

CUỘC THI SÁNG TẠO THANH THIẾU NIÊN
NHI ĐỒNG TOÀN QUỐC 2016 - 2017

HỆ THỐNG SMARTBOTS

Giải pháp nhà thông minh cho mọi ngôi nhà

MỤC LỤC

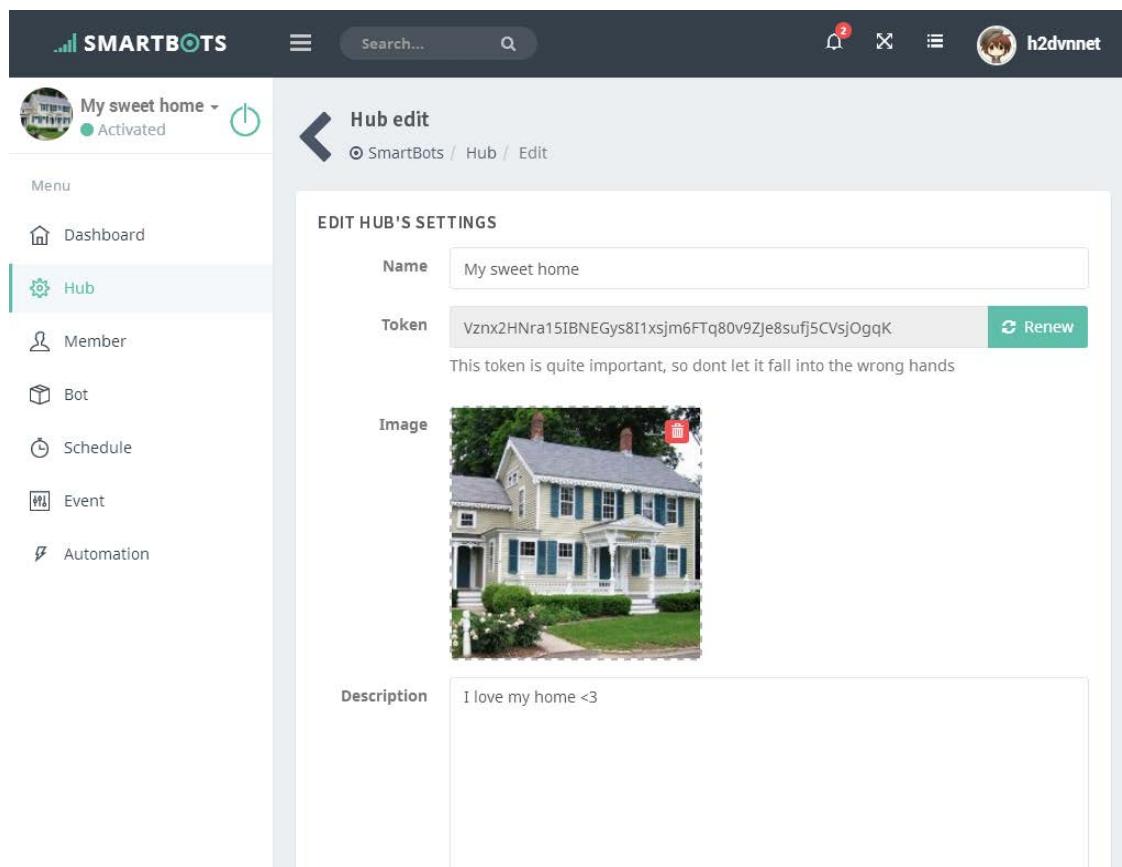
A. Giới thiệu ý tưởng.....	4
B. Mô tả dự án.....	5
C. Các thành phần – chức năng chính.....	7
1. Chức năng người dùng	7
2. Các chức năng cơ bản	8
3. Chức năng điều khiển, theo dõi tình trạng các bot	9
4. Chức năng điều khiển nhanh	10
5. Chức năng lập lịch tự động theo thời gian	11
6. Chức năng điều khiển theo kịch bản/điều kiện/sự kiện	12
7. Chức năng quản lý, phân quyền thành viên	13
8. Chức năng thông báo.....	14
9. Chức năng biểu đồ, biểu đồ thời gian thực.....	15
10. Chức năng theo dõi lịch sử hoạt động	15
D. Công nghệ - kỹ thuật nền tảng	16
1. Developver Friendly Code.....	16
2. Giao diện	17
3. Sử dụng LESS.....	18
4. Sử dụng jQuery	19
5. Sử dụng mô hình lập trình MVC.....	19
6. Lập trình hướng đối tượng.....	20
7. Sử dụng Npm - Composer – Bower	20
8. Sử dụng Redis.....	21
9. EventSource – Websocket: Socket.io – HTML Notification API.....	22
10. Route: Bộ định tuyến	23
11. Artisan.....	24
12. Model – Eloquent	25
13. Database: Migrate, Seeding.....	26

14. View – Blade	27
15. Authentication – Authorization.....	27
16. Localization – Languague: Translatable	28
17. Middleware: Request Filter	28
18. Job: Queue – Task Schedule	29
19. Event – Listenner.....	29
20. Elixir: Gulp – Browsersync.....	29
21. Testing - PHPUnit.....	30
22. Sử dụng Sentry	31
23. Sử dụng Twilio	31
24. Hiệu năng.....	32
25. Bảo mật.....	33
26. Thư viện PHP sử dụng.....	33
27. Thư viện Javascript sử dụng	34
28. Thư viện NodeJS sử dụng	35
29. Tài liệu tham khảo	35
30.Công cụ hỗ trợ.....	36
E. Hướng dẫn cài đặt phần mềm	37
F. Công nghệ phần cứng	41
1. Linh kiện – Module được sử dụng	44
2. Thông số kỹ thuật	45
3. Nguyên lý hoạt động.....	50
4. Các kỹ thuật được sử dụng	51
5. Thư viện hỗ trợ - tài liệu tham khảo	52
G. Hướng dẫn sử dụng phần mềm.....	53
H. Hướng dẫn cài đặt, sử dụng phần cứng	68
I. Tự đánh giá – kết luận	70
J. Hướng phát triển trong tương lai.....	70

A. Giới thiệu ý tưởng

Khái niệm “Nhà thông minh – Smarthome” dường như đã không còn xa lạ với nhiều người. Chúng ta có thể thấy những căn nhà tự động hóa hoàn toàn từ việc bật tắt đèn, âm nhạc, tự động giặt giũ, điều khiển tivi, rèm cửa,... hay điều chỉnh thiết bị qua tâm trạng của con người, sự thay đổi môi... Thị trường nhà thông minh tại Việt Nam hiện đang rất nhộn nhịp với rất nhiều thương hiệu khác nhau như BKAV, LUMI, GAMMA, Digihome... Nhưng đa số vẫn chưa thực sự phụ hợp với thị trường Việt Nam, khi người dân có mức sống không thực sự cao, còn các giải pháp nhà thông minh hiện tại thì quá đắt đỏ, sở hữu nhiều công nghệ và chức năng tiên tiến nhưng lại không thực sự cần thiết với đa số người dùng. Hơn nữa người dùng còn cảm thấy lo ngại về nhiều vấn đề khác như là quá trình lắp đặt, sử dụng,...

Nắm rõ được những điều đó, nhóm em đã đặt mục tiêu là tạo ra một giải pháp nhà thông minh khác biệt hơn, phù hợp với thị phần người tiêu dùng ở việt nam, chi phí thấp, với khả năng tích hợp, tùy biến cực kỳ cao, dễ dàng lắp đặt, sử dụng, sửa chữa và ngay cả tháo gỡ, đi cùng với đó là hiệu quả, đem lại tiện nghi cho người dùng.



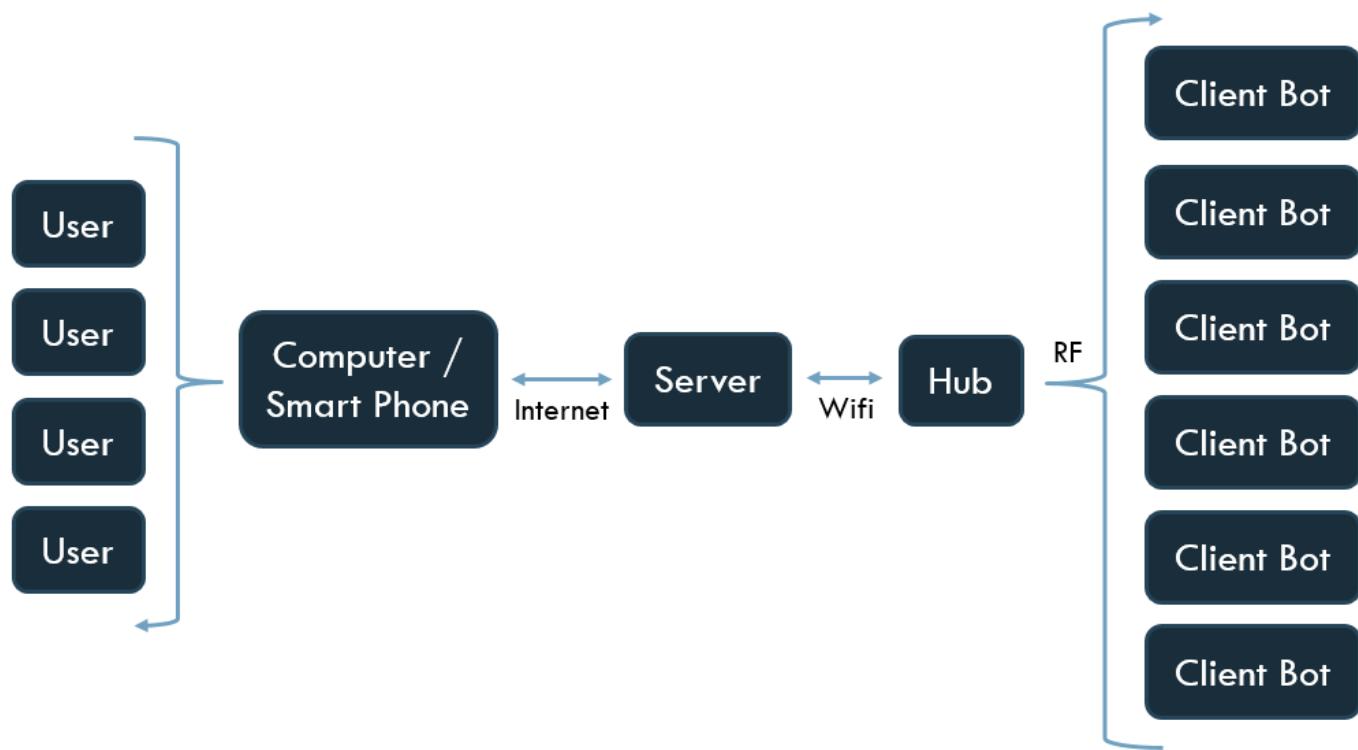
Hình 1. Giao diện chính của phần mềm

B. Mô tả dự án

- Smartbots Là một hệ thống có thể kết nối tất cả thiết bị điện thông thường như bóng đèn, rèm, cổng điện, máy bơm, điều hoà, bình nước nóng, thiết bị nghe nhìn, thiết bị an ninh chống đột nhập, thiết bị kiểm soát vào ra, thiết bị báo cháy,...
- Việc kết nối này sẽ giúp cho các thiết bị điện thông thường có khả năng "giao tiếp" với nhau, phối hợp cùng hoạt động để mang lại rất nhiều lợi ích cho người sử dụng.
- Smartbots gồm nhiều bot kết nối với nhau, mỗi bot với một chức năng riêng biệt, điều khiển từng loại thiết bị khác nhau. Các bot sẽ có vai trò như một ngón tay, được gắn phía trên các công tắc/nút nhấn/nút vặn/ổ cắm... thay cho người dùng điều khiển tất cả các thiết bị điện. Đây chính là điểm tạo nên sự khác biệt của Smartbots với các giải pháp đã xuất hiện trên thị trường, khiến cho Smartbots trở nên dễ lắp đặt, sử dụng và sửa chữa, không cần phải đi dây, đục tường, hay là thay thế các bảng điện, công tắc cũ.
- SmartBots có nhiều module, chúng là:
 - **SmartHub** (gọi tắt là hub) – trung tâm kết nối các bot nhỏ với nhau, kết nối trực tiếp với mạng
 - **FlipBot** – bot dùng riêng cho các loại công tắc
 - **PushBot** – bot dùng cho các loại nút nhấn
 - **TwistBot** – bot dùng cho các loại nút vặn, xoay
 - **PlugBot** – bot dùng cho các loại ổ cắm
 - **SenseBot** – bot dùng để nhận biết thông tin từ môi trường xung quanh: như nhiệt độ, độ ẩm, ánh sáng, âm thanh, đóng mở cửa,... SenseBot có nhiều loại ứng với từng thông tin mà nó nhận biết được, nhưng vẫn gọi chung là SenseBot
 - **IrHub** – là một trung tâm điều khiển các thiết bị có bộ nhận IR, có thể thay thế cho toàn bộ các IR remote.

Những phương thức hoạt động chính của các Bot (trừ SenseBot):

- **Hoạt động từ xa:** Có thể được người dùng điều khiển từ xa thông qua điện thoại, máy tính có kết nối mạng ở bất cứ đâu, bất cứ khi nào
- **Hoạt động bằng tay:** Ngoài cách điều khiển từ xa, người dùng còn có thể điều khiển các bot qua một nút nhấn nằm trên bot
- **Hoạt động tự động:** Nhờ sự hỗ trợ từ phần mềm, các bot có thể hoạt động một cách tự động nhờ việc tạo lập các lịch làm việc, các kịch bản làm việc tự động



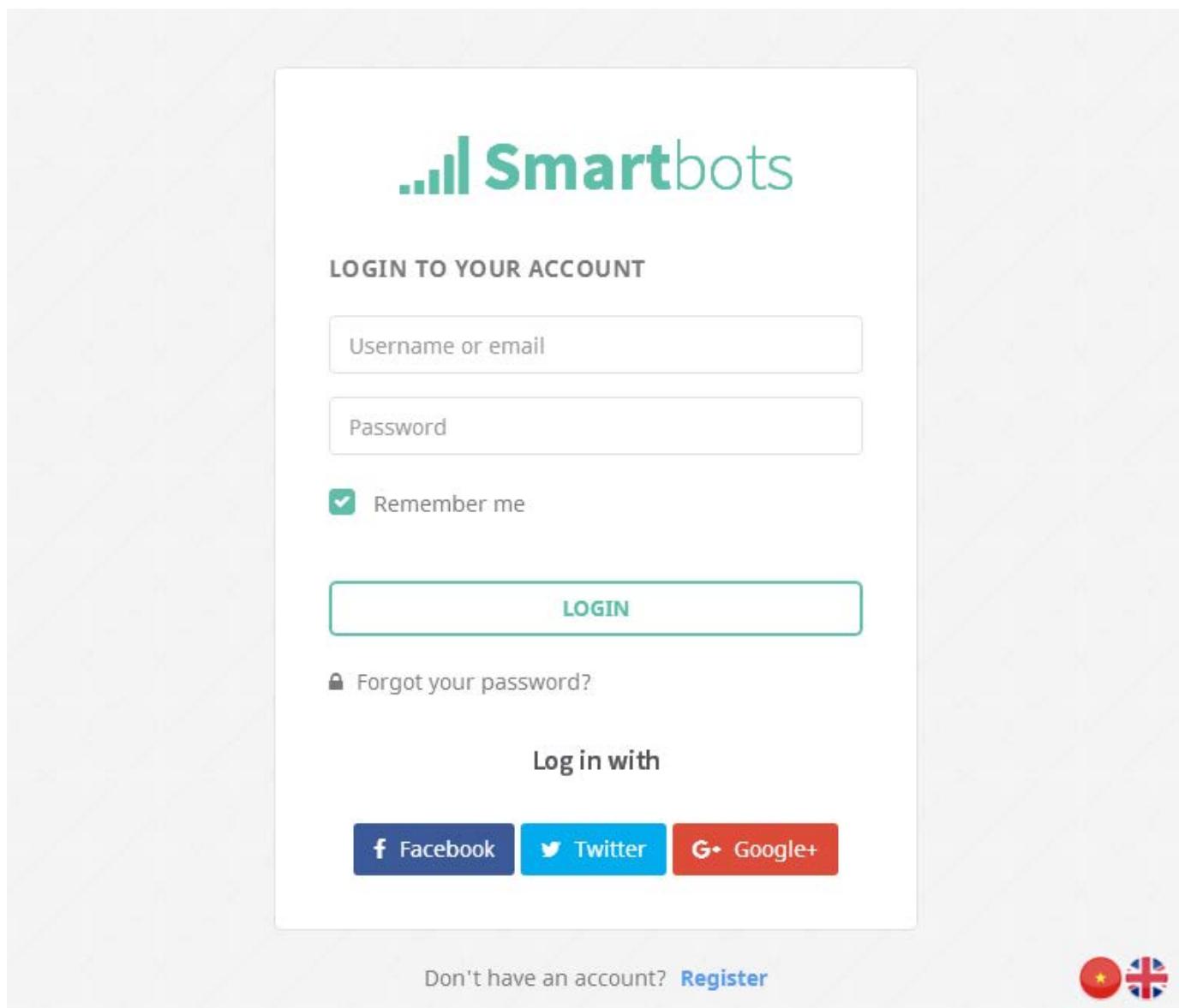
Sơ đồ biểu diễn cách hoạt động đơn giản của SmartBots (sẽ giải thích rõ ở phần “Công nghệ phần cứng”)

C. Các thành phần – chức năng chính

Phần mềm này là phần được chạy trên Server của hệ thống Smartbots, dưới dạng một Website, nên chức năng chính sẽ xoay quanh việc giúp người dùng dễ dàng tương tác và sử dụng Smartbots.

1. Chức năng người dùng

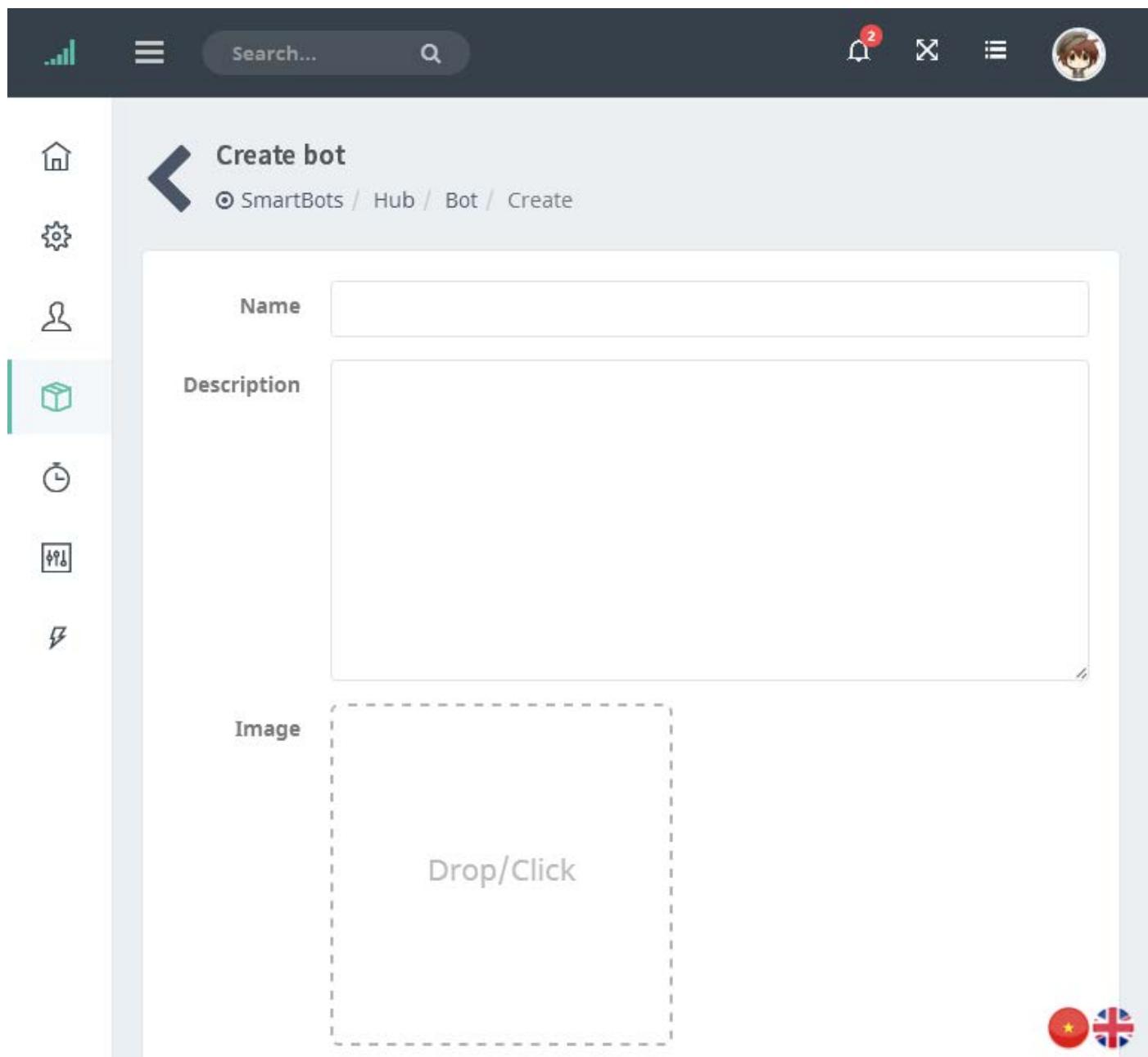
Các chứng năng liên quan đến người dùng như đăng ký, đăng nhập, xem thông tin cá nhân, thay đổi mật khẩu,... mà phần mềm hướng người dùng nào cũng cần phải có.



Hình 2: Chức năng người dùng

2. Các chức năng cơ bản: CRUDDR (Create – Read – Update – Delete – Deactivate – (Re) Activate)

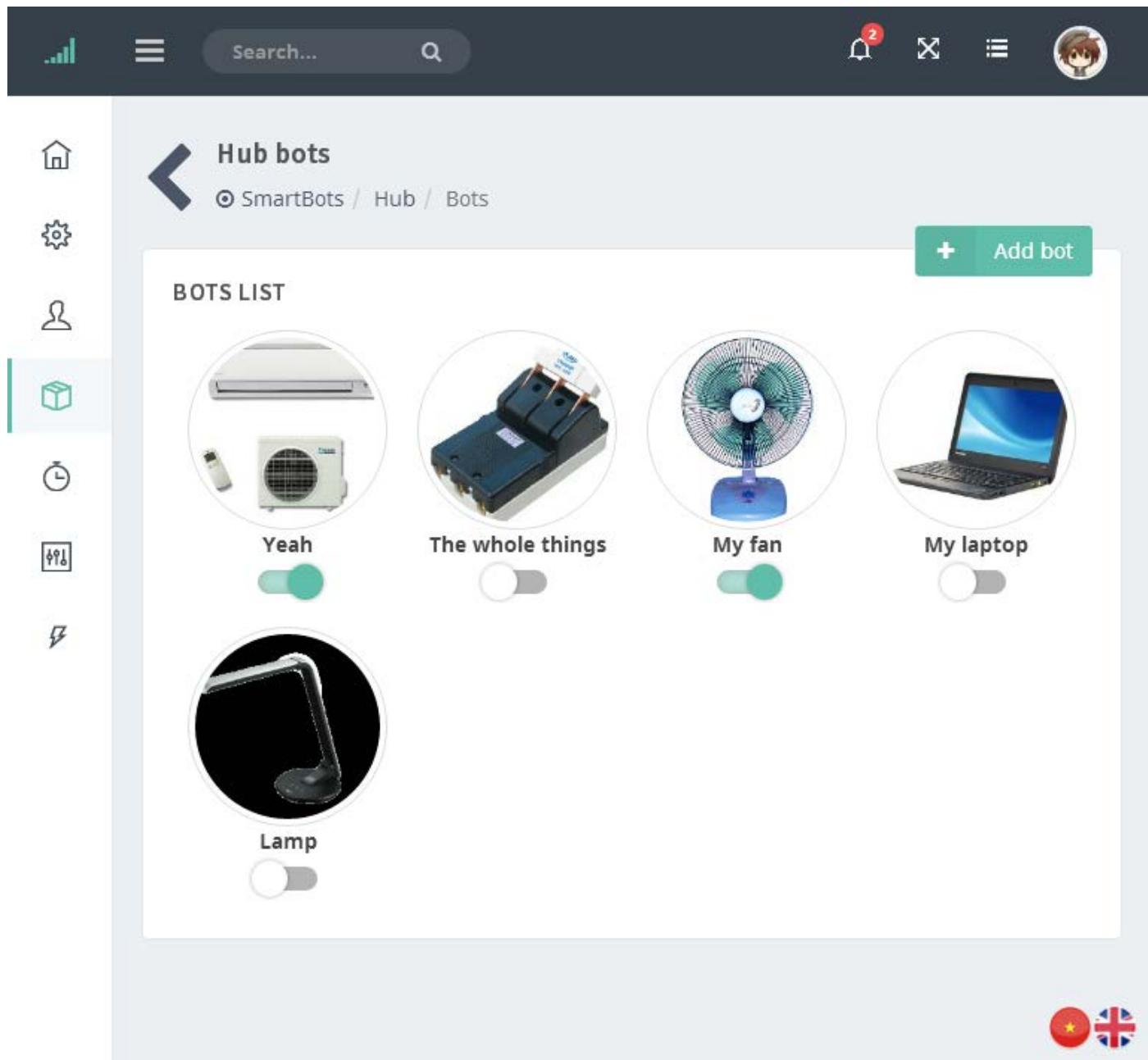
Các chức năng cơ bản như tạo, xem, sửa (cập nhật), xóa, hủy kích hoạt, kích hoạt với các thành phần của SmartBots như Hub, Bot, Schedule, Automation, User,....



Hình 3: Chức năng tạo bot

3. Chức năng điều khiển, theo dõi tình trạng các bot

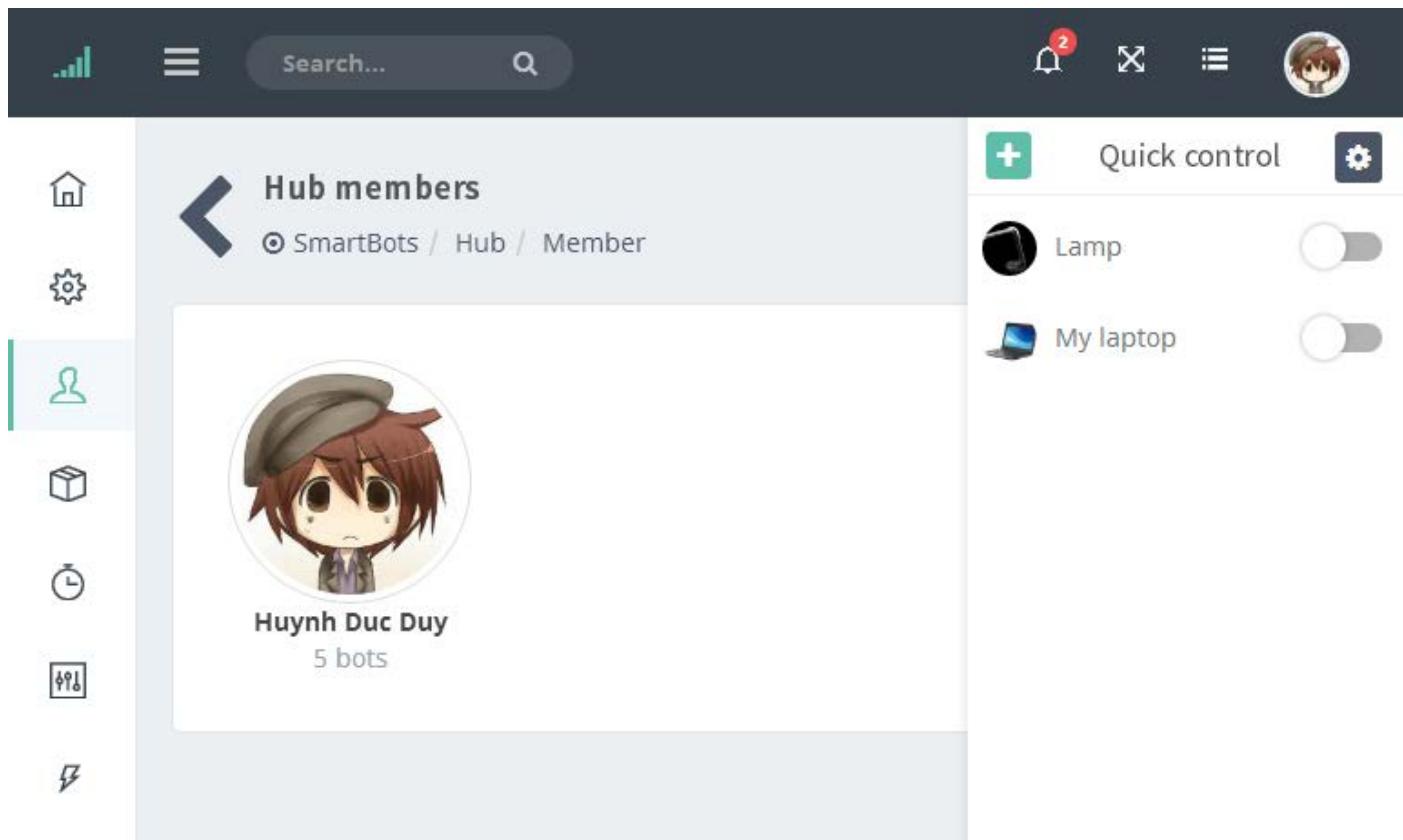
Chức năng chính và cơ bản nhất của phần mềm, đó là giúp người dùng thao tác điều khiển bật/tắt các bot. Ngoài ra còn có thể giúp người dùng theo dõi, quản lý tình trạng của bot, thời gian bật/tắt,....



Hình 4: Chức năng điều khiển, theo dõi bot

4. Chức năng điều khiển nhanh (Quick Control)

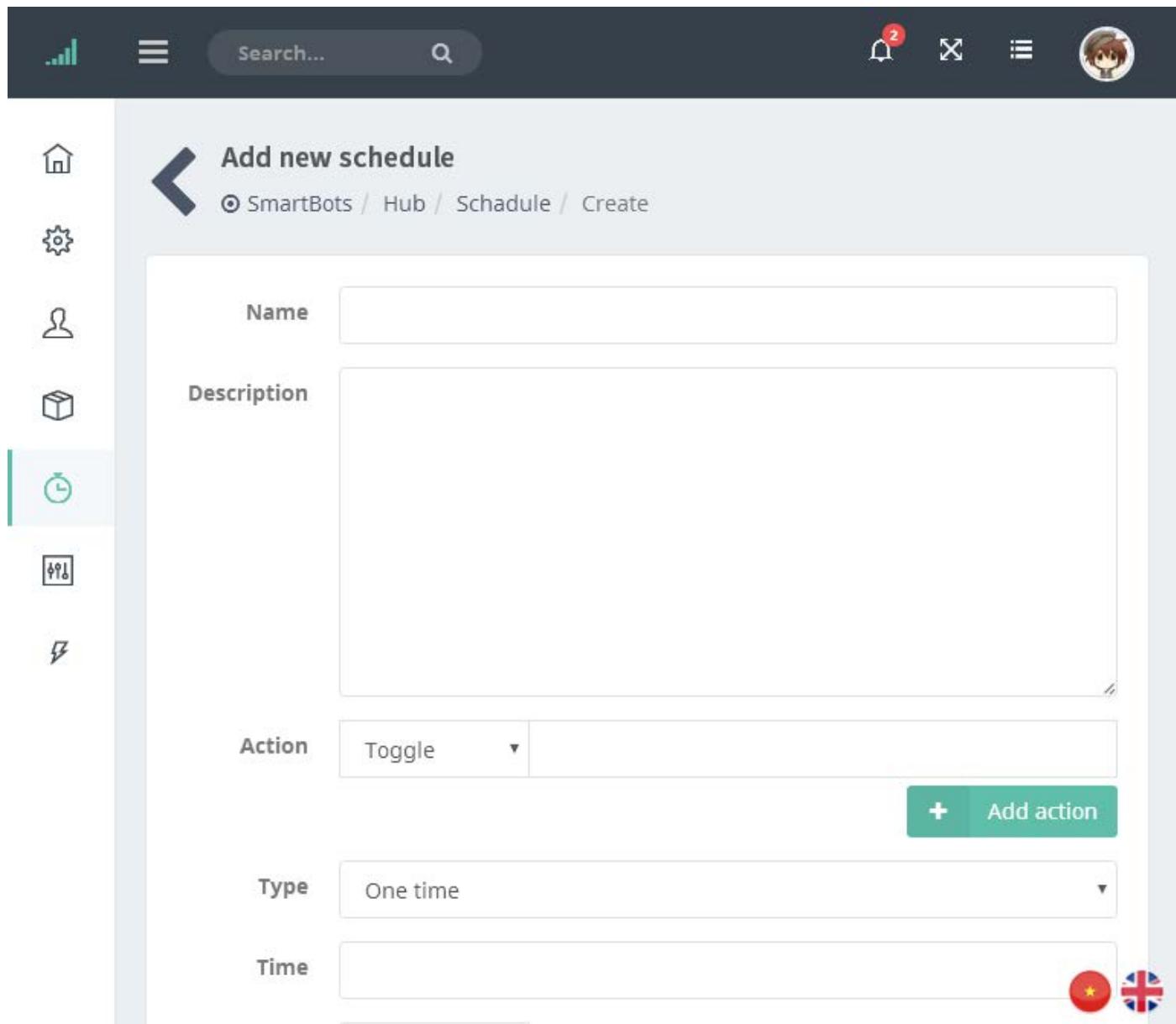
Ngoài cách điều khiển các bot từ danh sách, người dùng có thể thêm một số bot thường được điều khiển vào thanh menu bên phải, từ đó có thể điều khiển các bot từ bất cứ trang nào.



Hình 5: Chức năng điều khiển nhanh (thanh menu bên phải)

5. Chức năng lập lịch tự động theo thời gian (Schedule)

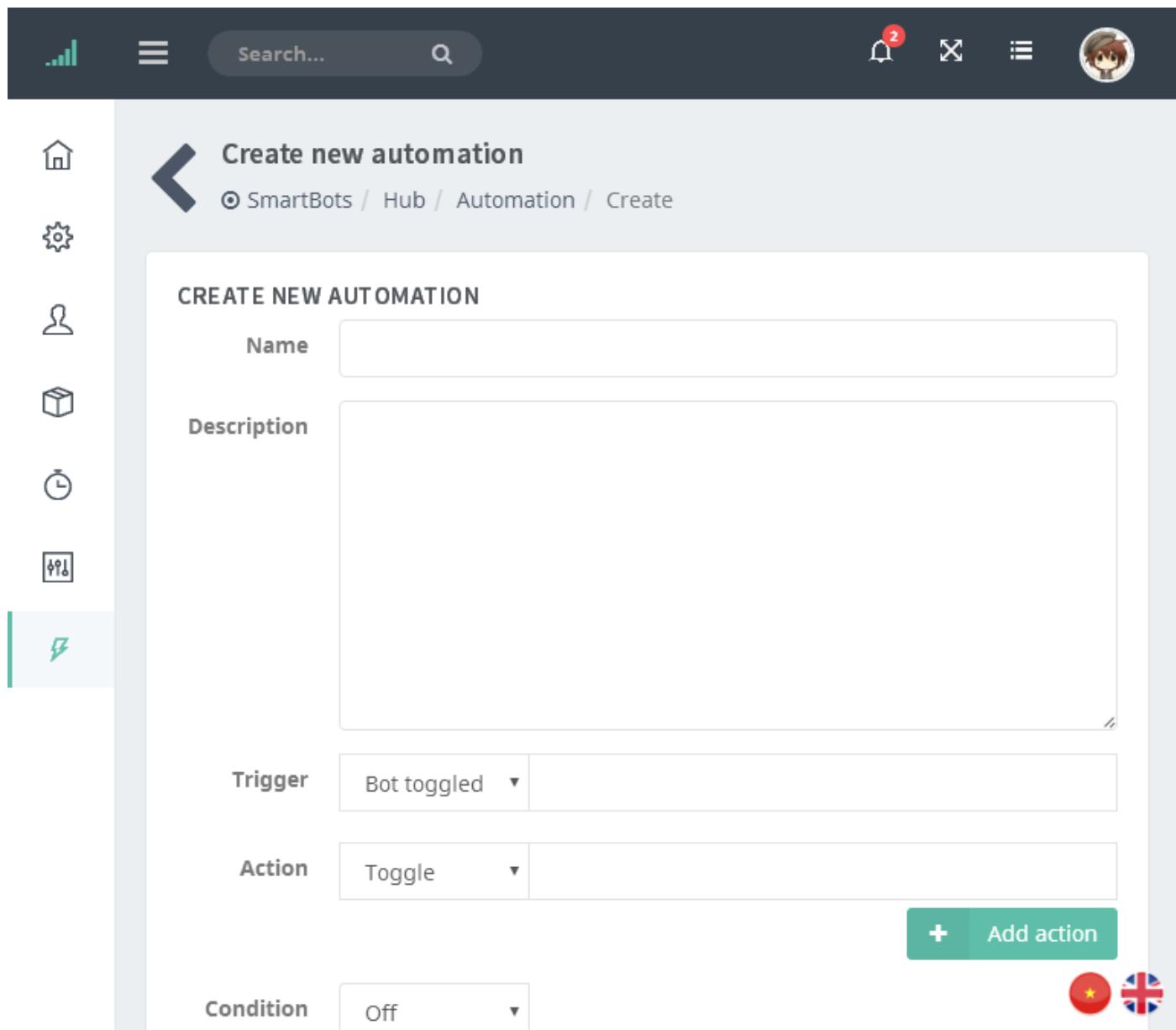
Ngoài cách điều khiển bằng tay, người dùng còn có thể tạo lập một lịch hoạt động sẵn cho các bot, với nhiều lựa chọn như lịch chạy một lần, lịch lặp lại sau mỗi giờ, lặp lại các ngày trong tuần với các điều kiện khác nhau,...



Hình 6: Giao diện tạo lịch mới

6. Chức năng điều khiển theo kịch bản/điều kiện/sự kiện (Automation/Event)

Người dùng còn có thể khiến các bot hoạt động tự động theo kịch bản, dựa trên các sự kiện có thể được kích hoạt (Các sự kiện mặc định hoặc các sự kiện được người dùng tạo ra và kích hoạt thủ công, ví dụ như: nếu <bot> bật thì <bot2> tắt, hay là: nếu <người dùng> về đến nhà thì <bot3> bật,...)



Hình 7: Giao diện tạo kịch bản hoạt động mới

7. Chức năng quản lý, phân quyền thành viên

Trong một ngôi nhà thì chắc chắn có nhiều thành viên, và tất nhiên ai muốn, cũng có quyền điều khiển các bot, nhưng khác nhau: ví dụ như Bob là người bố trong gia đình, và ông không muốn cậu con trai của mình có quyền bật máy nghe nhạc?. Nắm bắt được điều đó, phần mềm tạo ra hệ thống phân quyền cho các thành viên cùng tham gia (cùng sử dụng) các bot.

The screenshot shows the SmartBots app's member management screen. At the top, there are search bars and a user profile icon. Below, two sections labeled "Automation high permissions" and "When event fired" have search fields. A central "↔" icon indicates a relationship between these sections. Below this, a table lists permissions for various categories:

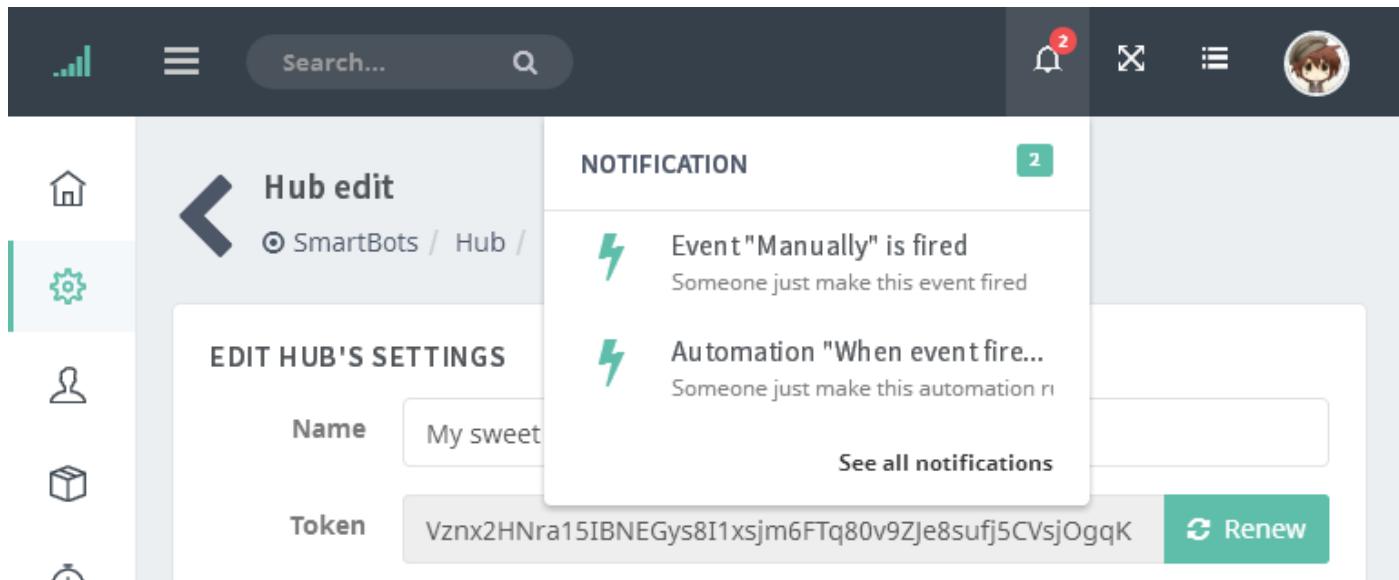
	Add/Create	View/Control	Edit/Delete
Hub	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bots	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Schedules	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Events	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Automations	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Members	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

At the bottom left is a blue "Add" button with a plus sign, and at the bottom right are flags for Vietnam and the United Kingdom.

Hình 8. Khi thêm thành viên, có thể phân quyền ch thành viên

8. Chức năng thông báo

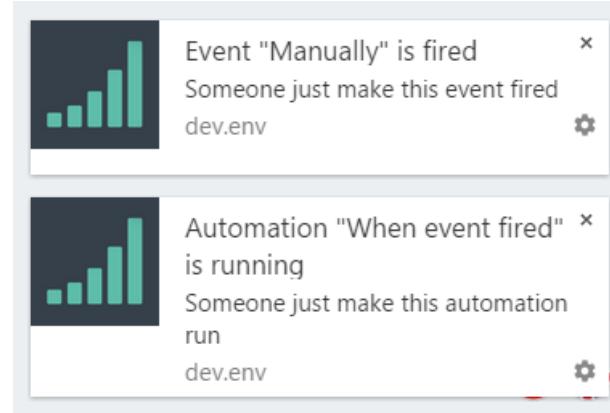
Để tiện lợi hơn cho người sử dụng, smartbots sẽ có khả năng thông báo đến cho người dùng một số sự kiện quan trọng (hay các sự kiện mà người dùng chọn) như việc có thiết bị được bật/tắt, một lịch/kịch bản nào đó vừa được chạy, hay khác hơn là thông báo khi thiết bị nào đó đang bật quá lâu, hoặc sử dụng quá nhiều điện năng,...



Hình 9: Thanh hiển thị thông báo



Hình 10: Popup hiển thị thông báo ở góc dưới trái (giống facebook)



Hình 11: Popup hiện thông báo từ HTML5 Notification API

9. Chức năng biểu đồ, biểu đồ thời gian thực

Ngoài cách quản lý trạng thái thông thường, phần mềm còn cung cấp cho người dùng nhiều biểu đồ và biểu đồ theo thời gian thực về sự hoạt động của các bot, giúp người dùng có cái nhìn trực quan hơn về hoạt động của các bot, hay là thói quen sử dụng các thiết bị điện trong nhà của mình.

* Chức năng đang trong giai đoạn phát triển

10. Chức năng theo dõi lịch sử hoạt động (History)

Trong quá trình hoạt động, phần mềm sẽ ghi lại toàn bộ những hành động của người dùng, và cả các hoạt động, tác vụ mà bot/hub thực hiện giúp người dùng dễ dàng hơn trong việc quản lý, theo dõi.

* Chức năng đang trong giai đoạn phát triển

D. Công nghệ - kỹ thuật nền tảng

Phần mềm được viết dưới dạng một website trực tuyến, sử dụng nhiều phần mềm hỗ trợ (Chi tiết sẽ được trích dẫn ở phần “tài liệu tham khảo, công cụ hỗ trợ,...” Phần mềm sử dụng ngôn ngữ lập trình PHP, Javascript – NodeJS trên cơ sở dữ liệu MariaDB cùng với HTML và CSS, LESS, Laravel Framework (một framework của Php), ExpressJS (một framework của NodeJS), Socket.io (Thư viện hỗ trợ giao tiếp thời gian thực), jQuery và rất nhiều plugin của jQuery,.. (sẽ nói chi tiết ở phần tiếp theo)

*** Toàn bộ phần code của dự án đều được chia sẻ công khai trên các repository của Smartbots: <https://github.com/smart-bots> ***

1. Developer Friendly Code

Phần mềm được viết thân thiện, dễ đọc, dễ hiểu, theo sát PSR Coding Standard, nhiều Best Practice và Design Pattern của Javascript, PHP và riêng Laravel, được comment đầy đủ ở các lớp, hàm và biến.

```
public $lockoutTime = 60; // second

/**
 * Time for user to verify account
 * @var float
 */
public $verify_ttl = 1.5; // minutes

/**
 * Show up login page
 * @return Illuminate\Contracts\View\View
 */
public function getLogin() {
    return view('account.login');
}

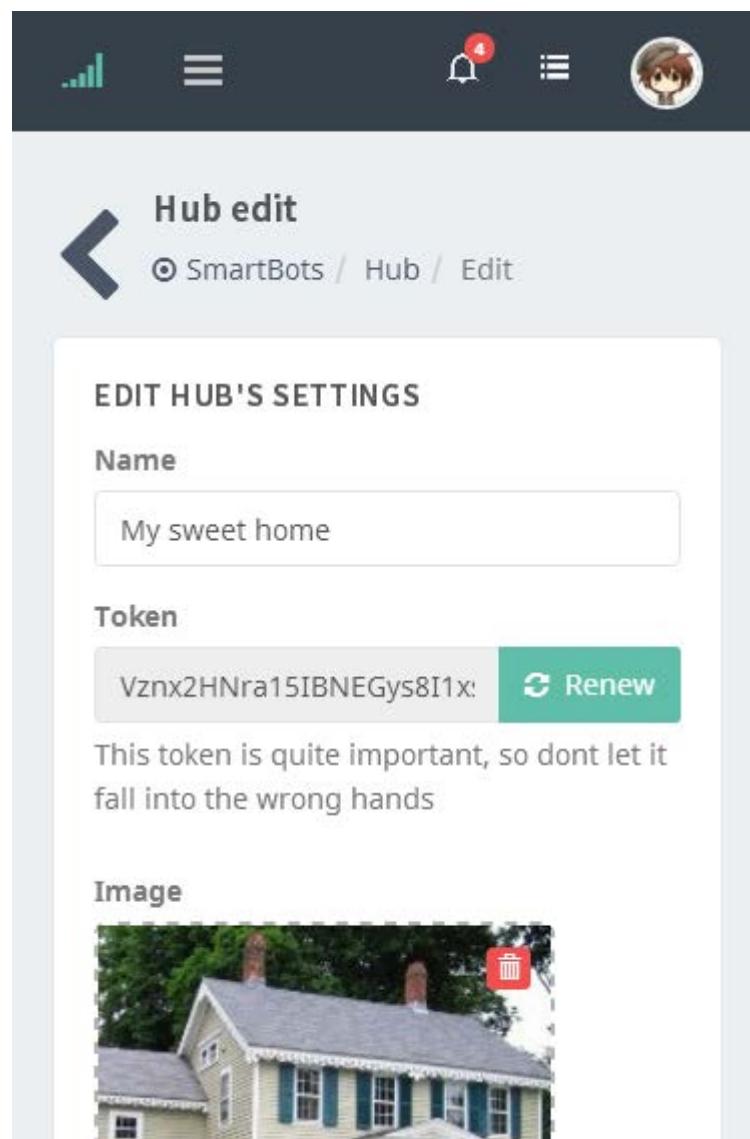
/**
 * Check if user login by username or email
 * @param Request $request
 * @return boolean
 */
public function loginWith(Request $request)
{
    return Is::email()->validate($request->username) ? 'email' : 'username';
}

/**
 * Handle a request to login
 * @param Request $request
 * @return Illuminate\Http\JsonResponse
 */
public function postLogin(Request $request) {
```

Hình 12. Bên trong một class với các biến và hàm được comment đầy đủ

2. Giao diện

Giao diện của phần mềm được thiết kế “Responsive”, tương thích với nhiều loại kích thước màn hình và nhiều loại trình duyệt khác nhau. Đã được kiểm thử ở nhiều kích thước và các trình duyệt: Chrome, Firefox, Firefox DevEdition, Cốc Cốc, Opera, Vivaldi, Yandex, Edge, Internet Explorer, Maxthon, Torch Browser. Sử dụng HTML5, CSS3, Twitter Bootstrap (một framework css, js nổi tiếng của twitter), giao diện phần mềm được Google Pagespeed Insights đánh giá 100/100 về trải nghiệm người dùng.



Hình 13: Giao diện trang dễ dàng thay đổi theo kích thước màn hình

3. Sử dụng LESS

LESS là một CSS preprocessor, biến cách viết CSS thông thường trở nên dễ dàng, ngắn gọn và hiệu quả hơn, đồng thời cũng dễ quản lý hơn với LESS cũng gần giống như một ngôn ngữ lập trình, có biến, có hàm, có kế thừa, có thể gọi CSS là một ngôn ngữ “Style Sheet Động”. Nhược điểm duy nhất là say khi viết LESS, ta phải compile nó ra CSS trơn, chứ không thể chèn luôn vào website vì browser chỉ có thể đọc được CSS. (Ở phần mềm này, chúng em dùng Gulp để compile)

```
707  /* =====
708  | Panels
709  ===== */
710
711  .panel {
712    border: none;
713    margin-bottom: 20px;
714    .panel-body {
715      padding: 20px;
716      p {
717        margin: 0px;
718      }
719      p + p {
720        margin-top: 15px;
721      }
722    }
723  }
724
725  .panel-heading {
726    border: none !important;
727    padding: 10px 20px;
728  }
729
```

Hình 14. Một số dòng LESS, trông như các câu lệnh if lồng nhau

4. Sử dụng jQuery

Phần mềm sử dụng jQuery framework – một framework của js, cùng với rất nhiều plugin của jQuery (sẽ liệt kê ở rõ ở phần sau) và các công nghệ của nó như Ajax. Công nghệ Ajax hiện đang sử dụng cho khá nhiều các quy trình làm việc của phần mềm để nâng cao sự tương tác, tốc độ tương tác với người dùng (Ví dụ như khi submit một form, ajax sẽ chỉ lấy phần phản hồi từ server, chứ không tải hết lại cả trang cùng với phản hồi như cách thông thường).

```

260     function hubReactivate() {
261         bootbox.confirm("R u sure?", function(result) {
262             if (result == true) {
263                 $.ajax({
264                     url : '{{ route('h::reactivate') }}',
265                     type : 'get',
266                     success : function (response)
267                     {
268                         hubStatus = $('#hubStatus').attr('href','javascript:hubDeactivate()');
269                         hubStatus.find('i').addClass('text-success').removeClass('text-danger');
270                         hubStatus.find('span').text('Activated');
271                         if (window.location.href == '{{ route('h::edit') }}') {
272                             hubReactivateBtn = $('#hubReactivateBtn').attr('id','hubDeactivateBtn')
273                             hubReactivateBtn.find('i').addClass('fa-ban').removeClass('fa-check-square');
274                             hubReactivateBtn.find('span').text('Deactivate');
275                         }
276                     }
277                 });
278             }
279         });
280     }
281     function hubLogout()
282     {
283         bootbox.confirm("R u sure?", function(result) {
284             if (result == true) {
285                 $.ajax({
286                     url : '{{ route('h::logout') }}',
287                     type : 'get',
288                     success : function (response)
289                     {
290                         window.location.href="{{ route('h::index') }}";
291                     }
292                 });
293             }
294         });
295     }

```

Hình 14: Một số đoạn Jquery điển hình

5. Sử dụng mô hình lập trình MVC (Model – View – Controller)

Cốt lõi của phần mềm đó chính là Laravel Framework. Đây là một framework sử dụng mô hình MVC kiểm cho việc lập trình phần mềm trở nên linh hoạt, dễ dàng debug, kiểm tra, sửa lỗi.

6. Lập trình hướng đối tượng:

Đơn giản, đối tượng hóa các tác vụ, một kỹ thuật mà ai cũng sử dụng

```
class MemberController extends Controller
{
    public function index()
    {
        $members = Hub::findOrFail(session('currentHub'))->members()->where('level', '!=', 0)->with('use
        for ($i = 0;$i < count($members);$i++) {
            $members[$i]['bots'] = Permission::where('member_id', $members[$i]['id'])->get()->count();
        }
        return view('hub.member.index')->withMembers($members);
    }
}
```

Hình 15. Dễ dàng tìm thấy các đoạn mã hướng đối tượng

7. Sử dụng Npm - Composer – Bower

Composer là một Dependency Management trong PHP, công cụ quản lý các thư viện của bên thứ 3 được sử dụng trong dự án. Bower cũng tương tự như vậy, nhưng nó quản lý các thư viện CSS và JS. Việc sử dụng Composer và Bower giúp tìm kiếm, cài đặt, nâng cấp và quản lý các thư viện tốt hơn, giảm tốc độ cho việc phát triển, nhất là trong khi làm việc với những dự án phức tạp.

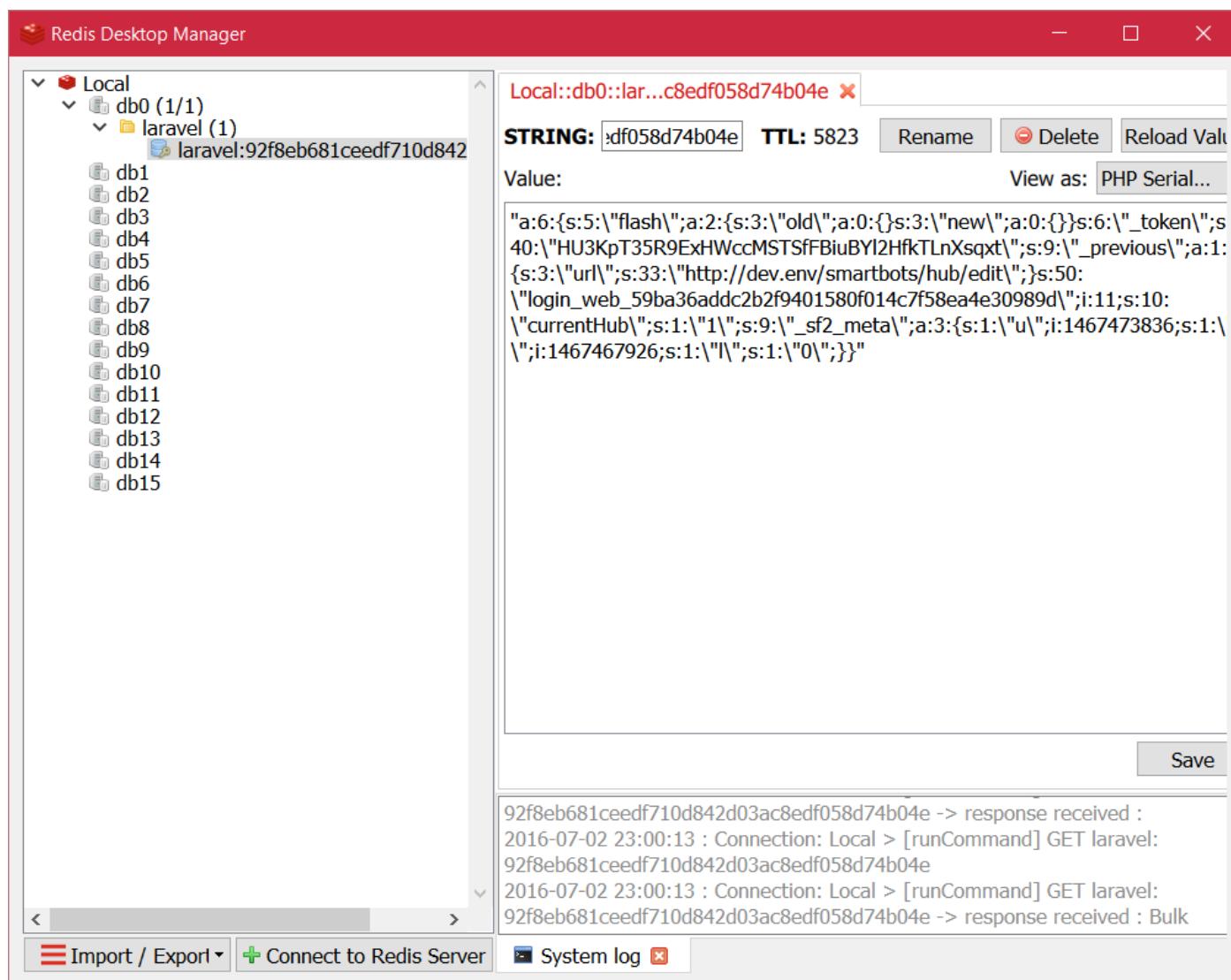
```
C:\xampp\htdocs\smartbots (smartbots@1.0.0)
\ composer update
Loading composer repositories with package information
Updating dependencies (including require-dev)
- Removing monolog/monolog (1.19.0)
- Installing monolog/monolog (1.20.0)
  Downloading: 100%
- Removing symfony/console (v3.0.7)
- Installing symfony/console (v3.0.8)
  Downloading: 100%
- Removing symfony/finder (v3.0.7)
- Installing symfony/finder (v3.0.8)
  Downloading: 100%
- Removing symfony/http-foundation (v3.0.7)
- Installing symfony/http-foundation (v3.0.8)
  Downloading: 100%
- Removing symfony/event-dispatcher (v3.1.1)
- Installing symfony/event-dispatcher (v3.1.2)
  Downloading: 100%
- Removing symfony/debug (v3.0.7)
- Installing symfony/debug (v3.0.8)
  Downloading: 100%
- Removing symfony/http-kernel (v3.0.7)
- Installing symfony/http-kernel (v3.0.8)

roadcastEvent
[2016-07-02 15:25:50] Process
ed: Illuminate\Broadcasting\B
roadcastEvent
[2016-07-02 15:25:50] Process
ed: Illuminate\Broadcasting\B
roadcastEvent
[2016-07-02 15:29:12] Process
ed: Illuminate\Broadcasting\B
roadcastEvent
[2016-07-02 15:29:12] Process
ed: Illuminate\Broadcasting\B
roadcastEvent
[2016-07-02 15:29:12] Process
ed: Illuminate\Broadcasting\B
roadcastEvent
Message received on "bot:change": {"event": "11.1", "data": {"id": 4, "state": 0}}
-----
Message received on "notification:new": {"event": "11.1", "data": {"id": 3, "user_id": 11, "hub_id": 1, "subject": "Automation \\When event fired\\ is running", "body": "Someone just make this automation run", "href": "#", "created_at": 1467498550, "holder": "icon:bolt"}}
-----
Message received on "notification:new": {"event": "11.1", "data": {"id": 4, "user_id": 11, "hub_id": 1, "subject": "Event \\Manually\\ is fired", "body": "Someone just make this event fired", "href": "#", "created_at": 1467498550, "holder": "icon:bolt"}}
-----
User #11 connected in hub #1
```

Hình 16: Sử dụng “composer update” để update core cho phần mềm

8. Sử dụng Redis

Redis là một hệ cơ sở dữ liệu mang phong cách NoSQL (tức lưu trữ dưới dạng khóa:giá trị (key – value)). Redis hỗ trợ nhiều hệ cấu trúc khác nhau với tốc độ cực nhanh. Khác với Memcached (được phần mềm sử dụng trước Redis, sau này chuyển sang sử dụng Redis) chỉ lưu được trên RAM, Redis còn hỗ trợ lưu trữ trên đĩa cứng, cho phép khắc phục khi sự cố xảy ra. Ngoài ra Redis cũng cung cấp nhiều cơ chế mạnh mẽ khác như Pub/Sub mà phần mềm chúng em đang sử dụng. Hiện tại chúng em đang sử dụng Redis để lưu trữ các Session, thiết lập hàng đợi (Job – Queue), Cache, ngoài ra còn sử dụng để giao tiếp thời gian thực với người dùng (qua cơ chế Pub/Sub)



Hình 17. Sử dụng Redis Desktop Manager để xem một session được lưu trữ

9. EventSource – Websocket: Socket.io – HTML Notification API

EventSource (hay còn được biết đến với cái tên ServerSentEvents) là một phương thức giao tiếp của máy chủ với người dùng, khác với giao thức http thông thường, EventSource được gửi chủ động từ Server chứ không đợi người dùng (client) gửi yêu cầu (request) tạo ra. EventSource được sử dụng trong phần mềm để gửi các dữ liệu một chiều đến cho người dùng (không cần phản hồi) ví dụ như trong chức năng xác nhận số điện thoại,...

HTML5 Websocket ra đời như một sự đổi mới khác của EventSource. Websocket là một giao thức mới sử dụng một TCP socket để giao tiếp 2 chiều với người dùng, ít tốn kém tài nguyên và vô cùng hiệu quả, tốc độ cao, tạo một kết nối bền bỉ đến người dùng, dễ dàng re-connect. Phần mềm sử dụng Websocket cho các chức năng như điều khiển bot, cập nhật bot, kiểm tra người dùng hoạt động, nhận thông báo (notification). Websocket không được hỗ trợ tốt trên php, nhưng với NodeJS thì khác, nên để dễ dàng hơn, phần mềm sử dụng Socket.io (một thư viện giao tiếp thời gian thực trên Websocket) để hỗ trợ giao tiếp thời gian thực với người dùng.

Chức năng thông báo (Notification) còn được sử dụng kết hợp với HTML5 Notification API để thông báo có thể tiếp xúc với người dùng tốt hơn ngay cả khi họ không ở trong tab của phần mềm.

```
1 var
2   app      = require('express')(),
3   http     = require('http').Server(app),
4   io       = require('socket.io')(http),
5   Redis    = require('ioredis'),
6   redis    = new Redis(),
7   redisClient = require("redis").createClient(),
8   PHPUnserialize = require('php-unserialize'),
9   crypto   = require('crypto'),
10  cookie   = require('cookie');
11
12 require('dotenv').config();
13
14 var port = process.env.SOCKETIO_PORT;
15 namespace = process.env.APP_NAMESPACE;
16
17 var userIdKey = 'login_web_59ba36addc2b2f9401580f014c7f58ea4e30989d';
18
19 var users = 0,
20 user = [];
21
22 //-----
23
24 io = io.of(namespace);
25
26 redis.psubscribe('*', function(err, count) { });
27
28 redis.on('pmassage', function(subscribed, channel, message) {
29
30   console.log('Message received on "' + channel + '": ' + message);
31   console.log('-----');
32
33   message = JSON.parse(message);
34
35   io.to(message.event).emit(channel, message.data);
36
37 });
38
39 io.on('connection', function(socket) {
```

Hình 18: Một đoạn code của server NodeJS, kết hợp giữa nhiều thành phần

*** Phần sau đây sẽ nói chủ yếu về các công nghệ, kỹ thuật của phần mềm đáng chú ý được sử dụng trên Laravel ***

10. Route: Bộ định tuyến

Tất cả các request của người dùng được gửi lên website đều được đi qua route, route nắm vai trò chỉ đường cho request, điều tiếp request đến nơi (controller,..) mà đó được xử lý. Route cũng rất có giá trị cho việc SEO, ngoài ra nó còn giống như một chuỗi các định nghĩa uri trong .htaccess vậy.

```

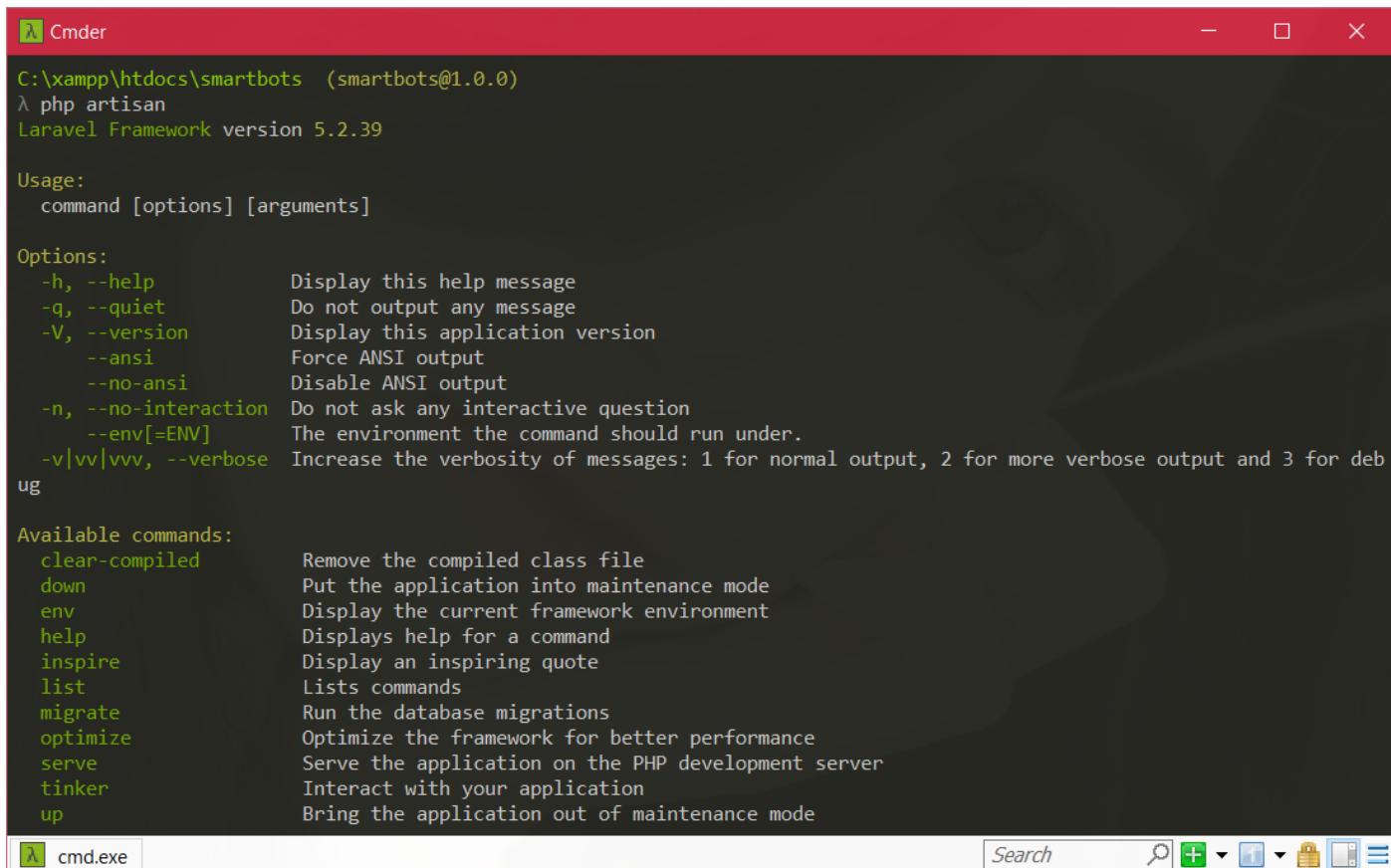
16   Route::group([
17     'prefix' => 'account',
18     'as'      => 'a'
19   ], function () {
20
21     Route::group(['middleware' => ['nonAuthed']], function() {
22       Route::get('login', 'UserController@getLogin')->name('::login');
23       Route::post('login', 'UserController@postLogin');
24       Route::get('register', 'UserController@getRegister')->name('::register');
25       Route::post('register', 'UserController@postRegister');
26       Route::get('forgot', 'UserController@getForgot')->name('::forgot');
27       Route::post('forgot', 'UserController@postForgot');
28     });
29
30     Route::group(['middleware' => ['authed']], function() {
31       Route::get('/', function () {
32         return redirect()->to(route('a::edit'),301);
33       });
34       Route::get('edit', 'UserController@edit')->name('::edit');
35       Route::post('edit', 'UserController@update')->name('::update');
36       Route::get('logout', 'UserController@logout')->name('::logout');
37       Route::post('logout', 'UserController@logout')->name('::logout');
38       Route::get('change-pass', 'UserController@getChangePass')->name('::changePass');
39       Route::post('change-pass', 'UserController@postChangePass');
40
41       Route::get('search/{query?}', 'UserController@search')->name('::search');
42     });
43   });
44 });

```

Hình 19. Một đoạn định tuyến trong routes.php

11. Artisan

Artisan là một CLI (Command-line interface – giao diện giao tiếp bằng dòng lệnh) được đi kèm với Laravel, giúp dễ dàng khởi tạo những thành phần cần thiết của dự án, thực hiện các tác vụ, tao tác với database,... bằng những dòng lệnh. Ngoài ra phần mềm còn thêm vào nhiều câu lệnh khác phục vụ cho nhiều mục đích riêng



```
C:\xampp\htdocs\smartbots  (smartbots@1.0.0)
λ php artisan
Laravel Framework version 5.2.39

Usage:
  command [options] [arguments]

Options:
  -h, --help            Display this help message
  -q, --quiet           Do not output any message
  -V, --version          Display this application version
  --ansi                Force ANSI output
  --no-ansi              Disable ANSI output
  -n, --no-interaction  Do not ask any interactive question
  --env[=ENV]             The environment the command should run under.
  -v|vv|vvv, --verbose   Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for deb
ug

Available commands:
  clear-compiled        Remove the compiled class file
  down                  Put the application into maintenance mode
  env                   Display the current framework environment
  help                  Displays help for a command
  inspire                Display an inspiring quote
  list                  Lists commands
  migrate               Run the database migrations
  optimize              Optimize the framework for better performance
  serve                 Serve the application on the PHP development server
  tinker                Interact with your application
  up                   Bring the application out of maintenance mode

cmd.exe
```

Hình 20: Các dòng lệnh của artisan

12. Model – Eloquent

Eloquent là một ORM (Object-relational mapping), đem đến một phương thức khác để làm việc với các Model, dễ dàng hơn thay vì chỉ sử dụng những dòng lệnh query đơn thuần và phức tạp.

```

16 class User extends Authenticatable
17 {
18     protected $table = 'users';
19     protected $fillable = ['username', 'name', 'email', 'phone', 'password', 'avatar'];
20     protected $hidden = ['password', 'remember_token'];
21     public $timestamps = false;
22
23     public function members() {
24         return $this->hasMany('SmartBots\Member', 'user_id');
25     }
26
27     public function botpermissions() {
28         return $this->hasMany('SmartBots\BotPermission', 'user_id');
29     }
30
31     public function schedulepermissions() {
32         return $this->hasMany('SmartBots\BotPermission', 'user_id');
33     }
34
35     public function schedules() {
36         return $this->hasMany('SmartBots\Schedule', 'user_id');
37     }
38
39     public function highpermissions() {
40         return $this->hasMany('SmartBots\HighPermission', 'user_id');
41     }
42
43     public function hubs() {
44         return $this->belongsToMany('SmartBots\Hub', 'members');
45     }
46
47     public function bots() {
48         $botpermissions = $this->botpermissions;
49         foreach ($botpermissions as $botpermission) {
50             if (!isset($bots)) {
51                 $bots = $botpermission->bots;
52             } else {
53                 $bots = $bots->merge($botpermission->bots);
54             }
55         }
56         return $bots;
57     }

```

Hình 21. Một đoạn mã của Model “User” với các relationship, scope

13. Database: Migrate, Seeding

Thông thường, để khởi tạo cơ sở dữ liệu cho dự án, họ sẽ dùng những giao diện như PhpMyAdmin, hoặc là các dòng lệnh sql. Nhưng với Laravel, nó được thực hiện một cách nhanh chóng hơn, dễ quản lý hơn bằng việc viết các dòng lệnh PHP ngay trong dự án, giúp dễ dàng quản lý việc tạo lập, các thuộc tính của cơ sở dữ liệu.

```
public function up()
{
    Schema::create('schedules', function (Blueprint $table) {
        $table->increments('id');
        $table->integer('user_id')->unsigned();
        $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
        $table->integer('hub_id')->unsigned();
        $table->foreign('hub_id')->references('id')->on('hubs')->onDelete('cascade');
        $table->string('name',100);
        $table->text('description');
        $table->text('action');
        $table->tinyInteger('type');
        $table->text('data');
        $table->tinyInteger('condition_type');
        $table->tinyInteger('condition_method');
        $table->text('condition_data');
        $table->text('activate_after');
        $table->text('deactivate_after_times');
        $table->text('deactivate_after_datetime');
    });
}
```

Hình 22. Một đoạn mã migration để khởi tạo một table mới

Ngoài ra để phục vụ cho mục đích phát triển phần mềm còn có sự hỗ trợ của Seeding (tạo dữ liệu giả) giúp dễ dàng hơn trong việc kiểm thử các chức năng cần nhiều dữ liệu.

```
public function run()
{
    $faker = Faker\Factory::create();
    for ($i=0;$i<10;$i++)
    {
        User::create([
            'username' => $faker->username,
            'name' => $faker->name,
            'email' => $faker->email,
            'phone' => $faker->phoneNumber,
            'password' => Hash::make($faker->password),
            'avatar' => 'http://loremflickr.com/200/200/?'.str_random(5)
        ]);
    }
}
```

Hình 23. Đoạn mã seed để tạo ra các người dùng giả phục vụ việc kiểm thử phần mềm

14. View – Blade

Hỗ trợ đắc lực cho View trong Laravel chính là Blade – một template engine được tính hợp. Blade giúp cho người phát triển không cần phải dùng những dòng lệnh PHP trong view, tăng tốc việc xử lý View, các thành phần của view.

```

1  @extends('hub.master')
2  @section('title', 'Hub Bots')
3  @section('additionHeader')
4    <link rel="stylesheet" href="{{ asset('resources/assets/plugins/bootstrap-switch/bootstrap-switch.min.css') }}>
5  @endsection
6  @section('additionFooter')
7    <script src="{{ asset('resources/assets/plugins/bootstrap-switch/bootstrap-switch.min.js') }}></script>
8  <script>
9    $fn_bootstrapSwitch_defaults_size = 'small';

```

Hình 24. Đoạn mã sử dụng Blade Template Engine

15. Authentication – Authorization

Hai công cụ này hỗ trợ rất nhiều trong việc quản lý, thiết lập chứng thực các request của người dùng, phân quyền người dùng, giúp giảm thiểu rất nhiều thời gian cho việc phát triển các chức năng này của dự án.

```

$rules = [
    'username'          => 'sometimes|required|between:6,255',
    'email'             => 'sometimes|required|email',
    'password'          => 'required|between:6,255',
];

$this->validate($request, $rules);

$isUsingThrottles = property_exists($this, 'maxLoginAttempts') ? true : false;

$lockedOut = $this->hasTooManyLoginAttempts($request);

if ($isUsingThrottles && $lockedOut) {

    event(new Lockout($request));

    $seconds = $this->secondsRemainingOnLockout($request);

    return redirect()->back()
        ->withInput($request->only($this->loginWith($request), 'remember'))
        ->withErrors([
            'custom' => trans('auth.throttle',[ 'second' => $seconds]),
        ]);
}

$credentials = $request->only($this->loginWith($request), 'password');

if (auth()->guard($this->guard)->attempt($credentials, $request->has('remember'))) {

    if ($isUsingThrottles) {
        $this->clearLoginAttempts($request);
    }

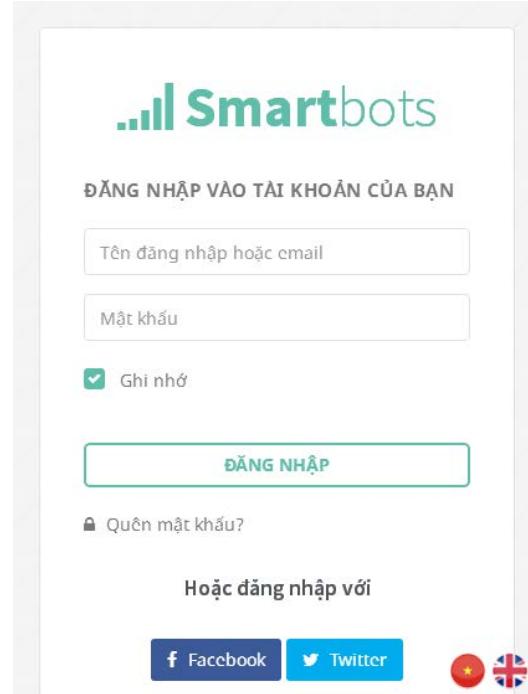
    return redirect()->route($this->redirectAfterLogin);
}

```

Hình 25. Chức năng đăng nhập được xử lý nhanh hơn giờ Authentication

16. Localization – Language: Translatable

Localization đã cung cấp một phương thức thuận tiện, đơn giản để có thể phát triển ứng dụng hỗ trợ nhiều ngôn ngữ khác nhau. Hiện tại phần mềm đang hỗ trợ 2 ngôn ngữ là Tiếng Anh và Tiếng Việt, trong tương lai dự kiến sẽ hỗ trợ nhiều ngôn ngữ hơn.



Hình 26. Có thể đổi qua lại giữa tiếng anh và tiếng việt bằng 2 nút nằm góc dưới màn hình

17. Middleware: Request Filter

Middleware của Laravel là một cơ chế khác hơn với Authentication và Authorization, nó giúp bổ sung thêm các lớp lọc để xác thực từng request riêng biệt của người dùng, giúp bảo mật cực kỳ tốt cho ứng dụng. Ví dụ điển hình là việc kiểm tra các CSRF-token khi người dùng gửi các POST/PUT/PATCH request.

```
protected function tokensMatch($request)
{
    $sessionToken = $request->session()->token();

    $token = $request->input('_token') ?: $request->header('X-CSRF-TOKEN');

    if (! $token && $header = $request->header('X-XSRF-TOKEN')) {
        $token = $this->encrypter->decrypt($header);
    }

    if (! is_string($sessionToken) || ! is_string($token)) {
        return false;
    }

    return hash_equals($sessionToken, $token);
}
```

Hình 27. Một hàm nhỏ trong đoạn mã kiểm tra CSRF-Token

18. Job: Queue – Task Schedule

Job và Queue của Laravel được phần mềm sử dụng với Redis (cơ sở dữ liệu đã giới thiệu ở phần trên) để thực hiện các tác vụ hàng chờ (như chạy các Smart Schedule, gửi email,...)

19. Event – Listener

Vì giao tiếp thời gian thực của phần mềm được hỗ trợ bởi NodeJS – ExpressJS – SocketIO và PHP server rất khó để giao tiếp thông tin với NodeJS, thế nên lại cần thêm một sự hỗ trợ khác đến từ Redis – từ cơ chế Pub/Sub của nó. Mỗi lần có một sự kiện cần được broadcast (thời gian thực), PHP server public một key mới trong một kênh (channel) nào đó lên Redis, còn NodeJS server thì Subscribe vào channel đó, mỗi khi nhận được dữ liệu mới, NodeJS gọi ngay socketio ra để gửi đến người dùng.

20. Elixir: Gulp – Browsersync

Gulp là một Javascript Task Runner được xây dựng trên nền tảng NodeJS, chúng giúp tự động hóa các tác vụ cần thực hiện với CSS, SASS, LESS, JS như compile, minify, format, gộp file. Elixir đơn giản là một công cụ giúp phát triển và sử dụng Gulp đơn giản hơn.

Browsersync lại là một công cụ khác hỗ trợ rất đắc lực cho việc phát triển Front-End, nó cung cấp một cơ chế live-reload (tự động tải lại) mỗi khi có thay đổi nào đó trong thành phần của code, ngoài ra còn đồng bộ hóa (sync) giữa các client với nhau (như đồng bộ hóa cuộn trỏ chuột (scroll), kích chuột (click) và các form)

Cả 3 Elixir, Gulp, Browsersync đều hoạt động trên nền tảng NodeJS.

```
elixir.extend('htmlmin', function () {
  new elixir.Task('htmlmin', function () {
    return gulp.src('./storage/framework/views/*')
      .pipe(htmlmin({
        collapseWhitespace: true,
        removeAttributeQuotes: true,
        removeComments: true,
        minifyJS: true,
      }))
      .pipe(gulp.dest('./storage/framework/views'));
  })
  .watch('./storage/framework/views/*');
});

elixir.extend('imagemin', function () {
  new elixir.Task('imagemin', function () {
    return gulp.src('./public/images/*')
      .pipe(imagemin({
        progressive: true
      }))
      .pipe(gulp.dest('./public/images'));
  })
  .watch('./public/images/*');
});

elixir(function(mix) {
```

Hình 28. Một đoạn mã của gulp, dùng để nén html, hình ảnh

21. Testing - PHPUnit

Kiểm thử là một phần, một giai đoạn không thể thiếu trong việc phát triển phần mềm, giúp hạn chế nhiều bug, lỗi phát sinh. Nhưng kiểm thử cũng là thứ mất nhiều thời gian, mà không một developer nào thích bởi nhiều lý do. Nhưng bỏ qua việc đó, kiểm thử vẫn rất cần thiết.

PHPUnit là một thư viện hỗ trợ việc kiểm thử trên php với vô số phương thức khác nhau. Ngoài ra, Laravel còn mở rộng nó thành các phương thức dễ hiểu, dễ đọc và dễ viết hơn. Phần mềm của chúng em cũng có một số Unit test (không nhiều vì vấn đề thời gian và nhân lực)

-- Nhìn chung, có thể nói là phần mềm của chúng em đã khai thác được khá nhiều sức mạnh từ Laravel (mặc dù có thể nói sức mạnh của nó là vô hạn), cùng với các công nghệ tuyệt vời khác (như websocket – socket.io, redis,...)

```
public function testRegister()
{
    $this->visit('/account/register')
        ->see('Register')
        ->type('tester','username')
        ->type('Tờ ét tờ','name')
        ->type('test@te.st','email')
        ->type('testtest','password')
        ->type('testtest','password_confirmation')
        ->check('agree_with_terms')
        ->press('Register')
        ->seeJson([
            'success' => true,
        ])
        ->seeInDatabase('users', ['username' => 'tester']);
}
```

Hình 29. Đoạn mã để test chức năng đăng ký

22. Sử dụng Sentry

Sentry là một “Bug tracker”, “Crash report” mang phong cách thời gian thực, cùng với thông tin về bug, crash đầy đủ hơn so với hệ thống log bình thường, cùng với một dashboard quản lý, thống kê các lỗi, giúp cho việc chỉnh sửa phần mềm, sửa lỗi trở nên dễ dàng hơn.

Category	Error Type	Description	Timestamp	Count	Users
Error	InvalidArgumentException	UserController.php:SmartBots\Http\Controllers\UserController in ...	Route [landing::index] not defined. 26 minutes ago — 27 minutes old php	2	1
Error	Error	smartbots/public/libs/jquery-legacy/jquery in Function.t.error	Syntax error, unrecognized expression: a[href="#"]. An hour ago — an hour old javascript	4	1
Error	Error	smartbots/public/libs/jquery-legacy/jquery in Function.Sizzle.error	Syntax error, unrecognized expression: a[href="#"]. 2 hours ago — 12 days old javascript	2	2
Error	Symfony\Component\Debug\Exception\FatalThrowableError	Verified.php:SmartBots...	Parse error: syntax error, unexpected ';' expecting ',' or ')'. 2 days ago — 2 days old php	1	1
Error	Symfony\Component\Debug\Exception\FatalThrowableError	AppServiceProvider.php...	Class 'SmartBots\Providers\App' not found. 2 days ago — 2 days old php	1	1
Error	Symfony\Component\Debug\Exception\FatalThrowableError	routes.php:SmartBots\...	Parse error: syntax error, unexpected 'return' (T_RETURN). 2 days ago — 2 days old php	1	1
Error	Illuminate\Routing\Exceptions\UrlGenerationException	NonAuthenticated.php:SmartB...	Missing required parameters for [Route: lang] [URI: language/{lang}]. 2 days ago — 2 days old php	2	1
Info	ErrorException	routes.php:Illuminate\Foundation\Bootstrap\HandleExceptions in handleError	Undefined property: Illuminate\Support\Facades\Request::\$lang 2 days ago — 2 days old php	1	1
Error	InvalidArgumentException	NonAuthenticated.php:SmartBots\Http\Middleware\NonAuthentic...	Route [lang] not defined. 2 days ago — 2 days old php	1	1
Error	Symfony\Component\Debug\Exception\FatalThrowableError	routes.php:SmartBots\...	Parse error: syntax error, unexpected ')'; expecting 'l'	2	1

Hình 30. Quản lý lỗi, bug bằng Sentry

23. Sử dụng Twilio

Là twilio là một call/sms API được sử dụng rộng rãi, với tốc độ cao và giá cả hợp lý. Hiện tại phần mềm dùng twilio để xác nhận số điện thoại của người dùng.

```
Route::get('/twilio-voice-request', function() {
    return response('<?xml version="1.0" encoding="UTF-8"?><Response><Reject reason="busy" /></Response>')
        ->header('Content-Type', 'application/xml');
})->name('twilioVoiceRequest');

Route::post('/twilio-voice-status-callback', 'UserController@handleTwilioVoiceStatusCallback')->name('twilioVoiceStatus');
```

Hình 31. Đoạn route xử lý dữ liệu của twilio gửi đến

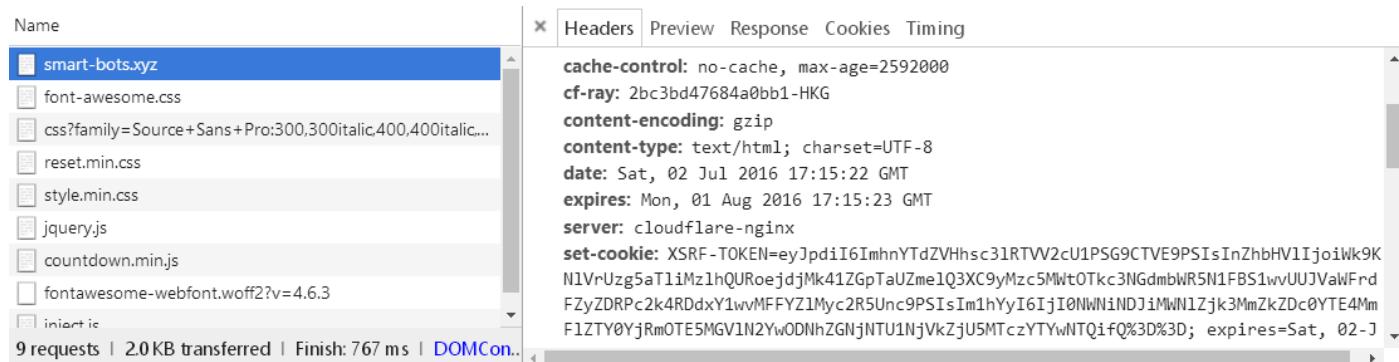
24. Hiệu năng

Để tăng tối đa hiệu năng, tốc độ, phần mềm đã tận dụng sức của Gzip (Mod Deflate) của apache để nén đến tối đa dung lượng ở server trước khi gửi đến client, từ đó giúp tiết kiệm (giảm) lượng lướn băng thông và tăng rất nhiều tốc độ tải.

Tận dụng tối đa sức mạnh của cache: các mod như expires,... giúp tăng tối đa thời gian lưu trữ (max-age) cho các file trên cache của trình duyệt.

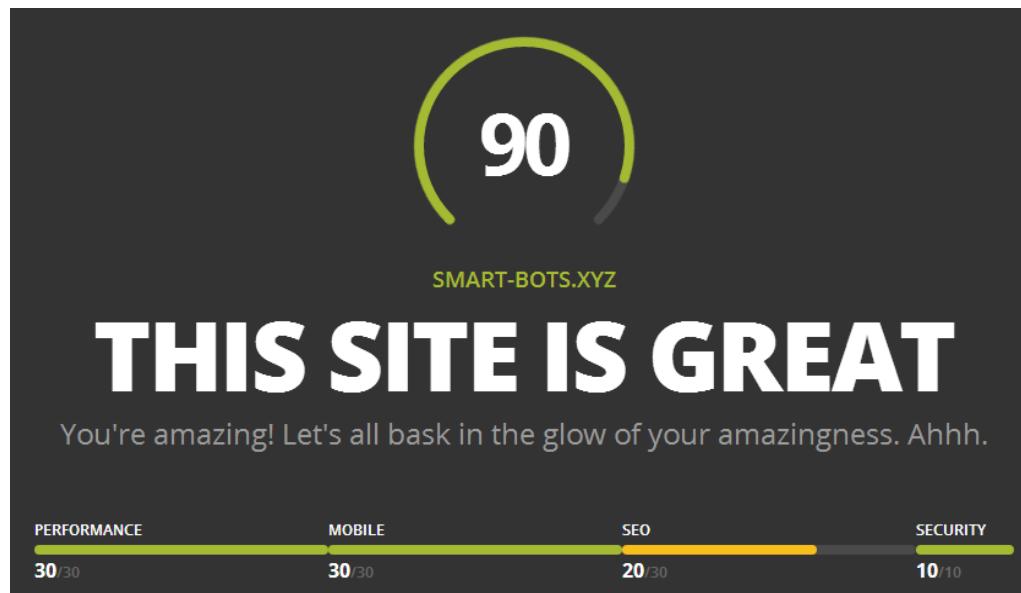
Ngoài gzip, phần mềm còn có chức năng tự động nén (minify) tất cả các html, css, js, ảnh ngày từ trên server (sử dụng các thư viện của gulp như htmlmin, imagemin,..., các site như tinypng, tinyjpg) làm giảm đến 80% dung lượng tải xuống cho client.

Ngoài ra, server hiện tại được đặt trên cơ sở của Amazon EC2, sử dụng Cloudflare DNS và cdn, nên tốc độ và hiệu năng cũng được tăng lên đáng kể.



Hình 32. Tốc độ load cực nhanh, dữ liệu cực nhỏ

Kết quả hiệu năng của phần mềm được kiểm định trên các site Google Pagespeed: Tốc độ: 75/100, Trải nghiệm người dùng: 100/100, Hiệu năng: 89/100. Website Grader:



25. Bảo mật

Bảo mật của phần mềm khá tốt, dựa trên cơ sở bảo mật đã có sẵn từ Laravel Framework như: hash, crypt rất cả các session, cookie (sử dụng chuẩn AES-128 – Rijndael-128), chống được CSRF (nhờ vào cơ chế kiểm tra CSRF-token mỗi request của client) , XSS, Sql Injection,...

Ngoài ra việc đặt server trên Amazon EC2 giúp việc quản lí các port dễ dàng hơn, sử dụng Cloudflare cũng giúp tránh được các cuộc tấn công từ chối dịch vụ (DDoS).

Ngoài ra phần mềm cũng bảo vệ các file, folder quan trọng như .env (chứa các thiết lập của phần mềm)

Ngoài ra phần mềm còn hỗ trợ giao thức bảo mật SSL (https), tự động backup cơ sở dữ liệu, khóa các ip khi gửi quá nhiều request.

26. Thư viện PHP sử dụng: có thể tìm thấy trong thư mục “vendor”

<https://github.com/briannesbitt/Carbon> thư viện hỗ trợ xử lý thời gian

<https://github.com/guzzle/guzzle> thư viện http client, hỗ trợ gửi request

<https://github.com/Respect/Validation> thư viện hỗ trợ xác thực dữ liệu

<https://github.com/Intervention/image> thư viện hỗ trợ xử lý hình ảnh

<https://github.com/nrk/predis> Redis adapter

<https://github.com/camroncade/timezone> thư viện hỗ trợ làm việc với timezone

<https://github.com/rap2hpoutre/laravel-log-viewer> thư viện bổ xung log viewer và laravel

<https://github.com/spatie/laravel-backup> thư viện hỗ trợ backup dữ liệu

<https://github.com/getsentry/sentry-laravel> Sentry adapter

<https://github.com/Propaganistas/Laravel-Phone> thư viện hỗ trợ làm việc với số điện thoại

<https://github.com/hoaproject/Eventsource> thư viện hỗ trợ làm việc với giao thức EventSource

<https://github.com/fzaninotto/faker> thư viện hỗ trợ tạo dữ liệu giả

<https://github.com/mockery/mockery> thư viện hỗ trợ tạo php unit test hiệu quả

<https://github.com/doctrine/dbal> thư viện hỗ trợ xử lý database

<https://github.com/laracasts/Laravel-5-Generators-Extended> thư viện bổ xung laravel generator

<https://github.com/barryvdh/laravel-translation-manager> thư viện hỗ trợ dịch ngôn ngữ

<https://github.com/themsaid/laravel-mail-preview> thư viện hỗ trợ việc kiểm thử email

<https://github.com/barryvdh/laravel-ide-helper> thư viện hỗ trợ cho các IDE

<https://github.com/barryvdh/laravel-debugbar> thư viện hỗ trợ cho việc test, phát triển

Và một số thư mục core được require có thể tìm thấy trong thư mục “vendor”

27. Thư viện Javascript sử dụng: Có thể tìm thấy trong thư mục “public/libs”

<https://github.com/amcharts/amcharts3> thư viện biểu đồ

<https://github.com/daneden/animate.css/> thư viện hiệu ứng động

<https://github.com/malsup/blockui> thư viện hỗ trợ tạo giao diện block

<https://github.com/twbs/bootstrap> framework của css, js đến từ twitter

<https://github.com/Eonasdan/bootstrap-datetimepicker> thư viện tạo datetime picker

<https://github.com/silviomoreto/bootstrap-select> thư viện tạo select input phong cách bootstrap

<http://detectmobilebrowsers.com/> thư viện phát hiện ứng dụng di động

<https://github.com/ftlabs/fastclick> thư viện hỗ trợ click nhanh

<https://github.com/FortAwesome/Font-Awesome> thư viện font-icon

<https://github.com/wycats/handlebars.js/> một template engine của js

<http://codecanyon.net/item/html-5-upload-image-ratio-with-drag-and-drop/8712634> thư viện tạo drag & drop để upload hình ảnh

<https://github.com/jackocnr/intl-tel-input> thư viện tạo giao diện số điện thoại trực quan

<https://github.com/jquery/jquery> framework javascript nổi tiếng

<https://github.com/inuyaksa/jquery.nicescroll> thư viện tạo thanh scroll

<https://github.com/flesler/jquery.scrollTo> thư viện hỗ trợ scroll tới đối tượng

<https://github.com/Modernizr/Modernizr> thư viện hỗ trợ xác nhận các tính năng của trình duyệt

<https://github.com/moment/moment> thư viện xử lý thời gian

<https://github.com/lou/multi-select> thư viện tạo giao diện multi select

<https://github.com/notifyjs/notifyjs> thư viện tạo notification

<https://github.com/riklomas/quicksearch> thư viện hỗ trợ search trên DOM

<https://github.com/getentry/raven-js> Sentry adapter cho js

<https://github.com/rochal/jQuery-slimScroll> thư viện tạo thanh scroll

<https://github.com/socketio/socket.io-client> socketIO client cho js

<https://github.com/t4t5/sweetalert> thay thế cho alert thông thường

<https://github.com/lykmapipo/themify-icons> thư viện font-icon

<https://github.com/tommoor/tinycon> thư viện hỗ trợ tạo số notification ngay trên favicon của

<https://github.com/twitter/typeahead.js/> thư viện auto suggest nổi tiếng của twitter

<https://github.com/fians/Waves> thư viện tạo hiệu ứng nút nhấn phong cách Material

<https://github.com/matthieu/WOW> thư viện tạo hiệu ứng khi scroll trang

28. Thư viện NodeJS sử dụng:

Có thể tìm thấy trong thư mục “node_modules”

<https://github.com/jshttp/cookie> thư viện xử lý, parser cookie

<https://github.com/brix/crypto-js> thư viện hỗ trợ en-decrypt

<https://github.com/motdotla/dotenv> thư viện hỗ trợ đọc file cấu hình .env

<https://github.com/expressjs/express> một framework của nodeJS

<https://github.com/luin/ioredis> thư viện hỗ trợ làm việc với Redis

<https://github.com/naholyr/js-php-unserialize> thư viện hỗ trợ dịch dữ liệu của php

https://github.com/NodeRedis/node_redis lại là một thư viện hỗ trợ làm việc với Redis

<https://github.com/request/request> thư viện hỗ trợ việc gửi request

<https://github.com/socketio/socket.io> thư viện giao tiếp thời gian thực

29. Tài liệu tham khảo:

Các website

<http://wikipedia.org>

<http://w3schools.com>

<http://bootsnipp.com>

<http://git-scm.com/doc>

<http://jstherightway.org/>

<https://laravel.com/docs/5.2/>

<http://getbootstrap.com/>

<http://api.jquery.com/>

<http://www.linxiang.info/l5-cs/>

<http://laptrinh.io>

<http://laracasts.com>

<http://laravel.io/forum>

<http://learninglaravel.net/>

<http://laravel-tricks.com/>

<https://mattstauffer.co/blog>

<http://www.laravelbestpractices.com/>

<http://www.phptherightway.com/>

<https://github.com/php-fig/fig-standards>

<http://stackoverflow.com>

30. Công cụ hỗ trợ

Phần mềm: XAMPP, MySQL Workbench, Redis Desktop Manager

Shell: Cmder

Công cụ giao tiếp, làm việc nhóm: Github (Chia sẻ code, code nhóm), Slack (giao tiếp giữa các thành viên)

Trình duyệt: Chrome, Firefox, Firefox DevEdition, Cốc Cốc, Opera, Vivaldi, Yandex, Edge, Internet Explorer, Maxthon, Torch Browser

Các service hỗ trợ: Sentry (quản lý bug – lỗi), Twilio (call/sms api), Mailgun (hỗ trợ gửi mail)

Các Extension của Chrome, Firefox, Opera: Accessibility Developer Tools, Ajax Debugger, CssViewer, EditThisCookie, Forget Me, Instant Wireframe, jQuery Audit, JSONView, Live HTTP Headers, Octo Linker, SnappySnippet, Tampermonkey, User-Agent Switcher, Wappalyzer, Web Developer, Web Developer Checklist, WhatFont, Window Resizer, Firebug

Các IDE – Editor: Sublime Text 3, Notepad++

Các plugin của Sublime text: Package Control, SublimeTextTrans, TrailingSpaces, Trimmer, Emmet, Alignment, SidebarEnhancements, DocBlockr, AdvancedNewFile, Icon Fonts, All Autocomplete, Hasher, Color Highlighter, BracketHighlighter, CodeFormatter, Minifier, StringEncode, PHP Syntax Checker, CodeIntel, PHPIintel, SublimeLinter, Laravel Blade Highlighter, HTML5, HTMLAttributes, CSSComb, Css Extended Completions, JavaScript Completions, Javascript Console, php-grammar, php-snippets, PHP Completion Kit, PHPCS, CSS Unminifier, ClickableUrls, Autoprefixer, Terminal, JSHint Gutter, Markdown Extended, LoremIpsum, Markdown Preview, Auto-save.

E. Hướng dẫn cài đặt phần mềm

Có thể sử dụng dự án thông qua địa chỉ: <http://smart-bots.xyz>, ngoài ra cũng có thể cài đặt thử phần mềm ngay trên máy tính.

Yêu cầu cần có:

- Ruby 2.3.0+
- Python 2.7.11+
- .NET Framework 4.5.1
- Git 2.9.0+
- Composer 1.1.2+
- Apache 2.2+ hoặc Nginx 1.10.1+
- PHP 7+
- MySQL 5.1+ hoặc MariaDB 10+
- Redis 3.0+
- NodeJS 4.4.4+
- NPM 2.15+
- Bower 1.7.9+
- Gulp 3.9+
- OpenSSL PHP Extension Enabled
- PDO PHP Extension Enabled
- Mysqli PHP Extension Enabled
- Mbstring PHP Extension Enabled
- Tokenizer PHP Extension Enabled
- MCrypt PHP Extension Enabled
- Fileinfo PHP Extension Enabled
- mod_rewrite Apache Enabled
- allow_url_fopen Apache Enabled
- GD Library 2.0+ hoặc Imagick PHP extension 6.5.7+

Một số thành phần khác: Tài khoản Sentry, Tài khoản Twilio

Sau khi cài đặt đủ các phần cần thiết, chạy câu lệnh:

```
git clone https://github.com/smart-bots/server.git smartbots  
cd smartbots  
composer install  
composer dumpautoload  
php -r "copy('.env.example', '.env');"
```

Mở file “.env” để cấu hình cho phần mềm (URL, DATABASE, SENTRY, TWILIO,...)

```
php artisan db:create  
php artisan key:set  
php artisan migrate
```

Cài đặt cấu hình cho Twilio

Tải <https://curl.haxx.se/ca/cacert.pem> và đặt ở thư mục bất kỳ

Chỉnh sửa file “php.ini”: thêm vào cuối dòng:

```
curl.caInfo = "PATH_TO/cacert.pem"
```

Sau đó chạy câu lệnh:

```
php artisan twilio:setup  
npm install -g gulp  
npm install -g bower  
npm install [thêm “-no-bin-links” nếu chạy trên windows]
```

Sau đó chạy các tác vụ song song cho phần mềm

```
cmd /c taskscheduling.bat
```

hoặc chạy file “taskscheduling.bat” nếu chạy trên windows, cho linux:

```
* * * * * php /path/to/artisan schedule:run >> /dev/null 2>&1
```

Tiếp theo, chạy:

```
php artisan queue:work --daemon  
npm start
```

Một số câu lệnh cần sử dụng trong quá trình chạy phần mềm:

```
gulp watch  
php artisan optimize  
php artisan route:cache  
php artisan config:cache
```

Để update các thành phần của phần mềm:

```
composer update  
npm update  
bower update  
gulp
```

Một số câu lệnh dùng trong trường hợp đặc biệt: Queue bị lỗi, fail

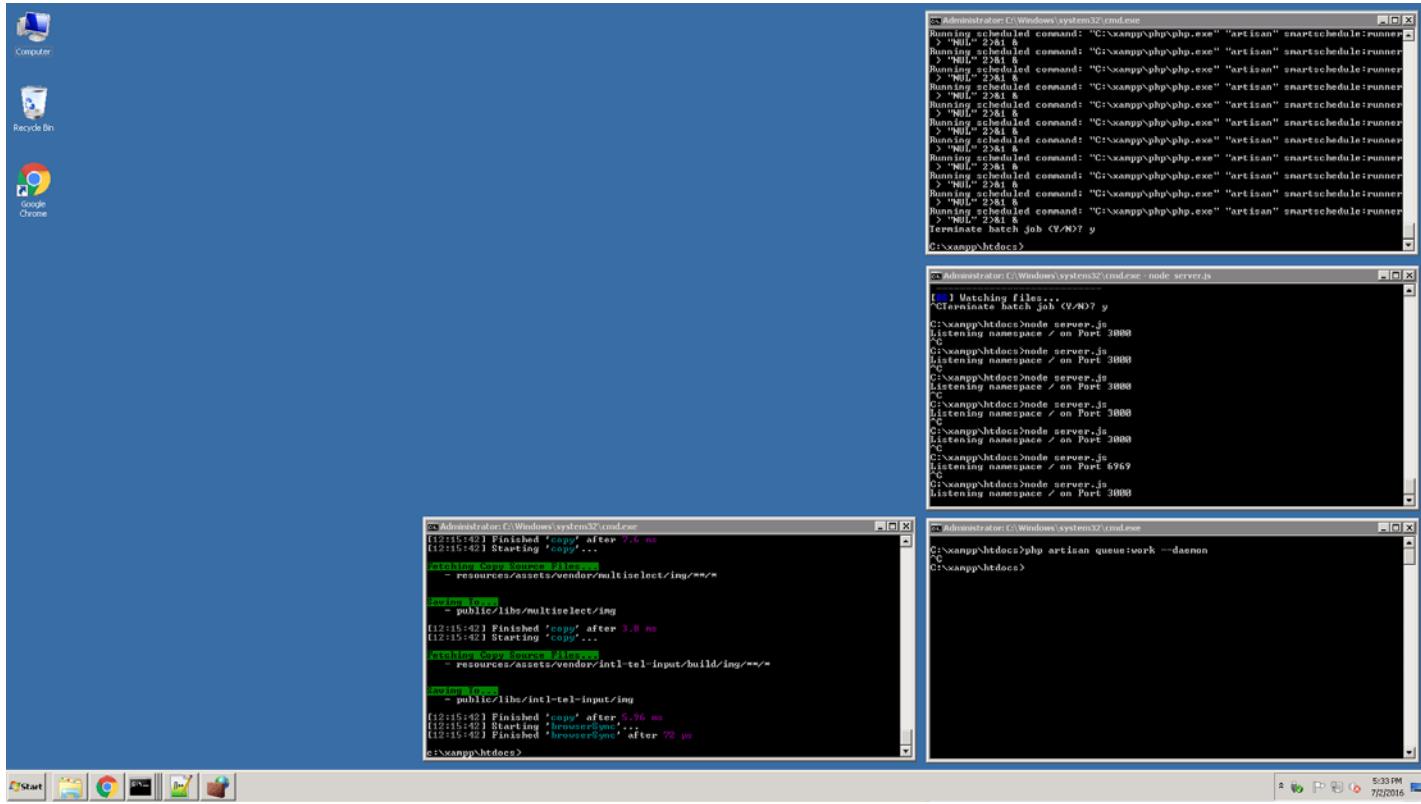
```
php artisan queue:restart
```

Các câu lệnh để reset phần mềm

```
php artisan storage:flush  
php artisan queue:flush  
php artisan view:clear  
php artisan cache:clear  
php artisan config:clear  
redis-cli flushall  
php artisan migrate:refresh
```

Kiểm thử phần mềm

```
php vendor/phpunit/phpunit/phpunit
```



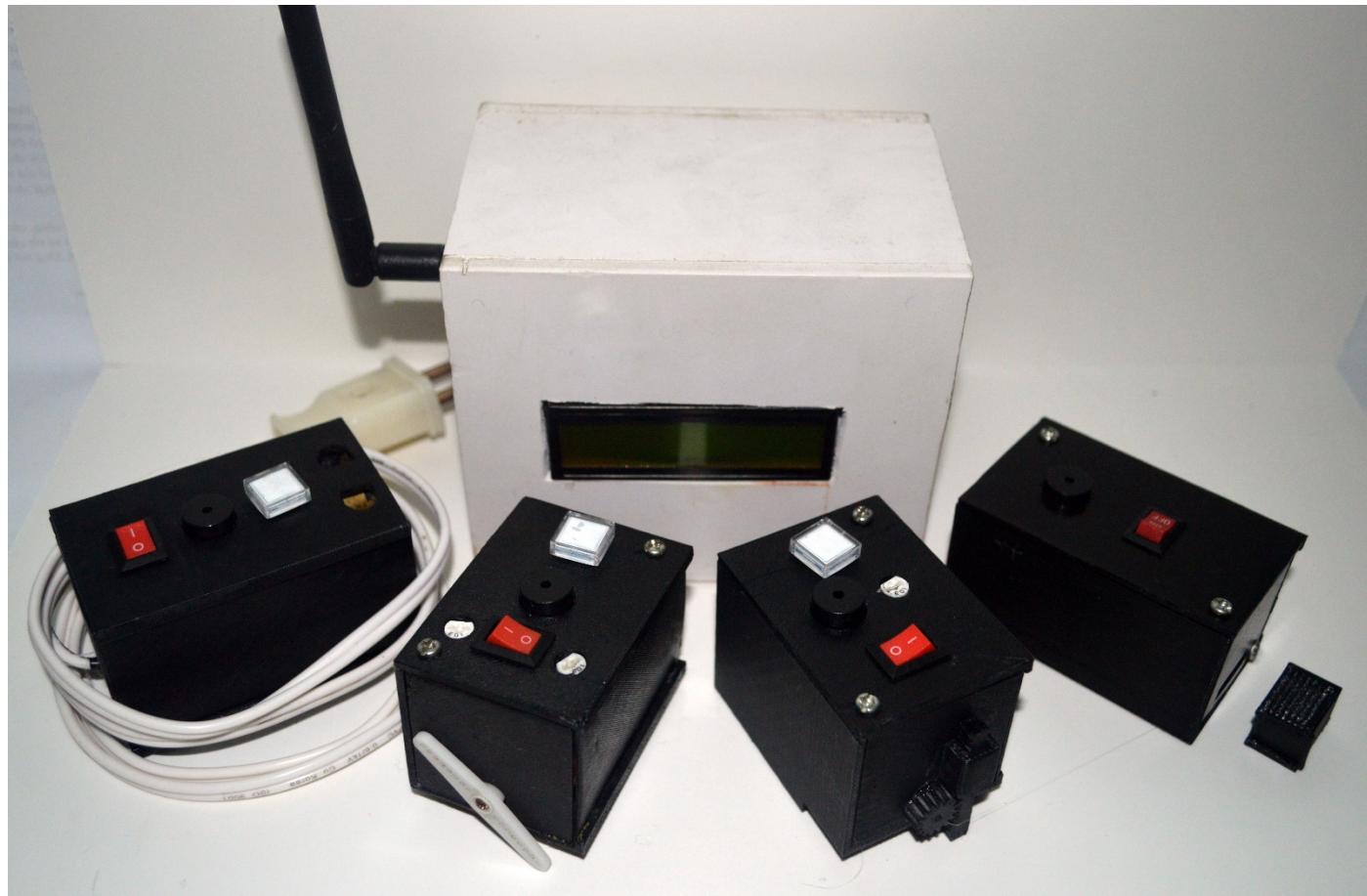
Hình 32. Chạy phần mềm trên máy chủ ảo Windows Server 2008 – Amazon EC2

F. Công nghệ phần cứng

Phần cứng của dự án được lập trình bằng ngôn ngữ Arduino (có thể biết đến như một tập hợp của C/C++), sử dụng Arduino IDE, ngoài ra các bản vẽ 3D sử dụng phần mềm Google SketchUp.

*** Toàn bộ phần code, model 3d, schematics của dự án đều được chia sẻ công khai trên các repository của Smartbots: <https://github.com/smart-bots> ***

Một số hình ảnh của SmarBots (các phiên bản cũ, phiên bản mới được bổ sung ở phần dưới):



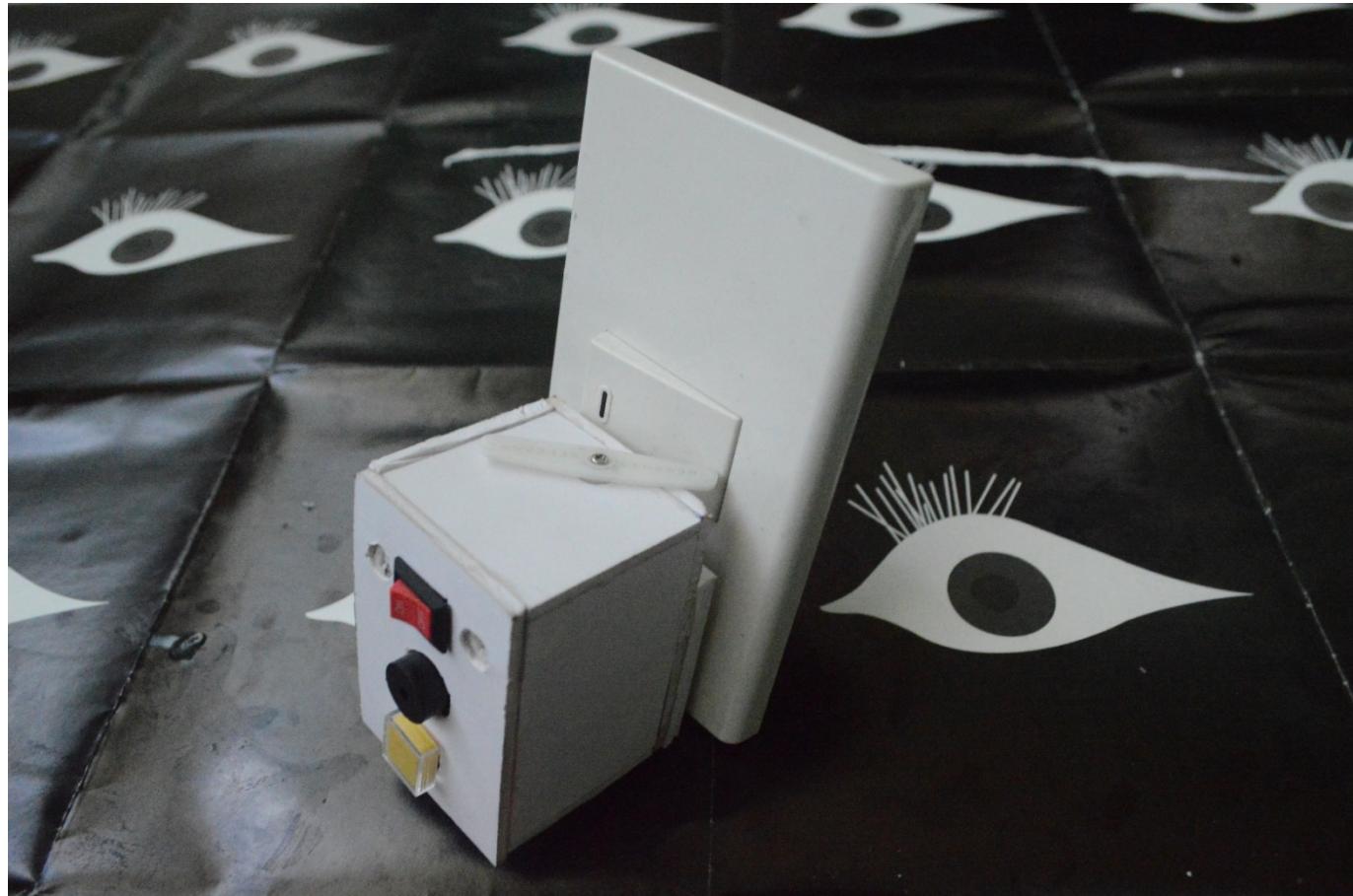
Hình ảnh tổng thể các Module hiện có của SmartBots



SmartHub v1



Flipbot v1



Flipbot v2



Plugbot v1

1. Linh kiện – Module được sử dụng

- Mạch [Arduino Mega 2560](#) : bo mạch xử lý chính
- Module ESP8266 v1 : module kết nối/xử lý wifi
- Module [NRF24Lo1](#) : module truyền/nhận tín hiệu qua RF
- Mạch [NodeMCU Devkit ESP8266 12E](#) mạch tích hợp xử lý/wifi
- Module [NRF24Lo1 PA LA](#) : module truyền/nhận tín hiệu qua RF ở khoảng cách xa
- Mạch [Arduino Nano](#) : Mạch xử lý
- Động cơ [Servo Sg90](#)
- Pin Lithium 18560
- Mobile Booster CNSY : Module tăng áp 3.3->5
- Module TP4056 Lithium Cell Charger : Module sạc pin
- Còi (Buzzer)
- Tụ điện, điện trở, nút nhấn, công tắc,...

[Chỉ dành cho bản pdf/doc/docx] Các linh kiện được gạch chân có hyperlink đến datasheet của nó

2. Thông số kỹ thuật

- **Khoảng cách hoạt động** (Khoảng cách từ Hub đến các Bot): 100m – 200m (Tùy điều kiện vật cản)

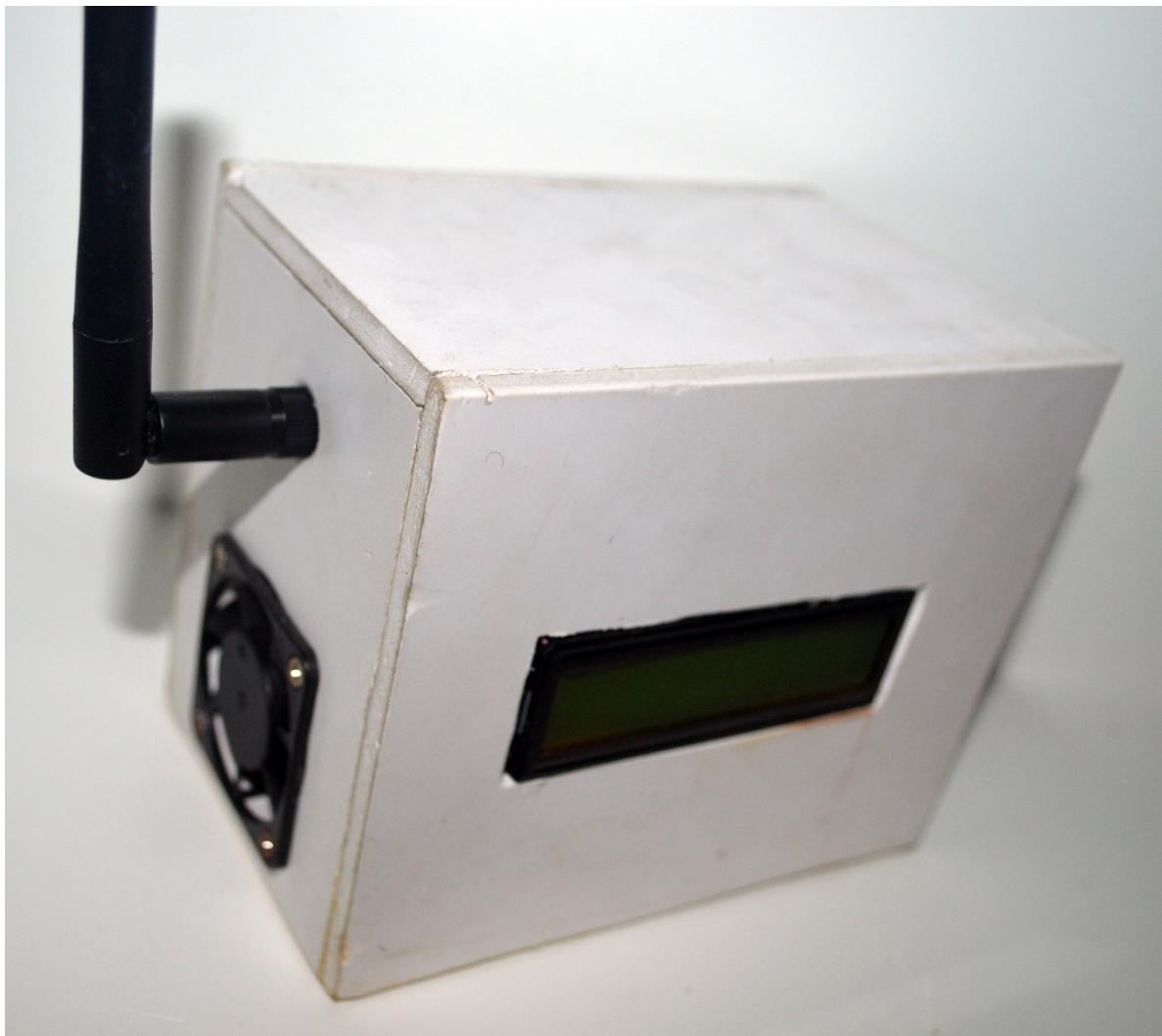
- **Thời lượng pin**: 2 - 3 ngày liên tục

- **Cách nối mạch**: Có thể tìm thấy trên github của project

2.1. Hub

- **Bản thiết kế**: Không có (Do Hub không bị giới hạn kích thước, thiết kế đơn giản nên nhóm không thiết kế trên máy tính – sẽ thiết kế và in 3d trong thời gian tới cho phiên bản SmartHub v3 – Sử dụng Raspberry Pi Pi 3B vượt trội hơn rất nhiều so với phiên bản cũ)

- **Hình ảnh thực**: SmartHub v2

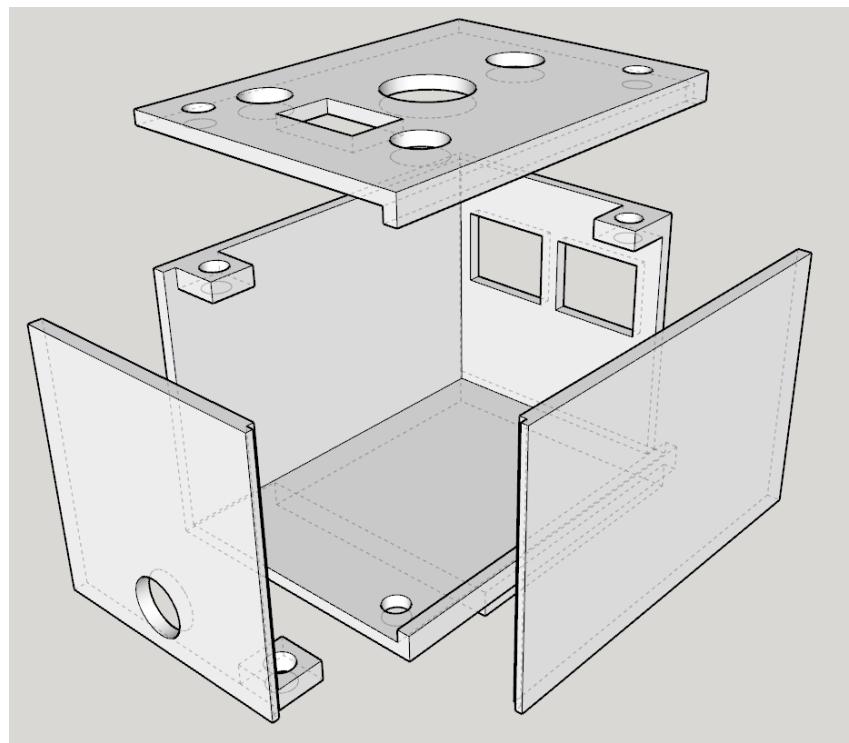


- **Kích thước thực**: 12cm*8cm*10cm

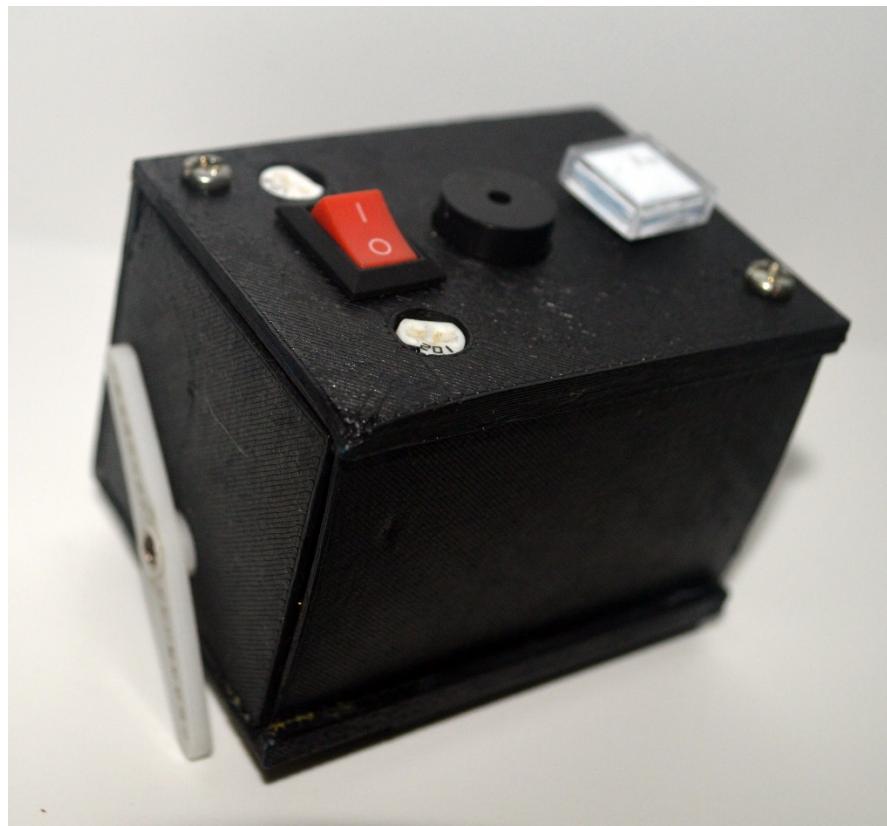
- **Nguồn cấp**: Nguồn USB 5V

2.2. Flip-Bot

- Bản thiết kế:



- Hình ảnh thực: FlipBot v3 (Phiên bản in 3D)

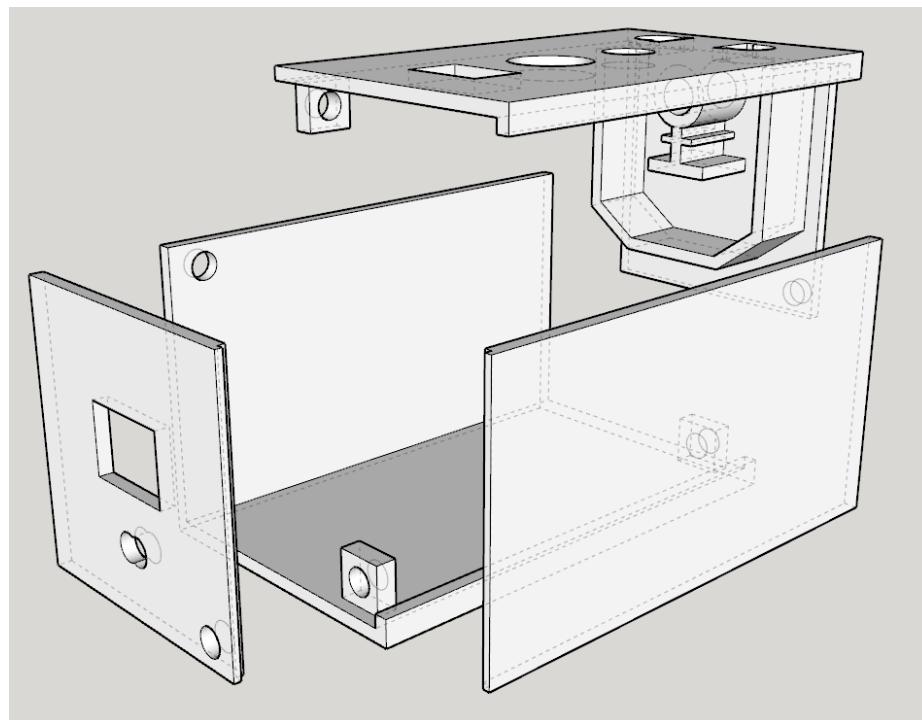


- Kích thước thực: 6cm*4.3cm*4.6cm

- Nguồn sử dụng: Pin đi kèm

2.3. Plug-bot

- Bản thiết kế:



- Hình ảnh thực:

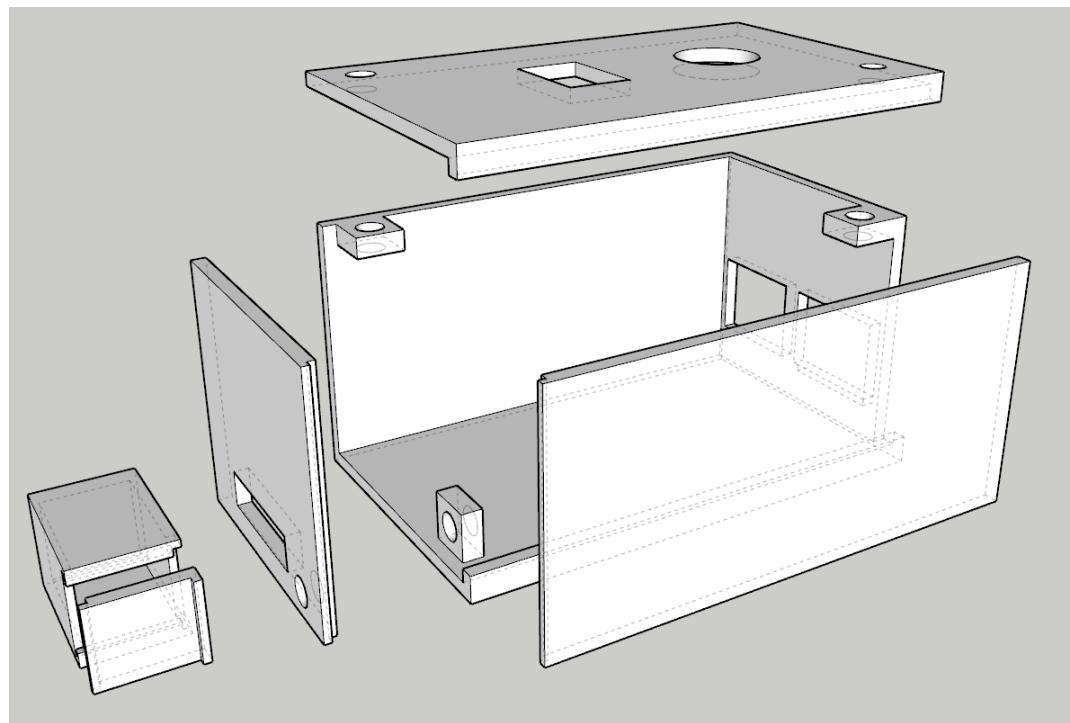


- Kích thước thực: 7.3cm*4.5cm*4.5cm

- Nguồn sử dụng: Điện dân dụng 220V

2.4. Sense-bot: Door

- Bản thiết kế:



- Hình ảnh thực: Sense-Bot: Door v1

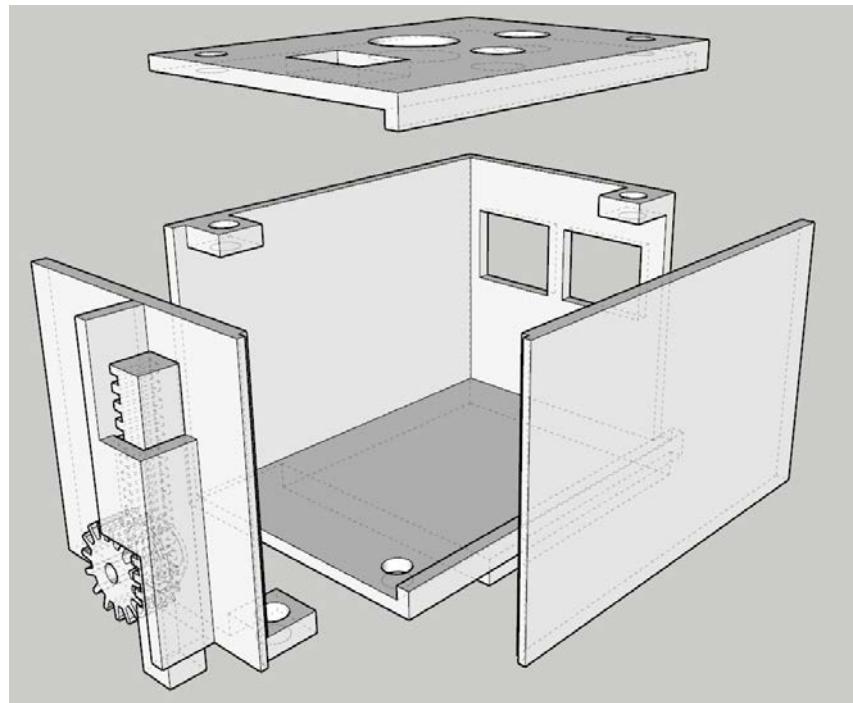


- Kích thước thực: 7cm*4cm*4cm

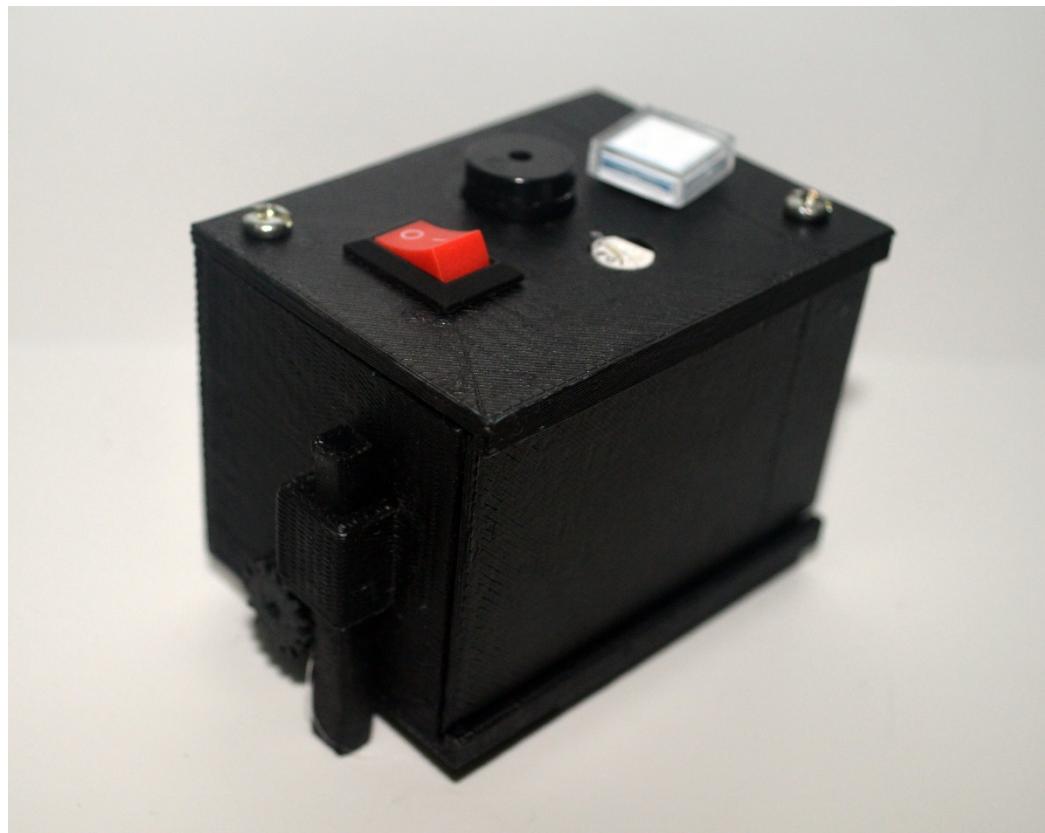
- Nguồn cấp: Pin đi kèm

2.5. Push-bot

- Bản thiết kế:



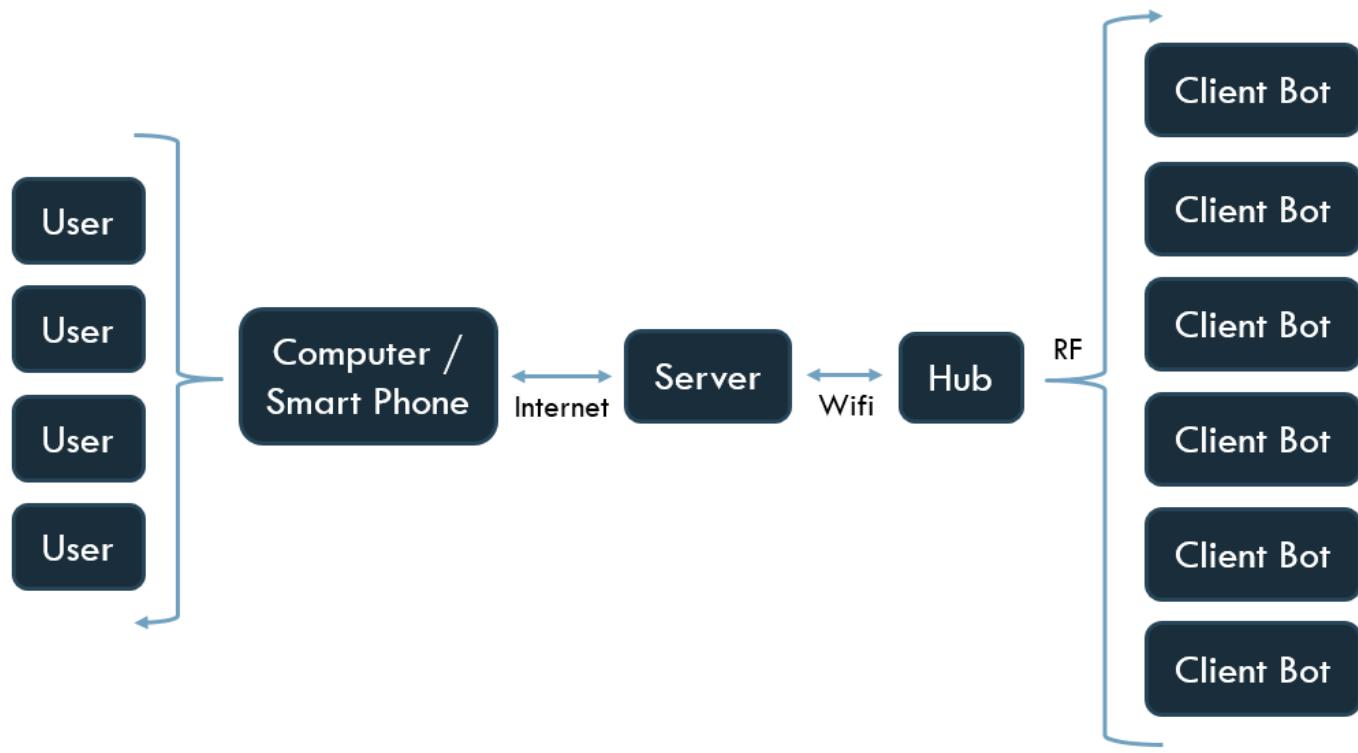
- Hình ảnh thực:



- Kích thước thực: 6.6cm*4.4cm*4.7cm

- Nguồn cấp: Pin đi kèm

3. Nguyên lý hoạt động



Để điều khiển một bot, người dùng sẽ sử dụng điện thoại hoặc máy tính đăng nhập vào server, thực hiện gửi một request đến server thông qua WiFi hoặc LAN, server sẽ xử lý và lưu nó vào dữ liệu, sau đó hub sẽ gửi một request vào Json API của server bằng wifi, lấy những thông tin và người dùng gửi, từ đó xử lý các dữ liệu nhận được và điều khiển từng bot bằng cách gửi qua sóng RF ở một kênh nhất định. Khi bot nhận được tín hiệu từ Hub, nó kiểm tra thông tin được gửi đi, nếu chính xác, bot sẽ thực hiện mệnh lệnh từ Hub và sau đó gửi ngược về Hub một tín hiệu thông báo là lệnh đã thực hiện thành công, tiếp tục Hub sẽ gửi tín hiệu về Server, và server sẽ hiển thị cho người dùng. Trong trường hợp bot không nhận được (do nhiều lý do: hết pin, bot bị tắt,...) thì tín hiệu sẽ không được hồi đáp về Server và người dùng cũng sẽ biết điều đó.

4. Các kỹ thuật được sử dụng

- Tiết kiệm năng lượng: Vì các bot đều sử dụng nguồn năng lượng từ pin (trừ hub, plug-bot và irHub), là nguồn năng lượng nhỏ nhẹ nhưng khuyết điểm là năng lượng cung cấp không phải là vô hạn, nên phải biết cách tận dụng nguồn năng lượng này một cách tốt nhất, tăng thời gian sử dụng của Bot từ khi sạc đầy đến khi hết pin, làm tăng hiệu năng sử dụng và trải nghiệm người dùng.

Để tiết kiệm năng lượng, dự án đã chọn sử dụng RF (2.4GHz) để làm phương thức giao tiếp chính thay vì bluetooth hay ZigBee, phương thức này có ưu điểm là khoảng cách giao tiếp rộng (từ 50m cho đến 100m), sử dụng ít năng lượng, giá thành rất rẻ, ít gặp sự cố và chỉ có một khuyết điểm là tính bảo mật không cao (Sẽ nói ở phần tiếp theo)

Ngoài ra, để đảm bảo sử dụng nguồn năng lượng hợp lý, các Bot luôn được đảm bảo ở trạng thái ngủ sâu (SLEEP_MODE_PWR_DOWN) và chỉ hoạt động trở lại khi nhận được tín hiệu RF, sau khi đã thực hiện hành động, Bot lại trở về trạng thái ngủ. Để thực hiện được điều này, các bot đã sử dụng các thư viện <sleep>, <power>; thư viện giúp đưa bot vào trạng thái ngủ; <interrupt>, <pin change interrupt>, <watchdog timer>; thư viện giúp bot “thức dậy” và làm việc. Nguyên lý hoạt động là nhờ vào cơ chế của module nRF và Arduin: mỗi khi module nRF nhận được tín hiệu từ đúng kênh mà nó đang “nghe”, nó sẽ kích hiệu điện thế lên ở một chân irq, chân irq này sẽ được nối với một pin có hỗ trợ interrupt của arduino: sử dụng “interrupt” hoặc “pin change interrupt” để khai báo chân irq, sử dụng “sleep” và “power” đưa bot vào trạng thái ngủ sâu, sau khi đã khai báo pin, mỗi khi có điện áp được kích vào pin đó, arduino sẽ tự động “thức” dậy và hoạt động bình thường.

- Bảo mật thông tin: Như đã nói ở trên, sử dụng RF có khuyết điểm là tính bảo mật không cao, vì nó giao tiếp thông qua các kênh tần số, nên các tín hiệu được gửi sẽ có thể được nhận bởi tất cả mọi thiết bị đọc RF (không giống như bluetooth hay zigbee – giao tiếp 1-1). Vấn đề này sẽ được giải quyết bằng cách mã hóa thông tin được truyền đi bằng chuẩn mã hóa Rijndael – chỉ có thể giải mã khi biết được “secret key”. Như đã biết thì mỗi bot đều đi kèm với một token gồm 10 ký tự tạo ra ngẫu nhiên (riêng với hub là 50

ký tự) – chúng chính là những “secret key” được nói ở trên. Thế nên cho dù các thiết bị khác có bắt được thông tin của bot và hub thì cũng không thể nào giải mã được chúng. Dùng cách này mặc dù sẽ hơi “mệt” cho các bot và hub vì phải giải mã, mã hóa liên tục các tín hiệu (còn có thể làm tăng thời gian phản hồi) nhưng đã giải quyết được vấn đề bảo mật.

5. Thư viện hỗ trợ - tài liệu tham khảo

- Thư viện Esp8266 (<https://github.com/esp8266/Arduino>) hỗ trợ lập trình cho esp8266
- Thư viện ArduinoJson (<https://github.com/bblanchon/ArduinoJson>) hỗ trợ parse Json để xử lý
- Thư viện LiquidCrystal_I2C (<https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>) hỗ trợ điều khiển LCD qua giao thức i2c
- Thư viện RF24 (<https://github.com/TMRh20/RF24>) hỗ trợ sử dụng Nrf24l01

Một số thư viện đi kèm với Arduino:

- Thư viện EEPROM (<https://www.arduino.cc/en/Reference/EEPROM>) hỗ trợ sử dụng bộ nhớ EEPROM
- Thư viện Wire (<https://www.arduino.cc/en/Reference/Wire>) hỗ trợ sử giao tiếp qua chuẩn I2C
- Thư viện Servo (<https://www.arduino.cc/en/Reference/Servo>) hỗ trợ điều khiển động cơ Servo
- Thư viện AVR Power ([avr/power.h](#)) và AVR Sleep ([avr/sleep.h](#)) hỗ trợ điều khiển nguồn, đưa bot vào trạng thái ngủ
- Thư viện AVR Interrupt ([avr/interrupt.h](#)) hỗ trợ sử dụng các interrupt

Tài liệu được tham khảo:

<http://arduino.cc>

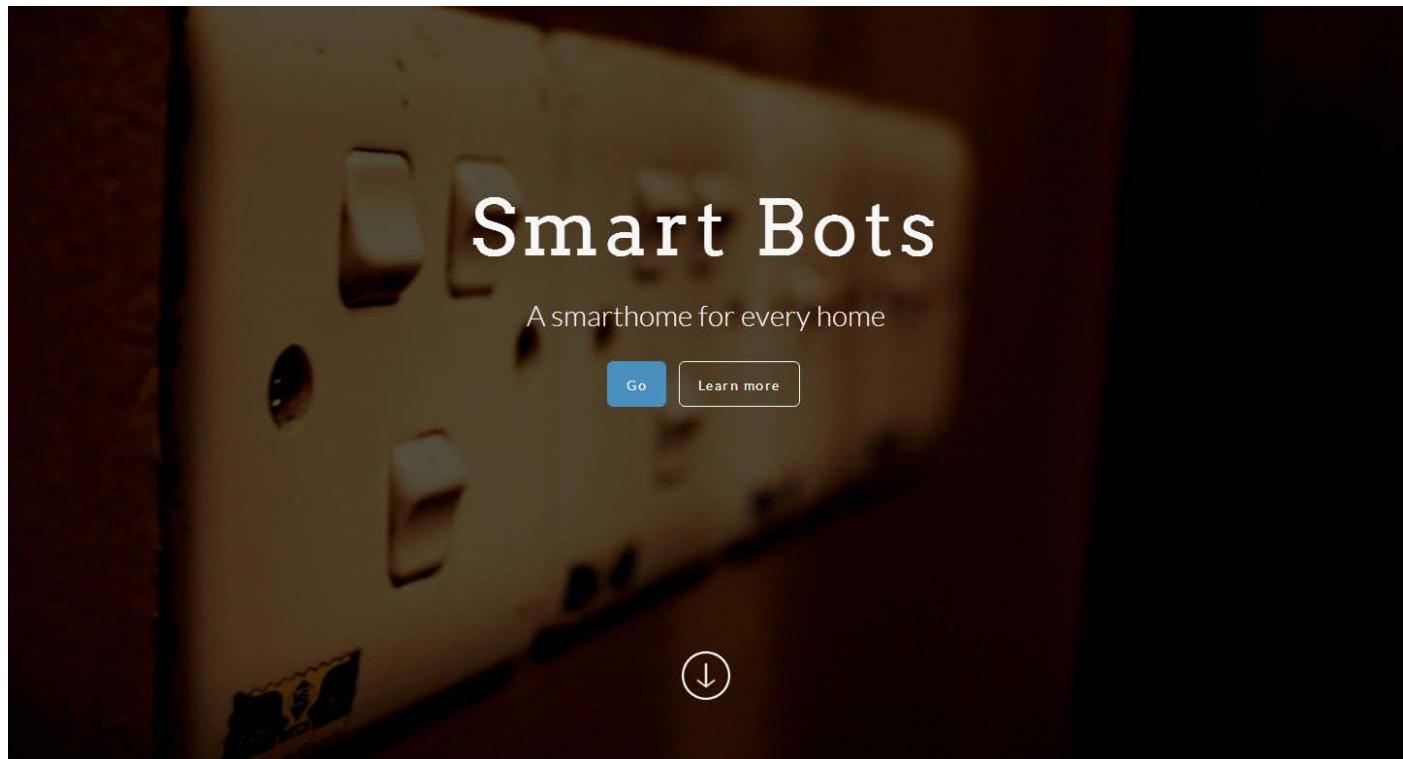
<http://arduino.vn>

<http://learn.adafruit.com>

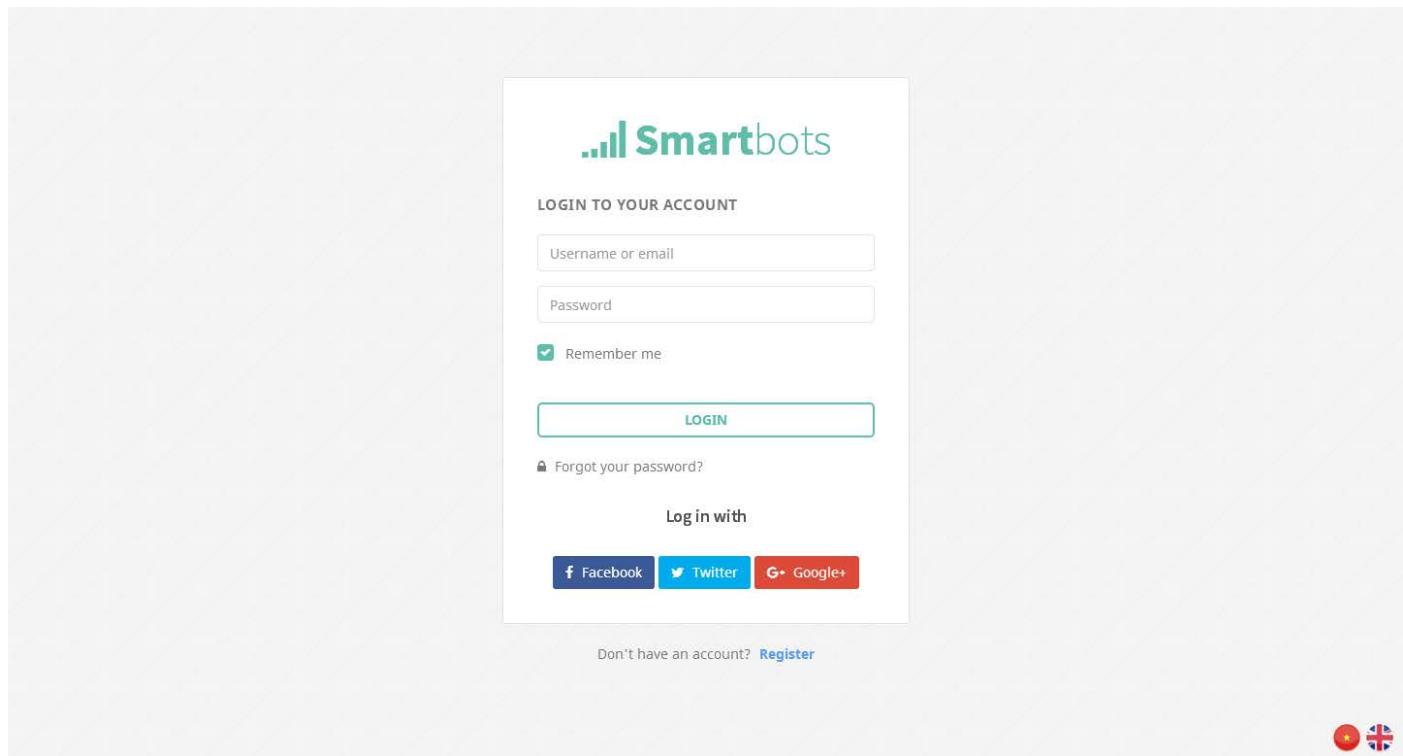
<http://learn.sparkfun.com>

G. Hướng dẫn sử dụng phần mềm

Truy cập vào phần mềm đã cài đặt ở local hoặc <https://smart-bots.xyz>

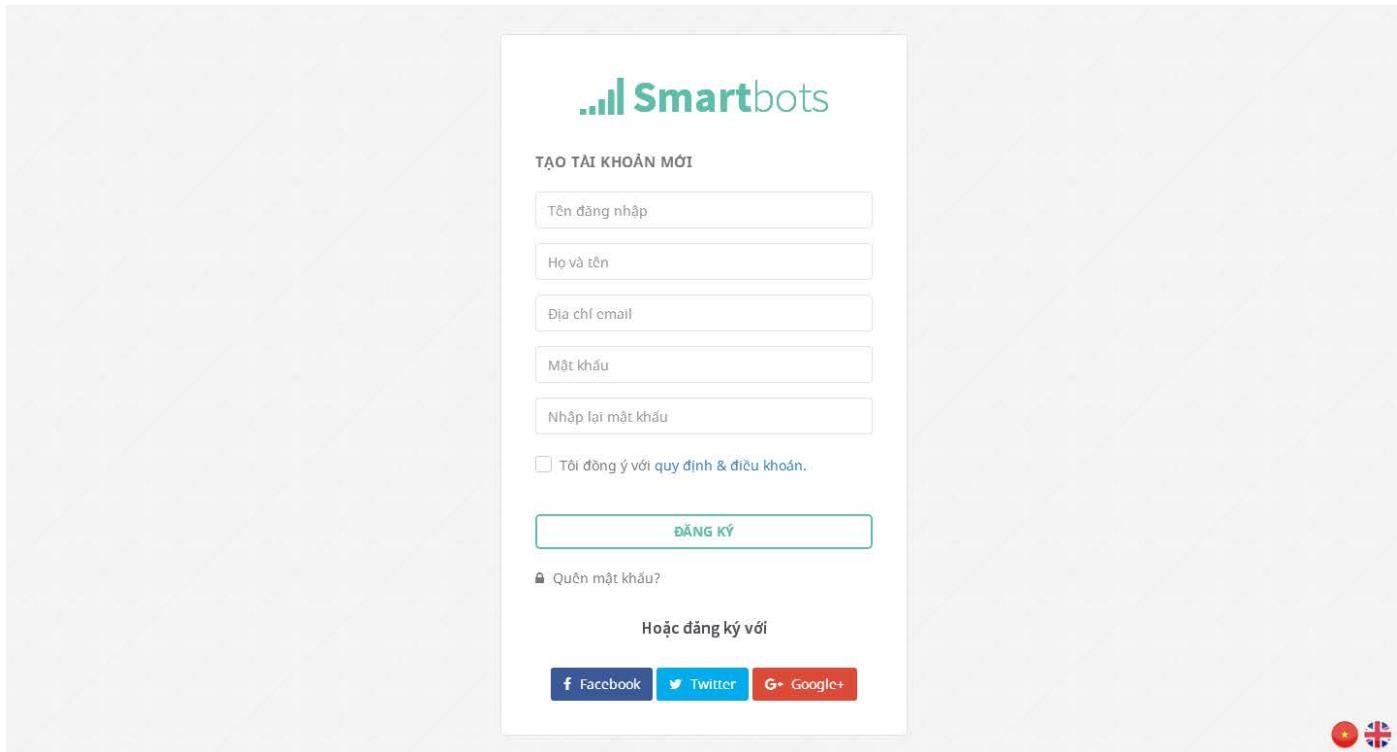


Ở trang chủ phần mềm, nhấn “Go” để tiếp tục. Người dùng sẽ được chuyển tới trang đăng nhập (cần phải có tài khoản để sử dụng phần mềm)



Phần mềm hiện đang hỗ trợ 2 ngôn ngữ là tiếng anh và tiếng việt, người dùng có thể chuyển đổi dễ dàng bằng 2 nút ở góc dưới phía bên phải màn hình.

Nếu người dùng chưa có tài khoản thì có thể “Đăng ký” bằng cách click vào link bên dưới.



Quá trình đăng ký và đăng nhập (và quên mật khẩu,...) khá đơn giản nên không cần hướng dẫn.

Người dùng đăng ký và đăng nhập thành công sẽ được chuyển tới trang chọn “Hub”, ở đây người dùng có thể đăng nhập vào hub, hoặc tạo hub mới

The screenshot shows the 'Hub login' section of the SmartBots platform. At the top, there is a search bar and a user profile icon labeled 'h2dvnnet'. Below the header, the page title 'Hub login' is displayed next to a back arrow. A breadcrumb navigation shows 'SmartBots / Hub / Login'. A message 'Choose your hub to start your session' is centered above a hub card. The hub card features a circular image of a yellow house with blue shutters, labeled 'My sweet home' and '6 bots'. To the right of the hub card is a green button with a plus sign and the text 'New hub'. In the bottom right corner of the main content area, there are small icons for the Vietnamese flag and the United Kingdom flag.

Để tạo hub mới, người dùng click vào nút “New hub” và điền các thông tin của hub như: tên (có thể là tên địa điểm hub được đặt,...) , mô tả cho hub (không bắt buộc), hình ảnh mô tả cho hub (không bắt buộc) và timezone hiện tại của hub (bắt buộc)

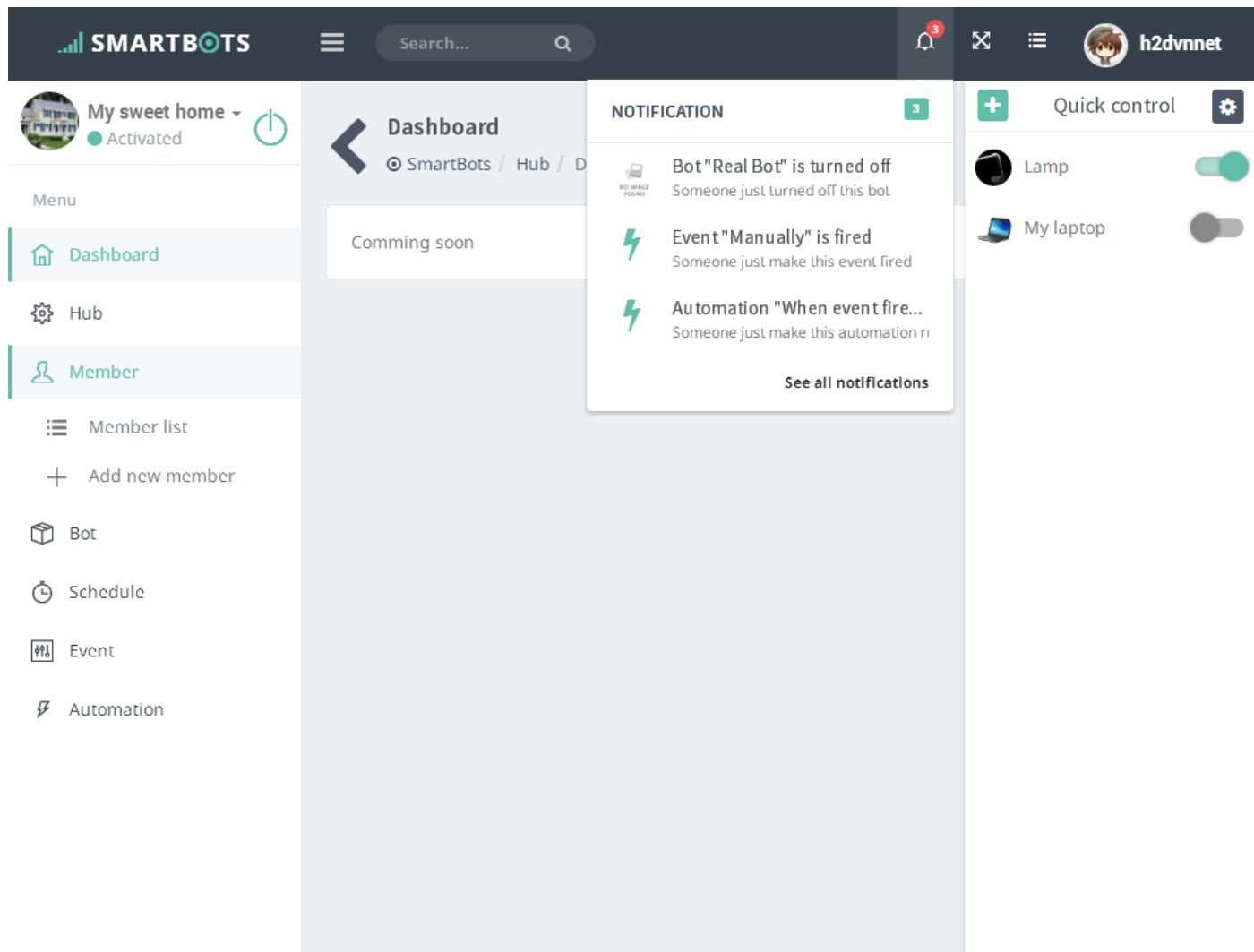
The screenshot shows a web-based application interface for creating a new hub. At the top, there is a search bar with placeholder text 'Search...' and a magnifying glass icon. To the right of the search bar is a user profile icon with the handle 'h2dvnnet' and a small 'X' button. Below the header, the page title 'Create new hub' is displayed next to a back arrow icon. The breadcrumb navigation shows the path: SmartBots / Hub / Create.

The main form is titled 'CREATE NEW HUB'. It contains several input fields:

- Name:** A text input field.
- Description:** A large text area for entering a detailed description.
- Image:** A placeholder area with dashed lines labeled 'Drop/Click' for uploading an image.
- Timezone:** A dropdown menu set to '(UTC-11:00) Midway Island'. To the right of the dropdown are two small circular icons, one with the Vietnamese flag and one with the British flag.

Sau khi tạo hub thành công có thể đăng nhập vào hub bằng cách click vào hình hoặc tên của hub.

Người dùng sẽ được chuyển tới trang Dashboard (hiện tại đang trong giai đoạn phát triển)



Tại đây người dùng có thể thấy được giao diện chính của phần mềm gồm:

- + Thanh menu bên trái, gồm thông tin sơ lượt về hub hiện hành, các liên kết tới các trang quản lí trong hub.
- + Thanh Navbar trên cùng: gồm logo SmartBots, thanh tìm kiếm (đang phát triển), cửa sổ thông báo (notification), cửa sổ người dùng (với avatar và tên đăng nhập người dùng hiện hành)
- + Thanh quick control bên phải: hiện thị các bot mà người dùng thêm vào chức năng điều khiển nhanh

Tiếp theo là trang “Hub”, trang này hiển thị tất cả các thông tin của hub hiện hành, cho phép người dùng chỉnh sửa thông tin (nếu người dùng có quyền)

The screenshot shows the configuration interface for a hub named "My sweet home". The hub token is displayed as a long string of characters. A note below the token says, "This token is quite important, so dont let it fall into the wrong hands". The hub's description is "I love my home <3". The timezone is set to "(UTC+07:00) Hanoi". At the bottom, there are buttons for "Save", "Deactivate", and "Delete".

Trong đó có một thông tin rất quan trọng chính là “Token”, token này cần trong quá trình cài đặt và sử dụng phần cứng. Người dùng có thể chỉnh sửa những thông tin này, sau đó nhấn Save, hoặc có thể xóa hub, hay hủy kích hoạt hub tạm thời,...

Tiếp theo là trang “Member” cho phép người dùng có quyền xem, chỉnh sửa những thành viên hiện tại của hub, hoặc thêm thành viên cho hub.

The screenshot shows the 'Hub members' section of the SmartBots interface. On the left, there's a sidebar with a 'My sweet home' profile picture and an 'Activated' status. The 'Member' option in the sidebar is highlighted. The main area shows two member profiles: 'Marion Larkin' with 3 bots and 'Huynh Duc Duy' with 6 bots. A green 'Add member' button is located in the top right of the main content area.

Người dùng có thêm thành viên mới vào hub:

The screenshot shows the 'Add new member' page. The sidebar has 'Member' selected. The main area has 'Add new member' highlighted. It shows a list of users under 'Bot low permissions' and a list of bot names under 'Bot high permissions'. At the bottom right, there are flags for Vietnam and the United Kingdom.

Bot low permissions	Bot high permissions
jnienow	Search...
jnienow Ms. Megane Murazik	Real Bot
jast.elroy Hilda Collins	Yeah
Yeah	The whole things
The whole things	My fan
My fan	My laptop
My laptop	Lamp
Lamp	

Nhập username của thành viên cần thêm và chọn nó ở dưới khay autocomplete, sau đó thiết lập các quyền cho thành viên: quyền cho bot, quyền cho schedule, quyền cho event,... hoặc quyền tối cao:

	Add/Create	View/Control	Edit/Delete
Hub	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Bots	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Schedules	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Events	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Automations	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Members	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Tiếp theo là trang “Bot” cho phép người dùng xem danh sách các bot, trạng thái thời gian thực của nó (tự động điều chỉnh)

The screenshot shows the SmartBots web interface. The top navigation bar includes the SmartBots logo, a search bar, and user profile information (h2dvnnet). The left sidebar menu is open, showing options like Dashboard, Hub, Member, Bot (which is selected and highlighted in green), Add new bot, Schedule, Event, and Automation. The main content area is titled "Hub bots" and shows a "BOTS LIST" with five entries:

- Real Bot: NO IMAGE FOUND (Status: Activated)
- Yeah: Yeah (Status: Activated)
- The whole things: The whole things (Status: Deactivated)
- My fan: My fan (Status: Deactivated)
- My laptop: My laptop (Status: Deactivated)

A "Lamp" bot is also listed below the main list, with its status switch set to "On".

Người dùng có thể thêm bot mới vào hub:

The screenshot shows the SmartBots platform interface. On the left, there is a sidebar with the following menu items:

- Dashboard
- Hub
- Member
- Bot** (highlighted)
- Bot list
- Add new bot
- Schedule
- Event
- Automation

The main area is titled "Add new bot". It contains the following fields:

- Name: An input field.
- Description: A large text area.
- Image: A dashed rectangular area with the placeholder text "Drop/Click".
- Type: A dropdown menu set to "Flip-bot".
- Token: An input field with a "Verify" button next to it.
- Get notify: A toggle switch.
- Flags: Buttons for Vietnamese and English.

Người dùng sẽ phải nhập các thông tin của bot như tên, mô tả, hình ảnh minh họa, loại của bot (các loại module như đã nói ở phần đầu), token của bot (mỗi bot khi được sản xuất sẽ mang token riêng gồm 10 ký tự bất kỳ, được in trên vỏ của bot – không trùng lặp – không thể thay đổi). Sau đó là các thiết lập như: có nhận thông báo khi bot hoạt động, hoặc là phân quyền cho các thành viên các với bot này,...

“Schedule” đơn giản là hiển thị danh sách các lịch để người dùng quản lý, có thể thêm lịch hoạt động mới:

The screenshot displays the SmartBots platform's interface for managing schedules. On the left, a sidebar lists various sections: Dashboard, Hub, Member, Bot, Schedule (which is currently selected and highlighted in green), Schedule list, Add new schedule, Event, and Automation. The main content area is titled "Schedule" and contains the following form fields:

- Name:** A text input field.
- Description:** A large text area for additional notes.
- Action:** A dropdown menu set to "Toggle".
- Type:** A dropdown menu set to "One time".
- Time:** An empty text input field for specifying the time.
- Condition:** A dropdown menu set to "Off".
- Get notify:** A toggle switch that is currently off.
- Low permissions:** A section containing two search input fields and a list of users: "Search..." (with "kian.pagac" listed), "Search..." (empty), and a small icon with the Vietnamese flag and the text "h2dvnnet".
- Add action:** A green button with a plus sign.

Người dùng sẽ nhập các thông tin cơ bản, sau đó đến “Action” – hành động mà lịch sẽ thực hiện. Có 3 loại hành động cơ bản với bot, đó là “Toggle” thay đổi giữa bật/tắt, hoặc bật hay tắt một bot nào đó. “Type” là kiểu của lịch, có 2 kiểu là “One time” hoạt động một lần duy nhất, nếu chọn kiểu này, người dùng sẽ phải chọn một mốc thời gian cho nó.

Kiểu thứ 2 là “Repeat”: lịch sẽ lặp đi lặp lại theo một quy tắc nào đó. Nếu chọn kiểu này, người dùng sẽ phải đặt cho nó những tần số hoạt động. Hiện tại hỗ trợ những tần số:

Frequency	Value	Unit	Action
Every	2	minute(s)	▼
Every	4	hour(s)	▼ At
Every	3	day(s)	▼ At
Every	5	month(s)	▼ At
Every	1	year(s)	▼ At

+ Add frequency

Người dùng cũng có thể chọn nhiều tần số.

Tiếp đến là tạo “Event”, event hoạt động rất đơn giản, có thể kích hoạt event bằng tay, hoặc event tự kích hoạt khi có sense-bot nhận dữ liệu, hoặc một bot nào đó thay đổi. Mỗi khi event được kích hoạt, nó sẽ kéo theo một “Automation” nào đó (hoặc là không) được người dùng định nghĩa. Mục đích chủ yếu của event là giúp người dùng có cái nhìn trực quan hơn mỗi khi có thay đổi nào đó.

The screenshot shows the SmartBots platform interface. On the left, there is a sidebar with the following menu items:

- Dashboard
- Hub
- Member
- Bot
- Schedule
- Event** (highlighted)
- Event list
- + Add new event
- Automation

The main content area is titled "Create new event" and shows the "CREATE NEW EVENT" form. The form fields include:

- Name: A text input field.
- Action: A dropdown menu set to "Toggle".
- Get notify: A toggle switch that is off.
- Low permissions: A section containing two search input fields and a list of users: "kian.pagac" and "h2dvnet".
- High permissions: A section containing two search input fields and a list of users: "kian.pagac" and "h2dvnet".
- Users can view/control this bot: A section with two search input fields and a list of users: "kian.pagac" and "h2dvnet".

At the bottom right of the main content area, there are two small flags: the Vietnamese flag and the British flag.

Tạo event khá đơn giản, chỉ cần nhập tên và hành động mà nó sẽ được kích hoạt (hoặc là để trống để kích hoạt bằng tay)

Tiếp theo là “Automation”, automation cũng giống như là schedule, là một chuỗi các hành động, nhưng khác ở chỗ được kích hoạt bằng event, hoặc bằng một hành động trước đó chứ không phải kích hoạt theo thời gian, tần số. Như đã nhắc ở trên, “event” cũng là một trigger của automation.

The screenshot displays the SmartBots application interface. On the left, a sidebar menu lists various features: Dashboard, Hub, Member, Bot, Schedule, Event, Automation (which is currently selected and highlighted in green), Automation list, and Add new automation. The main content area is titled "CREATE NEW AUTOMATION". It contains several input fields and dropdown menus:

- Name:** An empty text input field.
- Description:** A large text area for describing the automation.
- Trigger:** A dropdown menu set to "Bot turned".
- Action:** A dropdown menu set to "Toggle".
- Condition:** A dropdown menu set to "Off".
- Get notify:** A toggle switch that is currently off.
- Low permissions:** A section containing two search input fields labeled "Search..." and a list of users: "kian.pagac" and "h2dvnnet".
- Add action:** A green button with a plus sign and the text "Add action".

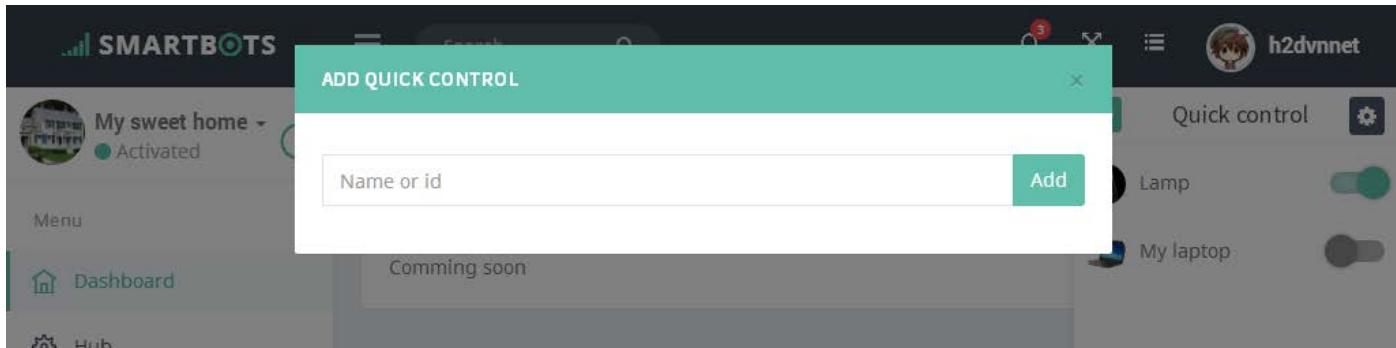
Giống như Schedule, người dùng cũng có thể thêm điều kiện hoạt động cho automation (ví dụ như chỉ hoạt động như một bot nào đó tắt,...)

Một chức năng nữa rất hay và quan trọng đó chính là “Quick control”

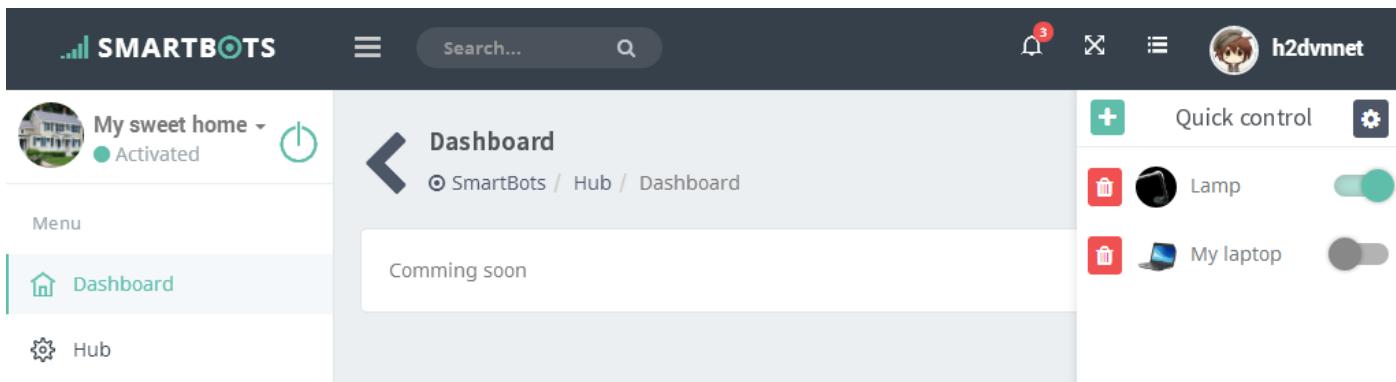
The screenshot shows the SmartBots interface. On the left is a sidebar with icons for Menu, Dashboard, Hub (which is selected), Member, Bot, Schedule, Event, and Automation. The main area is titled "Hub edit" under "SmartBots / Hub / Edit". It shows "EDIT HUB'S SETTINGS" with fields for Name ("My sweet home"), Token (a long string of characters), and an Image of a yellow house with blue shutters. Below the image is a "Description" field containing "I love my home <3". On the right, there's a "Quick control" section with a "+" button, a "Lamp" icon (green switch), and a "My laptop" icon (grey switch).

Như đã giới thiệu từ đầu, quick control giúp người dùng điều khiển các bot từ bất cứ đâu. Có thể thấy, thanh quick control luôn xuất hiện ở bất cứ trang nào của người dùng.

Để thêm bot vào Quick control, người dùng nhấn vào dấu “+”, sau đó, một cửa sổ mới hiện ra, người dùng nhập tên của bot cần thêm rồi chọn ở phía dưới, sau đó submit.



Để xóa quick control, người dùng ấn vào dấu hình bánh răng. Sau đó các nút mới (có hình thùng rác sẽ hiện ra), click vào những nút đó sẽ xóa bot khỏi thanh quick control.



H. Hướng dẫn cài đặt, sử dụng phần cứng

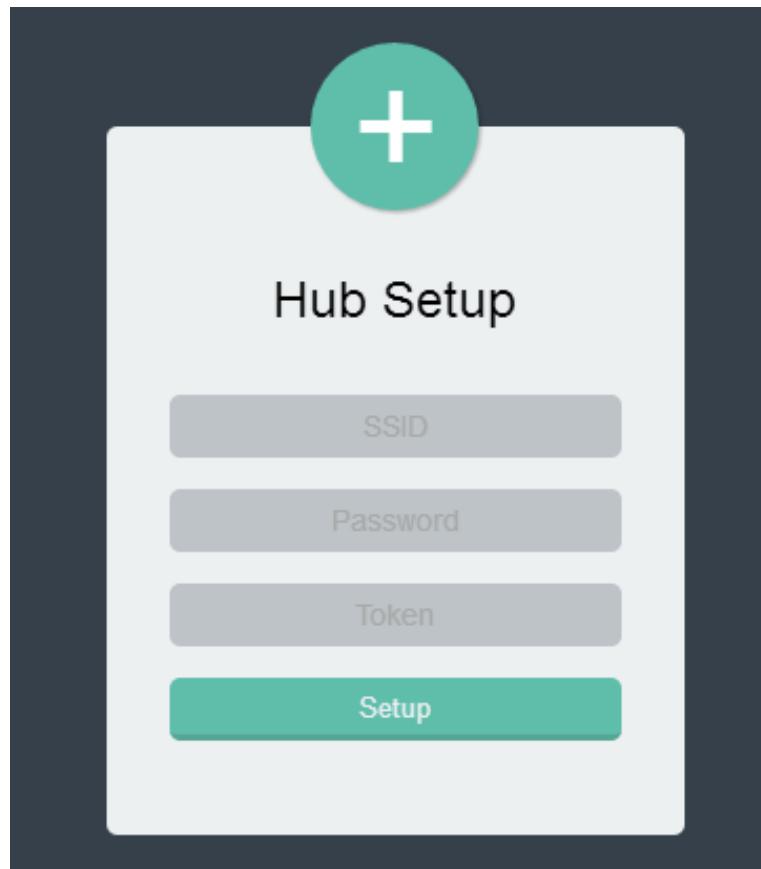
Đầu tiên người dùng phải cấp nguồn cho Hub (qua adapter đi kèm), gắn các bot cần sử dụng lên đúng vị trí, sau đó bật công tắc cho bot.

Ở lần sử dụng đầu tiên, Hub yêu cầu người dùng phải cài đặt các thông tin cơ bản để hub có thể hoạt động: Tên, mật khẩu của wifi, token của hub (được lấy ở phần mềm).

Người dùng có thể thấy, trên màn hình của hub sẽ hiện lên những thông tin: Tên, mật khẩu của wifi (phân cách bởi dấu |), địa chỉ IP của hub trong wifi đó.



Người dùng sẽ kết nối vào wifi với thông tin đó, truy cập vào địa chỉ IP. Giao diện cài đặt sẽ hiện ra cho người dùng, người dùng sẽ điền các thông tin cho hub và xác nhận. Sau đó hub sẽ tự động khởi động lại và sử dụng các thông tin vừa được cài đặt.



Trong trường hợp muốn thay đổi cài đặt, hoặc xảy ra sai sót khi cài đặt hoặc hoạt động, Hub cung cấp cho người dùng một nút nhấn “Reset” ở đằng sau, khi nhấn nút, Hub sẽ tiến hành reset tất cả thông tin và người dùng có thể cài đặt lại như lần đầu tiên.

Trong quá trình hoạt động, Hub mặc định sẽ tắt màn hình để tiết kiệm điện, nhưng khi có lỗi xảy ra, màn hình sẽ sáng lên cùng với lỗi để người dùng dễ nhận biết và sửa chữa.

I. Tự đánh giá – kết luận

Sau một thời gian tìm hiểu và nghiên cứu, nhóm đã bước đầu thực hiện được mục tiêu ban đầu của mình là tạo ra một mô hình nhà thông minh mới hơn và phù hợp hơn với người dùng Việt Nam: chi phí thấp, dễ lắp đặt, sử dụng cùng với khả năng tích hợp và tùy biến cao, hoàn toàn có thể thay thế được các giải pháp nhà thông minh hiện tại đã có. Đáng chú ý là tính tùy biến, ứng dụng cao, có khả năng thương mại hóa, dễ dàng phát triển thành hệ thống an ninh, quan trắc môi trường giá rẻ,...

Hiện tại vì trình độ, thời gian và nguồn lực còn hạn chế nên nhóm chỉ mới thực hiện được một phần nhỏ trong kế hoạch đã đề ra lúc đầu, vẫn còn chứa nhiều khuyết điểm cần khắc phục như: thiết kế chưa tối ưu, chưa đáp ứng được nhu cầu thẩm mỹ (một điểm cũng khá quan trọng của sản phẩm), hiệu xuất làm việc chưa cao,....

J. Hướng phát triển trong tương lai

Nhóm đã có một kế hoạch rõ ràng cho những việc cần làm trong tương lai, đó là tối ưu hóa lại dự án, loại bỏ những khuyết điểm, hạn chế, phát triển thêm nhiều chức năng cần thiết cho người dùng, tăng trải nghiệm và sự thoả mái khi sử dụng cho người dùng. Cụ thể:

- Các module đang được nghiên cứu sẽ xuất hiện trong thời gian tới:

- Fire-sense-bot: cảm biến lửa
- Motion-sense-bot: cảm biến chuyển động
- Temperature-sense-bot: cảm biến nhiệt độ
- Humidity-sense-bot: cảm biến độ ẩm
- Twist-bot
- IrHub
- SmartHub V3

- Hiện tại song song với việc nâng cấp phiên bản cũ nhóm đang phát triển phiên bản 2 của SmartBots Server mới hơn, tốt hơn, chia phần mềm làm 3 phần riêng biệt: RESTful Service, Client (Web – Native) và Hardware Service
 - o RESTful Service
 - o Sử dụng server Linux (Ubuntu) thay cho Windows
 - o Cơ sở dữ liệu sử dụng MongoDB thay cho MariaDB
 - o Sử dụng JWT (Json Web Token) thay cho Session – Cookie thông thường
 - o Web Client: Chỉ sử dụng AngularJS và AngularMaterial -> tăng tối đa tốc độ, trải nghiệm người dùng
 - o Native Client: Sử dụng React kết hợp Redus
 - o Hardware Service: Sử dụng MQTT (một phương thức giao tiếp tốt hơn) – Mosquitto

- Em mong nhà trường sẽ tạo điều kiện thêm cho em để em có đủ thời gian và nguồn lực để tiếp tục phát triển dự án hoàn thiện hơn.