOS, React Native sử dụng JavaScriptCore được cung cấp bởi nền tảng Ios, được giới thiệu lần đầu tiên trên ios 7.

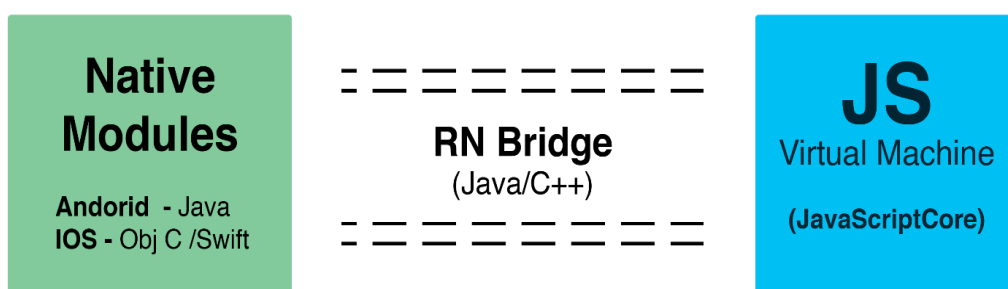
Với Android React Native đóng gói JavaScriptCore cùng với ứng dụng. Do đó sẽ làm tăng kích thước của app.

Trong trường hợp sử dụng chế độ debug Chrome, code JavaScript sẽ chạy trên Chrome chứ không chạy trên thiết bị và giao tiếp với native code thông qua Websocket, sử dụng với V8 engine. Điều này cho phép chúng ta quan sát được

nhiều thông tin hơn trên công cụ Chrome debugging như network requests, console logs, ...

### 1.3. React Native Bridge

React native bridge là một cầu nối C++/Java giúp giao tiếp giữa luồng native và luồng Javascript. Một giao thức custom sẽ được sử dụng cho việc gửi các messages giữa 2 luồng



- Trong hầu hết các trường hợp, các lập trình viên sẽ viết toàn bộ ứng dụng React Native bằng JavaScript. Sau đó để chạy một ứng dụng như vậy, một câu lệnh cần được thực hiện thông qua CLI: `react-native run-android` hoặc `react-native run-ios`. Lúc này, React Native CLI sẽ sinh ra một node packager/bundler làm nhiệm vụ đóng gói JS code vào trong một file có tên `main.bundle.js`. Sau đó, mỗi lần ứng dụng React Native được khởi chạy, thành phần đầu tiên được nạp vào là ngõ vào của native. Native Thread tạo ra một máy ảo JS (JSVM) để chạy JS code đã được đóng gói. JS code đó là toàn bộ business logic của ứng dụng. Luồng Native từ đó sẽ gửi các messages thông qua RN bridge để khởi chạy một ứng dụng JS. Luồng Javascript được sinh ra bắt đầu gửi các lệnh, yêu cầu tới luồng native thông qua RN Bridge. Các lệnh bao gồm các view sẽ được nạp vào, các thông tin nhận từ phần cứng, ... Ví dụ, nếu luồng JS muốn một view và text được khởi tạo nó sẽ gửi yêu cầu vào trong một message và gửi nó qua Native thread để render chúng.
- Native thread sẽ thực hiện các hoạt động và gửi lại kết quả cho JS để đảm bảo các yêu cầu đã được thực hiện chính xác.

## **1.4. Luồng trong React Native**

### **1.4.1. Main thread (Native Queue)**

Đây là luồng chính được sinh ra đầu tiên để khởi chạy ứng dụng. Nó được nạp vào ứng dụng và khởi chạy JS thread để thực thi các đoạn mã Javascript. Luồng native cũng sẽ lắng nghe các sự UI như ‘press’, ‘touch’... Các sự kiện này được chuyển tới luồng JS qua RN Bridge. Mỗi lần Javascript nạp, luồng JS gửi thông tin những thứ cần thiết để hiện lên màn hình. Những thông tin này được sử dụng bởi shadow node thread để tính toán các bố cục. The shadow thread đơn giản giống như một công cụ toán học đưa ra các quyết định cuối cùng cách tính toán vị trí của view. Các lệnh này sau đó được truyền trả lại cho main thread để hiển thị view lên giao diện.

### **1.4.2. Javascript thread (JS Queue)**

Đây là hàng đợi luồng được chạy bởi main bundled JS thread đã nói ở bên trên. Luồng này chạy tất cả các business logic, ... code được viết bằng React Native.

### **1.4.3. Custom Native Modules**

Phần luồng được sinh ra bởi React Native, chúng ta có thể sinh ra các luồng trên custom native modules do chúng ta xây dựng để tăng hiệu năng của ứng dụng. Ví dụ: Animations được xử lý trong React Native bằng một luồng native riêng biệt thực hiện các việc từ luồng JS.

## **1.5. View Manager**

- View Manager là một native module giúp ánh xạ JSX Views tới Native views. Ví dụ:

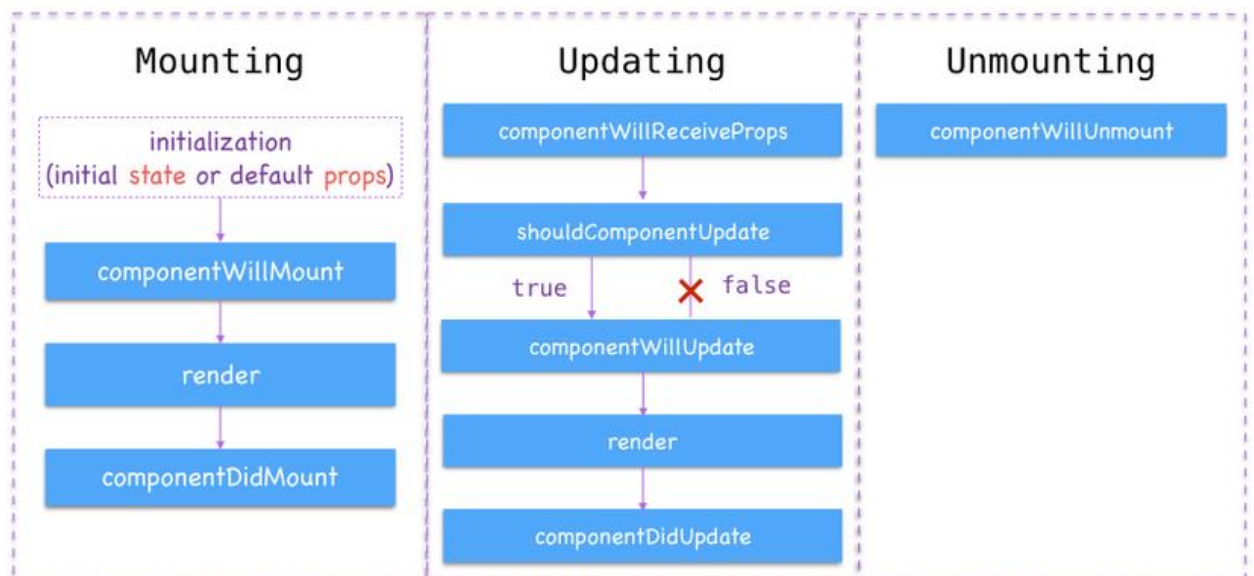
```
import React, { Component } from 'react';
import { Text, View, AppRegistry } from 'react-native';

class HelloWorldApp extends Component {
  render() {
    return (
      <View style={{padding:40}}>
        <Text>xin chào ICTU!</Text>
      </View>
    );
  }
}

export default HelloWorldApp;
AppRegistry.registerComponent('HelloWorldApp', () => HelloWorldApp);
```

- Khi chúng ta viết <Text />, Text View Manager sẽ thực thi new TextView(getContext()) trong trường hợp là Android View Managers là một class đơn giản kế thừa ViewManager trong Android và là một subclasses của RTCViewManager trong IOS

## 1.6. Vòng đời của React Componet



### 1.6.1. Contructor(props)

- Được gọi khi một thể hiện của component được tạo ra.
- Có thể dùng để khởi tạo state cho component.
- Cũng có thể dùng để "bind" các hàm của component.
- Nếu phải cài đặt hàm này thì phải khai báo 1 tham số props cho nó và phải gọi super(props) đầu tiên.



- Nếu không làm gì thì không phải cài đặt hàm này.

#### **1.6.2. `componentWillMount()`**

- Được gọi trước khi `render()`.
- Dùng để đăng ký các sự kiện toàn cục.
- Dựa vào các props để tính toán và set lại state

#### **1.6.3. `render()`**

- Hàm này bắt buộc phải có trong component().
- Trả về 1 đối tượng JSX (có thể lồng các đối tượng với nhau nhưng phải có 1 đối tượng gói tất cả các đối tượng lại) để hiển thị hoặc null / false nếu không muốn hiển thị gì.
- Không được gọi `setState()` trong hàm này (cũng như trong các hàm mà hàm này gọi đến), bởi khi gọi `setState()` thì hàm `render` sẽ được gọi => gây ra lặp vô hạn.

#### **1.6.4. `componentDidMount()`**

- Ngay sau khi hàm `render` được gọi đến lần đầu tiên chạy xong thì hàm này sẽ được chạy.
- Thường dùng để fetch dữ liệu từ server và sau đó `setState` để render dữ liệu ra.
- Đến đây thì các phần tử đã được sinh ra rồi, và có thể tương tác với DOM bằng JS trong hàm này.

#### **1.6.5. `componentWillReceiveProps(nextProps)`**

- Hàm này được chạy khi mà props của component đã được sinh ra có sự thay đổi.
- Phải gọi `setState()` nếu muốn render lại.

#### **1.6.6. `shouldComponentUpdate(nextProps, nextState)`**

- Được gọi trước `render`.

- Trả về true / false. Nếu false thì sẽ không render lại. Mặc định là true.

#### **1.6.7. `componentWillUpdate(nextProps, nextState)`**

- Được gọi ngay sau `shouldComponentUpdate()` nếu hàm này trả về true.
- Không gọi `setState()` trong hàm này bởi hàm này là để chuẩn bị update cho đối tượng chứ không phải tạo ra 1 update mới, sẽ tạo ra lặp vô hạn.
- Hàm render sẽ được gọi ngay sau hàm này.

#### **1.6.8. `componentDidUpdate(prevProps, prevState)`**

- Được gọi ngay sau `render()` từ lần render thứ 2 trở đi.
- Đây cũng là 1 cơ hội để thao tác với các phần tử DOM bằng JS.

#### **1.6.9. `componentWillUnmount()`**

- Được gọi khi 1 component được loại bỏ khỏi DOM.
- Thực hiện các thao tác dọn dẹp như huỷ các timer, loại bỏ các phần tử thừa, ...

### **1.7. Ưu nhược điểm của React Native**

#### **1.7.1. Ưu điểm**

- Dễ dàng học nếu đã biết về javascripts
- Hot reloading! Không cần tốn thời gian để recompiling. thông thường dev khi test các thay đổi mỗi lần code được viết. Để thực hiện được, app cần phải được tổng hợp lại và cài đặt hoặc trong 1 simulator hoặc 1 thiết bị thật sự. Với React Native, phần lớn thời gian, bạn không cần phải tổng hợp lại app mỗi lần có thay đổi. Bạn chỉ cần làm mới app trong simulator, emulator hoặc thiết bị. Thậm chí còn có 1 tính năng là Live Reload để tự động refresh app mỗi lần phát hiện 1 thay đổi trong code.
- Chạy được đa nền tảng khi Bạn không cần biết Objective-C, Swift hay Java. Với Javascript và JSX, bạn có thể xây dựng một ứng dụng hoạt động đa hệ điều hành rất tốt
- Một cộng đồng hỗ trợ rất lớn

- Nhiều năm trở lại đây, React Native đang trở lên rất phổ biến, nhiều nhà lập trình đang đóng góp để làm React Native tốt hơn mỗi ngày. Đặc biệt là nó được chống lưng bởi tập đoàn Facebook.
- React Native Github repro là một nguồn mở và có hàng nghìn cộng tác viên hoạt động rất năng nổ.
- Stack Overflow là một địa chỉ nơi mà có nhiều tài liệu và câu hỏi đã được trả lời về React Native.
- Cộng đồng rất lớn và đang dần phát triển hơn nữa. Nhiều vấn đề đã và đang được giải quyết và bạn sẽ không cần phải tốn thời gian để nghiên cứu lại trong suốt quá trình phát triển của bản thân.
- Hiệu năng tương đối ổn định.
- Tiết kiệm tiền.
- Team phát triển nhỏ.
- Tiết kiệm tiền.
- Team phát triển nhỏ.
- Ứng dụng tin cậy và ổn định.
- Trải nghiệm người dùng tốt hơn là hybrid app.

### **1.7.2. Nhược điểm**

- Vẫn đòi hỏi native code.
- Hiệu năng sẽ thấp hơn với app thuần native code.
- Bảo mật không cao do dựa trên JS.
- Quản lý bộ nhớ.
- Khả năng tùy biến cũng không thực sự tốt đối với một vài module

## **1.8. Hướng dẫn cài đặt React Native**

### **1.8.1. Cài đặt các thành phần bắt buộc cho môi trường phát triển**

- Môi trường phát triển MacOS
  - Máy ảo phát triển iOS

- Máy ảo phát triển Android
- Môi trường phát triển Windows
  - Máy ảo phát triển iOS (Hiện tại không hỗ trợ)
  - Máy ảo phát triển Android
- Môi trường phát triển Linux
  - Máy ảo phát triển iOS (Hiện tại không hỗ trợ)
  - Máy ảo phát triển Android

### 1.8.2. Trên môi trường phát triển MacOS

Để chạy ứng dụng trên iOS bạn cần phải cài đặt Node.js, Watchman, giao diện dòng lệnh React Native, và Xcode

Node, Watchman

Chạy các dòng lệnh dưới đây trong Terminal sau khi đã cài đặt Homebrew:

“brew install node”

“brew install watchman”

#### ■ React Native CLI

Node.js đã cung cấp npm, và chúng ta sẽ sử dụng nó để cài đặt giao diện dòng lệnh cho React Native. Chạy lệnh sau ở cửa sổ Terminal:

“npm install -g react-native-cli”

#### ■ Xcode

cài đặt Xcode thông qua Mac App Store. Việc cài đặt Xcode sẽ đồng thời cài đặt máy ảo iOS và tất cả những công cụ cần thiết để bạn có thể build ứng dụng iOS.

#### ■ Kiểm tra việc cài đặt React Native

Sử dụng giao diện dòng lệnh React Native để tạo ra một project mới của React Native ví dụ như AwesomeProject sau đó chạy lệnh react-native run-ios bên trong thư mục project mới được tạo.

```
react-native init AwesomeProject  
cd AwesomeProject  
react-native run-ios
```

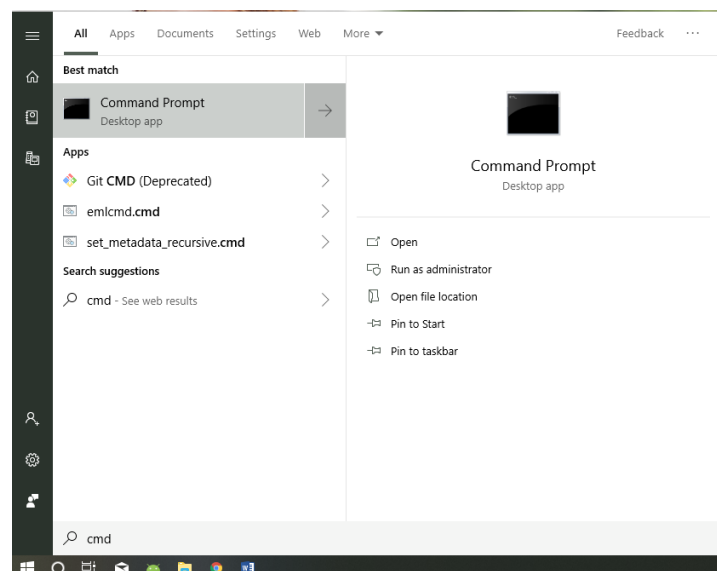
Bạn sẽ nhìn thấy ứng dụng mới của bạn được chạy trên máy ảo iOS. Câu lệnh `react-native run-ios` là cách để chạy ứng dụng của bạn. Bạn cũng có thể chạy ứng dụng này trong Xcode

### 1.8.3. Máy ảo phát triển Android

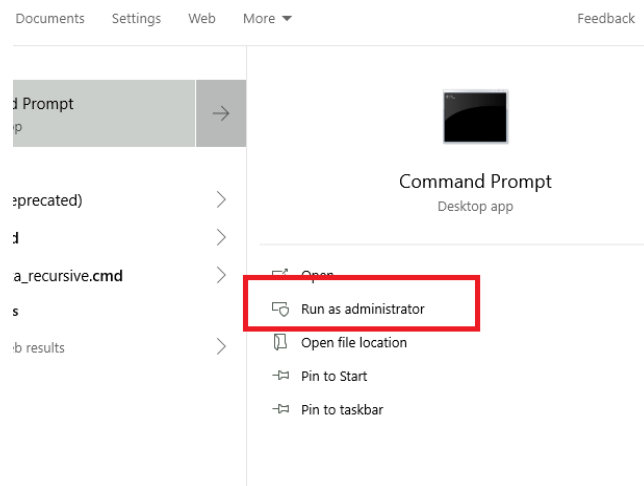
Để chạy ứng dụng trên Android bạn cần phải cài đặt Node.js, Watchman, giao diện dòng lệnh React Native, và Android Studio.

React Native yêu cầu một phiên bản bộ công cụ phát triển Java SE (JDK), cũng như Python 2. Cả hai đều có thể được cài đặt bằng Chocolatey

- Chocolatey
  - Cài với Command Prompt(cmd):
  - Để mở cmd bạn ấn phím windows trên bàn phím, và tìm kiếm với từ khóa cmd

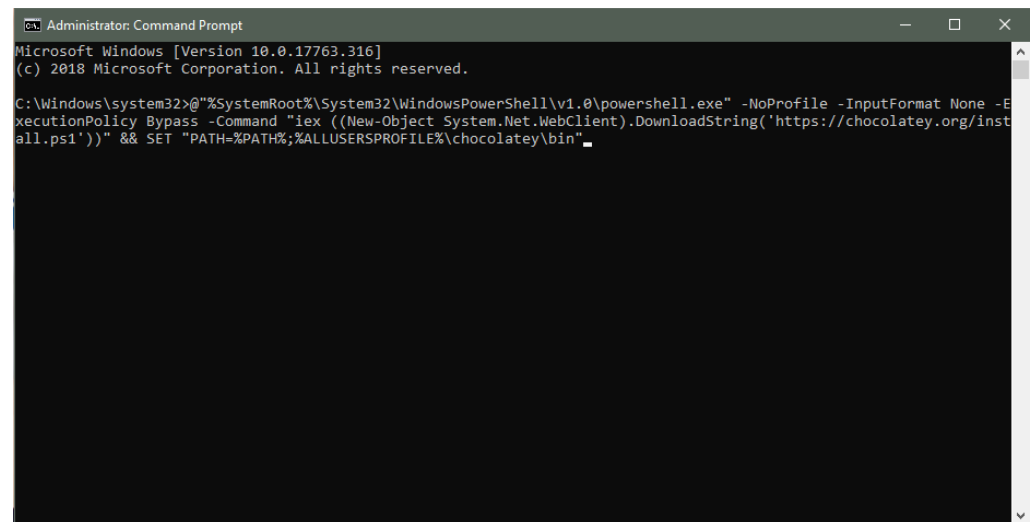


- Tiếp theo bạn chọn “Run as administrator”:



- Tiếp theo, bạn sao chép lệnh sau và dán vào cmd:

```
"@"%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.
exe" -NoProfile -InputFormat None -ExecutionPolicy Bypass -Command
"iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))" && SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"
```



- Bạn ấn enter và đợi cài Chocolatey

```
Administrator: Command Prompt
Restricting write permissions to Administrators
We are setting up the Chocolatey package repository.
The packages themselves go to 'C:\ProgramData\chocolatey\lib'
(i.e. C:\ProgramData\chocolatey\lib\yourPackageName).
A shim file for the command line goes to 'C:\ProgramData\chocolatey\bin'
and points to an executable in 'C:\ProgramData\chocolatey\lib\yourPackageName'.

Creating Chocolatey folders if they do not already exist.

WARNING: You can safely ignore errors related to missing log files when
upgrading from a version of Chocolatey less than 0.9.9.
'Batch file could not be found' is also safe to ignore.
'The system cannot find the file specified' - also safe.
chocolatey.nupkg file not installed in lib.
Attempting to locate it from bootstrapper.
PATH environment variable does not have C:\ProgramData\chocolatey\bin in it. Adding...
WARNING: Not setting tab completion: Profile file does not exist at
'C:\Users\Hiep\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1'.
Chocolatey (choco.exe) is now ready.
You can call choco from anywhere, command line or powershell by typing choco.
Run choco /? for a list of functions.
You may need to shut down and restart powershell and/or consoles
first prior to using choco.
Ensuring chocolatey commands are on the path
Ensuring chocolatey.nupkg is in the lib folder
C:\Windows\system32>
```

- Nếu cmd của bạn hiện như hình vậy là đã tải và cài đặt xong. Để kiểm tra xem đã thành công chưa bạn gõ lệnh “choco” nếu hiển thị như hình bên dưới là thành công:

```
Command Prompt
C:\Users\Hiep>choco
Chocolatey v0.10.11
Please run 'choco -?' or 'choco <command> -?' for help menu.
C:\Users\Hiep>
```

- Nếu bạn cài với PowerShell, bạn copy lệnh sau và dán vào PowerShell:

“Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))”

- Sau khi đã cài Chocolatey xong bắt đầu cài Node, Python2, JDK. Sử dụng lệnh:

“choco install -y nodejs.install python2 jdk8”

- và bắt đầu tiến hành cài đặt (chạy cmd hoặc PowerShell bằng quyền admin):

```

Installing 64 bit version
Installing nodejs.install...
nodejs.install has been installed.
nodejs.install may be able to be automatically uninstalled.
Environment Vars (like PATH) have changed. Close/reopen your shell to
see the changes (or in powershell/cmd.exe just type `refreshenv`).
The install of nodejs.install was successful.
Software installed as 'msi', install location is likely default.

python2 v2.7.15 [Approved]
python2 package files install completed. Performing other installation steps.
Downloading python2 64 bit
from 'https://www.python.org/ftp/python/2.7.15/python-2.7.15.amd64.msi'
Progress: 100% - Completed download of C:\Users\Hiep\AppData\Local\Temp\chocolatey\python2\2.7.15\python-2.7.15.amd64.msi (19.31 MB).
Download of python-2.7.15.amd64.msi (19.31 MB) completed.
Hashes match.
Installing python2...
python2 has been installed.
Installed to 'C:\Python27'
Adding C:\Python27 to PATH if needed
python2 may be able to be automatically uninstalled.
Environment Vars (like PATH) have changed. Close/reopen your shell to
see the changes (or in powershell/cmd.exe just type `refreshenv`).
The install of python2 was successful.
Software installed as 'msi', install location is likely default.
Progress: Downloading jdk8 8.0.201... 100%

jdk8 v8.0.201 [Approved]
jdk8 package files install completed. Performing other installation steps.
Downloading JDK from http://download.oracle.com/otn-pub/java/jdk/8u201-b09/42970487e3af4f5aa5bca3f542482c60/jdk-8u201-windows-x64.exe
Installing jdk8...
jdk8 has been installed.
PATH environment variable does not have C:\Program Files\Java\jdk1.8.0_201\bin in it. Adding...
Environment Vars (like PATH) have changed. Close/reopen your shell to
see the changes (or in powershell/cmd.exe just type `refreshenv`).
The install of jdk8 was successful.
Software installed as 'exe', install location is likely default.

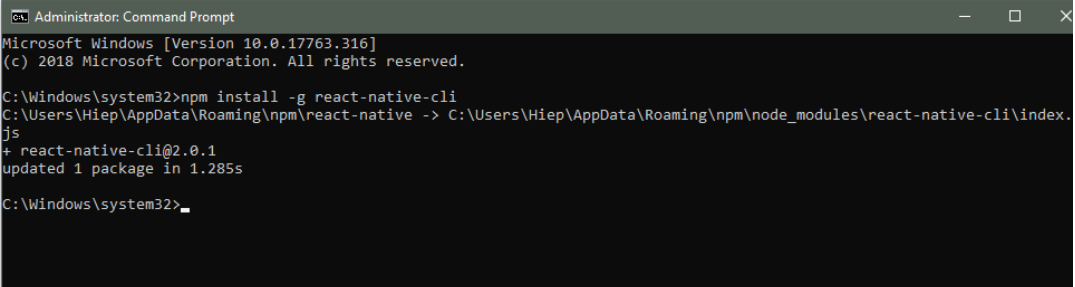
Chocolatey installed 3/3 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
PS C:\Windows\system32>

```

- Như vậy là đã cài đặt thành công jdk, node js, python2.

#### 1.8.4. cài đặt React Native CLI

- Chạy lệnh CMD sau để cài đặt React Native CLI: “npm install -g react-native-cli” trên cmd hoặc powerShell với quyền admin



```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.316]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>npm install -g react-native-cli
C:\Users\Hiep\AppData\Roaming\npm\react-native -> C:\Users\Hiep\AppData\Roaming\npm\node_modules\react-native-cli\index.js
+ react-native-cli@2.0.1
updated 1 package in 1.285s

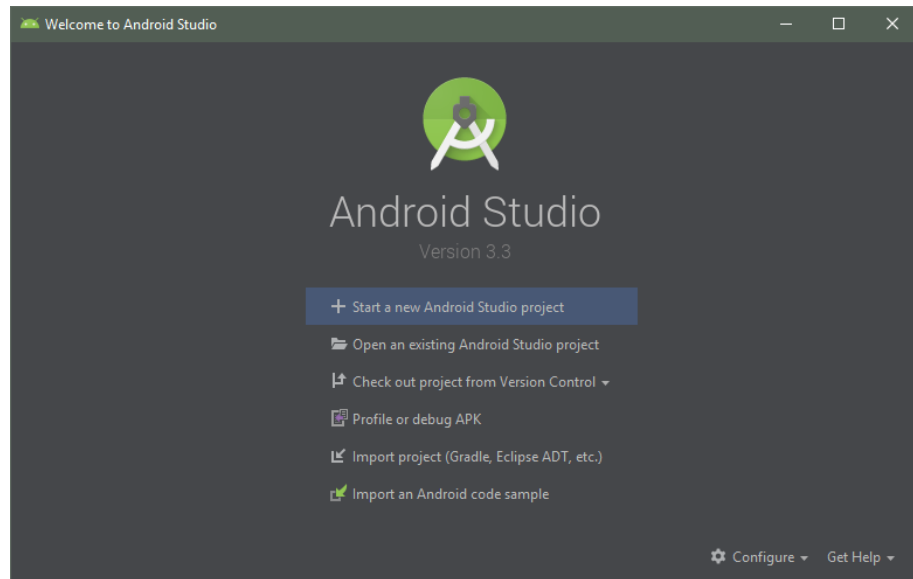
C:\Windows\system32>

```

#### 1.8.5. Cài đặt môi trường android

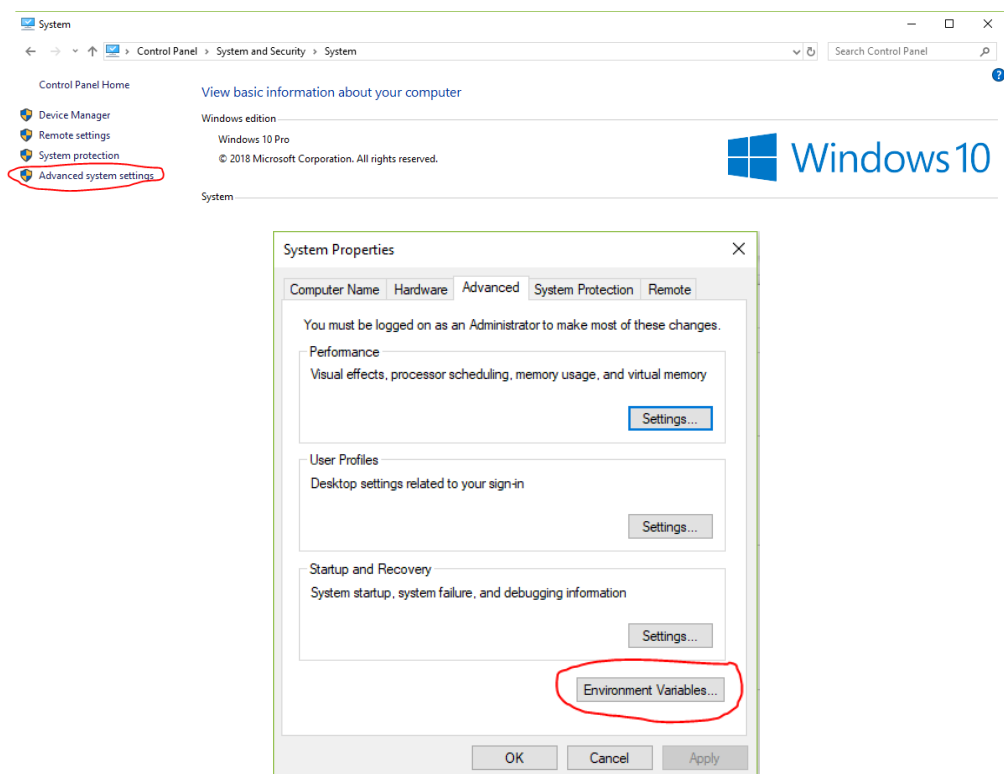
- Truy cập trang web: <https://developer.android.com/studio/index.html>, để tải android studio sau khi tải xong bạn tiến hành cài đặt

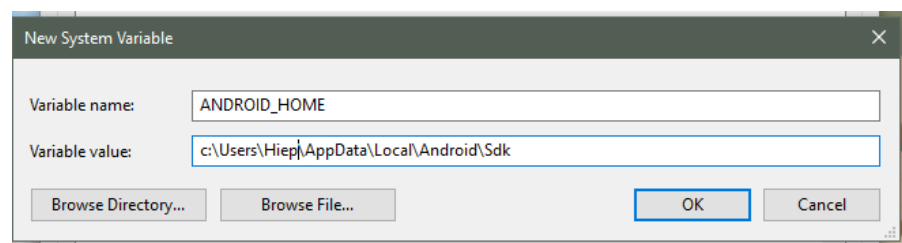
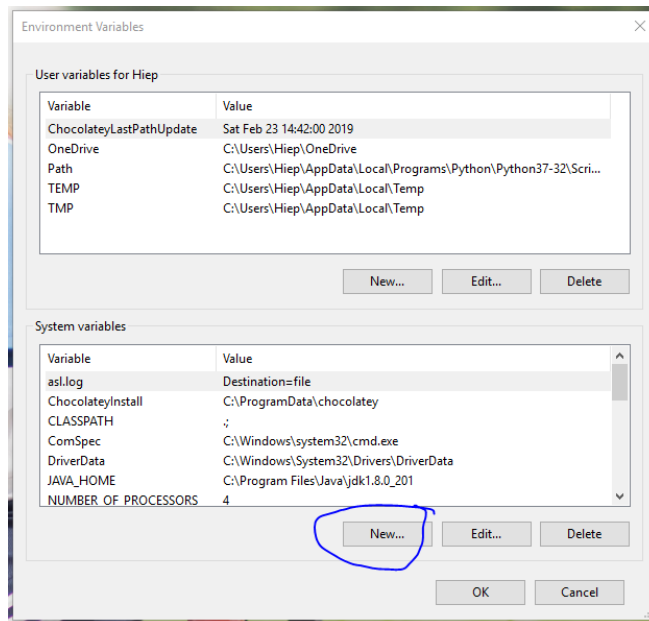




- Cài đặt biến môi trường `ANDROID_HOME`

Để React Native build app bạn phải chỉ định chỗ bạn cài Android Sdk ở đâu. Bạn làm theo hướng dẫn sau

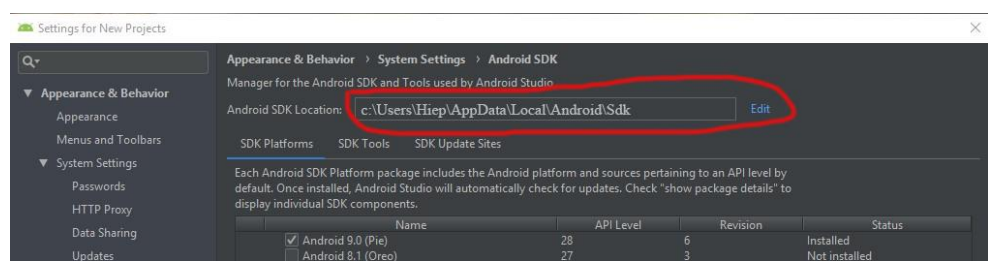




Ở dòng variable value bạn điền đường dẫn sdk mặc định thường là:

“c:\Users\YOUR\_USERNAME\AppData\Local\Android\Sdk”

Hoặc bạn có thể tìm thấy đường dẫn sdk thông qua android studio, vào mục setting -> Android SDK



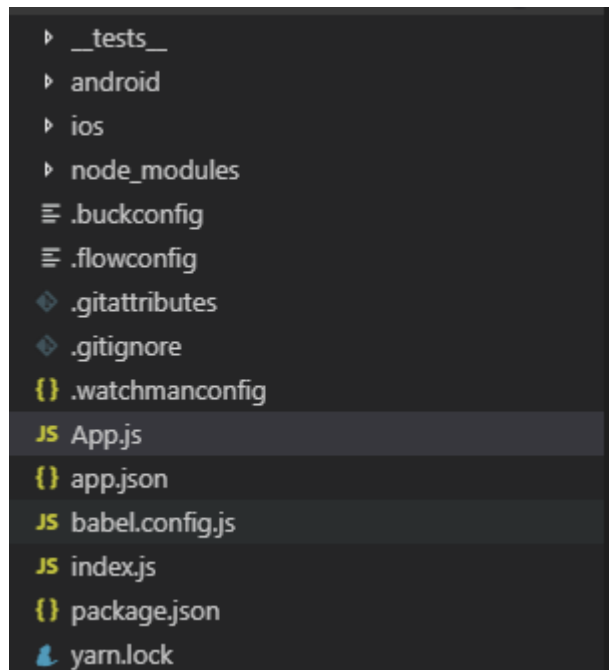
## - Kiểm tra việc cài đặt React Native

Sử dụng giao diện dòng lệnh React Native để tạo ra một project mới của React Native ví dụ như AwesomeProject sau đó chạy lệnh react-native run-ios bên trong thư mục project mới được tạo.

```
react-native init AwesomeProject  
cd AwesomeProject  
react-native run-android
```

Nếu như các cài đặt của bạn là đúng đắn, bạn sẽ nhìn thấy ứng dụng của bạn chạy trên máy ảo Android nhanh chóng. Câu lệnh `react-native run-android` là một cách để chạy ứng dụng của bạn. Bạn cũng có thể chạy ứng dụng này từ Android Studio

### 1.9. Cấu trúc thư mục React Native



Cấu trúc thư mục của react native

- Thư mục Android

Nơi chứa code native android các thông tin về ứng dụng trên Android, bạn có thể chạy project trên android bằng thư mục này

- Thư mục IOS

Nơi chứa code native IOS, các thông tin về ứng dụng trên IOS, bạn có thể chạy trên xcode bằng thư mục này

- Thư mục node\_modules

Đây là thư mục chứa các thư viện của react native, các thư viện cơ bản khi bạn mới tạo hay các thư viện bạn tải về và thêm vào project đều được nằm ở đây. Các thư viện có thể được code bằng ngôn ngữ javascripts hay là các cầu nối giữa react native và các modules bên native

- File App.js

Đây là file được chạy sau khi chạy file index.js khi bạn mới khởi tạo ứng dụng

```
9
10 import React, {Component} from 'react';
11 import {Platform, StyleSheet, Text, View} from 'react-native';
12
13 const instructions = Platform.select({
14   ios: 'Press Cmd+R to reload,\n' + 'Cmd+D or shake for dev menu',
15   android:
16     'Double tap R on your keyboard to reload,\n' +
17     'Shake or press menu button for dev menu',
18 });
19
20
21 export default class App extends Component {
22   render() {
23     return (
24       <View style={styles.container}>
25         <Text style={styles.welcome}>Welcome to React Native!</Text>
26         <Text style={styles.instructions}>To get started, edit App.js</Text>
27         <Text style={styles.instructions}>{instructions}</Text>
28       </View>
29     );
30   }
31 }
32
33 const styles = StyleSheet.create({
34   container: {
35     flex: 1,
36     justifyContent: 'center',
37     alignItems: 'center',
38     backgroundColor: '#F5FCFF',
39   },
40   welcome: {
41     fontSize: 20,
42     textAlign: 'center',
43     margin: 10,
44   },
45   instructions: {
46     textAlign: 'center',
47     color: '#333333',
48     marginBottom: 5,
49   },
50 });
51
```

- File index.js

Là file được chạy đầu tiên khi ứng dụng khởi chạy, đây là nơi đăng ký ứng dụng và xác định file nào sẽ được chạy tiếp theo

```

1  /**
2   * @format
3   * @lint-ignore-every XPLATJSCOPYRIGHT1
4   */
5
6  import {AppRegistry} from 'react-native';
7  import App from './App';
8  import {name as appName} from './app.json';
9
10 AppRegistry.registerComponent(appName, () => App);
11

```

- Package.json

Đây là file khai báo các thư viện được sử dụng trong project, các thông tin về react native, cũng như các thông tin về ứng dụng

```

{
  "name": "Demo",
  "version": "0.0.1",
  "private": true,
  "scripts": {
    "start": "node node_modules/react-native/local-cli/cli.js start",
    "test": "jest"
  },
  "dependencies": {
    "react": "16.6.3",
    "react-native": "0.58.6"
  },
  "devDependencies": {
    "babel-core": "7.0.0-bridge.0",
    "babel-jest": "24.5.0",
    "jest": "24.5.0",
    "metro-react-native-babel-preset": "0.53.1",
    "react-test-renderer": "16.6.3"
  },
  "jest": {
    "preset": "react-native"
  }
}

```

## CHƯƠNG 2: XÂY DỰNG ỨNG DỤNG HỖ TRỢ XEM LỊCH CÁ NHÂN TRÊN NỀN TẢNG IOS

### 2.1. Mục đích và ý nghĩa của ứng dụng hỗ trợ xem lịch học cá nhân

Thời sinh viên lịch học cá nhân là thứ rất quan trọng đối với sinh viên, từ yêu cầu thiết yếu đó em đã có ý tưởng xây dựng ứng dụng hỗ trợ lịch học cá nhân. Ứng dụng được thực hiện dựa trên hai mục đích sau:

- Thứ nhất là nghiên cứu công nghệ lập trình lập trình ứng dụng với React native
- Thứ hai là phát triển ứng dụng có tính thực tiễn cao, có khả năng triển khai vào thực tế, giúp mọi người dùng có nhu cầu quản lý lịch cá nhân có thể thao tác dễ dàng, tiện lợi và ổn định

Trong quá trình thực hiện bài thực tập chuyên ngành “Tìm hiểu về React Native và xây dựng ứng dụng hỗ trợ xem lịch học cá nhân trên nền tảng IOS” em vẫn còn thiếu kinh nghiệm, vì vậy không thể tránh khỏi những sai sót, nhầm lẫn trong bài, mong cô giáo và các bạn đưa ra ý kiến để giúp em rút ra bài học để làm dự án sau tốt hơn.

### 2.2. Mô tả bài toán

Khi người dùng bắt đầu sử dụng ứng dụng , người dùng cần tải thời khoá biểu cá nhân tại trang web đăng ký tín chỉ , chọn và tải thời khoá biểu dưới dạng “Hiển thị theo thứ học”.

Người dùng tải lên thời khoá biểu dưới dạng thứ học tại trang web <http://tkb.sinhvien.xyz/upload.php>, lịch học sẽ được lưu lại trên hệ thống

Khi sử dụng người dùng mới sử dụng chỉ cần nhập mã sinh viên của mình trên ứng dụng và có thể xem lịch học cá nhân dưới dạng tuần hoặc dạng tháng , các lần sau ứng dụng đã tự động lưu lại lịch học cá nhân của người dùng nên

không cần nhập mã sinh viên để lấy lịch học, trừ trường hợp người dùng thay đổi lịch học thì chỉ cần vào form thêm lịch học và nhập mã sinh viên

Người dùng có thể thêm ghi chú ở các ngày trên lịch, người dùng chỉ cần nhập tiêu đề ghi chú hoặc nội dung ghi chú và ấn lưu là có thể thực hiện ghi chú thành công. Người dùng có thể sửa ghi chú, hoặc xóa ghi chú khi cần thiết

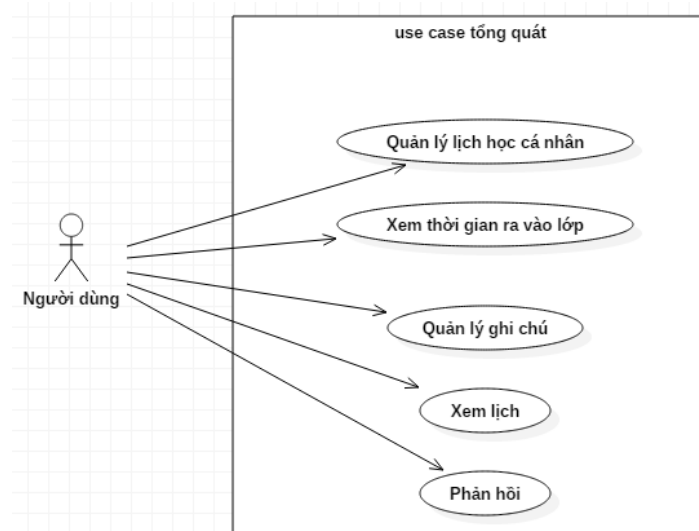
Ngoài ra người dùng có thể xem được thời gian ra vào lớp của ở chế độ có kết nối mạng internet hoặc không có kết nối mạng internet

Ứng dụng dễ dàng sử dụng, tiện lợi và tính ổn định cao

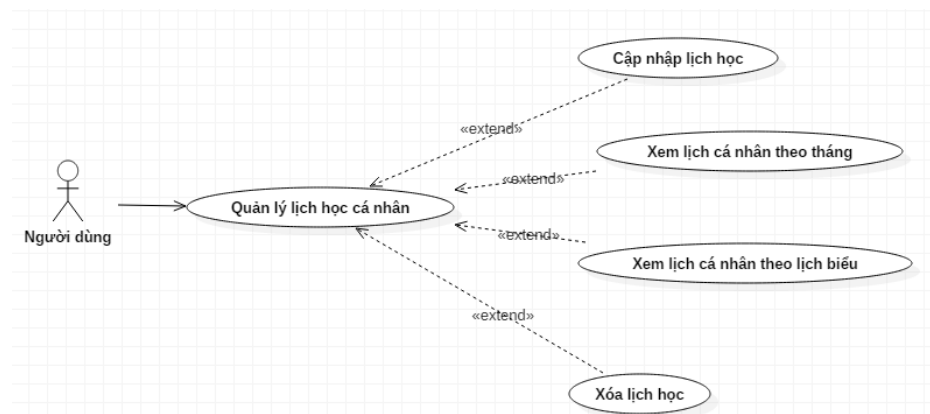
## 2.3. Phân tích, thiết kế

### 2.3.1. Biểu đồ use case

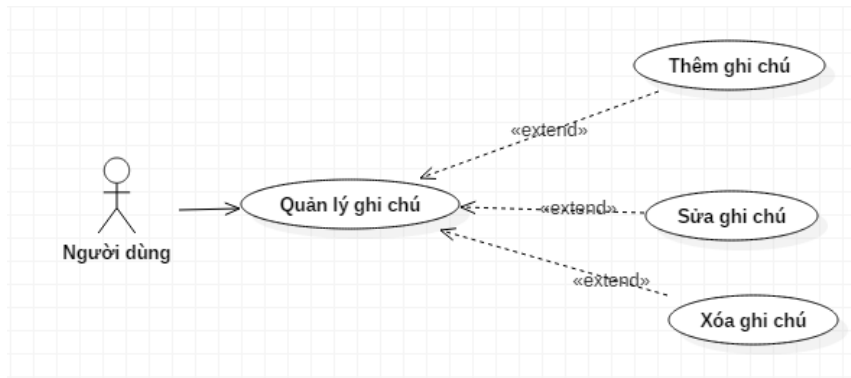
#### 2.3.1.1. Biểu đồ use case tổng quát



#### 2.3.1.2. Biểu đồ UC quản lý lịch học cá nhân



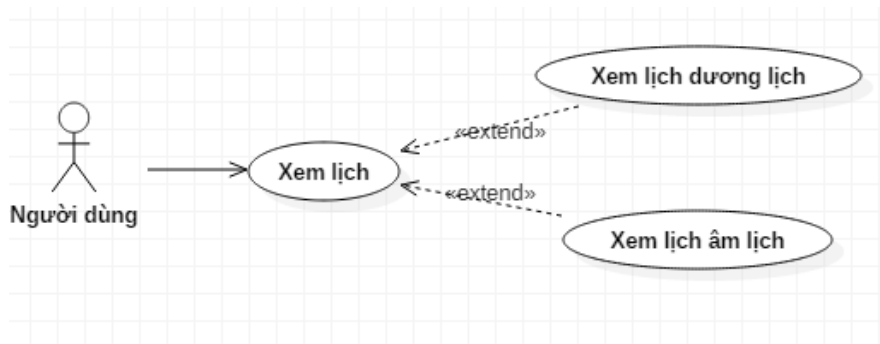
### 2.3.1.3. Biểu đồ UC Quản lý ghi chú



### 2.3.1.4. Biểu đồ UC xem lịch thời gian ra vào lớp



### 2.3.1.5. Biểu đồ UC xem lịch





### 2.3.2. Xây dựng kịch bản cho use case

#### 2.3.2.1. Biểu diễn kịch bản cho UC chức năng thêm lịch học cá nhân

Tên use case	Cập nhập lịch học cá nhân
Tác nhân chính	Người dùng
Tiền điều kiện	
Đảm bảo thành công	Lấy được lịch học cá nhân
Kích hoạt	Người dùng ấn lấy lịch học

Chuỗi sự kiện chính:

1. Người dùng vào form lấy lịch học
2. Người dùng nhập mã sinh viên
3. Người dùng ấn nút “Lấy lịch học”
4. Hệ thống gửi mã sinh viên đến web services để kiểm tra
5. Web services trả về lịch học cho ứng dụng
6. Ứng dụng chuyển người dùng về form xem lịch cá nhân và hiển thị lịch học

Ngoại lệ:

- 2.1 Người dùng không nhập mã “Yêu cầu nhập mã sinh viên”
- 2.2 Người dùng nhập mã sinh viên không tồn tại trong hệ thống, hệ thống yêu cầu: “Mã sinh viên chưa tồn tại trên hệ thống yêu cầu kiểm tra lại”

### 2.3.2.2. Biểu diễn kịch bản cho UC chức năng xóa lịch học cá nhân

Tên Use Case	xóa lịch học cá nhân
Tác nhân chính	Người dùng
Tiền điều kiện	Đã lấy lịch học cá nhân thành công
Thành công	Hiện thị thông báo xóa thành công
Kích hoạt	Người dùng ấn vào nút “xóa lịch học”
Chuỗi sự kiện chính:	
1. Người dùng ấn vào nút xóa lịch học	
2. Hệ thống thông báo “bạn có muốn xóa lịch học cá nhân hay không?”	
3. Người dùng chọn “Đồng ý”	
4. Hệ thống xóa lịch học cá nhân thành công	
Ngoại lệ	
2a: Người dùng chọn “Hủy”	

**2.3.2.3. Biểu diễn kịch bản cho UC chức năng xem lịch học cá nhân theo lịch biểu**

Tên Use Case	xóa lịch học cá nhân
Tác nhân chính	Người dùng
Tiền điều kiện	Đã lấy lịch học cá nhân thành công
Thành công	Hiển thị lịch học cá nhân và ghi chú theo tháng
Kích hoạt	Người dùng ấn vào icon hiển thị lịch học theo tuần
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Người dùng ấn vào icon hiển thị lịch học theo tuần</li> <li>2. Hệ thống thống hiển thị lịch học cá nhân và ghi chú dưới dạng tuần</li> </ol>	

**2.3.2.4. Biểu diễn kịch bản cho UC chức năng xem lịch học cá nhân theo tháng**

Tên Use Case	xem lịch học cá nhân theo tháng
Tác nhân chính	Người dùng
Tiền điều kiện	Đã lấy lịch học cá nhân thành công
Thành công	Hiển thị lịch học cá nhân và ghi chú theo tháng
Kích hoạt	Người dùng ấn vào icon hiển thị lịch học theo tháng

Chuỗi sự kiện chính:

1. Người dùng ấn vào icon hiển thị lịch học theo tháng
2. Hệ thống thông hiển thị lịch học cá nhân và ghi chú dưới dạng theo tháng

#### 2.3.2.5. Biểu diễn kịch bản cho UC chức năng Thêm ghi chú

Tên Use Case	Thêm ghi chú
Tác nhân chính	Người dùng
Tiền điều kiện	
Kích hoạt	Người dùng ấn vào icon lưu
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"><li>1. Người dùng ấn vào icon thêm ghi chú</li><li>2. Form thêm ghi chú được hiển thị</li><li>3. Người dùng nhập nội dung ghi chú và tiêu đề ghi chú</li><li>4. Người dùng ấn icon lưu</li></ol> <p>Hệ thống thông báo thành công và chuyển về màn hình lịch</p>	
<p>Ngoại lệ:</p> <p>2a: Người dùng bỏ trống cả hai trường nội dung và tiêu đề ghi chú. Hệ thống yêu cầu nhập ít nhất 1 trong hai trường</p>	

#### 2.3.2.6. Biểu diễn kịch bản cho UC chức năng sửa ghi chú

Tên Use Case	Sửa ghi chú
Tác nhân chính	Người dùng
Tiền điều kiện	
Thành công	Hiển thị thông báo thêm ghi chú thành công
Kích hoạt	Người dùng ấn vào icon sửa
Chuỗi sự kiện chính:	
1. Người dùng ấn vào icon sửa ghi chú	
2. Người dùng nhập nội dung ghi chú và tiêu đề ghi chú cần sửa	
3. Người dùng ấn icon sửa	
4. Hệ thống thông báo thành công và chuyển về màn hình lịch	
Ngoại lệ:	
2a: Người dùng bỏ trống cả hai trường nội dung và tiêu đề ghi chú. Hệ thống yêu cầu nhập ít nhất 1 trong hai trường	

#### 2.3.2.7. Biểu diễn kịch bản cho UC chức năng xóa ghi chú

Tên Use Case	xóa ghi chú
Tác nhân chính	Người dùng
Tiền điều kiện	Ghi chú muốn xóa tồn tại
Thành công	Hiển thị thông báo xóa thành công và chuyển về màn hình chính
Kích hoạt	Người dùng ấn vào nút “xóa lịch học”
Chuỗi sự kiện chính:  1. Người dùng ấn vào icon xóa ghi chú  2. Hệ thống thông báo “bạn có muốn xóa ghi chú hay không?”  3. Người dùng chọn “Đồng ý”  4. Hệ thống xóa lịch ghi chú thành công	
Ngoại lệ  3a: Người dùng chọn “Hủy”	

**2.3.2.8. Biểu diễn kịch bản cho UC chức năng xem thời gian ra vào lớp online**

Tên Use Case	xem thời gian ra vào lớp online
Tác nhân chính	Người dùng
Tiền điều kiện	1. Đã lấy lịch học cá nhân thành công 2. Điện thoại có kết nối tới mạng internet
Thành công	Hiển thị thời gian ra vào lớp
Kích hoạt	Người dùng ấn vào xem lịch học cá nhân
Chuỗi sự kiện chính:  1. Người dùng ấn vào xem lịch học cá nhân  2. Hiển thị lịch học cá nhân	
Ngoại lệ:  1. Điện thoại không có kết nối tới internet	

**2.3.2.9. Biểu diễn kịch bản cho UC chức năng xem thời gian ra vào lớp offline**

Tên Use Case	xem thời gian ra vào lớp online
Tác nhân chính	Người dùng
Tiền điều kiện	1. Đã lấy lịch học cá nhân thành công

	2. Điện thoại có kết nối tới mạng internet
Thành công	Hiển thị thời gian ra vào lớp
Kích hoạt	Người dùng ấn vào xem lịch học cá nhân
Chuỗi sự kiện chính:  3. Người dùng ấn vào xem lịch học cá nhân  4. Hiển thị lịch học cá nhân	
Ngoại lệ:  1. Điện thoại không có kết nối tới internet	

#### **2.3.2.10. Biểu diễn kịch bản cho UC chức năng xem lịch âm lịch**

Tên Use Case	Xem lịch âm lịch
Tác nhân chính	Người dùng
Tiền điều kiện	
Thành công	Hiển thị lịch âm lịch
Kích hoạt	Người dùng mở ứng dụng
Chuỗi sự kiện chính:  1. Người dùng mở ứng dụng  2. Hiển thị lịch âm lịch	

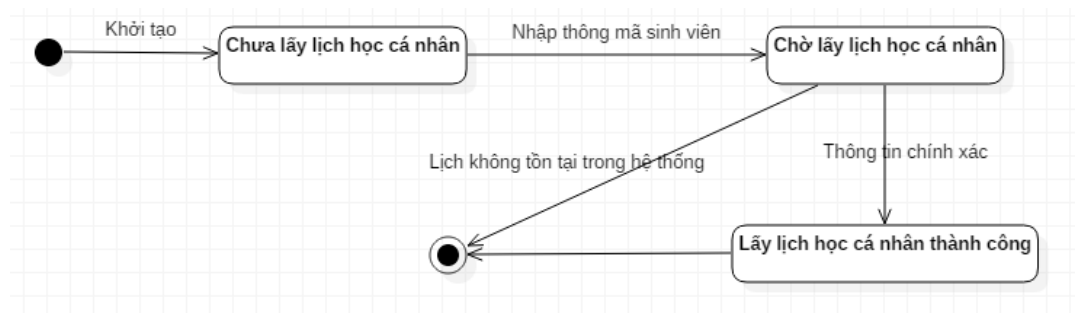


### 2.3.2.11. Biểu diễn kịch bản cho UC chức năng xem lịch dương lịch

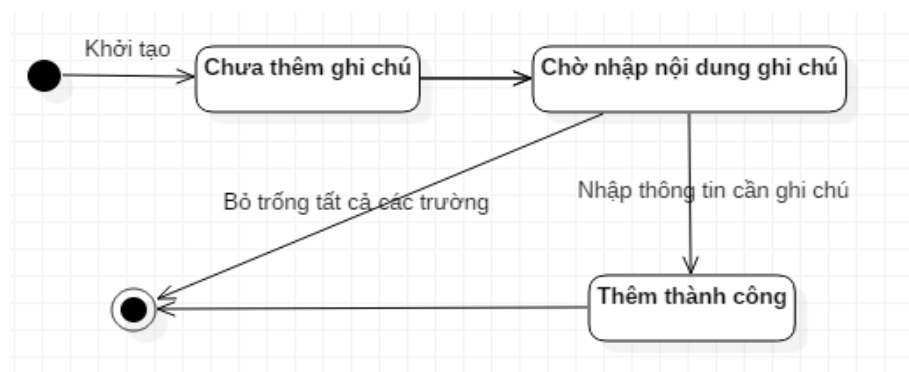
Tên Use Case	Xem lịch dương lịch
Tác nhân chính	Người dùng
Tiền điều kiện	
Thành công	Hiển thị lịch dương lịch
Kích hoạt	Người dùng mở ứng dụng
Chuỗi sự kiện chính:  3. Người dùng mở ứng dụng  4. Hiển thị lịch âm dương	

### 2.3.3. Biểu đồ trạng thái

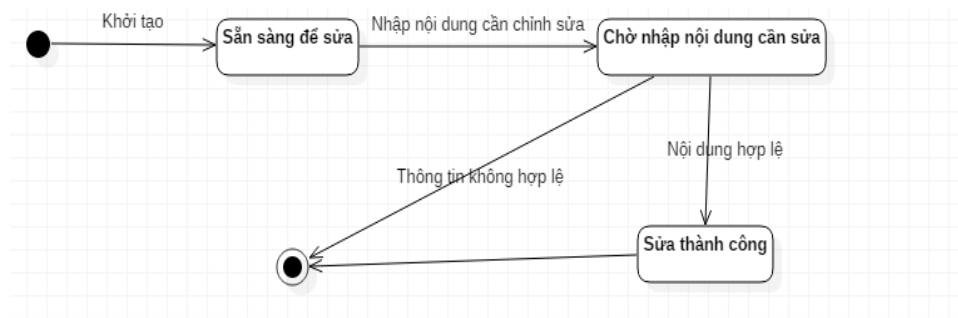
#### 2.3.3.1. Lấy lịch học cá nhân



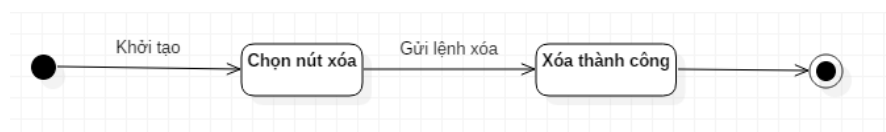
#### 2.3.3.2. Thêm ghi chú



### 2.3.3.3. Sửa ghi chú

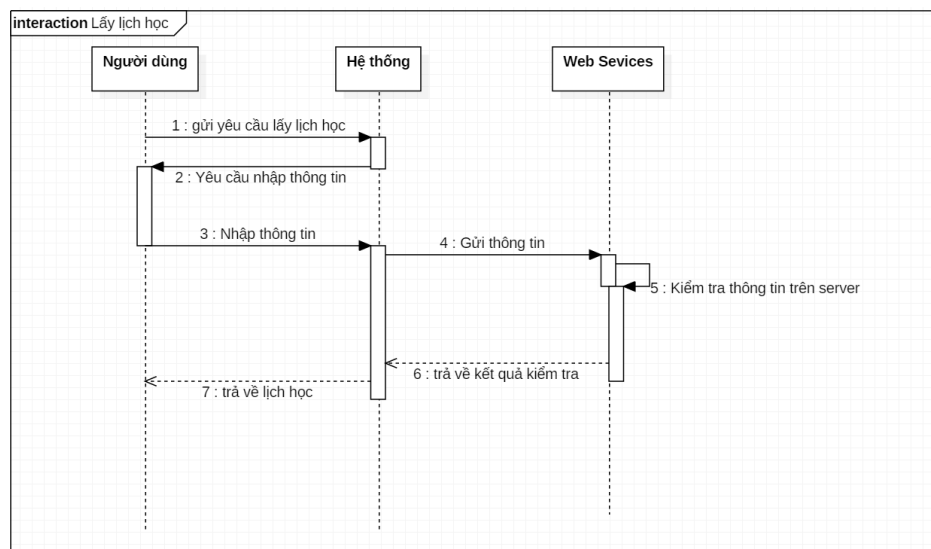


### 2.3.3.4. Chức năng xóa

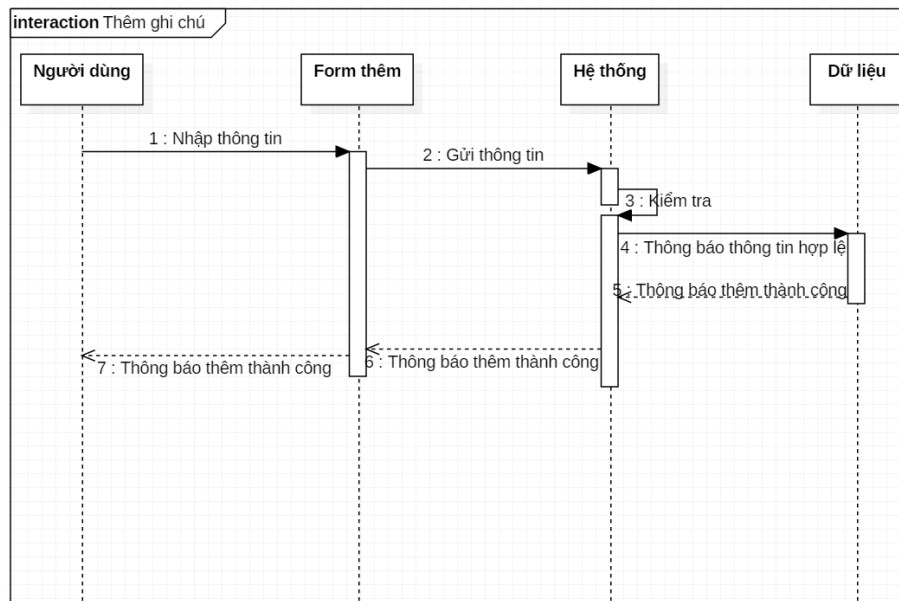


## 2.3.4. Biểu đồ trình tự

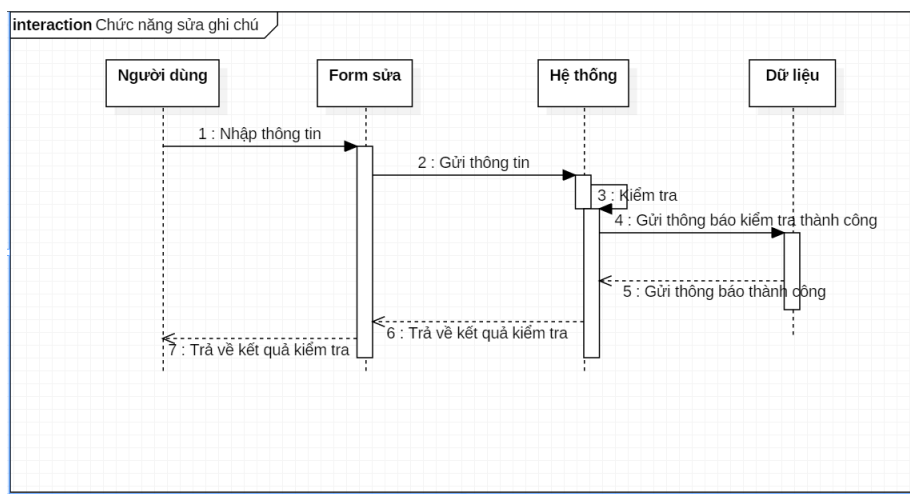
### 2.3.4.1. Chức năng thêm lịch học



### 2.3.4.2. Chức năng thêm ghi chú

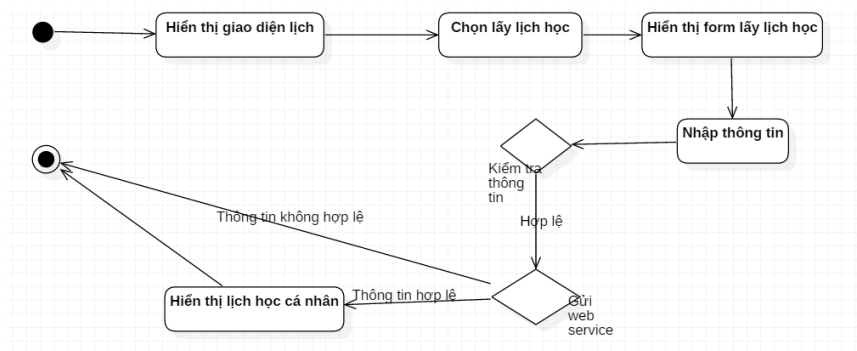


### 2.3.4.3. Chức năng sửa ghi chú

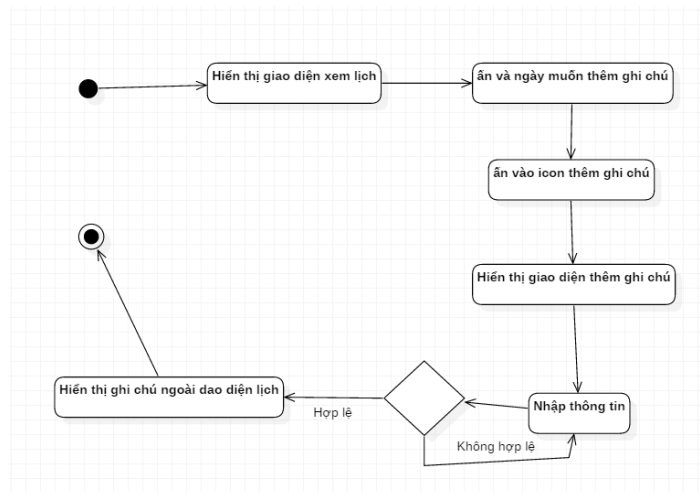


## 2.3.5. Biểu đồ hoạt động

### 2.3.5.1. Thêm lịch học



### 2.3.5.2. Thêm ghi chú



## CHƯƠNG 3: CÀI ĐẶT ỨNG DỤNG VÀ CHẠY THỬ NGHIỆM

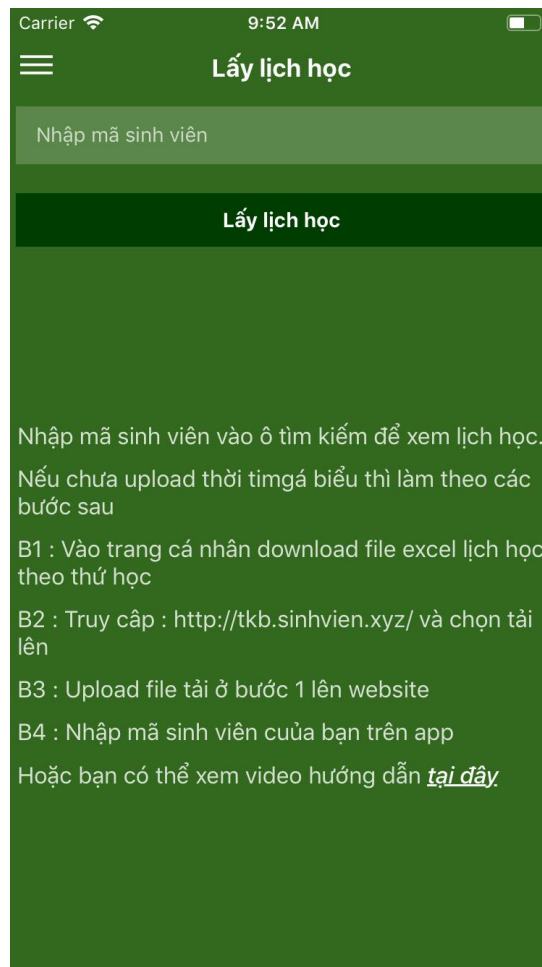
### 3.1. Cài đặt ứng dụng

Yêu cầu phần mềm:

- Hệ điều hành: IOS 9 trở lên

### 3.2. Chạy thử ứng dụng

#### 3.2.1. Giao diện lấy cập nhập lịch học



*H3.1: Hình ảnh giao diện cập nhập lịch học*

Đây là giao diện lấy lịch học cá nhân người dùng sẽ sử dụng mã sinh viên để lấy lịch học (khi đã upload file lịch học lên website). Người dùng chưa rõ các bước có thể ấn vào “tại đây” trên màn hình lấy lịch học để xem video hướng dẫn

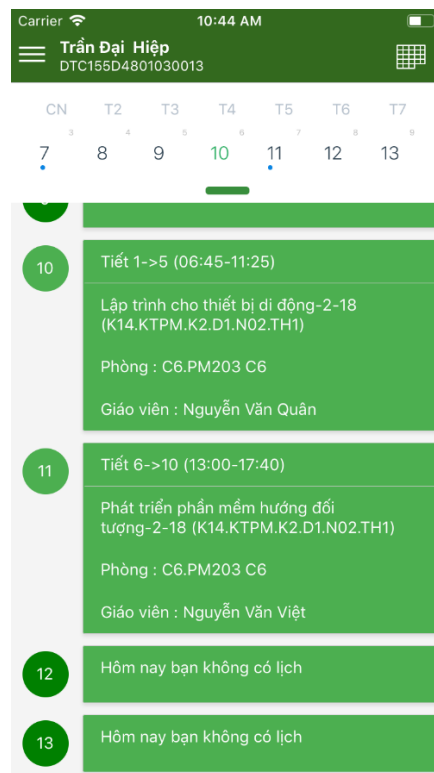
### 3.2.2. Giao diện xem lịch âm lịch và dương lịch



#### H3.2: Hình ảnh giao diện cập nhập lịch âm lịch và dương lịch

Đây là giao diện xem lịch âm lịch hoặc dương lịch, cũng có thể xem lịch học cá nhân và ghi chú nhanh tại đây, người dùng có thể ấn vào icon đổi kiểu xem lịch ở góc bên phải trên

### 3.2.3. Giao diện xem lịch theo lịch biểu



*H3.3: Hình ảnh giao diện xem lịch học theo lịch biểu*

Đây là giao diện xem lịch học theo tuần, người dùng có thể xem lịch học cá nhân hoặc xem ghi chú tại

### 3.2.4. Giao diện xem thời gian ra vào lớp



Carrier 10:45 AM

Thời gian ra vào lớp

Lịch học mùa đông từ : 15/10  
Lịch học mùa hè từ : 15/4

Tiết	Mùa hè	Mùa đông
1	6:30 - 7:20	6:45 - 7:35
2	7:25 - 8:15	7:40 - 8:30
3	8:25 - 9:15	8:40 - 9:30
4	9:25 - 10:15	9:40 - 10:30
5	10:20 - 11:10	10:35 - 11:25
6	13:00 - 13:50	13:00 - 13:50
7	13:55 - 14:45	13:55 - 14:45
8	14:55 - 15:45	14:55 - 15:45
9	15:55 - 16:45	15:55 - 16:45
10	16:50 - 17:40	16:50 - 17:40
11	18:15 - 19:5	18:15 - 19:5
12	19:10 - 19:60	19:10 - 19:60
13	20:30 - 21:20	20:30 - 21:20
14	21:30 - 22:20	21:30 - 22:20
15	22:30 - 23:20	22:30 - 23:20

*H3.4: Hình ảnh giao diện thời gian ra vào lớp*

Đây là giao diện xem thời gian ra vào lớp, tại đây người dùng có thể xem lịch mùa đông và mùa hè, theo trường học tương ứng của bạn



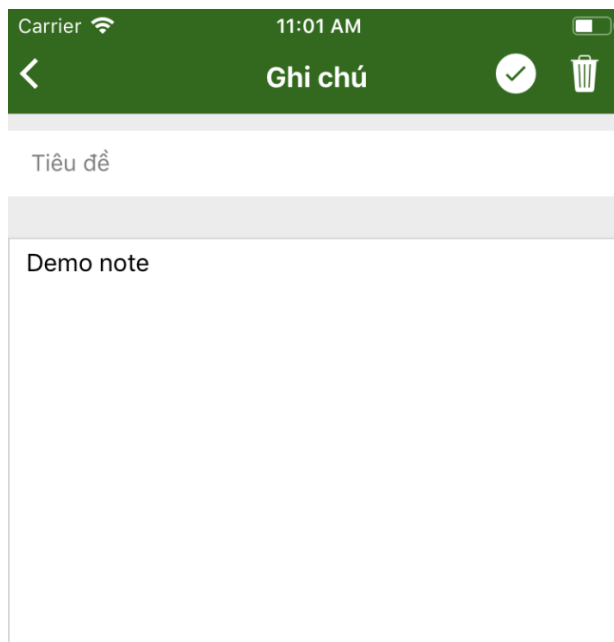
### 3.2.5. Giao diện chi tiết lịch học



*H3.5: Hình ảnh giao diện xem chi tiết lịch theo ngày*

Đây là giao diện xem chi tiết lịch học khi xem lịch học dưới dạng theo tháng, ở đây người dùng có thể thêm ghi chú vào ngày hiện tại, có thể ấn vào ghi chú để sửa ghi chú hoặc xóa ghi chú

### 3.2.6. Giao diện sửa ghi chú



*H3.6: Hình ảnh giao diện sửa ghi chú*

Ở giao diện này người dùng có thể sửa ghi chú và ấn nút lưu lại để thực hiện sửa ghi chú. Người dùng cũng có thể xóa ghi chú tại bằng cái ấn vào biểu tượng thùng rác ở góc trên bên phải

## KẾT LUẬN

### Thành công

- Có thêm hiểu biết về react native, giao tiếp với web services
- Qua quá trình làm thực tập chuyên ngành, củng cố được kiến thức về quy trình làm phần mềm
- Củng cố kiến thức về: React native, javascripts, PHP
- Rút ra được nhiều kinh nghiệm từ những lần làm sai.
- Những điều trên chính là nền tảng kiến thức giúp em về sau thực hiện những bài tập lớn tốt hơn.

### 3.3. Khó khăn

- Do còn thiếu kinh nghiệm và một số kiến thức còn chưa được tốt vì thế mà trong quá trình thực hiện còn gặp nhiều khó khăn

## This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

## TÀI LIỆU THAM KHẢO

- [1] Cộng đồng StackOverFlow (<https://stackoverflow.com>)
- [2] Trang web WIKIPEDIA (<https://vi.wikipedia.org>)
- [3] Trang web W3SCHOOL (<https://www.w3schools.com>)
- [4] Trang web reactnavigation (<https://reactnavigation.org>)
- [5] Trang web react-native (<https://facebook.github.io/react-native/>)
- [6] Trang web techtalk (<https://techtalk.vn/>)
- [7] Trang web fullstackstation (<https://fullstackstation.com/>)
- [8] Trang web redux (<https://redux.js.org/basics/usage-with-react>)
- [9] Thư viện react-native-calendars (<https://github.com/wix/react-native-calendars>)
- [10] Trang web viblo (<https://viblo.asia/>)