

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....~O~.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Cần Thơ, ngày tháng 04 năm 2015

Giảng viên hướng dẫn

ThS. Lâm Chí Nguyễn

NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN

.....~O~.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Cần Thơ, ngày tháng 04 năm 2015

Giảng viên phản biện

NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN

.....~O~.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Cần Thơ, ngày tháng năm 2015

Giảng viên phản biện

LỜI CẢM ƠN

.....~O~.....

Trong suốt quá trình thực hiện đề tài luận văn tốt nghiệp “Xây dựng hệ thống quản lý, điều khiển các dịch vụ mạng bằng giao diện web” em luôn nhận được sự giúp đỡ từ giảng viên, bạn bè và các anh chị để em có thể thực hiện tốt đề tài của mình.

Trước tiên, em xin gửi lời cảm ơn chân thành đến thầy Lâm Chí Nguyễn đã tận tình hướng dẫn và chỉ ra con đường tốt để em có thể được làm luận văn đồng thời học hỏi và rèn luyện thêm nhiều kinh nghiệm.

Trên hết em xin gửi lời cảm ơn sâu sắc tới quý thầy cô trường Đại học Cần Thơ nói chung và quý thầy cô khoa Công nghệ thông tin và Truyền thông nói riêng, quý thầy cô đã truyền cho em những kiến thức, kinh nghiệm quý báu trong suốt 4 năm qua để em có đủ hành trang và nghị lực bước vào đời và có thể trở thành người có ích cho xã hội.

Kế đến là xin cảm ơn bạn cùng làm chung đề tài với Module cung cấp thư viện điều khiển đã hỗ trợ và cùng nghiên cứu để cả hai có thể hoàn thành đề tài.

Mặc dù đã cố gắng hoàn thành luận văn trong phạm vi và khả năng nhưng chắc chắn sẽ không tránh khỏi những thiếu sót và cũng như chưa thể hoàn thiện yêu cầu đề tài đặt ra trong khoảng thời gian ngắn. Em kính mong nhận được sự thông cảm và tận tình chỉ bảo của quý thầy cô và các bạn, hi vọng đề tài của em sẽ được phát triển và hoàn thiện để đưa vào sử dụng.

Cần Thơ, ngày tháng năm 2015

Sinh viên thực hiện

Thái Thanh Nhân

MỤC LỤC

.....❧.....	
LỜI CẢM ƠN	4
MỤC LỤC.....	5
KÍ HIỆU VÀ VIẾT TẮT	7
DANH MỤC HÌNH VẼ	8
ABSTRACT	11
TÓM TẮT	12
PHẦN 1: GIỚI THIỆU.....	13
1.1 Đặt vấn đề.....	13
1.2 Lịch sử giải quyết.....	13
1.3 Mục tiêu đề tài.....	14
1.4 Đối tượng và phạm vi nghiên cứu.....	14
1.5 Nội dung nghiên cứu.....	15
1.6 Những đóng góp chính.....	15
1.7 Bố cục quyền luận văn.....	16
PHẦN 2: NỘI DUNG.....	17
CHƯƠNG 1: MÔ TẢ BÀI TOÁN	17
I. Yêu cầu đề tài	17
II. Mô hình kiến trúc hệ thống ứng dụng	17
III. Phạm vi chức năng.....	18
IV. Phạm vi triển khai ứng dụng.....	19
CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP.....	20
I. THIẾT KẾ GIẢI PHÁP.....	20
I.1.1. Hướng thiết kế hệ thống	20
I.1.2. Mô hình Web MVC	21
I.1.3. Thiết kế cơ sở dữ liệu	22
I.1.4. Các chức năng trong hệ thống quản lý server.....	24
II. CÀI ĐẶT GIẢI PHÁP	36
II.1. Cài đặt cấu trúc template website với Tiles.....	36

II.1.1. Cấu hình tiles dependency.....	36
II.1.2. Cấu hình cấu trúc giao diện trong tiles*.xml.....	36
II.2. Cấu hình Mongo kết nối với các Model.....	38
II.2.1. Cấu hình mongodb dependency	38
II.2.2. Định nghĩa các Model tương ứng với các Documents	38
II.2.3. Khai báo các beans tương tác với mongodb.....	39
II.2. Cấu hình RequestMapping trong Controller để tương tác với Views.....	39
II.2.1. Cấu hình Controller và các RequestMapping.....	40
II.2.2. Thiết lập token để xác minh phiên làm việc của người dùng, nâng cao bảo mật.	40
CHƯƠNG 3: KIỂM THỬ VÀ ĐÁNH GIÁ.....	42
I. MỤC TIÊU KIỂM THỬ.....	42
II. KỊCH BẢN KIỂM THỬ VÀ ĐÁNH GIÁ.....	42
II.1. Quản lý hệ thống chính.....	42
II.2. Chức năng quản lý và cấu hình dịch vụ trên server	44
II.3. Chức năng quản lý hệ thống dành cho Admin	54
PHẦN 3: KẾT LUẬN	58
KẾT QUẢ ĐẠT ĐƯỢC VÀ HƯỚNG PHÁT TRIỂN.....	58
HẠNH CHẾ	59
TÀI LIỆU THAM KHẢO	60
PHỤ LỤC.....	61
PHỤ LỤC: CÀI ĐẶT VÀ CẤU HÌNH SPRING TOOL SUITE	61
1. Cài đặt	61
PHỤ LỤC: CÀI ĐẶT VÀ CẤU HÌNH MONGO DATABASE.....	62
1. Cài đặt	62
2. Truy vấn	63

KÍ HIỆU VÀ VIẾT TẮT

.....~O~.....

Từ/kí hiệu viết tắt	Nguyên bản
JSON	JavaScript Object Noation
SSH	Secure Shell
POM	Project Object Model
Server	Máy chủ
FTP	File Tranfer Protocol

Bảng – Kí hiệu viết tắt

DANH MỤC HÌNH VẼ

.....~O~.....

Hình 2.1 – Kiến trúc ứng dụng.....	18
Hình 2.2 – Mô hình phát triển ứng dụng	20
Hình 2.4 – Sơ đồ mô hình MVC trong Spring web	21
Hình 2.5 – Mô hình Spring web MVC Sequence.....	22
Hình 2.6 - Dữ liệu lưu trữ user và server	23
Hình 2.7 – Sơ đồ hoạt động của hệ thống.....	25
Hình 2.8 – Sơ đồ Use Case của hệ thống chính.....	26
Hình 2.9 – Sơ đồ Use case của chức năng quản lý và cấu hình dịch vụ	30
Hình 2.10 – Cấu hình tiles dependency	36
Hình 2.11– Khai báo cấu hình Tiles trong servlet-context	37
Hình 2.12 – Mô tả một giao diện với Tiles	37
Hình 2.13 – Cấu hình cây thư mục các tập tin định nghĩa cho Tiles Templates	37
Hình 2.14 – Khai báo dependency mongodb.....	38
Hình 2.15 – Khai báo lớp model User tương tác với mongodb.....	39
Hình 2.16 – Lớp DAO dùng để thao tác với cơ sở dữ liệu mongodb.....	39
Hình 2.17 – Khai báo các beans	39
Hình 2.18 – Mô tả cài đặt cho một RequestMapping (/dhcpinstall).....	40
Hình 2.19 – Các Controller tương ứng các chức năng	40
Hình 2.30 – Token trong khi sử dụng.....	41
Hình 2.31 – Đăng kí thành viên, thử nhập thông tin sai định dạng.....	42
Hình 2.33 - Đăng kí thành công, và giao diện đăng nhập.....	43
Hình 2.35 – Đăng nhập sai thông tin.....	43
Hình 2.37 – Trang chủ quản lý hiện danh sách server.....	43
Hình 2.39 – Giao diện thay đổi thông tin người dùng.....	44
Hình 2.40 - Nhập thông tin để thêm Server	44
Hình 2.41 –Yêu cầu cấu hình trên server để hệ thống hoạt động.....	44
Hình 2.42 – Trạng thái các server vừa thêm	45
Hình 2.43 – Kết quả hiển thị thông tin server.....	45
Hình 2.44 - Kết quả hiển thị biểu đồ.....	45
Hình 2.45 – Màn hình yêu cầu nhập sudoer user	46
Hình 2.46 – Nhập không đúng sudoer user	46
Hình 2.47 - Giao diện điều khiển power.....	46
Hình 2.48 – Cửa sổ xác nhận.....	47
Hình 2.49 – Máy chủ khởi động lại sau khi nhấn nút Reboot.....	47
Hình 2.50 – Giao diện cấu hình Network Configuration.....	47
Hình 2.51– Giao diện cấu hình Eth cho network card	47
Hình 2.52 – Giao diện cấu hình DNS nameserver.....	48

Hình 2.53 – Giao diện cấu hình trực tiếp trên file interfaces	48
Hình 2.54 – Cấu hình SSH	48
Hình 2.55 – Lưu thông tin thay đổi	49
Hình 2.56 – Kiểm tra file cấu hình trên server đã thay đổi	49
Hình 2.57 – Chức năng cấu hình vào file sshd_config	49
Hình 2.58 – Xem logs file của dịch vụ SSH	49
Hình 2.59 – Khi server chưa cài dịch vụ vsftpd	50
Hình 2.60 – Cấu hình thông số dịch vụ FTP	50
Hình 2.61 – Cấu hình từ file vsftpd.conf	50
Hình 2.62 – Sau khi cấu hình và kiểm tra thông tin trên server	51
Hình 2.63 – Xem logs hệ thống của dịch vụ vsftpd	51
Hình 2.64 – Cấu hình các thông số của dịch vụ DHCP	51
Hình 2.65 - Stop, Start, Restart, remove dịch vụ	52
Hình 2.66 – Thêm, sửa subnet	52
Hình 2.67 – Chức năng thêm, Sửa hostfixs	52
Hình 2.68 – Chức năng cấu hình trực tiếp bằng file conf	53
Hình 2.69 - Sau khi lưu thông tin cấu hình	53
Hình 2.70– Chức năng hiển thị logs và errors	53
Hình 2.71– Chức năng thống kê và quản lý người dùng	54
Hình 2.72 – Chức năng thay đổi thông tin người dùng	54
Hình 2.73– Chức năng View Logs	55
Hình 2.74 – Kiểm tra giao diện chính trên di động	55
Hình 2.75 – Các bản đồ (chưa hiển thị tốt trên màn hình dọc)	56
Hình 2.76 – Hiển thị các chức năng cấu hình	56
Hình 2.77 – Kiểm tra giao diện quản trị của admin	56

DANH MỤC BẢNG

.....❧.....

<i>Bảng 2.1 – Mô tả chức năng đăng kí</i>	<i>27</i>
<i>Bảng 2.2 – Mô tả chức năng đăng nhập</i>	<i>28</i>
<i>Bảng 2.3 – Mô tả chức năng thay đổi thông tin người dùng</i>	<i>28</i>
<i>Bảng 2.4 – Mô tả chức năng thêm server</i>	<i>29</i>
<i>Bảng 2.5 – Mô tả chức năng hiển thị danh sách và trạng thái server</i>	<i>30</i>
<i>Bảng 2.6 – Mô tả chức năng hiển thị và quan sát thông tin server</i>	<i>31</i>
<i>Bảng 2.7 – Mô tả chức năng quản lý nguồn</i>	<i>31</i>
<i>Bảng 2.8 – Mô tả chức năng cấu hình network card interfaces</i>	<i>32</i>
<i>Bảng 2.9 – Mô tả chức năng cấu hình dịch vụ SSH</i>	<i>33</i>
<i>Bảng 2.10 – Mô tả chức năng cài đặt và cấu hình dịch vụ FTP</i>	<i>34</i>
<i>Bảng 2.11 – Mô tả chức năng cài đặt và cấu hình dịch vụ DHCP</i>	<i>34</i>
<i>Bảng 2.12 – Mô tả chức năng thống kê và hiển thị thông tin thành viên</i>	<i>35</i>
<i>Bảng 2.13 – Mô tả chức năng hiển thị logs hoạt động</i>	<i>36</i>
<i>Bảng 2.14: Kết quả kiểm thử</i>	<i>57</i>

ABSTRACT

.....~O~.....

Nowadays, the advent of Cloud Computing with the convenience and reasonable prices makes use of a service like VPS , Cloud VPS or Server become easy (the average cost of \$5 a month for Cloud VPS). However, there are many problems in the management server for beginners. To resolve the difficulties that we decided to implement the thesis "Building management systems, network services controlled web based".

The main purpose of this thesis is to develop a set of tools to remotely administer network services of the server, supports system administrator managing server efficiently and quickly./.

TÓM TẮT

.....~O~.....

Hiện nay, với giá thành tương đối rẻ, và sự tiện dụng cùng với sự ra đời của Cloud, thì rất dễ dàng để có thể sử dụng một dịch vụ VPS, Cloud VPS, hoặc Server (sau đây xin gọi chung là “Server”) với giá chỉ từ 100.000đ một tháng với Cloud VPS. Bên cạnh đó vấn đề quản lý tập trung các Server ở nhiều nhà cung cấp và đối với người không chuyên cũng trở nên khó khăn cấp thiết. Để giải quyết phần nào khó khăn đó chúng tôi đã thực hiện đề tài “Xây dựng hệ thống quản lý, điều khiển các dịch vụ mạng bằng giao diện WEB”. Mục đích của đề tài là xây dựng một công cụ quản trị tập trung các dịch vụ mạng trên Server từ xa, hỗ trợ người quản trị có thể quản lý các Server của mình một cách hiệu quả và nhanh chóng.

PHẦN 1: GIỚI THIỆU

1.1 Đặt vấn đề

Với một người quản trị có sử dụng nhiều Server ở nhiều dịch vụ khác nhau tại nhiều nhà cung cấp khác nhau thì việc cấu hình, quan sát thường phải thực hiện trên mỗi Server được cài đặt và nếu có nhiều Server cài đặt nhiều dịch vụ thì việc quản lý sẽ trở nên khó khăn. Hiện tại cũng có không ít phần mềm, dịch vụ có thể điều khiển cài đặt các dịch vụ vào Server, nhưng thường là các ứng dụng chạy trên máy tính, sử dụng bằng dòng lệnh phức tạp, một số các hệ thống chạy trên web nhưng phải cài đặt vào chính Server được quản trị. Để giúp người quản lý có thể nhanh chóng cấu hình, cài đặt dịch vụ trên Server mọi lúc, mọi nơi, thuận tiện và thích hợp cho người dùng không chuyên thì một dịch vụ web trực quan và quản lý tập trung là cần thiết.

Ngoài ra nhằm để hiểu được phương thức hoạt động, các công nghệ hiện nay, để khai thác và tìm hiểu công nghệ nên cần thiết xây dựng một hệ thống dựa trên các công nghệ đó.

1.2 Lịch sử giải quyết

Hiện nay đã có nhiều công cụ hỗ trợ quản trị Server từ xa bằng dòng lệnh, hoặc quản trị chuyên sâu được cài đặt trên chính Server đó, nhưng ứng dụng Cấu hình và cài đặt dịch vụ từ xa trên nền web thì rất ít.

Trên thế giới có dịch vụ ServerPilot.io thành lập bởi Justin Samuel vào năm 2012 cung cấp chức năng tương tự, nhưng giá mỗi tháng sử dụng các dịch vụ cơ bản là 10\$ hoặc cấu hình các dịch vụ nâng cao hơn với giá 49\$.

Ngoài ra các công cụ nổi tiếng như cPanel (cpanel.com), Directadmin (directadmin.com), Webmin (webmin.com),... là các công cụ quản trị, cấu hình chuyên nghiệp từ trả phí đến miễn phí, ví dụ như webmin đòi hỏi phải cài đặt vào mỗi server, hoặc Cpanel hay directadmin phải mua license sử dụng cho từng server khác nhau.

Ở Khoa Công nghệ thông tin và truyền thông – Trường đại học Cần Thơ, có luận văn “Thiết kế hệ thống tích hợp hiển thị thông tin, quản lý thiết bị mạng và dịch vụ mạng” – Nguyễn Thành Nhân, Nguyễn Hữu Nhật - 2012 đã cho ra ứng dụng chức năng như vậy và đã có sản phẩm trên Desktop và tôi tham khảo nghiên cứu này để triển khai lên web và phát triển cho phép nhiều người sử dụng, khắc phục được việc phải sử dụng root trong cấu hình mà chỉ cần dùng tài khoản sudoer user.

Các sản phẩm, đề tài trên thường là những ứng dụng Desktop, không thể truy cập mọi lúc mọi nơi trên nhiều máy và nhiều hệ điều hành khác nhau, hay các ứng dụng chỉ chạy trên chính Server được cài đặt, không thể cùng quản lý đồng thời nhiều máy chủ. Vì vậy, với đề tài này sẽ cho phép tích hợp các chức năng quản trị từ xa, quản lý tập trung nhiều Server và chạy trên nền web để có thể truy cập mọi lúc, mọi nơi và đa dạng hệ điều hành, thiết bị khác nhau vì chạy trên nền Web.

1.3 Mục tiêu đề tài

Đề tài nhằm tìm hiểu và vận dụng các công nghệ hiện tại để xây dựng hệ thống điều khiển, cấu hình và theo dõi Server từ xa qua trên web.

Việc xây dựng hệ thống được tiến hành theo cách chia Module bao gồm 2 thành phần chính: Module giao diện web với nhiệm vụ quản lý người dùng và quản lý thông tin Server và kết nối các Module cung cấp thư viện để thao tác cài đặt và theo dõi các dịch vụ trên Server.

Theo phân công module được đảm trách trong báo cáo này là “Hệ thống web quản lý và thao tác cấu hình server”.

Mục tiêu cụ thể trong Module này:

- Xây dựng hệ thống web để triển khai các module cấu hình dịch vụ.
- Hỗ trợ người dùng cài đặt và cấu hình dịch vụ được hỗ trợ trên giao diện web bằng máy tính hoặc thiết bị cầm tay.
- Hỗ trợ Admin xem thông tin và quản lý người dùng.

1.4 Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu trong luận văn này là công nghệ lập trình web sử dụng Spring WEB MVC kết hợp với Bootstrap phục vụ cho việc cài đặt và cấu hình các dịch vụ mạng trên máy chủ từ xa thông qua giao thức SSH.

Ứng dụng hoạt động trên nền localhost, máy chủ Apache Tomcat và hệ quản trị cơ sở dữ liệu MongoDB.

Phạm vi chức năng chính giới thiệu trong báo cáo này là:

- Thiết kế hệ thống trên nền Spring WEB MVC.
- Quản lý người dùng.
- Quản lý các chức năng và các dịch vụ trên Server: Network Interfaces, SSH, DHCP, FTP, (Chỉ giới hạn ở 4 chức năng chính).

1.5 Nội dung nghiên cứu

Nội dung công việc được thực hiện theo thời gian với những công việc cụ thể:

- Tuần 1: Phân tích nhu cầu và mục tiêu của đề tài, tìm kiếm và tham khảo các tài liệu liên quan.
- Tuần 2: Tìm hiểu về Spring Tool, MongoDB và cài đặt.
- Tuần 3: Nghiên cứu cấu hình và lập trình thao tác cơ bản với spring MVC.
- Tuần 4: Nghiên cứu sử dụng Tiles Template để thiết kế giao diện cho website với Spring MVC.
- Tuần 5: Tìm hiểu cấu hình và thao tác Spring MVC với MongoDB, Xây dựng chức năng đăng kí và chứng thực người dùng.
- Tuần 6: Xây dựng chức năng thêm và quản lý thông tin Server.
- Tuần 7: Nghiên cứu kết nối và thao tác với server theo giao thức SSH bằng java thông qua thư viện jSch.
- Tuần 8,9,10,11,12,13,14,15,16: Nghiên cứu kết hợp các hàm điều khiển vào hệ thống
 - Tuần 8,9: Nghiên cứu cải tiến chức năng để có thể cấu hình dịch vụ với quyền Sudoer mà không cần user root. Xây dựng và thử nghiệm chức năng cấu hình Network Card Interfaces với sudoer user.
 - Tuần 10,11: Xây dựng và thử nghiệm chức năng cấu hình dịch vụ SSH.
 - Tuần 12,13: Xây dựng và thử nghiệm chức năng cài đặt và cấu hình dịch vụ FTP.
 - Tuần 14,15,16: Xây dựng và thử nghiệm chức năng cài đặt và cấu hình dịch vụ DHCP.
- Tuần 17: Xây dựng chức năng cập nhật thông tin Profile của người dùng. Xây dựng module quan sát, lấy thông tin Server và vẽ biểu đồ thể hiện.
- Tuần 18: Xây dựng trang quản lý dành cho Admin: chức năng thống kê cơ bản, chức năng quản lý người dùng, xem logs.
- Tuần 19: Thử nghiệm tổng thể, và sửa chữa các chức năng điều khiển Server. Viết và chỉnh sửa báo cáo.
- Tuần 20: Bảo vệ luận văn.

1.6 Những đóng góp chính

Về lý thuyết đề tài đã hoàn thành một số chức năng, ứng dụng các thư viện xử lý vào trong hệ thống nhiều người sử dụng.

Nâng cao khả năng hoạt động, linh hoạt hơn khi hoạt động được trên thiết bị cầm tay.

Khắc phục được khuyết điểm dùng user root để cấu hình dịch vụ của các ứng dụng khác.

Đã thiết kế và triển khai thành công hệ thống trên môi trường web đưa vào thử nghiệm.

1.7 Bố cục quyền luận văn

Nội dung chính của tài liệu này được trình bày ở các phần sau:

- Chương 1: Mô tả bài toán:

Chương này trình bày yêu cầu của đề tài, mô tả kiến trúc của hệ thống, phạm vi chức năng và triển khai của hệ thống. Đánh giá các giải pháp và cách tiếp cận các giải pháp được lựa chọn.

- Chương 2: Thiết kế và cài đặt giải pháp

Chương này trình bày kiến trúc tổng thể của hệ thống, trình bày các chức năng chính, kết cấu thiết kế và cách thức cài đặt triển khai ứng dụng theo ngôn ngữ JavaEE và Spring web MVC.

- Chương 3: Kiểm thử và đánh giá

Chương này của tài liệu sẽ trình bày mục tiêu kịch bản kiểm thử và đánh giá kết quả kiểm thử.

PHẦN 2: NỘI DUNG

CHƯƠNG 1: MÔ TẢ BÀI TOÁN

Chương này trình bày yêu cầu của đề tài, phát triển các tính năng dựa trên nền Spring Web MVC, mô tả kiến trúc của hệ thống, phạm vi chức năng và triển khai của hệ thống.

I. Yêu cầu đề tài

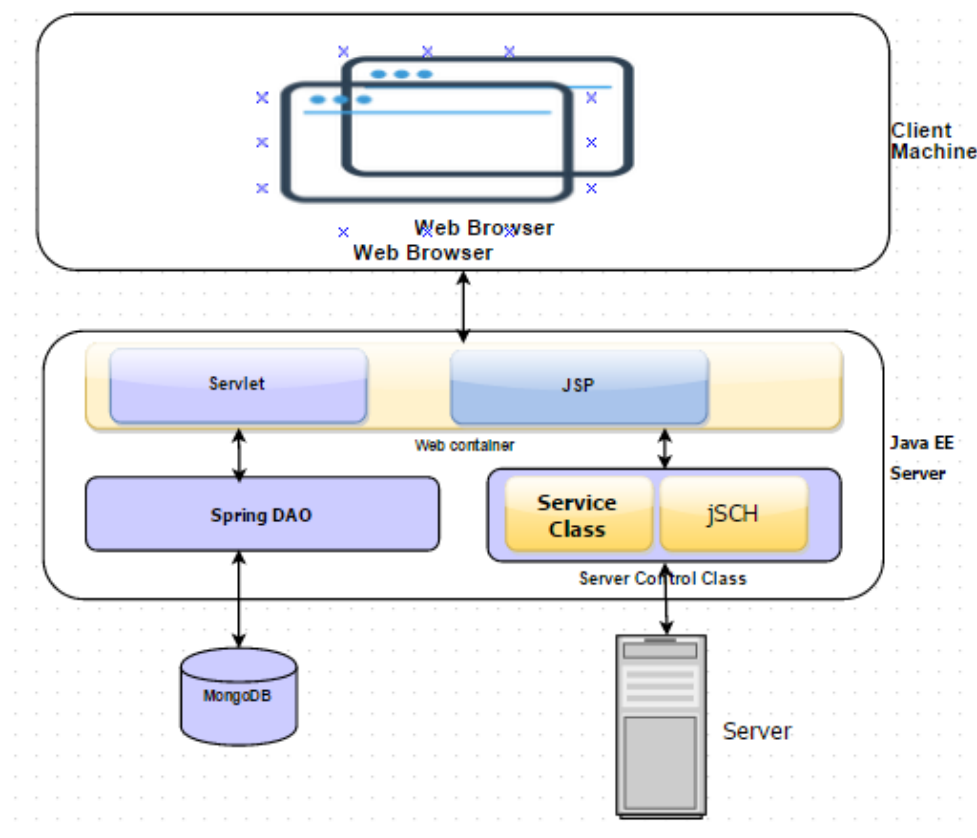
Việc thực hiện ứng dụng chạy trên nền web, và cho phép người dùng sử dụng vì vậy tính chất bảo mật, xử lý thông tin lưu trữ của người dùng và server của người dùng rất quan trọng. Vì vậy ứng dụng tạo và sử dụng một user riêng trên server người dùng, nhưng không có quyền hệ thống. Mặc định user này sẽ chỉ nhận các thông tin trạng thái của server người dùng và mật khẩu do người dùng tạo. Khi cấu hình các chức năng nâng cao, mới yêu cầu người dùng nhập thông tin của sudoer user, và chỉ lưu thông tin trong một phiên làm việc với mỗi server.

Đề tài hướng đến mục tiêu như sau:

- Đáp ứng yêu cầu nhiều người sử dụng. Có khả năng xử lý trực tuyến.
- Cấu hình dịch vụ không sử dụng root user để tăng bảo mật và chỉ cần sử dụng quyền sudoer.
- Chương trình chạy trên giao diện Web với kiến trúc Client-Server, hệ thống máy chủ ứng dụng phiên bản Ubuntu từ 12.04 hoặc CentOS từ 6.5, các máy trạm (windows xp đến windows 10, các distro Linux), và các thiết bị Smartphone, tablet (Android 4.0 trở lên, IOS 6 trở lên và Windows Phone 8 trở lên).
- Website hỗ trợ bảo mật hạn chế các lỗi dữ liệu chứng thực người dùng để bảo vệ thông tin của người dùng.

II. Mô hình kiến trúc hệ thống ứng dụng

Xây dựng ứng dụng nền web với kiến trúc phân tầng và cơ sở dữ liệu tập trung.



Hình 2.1 – Kiến trúc ứng dụng

Website viết theo cấu trúc Spring MVC với 3 thành phần Model, View và Controller hoạt động xung quanh một DispatcherServlet xử lý tất cả các yêu cầu Http request và trả về Http response. Các công việc xử lý yêu cầu được mô tả trong mô hình sau.

Người dùng sử dụng giao diện thông qua View để sử dụng và thao tác với hệ thống thông qua DispatcherServlet, các controller tương tác với các ServiceClass để thao tác đến server từ xa thông qua JSCH Framwork.

III. Phạm vi chức năng

Ứng dụng giải quyết các chức năng sau:

STT	Mô tả yêu cầu	Phân loại
Trang quản lý server, dành cho người dùng		
1	Người dùng đăng kí và đăng nhập vào hệ thống	Dữ liệu
2	Quản lý thông tin server	Dữ liệu
3	Quản lý cấu hình network trên server	Cấu hình

4	Quản lý cấu hình dịch vụ SSH trên server	Cấu hình
5	Cài đặt và cấu hình dịch vụ FTP trên server	Cấu hình
6	Cài đặt và cấu hình dịch vụ DHCP trên server	Cấu hình
Trang quản lý dành cho Admin hệ thống		
1	Quản lý thông tin người dùng	Dữ liệu
2	Xem thông tin logs	Dữ liệu

Phân tích giải pháp:

Đề tài nhằm ứng dụng vào triển khai sử dụng các hàm thư viện cấu hình và cài đặt dịch vụ trên server, chọn công nghệ lập trình Java EE và sử dụng Spring web MVC framework.

Các công nghệ mà đề tài sử dụng là:

- Hệ thống web phát triển theo mô hình MVC web.
- Cơ sở dữ liệu sử dụng MongoDB để lưu trữ thông tin.
- Layout sử dụng Tiles Framework để thiết kế
- Giao diện sử dụng Bootstrap và AJAX để tăng trải nghiệm người dùng.
- Đặc biệt đề tài tìm hiểu và cài đặt các chức năng giúp tương tác và cấu hình cho server Ubuntu.

IV. Phạm vi triển khai ứng dụng

Ứng dụng được triển khai trên localhost, hoặc Ubuntu Server 14.04 và cài đặt Jre 7, Apache Tomcat 7, MongoDB3.0

Ứng dụng cho phép người dùng thao tác và cấu hình từ xa các Server chạy hệ điều hành Ubuntu Server 14.04 trở lên, cài đặt và cấu hình Open SSH, enable chứng thực bằng mật khẩu và cho phép kết nối SSH từ xa.

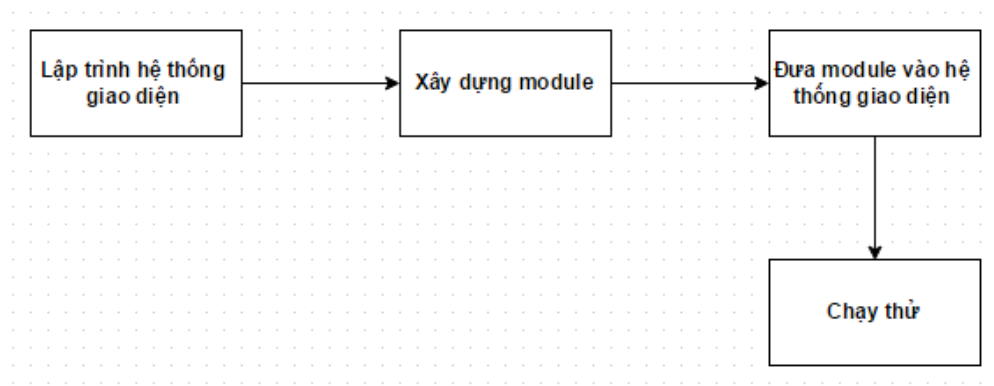
CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP

I. THIẾT KẾ GIẢI PHÁP

Chương này trình bày về hướng thiết kế ứng dụng, màn hình giao diện, thiết kế database và sơ đồ cách thức hoạt động của ứng dụng, cách thức cài đặt các thiết kế bằng ngôn ngữ Java và các kỹ thuật liên quan khác.

I.1.1. Hướng thiết kế hệ thống

Hệ thống điều khiển server được phát triển theo từng giai đoạn với từng module phát triển rộng ra thành một ứng dụng tổng.

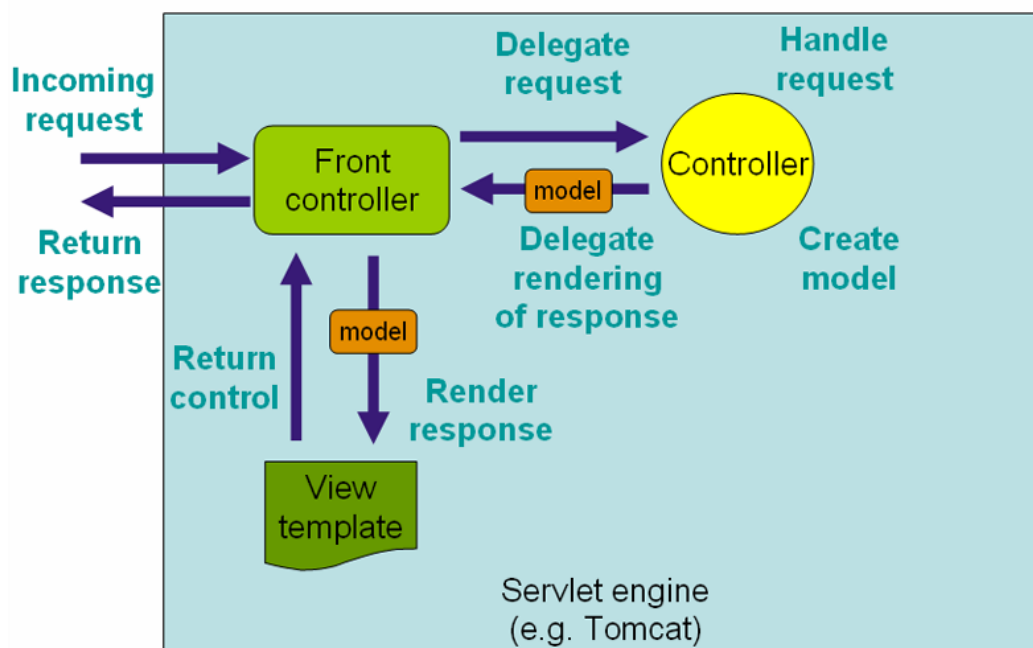


Hình 2.2 – Mô hình phát triển ứng dụng

Quy trình được chia làm 4 giai đoạn:

- Giai đoạn 1: Xây dựng hệ thống giao diện web
- Giai đoạn 2: Xây dựng module với các Service Control Class
- Giai đoạn 3: Đưa module vào hệ thống giao diện web
- Giai đoạn 4: Chạy thử nghiệm chức năng
- Quay về giai đoạn 2 để xây dựng module khác.

I.1.2. Mô hình Web MVC



Hình 2.4 – Sơ đồ mô hình MVC trong Spring web

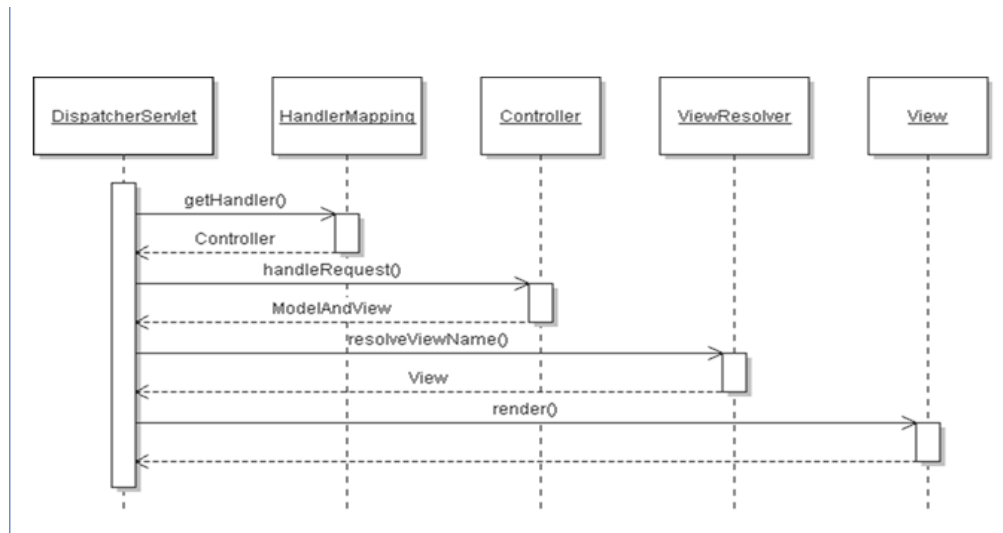
(Nguồn: <http://docs.spring.io/spring-framework/docs/2.0.8/reference/mvc.html>)

Mô hình web MVC với 3 thành phần chính:

Model: đóng gói dữ liệu ứng dụng và bao gồm POJO (Plain Old Java Object), hay ta còn gọi là các Domain Object.

View: Chịu trách nhiệm nhận giá trị của model và vẽ ra trang html mà trình duyệt có thể hiển thị được. Là các trang jsp giúp cho việc hiển thị dữ liệu hình thành giao diện.

Controller: Chịu trách nhiệm nhận và xử lý các yêu cầu từ người dùng và tạo các model phù hợp và trả về cho view, chứa các phương thức có khả năng nhận yêu cầu, xử lý yêu cầu, cập nhật dữ liệu, và chuyển tiếp dữ liệu trong ứng dụng.



Hình 2.5 – Mô hình Spring web MVC Sequence

(Nguồn: <http://jaiswaltraining.com/Spring/ch4.php>)

Mô tả đơn giản luồng xử lý trong mô hình:

Web Browser gửi request đến server thông qua các control trên form HTML hay JSP, hay query string url hay qua cookies.

DispatcherServlet – Controller đón nhận request thông qua Mapping và xác định Model tương ứng để tạo ra handle để đón nhận các giá trị nhập từ request để lưu trữ và xử lý.

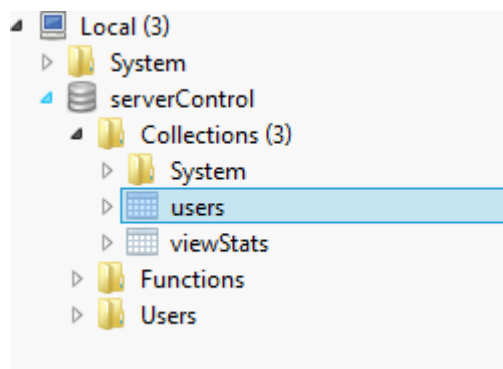
Model thực hiện xử lý, kết nối dữ liệu vật lý dưới DBMS (nếu có) và trả kết quả trả về cho Controller

Kết quả xử lý được chuyển vào DispatcherServlet – Controller, Controller thực hiện tạo hay lựa chọn View để từ đó đưa kết quả xử lý hay dữ liệu lấy từ Model để cập nhật lại trang kết quả View.

Controller render View cho người dùng để browser có thể trình bày dữ liệu trong Web Browser.

I.1.3. Thiết kế cơ sở dữ liệu

Hệ thống điều khiển cấu hình server sử dụng mongodb để lưu trữ dữ liệu người dùng, dữ liệu quan sát và thông tin server, nên phần dữ liệu lưu trữ không có cấu trúc, chỉ lưu những thông tin cần thiết như: users, servers, logs... Hình dưới đây là documents (tables) của hệ thống dưới sự hiển thị của Robomongo.



Hình 2.6 - Dữ liệu lưu trữ user và server

Chức năng mỗi document trong hệ thống:

Users: lưu trữ thông tin cơ bản của một người dùng, bao gồm:

- Email, password (được mã hóa dưới dạng md5) .
- Role với các role: -1 – người dùng bị khóa, 1-người dùng đang sử dụng, 2 - admin.
- firstName, lastName: để lưu trữ thông tin người dùng.
- Servers: lưu trữ danh sách các server của người dùng.

Sau đây là dữ liệu mẫu:

```
{
  "_id" : ObjectId("5549f1bb5cf17d16d494d96f"),
  "_class" : "vn.edu.cit.model.User",
  "email" : "thanhnhhan2tn@gmail.com",
  "passWord" : "25d55ad283aa400af464c76d713c07ad",
  "role" : 1,
  "firstName" : "Nhan",
  "lastName" : "Thai Thanh",
  "servers" : [ {}]
}
```

Servers: lưu thông tin về server của mỗi người dùng sử dụng hệ thống.

- ServerAddress: địa chỉ kết nối
- Port: cổng SSH để kết nối đến server.
- ServerName: tên người dùng đặt để nhận dạng.
- ServerUsername: tên người dùng, mặc định là svcontrol để nhận biết được các tình trạng cơ bản của server.
- ServerPassword: mật khẩu của người dùng svcontrol
- Status: thể hiện thông tin của servers

Sau đây là dữ liệu mẫu:

```
{  
  
    "serverAddress" : "130.211.103.78",  
  
    "port" : 22,  
  
    "serverName" : "Google VPS 2",  
  
    "serverUsername" : "svcontrol",  
  
    "serverPassword" : "12345",  
  
    "status" : []  
  
}
```

Logs: Lưu thông tin về các hoạt động của hệ thống

- dateTime: lưu ngày giờ của hoạt động.
- logContent: lưu ghi chú hoạt động

Sau đây là dữ liệu mẫu

```
{  
  
    "dateTime" : "20:08 09-05-2015",  
  
    "logContent" : "New user created: test@servercontrol.test",  
  
}
```

I.1.4. Các chức năng trong hệ thống quản lý server

I.1.4.1. Sơ đồ hoạt động của hệ thống



Hình 2.7 – Sơ đồ hoạt động của hệ thống

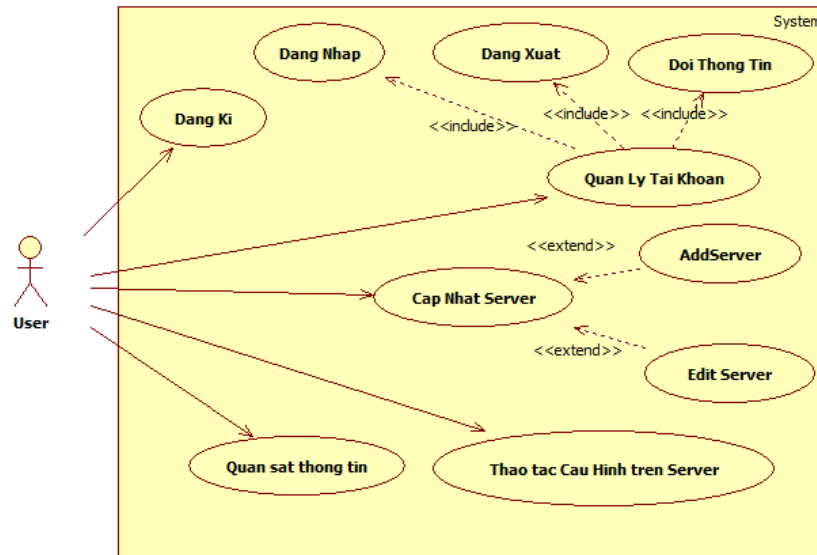
Mô tả hoạt động:

- Đăng ký tài khoản: đây là chức năng bắt buộc mà người dùng để sử dụng hệ thống phải thực hiện.
- Đăng nhập: Sau khi đăng ký, người dùng phải đăng nhập vào hệ thống.
- AddServer: hoạt động này dùng cho người dùng muốn thêm thông tin server mới vào hệ thống quản lý.
- Thay đổi thông tin người dùng: hoạt động này dành cho người dùng muốn thay đổi thông tin của mình như tên, họ, mật khẩu,...
- Để tiến hành cấu hình, quan sát một server, người dùng phải chọn một server cụ thể được liệt kê trên giao diện.
- Quan sát các thông tin server: hoạt động này cho người dùng quan sát tình trạng hiện thời của server.
- Service Config: Hoạt động này cho phép người dùng tiến hành cấu hình và cài đặt các dịch vụ của server.

- Quản lý Power, Cấu hình NIC, Cấu hình Ftp, Cấu hình SSH, cấu hình DHCP server: các hoạt động này cho phép cấu hình các dịch vụ trên server.

I.1.4.2. Các chức năng quản lý hệ thống chính

a. Sơ đồ Use case của chức năng quản lý hệ thống chính



Hình 2.8 – Sơ đồ Use Case của hệ thống chính

Chi tiết các chức năng chính nằm trong nhóm chức năng quản lý hệ thống chính:

b. Chức năng đăng kí - F01

Mã chức năng	F01
Tên chức năng	Đăng kí thành viên
Mức độ ưu tiên	Cao
Nội dung	Đây là chức năng đầu tiên của hệ thống, để sử dụng hệ thống, người dùng phải đăng kí thành viên.
Người dùng	Người muốn sử dụng hệ thống
Xử lý	Sử dụng Object name vn.cit.contacts.User và document users trên mongodb để kết nối dữ liệu.

	<p>Dữ liệu đầu vào: email, lastName, firstName, password.</p> <p>Sau khi nhập thông tin đăng kí, client thực hiện kết nối đến server và thực thi request registerAction().</p> <p>Server khi nhận được yêu cầu sẽ thực hiện 1 truy vấn xuống database (mongodb) để thêm tài khoản với thông tin được truyền từ client.</p>
Kết quả	Người dùng mới được tạo trong cơ sở dữ liệu.
Ghi chú	Nếu email đăng kí trùng với email của người dùng khác trong cơ sở dữ liệu, thì đăng kí không thành công.

Bảng 2.1 – Mô tả chức năng đăng kí

c. Chức năng đăng nhập- F02

Mã chức năng	F02
Tên chức năng	Đăng nhập
Mức độ ưu tiên	Cao
Nội dung	Đây là chức năng bắt buộc của hệ thống, để sử dụng hệ thống người dùng phải đăng nhập vào hệ thống.
Người dùng	Thành viên
Xử lý	<p>Sử dụng object name users và documents trên mongodb là <i>users</i> cho client kết nối.</p> <p>Sau khi nhập thông tin đăng nhập (email và password), client thực hiện kết nối đến server và thực thi request checkLogin(email, password).</p> <p>Server khi nhận được yêu cầu xác thực sẽ thực hiện 1 truy vấn xuống database (mongodb) để tìm tài khoản với thông tin được truyền từ client.</p>

Kết quả	<p>Nếu tìm thấy user trong database, server trả về kết quả true và chuyển vào trang Dashboard, nếu không tìm thấy server trả về false và thông báo lỗi.</p> <p>Nếu client nhận về kết quả từ server là “true”, chuyển vào trang quản lý, nếu nhận kết quả trả về “false” hiện thông báo và tiếp tục cho người dùng nhập lại thông tin.</p>
Ghi chú	

Bảng 2.2 – Mô tả chức năng đăng nhập

d. Chức năng thay đổi thông tin - F03

Mã chức năng	F03
Tên chức năng	Thay đổi thông tin
Mức độ ưu tiên	Thấp
Nội dung	Chức năng này nhằm giúp người dùng thay đổi thông tin của mình, bao gồm mật khẩu, số điện thoại, tên, họ,..
Người dùng	Thành viên
Xử lý	<p>Người dùng nhập các thông tin yêu cầu, nhấn Save để lưu thay đổi.</p> <p>Nếu không muốn đổi mật khẩu, người dùng có thể để trống.</p> <p>Người dùng có thể thay đổi avatar bằng các vào trang dịch vụ Gravatar để thay đổi.</p>
Kết quả	Thông tin của người dùng được cập nhật vào cơ sở dữ liệu.
Ghi chú	Người dùng không được phép thay đổi email

Bảng 2.3 – Mô tả chức năng thay đổi thông tin người dùng

e. Chức năng AddServer – F04

Mã chức năng	F04
--------------	-----

Tên chức năng	AddServer
Mức độ ưu tiên	Cao
Nội dung	Đây là chức năng cho phép người dùng thêm thông tin server vào hệ thống để quản lý.
Người dùng	Thành viên
Xử lý	Người dùng nhập các thông tin yêu cầu, nhấn Save để lưu thay đổi. Sau khi save, hệ thống sẽ được chuyển đến trang hiển thị yêu cầu người dùng tiến hành cấu hình cơ bản trên Server để hệ thống có thể tương tác được với server.
Kết quả	Một server mới được thêm vào cơ sở dữ liệu
Ghi chú	

Bảng 2.4 – Mô tả chức năng thêm server

f.Hiển thị danh sách và trạng thái Server – F05

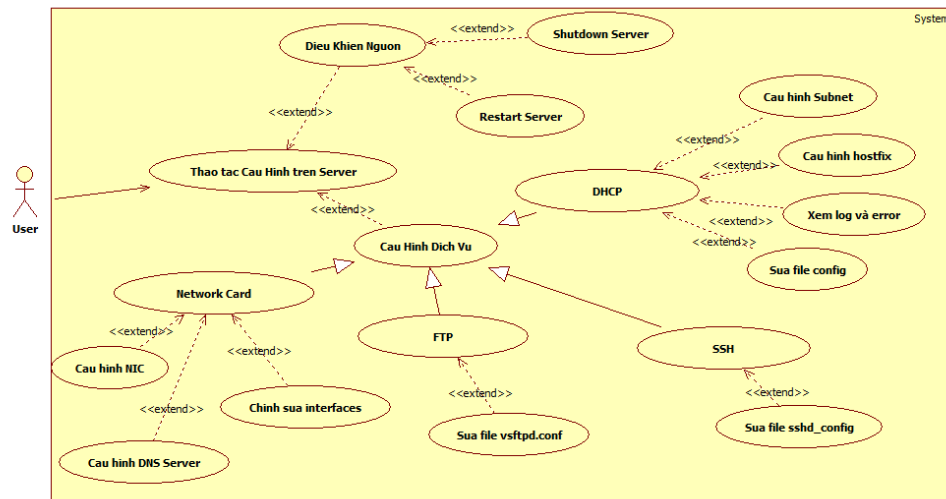
Mã chức năng	F05
Tên chức năng	Hiện danh sách Server
Mức độ ưu tiên	Cao
Nội dung	Đây là chức năng ở màn hình chính sau khi đăng nhập, và thêm các server. Và là đường chính để người dùng có thể thao tác với các Servers
Người dùng	Thành viên
Xử lý	Sau khi đăng nhập vào hệ thống, và đã thêm các Servers, thì hệ thống sẽ lấy thông tin từ cơ sở dữ liệu mongoDB và hiển thị lên giao diện này. Sau khi hiển thị danh sách server, thư viện Ajax sẽ lấy thông tin trạng thái của Server để hiển thị trạng thái của server và cho phép cấu hình

Kết quả	Các server được liệt kê trên giao diện
Ghi chú	

Bảng 2.5 – Mô tả chức năng hiển thị danh sách và trạng thái server

I.1.4.3. Chức năng quản lý và cấu hình các dịch vụ trên server

a. Sơ đồ Use case của chức năng quản lý và cấu hình dịch vụ



Hình 2.9 – Sơ đồ Use case của chức năng quản lý và cấu hình dịch vụ

b. Chức năng hiển thị và quan sát thông tin server – F06

Mã chức năng	F06
Tên chức năng	Hiển thị và quan sát thông tin server
Mức độ ưu tiên	Trung bình
Nội dung	Ở chức năng này, khi người dùng vào trang hiển thị thông tin, hệ thống sẽ tiến hành lấy thông tin từ server và hiển thị lên giao diện, đồng thời vẽ biểu đồ hoạt động của server dựa trên thông tin trạng thái đó.
Người dùng	Thành viên
Xử lý	Thông tin được lấy ngay khi người dùng xem trang bằng ajax và vẽ biểu đồ hoạt động, ứng dụng không lưu được

	hoạt động của server trong thời gian dài và trong khi tắt hệ thống.
Kết quả	Thông tin được hiển thị trên giao diện và vẽ biểu đồ tổng quát.
Ghi chú	

Bảng 2.6 – Mô tả chức năng hiển thị và quan sát thông tin server

c. Chức năng quản lý nguồn

Mã chức năng	F07
Tên chức năng	Quản lý nguồn
Mức độ ưu tiên	Cao
Nội dung	Chức năng này giúp người dùng có thể shutdown hoặc reboot server từ xa bằng nút bấm trên giao diện.
Người dùng	Thành viên
Xử lý	<p>Khi người dùng truy cập giao diện Service Config, nhập user/password của sudoer, giao diện điều khiển nguồn sẽ hiển thị như trên.</p> <p>Khi người dùng chọn vào nút Shutdown (màu đỏ) thì hệ thống hỏi xác nhận có thực hiện thao tác hay không, nếu người dùng bấm Yes, hệ thống sẽ thực thi hàm Shutdown(sv) để gửi lệnh yêu cầu shutdown server.</p> <p>Khi người dùng chọn vào nút Reboot (màu cam) thì hệ thống hỏi xác nhận có thực hiện thao tác hay không, nếu người dùng bấm Yes, hệ thống sẽ thực thi hàm Reboot (sv) để gửi lệnh yêu cầu Reboot server.</p>
Kết quả	Server sẽ shutdown hoặc reboot theo thao tác.
Ghi chú	

Bảng 2.7 – Mô tả chức năng quản lý nguồn

d. Chức năng quản lý cấu hình network card interface

Mã chức năng	F08
Tên chức năng	Quản lý cấu hình network card interface
Mức độ ưu tiên	Cao
Nội dung	Giúp người dùng có thể cấu hình cơ bản các thông tin trong cấu hình network card interface
Người dùng	Thành viên
Xử lý	<p>Khi vào trang cấu hình network card interface, hệ thống sẽ load thông tin cấu hình từ server, và hiển thị lên giao diện</p> <p>Người dùng tiến hành thao tác thêm sửa các thông tin, và lưu lại.</p> <p>Người dùng muốn cấu hình nâng cao hơn bằng việc sửa trực tiếp trong file interfaces bằng cách vào mục Edit config file...</p>
Kết quả	Thông tin mới được cập nhật và upload lên server
Ghi chú	

Bảng 2.8 – Mô tả chức năng cấu hình network card interfaces

e. Chức năng quản lý cấu hình dịch vụ SSH

Mã chức năng	F09
Tên chức năng	Quản lý cấu hình dịch vụ SSH
Mức độ ưu tiên	Cao
Nội dung	Giúp người dùng có thể cấu hình cơ bản các thông tin của dịch vụ SSH
Người dùng	Thành viên
Xử lý	<p>Khi vào trang cấu hình SSH, hệ thống sẽ load thông tin cấu hình từ server, và hiển thị lên giao diện</p> <p>Người dùng tiến hành thao tác thêm sửa các thông tin, và lưu lại.</p>

	Người dùng muốn cấu hình nâng cao hơn bằng việc sửa trực tiếp trong file <code>sshd_config</code> bằng cách vào mục Edit config file...
Kết quả	<p>Thông tin mới được cập nhật và upload lên server.</p> <p>Dịch vụ SSH sẽ tự restart lại sau khi người dùng cập nhật.</p> <p>Chức năng xem logs của dịch vụ SSH giúp hiển thị logs của dịch vụ.</p>
Ghi chú	SSH là phương thức chính để kết nối với hệ thống, nếu người dùng cấu hình không chính xác thì hệ thống sẽ mất kết nối với server, buộc lòng người dùng phải đăng nhập qua server theo cách thủ công thông thường.

Bảng 2.9 – Mô tả chức năng cấu hình dịch vụ SSH

f.Chức năng cài đặt và cấu hình dịch vụ FTP

Mã chức năng	F10
Tên chức năng	Cài đặt và cấu hình dịch vụ FTP
Mức độ ưu tiên	Cao
Nội dung	Giúp người dùng có thể cài đặt và cấu hình cơ bản các thông tin của dịch vụ FTP.
Người dùng	Thành viên
Xử lý	<p>Khi vào trang cấu hình FTP, hệ thống sẽ load thông tin cấu hình từ server, và hiển thị lên giao diện</p> <p>Người dùng tiến hành thao tác thêm thay đổi các thông tin cần thiết, và lưu lại.</p> <p>Người dùng muốn cấu hình nâng cao hơn bằng việc sửa trực tiếp trong file <code>vsftpd.conf</code> bằng cách vào mục Edit config file...</p>

	Chức năng xem logs của dịch vụ FTP giúp hiển thị logs của dịch vụ.
Kết quả	Thông tin mới được cập nhật và upload lên server. Dịch vụ FTP sẽ tự restart lại sau khi người dùng cập nhật.
Ghi chú	

Bảng 2.10 – Mô tả chức năng cài đặt và cấu hình dịch vụ FTP

g. Chức năng cài đặt và cấu hình dịch vụ DHCP

Mã chức năng	F11
Tên chức năng	Cài đặt và cấu hình dịch vụ DHCP
Mức độ ưu tiên	Cao
Nội dung	Giúp người dùng có thể cài đặt và cấu hình cơ bản các thông tin của dịch vụ DHCP.
Người dùng	Thành viên
Xử lý	Khi vào trang cấu hình DHCP, hệ thống sẽ load thông tin cấu hình từ server, và hiển thị lên giao diện Người dùng tiến hành thao tác thêm thay đổi các thông tin cần thiết, và lưu lại. Người dùng muốn cấu hình nâng cao hơn bằng việc sửa trực tiếp trong file msc.conf bằng cách vào mục Edit config file... Chức năng xem logs của dịch vụ DHCP giúp hiển thị logs của dịch vụ.
Kết quả	Thông tin mới được cập nhật và upload lên server. Dịch vụ DHCP sẽ tự restart lại sau khi người dùng cập nhật.
Ghi chú	

Bảng 2.11 – Mô tả chức năng cài đặt và cấu hình dịch vụ DHCP

I.1.4.4. Chức năng của trang quản lý Admin

a. Chức năng quản lý thành viên– F12

Mã chức năng	F12
Tên chức năng	Quản lý thành viên
Mức độ ưu tiên	Cao
Nội dung	Giúp Admin hệ thống có thể xem thông tin, quản lý thành viên.
Người dùng	Admin
Xử lý	Khi đăng nhập vào hệ thống, nếu người dùng có quyền quản trị Admin, sẽ có chức năng AdminCP. Vào giao diện AdminCP, hệ thống sẽ load thông tin các user và hiển thị trên giao diện. Admin có thể sửa thông tin thành viên, đổi nhóm thành viên,..
Kết quả	Danh sách thành viên được hiển thị. Thông tin thành viên được cập nhật
Ghi chú	

Bảng 2.12 – Mô tả chức năng thống kê và hiển thị thông tin thành viên

b. Chức năng hiển thị Logs được ghi lại của hệ thống – F13

Mã chức năng	F13
Tên chức năng	Hiển thị logs hoạt động
Mức độ ưu tiên	Cao
Nội dung	Giúp Admin hệ thống có thể xem thông tin log của hệ thống.
Người dùng	Admin
Xử lý	Chức năng này dành cho admin quản trị hệ thống, giúp admin biết được log hoạt động được hệ thống ghi lại.

Kết quả	Danh sách logs hiển thị trên giao diện.
Ghi chú	

Bảng 2.13 – Mô tả chức năng hiển thị logs hoạt động

II. CÀI ĐẶT GIẢI PHÁP

Từ các chức năng được liệt kê ở trên, phần này sẽ trình bày cách triển khai ứng dụng, các giải pháp áp dụng.

II.1. Cài đặt cấu trúc template website với Tiles

Giao diện web được chia nhiều thành phần và các thành phần này có thể sử dụng lại giữa các trang khác nhau của trang web khác nhau, và nếu sau này có 1 sự thay đổi nào thì tất cả các trang đều phải thay đổi. Do đó chúng ta sử dụng cơ chế templatzation để thiết kế layout ứng dụng web một cách có cấu trúc và sử dụng layout đó cho việc thiết kế giao diện thay vì phải soạn lại nhiều lần các thành phần của trang web thì Tiles Framework hỗ trợ chúng ta làm việc này.

Một cách bố trí giao diện phổ biến của các website được định nghĩa trong một file cấu hình trung tâm và giao diện này có thể được kế thừa trên tất cả các trang web của ứng dụng web.

II.1.1. Cấu hình tiles dependency

Để có thể làm việc được với Tiles, chúng ta cần phải cấu hình thư viện của tiles, ở đây với sự hỗ trợ của maven, chúng ta sẽ tiến hành khai báo dependency trong file pom.xml của project.

```
<!-- Tiles -->
<dependency>
  <groupId>org.apache.tiles</groupId>
  <artifactId>tiles-extras</artifactId>
  <version>3.0.5</version>
</dependency>
```

Hình 2.10 – Cấu hình tiles dependency

II.1.2. Cấu hình cấu trúc giao diện trong tiles*.xml

Để cấu hình Tiles, một file xml được định nghĩa trong thư mục WEB-INF của spring project, và định nghĩa một listener trong servlet-context.xml.

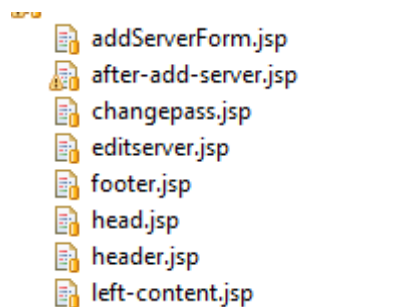
```
<!-- Tiles -->
<beans:bean id="viewResolver"
  class="org.springframework.web.servlet.view.UrlBasedViewResolver">
  <beans:property name="viewClass">
    <beans:value>
      org.springframework.web.servlet.view.tiles3.TilesView
    </beans:value>
  </beans:property>
</beans:bean>
<beans:bean id="tilesConfigurer"
  class="org.springframework.web.servlet.view.tiles3.TilesConfigurer">
  <beans:property name="definitions">
    <beans:list>
      <beans:value>/WEB-INF/*-tiles.xml</beans:value>
    </beans:list>
  </beans:property>
</beans:bean>
```

Hình 2.11– Khai báo cấu hình Tiles trong servlet-context

Trong file tiles.xml chúng ta định nghĩa một template có tên là template.default. Giao diện này chứa các thông tính như : Head, Header, Left, Right, Sidebar và Footer. Sau đó giao diện này sẽ được kế thừa và xác định giao diện cho các trang *.jsp tương ứng với mỗi template.

```
<tiles-definitions>
  <definition name="template.default" template="/WEB-INF/views/themes/default/main.jsp">
    <put-attribute name="head"
      value="/WEB-INF/views/themes/default/modules/head.jsp" />
    <put-attribute name="header"
      value="/WEB-INF/views/themes/default/modules/header.jsp" />
    <put-attribute name="sidebar"
      value="/WEB-INF/views/themes/default/modules/sidebar.jsp" />
    <put-attribute name="left-content"
      value="/WEB-INF/views/themes/default/modules/left-content.jsp" />
    <put-attribute name="right-content"
      value="/WEB-INF/views/themes/default/modules/right-content.jsp" />
    <put-attribute name="footer"
      value="/WEB-INF/views/themes/default/modules/footer.jsp" />
    <put-attribute name="addServerForm"
      value="/WEB-INF/views/themes/default/modules/addServerForm.jsp" />
    <put-attribute name="LoginForm"
      value="/WEB-INF/views/themes/default/modules/LoginForm.jsp" />
    <put-attribute name="mainContent"
      value="/WEB-INF/views/themes/default/modules/mainContent.jsp" />
  </definition>
```

Hình 2.12 – Mô tả một giao diện với Tiles



Hình 2.13 – Cấu hình cây thư mục các tập tin định nghĩa cho Tiles Templates

II.2. Cấu hình Mongo kết nối với các Model

II.2.1. Cấu hình mongodb dependency

Để làm việc với mongodb, phải cài đặt thư viện mongo vào project, ở đây sẽ tiến hành khai báo trong pom.xml với nội dung như sau:

```
<!-- MongoDB -->
<dependency>
    <groupId>org.springframework.data</groupId>
    <artifactId>spring-data-mongodb</artifactId>
    <version>1.7.0.RELEASE</version>
</dependency>

<dependency>
    <groupId>org.mongodb</groupId>
    <artifactId>mongo-java-driver</artifactId>
    <version>3.0.0</version>
</dependency>
```

Hình 2.14 – Khai báo dependency mongodb

II.2.2. Định nghĩa các Model tương ứng với các Documents

Để thao tác dữ liệu với mongodb, ta phải định nghĩa các Class tương ứng ở đây là lớp Model trong mô hình MVC. Với các hàm getter setter cho phép thao tác với dữ liệu của đối tượng.

Sau khi định nghĩa các Model, ta thiết lập các lớp DAO để tương tác với cơ sở dữ liệu, DAO bao gồm các phương thức thao tác: create, get, update, ..

@Document (collection="collectionname"): Mô tả tên của document tương ứng trong mongodb

@Id : Tương ứng với Id trong mongodb.

- Sau đây là ví dụ từ lớp User và documents users trong mongodb.

```
12 @Document(collection = "users")
13 public class User {
14     @Id
15     private String id;
16     private String email;
17     private String password;
18     private int role;
19     private String firstName;
20     private String lastName;
21     private String sdt;
22     List<Server> servers;
23     private String date;
24
25     public String getId() {}
26
27     public void setId(String id) {}
28
29     public String getDate() {}
30
31     public void setDate(String date) {}
32
33     public User() {}
34
35     public User(String email, String password, int role, String firstName, String lastName,
36
37
```

Hình 2.15 – Khai báo lớp model User tương tác với mongodb

```
public interface UserDAO {  
    public void createUser(User user);  
  
    public User getUser(String email);  
  
    public void updateUser(User user);  
  
    public List<User> getUsers();  
}
```

Hình 2.16 – Lớp DAO dùng để thao tác với cơ sở dữ liệu mongodb

II.2.3. Khai báo các beans tương tác với mongodb

Để Model và documents của mongodb tương tác với nhau, ta khai báo trong tập tin servlet-context.xml các beans tương ứng với từng Model.

```
<beans:bean id="userDAO" class="vn.edu.cit.dao.UserDAOImpl" />  
<beans:bean id="serverDAO" class="vn.edu.cit.dao.ServerDAOImpl" />  
<beans:bean id="logDAO" class="vn.edu.cit.dao.LogDAOImpl" />
```

Hình 2.17 – Khai báo các beans

- Document *users* tương ứng với Model *User*

```
{  
    "_id" : ObjectId("5549f1bb5cf17d16d494d96f"),  
    "_class" : "vn.edu.cit.model.User",  
    "email" : "thanhnhhan2tn@gmail.com",  
    "passWord" : "25d55ad283aa400af464c76d713c07ad",  
    "role" : 1,  
    "firstName" : "Nhan",  
    "lastName" : "Thai Thanh",  
    "servers" : [ {}]  
}
```

II.2. Cấu hình RequestMapping trong Controller để tương tác với Views

Để các Controller có thể nhận và xử lý các yêu cầu từ client, cần khai báo các RequestMapping tương ứng với từng View của từng chức năng.

II.2.1. Cấu hình Controller và các RequestMapping

@Controller: Annotation cho phép thông tin cho hệ thống class được sử dụng là Controller

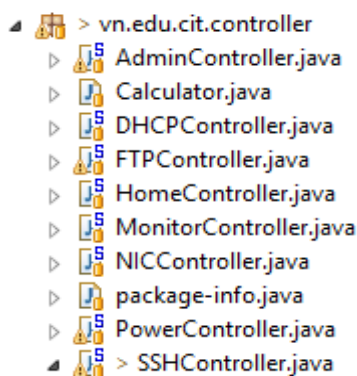
@RequestMapping(value="<view", method="<methodname"): Định nghĩa các Request với value là các view tương ứng và method là các phương thức nhận từ client.

```
@RequestMapping(value = "/serviceconfig/dhcpinstall/{ip}/{cc}", method = RequestMethod.GET)
public String dhcpSetup(HttpServletRequest request, HttpSession session, @PathVariable(value = "ip") String ip,
    @PathVariable(value = "cc") String c, ModelMap mm, RedirectAttributes redirectAtt) {
    User user = (User) session.getAttribute("user");
    String cc = (String) session.getAttribute("cc");
    if (user != null && cc.equals(c)) {
        Server server = serverDAO.getServer(user, ip);
        Server sv = new Server(server);
        sv.setServerUsername((String) session.getAttribute("sudouser"));
        sv.setServerPassword((String) session.getAttribute("sudopass"));

        DHCPConfig dhcpconfig = new DHCPConfig();
        ServerConfig serverConf = new ServerConfig();
        if (serverConf.checkSudoer(sv)) {
            Boolean check = dhcpconfig.Install(sv);
            if (check) { // return true
                _log.info("Install DHCP service ");
                redirectAtt.addFlashAttribute("display", "block");
                redirectAtt.addFlashAttribute("message", "(Install DHCP Success!)");
            } else {
                _log.info("Install DHCP service Failed");
                redirectAtt.addFlashAttribute("display", "block");
                redirectAtt.addFlashAttribute("message", "(Install DHCP failed!)");
            }
            return "redirect:/services/" + ip + "/" + c;
        } else {
            // check sudoer user
            return "redirect:/";
        }
    } else {
        return "redirect:/login";
    }
}
```

Hình 2.18 – Mô tả cài đặt cho một RequestMapping (/dhcpinstall)

Tương ứng từng chức năng, phải khai báo các Controller tương ứng, sau đây là các Controller được định nghĩa trong hệ thống:



Hình 2.19 – Các Controller tương ứng các chức năng

II.2.2. Thiết lập token để xác minh phiên làm việc của người dùng, nâng cao bảo mật.

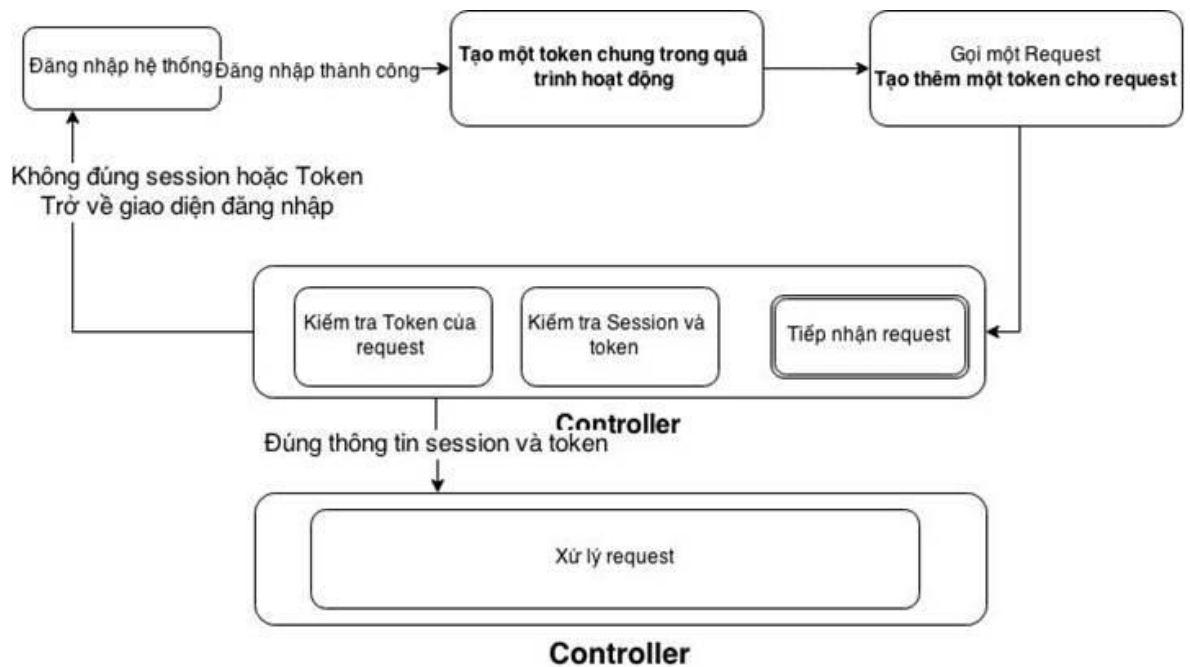
Khi sử dụng hệ thống, khả năng người dùng bị khai thác thông tin rất cao, kiểm thử trong trường hợp người dùng đang sử dụng hệ thống để cấu hình một server từ xa, và nhận được một email chứa link liên kết thực thi lệnh shutdown hàng

loạt các server. Thì hệ thống sẽ gửi lệnh shutdown như người dùng đang nhấn lệnh shutdown từ giao diện.

Để tránh được việc này, ứng dụng sử dụng thêm giải pháp tạo token, như một khóa phụ, để xác minh có đúng người dùng thao tác hay không.

Để khởi tạo token, khi người dùng đăng nhập trong hệ thống sử dụng sinh một mã ngẫu nhiên bằng cách mã hóa MD5 chuỗi bao gồm: ip của người dùng, email và password.

Đây là mã token sinh ra và hoạt động trong quá trình làm việc:



Hình 2.30 – Token trong khi sử dụng hệ thống

CHƯƠNG 3: KIỂM THỬ VÀ ĐÁNH GIÁ

I. MỤC TIÊU KIỂM THỬ

Kiểm thử hệ thống sau khi xây dựng nằm một số mục tiêu sau:

- Các chức năng hoạt động tốt, không lỗi.
- Đáp ứng yêu cầu bài toán và kỹ thuật trong quá trình thiết kế.
- Các chức năng hoạt động ổn định, đúng với mong đợi.
- Ứng dụng xử lý dữ liệu chính xác, phù hợp với nhu cầu.

II. KỊCH BẢN KIỂM THỬ VÀ ĐÁNH GIÁ

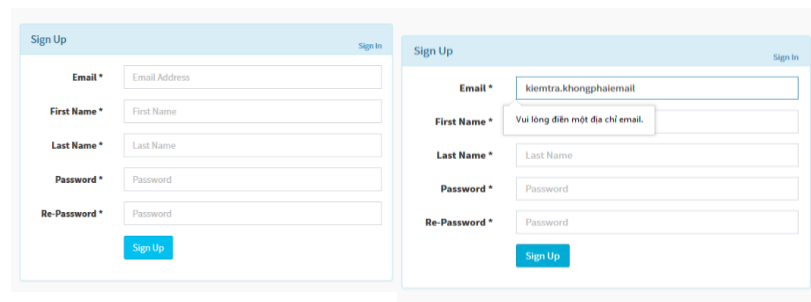
Hệ thống được kiểm thử 2 môi trường:

- Môi trường tomcat 8 tích hợp vào Spring tool suite để build localhost và chạy trên hệ điều hành Windows. Địa chỉ ứng dụng: <http://localhost:8080/cit/>
- Môi trường tomcat 8, cài đặt trên Server chạy hệ điều hành Ubuntu Server, Jre 7.
- Quy trình thực hiện được phân chia như sau:

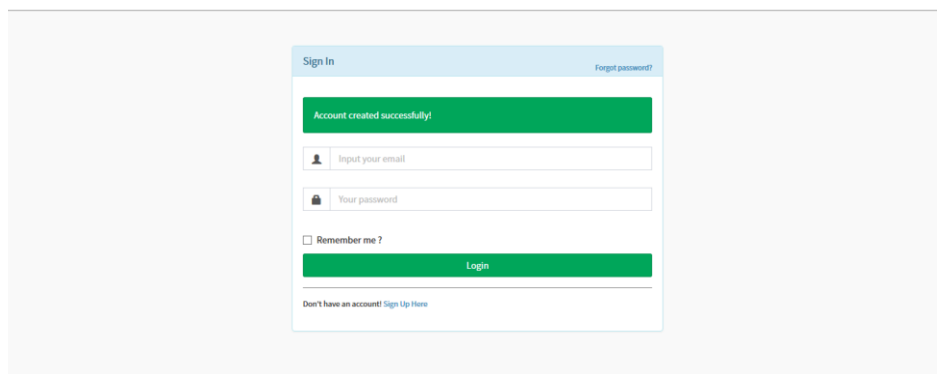
II.1. Quản lý hệ thống chính

II.1.1. Chức năng đăng kí

Để sử dụng hệ thống, người dùng cần đăng kí thông tin vào hệ thống. Nếu đăng kí trùng thông tin email, hệ thống cảnh báo và yêu cầu đăng kí lại.



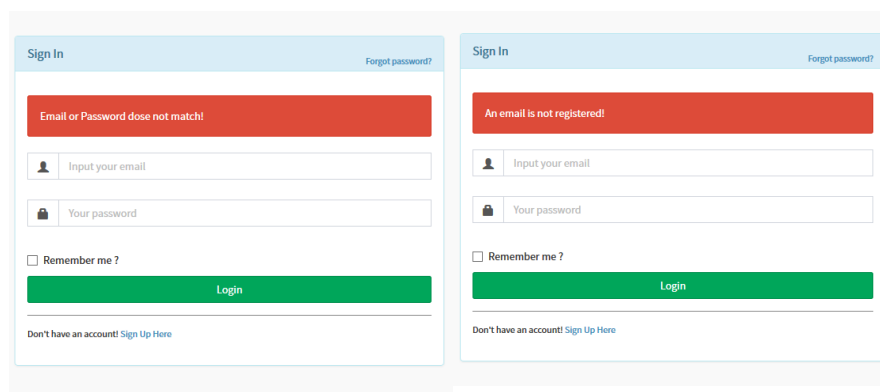
Hình 2.31 – Đăng kí thành viên, thử nhập thông tin sai định dạng



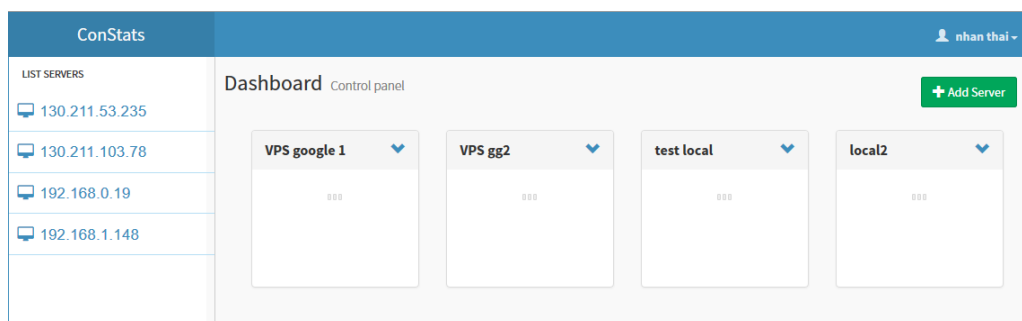
Hình 2.33 - Đăng kí thành công, và giao diện đăng nhập

II.1.1. Chức năng đăng nhập

Để sử dụng hệ thống, sau khi đăng kí, người dùng cần phải đăng nhập vào hệ thống, nếu không đăng nhập, hệ thống sẽ tự động chuyển trang về trang đăng nhập.



Hình 2.35 – Đăng nhập sai thông tin



Hình 2.37 – Trang chủ quản lý hiện danh sách server

II.1.1. Chức năng thay đổi thông tin người dùng

Dành cho người dùng cần sử đổi thông tin cá nhân, mật khẩu,..

Form titled "Hello nhan!". It contains a "G-Avatar" section with a blue circular avatar containing the letter 'n' and a link to Gravatar. Below this are input fields for "First Name" (nhan), "Last Name" (thai), "Email" (thanhnhhan2tn@gmail.com), "Password" (with a hint to keep it blank), "Phone", and "Number of server(s)" (4). A "Save" button is at the bottom left.

Hình 2.39 – Giao diện thay đổi thông tin người dùng

II.2. Chức năng quản lý và cấu hình dịch vụ trên server

II.2.1. Chức năng AddServer

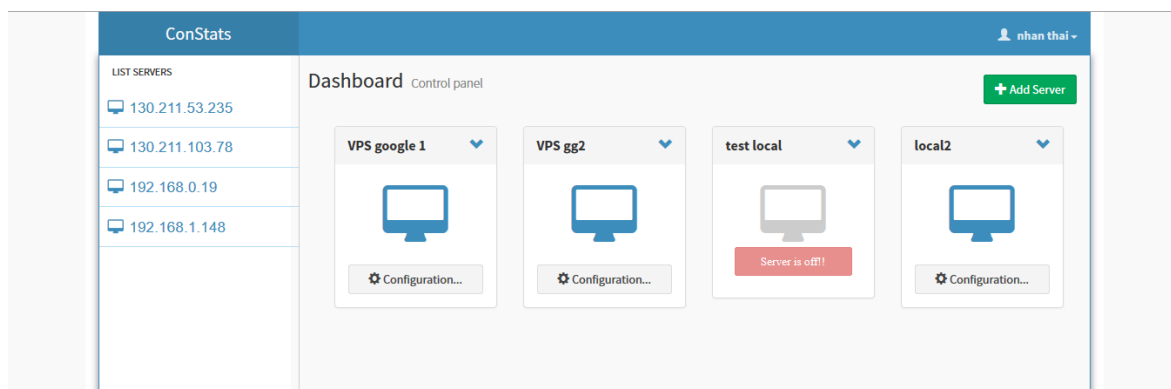
Người dùng thêm server để quản lý dịch vụ và cài đặt dịch vụ.

Dialog box titled "Add new server". It contains input fields for "Server Name" (Tên Server), "IP" (192.168.0.25), "SSH Port" (22), and "Password for 'svcontrol'" (masked with dots). "Close" and "Save" buttons are at the bottom right.

Hình 2.40 - Nhập thông tin để thêm Server

Terminal window titled "Copy command to your Server". It contains the following text:
You should Enable Root login to server with SSH `sudo apt-get install nano && sudo nano /etc/ssh/sshd_config`
Find "PermitRootLogin without-password." change to "PermitRootLogin yes"
Find "PasswordAuthentication no" change to "PasswordAuthentication yes"
`sudo service ssh restart`
`sudo apt-get update && sudo apt-get install whois -y && sudo useradd svcontrol -p $(mkpasswd -m SHA-512 12345678)`
An "OK" button is at the bottom left.

Hình 2.41 – Yêu cầu cấu hình trên server để hệ thống hoạt động



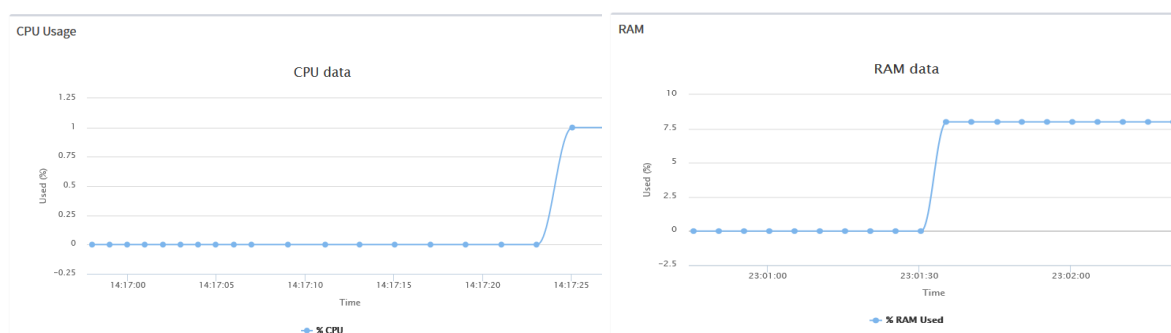
Hình 2.42 – Trạng thái các server vừa thêm

II.2.2. Chức năng hiển thị thông tin server và biểu đồ

- Kiểm thử chức năng hiển thị thông tin, thông tin hiển thị đúng, chính xác.
- Thời gian load thông tin chậm.
- Sơ đồ vẽ chưa chính xác các thông số thay đổi quá nhanh.

System Information	
Hostname:	ubuntu
OS Version:	Ubuntu 14.04.1 LTS
Kernel:	Linux ubuntu 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:08 UTC 2014 x86_64 x86_64 GNU/Linux
Processor Info:	Intel(R) Core(TM) i5-3337U CPU @ 1.80GHz
Uptime:	02:07:18
Memory:	11.1% (222.8/2002.0 MB)
CPU Usage:	<div><div></div></div>
Cpu Loadaverage:	0.00 0.04 0.05 3/227 20717

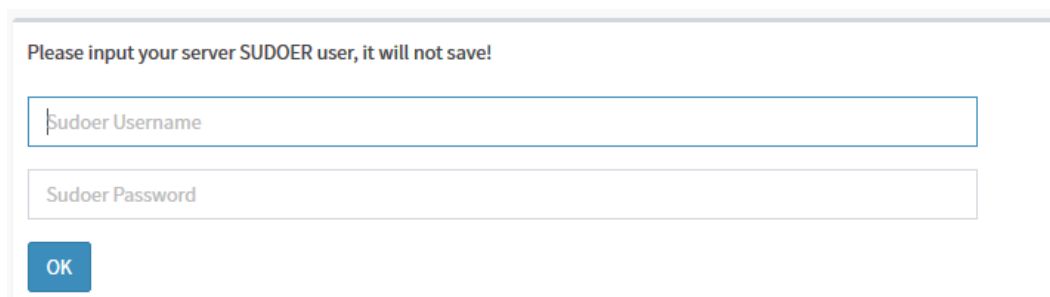
Hình 2.43 – Kết quả hiển thị thông tin server



Hình 2.44 - Kết quả hiển thị biểu đồ

II.2.3. Chức năng điều khiển power

Sau khi vào giao diện Service Control, hệ thống hiển thị form yêu cầu người dùng nhập vào sudoer user để tiến hành quản trị server



Please input your server SUDOER user, it will not save!

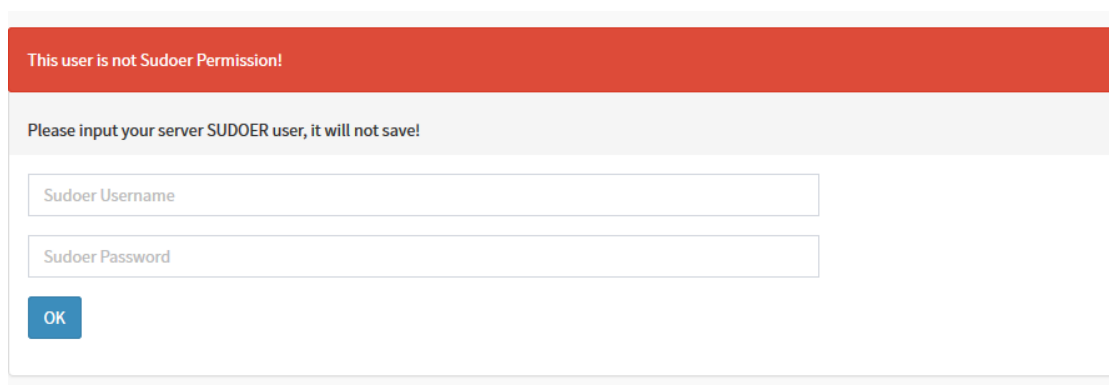
Sudoer Username

Sudoer Password

OK

Hình 2.45 – Màn hình yêu cầu nhập sudoer user

Nếu người dùng nhập user không chính xác hoặc user không có quyền sudoer, hệ thống đưa ra cảnh báo và yêu cầu nhập lại.



This user is not Sudoer Permission!

Please input your server SUDOER user, it will not save!

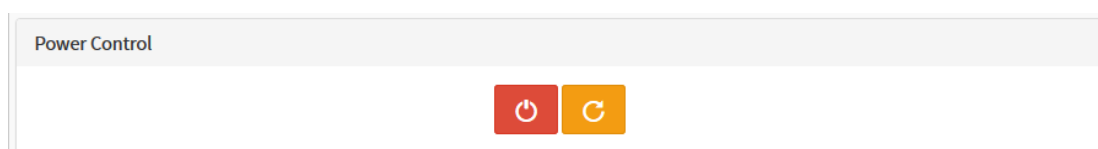
Sudoer Username

Sudoer Password

OK

Hình 2.46 – Nhập không đúng sudoer user

Nếu nhập đúng thông tin sudoer user, giao diện điều khiển sẽ hiển thị

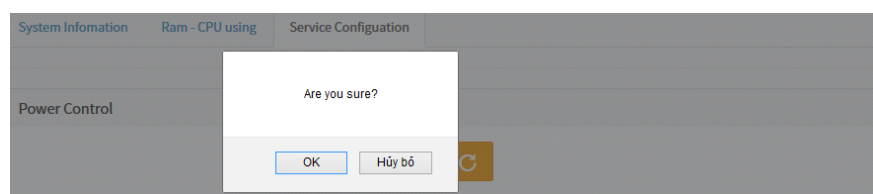


Power Control

Power On (red button with power icon) Power Off (yellow button with refresh icon)

Hình 2.47 - Giao diện điều khiển power

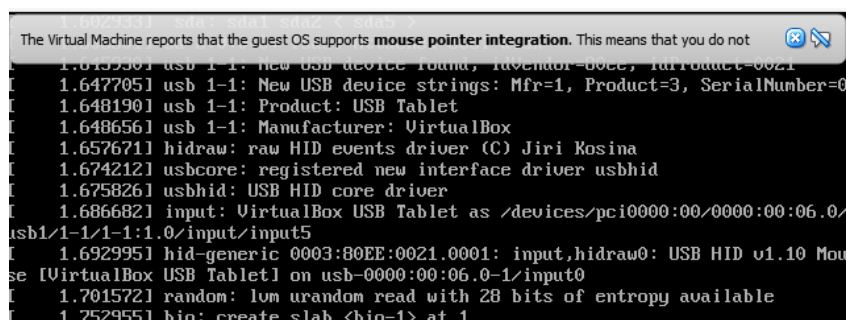
Khi người dùng bấm vào nút lệnh, hộp thoại confirm hiện ra yêu cầu xác nhận lần nữa, nếu người dùng chọn Ok, thì sẽ thực thi tác vụ.



Are you sure?

OK Hủy bỏ

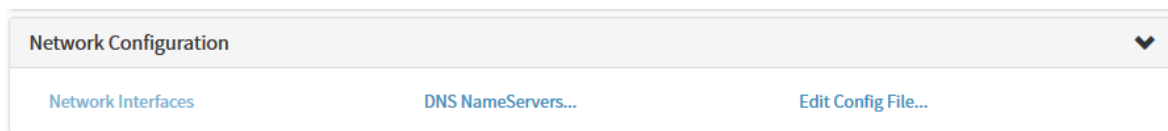
Hình 2.48 – Cửa sổ xác nhận



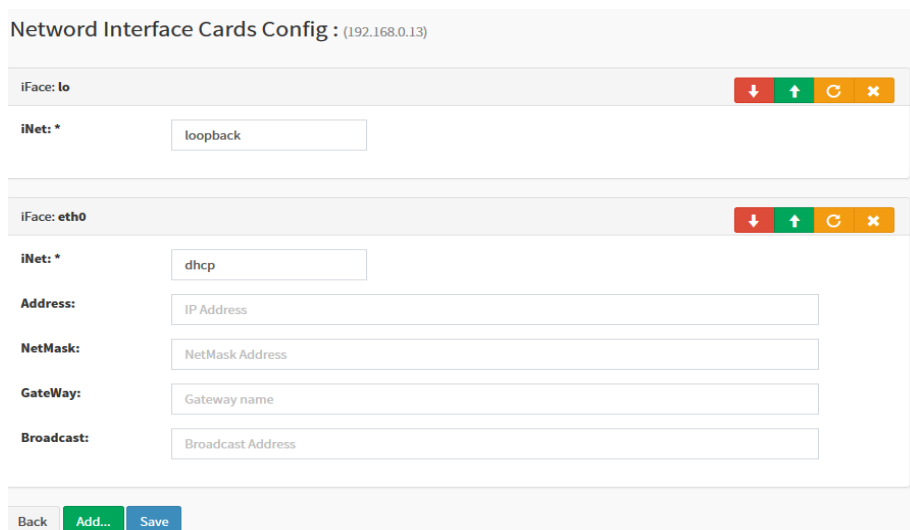
Hình 2.49 – Máy chủ khởi động lại sau khi nhấn nút Reboot

II.2.4. Chức năng cấu hình network card interfaces

Để cấu hình network card interfaces người dùng bấm vào nút Network Configuration. Người dùng nhập các thông tin, lưu lại, hệ thống sẽ cập nhật vào tập tin *interfaces*.



Hình 2.50 – Giao diện cấu hình Network Configuration



Hình 2.51 – Giao diện cấu hình Eth cho network card

Network Interface Cards Config : (192.168.0.13)

DNS NameServer

8.8.8.8

Back Add... Save

Hình 2.52 – Giao diện cấu hình DNS nameserver

Edit Network Interface Config File

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

dns-nameservers 8.8.8.8

# Source interfaces
# Please check /etc/network/interfaces.d before changing this file
# as interfaces may have been defined in /etc/network/interfaces.d
source /etc/network/interfaces.d/*.cfg
```

Back Save

Hình 2.53 – Giao diện cấu hình trực tiếp trên file interfaces

II.2.5. Chức năng cấu hình dịch vụ SSH

Để sử dụng chức năng cấu hình dịch vụ ssh (sshd_config) người dùng truy cập vào nút SSH Server Configuration.

Ssh Service Config : (192.168.0.13)

Port: * 22

Listen Address: * Input Listen Address

Login Grace Time: * Input Login Grace Time

Permit Root Login: * Yes ☒ No ☐

Password Authentication: * Enable ☒ Disable ☐

Permit empty password: * Enable ☐ Disable ☒

Allow User:

Back Save

Edit Config file... >

Logs file... >

Hình 2.54 – Cấu hình SSH

Ssh Service Config : (192.168.0.13)

Cập nhật thành công! (Update Success!)

Port: *

Listen Address: *

Login Grace Time: *

Permit Root Login: * Yes ☒ No ☐

Password Authentication: * Enable ☒ Disable ☐

Permit empty password: * Enable ☒ Disable ☐

Allow User:

Hình 2.55 – Lưu thông tin thay đổi

```
GNU nano 2.2.6      File: /etc/ssh/sshd_config
Port 22
PermitRootLogin yes
PermitEmptyPasswords yes
PasswordAuthentication yes
```

Hình 2.56 – Kiểm tra file cấu hình trên server đã thay đổi

Edit Network Interface Config File

Port 22
PermitRootLogin yes
PermitEmptyPasswords yes
PasswordAuthentication yes

Hình 2.57 – Chức năng cấu hình vào file sshd_config

Logs

```
May 11 00:24:00 ubuntu sudo: ubuntu : TTY=unknown ; PWD=/home/ubuntu ; USER=root ; COMMAND=/bin/chmod 771 /home/ubuntu
May 11 00:24:00 ubuntu sudo: pam_unix(sudo:session): session opened for user root by (uid=0)
May 11 00:24:00 ubuntu sudo: pam_unix(sudo:session): session closed for user root
May 11 00:24:00 ubuntu sshd[10793]: Accepted password for svcontrol from 192.168.0.16 port 64504 ssh2
May 11 00:24:01 ubuntu sudo: ubuntu : TTY=unknown ; PWD=/home/ubuntu ; USER=root ; COMMAND=/bin/echo -e var=$(sshd -t
May 11 00:24:01 ubuntu sudo: pam_unix(sudo:session): session opened for user root by (uid=0)
May 11 00:24:01 ubuntu sshd[10799]: Accepted password for ubuntu from 192.168.0.16 port 64505 ssh2
May 11 00:24:01 ubuntu sudo: ubuntu : TTY=unknown ; PWD=/home/ubuntu ; USER=root ; COMMAND=/usr/bin/tail -10 /var/log/
May 11 00:24:01 ubuntu sudo: pam_unix(sudo:session): session opened for user root by (uid=0)
May 11 00:24:01 ubuntu sudo: pam_unix(sudo:session): session closed for user root
```

Errors

Hình 2.58 – Xem logs file của dịch vụ SSH

II.2.6. Chức năng cấu hình dịch vụ FTP

Tại trang chủ cấu hình dịch vụ người dùng truy cập vào nút *FTP Server Configuration*

Tại giao diện nếu Server chưa cài đặt dịch vụ vsftpd, thì giao diện hiển thị nút lệnh để gọi chức năng cài đặt dịch vụ, sau khi cài đặt, hệ thống sẽ trả về trang cấu hình FTP, người dùng có thể thay đổi các thông số mà hệ thống hỗ trợ.

FTP Service Config : (192.168.0.13)

[Click here to Install VSFTPD Service..](#)

Hình 2.59 – Khi server chưa cài dịch vụ vsftpd

Install/Remove service	
Listen: *	Yes <input type="radio"/> No <input type="radio"/>
Anonymous: *	Enable <input type="radio"/> Disable <input type="radio"/>
Local: *	Enable <input type="radio"/> Disable <input type="radio"/>
Write: *	Enable <input type="radio"/> Disable <input type="radio"/>
Anonymous Upload: *	Enable <input type="radio"/> Disable <input type="radio"/>
Anonymous Make Dir Write: *	Enable <input type="radio"/> Disable <input type="radio"/>
Connect from port 20?: *	Yes <input type="radio"/> No <input type="radio"/>
Deny email: *	Enable <input type="radio"/> Disable <input type="radio"/>
Chroot Local user: *	Enable <input type="radio"/> Disable <input type="radio"/>
Chroot List: *	Enable <input type="radio"/> Disable <input type="radio"/>
Chroot List File: *	<input type="text"/>

[Save](#) [Back](#)

Hình 2.60 – Cấu hình thông số dịch vụ FTP

FTP Config File

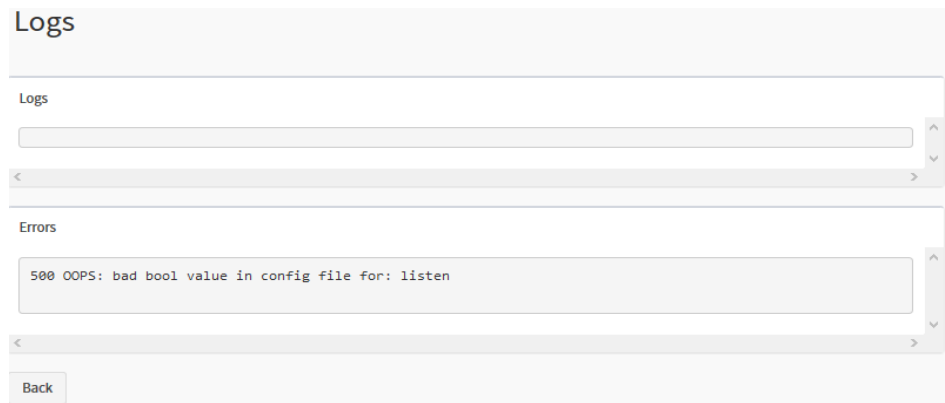
```
listen=YES
anonymous_enable=YES
local_enable=YES
write_enable=YES
anon_upload_enable=YES
anon_mkdir_write_enable=YES
connect_from_port_20=YES
deny_email_enable=YES
chroot_local_user=YES
chroot_list_enable=YES
chroot_list_file=text
```

[Back](#) [Save](#)

Hình 2.61 – Cấu hình từ file vsftpd.conf

```
GNU nano 2.2.6 File: /etc/vsftpd.conf
listen=YES
anonymous_enable=YES
local_enable=YES
write_enable=YES
anon_upload_enable=YES
anon_mkdir_write_enable=YES
connect_from_port_20=YES
deny_email_enable=YES
chroot_local_user=YES
chroot_list_enable=YES
chroot_list_file=text
```

Hình 2.62 – Sau khi cấu hình và kiểm tra thông tin trên server

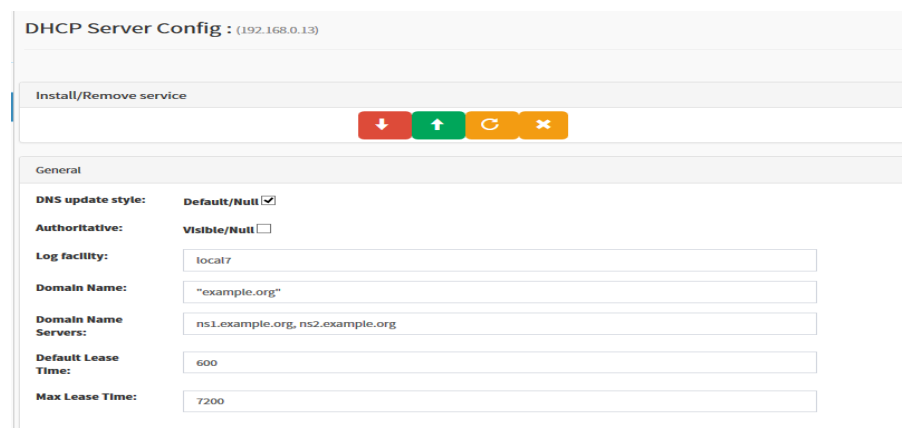


Hình 2.63 – Xem logs hệ thống của dịch vụ vsftpd

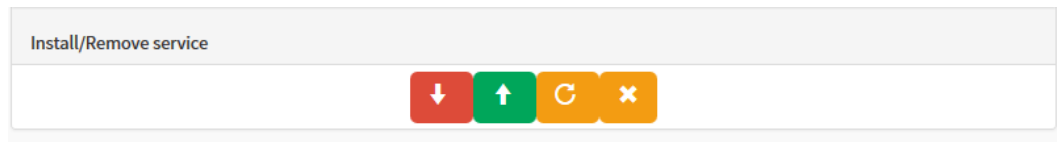
II.2.7. Chức năng cấu hình dịch vụ DHCP

Người dùng bấm vào nút lệnh DHCP Service Configuration để tiến hành cài đặt và cấu hình dịch vụ dhcp.

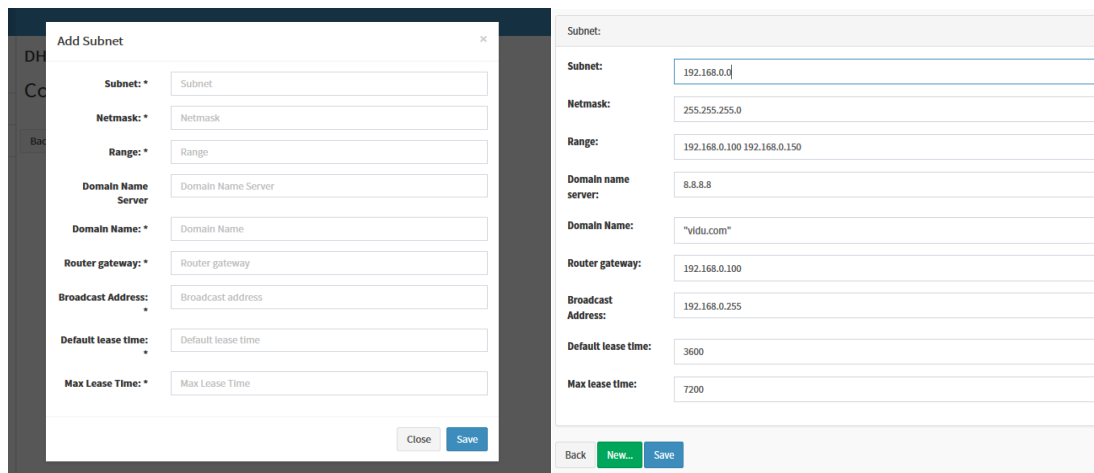
Nếu server chưa được cài đặt gói dịch vụ isc-dhcp-server thì hệ thống hiển thị nút lệnh cài đặt



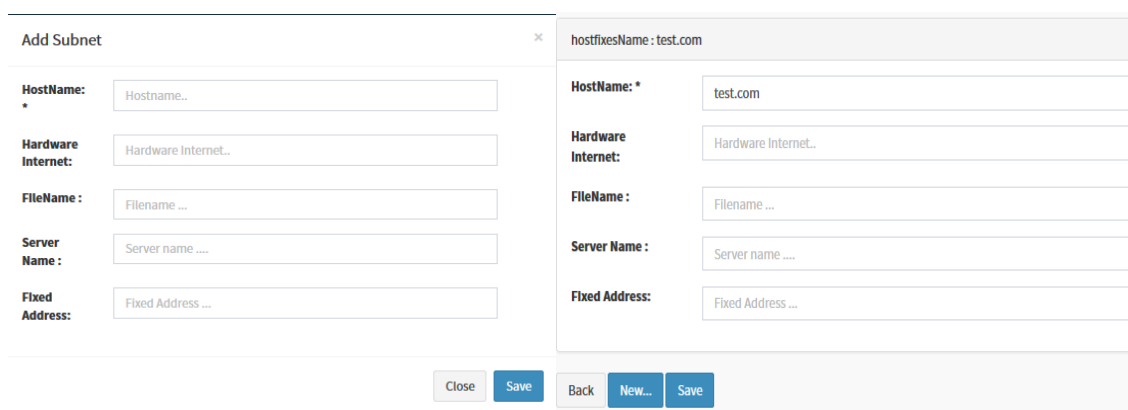
Hình 2.64 – Cấu hình các thông số của dịch vụ DHCP



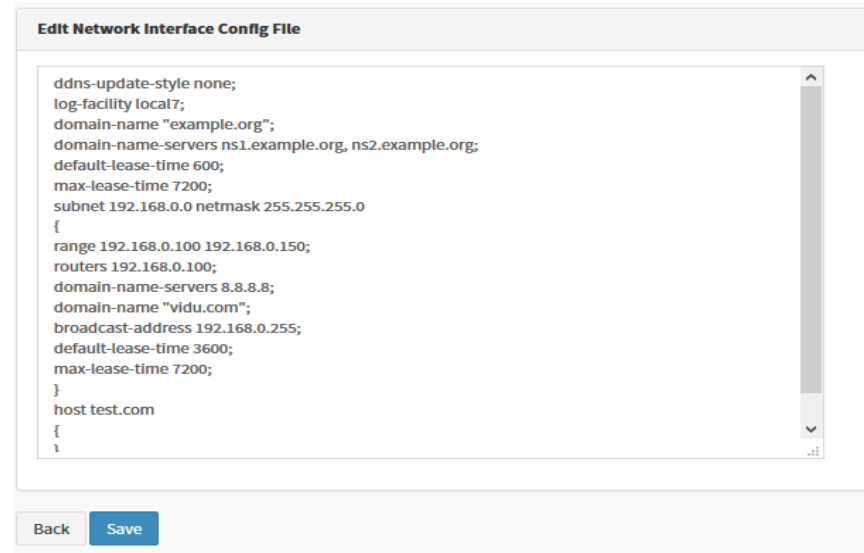
Hình 2.65 - Stop, Start, Restart, remove dịch vụ



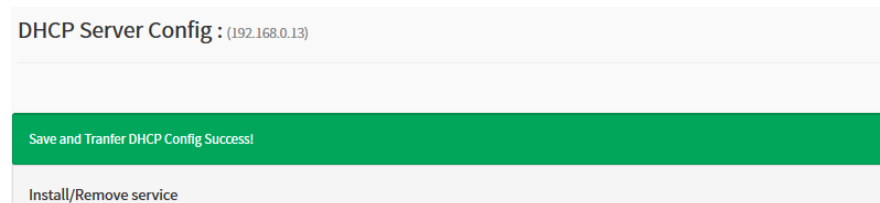
Hình 2.66 – Thêm, sửa subnet



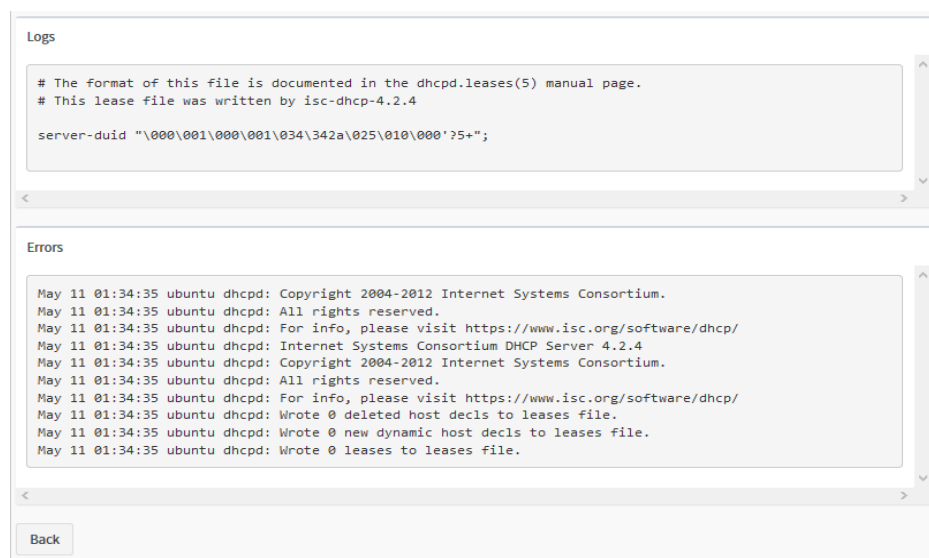
Hình 2.67 – Chức năng thêm, Sửa hostfixs



Hình 2.68 – Chức năng cấu hình trực tiếp bằng file conf



Hình 2.69 - Sau khi lưu thông tin cấu hình

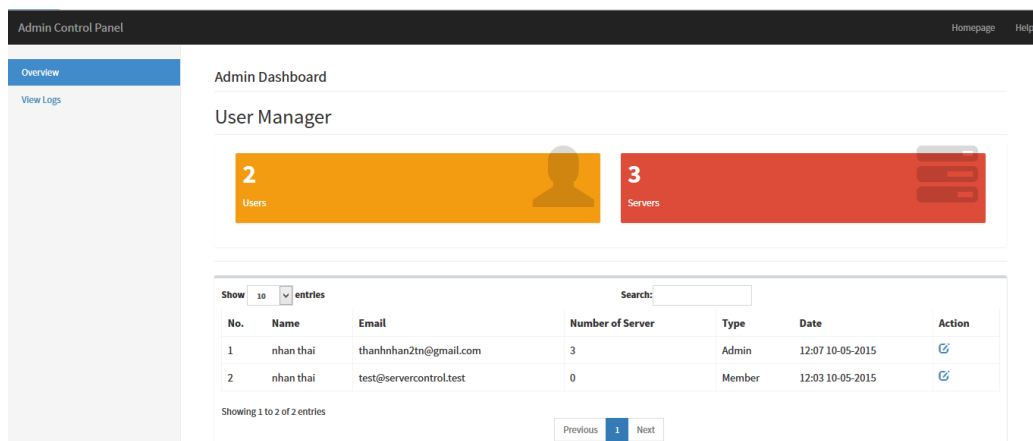


Hình 2.70– Chức năng hiển thị logs và errors

II.3. Chức năng quản lý hệ thống dành cho Admin

II.3.1. Chức năng thống kê số lượng và quản lý thành viên

Admin truy cập vào hệ thống quản lý, nếu như xác nhận đúng quyền admin thì hệ thống hiển thị thông tin thống kê người dùng, và danh sách các người dùng cho phép admin quản lý thông tin.



Hình 2.71– Chức năng thống kê và quản lý người dùng

Muốn sửa đổi thông tin hoặc chuyển quyền sử dụng cho một thành viên, admin nhấn vào biểu tượng edit để sửa thông tin.

User Manager

Edit user

First Name:

Last Name:

Member type:

Email:

Password:

Phone:

Hình 2.72 – Chức năng thay đổi thông tin người dùng

II.3.1. Chức năng hiển thị logs sử dụng hệ thống

Admin chuyển qua chức năng quản lý logs sử dụng của hệ thống bằng liên kết “View Logs” trên sidebar, hệ thống sẽ lấy thông tin logs và hiển thị trên giao diện. Admin có thể tìm kiếm nhanh thông tin, để xem chi tiết các bản ghi logs.

View Logs

Show	10	▼	entries	Search:	
Date.	Logs Content				
01:07 10-05-2015	user thanhnhhan2tn@gmail.com add new Server success!				
01:07 10-05-2015	user thanhnhhan2tn@gmail.com was removed a server!				
01:07 10-05-2015	user thanhnhhan2tn@gmail.com add new Server success!				
08:26 10-05-2015	user thanhnhhan2tn@gmail.com was removed a server!				
08:26 10-05-2015	user thanhnhhan2tn@gmail.com was removed a server!				
08:26 10-05-2015	user thanhnhhan2tn@gmail.com add new Server success!				
12:03 10-05-2015	user thanhnhhan2tn@gmail.com add new Server success!				
12:03 10-05-2015	User register fail with duplicate				
12:03 10-05-2015	User test@servercontrol.test login failed				
12:03 10-05-2015	new user test@servercontrol.test create success!				

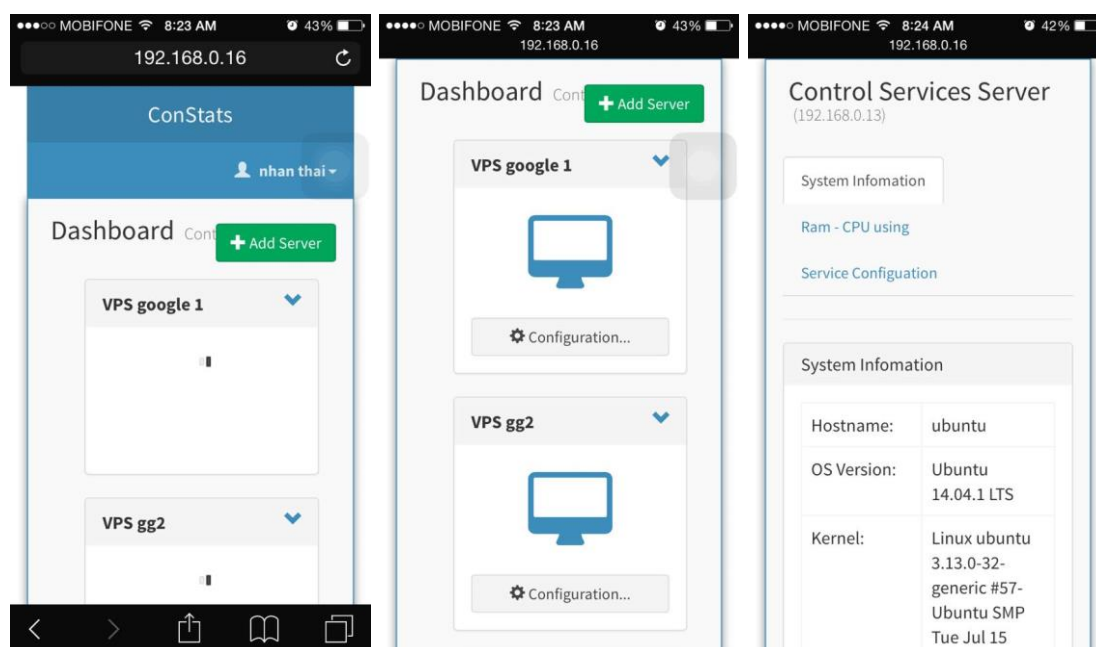
Showing 1 to 10 of 17 entries

Previous 1 2 Next

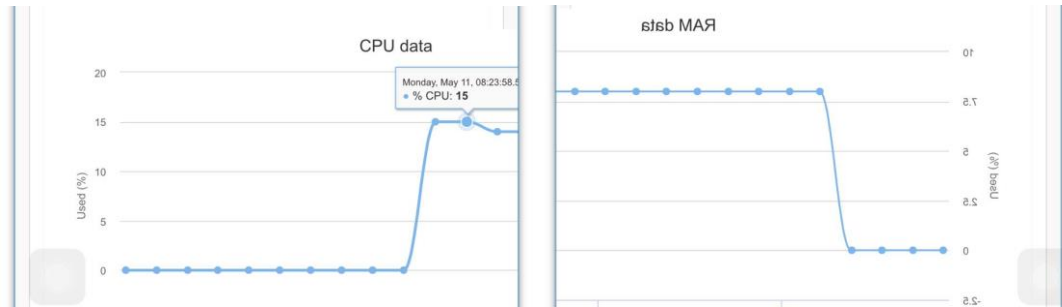
Hình 2.73– Chức năng View Logs

II.4. Kiểm thử giao diện trên di động

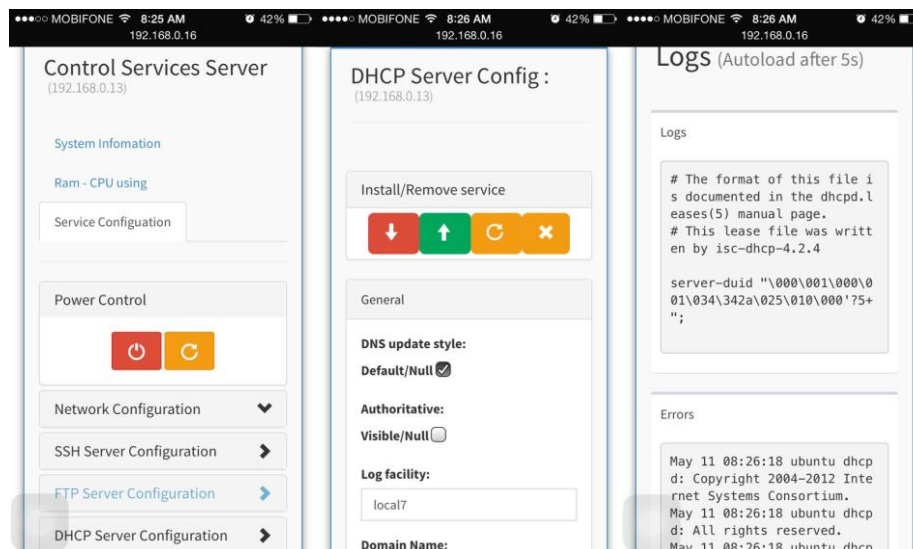
Tiến hành kiểm tra độ tương thích với màn hình điện thoại, hệ thống được kiểm tra bằng điện thoại Iphone 5C, màn hình 4.0”, 640x1136 pixels cho được kết quả như sau:



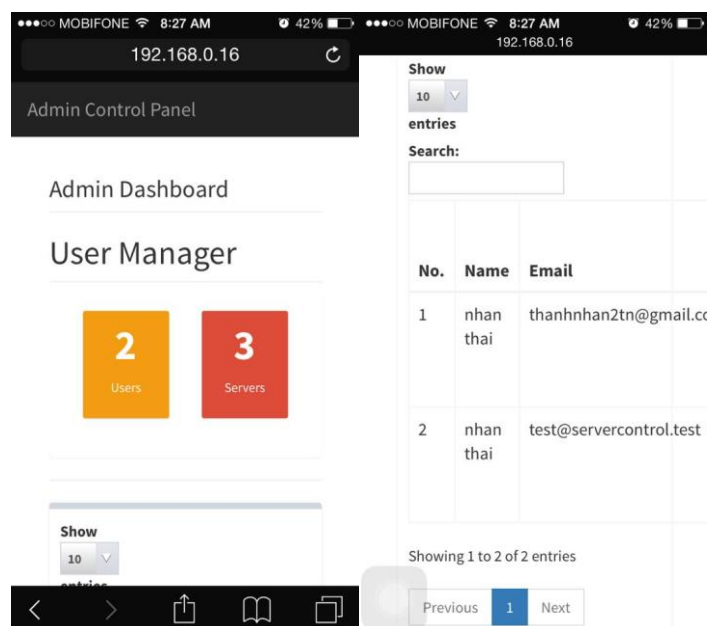
Hình 2.74 – Kiểm tra giao diện chính trên di động



Hình 2.75 – Các bản đồ (chưa hiển thị tốt trên màn hình dọc)



Hình 2.76 – Hiển thị các chức năng cấu hình



Hình 2.77 – Kiểm tra giao diện quản trị của admin

Kết quả trên 80% các chức năng hiển thị tốt trên màn hình của điện thoại.

Kết quả kiểm thử

STT	Chức năng	Kết quả
01	Đăng kí	Hoàn thành
02	Đăng nhập	Chưa hoàn thành (quên mật khẩu)
03	Thay đổi thông tin	Hoàn thành
04	Thêm Server	Hoàn thành
05	Hiển thị thông tin server	Hoàn thành
06	Biểu đồ	Chưa hoàn thành
07	Điều khiển power	Hoàn thành
08	Cấu hình dịch vụ SSH	Hoàn thành
09	Cấu hình dịch vụ FTP	Hoàn thành
10	Cấu hình dịch vụ DHCP	Hoàn thành
11	Thống kê và quản lý người dùng	Hoàn thành
12	Xem logs hoạt động của hệ thống	Hoàn thành
13	Tương thích màn hình thiết bị cầm tay	Hoàn thành

Bảng 2.14: Kết quả kiểm thử

PHẦN 3: KẾT LUẬN

KẾT QUẢ ĐẠT ĐƯỢC VÀ HƯỚNG PHÁT TRIỂN

Kết quả đạt được:

Hệ thống mạng ngày càng rộng lớn, dịch vụ Server càng trở nên đa dạng và số lượng tăng nhanh. Việc quản lý, cấu hình server ngày càng trở nên là vấn đề phức tạp đối với các nhà quản trị. Việc xây dựng được hệ thống hỗ trợ quản lý tiện dụng là vấn đề hết sức cần thiết.

Luận văn này, nghiên cứu tìm hiểu các công nghệ mới và nghiên cứu từ dự án thực tế để rút kết lại những phần cốt lõi mở ra hướng phát triển ứng dụng một cách nhanh chóng và dễ dàng trên các công nghệ mới đó.

Hệ thống nếu được đầu tư và phát triển đến mức toàn diện có thể trở thành hệ thống thực sự cần thiết với các nhà quản trị và tiện dụng đến mức có thể quản trị, cấu hình hay nhận thông báo xử lý lỗi của một hệ thống Server chỉ với thiết bị smartphone.

Những vấn đề đạt được:

- Nắm bắt được phần cốt lõi và triển khai được khung hệ thống.
- Hiểu và áp dụng được giao thức SSH cho cấu hình cũng như lấy thông tin từ server
- Nắm bắt được công cụ git và áp dụng một cách uyển chuyển để quản lý project một cách thuận tiện.

Hướng phát triển:

Hệ thống nếu phát triển một các toàn diện:

- Sẽ tiếp tục phát triển hoàn thiện các dịch vụ để có thể cấu hình nâng cao trên Server.
- Phát triển hệ thống giám sát server thời gian thực và có thể cảnh báo người dùng nếu như server quá tải, xảy ra downtime..
- Hỗ trợ tốt đa người dùng.
- Có thể phát triển thêm ứng dụng kết nối với thiết bị di động, theo dõi, cảnh báo.

HẠN CHẾ

Do phạm vi luận văn có giới hạn, nên kết quả chỉ dừng lại ở mức cơ bản với 2 dịch vụ và 2 chức năng hệ thống và chưa tương thích hoàn toàn với các thiết bị cầm tay.

Chỉ sử dụng các thiết bị trên mô hình với sự hỗ trợ của VirtualBox và một vài dịch vụ Server cho phép dùng thử, chưa thể nắm hết được tất cả các dịch vụ Server trên thị trường.

Kết nối đến Server xử lý còn chậm vì qua nhiều bước xử lý kết nối đến server, kiểm tra bảo mật, và ảnh hưởng bởi tốc độ mạng, trải nghiệm người dùng chưa cao.

Chưa thể tiến hành phân tích và theo dõi các dịch vụ, và server một cách định thời mà chỉ thể hiện khi người dùng vào xem hệ thống.

TÀI LIỆU THAM KHẢO

- [1]. Nguyễn Thành Nhân, 2012, Thiết kế hệ thống tích hợp hiển thị thông tin quản lý thiết bị mạng và dịch vụ mạng. *Module Cài đặt và theo dõi các dịch vụ mạng*.
- [2]. Nguyễn Hoàng Diệu, 2014, Hệ thống quản lý thiết bị mạng theo mô hình client – server.
- [3]. Website Tutorialspoint, *Spring - MVC Framework Tutorial*
http://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm (truy cập ngày 31 tháng 02 năm 2015)
- [4]. Diễn đàn vForum, “Mô hình Spring Web MVC , JSP Servlet” (2011)
<http://vforum.vn/diendan/showthread.php?15878-Mo-hinh-MVC-Java-Web-JSP-Servlet-va-vi-du-demo> (truy cập ngày 31 tháng 02 năm 2015).
- [5]. Nikunjp, Remote SSH: Using JSCH.
<http://nikunjp.wordpress.com/2011/07/30/remote-ssh-using-jsch-with-expect4j/> (truy cập ngày 10 tháng 03 năm 2015).
- [6]. Mr.Ddatnh, Giới thiệu về Spring Framework (12-08-2014)
<http://laptrinh.vn/d/3910-gioi-thieu-ve-spring-framework.html> (truy cập ngày 31 tháng 02 năm 2015).
- [7]. Anh Le, Tổng quan MongoDB (07-2014)
<http://bigsonata.com/mongodb/> (Truy cập tháng 03 năm 2015)
- [8]. The Spring Framework - Reference Documentation, “Chapter 13. Web MVC framework”
<http://docs.spring.io/spring-framework/docs/2.0.8/reference/mvc.html> (truy cập ngày 31 tháng 02 năm 2015).
- [9]. Pankaj, Spring Data MongoDB Example Tutorial (2014)
<http://www.journaldev.com/4144/spring-data-mongodb-example-tutorial>
- [10]. Nguyễn Văn Cường, mongoDB toàn tập,
<http://code.freetuts.net/mongodb> (truy cập tháng 03 năm 2015)
- [11]. Tiles Framework - <http://docs.spring.io/spring/docs/3.2.x/spring-framework-reference/html/view.html#view-tiles> (truy cập tháng 02 năm 2015)
- [12]. HighCharts - <http://www.highcharts.com/> (truy cập tháng 05 năm 2015)

PHỤ LỤC

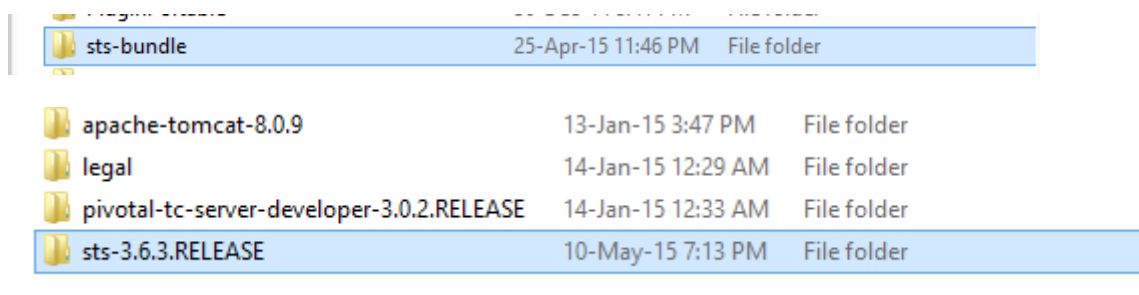
PHỤ LỤC: CÀI ĐẶT VÀ CẤU HÌNH SPRING TOOL SUITE

1. Cài đặt

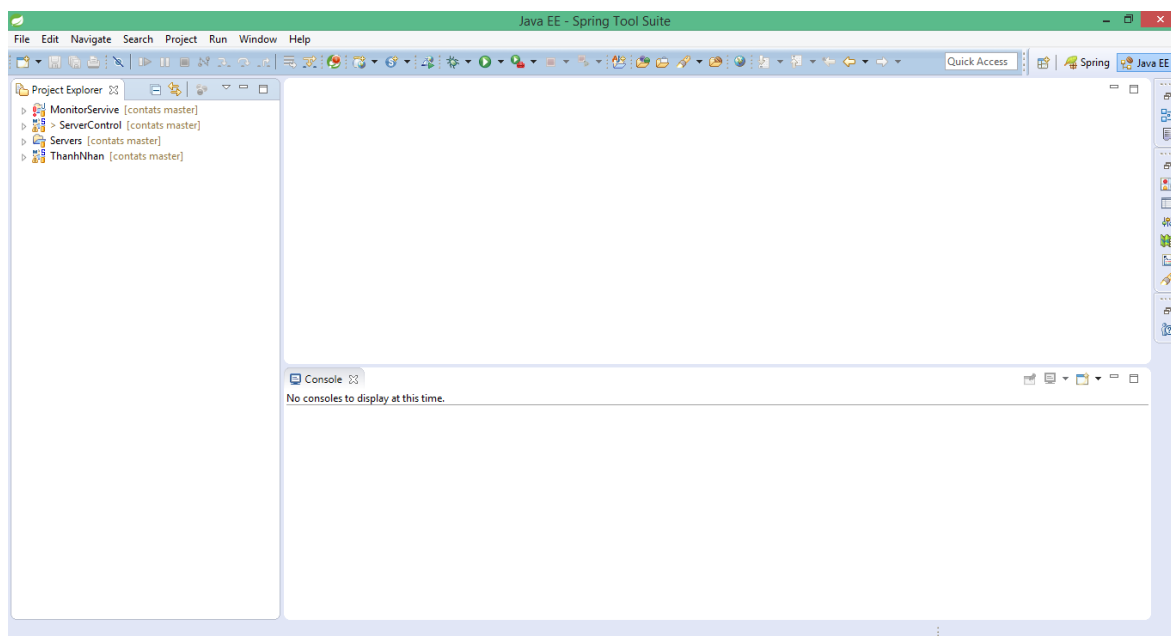
Hướng dẫn cài đặt bộ Spring Tool Suite trên windows.

Tải về gói cài đặt Spring tool suite tại địa chỉ: <https://spring.io/tools>

Giải nén vào thư mục trên máy tính



Khởi chạy tập tin STS.exe,



Giao diện chính của Spring Tool Suite (giống với eclipse)

Cấu hình đường dẫn JDK và JRE của hệ thống tương tự như với Eclipse, hiện cách thức cấu hình đã nói rất nhiều trong các tài liệu, vì vậy xin không nhắc đến ở tài liệu này.

PHỤ LỤC: CÀI ĐẶT VÀ CẤU HÌNH MONGO DATABASE

1. Cài đặt

Tải về phiên bản mongodb tương thích ở địa chỉ:

<http://www.mongodb.org/downloads>

Sau khi tải về, giải nén ra thư mục. vào thư mục mongodb tạo 2 thư mục data và logs để lưu dữ liệu và log. Cấu trúc thư mục mongodb:

Name	Date modified	Type
bin	26/08/2014 10:00 ...	File folder
data	04/10/2014 5:33 PM	File folder
logs	29/10/2014 7:08 PM	File folder
GNU-AGPL-3.0	08/08/2014 8:30 PM	0 File
install.bat	29/10/2014 10:29 ...	Windows Batch File
README	08/08/2014 8:30 PM	File
THIRD-PARTY-NOTICES	08/08/2014 8:30 PM	File

Tạo file install.bat trong thư mục mongodb với nội dung cấu hình như sau:

```
mongod.exe --install --journal
```

```
--dbpath D:\softs\mongodb-win32-i386-2.6.4\data\db
```

```
--logpath D:\softs\mongodb-win32-i386-2.6.4\logs\mongodb.logs
```

```
--serviceName "serverControl"
```

```
--serviceDescription "ThanhNhan ServerControll"
```

Giải thích tham số trong install.bat:

- *dbpath*: là đường dẫn đến thư mục chứa dữ liệu (data), nên chọn đường dẫn tuyệt đối để tránh phát sinh lỗi.

- *logpath*: là đường dẫn đến thư mục chứa log.

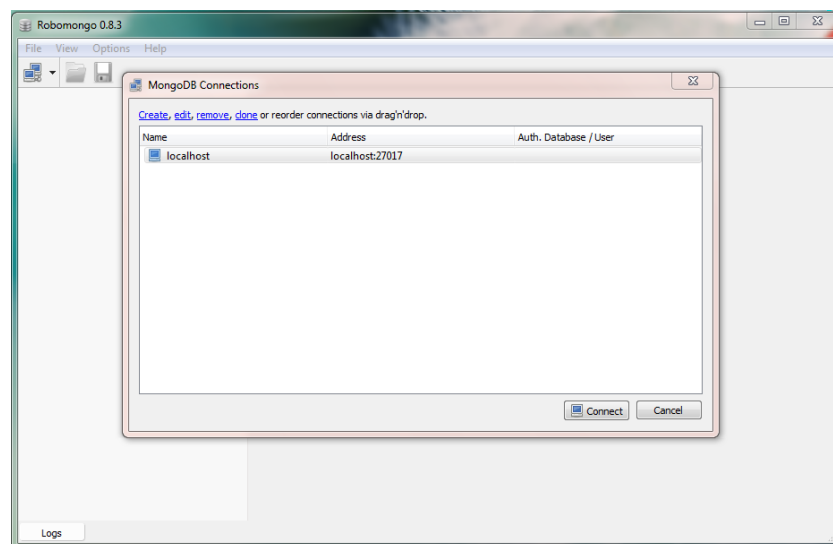
- *serviceName*: là tên service sẽ được hệ điều hành quản lý khi cài đặt thành công thành service.

- *serviceDescription*: là phần thông tin chi tiết của service.

Sau khi tạo file install.bat, mở cmd tại thư mục bin của mongodb và thực thi install.bat.

2. Truy vấn

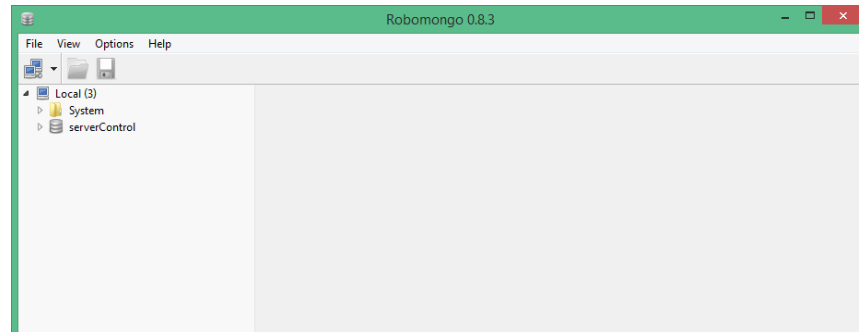
Giống như CSDL quan hệ, MongoDB cũng hỗ trợ truy vấn với các câu điều kiện nhưng phức tạp. Robomongo là một công cụ cho phép thiết lập kết nối và thực hiện các truy vấn cũng như hiển thị kết quả với các câu truy vấn MongoDB, hình dưới là giao diện tổng quát của Robomongo.



Giao diện tổng quát Robomongo

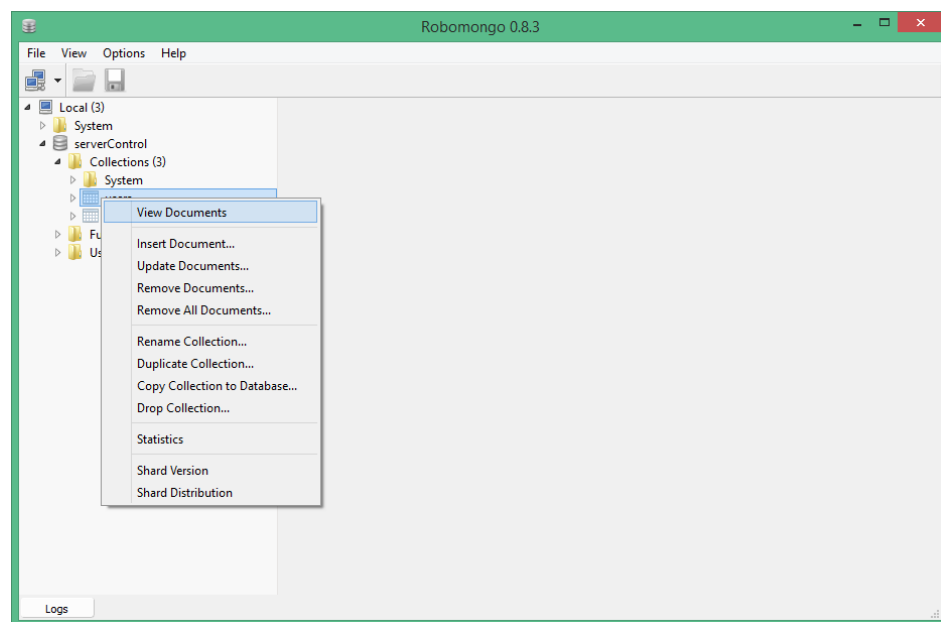
Để tạo kết nối đến MongoDB cần tạo mới 1 connection với 1 số thông tin như sau:

- Chọn Create, 1 popup hiện lên và điền các thông tin cần thiết
- Address: địa chỉ ip server MongoDB
- Port: mặc định là 27017
- Nếu database có cần xác thực thì nhập thông tin xác thực và chọn database, sau đó thực hiện kết nối. Hình 2.14 là giao diện Robomongo khi kết nối thành công đến MongoDB.



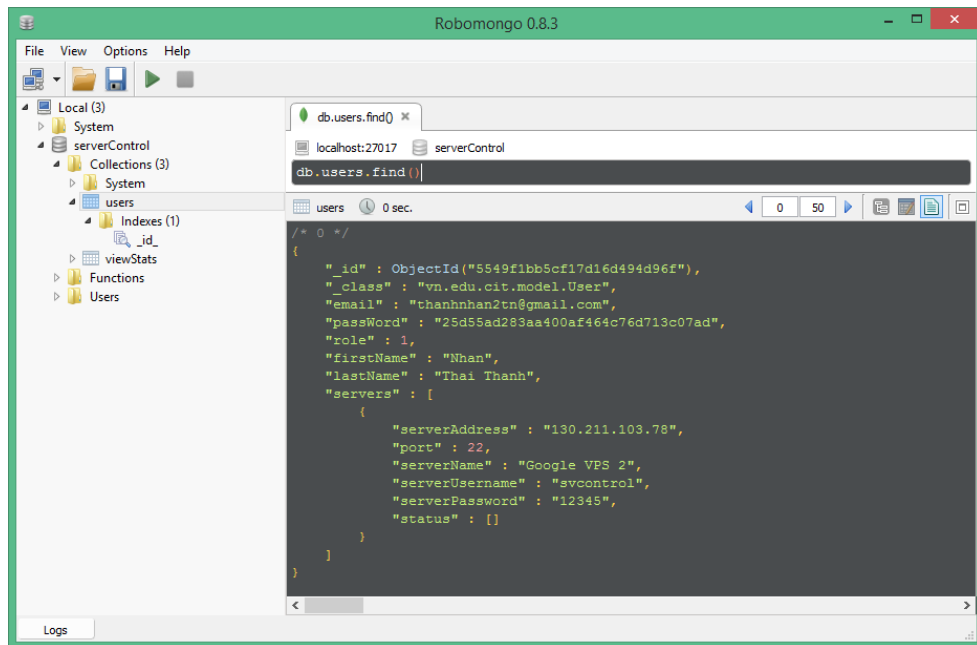
Giao diện kết nối thành công MongoDB

Để hiển thị toàn bộ dữ liệu của một document (table) của một database, chọn vào document và chọn “View Documents”.



Hiển thị document trong Robomongo

Sau khi hiển thị dữ liệu của 1 document nào đó, ta có thể thực hiện các câu truy vấn như CSDL quan hệ. Các câu truy vấn cần nhập vào khung truy vấn và nhấn f5 để hiển thị kết quả.



Truy vấn trong Robomongo

Sau đây là phần so sánh giữa các câu lệnh truy vấn cơ bản của CSDL quan hệ (mysql) và MongoDB.

SQL Statement	Mongo Statement
CREATE TABLE USERS (a Number, b Number)	implicit; can also be done explicitly with <code>db.createCollection("mycoll")</code>
ALTER TABLE users ADD ...	implicit
INSERT INTO USERS VALUES(3,5)	<code>db.users.insert({a:3,b:5})</code>
SELECT a,b FROM users	<code>db.users.find({}, {a:1,b:1})</code>
SELECT * FROM users	<code>db.users.find()</code>
SELECT * FROM users WHERE age=33	<code>db.users.find({age:33})</code>
SELECT a,b FROM users WHERE age=33	<code>db.users.find({age:33}, {a:1,b:1})</code>
SELECT * FROM users WHERE age=33 ORDER BY name	<code>db.users.find({age:33}).sort({name:1})</code>

<code>SELECT * FROM users WHERE age>33</code>	<code>db.users.find({age:{\$gt:33}})</code>
<code>SELECT * FROM users WHERE age!=33</code>	<code>db.users.findt.find({age:{\$ne:4}})</code>
<code>SELECT * FROM users WHERE name LIKE "%Joe%"</code>	<code>db.users.find({name:/Joe/})</code>
<code>SELECT * FROM users WHERE name LIKE "Joe%"</code>	<code>db.users.find({name:/^Joe/})</code>
<code>SELECT * FROM users WHERE age>33 AND age<=40</code>	<code>db.users.find({'age':{\$gt:33,\$lte:40}})</code>
<code>SELECT * FROM users ORDER BY name DESC</code>	<code>db.users.find().sort({name:-1})</code>
<code>SELECT * FROM users WHERE a=1 and b='q'</code>	<code>db.users.find({a:1,b:'q'})</code>
<code>SELECT * FROM users LIMIT 10 SKIP 20</code>	<code>db.users.find().limit(10).skip(20)</code>
<code>SELECT * FROM users WHERE a=1 or b=2</code>	<code>db.users.find({ \$or : [{ a : 1 } , { b : 2 }] })</code>
<code>SELECT * FROM users LIMIT 1</code>	<code>db.users.findOne()</code>
<code>SELECT DISTINCT last_name FROM users</code>	<code>db.users.distinct('last_name')</code>
<code>SELECT COUNT(*y) FROM users</code>	<code>db.users.count()</code>
<code>SELECT COUNT(*y) FROM users where AGE > 30</code>	<code>db.users.find({age: {'\$gt': 30}}).count()</code>
<code>SELECT COUNT(AGE) from users</code>	<code>db.users.find({age: {'\$exists': true}}).count()</code>

Một số so sánh giữa lệnh truy vấn của MySQL và MongoDB