

XÂY DỰNG CỔNG THÔNG TIN ĐIỆN TỬ KHOA CÔNG NGHỆ THÔNG TIN TRÊN THIẾT BỊ ĐA NỀN TẢNG

BUILDING WEB PORTAL OF UNETI'S INFORMATION TECHNOLOGY FACULTY ON MULTI-PLATFORM DEVICES

Lương Thị Thảo Hiếu, Nguyễn Duy Quý

Khoa Công nghệ thông tin, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp

Đến tòa soạn ngày 20/06/2020, chấp nhận đăng ngày 23/07/2020

Tóm tắt: Hiện nay, Khoa Công nghệ thông tin (CNTT) Trường Đại học Kinh tế - Kỹ thuật Công nghiệp đang sử dụng hệ thống website xây dựng trên nền tảng mã nguồn mở laravel PHP. Ý tưởng của chúng tôi muốn xây dựng một cổng thông tin điện tử chạy song song hệ thống website của khoa. Người dùng có thể duyệt web trên các thiết bị sử dụng hệ điều hành khác nhau như Android, IOS, Windows phone. Trong bài báo này chúng tôi nghiên cứu kiến trúc mobile web service, sau đó tích hợp RESFul web service, sử dụng lập trình react native, xây dựng hệ thống web Khoa CNTT thiết bị đa nền tảng với đầy đủ chức năng: giới thiệu khoa, đào tạo, tra cứu thông tin sinh viên, tra cứu lịch thi, lịch dạy, lương giảng viên. Các thông tin này được cập nhật liên tục từ các hệ thống web của trường và khoa.

Từ khóa: REACT NATIVE, mobile web service, RESTful, JSON.

Abstract: Currently, the Faculty of Information Technology of UNETI is using a website system built on the open source laravel PHP platform. Our idea is building an Web portal runs parallel with the department's website system. Users can browse the web on devices using different Operating Systems such as Android, IOS, Windows Phone. In this paper, we study mobile web service architecture, then integrate RESFul web service, use React Native programming, building web system running on multi-platform device with many functions: introduce the department, training, look up student information, look up exam schedules, teaching schedule, lecturer salary. This information is constantly updated from web system of faculty and UNETI.

Keywords: REACT NATIVE, mobile web service, RESTful, JSON.

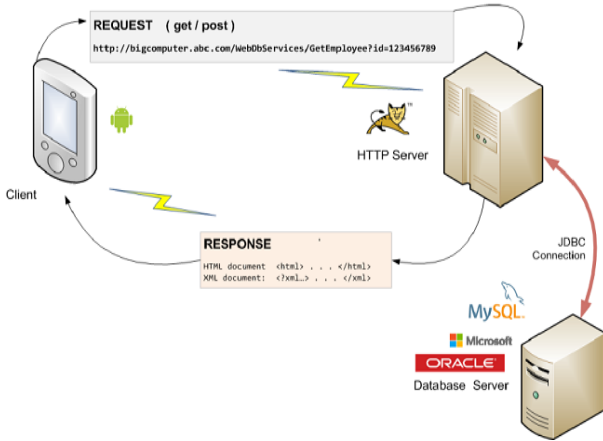
1. GIỚI THIỆU

Các thiết bị di động (điện thoại thông minh và thiết bị cầm tay) được xem là các thiết bị hạn chế tài nguyên và không phù hợp để sử dụng dịch vụ internet. Các nhà thiết kế di động luôn tập trung mở rộng nâng cấp phần cứng: tăng dung lượng bộ nhớ, tăng cường khả năng tính toán, đồng thời cải thiện hiệu suất các ứng dụng di động. Sự tách biệt giữa giao diện người dùng và dịch vụ logic được cung cấp

bởi Web Service là cơ hội mới mang lại dịch vụ internet cho các thiết bị di động. Các ứng dụng chạy trên thiết bị di động, thông qua Web Service, có thể thích nghi với khả năng hạn chế của từng thiết bị.

Để tích hợp các công nghệ Web Service vào các thiết bị di động tạo thành kiến trúc Mobile Web Service tối ưu, đáp ứng nhu cầu người sử dụng, cần quan tâm đến tính chặt chẽ của thiết bị này và hệ thống truyền thông di động. Về

cơ bản, Web Service được chia thành hai loại chính: RESTful và SOAP based Web Service [1]. Sự phân chia này dựa trên kiểu kiến trúc sử dụng để thực hiện công nghệ.



Hình 1. Web service cho ứng dụng đa lớp

SOAP viết tắt của Simple Object Access Protocol - giao thức truy cập đối tượng đơn giản - đây là công nghệ hướng đối tượng, định nghĩa các giao thức dùng để trao đổi thông điệp dựa trên XML. Năm 2000 Fieding cho ra đời nguyên lý REST. REST-Representational State Transfer - công nghệ hướng tài nguyên, chứa một tập các chuẩn thiết kế, định nghĩa cách thức đặc biệt sử dụng các chuẩn web như HTTP và URI. REST đã trở thành công nghệ thực thi phổ biến để phát triển Web Service. REST triển khai độc lập, đơn giản, gọn nhẹ, khả năng tương tác cao và gần như thay thế SOAP. Trong bài báo này chúng tôi giới thiệu một số khái niệm Web Service, so sánh XML với JSON, so sánh hai công nghệ thiết kế Web Service: SOAP based và RESTful. Nghiên cứu, lựa chọn kiến trúc mobile web service, đề xuất mô hình hệ thống, mô hình ứng dụng áp dụng xây dựng hệ thống web quảng cáo Khoa CNTT trên nền tảng Android, phục vụ nhu cầu tra cứu thông tin sinh viên, thông tin tuyển sinh, quảng cáo Khoa CNTT, lịch cá nhân giảng viên,...

2. MỘT SỐ KHÁI NIỆM

Khái niệm 2.1

Web service (dịch vụ web) là phương thức trao đổi giữa hai hay nhiều thiết bị điện tử trên môi trường Word Wide Web. Dịch vụ web “là một hệ thống phần mềm được thiết kế để hỗ trợ giao tiếp máy với máy trên mạng”, nó cung cấp thông tin thô, khó hiểu với người dùng, chính vì vậy được sử dụng bởi tầng ứng dụng. Các ứng dụng sẽ chế biến dữ liệu thô trước khi trả về cho người dùng cuối. Các hệ thống khác có thể giao tiếp với Web Service bằng cách sử dụng tín hiệu SOAP, kết hợp với HTTP và XML hoặc JSON. Web Service là nền tảng trung gian và thường dưới dạng text, có thể phát triển, chạy và truy cập dựa trên công nghệ phức tạp.

Phân loại theo cách sử dụng: Web service sử dụng máy chủ cố định và được tiêu thụ bởi máy trạm gọi là Station Web Service, trong khi đó các dịch vụ được tiêu thụ bởi thiết bị mobile gọi là Mobile Web Service.

Mobile Web Service được triển khai trên thiết bị di động qua môi trường mạng không dây đã giải quyết được những thách thức lớn về hạn chế tài nguyên và đặc tính truy cập bằng thông không dây. Những thách thức này tổng kết lại như sau:

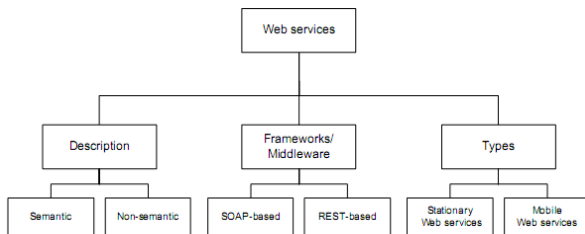
Hạn chế tài nguyên: Vấn đề ràng buộc tài nguyên trên mobile là hạn chế màn hình hiển thị.

Kết nối gián đoạn: Các thiết bị di động thường xuyên thay đổi thao tác thực hiện trên mạng và kết quả là bất kỳ dịch vụ nào cũng có thể trở nên tạm thời không tiếp cận được. Điều này đặt ra một thách thức lớn cho việc cung cấp các dịch vụ web tin cậy trong môi trường mạng không dây thay đổi liên tục.

Tính địa chỉ: Thiết bị di động luôn thay đổi điểm gắn kết vào mạng khi chúng định vị lại.

Khả năng mở rộng: Mobile Web Service không thực hiện tốt trong trường hợp có sự tương tranh truy cập dịch vụ web.

Tài nguyên không đồng nhất: Việc cung cấp một dịch vụ web mobile thực hiện độc lập và xuyên suốt là một vấn đề phức tạp.



Hình 2. Tổng quan Web service

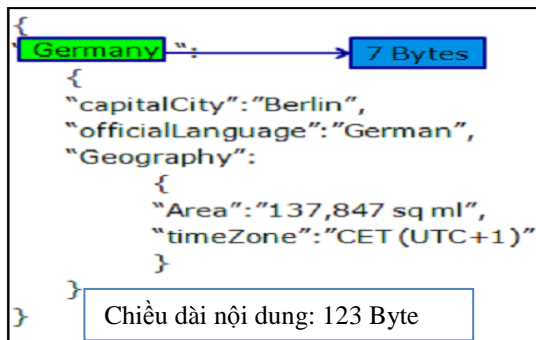
Khái niệm 2.2

JSON (Java Script Object Notation) [1]

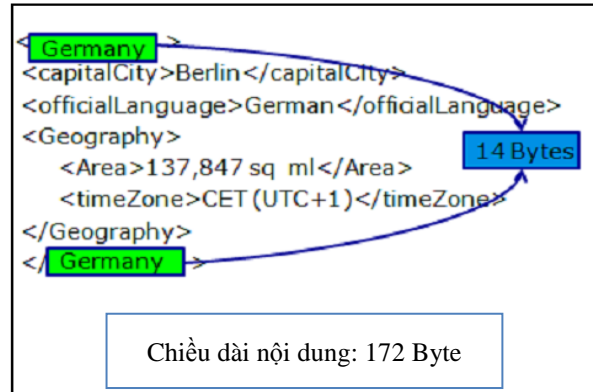
Trong những năm gần đây, một định dạng trao đổi dữ liệu mới được đưa vào sử dụng phổ biến, đó là JSON (Java Script Object Notation), đây là một chuẩn mở để biểu diễn dữ liệu trên Web, sử dụng text-based để biểu diễn dữ liệu và sử dụng các kí tự như cặp ngoặc "[{}]", dấu ":" và dấu ",". Dữ liệu được biểu diễn sử dụng các cặp key-value [1].

JSON có đặc tính nhẹ, sử dụng text để mô tả, trao đổi dữ liệu độc lập ngôn ngữ. JSON định nghĩa một tập hợp các luật có cấu trúc cho việc biểu diễn dữ liệu trên di động.

Cấu trúc của JSON: là một định dạng thông điệp trộn lẫn, chứa nhiều cấu trúc dữ liệu khác nhau như: integer, objects, boolean.



Hình 3. Định dạng JSON



Hình 4. Định dạng XML

Qua hình 3, hình 4, cho thấy định dạng dữ liệu JSON có chiều dài nội dung nhỏ hơn 28.5% so với XML. Lý do của sự sai khác này do lặp lại nội dung trong mô tả XML, ví dụ cần thêm 7 byte để mô tả từ “Germany”. Điều này gây tiêu tốn bộ nhớ, tăng thời gian xử lý, và không phù hợp với thiết bị di động hạn chế tài nguyên.

Khái niệm 2.3

Web service tuân thủ kiến trúc SOAP và giao thức SOAP gọi là SOAP Based Web service. Web service xây dựng trên nguyên lý REST gọi là Restful Web Service. So sánh giữa hai công nghệ này như sau:

SOAP based: Được thiết kế để thực hiện các tương tác với các hệ thống từ xa. Trong thiết kế này các nhà cung cấp dịch vụ và khách hàng cần thiết lập một sự hiểu biết chung của cú pháp dịch vụ và các thao tác thực hiện. Mỗi web service soap based có giao diện riêng và được mô tả bởi Web Services Description Language (WSDL). Sử dụng XML để truyền thông điệp. Điểm mạnh của giao thức truyền thông điệp này là có khả năng làm việc trong môi trường không đồng nhất và độc lập thiết bị. Tuy nhiên tồn tại một số hạn chế:

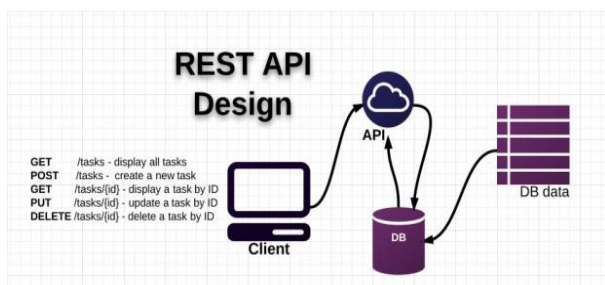
a) Phức tạp: Để triển khai dịch vụ SOAPbased cần nhiều kinh nghiệm do độ phức tạp giao

thức. Thêm vào đó các yêu cầu được viết bằng ngôn ngữ tự nhiên gây mất thời gian, không phù hợp với sự hạn chế tài nguyên của thiết bị di động.

b) Khả năng truy cập và giao diện hạn chế.

c) Khả năng tương tác kém: Mỗi web service có giao diện riêng.

Rest based [2]: RESTful Web Service (RESTful web API) là web API sử dụng HTTP xây dựng trên kiến trúc REST. Restful web service thu hút được sự quan tâm từ cộng đồng web do đơn giản, linh hoạt và khả năng tương tác cao. Các dịch vụ thiết kế theo hướng này khai thác tài nguyên web qua địa chỉ URI duy nhất. Người dùng truy cập tài nguyên thông qua URI. Hướng tiếp cận này phù hợp với các thiết bị di động, nó tách biệt giao diện người dùng với lưu trữ dữ liệu đồng thời cải tiến sự linh hoạt của giao diện trên các nền tảng khác nhau và đơn giản hóa các thành phần server bằng cách chuyển chúng thành phi trạng thái. RESTful Web Service nhẹ, dễ mở rộng và bảo trì, đưa ra tài nguyên thông qua URI và sử dụng 4 phương thức của HTTP để tạo, khôi phục, cập nhật, xóa tài nguyên. [4]



Hình 5. Cơ chế REST API

Một số lợi ích của công nghệ này như sau:

Tính tương tác: RESTful kế thừa tính tương tác của HTTP.

Tính định địa chỉ: Mỗi nguồn tài nguyên có địa chỉ URI duy nhất và người dùng có thể lấy

thông qua kết nối giữa các tài nguyên.

Phi trạng thái: Những yêu cầu trong tiếp cận RESTful là độc lập.

Giao diện duy nhất: Hai máy khách HTTP bất kỳ có thể giao tiếp trực tiếp với nhau mà không cần cấu hình đặc biệt nào. Trái lại SOAP cần cả hai bên server và client phải đồng ý và thống nhất chung một số phương thức, kiểu dữ liệu và mô hình định địa chỉ.

Bảng 1. So sánh giữa SOAP-based và REST-based

Đặc tính	SOAP based	REST based
Kiến trúc	Hướng dịch vụ	Hướng tài nguyên
Giao thức	Bất kỳ	HTTP
Mô tả	WSDL	Không theo chuẩn
Mô hình dữ liệu	Ẩn	Hiện
Biểu diễn dữ liệu	XML	XML, JSON
An toàn	Dựa trên dịch vụ web	Dựa trên HTTP

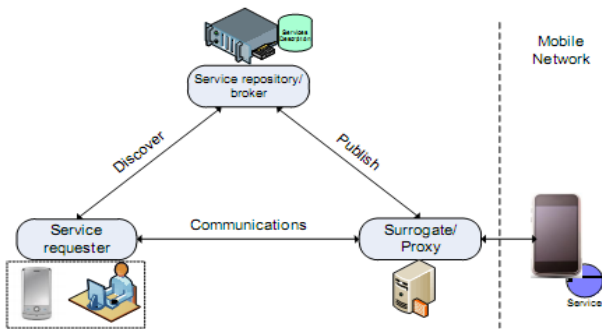
3. KIẾN TRÚC MOBILE WEB SERVICE [2][3]

3.1. Kiến trúc Proxy-based

Đây là kiến trúc dễ tiếp cận nhất, tránh được nhiều thách thức về việc đối mặt với sự thực thi web service trong điều kiện ràng buộc tài nguyên. Proxy thường là một máy high-end được gắn cố định với một mạng. Từ đó về lý thuyết nó có băng thông không giới hạn để giảm thiểu việc sử dụng băng thông trong thiết bị di động, đủ năng lực xử lý để giảm tải các thiết bị hạn chế tài nguyên và thực hiện quá trình sử dụng tài nguyên chuyên sâu.

Đồng thời phục vụ di động trong trường hợp ngắt kết nối và đáp ứng nhu cầu truy cập cao trong khi vẫn duy trì độ trễ hợp lý. Trong kiến trúc này chứa thiết bị mobile hosting Web

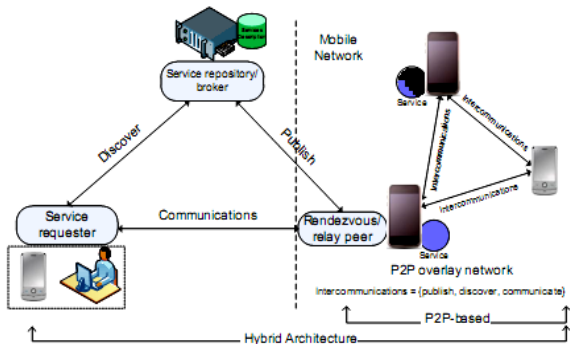
service kết nối với máy high-end đóng vai trò như một proxy. Proxy biểu diễn điểm cuối của web service tới máy khách. Kiến trúc này giải quyết nhiều vấn đề thách thức liên quan đến mobile web service, ví dụ: Proxy có khả năng lưu trữ lớn, phục vụ một lượng lớn các máy client liên tục. Nó cũng giải quyết tính không đồng nhất của các thiết bị di động khác nhau.



Hình 6. Kiến trúc proxy-based

3.2. Kiến trúc P2P

Đây là kiến trúc phân tán. Kiến trúc này dựa trên cơ chế quảng cáo mạng P2P-Peer to peer để xuất bản và khai thác web service. Cơ chế này quản lý tính di động của các node, quản lý vị trí và ràng buộc thông tin của web service.

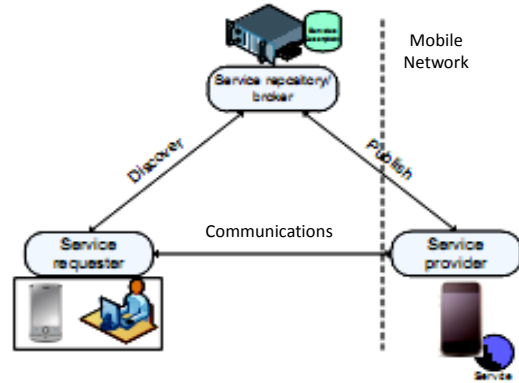


Hình 7. Kiến trúc P2P

3.3. Kiến trúc bất đối xứng

Kiến trúc này được thiết kế đặc biệt với mục đích tương tác các dịch vụ mobile có chu kỳ sống dài, cho phép các dịch vụ chạy không đồng bộ và độc lập. Trong kiến trúc này client

yêu cầu dịch vụ và chờ phản hồi mà không khóa thiết bị trong suốt thời gian thực hiện. Phản hồi được gửi đến bất kỳ khi nào nó sẵn sàng.



Hình 8. Kiến trúc bất đối xứng

4. XÂY DỰNG CÔNG THÔNG TIN ĐIỆN TỬ KHOA CÔNG NGHỆ THÔNG TIN TRÊN THIẾT BỊ NỀN TẢNG

4.1. Xây dựng hệ thống

Ý tưởng: Xây dựng hệ thống web trên thiết bị đa nền tảng chạy song song với web của khoa. Người dùng sử dụng điện thoại Android, Iphone để duyệt web và tra cứu thông tin.

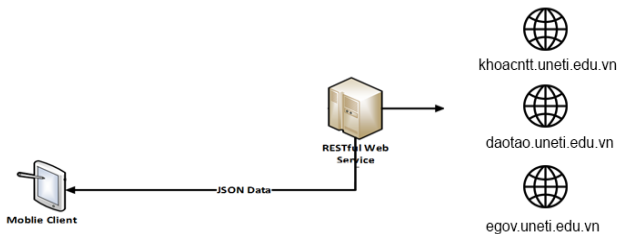
Trong 3 kiến trúc mobile web service trình bày trên, chúng tôi lựa chọn kiến trúc proxy-based, đưa vào xây dựng hệ thống mobile web phù hợp với hiện trạng cơ sở vật chất tại Khoa CNTT. Đề xuất giải pháp tích hợp module Restful web service trên kiến trúc proxy-based, nhận tham số từ thiết bị di động, thực hiện yêu cầu dịch vụ và trả về kết quả cho thiết bị di động dưới dạng dữ liệu JSON, loại bỏ quá trình xử lý XML nặng trên thiết bị di động, hệ thống luôn đảm bảo yêu cầu truy cập thông tin liên tục.

Yêu cầu phần cứng, phần mềm: Về phần cứng chúng tôi sử dụng máy chủ cài hệ điều hành Ubuntu Server 16.04. Về phần mềm: Sử dụng lập trình java tạo module Restful webservice, module này tích hợp trên máy chủ của khoa. Ý nghĩa của module này dùng để lấy thông tin từ web nhà trường, chuyển về

dạng JSON. Sau đó để đưa dữ liệu JSON hiển thị lên mobile, chúng tôi dùng công nghệ lập trình react native ra đời năm 2015 [5]. Về bản chất dùng React lập trình tạo fle định dạng (.apk cho android hoặc .ipa cho ios). Người dùng cài đặt các file này lên thiết bị đa nền tảng, khi chạy ứng dụng sẽ thực hiện kết nối không dây, truy cập đến máy chủ đặt tại khoa CNTT, sử dụng máy chủ với cấu hình (bộ xử lý Core i7-9700K, CPU@ 3.6 Hz, RAM 32 GB, 2Card GPU, 16 GB), đáp ứng được một lượng lớn người dùng truy cập web, dữ liệu được cập nhật liên tục.

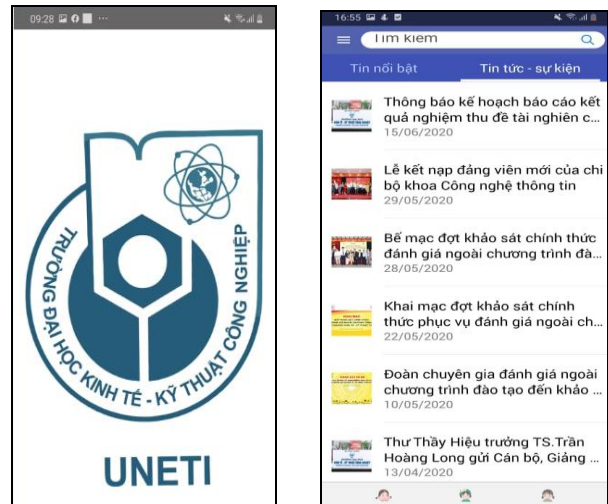
4.2.Thực nghiệm và kết quả

Chúng tôi xây dựng hoàn chỉnh hệ thống web Khoa CNTT trên thiết bị đa nền tảng với một số chức năng chính sau: Mục giới thiệu khoa bao gồm cơ cấu tổ chức, đội ngũ giảng viên. Mục thông tin đào tạo khoa, bao gồm: Giới thiệu chuẩn đầu ra, giới thiệu chương trình đào tạo cao đẳng và đại học ngành CNTT. Mục tra cứu thông tin sinh viên, chúng tôi đã thử nghiệm, có thể tra cứu thông tin sinh viên toàn trường từ khóa 1 đến khóa 13 như: xem điểm các môn học của năm học, xem lịch học, lịch thi, công nợ... Mục tra cứu thông tin giảng viên, tra cứu được thông tin các giảng viên đang công tác tại Trường Đại học Kinh tế - Kỹ thuật Công nghiệp. Giảng viên đăng nhập bằng tài khoản egov cá nhân sẽ xem được lịch giảng dạy, lịch coi thi, tiền lương.

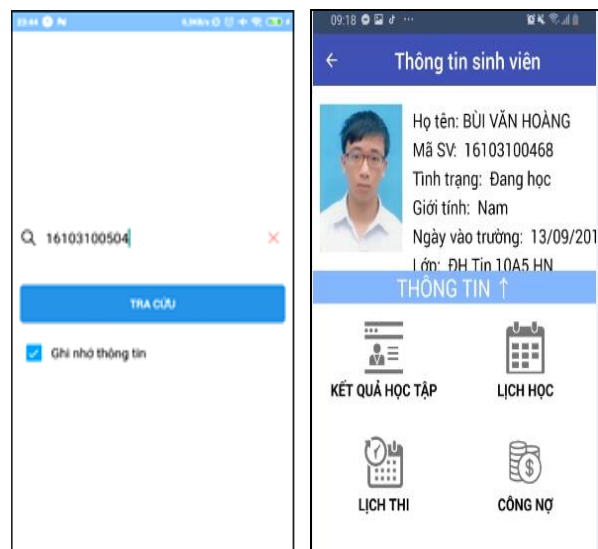


Hình 9. Mô hình hệ thống mobile web service IT UNETI

Một số hình ảnh giao diện:

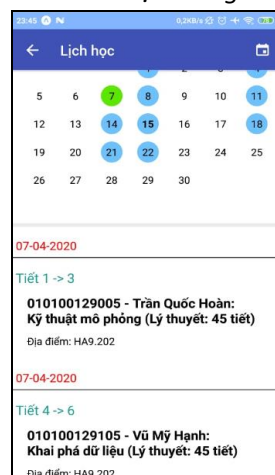


Hình 10. Giao diện hiển thị tin tức

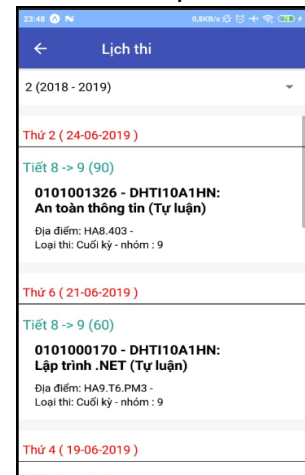


a. Giao diện trang chủ

b. Giao diện tin tức



c. Lịch học



d. Lịch thi

Hình 11. Giao diện đăng nhập sinh viên

4. KẾT LUẬN

Qua nghiên cứu về kiến trúc mobile web service, các vấn đề xử lý dữ liệu trong điều kiện ràng buộc tài nguyên, chúng tôi đề xuất mô hình hệ thống, tích hợp module Restful Web Service phía server, kết hợp xử lý dữ liệu

JSON phía mobile bằng react native, xây dựng hoàn chỉnh ứng dụng web chạy trên thiết bị đa nền tảng. Ứng dụng này ngoài chức năng truyền tải tin tức, còn tổng hợp nhiều tính năng dành cho sinh viên, giảng viên.

TÀI LIỆU THAM KHẢO

- [1] Ei Ei Thu, Than Nwe Aung *Developing Mobile Application Framework By Using RESTful Web Service with JSON Parser*, University of Computer Studies Mandalay (UCSM), Mandalay, Myanmar, August, 2015.
- [2] Khalid Elgazzar, Patrick Martin, Hossam: *Mobile Web Services: State of the Art and Challenges*, Queen's University, Canada, 2014.
- [3] KamalEldin Mohamed^a, a* *Performance Analysis of Web Services on Mobile Devices*, October, 2012.
- [4] Alin COBĂRZAN, *Consuming Web Services on Mobile Platforms*, Babeş-Bolyai University, March, 2010.
- [5] Devin Abbot *Full stack React native*, Sanfranc Cisco, (2017).

Thông tin liên hệ: **Lương Thị Thảo Hiếu**

Điện thoại: 0942160880 - Email:ltthieu@uneti.edu.vn

Khoa Công nghệ thông tin, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp.