|  |
| --- |
| Garage RPI |
| Projet remis à Denis Drolet |
| Documentation concernant les méthodes de travail ainsi que mode d’emplois |

09/03/2016

Contenu

[Logiciels utilisés 2](#_Toc445486743)

[Matériels utilisés 2](#_Toc445486744)

[Contraintes 2](#_Toc445486745)

[Page de connexion 3](#_Toc445486746)

[Contrôle des portes 4](#_Toc445486747)

[Page administration 5](#_Toc445486748)

[Procédure d’installation 6](#_Toc445486749)

[Mise en place du site web 8](#_Toc445486750)

## Logiciels utilisés

* HTML
* JAVASCRIPT
* PHP
* PYTHON
* SQLITE
* SSMTP
* Apache2

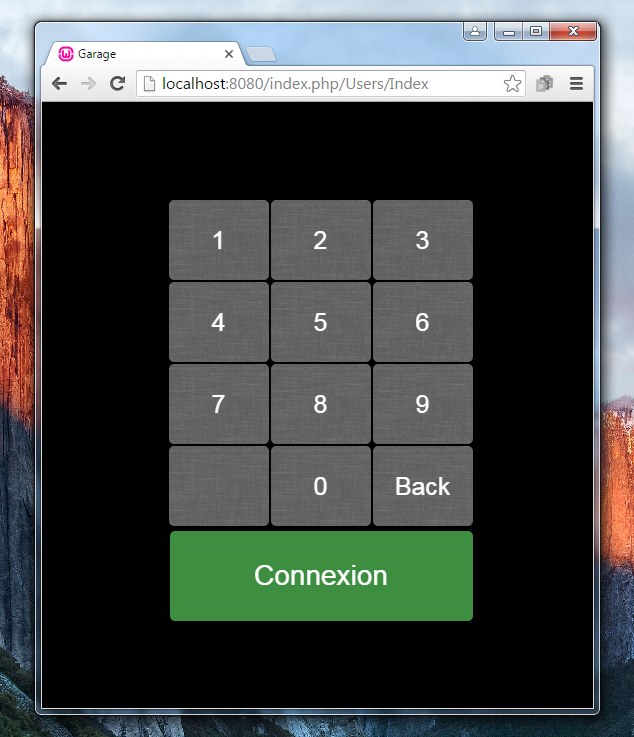
## Matériels utilisés

* RPI

## Contraintes

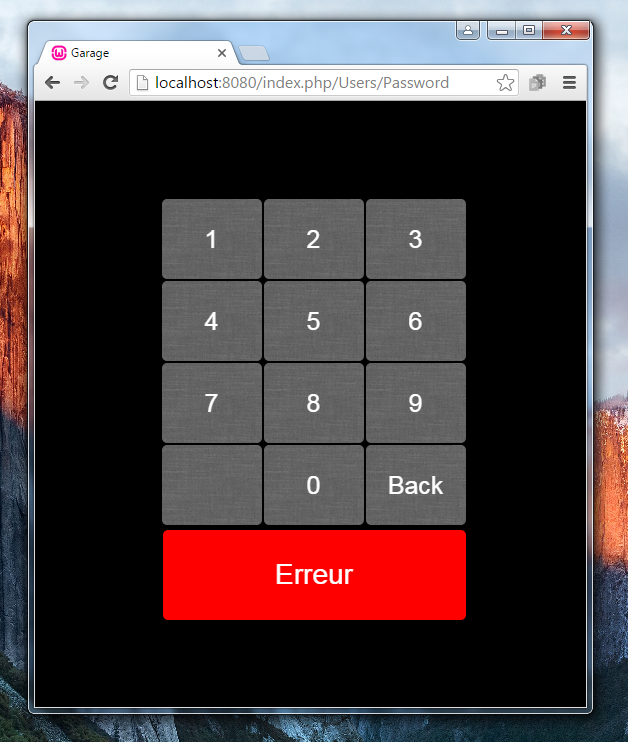
* 6 semaines
* Équipe de 2 individus

## Page de connexion



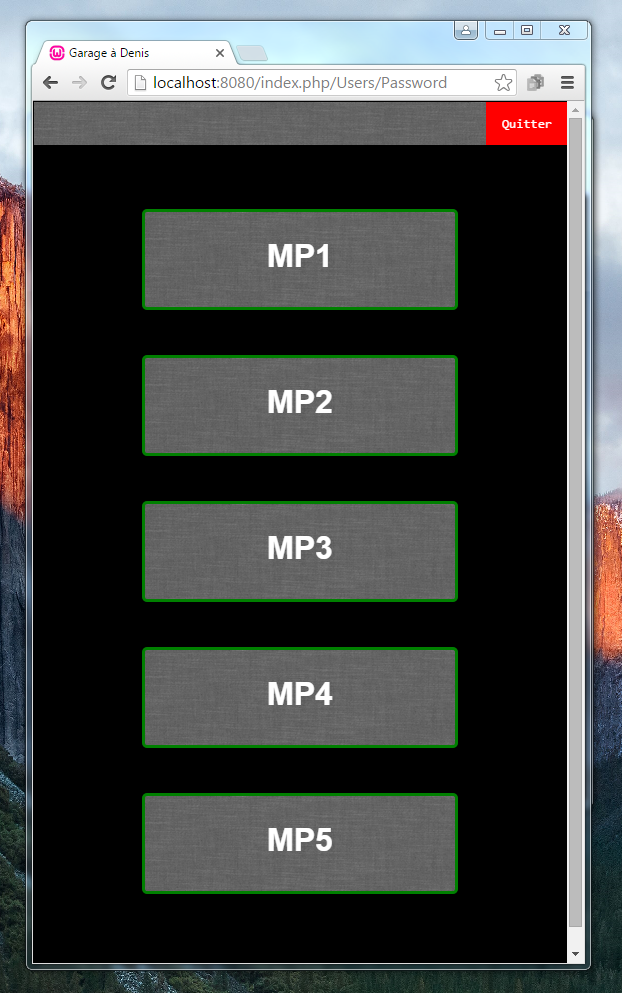
Sur la page de connexion 3 codes sont possible : un code d’administrateur qui permet la modification des informations concernant le système.

Un indicateur apparait au-dessus du pavé de numéro pour permettre un visuel plus intuitif.



Dans le cas où un mauvais code est fourni au système, la page devient inaccessible sur une durée de 10 secondes et redonne par la suite la possibilité d’un nouveau code.

## Contrôle des portes



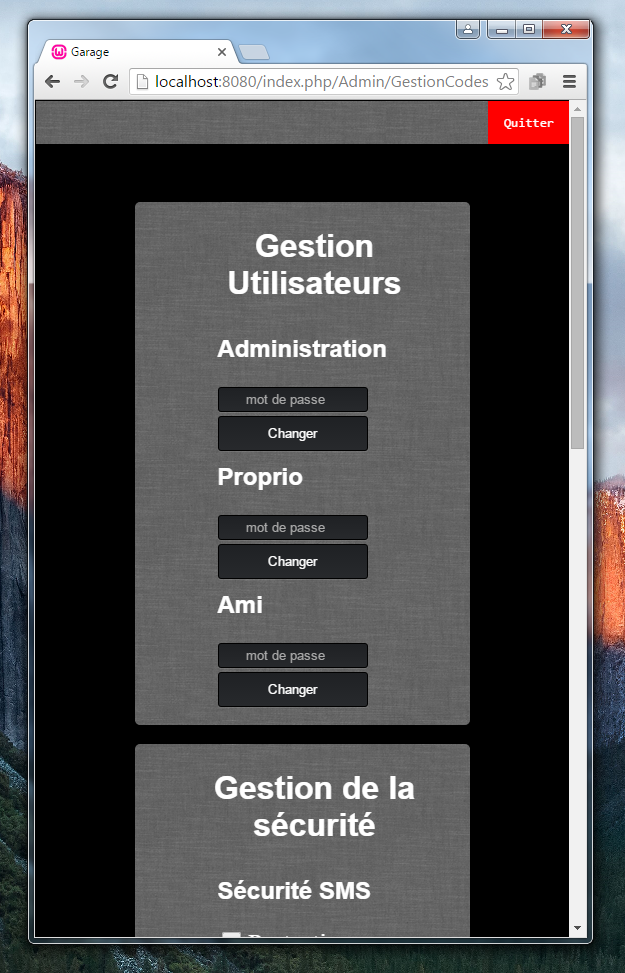
La page de gestion des porte/éléments surveillé par le RPI apparaisse sur cette page.

La page affiche les diffèrent éléments contrôlables ou non ainsi que sont statuts ouvert/fermé.

Un clique sur une des boites listées permettra d’envoyé un signal sur le relais lié à la porte, permettant ainsi d’ouvrir ou de fermer la porte.

Si un élément listé ci-joint ne contient pas de contrôle lié pour l’ouverture ou la fermeture, rien ne se passe et le système actualise la page simplement.

## Page administration



La page d’administration offre tous les options de configuration possible pour le système RPI.

Passant des mots de passe des différents utilisateurs jusqu’à la configuration des GPIO connecté.

Seul le code administrateur peut atteindre cette page qui est hautement sécurisé.

Voici la liste des éléments modifiables par la page administrateur

* Code des utilisateurs
* Activé / Désactivé la protection SMS
* La personne ressources à contacté
* Le délai avant la fermeture des portes
* Ajouter un nouvel entré de GPIO
* Modification de GPIO existant
* Gestion des distributeurs lors d’envoi de courriel/sms

## Procédure d’installation

NOTE : Tous les lignes de commande son effectué avec le terminal que l’on retrouve dans les outils du Raspberry

**Mise à jour du Raspberry**

Avant d’installer le serveur, assurons nous d’avoir une machine bien à jour. Pour ce faire nous devons posséder les droits administrateurs, soit en étant connecté en root, soit via la commande sudo.

Sudo apt-get update

Sudo apt-get upgrade

**Apache**

Installez d'abord le paquet apache2 en tapant la commande suivante dans le Terminal :

Sudo apt-get install apache2

**PHP**

Pour permettre à votre serveur Apache de traiter les fichiers PHP, vous aurez besoin d'installer PHP5 et le module PHP5 pour Apache. Tapez la commande suivante pour effectuer l’installation :

Sudo apt-get install php5

**Lancer Apache2**

Nous allons maintenant mettre en marche notre serveur apache. Vous pourrez ainsi essayer si votre serveur est fonctionnel en visitant la page localhost directement de votre navigateur web choisi.

Pour partir les services Apache2

Sudo service apache2 start

Pour arrêter les services Apache2

Sudo service apache2 stop

Pour redémarrer les services Apache2

Sudo service apache2 restart

**Installation SSMTP**

Pour permettre au site web d’envoyer des courriels, il faut tout d’abord configurer un service SSMTP. Pour se faire, installer les packets suivant :

Sudo aptitude install ssmtp

Il nous reste qu’à configurer les fichiers de configuration pour envoyer des courriers depuis notre adresse désirée. Pour ce faire, il y a deux fichiers que nous allons modifier. Cet exemple est tiré d’un utilisateur Gmail.

Le premier **/etc/ssmtp/ssmtp.conf** pour la configuration :

mailhub=smtp.gmail.com:587

UseSTARTTLS=YES

FromLineOverride=YES

AuthUser=login\_gmail

AuthPass=password\_gmail

Et le deuxième, **/etc/ssmtp/revaliases** afin de spécifier quel compte SMTP doit être utilisé pour tel ou tel utilisateur sur la machine :

root:username@gmail.com:smtp.gmail.com:587

Julien:username@gmail.com:smtp.gmail.com:587

## Mise en place du site web

**Copie du projet**

Pour commencer, copier le dossier **ProjetGarage** du projet dans le répertoire **/var/www**

Note : Vous devez copier le dossier au complet afin d’avoir le chemin suivant : **/var/www/ProjetGarage**

**Modification des fichier apache**

Pour faire pointer apache sur notre projet, il va falloir modifier quelque fichier de configuration d’Apache. Pour ce faire, vous utiliserez le terminal.

Naviguer jusqu’au répertoire Apache2 qui se trouve dans le dossier etc, à l’aide de la commande **CD**

Cd /etc/apache2/sites-available

Maintenant nous allons modifier notre premier fichier de configuration « default-ssl.conf ». Je vous suggère d’utiliser l’éditeur de texte pico, mais n’importe quel peut être utilisé.

Sudo pico default-ssl.conf

Modifier le fichier pour qu’il ressemble à ceci et **redémarrer le serveur apache** avec la commande vu plus haut :

<VirtualHost \*:80>

DocumentRoot /var/www/ProjetGarage/public

ServerName localhost

</VirtualHost>

Le site web est maintenant accessible. Pour le vérifier, ouvrir votre navigateur web et ouvrez la page localhost.

**Droit à la base de données**

Pour donner les droits d’écriture au site web vers la base de données, nous devons changer les privilèges sur le fichier garage.sqlite

Naviguer jusqu’à l’emplacement du fichier

Cd /var/www/ProjetGarage/app/bd

Donner les droits à **www-data** au fichier bade de données

Sudo chown www-data garage.sqlite

NOTE : Vous pouvez utiliser la command **ls –la** dans le répertoire du fichier affin de vérifier qui possède maintenant les privilèges sur le fichier.