

# Réponses aux demandes de bugs et assistance

---

*Du 27 février au 13 Mars*

**Recettage, corrections de bugs, ajout de fonctionnalités, et réponse aux demandes sur l'application Vroom via le logiciel de gestion de projet Redmine. Codage sur Eclipse PHP, Notepad++ en JQuery, PHP, JSon, avec le MVC Backbone, l'API JSon Restful et utilisation du système de gestion de version GitHub.**

Un bug concernant l'upload de fichiers avec un format non compatible lors de la saisie d'un dossier devait être résolu. De plus, afin de résoudre certains problèmes liés à l'utilisation de l'application Vroom, la recherche dossiers devait présenter tous les dossiers dans lequel l'utilisateur était impliqué, et flouter ceux dans lequel il ne l'était pas, en laissant apparaître les critères non sensibles. L'application devait également mettre à jour automatiquement le login de l'utilisateur s'il était identique à son numéro de téléphone ou son adresse e-mail, à chaque fois qu'il désire modifier ces derniers. Enfin, l'utilisateur devait avoir la possibilité de modifier et d'ajouter des informations personnelles sur son compte.

Le recettage général de l'application a permis de relever quelques bugs. De plus, plusieurs demandes liées à l'amélioration de l'application Vroom, utilisée pour la gestion de dossiers d'expertise par des experts, des assurés, des garages et des assurances, ont été formulées, notamment pour la correction de bugs et l'ajout de fonctionnalités. Ces demandes sont remontées de manière orale, via e-mail, ou par le biais du bouton « signaler un bug » présent sur l'application.

Dans le cas de l'upload de fichiers il est apparu que sous certains formats, il ne fonctionnait pas. Pour la recherche, nombre d'utilisateurs pensants entrer les bons critères, ne trouvaient pas le dossier désiré, remontant l'information au département informatique.

De nombreux utilisateurs, puisque l'application en offrait la possibilité, utilisaient comme login leur adresse e-mail ou leur numéro de téléphone. Cependant, les changements de numéro ou d'adresse s'opèrent fréquemment, et les utilisateurs s'apercevaient donc au moment d'une tentative de connexion que leurs identifiants étaient erronés.

Enfin, de nombreuses demandes d'utilisateurs faisaient part de l'impossibilité de modifier leurs informations personnelles sur l'application, demandant à chaque fois au département informatique la modification effective.

Cependant, ces demandes et reports étant nombreux, il fallait donc être capable d'analyser ces problèmes, répartir les tâches et les traiter de manière efficace. Une fois ceci fait, tout le travail de maintenance pouvait commencer, une fois la problématique cernée et les apports à ajouter ou modifier.

Dans le cas de l'upload de fichiers, c'est le recettage qui a permis de constater son existence. En effet, il est apparu que n'importe quel format de fichiers pouvaient être ajoutés au dossier

d'expertise, alors que l'application ne supportait que les fichiers images ou PDF, conformément aux conditions de saisies du dossier.

Après avoir vérifié le fonctionnement de la recherche et de la cohérence des données renvoyées par la base de données, il est apparu que l'utilisateur ne rentrait simplement pas les bons critères, ou tentait d'accéder à un dossier auquel il n'avait aucun droit de regard, en le signalant comme étant un bug au département informatique.

L'on a également constaté que de nombreux utilisateurs utilisaient en login l'adresse e-mail ou le téléphone. Cependant, les changements de numéro ou d'adresse s'opèrent fréquemment, et les utilisateurs ne pensaient pas toujours à modifier leur login. Ainsi, lorsqu'ils tentaient de se connecter à nouveau, le login correspondait à l'ancien e-mail ou numéro.

Enfin, l'application ne proposait pas de fonction qui permettait à un utilisateur de modifier ou d'ajouter des informations personnelles, hormis le login ou le mot de passe.

C'est l'utilisation du système de gestion de projet Redmine qui a finalement permis un traitement plus rapide des reports, des ajouts et des bugs liés à l'application. En effet, à chaque fois qu'une demande est faite, elle peut être attribuée à un développeur et est triée en fonction de son type et de sa priorité, en étant complétée par sa description. Cette répartition a donc permis de traiter les demandes décrites ci-dessous.

Après une analyse du code, le dysfonctionnement de l'upload provenait d'une absence de conditions autour du format à entrer.

Afin de signifier à l'utilisateur qui entrait les bons critères de recherches, que le dossier existait bien mais que l'utilisateur n'y avait pas accès, il a été décidé de faire apparaître tous les dossiers lors d'une recherche, mais de flouter les résultats non autorisés, en ne laissant apparaître que les cases correspondantes aux critères non sensibles.

Pour remédier aux oublis des utilisateurs, la solution retenue se charge de modifier le login en fonction de l'e-mail ou du numéro de téléphone, en se basant sur les critères de cohérences entrés par l'utilisateur. Cependant, si l'utilisateur modifie uniquement le login, l'e-mail ou le numéro ne change pas.

Enfin, pour la modification d'informations, un bouton qui permet de changer de mode et de faire apparaître des champs de saisie pré-remplis codés en javascript a été installé.

Le nombre total de mandants est affichés en haut à droite			Dot correspond au véritable nombre de mandants dans la base de données	Pass	Not done
Click sur le bouton de pagination vers l'arrière			Avance d'une page et doit être désactivé à la dernière page	Pass	Pass
Click sur le bouton de pagination vers l'avant			Recule d'une page et doit être désactivé à la première page	Pass	Pass
Saisie d'un numéro de page correct dans le cadre de pagination + Entrée			Accès à la page de la liste des mandants choisie	Pass	Pass
Saisie d'un numéro de page incorrect dans le cadre de pagination + Entrée			Aucune réaction du site, l'utilisateur reste sur la même page	Warning	Erreur lors/qn n'entre 0
Recherche via la sélection par nom du mandant			Seuls les mandants correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par prénom du mandant			Seuls les mandants correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par identifiant du mandant			Seuls les mandants correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par email du mandant			Seuls les mandants correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par téléphone du mandant			Seuls les mandants correspondant avec le critère de recherche sont affichés	Pass	Pass
Click sur un des mandants de la liste pour sa page "utilisateur"			Accès à la page "utilisateur" correspondant au mandant cliqué	Pass	Pass
Le nombre total de membres du cabinet est affichés en haut à droite			Dot correspond au véritable nombre de mandants dans la base de données	Pass	Pass
Click sur le bouton de pagination vers l'arrière			Avance d'une page et doit être désactivé à la dernière page	Pass	Pass
Click sur le bouton de pagination vers l'avant			Recule d'une page et doit être désactivé à la première page	Pass	Not done
Saisie d'un numéro de page correct dans le cadre de pagination + Entrée			Accès à la page de la liste des mandants choisie	Pass	Pass
Saisie d'un numéro de page incorrect dans le cadre de pagination + Entrée			Aucune réaction du site, l'utilisateur reste sur la même page	Warning	Pass
Recherche via la sélection par nom du staff			Seuls les staff correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par prénom du staff			Seuls les staff correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par identifiant du staff			Seuls les staff correspondant avec le critère de recherche sont affichés	Warning	Quand une lettre est rentrée, le chargement est très long
Recherche via la sélection par email du staff			Seuls les staff correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par téléphone du staff			Seuls les staff correspondant avec le critère de recherche sont affichés	Pass	Pass
Click sur un des staffs de la liste pour sa page "utilisateur"			Accès à la page "utilisateur" correspondant au staff cliqué	Pass	Pass
Le nombre total de gangs est affichés en haut à droite			Dot correspond au véritable nombre de gangs dans la base de données	Pass	Not done
Click sur le bouton de pagination vers l'arrière			Avance d'une page et doit être désactivé à la dernière page	Pass	Pass
Click sur le bouton de pagination vers l'avant			Recule d'une page et doit être désactivé à la première page	Pass	Pass
Saisie d'un numéro de page correct dans le cadre de pagination + Entrée			Accès à la page de la liste des gangs choisie	Pass	Pass
Saisie d'un numéro de page incorrect dans le cadre de pagination + Entrée			Aucune réaction du site, l'utilisateur reste sur la même page	Warning	Erreur lors/qn n'entre 0
Recherche via la sélection par libellé de gang			Seuls les gangs correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par identifiant du gang			Seuls les gangs correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par email du gang			Seuls les gangs correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par téléphone du gang			Seuls les gangs correspondant avec le critère de recherche sont affichés	Pass	Pass
Click sur un des gangs de la liste pour sa page "utilisateur"			Accès à la page "utilisateur" correspondant au gang cliqué	Pass	Pass
Le nombre total d'assurés est affichés en haut à droite			Dot correspond au véritable nombre d'assurés dans la base de données	Pass	Not done
Click sur le bouton de pagination vers l'arrière			Avance d'une page et doit être désactivé à la dernière page	Pass	Pass
Click sur le bouton de pagination vers l'avant			Recule d'une page et doit être désactivé à la première page	Pass	Pass
Saisie d'un numéro de page correct dans le cadre de pagination + Entrée			Accès à la page de la liste des assurés choisie	Pass	Pass
Saisie d'un numéro de page incorrect dans le cadre de pagination + Entrée			Aucune réaction du site, l'utilisateur reste sur la même page	Warning	Erreur lors/qn n'entre 0
Recherche via la sélection par nom / prénom de l'assuré			Seuls les assurés correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par identité de l'assuré			Seuls les assurés correspondant avec le critère de recherche sont affichés	Warning	Ne fonctionne pas quant on rentre uniquement le prénom (Ou le prénom avant le nom)
Recherche via la sélection par email de l'assuré			Seuls les assurés correspondant avec le critère de recherche sont affichés	Pass	Pass
Recherche via la sélection par téléphone de l'assuré			Seuls les assurés correspondant avec le critère de recherche sont affichés	Pass	Pass
Click sur Export CSV depuis la liste principale			Exporte un fichier "assures.csv"	Pass	Pass
Click sur Export CSV depuis une recherche			Exporte un fichier "recherche_indefinis.csv"	Warning	Fichier qui contient le contenu pas les bonnes données
Click sur un des assurés de la liste pour sa page "utilisateur"			Accès à la page "utilisateur" correspondant au garage cliqué	Pass	Pass

Le plan de recette établi par le chef de projet a permis de constater quelques bugs sur la version développeur juste avant la mise en production. Ainsi, chaque amélioration, modification ou correction de bug a d'abord été effectué en local, porté sur la version développeur, où un recettage complet est effectué de chaque version de l'application, puis une fois validé, a été mise en production sur le serveur loué par l'entreprise.

## Accueil du logiciel redmine avec la liste des demandes assignées

Projet Expertise & Concept » VROOM v2

Enter Search Text

VROOM v2

APERÇU

ACTIVITÉ

DEMANDES

NOUVELLE DEMANDE

GANTT

CALENDRIER

ANNONCES

DOCUMENTS

WIKI

FICHIERS

DÉPÔT

CONFIGURATION

Feature #39

Modifier

Surveiller

Copier

Garage recherche d'un dossier avec immat --> ce dossier existe mais il ne vous est pas attribué (si vous avez le véhicule c'est chaud faut le rendre)

« Précédent | 123 sur 153 | Suivant »

Ajouté par Thierno Rignoux il y a environ un mois. Mis à jour il y a 30 jours.

Statut:

Priorité:

Assigné à:

Catégorie:

Version cible:

Resolved

Urgent

Remi Péral

-

-

Début:

Echéance:

% réalisé:

Temps estimé:

24/02/2015

0%

4.00 heures

Sous-tâches

Ajouter

Demandes liées

Ajouter

Lié à Feature #28: Un garage doit avoir l'info que le véhicule se trouve dan...

Resolved

24/02/2015

Historique

Mis à jour par Thierno Rignoux il y a environ un mois

#1

• Lié à Feature #28: Un garage doit avoir l'info que le véhicule se trouve dans un autre garage et lui donne la possibilité de nous informer que c'est lui qui l'a ajouté

Mis à jour par Thierno Rignoux il y a environ un mois

#2

• Assigné à mis à Remi Péral

Mis à jour par Remi Péral il y a 30 jours

#3

• Statut changé de Backlog à Resolved

Révisions associées

Révision 9bd32766

Ajouté par Thierno Rignoux il y a environ un mois

Dans la recherche quand on choisit un état on n'affiche que des dossiers de cet état(refs #39 )

Révision adac1a4a

Ajouté par Remi Péral il y a environ un mois

REFS #39 - Gestion des dossiers non attribués

Révision 46061cf1

Ajouté par Thierno Rignoux il y a environ un mois

REFS #39 Correction code Rémi

Demandes

Voir toutes les demandes

Résumé

Calendrier

Gantt

Rapports personnalisés

Demandes en cours

Demandes Résolues

En attente de Feedback

Nouvelles demandes

## Détail des demandes

					Enter Search Text	Normal	dans le staf nous avons plusieurs fois les même personnes	04/03/2015 18:50	VROOM v2
	81	Bug	New	Normal			pour le mandant 191BOU j'ai une 20aine d'accès avec le même mail	04/03/2015 18:50	
	80	Bug	New	Normal			le stat ne doit pas avoir accès à la gestion des tournées et à la gestion du cabinet	04/03/2015 18:40	
	79	Bug	New	Normal			dossier manquant	04/03/2015 14:01	
	78	Feature	New	High			Appli mobile : récupération automatique du token	04/03/2015 11:09	
	77	Bug	Backlog	Immediate			Renseigner les nouveaux cabinets sur expertiseconcept.com	Remi Péral 13/03/2015 10:10	
	76	Bug	Backlog	Urgent			Définir jours de passage Cabinet Gersois	04/03/2015 11:08	
	75	Feature	Backlog	Normal			Intégration HERMEX (après BCE)	04/03/2015 11:08	
	74	Feature	New	High			Intégration CME (Theuil)	04/03/2015 11:08	
	73	Feature	New	High			Intégration Exac	04/03/2015 11:07	
	72	Feature	Backlog	Normal			Intgration Garnier	04/03/2015 11:07	
	71	Feature	Backlog	High			Intégration BCE	04/03/2015 11:07	
	70	Feature	Backlog	High			Intégration Lemaire	04/03/2015 11:07	
	69	Feature	Backlog	Urgent			Tuer la V1 pour Avezou d'ici 8 jours	04/03/2015 11:06	
	68	Feature	Backlog	Urgent			RDV d'expertise : process à revoir	04/03/2015 11:06	
	67	Bug	Resolved	Immediate			Lacour > le status de certains dossiers reste "EN_ATTENTE" alors qu'une date de RDV d'expertise a été donnée.	24/03/2015 16:27	
	66	Bug	Resolved	Immediate			Upload de documents : problème chez Lacour	05/03/2015 10:29	
	65	Bug	New	Normal			Pb int:gratiation	04/03/2015 08:41	
	64	Feature	Backlog	Immediate			Ajouter petit texte qui spécifie la disparition de la V1 pour les membres du	Thierno 04/03/2015 11:10	

## Liste complète des demandes

Le logiciel de gestion de projet Redmine a quant à lui permis une centralisation des bugs, des reports, mais aussi des améliorations à apporter et a permis une expression plus fluide des besoins du client. Triés par importance et répartis à chaque développeur, Redmine a fluidifié le traitement de chaque demande.

## Upload de fichiers

Afin de compléter son dossier, l'utilisateur a la possibilité d'y joindre un document pdf ou un fichier image. Pour cela, il suffit de cliquer sur le bouton « Joindre une image au dossier » et une boîte de dialogue s'ouvre alors. Il peut au choix, sélectionner un ou plusieurs fichiers. Il ne peut cependant pas sélectionner un autre fichier d'un format différent de ceux cités ci-dessus. Si tel est le cas, l'application doit le lui signifier.

```
fileSelected : function(ev) {  
  
    if(document.all && !window.atob) { // gi <= IE9  
        names = $('#new_document').val();  
    } else {  
        var files = $('#new_document').prop("files");  
        var names = $.map(files, function(val) {  
            return val.name;  
        });  
    }  
    testExtension = false;
```

*La méthode « fileSelected » est appelée lors d'un clic sur le bouton « Joindre une image ». Le code récupère ensuite le nom et l'extension de tous les fichiers sélectionnés séparément.*

L'application doit ensuite pouvoir vérifier le format de chaque fichier, et l'ajouter au dossier s'ils sont tous conformes.

```
for (var i = 0; i < names.length; i++) {  
    var filetype = names[i].split(".");  
  
    filetype = filetype[filetype.length - 1];  
  
    if (this.typeDocs == "") {  
        if (filetype.toLowerCase() == "pdf" || filetype.toLowerCase() == "jpg" || filetype.toLowerCase() == "jpeg" ||  
            filetype.toLowerCase() == "png" || filetype.toLowerCase() == "gif" ) {  
            testExtension = true;  
        }  
    }  
    if (testExtension == false) {  
        swalvroom("Erreur de type de fichier", "Merci de mettre des fichiers image (.jpeg, .png, etc..) OU un fichier PDF.", "warni  
        document.getElementById("new_document").value = "";  
        return false;  
    }  
}
```

*Le code vérifie ensuite les extensions de tous les formats de chaque fichier et teste s'ils sont conformes. Si ce n'est pas le cas, la variable « testExtension » est initialisé à false et un message d'erreur se déclenche*



*Box d'avertissement*

## Recherche et dossiers non autorisés

Pour parvenir à afficher et flouter les dossiers non autorisés mais correspondants à certains critères de recherches par l'utilisateur, l'application a tout d'abord exécuté une requête en fonction de ces critères, mais dont son id n'était pas présent dans la liste des accès au dossier. La requête est contenue dans la fonction `getUnauthorizedDossiers`.

```
$id_dossier = $request->get("id_dossier");
$nom = $request->get("nom");

if ($request->get("offset") != null) {
    $offset = $request->get("offset");
} else {
    $offset = 0;
}

if ($request->get("limit") != null) {
    $limit = $request->get("limit");
} else {
    $limit = 10;
}

$numeroPage = $offset;
$offset = $offset * $limit;
$nbElementsARemplir = $offset - $nbDossiersAutorises;

$immatriculation = $request->get("immatriculation");
if ($immatriculation != "") {
    $immatriculation = FormatService::filterImmat($immatriculation);
}

$marque_vehicule = $request->get("marque");
$modele_vehicule = $request->get("modele");
$etat = $request->get("etat");
$type = $request->get("type");

if ($status_filter != null) {
    $etat = $status_filter;
}

$id_cabinet = $this->authenticated_user->entity->id_cabinet;
$registre = RegistreGlobal::findFirst($this->authenticated_user->entity->id);
```

*Le code récupère toutes les informations relatives à la recherche de l'utilisateur, ainsi que le numéro de la page actuelle, la limite, et le nombre de dossiers non autorisés à remplir en fonction de la page actuelle*

Afin de minimiser le temps d'exécution du code, l'application passe en revue les cas d'exclusions, c'est-à-dire tous les cas où la fonction ne doit rien renvoyer, même si la requête contient des résultats.

```

if ($this->authenticated_user->entity->role == "assure") {
    return false;
}

if ($id_dossier == "" && $nom == "" && $immatriculation == "" && $modele_vehicule == "" && $marque_vehicule == "" && ($etat == "" ||
    return false;
}

if ($etat != "" && $etat != "tous" && $marque_vehicule == "" && $immatriculation == "" && $modele_vehicule == "" && $nom == "") {
    return false;
}

```

*Un staff doit avoir accès à tous les dossiers. Un simple assuré n'a accès qu'à ses propres dossiers. Une recherche vide doit afficher uniquement les dossiers autorisés et la recherche uniquement par état ne doit afficher que les dossiers concernés autorisés. Si une de ces conditions est respecté, la fonction renvoie simplement false*

C'est ensuite que la requête se construit. Partant du principe que les critères entrés par l'utilisateur ne seront pas exactement les mêmes que ceux contenues dans la base de données, il a fallu pour chaque case, adapter la requête afin que le résultat soit le plus large possible.

```

$id_identite = $this->authenticated_user->entity->id;
$query_data = "SELECT * FROM dossiers_list L LEFT JOIN dossier_entity ON L.id = dossier_entity.id_dossier
WHERE dossier_entity.id_entite != $id_identite AND ";

$firstElementInRequete = false;
if ($nom != "") {
    $query_data = $query_data . "nom_prenom LIKE '%" . $nom . "%'";
    $firstElementInRequete = true;
}

if ($id_dossier != "") {
    if ($firstElementInRequete == false) {
        $query_data = $query_data . "internal_id = '$id_dossier'";
        $firstElementInRequete = true;
    } else {
        $query_data = $query_data . " AND internal_id = '$id_dossier'";
    }
}

if ($immatriculation != "") {
    if ($firstElementInRequete == false) {
        $query_data = $query_data . "immatriculation LIKE '$immatriculation'";
        $firstElementInRequete = true;
    } else {
        $query_data = $query_data . " AND immatriculation LIKE '$immatriculation'";
    }
}

```



```

if ($etat != "" && $etat != "tous") {
    // les dossiers clos sont mixés
    $etats_clos = array(
        "DOSSIER_CLOS_VR",
        "DOSSIER_CLOS_ER",
        "DOSSIER_CLOS_VEI",
        "FACTURE_NOT_OK"
    );

    if (in_array($etat, $etats_clos)) {
        if ($firstElementInRequete == false) {
            $query_data = $query_data . " &&&& IN ('DOSSIER_CLOS_VR', 'DOSSIER_CLOS_ER', 'DOSSIER_CLOS_VEI', 'FACTURE_NOT_OK') ";
            $firstElementInRequete = true;
        } else {
            $query_data = $query_data . " AND &&&& IN ('DOSSIER_CLOS_VR', 'DOSSIER_CLOS_ER', 'DOSSIER_CLOS_VEI', 'FACTURE_NOT_OK') ";
        }
    } else {
        $wheres[] = " &&&& = :&&&& ";
        $values[":&&&&"] = $etat;

        if ($firstElementInRequete == false) {
            $query_data = $query_data . " &&&& LIKE '$etat' ";
            $firstElementInRequete = true;
        } else {
            $query_data = $query_data . " AND &&&& LIKE '$etat' ";
        }
    }
}

$query_data .= " GROUP BY L.id ";

```

*La requête se construit en fonction de chaque critère récupéré précédemment, mais récupère uniquement ceux dont l'id de l'utilisateur ne correspond pas. Le code teste s'ils sont présents, et concatène chaque partie dans la variable \$query\_data.*

Le code doit ensuite pouvoir, à partir de cette requête, renvoyer un tableau contenant les dossiers non autorisés, en fonction de la page sur laquelle l'utilisateur se trouve. En fait, à chaque fois que l'utilisateur change de page, la fonction est appelée mais ne renvoie pas toujours le même nombre de dossiers non autorisés, ni à la même position. Afin de pouvoir réaliser cela, nous nous servons de l'Offset, de la limite et du nombre total de dossiers non autorisés renvoyés.

```

// Requête qui va permettre d'évaluer le nombre total de dossiers non autorisés à partir des critères entrés
$query_count = $this->getDI()
    ->getShared('db')
    ->prepare($query_data);
$query_count->setFetchMode(Phalcon\Db::FETCH_ASSOC);
$query_count->execute($values);
$data_count = $query_count->fetchAll();

$nbDossiersNonAutorises = count($data_count);

```

*L'application exécute la requête et compte le nombre de résultats*

```

if (($nbElementsARemplir) < 20) {
    $offset = 0;
    if ($nbElementsARemplir <= - 20) {
        $nbElementsARemplir = 0;
        return $nbDossiersNonAutorises;
    } else {
        $nbElementsARemplir = $nbElementsARemplir + 20;
    }
} else {
    $offset = $offset - (ceil($nbDossiersAutorises / 10) * 10);
}

$limite = "limit " . $offset . "," . $nbElementsARemplir;
$query_data .= $limite;

```

*Détermine la limite et l'offset à appliquer à la requête. Ils changent tous deux en fonction du nombre attendu à remplir de dossiers non autorisés. Ces conditions sont ensuite concaténés à la requête*

Enfin, la requête est exécutée. L'application vérifie alors qu'il n'y ait pas d'éventuels doublons avec les dossiers autorisés, et si c'est le cas, renvoie leur nombre. Elle se charge ensuite de parcourir le tableau généré par la requête et de flouter ou non les critères entrés par l'utilisateur.

```

if (($id_dossier != "" && $data[0] != "") || ($nom != "" && $data[0] != "")) {

    $requeteId = "SELECT count(DISTINCT dossier_entity.id_entite) AS score FROM dossier_entity
    WHERE dossier_entity.id_dossier = " . $data[0]['id'];

    $count = $this->getDI()
        ->getShared('db')
        ->prepare($requeteId);
    $count->execute($values);

    $nbDoublons = $count->fetch()["score"];
}

```

*Permet de déterminer si la liste de dossiers non autorisés est un doublon de la liste de dossiers autorisés*

```

foreach ($data as $key_element => $element) {
    $data[$key_element]['nbDossiers'] = $nbDossiersNonAutorises;
    $data[$key_element]['nbDoublons'] = $nbDoublons;

    if ($id_dossier != $data[$key_element]['internal_id']) {
        $data[$key_element]['internal_id'] = "obfuscated";
    }

    if ($nom != $data[$key_element]['nom_prenom']) {
        $data[$key_element]['nom_prenom'] = "obfuscated";
    }

    if ($immatriculation != $data[$key_element]['immatriculation']) {
        $data[$key_element]['immatriculation'] = "obfuscated";
    }

    if ((( $modele_vehicule != $data[$key_element]['modele'] ) && ( $marque_vehicule
    != $data[$key_element]['constructeur'] )) || ( $data[$key_element]['constructeur'] == "INCONNU"
    || ( $data[$key_element]['modele'] == "" ) ) {

        $data[$key_element]['modele'] = "obfuscated";
        $data[$key_element]['constructeur'] = "obfuscated";
    }

    if ($etat != $data[$key_element]['etat_description'] || $etat == "tous") {
        $data[$key_element]['etat_description'] = "obfuscated";
    }
}

```

*Boucle qui permet de déterminer si un élément doit être vidé ou non. Il l'est si le critère recherché n'est pas contenu dans cet élément, la case du critère est alors remplacée par « obfuscated » dont une propriété dans le CSS se charge de flouter et qui apparaîtra comme ceci même si on inspecte l'élément.*

La méthode `getDossiersAction` va quant à elle, retourner les dossiers autorisés. Mais, il faut pouvoir être en conformité dans le cas où une page doit afficher les dossiers autorisés et non autorisés. C'est également cette méthode qui va permettre d'afficher tous les éléments désirés.

Après avoir exécuté la recherche en fonction des critères, on appelle la fonction des dossiers non autorisés et on vérifie que la liste renvoyé n'est pas un doublon de la liste des dossiers autorisés, qui est un cas constaté lors du recettage.

```

$lesDossiersNonAutorises = $this->getUnauthorizedDossiers($request, $nbDossiers);

$nbDossiersNonautorises = 0;

] if ($lesDossiersNonAutorises != false) {
]
    if ($nbElementsARemplir <= - 20) {
        $nbDossiersNonautorisesTotal = $lesDossiersNonAutorises;
    } else {
        $nbDossiersNonautorisesTotal = $lesDossiersNonAutorises[0]['nbDossiers'];
    }

    if ($lesDossiersNonAutorises[0]['nbDoublons'] == $nbDossiers) {
        $nbDossiersNonautorisesTotal = 0;
        $lesDossiersNonAutorises = false;
    }

    $nbDossiersNonautorises = count($lesDossiersNonAutorises);
}

```

*Le code exécute puis procède aux vérifications nécessaires sur la liste des dossiers non autorisés*

Enfin, le code, en fonction de la situation de la page, remplit le tableau final soit de dossiers autorisés, soit de dossiers non autorisés, soit de dossiers autorisés et non autorisés.

```
$returned_data = $data->fetchAll();

if ($lesDossiersNonAutorises != false && $nbElementsARemplir > - 20) {

    if ((count($returned_data) < 20) && ($nbDossiersNonautorises < 20)) {

        $returned_data = array_merge($returned_data, $lesDossiersNonAutorises);
    } else {
        $returned_data = $lesDossiersNonAutorises;
    }
}
```

*Le code effectue les différents tests en fonction du nombre de dossiers autorisés ou non de chaque tableau à chaque passage, c'est-à-dire à chaque fois que l'on change de page*

RECHERCHE DOSSIER

Identifiant du dossier

Nom assuré

Jean

Immatriculation du véhicule

Marque du véhicule

Renault

Modele du véhicule

Statut du dossier

Rechercher

Mes 264 dossiers

Page 9 sur 14

N° du dossier	Immatriculation	Marque - Modèle	Nom de l'assuré	Cabinet	Status	Date de création
		RENAULT - KANGOO EXP 1.5 DCI		Avezou-Chastaing	Dossiers clos	20/02/2014
		RENAULT - CLIO 3 1.2 TCE 100 CH DY		Avezou-Chastaing	Dossiers clos	24/02/2014
		RENAULT - KOLEOS		Avezou-Chastaing	Dossiers clos	27/02/2014
		RENAULT - MEG III EST DCI 130 PRIV		Avezou-Chastaing	Dossiers annulés	04/03/2014
		RENAULT - CLIO III		Avezou-Chastaing	Dossiers clos	05/03/2014
		RENAULT - MASTER		Avezou-Chastaing	Dossiers clos	05/03/2014
		RENAULT - KANGOO II		Avezou-Chastaing	Dossiers clos	07/03/2014
		RENAULT - SCENIC GRAND		Avezou-Chastaing	Dossiers clos	13/03/2014
		RENAULT - TRAFIC		Avezou-Chastaing	Dossiers clos	17/03/2014

*Cas où la page est remplie de dossiers non autorisés*

2014003008	CZ-848-EE	SCENIC II DCI 105 CFT DY	ONEL ELISABETH	Avezou- Chastaing	Dossiers clos	26/02/2014
2014002841	8389KS82	VOLKSWAGEN - JETTA 2.0 TDI 140 CH CON	AMARA JEANINE	Avezou- Chastaing	Dossiers clos	24/02/2014
2014002772	AH-860-HG	RENAULT - TWINGO	DA ROLD THIERRY	Avezou- Chastaing	Dossiers clos	21/02/2014
2014002760	AQ-233-BA	RENAULT - CLIO PRIVILE	LE DANTEC, JORIS	Avezou- Chastaing	Dossiers annulés	21/02/2014
2014002674	AH-970-NQ	RENAULT - CLIO 1.5	SEGOVIA DARIO	Avezou- Chastaing	Dossiers clos	20/02/2014
2014002714	1975KQ82	RENAULT - CLIO 1.5 DCI 65 CFT PK C	DE MEYER KATHY	Avezou- Chastaing	Dossiers clos	20/02/2014
2014002657	AL-549-AB	RENAULT - GRAND SCEN LUXE	AZARKAN ARAZKI	Avezou- Chastaing	Dossiers clos	19/02/2014
		RENAULT - KANGOO EXP 1.5 DCI		Avezou- Chastaing	Dossiers clos	20/02/2014

*Cas où la page est remplie de dossiers autorisés et non autorisés*

## Cohérence des données

Lors d'un clic sur le bouton « modifier » de la partie « Informations de connexion » du compte utilisateur, la méthode « setAction » est appelée. C'est elle qui se charge de la cohérence des données

```
$request = new Request();
$params = $request->getAll();
$correspondanceMail = false;
$correspondanceTel = false;

if ($id_user !== false) {
    $user = User::findFirst(array(
        '(id = :id:)',
        'bind' => array(
            'id' => $id_user
        )
    ));

    $nbUsers = count(User::find(array(
        '(id_individu = :id_individu:)',
        'bind' => array(
            'id_individu' => $user->id_individu
        )
    )));
}
```

*Le code initialise puis récupère les données de l'utilisateur qui souhaite modifier ses informations, ainsi que les informations en elles-mêmes*

```

if($user->login == $user->email)
{
    $correspondanceMail = true;
}

if(($user->login == $user->telephone_portable))
{
    $correspondanceTel = true;
}
}

```

*Le code teste si le login correspond à l'email, au telephone ou non*

```

if($correspondanceMail == true)
{
    $user->login = $params["email"];
}
if($correspondanceTel == true)
{
    $user->login = $params["telephone_portable"];
}

```

*Si le teste s'avère être vrai pour l'un ou l'autre, les paramètres sont modifiés automatiquement*

## Modification des informations personnelles

Lorsqu'un utilisateur décide de modifier ses informations personnelles, il appuie sur le bouton « modifier vos informations »

Mon Compte

Mes Informations Informations de connexion Options

Nom : Thierno Rignoux

Rôle : Expert

Email : t.ignoux@univ-lyon1.fr

Téléphone : 06 88 88 88 88

Mobile : 06 88 88 88 88

Fax : .

Modifier vos informations

Page d'affichage des informations personnelles

Le clic sur ce bouton appelle une méthode en JQuery

```
<li class = "getResultat"><span class="label">Code Postal: </span><span class="resultat"><p><%= user.get("adresse").cp %></p></span></li><div class="label infos_input" >Code Postal : <input type="cpInfos" value="<%= user.get("adresse").cp %>" id="changeCpInfos" type="text"></div></li><li class = "getResultat"><span class="label">Ville : </span><span class="resultat"><p><%= user.get("adresse").ville.libelle %></p></span></li><div class="label infos_input" >Ville : <input type="villeInfos" value="<%= user.get("adresse").ville.libelle %>" id="changeVilleInfos" type="text"></div></li></div> <div class="infos" > <div class="label infos_input" >Adresse : <input type="adresse1Infos" value="" id="changeAdresse1Infos" type="text"> <input type="adresse2Infos" value="" id="changeAdresse2Infos" type="text"> </div> <div class="label infos_input" >Code Postal : <input type="cpInfos" value="" id="changeCpInfos" type="text"> </div> <div class="label infos_input" >Ville : <input type="villeInfos" value="" id="changeVilleInfos" type="text"> </div> </div> <div class="infos" > <div class="label infos_input" >Email : <input type="emailInfos" value="<%= user.get("email") %>" id="changeEmailInfos" type="text"> </div> <div class="label infos_input" >Téléphone : <input type="telInfos" value="<%= user.get("telephone_fixe") %>" id="changeTelInfos" type="text"> </div> <div class="label infos_input" >Mobile : <input type="mobileInfos" value="<%= user.get("telephone_portable") %>" id="changeMobileInfos" type="text"> </div> <div class="label infos_input" >Fax : <input type="faxInfos" value="<%= user.get("fax") %>" id="changeFaxInfos" type="text"> </div> </div> <div class="infos" > <div class="label infos_input" >Nom : <input type="nomInfos" value="<%= user.get("nom") %>" id="changeNomInfos" type="text"> </div> <div class="label infos_input" >Prénom : <input type="prenomInfos" value="<%= user.get("prenom") %>" id="changePrenomInfos" type="text"> </div> <div class="label infos_input" >Rôle : <input type="roleInfos" value="<%= user.get("role") %>" id="changeRoleInfos" type="text"> </div> </div> <div class="infos" > <div class="label infos_input" >Mot de passe : <input type="password" value="" id="password" type="password"> </div> <div class="label infos_input" >Confirmer le mot de passe : <input type="password" value="" id="confirm_password" type="password"> </div> </div> <div class="infos" > <div class="label infos_input" >Annuler : <input type="button" value="Annuler" id="cancelSaveButton" type="button"> </div> <div class="label infos_input" >Enregistrer : <input type="button" value="Enregistrer" id="saveInfosButton" type="button"> </div> <div class="label infos_input" >Modifier vos informations : <input type="button" value="Modifier vos informations" id="changeInfosButton" type="button"> </div> </div>
```

La page html de la vue correspondante contient les inputs des informations à modifier et les informations elles-mêmes. Les inputs apparaissent lors du clic et les informations disparaissent

```

displayInfosChange: function(e){
    $(".infos_input").fadeIn();
    $(".getResultat").fadeOut();
    $(e.currentTarget).hide();
},

cancelInfos: function(e){
    $(".infos_input").fadeOut();
    $(".getResultat").fadeIn();
    $(e.currentTarget).hide();
},

```

*Ces fonctions en JQuery permettent d'afficher les inputs regroupés ou d'annuler et de faire réapparaître les informations de base*

Mon Compte

Mes Informations Informations de connexion Options

● Nom : Thierno Rignoux

● Rôle : Expert

● Adresse :

● Code Postal :

● Ville :

● Email : thierno@gmail.com

● Téléphone :

● Mobile : 06 999 9999

● Fax:

Enregistrer Annuler

### *Page d'affichage des informations modifiables*

Une fois les informations entrées, l'appui sur le bouton « Enregistrer » va appeler la méthode « InfosChange » qui va se charger de modifier tous les critères nécessaires et afficher une box de confirmation de modification.



```

InfosChange: function(e){

    NProgress.configure({ parent: '#content' });
    NProgress.start();

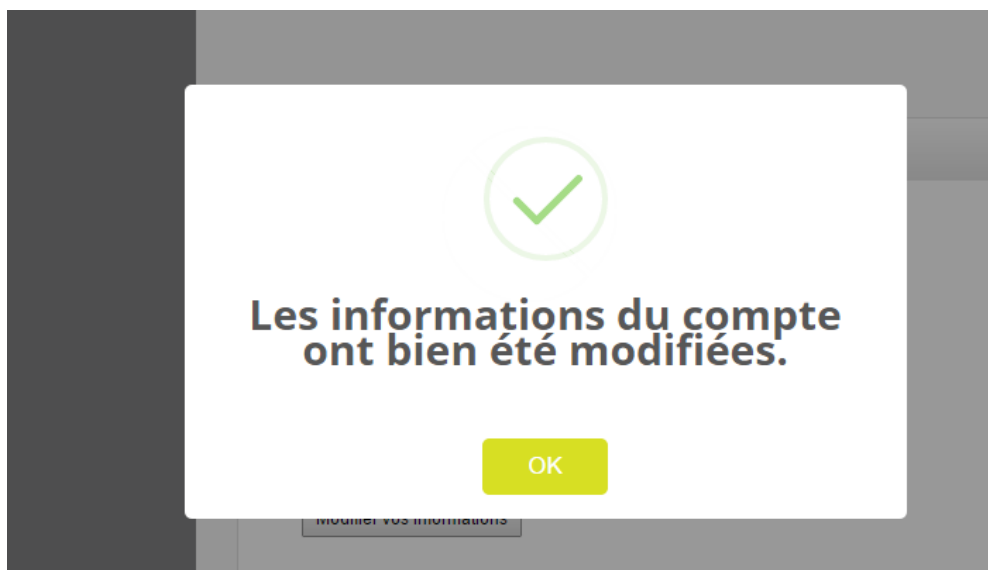
    var arrayInfo = App.models.user.getConnectionData();

    arrayInfo["nom"] = $("#changeNomInfos").val();
    arrayInfo['prenom'] = $("#changePrenomInfos").val();
    arrayInfo['email'] = $("#changeEmailInfos").val();
    arrayInfo['adresse1'] = $("#changeAdresse1Infos").val();
    arrayInfo['adresse2'] = $("#changeAdresse2Infos").val();
    arrayInfo['cp'] = $("#changeCpInfos").val();
    arrayInfo['ville'] = $("#changeVilleInfos").val();
    arrayInfo['telephone_fixe'] = $("#changeTelInfos").val();
    arrayInfo['telephone_portable'] = $("#changeMobileInfos").val();
    arrayInfo['fax'] = $("#changeFaxInfos").val();

    request(App.commons.URLs.setEntite + App.models.user.get("id_individu"), arrayInfo, function(data) {
        if(data.response == "OK") {
            swalvroom("Les informations du compte ont bien été modifiées.", "", "success");
        } else {
            swalvroom("Erreur pendant l'enregistrement des informations du compte.", "", "warning");
        }

        $(".infos_input").fadeOut();
        $(".getResultat").fadeIn();
        NProgress.done();
    }, true);
},

```



*Box de confirmation*

La résolution de ces différents bugs et demande, outre les connaissances techniques et l'apprentissage de langage et de méthodes jusqu'ici alors méconnus, m'a permis de m'intégrer dans l'équipe de développement et de bénéficier d'une véritable expérience en matière de gestion de projet et d'une compréhension-réponse à une expression du besoin.