

# Rapport de stage projet SPANC



<https://github.com/Gotlub/SoniaProject>

## Introduction :

Le Service Public d'Assainissement Non Collectif a besoin de donner des rendez-vous sur les mêmes lieux tous les dix ans. Ils travaillent avec un logiciel de gestion et des tableurs qui ne leur permettent pas de trier efficacement les anciens rendez-vous pour en définir les nouveaux.

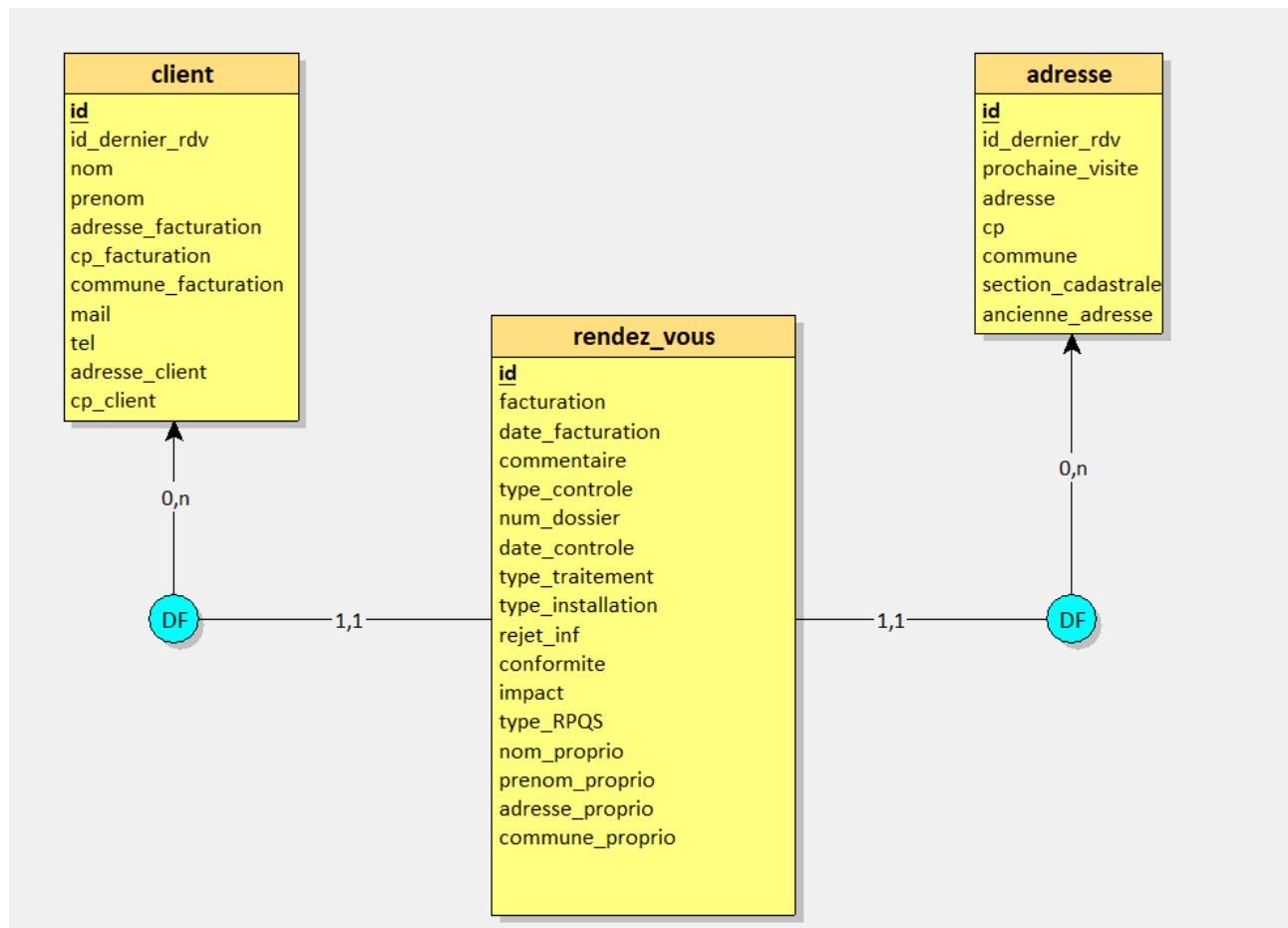
Le projet est donc mis en place pour répondre à cette problématique.

## Outils pour la réalisation du projet :

- Python pour extraire les données des fichiers CSV
- MySQL pour la gestion de la base de données
- Symfony 6.4 et PHP version supérieure ou égale à 8.2
- Node.js version supérieure ou égale à 10

## Mise en place du projet :

Les données du Service Public d'Assainissement Non Collectif sont sur des documents Excel non normalisés. Après quelques jours de cartographie des données, la décision concernant le format de la base de données a été prise.



Le but du site sera de pouvoir trier et filtrer les différents rendez-vous sur plusieurs critères et de pouvoir extraire ces données au format Excel.

Les données des fichiers .Xlsx sont exporté en .Csv après remplacement des sauts de lignes.

- Menu Edit -> Find & Replace  
Find: \n  
Replace: (leave empty)  
Other Options 1: [x] Regular expressions
- Click Replace All

La structure de la base de données a été générée avec Symfony a partir des entités.

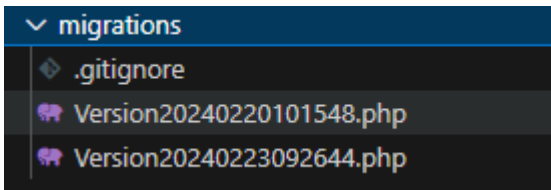
```
PS C:\Projetsonia\Projet> php bin/console d:d:c
Created database `sonia` for connection named default
PS C:\Projetsonia\Projet> php bin/console d:m:m

WARNING! You are about to execute a migration in database "sonia" that could result in schema changes and data loss. Are you sure you wish to continue? (yes/no) [yes]:
> yes

[notice] Migrating up to DoctrineMigrations\Version20240223092644
[notice] finished in 459ms, used 26M memory, 2 migrations executed, 9 sql queries

[OK] Successfully migrated to version: DoctrineMigrations\Version20240223092644

PS C:\Projetsonia\Projet>
```



Chaque document Excel doit être correctement cartographié (les documents ne sont pas standardisés et les colonnes sont différentes d'année en année).

```
[[['a', '0'], ['b', '1'], ['c', '2'], ['d', '3'], ['e', '4'], ['f', '5'], ['g', '6'], ['h', '7'],
['i', '8'], ['j', '9'], ['k', '10'], ['l', '11'], ['m', '12'], ['n', '13'], ['o', '14'], ['p',
'15'], ['q', '16'], ['r', '17'], ['s', '18'], ['t', '19'], ['u', '20'], ['v', '21'], ['w',
'22'], ['x', '23'], ['y', '24'], ['z', '25']]

client :  nom          10
         prenom       11

adresse :  adresse     12
         cp           NONE
         commune      13
         section_cadastrale 8
         ancienne_adresse

rdv :
facturation 0
date_facturation none
commentaire 5
type_controle 1
num_dossier 2
date_controle 3
type_traitement none
type_installation 20
rejet_inf none
conformite NONE
impact 7
type_RPQS 6
adresse_facturation 18
cp_facturation 19
commune_facturation 20
nom_proprietaire 16
prenom_proprietaire 17
```

Le script d'acquisition des données a été conçu de manière à ce que l'équipe informatique puisse le prendre en charge les années à venir.

Le tout accompagné d'un PDF explicatif sur comment extraire et mapper les données depuis Excel.

```
1 import mysql.connector
2
3 def dateFormatXX( LaDate):
4     timeFormat = "2024-01-01" #date par default a specifier dans chaque fichier (année par default si format non compatible ) format yyyy-mm-dd !!
5     if LaDate.replace("/", "").isdigit() and len(LaDate.split("/")[0]) == 2 and len(LaDate.split("/")[1]) == 2 and len(LaDate.split("/")[2]) == 4:
6         time = LaDate.split("/")
7         timeFormat = "-".join(time[::-1])
8     return timeFormat
9
10 def adresseSpliteur(adresse):
11     listeAdresse = adresse.split(" ")
12     #Si le premier est de type numérique
13     if listeAdresse[0].isdigit() :
14         return listeAdresse[0], " ".join(listeAdresse[1:])
15     #Sinon retourne un numero de voie vide
16     return "", adresse
17
18
19 fileR = open("c:\\Projetsonia\\2024\\2024.txt", "r", encoding = "utf8") #chemin d'accès du fichier (CSV sans l'entête copier dans un fichier .txt)
20 connection_params= {
21     'host' : "localhost",
22     'user' : "root",
23     'password' : "",
24     'database' : "sonia"
25 }
26
27 lines = fileR.readlines()
28 for n, line in enumerate(lines) :
29     line = line.replace("'", "")
30     line = line.replace(",",".")
31     line = line.replace("","")
32     liste = line.split(";")
33     timeFormat = dateFormatXX(liste[15]) #date_controle !!
34     adresseRdvNum, adresseRdvVoie = adresseSpliteur(liste[20]) #adresse !!
35     stringClientSelect = "select id from client where nom = '" + liste[17] + "' and prenom = '" + liste[18] + "' LIMIT 0, 1"
36     stringClientInsert = "insert into client (nom, prenom) \
37         values ('" + liste[17] + "', '" + liste[18] + "')"
38     stringAdresseSelect = "select id from adresse as a where a.numero = '" + adresseRdvNum + "' and a.adresse = '" + adresseRdvVoie + "' and a.cp
39         = '" + liste[21] + "' and a.section_cadastrale = '" + liste[19] + "' LIMIT 0, 1"
40     stringAdresseInsert = "insert into adresse (numero, adresse, cp, commune, section_cadastrale) \
41         values ('" + adresseRdvNum + "', '" + adresseRdvVoie + "', '" + liste[21] + "', '" + liste[22] + "', '" + liste[19] + "')"
42     stringRdvInsert = "INSERT INTO rendez_vous (client_id, adresse_id, facturation, date_facturation, commentaire, type_controle, num_dossier,
43         date_controle, type_traitement, type_installation, rejet_inf, conformite, impact, type_rpcs, adresse_facturation, cp_facturation,
44         commune_facturation, nom_proprietaire, prenom_proprietaire ) Values \n "
45     with mysql.connector.connect(**connection_params) as db :
46         with db.cursor() as c:
47             c.execute(stringClientSelect)
48             id_client = c.fetchone()
49             if id_client is None :
50                 c.execute(stringClientInsert)
51                 db.commit()
52             c.execute(stringClientSelect)
53             id_client = c.fetchone()
54             c.execute(stringAdresseSelect)
55             id_adresse = c.fetchone()
56             if id_adresse is None :
57                 c.execute(stringAdresseInsert)
58                 db.commit()
```

```
        c.execute(stringAdresseSelect)
        id_adresse = c.fetchone()
        listeRdv = [ id_client[0], # client_id
                    id_adresse[0], # adresse_id
                    liste[3], # facturation
                    liste[1], # date_facturation
                    liste[11], # commentaire
                    liste[13], # type_controle
                    liste[14], # num_dossier
                    timeFormat, # date_controle
                    liste[23], # type_traitement
                    liste[24], # type_installation
                    liste[25], # rejet_inf
                    liste[26], # conformite
                    liste[29], # impact
                    liste[27], # type_rpcs
                    liste[30], # adresse_facturation
                    liste[31], # cp_facturation
                    liste[32], # commune_facturation
                    liste[17], # nom_proprietaire
                    liste[18] # prenom_proprietaire
                ]
        stringRdvInsert += "(" + ",".join([ "'" + str(k) + "'" for k in listeRdv]) + ")"
        c.execute(stringRdvInsert)
        db.commit()
```

Pour le développement, un script batch Windows a été mis en place pour exécuter tous les scripts Python d'import de données.

```
scriptloader.bat
1 @echo off
2 REM Exécution des scripts Python
3
4 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/trigger.py
5 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2024/finalsript.py
6 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2023/finalsript.py
7 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2022/finalsript.py
8 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2021/finalsript.py
9 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2020/finalsript.py
10 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2019/finalsript.py
11 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2018/finalsript.py
12 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2017/finalsript.py
13 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2016/finalsript.py
14 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2015/finalsript.py
15 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2014/finalsript.py
16 C:/Users/nfrere/AppData/Local/Programs/Python/Python312/python.exe c:/Projetsonia/2013/finalsript.py
17
18 pause
```

File	SQL Script
2021	DROP PROCEDURE IF EXISTS befDernierRdvClient ;
2022	CREATE PROCEDURE befDernierRdvClient (IN 'idrdv' INTEGER) BEGIN DECLARE id_clientDer INTEGER; SELECT id INTO id_clientDer FROM client WHERE dernier_rdv_id = idrdv; IF ( id_clientDer is not null ) THEN UPDATE client SET dernier_rdv_id = (SELECT id from rendez_vous where client_id = id_clientDer and id != idrdv and date_controle = (SELECT MAX(date_controle) from rendez_vous where client_id = id_clientDer and id != idrdv) LIMIT 0, 1) WHERE id = id_clientDer; END IF; END
2023	
2024	
divers	DROP PROCEDURE IF EXISTS befDernierRdvAdresse ;
Pour Nicolas	CREATE PROCEDURE befDernierRdvAdresse (IN 'idrdv' INTEGER) BEGIN DECLARE id_adresseDer INTEGER; SELECT id INTO id_adresseDer FROM adresse WHERE dernier_rdv_id = idrdv; IF ( id_adresseDer is not null ) THEN UPDATE adresse SET dernier_rdv_id = (SELECT id from rendez_vous where adresse_id = id_adresseDer and id != idrdv and date_controle = (SELECT MAX(date_controle) from rendez_vous where adresse_id = id_adresseDer and id != idrdv) LIMIT 0, 1) WHERE id = id_adresseDer; END IF; END
Projet	
SAVEBDD	
bddCreate.txt	
bddIndex.txt	DROP PROCEDURE IF EXISTS aftDernierRdvClient ;
config Libre saut ligne	CREATE PROCEDURE aftDernierRdvClient (IN 'idrdv' INTEGER, IN 'idclient' INTEGER) BEGIN DECLARE id_clientMax INTEGER; DECLARE id_dernierRdvAct INTEGER; SELECT id INTO id_clientMax from rendez_vous where client_id = idclient and date_controle = (SELECT MAX(date_controle) from rendez_vous where client_id = idclient) LIMIT 0, 1; UPDATE client SET dernier_rdv_id = id_clientMax where id = idclient; END
config password	
mcd.lo1	
mcd	DROP PROCEDURE IF EXISTS aftDernierRdvAdresse ;
OldFacture-master	CREATE PROCEDURE aftDernierRdvAdresse (IN 'idrdv' INTEGER, IN 'idadresse' INTEGER) BEGIN DECLARE id_adresseMax INTEGER; DECLARE id_dernierRdvAct INTEGER; SELECT id INTO id_adresseMax from rendez_vous where adresse_id = idadresse and date_controle = (SELECT MAX(date_controle) from rendez_vous where adresse_id = idadresse) LIMIT 0, 1; UPDATE adresse SET dernier_rdv_id = id_adresseMax where id = idadresse; END
scriptloader	
scriptloader	
scriptSQLmodel	DROP TRIGGER IF EXISTS insDernierRdvEntite;

Des déclencheurs (triggers) ont été implémentés pour que chaque client et chaque adresse aient l'ID de leur dernier rendez-vous dans leur table respective.

Le site utilise le bundle knp\_paginator pour la pagination. Pour intégrer knp\_sort, qui fait partie du même bundle, les requêtes d'accès à la base de données doivent être rédigées en DQL (Doctrine Query Language).

Date de contrôle	Status	Status Dossier	Type du contrôle	N°Dossier	Client	Adresse	Commune	Cadastre	Prochaine visite	EF EDN
17/01/2019	Facturation	Dossier					la ferte			<input type="checkbox"/> EF <input type="checkbox"/> EDN
17/01/2021										<input type="button" value="filtrer"/>
01/01/2020	Réalisé	VALIDE	Vente	005 V2020-005		LA FERTE EN OUCHE 61550	434 ZD 5		Non renseigné	
01/01/2020	Réalisé	VALIDE	Vente	74 V2020-074		LA FERTE EN OUCHE 61550	47 ZE 17		Non renseigné	
01/01/2020	Réalisé	VALIDE	Vente	128 V2020-128		LA FERTE EN OUCHE 61550	205 D 403		Non renseigné	
01/01/2020	Réalisé	VALIDE	Vente	031 N2020-031		LA FERTE EN OUCHE 61550	136 F 136151163		Non renseigné	
01/01/2020	Réalisé	VALIDE	Examen Prélabile	014 N2020-014		LA FERTE EN OUCHE 61550	047 ZE 1112		Non	

Utilisation de knp\_sort dans la vue :

```
<th class="text-left align-top" scope="col" {% if pagination.isSorted('r.type_controle') %} class="text-left align-top sorted"{% endif %}>{{ knp_pagination_sortable(pagination, 'Type du contrôle', 'r.type_controle') }}</th>
<div class="form-group mr-1 mb-2">
  <input type="text" class="input-group input-group-lg" name="typeCon" value="{% if typeCon|default and typeCon != '***' %}{{ typeCon }}{% endif %}">
</div>
</th>
<th class="text-left align-top" scope="col" {% if pagination.isSorted('r.num_dossier') %} class="text-left align-top sorted"{% endif %}>{{ knp_pagination_sortable(pagination, 'N°Dossier', 'r.num_dossier') }}</th>
<div class="form-group mr-1 mb-2">
  <input type="text" class="input-group input-group-lg" name="numDossier" value="{% if numDossier|default and numDossier != '***' %}{{ numDossier }}{% endif %}">
</div>
</th>
```

Requêtes en DQL

Ex rendez-vous :

```
public function paginationQuery()
{
    return $this->createQueryBuilder('r')
        ->leftjoin('r.client', 'c')
        ->leftjoin('r.adresse', 'a')
        ->orderBy('r.date_controle', 'DESC')
        ->getQuery()
    ;
}
```

## Les requêtes complexe avec filtre sont directement intégrés au contrôleur

```
#[Route('/rdv/{dataConDeb}/{dataConFin}/{typeCon}/{numDossier}/{clientNom}/{Adresse}/{abCommune}/{abCommuneExaF}/{abCadastr}/prorioCommF/{prorioCommF}', name: 'rdv.findAllContainReq', methods: ['GET'])]
public function findAllContainReq(RendezvousRepository $RendezvousRepository,
Request $request, PaginatorInterface $paginator,
string $dataConDeb, string $dataConFin, string $typeCon, string $numDossier, string $clientNom,
string $Adresse, string $abCommune, string $abCommuneExaF, string $abCadastr, string $prorioCommF, string $prorioCommuneF): Response{
    $Interup = false;
    $requete = $RendezvousRepository->createQueryBuilder('r')
        ->join('r.client', 'c')
        ->join('r.adresse', 'a')
        ->where('i = :triche')
        ->setParameter('triche', 1);
    if($dataConDeb != ""){
        $requete->andWhere("r.data_controlle >= :dataConDeb")
        ->setParameter('dataConDeb', $dataConDeb);
        $Interup = true;
    }
    if($dataConFin != ""){
        $requete->andWhere("r.data_controlle <= :dataConFin")
        ->setParameter('dataConFin', $dataConFin);
        $Interup = true;
    }
    if($typeCon != ""){
        $requete->andWhere("r.type_controlle like :typeCon")
        ->setParameter('typeCon', 'X.'.$typeCon.'X');
        $Interup = true;
    }
    if($numDossier != ""){
        $requete->andWhere("r.num_dossier like :numDossier")
        ->setParameter('numDossier', 'X.'.$numDossier.'X');
        $Interup = true;
    }
    if($clientNom != ""){
        $requete->andWhere("c.nom like :clientNom")
        ->setParameter('clientNom', 'X.'.$clientNom.'X');
        $Interup = true;
    }
    if($Adresse != ""){
        $requete->andWhere("a.adresseVisite like :Adresse")
        ->setParameter('Adresse', 'X.'.$Adresse.'X');
        $Interup = true;
    }
    if($abCommune != ""){
        $requete->andWhere("a.commune like :abCommune")
        ->setParameter('abCommune', 'X.'.$abCommune.'X');
        $Interup = true;
    }
    if($abCommuneExaF != ""){
        $requete->andWhere("a.commune = :abCommuneExaF")
        ->setParameter('abCommuneExaF', $abCommuneExaF);
        $Interup = true;
    }
    if($abCadastr != ""){
        $requete->andWhere("a.section_cadastrale like :abCadastr")
        ->setParameter('abCadastr', 'X.'.$abCadastr.'X');
        $Interup = true;
    }
    if($prorioCommF != ""){
        $requete->andWhere("r.nom_proprietaire like :prorioCommF")
        ->setParameter('prorioCommF', 'X.'.$prorioCommF.'X');
        $Interup = true;
    }
    if($prorioCommuneF != ""){
        $requete->andWhere("r.commune_facturation like :prorioCommuneF")
        ->setParameter('prorioCommuneF', 'X.'.$prorioCommuneF.'X');
        $Interup = true;
    }
    if($Interup){
        return $this->redirectToRoute("rdv.index");
    }
    $paginator = $paginator->paginate(
        $requete->getQuery(),
        $request->query->get('page', 1),
        20
    );
    $trie = 1;
    return $this->render('rendez_vous/index.html.twig', [
        'paginator' => $paginator,
        'dataConDeb' => $dataConDeb,
        'dataConFin' => $dataConFin,
        'typeCon' => $typeCon,
        'numDossier' => $numDossier,
        'clientNom' => $clientNom,
        'Adresse' => $Adresse,
        'abCommune' => $abCommune,
        'abCommuneExaF' => $abCommuneExaF,
        'abCadastr' => $abCadastr,
        'prorioCommF' => $prorioCommF,
        'prorioCommuneF' => $prorioCommuneF,
        'trie' => $trie
    ]);
}
```

Bien que knp filter propose à peu près les même fonction que les filtres actuels, le choix a été fait de conserver les filtres « fait main » pour permettre de les avoir dans l'URL.

← → ↻ Non sécurisé 192.168.1.100/spanc/rdv/\*\*/\*\*/vente/\*\*/\*\*/route/%2Taigle/\*\*/\*\*/\*\*/\*\*/

● Adresse ● PhpSpreadsheet Do... ● PHPSpreadSheet Be... ● Managing spreadsh... | Tous les favoris

Adresse Client Rendez-vous Mot de passe Logout

# Rendez-vous index

Export Xlsx Ajouter un Rendez-vous

Date de contrôle	Type du contrôle	N°Dossier	Client	Adresse	Commune	Cadastr S	Nom propriétaire	Commune propriétaire	Prochaine visite	
jj/mm/aaaa	<input type="checkbox"/>			route	l'algie					filtrer
jj/mm/aaaa	<input type="checkbox"/> vente				nom exact					



Ainsi les filtres peuvent être sauvegardés et échangés. Et de plus, de la sorte les filtres ont plus de modularité (having etc .. )

Exemple dans ClientController :

```
}
$requete->groupBy('c.id')
    ->orderBy('c.nom', 'ASC');
if($nbRdvF != "**"){
    $requete->having("count(r.id) = :nbRdvF")
    ->setParameter('nbRdvF', $nbRdvF);
    $interup = true;
}
if(!$interup){
    return $this->redirectToRoute('client.index');
}
$pagination = $paginator->paginate(
    $requete->getQuery(),
    $request->query->get('page', 1),
    20,
    array('wrap-queries'=>true)
);
$trie = 1;
```

Plusieurs thèmes ont été intégrés au projet pour pouvoir les présenter à l'équipe qui l'utilisera.

```
PS C:\Projetsonia\Projet> git branch
* Spacelab
  darkly
  main
  slate
  solar
  yeti
```

Adresse Client Rendez-vous Logout

## Adresse index

Export Xlsx Ajouter une adresse

Prochaine visite Dernier Rdv a cette adresse

Adresse Code postal Commune Cadastre Client a la derniere visite Nombre de contrôles filtrer

Non renseigné	14/06/2017	1 visite
Non renseigné	15/06/2017	1 visite
Non renseigné	30/06/2017	1 visite



```
PS C:\Projetsonia\Projet> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

## Mise en place de l'export au format Excel :

Export Xlsx

Installation du bundle phpspreadsheet pour exporter les données au format Xlsx (<https://phpspreadsheet.readthedocs.io/en/latest/>).

Depuis l'index des clients, adresses et rendez-vous à partir du moment où un champ de recherche est renseigné, il est possible d'exporter le résultat exploitable au format Excel. Le noms des fichiers sont générés par les champs de recherche.

	Client-__-__-__-les-__-__	Feuille de calcul Microsoft Excel	80 Ko
	Adresse-__-__-2015-01-23-2016-11-23-__-__-RAI-__-__	Feuille de calcul Microsoft Excel	19 Ko
	Rdv-2013-01-22-2018-01-22-vente-__-__-__-__-__	Feuille de calcul Microsoft Excel	45 Ko

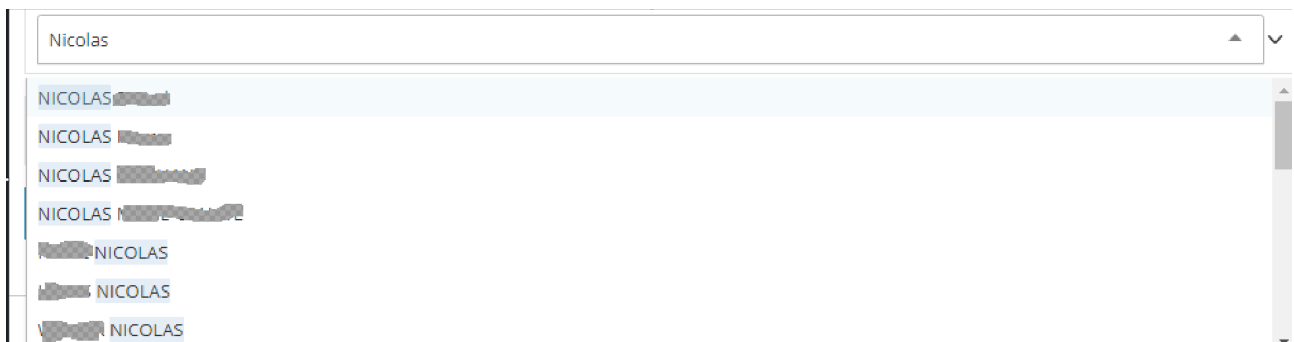
Cela permet à l'équipe du service d'organiser de nouveau rendez-vous en filtrant par date de dernière visite, ville, cadastre etc .. et de pouvoir rester sur un logiciel qu'ils maîtrisent : Microsoft Excel.

```
$contentType = 'application/vnd.openxmlformats-officedocument.spreadsheetml.sheet';
$writer = new Xlsx($file);
$response = new StreamedResponse();
$filename = implode('-', ['Rdv', $dateConDeb, $dateConFin, $typeCon, $numDossier, $clientNom, $adresse, $aDcommune, $aDcommuneExaF, $aDcadastre, $prprioNomF, $prprioCommuneF]);
$response->headers->set('Content-Type', $contentType);
$response->headers->set('Content-Disposition', 'attachment;filename="'.$filename.'.xlsx"');
$response->setPrivate();
$response->headers->addCacheControlDirective('no-cache', true);
$response->headers->addCacheControlDirective('must-revalidate', true);
$response->setCallback(function() use ($writer) {
    $writer->save('php://output');
});
return $response;
}
```

### Ajout du bundle autoCompleteBundle :

```
Symfony\WebpackEncoreBundle\WebpackEncoreBundle::class => ['all' => true],
Symfony\UX\StimulusBundle\StimulusBundle::class => ['all' => true],
Symfony\UX\Autocomplete\AutocompleteBundle::class => ['all' => true],
```

Utilisation du bundle ux\_autocompleteBundle pour pouvoir gérer les grandes quantités des données dans les formulaires.




Dans le formulaire des rendez-vous :

```
->add('client', EntityType::class, [
    'class' => Client::class,
    'query_builder' => function (ClientRepository $er): QueryBuilder {
        return $er->createQueryBuilder('c')
            ->orderBy('c.nom', 'ASC');
    },
    'required' => false,
    'placeholder' => 'client',
    'autocomplete'=>true
])
->add('adresse', EntityType::class, [
    'class' => Adresse::class,
    'query_builder' => function (AdresseRepository $er): QueryBuilder {
        return $er->createQueryBuilder('a')
            ->orderBy('a.commune', 'ASC')
            ->addOrderBy('a.adresseVisite', 'ASC');
    },
    'required' => false,
    'placeholder' => 'adresse',
    'autocomplete'=>true
])
->add('submit', SubmitType::class, [
    'label' => 'Valider'
```

## Ajout de l'identification pour accéder au service :

Adresse Client Rendez-vous



SPANC CDC PAYS DE L'AIGLE

Nom utilisateur

Mot de passe

Sign in

Le site n'étant destiné à être accessible qu'en local, l'algorithme choisi pour hasher les mots de passe est assez simple.

```
config > packages > ! security.yaml
1  security:
2      enable_authenticator_manager: true
3      # https://symfony.com/doc/current/security.html#loading-the-user-the-user-provider
4      password_hashers:
5          #Symfony\Component\Security\Core\User\PasswordAuthenticatedUserInterface:
6              #algorithm: sha256
7              #encode_as_base64: false
8          App\Entity\User:
9              algorithm: sha256
10             encode_as_base64: false
11      providers:
12          # used to reload user from session & other features (e.g. switch_user)
13          app_user_provider:
14              entity:
15                  class: App\Entity\User
16                  property: username
17  firewall:
```

Le but était de pouvoir créer des utilisateurs ou modifier leurs mots de passe directement depuis la base de données (par exemple : « insert into user (username, password) values ('utilisateur', sha2('password', 256)) »). Cependant, malgré que les changements d'algorithmes étaient visibles (tests effectués avec la création d'utilisateurs depuis le fixturesBundle et userPasswordHasherInterface), les résultats ne correspondaient pas à ce que l'algorithme devait encoder.

Pour remédier à cela, depuis le serveur Symfony, il est possible en ligne de commande de générer le hash d'un mot de passe.

```
PS C:\Projetsonia\Projet> php bin/console security:hash-password leMotDepasse

Symfony Password Hash Utility
=====

! [NOTE] The command will take care of generating a salt for you. Be aware that some hashers advise to let them
        generate their own salt. If you're using one of those hashers, please answer 'no' to the question below.
!       Provide the 'empty-salt' option in order to let the hasher handle the generation itself.

Confirm salt generation ? (yes/no) [yes]:
> no

-----
Key                Value
-----
Hasher used        Symfony\Component\PasswordHasher\Hasher\MessageDigestPasswordHasher
Password hash      6024229fffababb73cd45e9c0431eee6abbf036ee7b7c7751686d72ad79d6499
-----

[OK] Password hashing succeeded
```

Les mots de passe par défaut des utilisateurs ont donc été créés de cette manière.

```
Insert into user (username, roles, password) VALUES ('sonia', '["ROLE_USER"]',
'528ad1c28a70e18cb191896a04a18912433a4461f962ff556a6514466442dadb');
```

Ici pour l'exemple 528ad1c28a70e18cb191896a04a18912433a4461f962ff556a6514466442dadb correspond au hash de « **password** » depuis Symfony.


**Configuration du pare-feu (firewall) pour que le site ne soit accessible qu'aux utilisateurs enregistrés :**

```
firewalls:
    dev:
        pattern: ^/(_(profiler|wdt)|css|images|js)/
        security: false
    main:
        lazy: true
        provider: app_user_provider
        form_login:
            login_path: app_login
            check_path: app_login
            default_target_path : rdv.index
        logout:
            path: app_logout
```

```
access_control:
    # - { path: ^/admin, roles: ROLE_ADMIN }
    - { path: ^/csv, roles: ROLE_USER }
    - { path: ^/rdv, roles: ROLE_USER }
    - { path: ^/adresse, roles: ROLE_USER }
    - { path: ^/client, roles: ROLE_USER }
```

## Création d'un formulaire pour changer de mot de passe :

Adresse Client Rendez-vous Mot de passe Logout



### Formulaire de modification de mot de passe

Utilisateur : admin


Mot de passe

Nouveau mot de passe

Confirmation du Nouveau mot de passe

[Changer mon mot de passe](#)

Adresse Client Rendez-vous Mot de passe Logout



### Rendez-vous index

[Ajouter un Rendez-vous](#)

Le mot de passe a été modifié.

Date de contrôle	Type de contrôle	N°Dossier	Client	Adresse	Commune	Cadastre S	Nom propriétaire	Commune propriétaire	Prochaine visite
//mm/aaaa	<input type="text"/>								
//mm/aaaa	<input type="text"/>				nom exact				

[Ajouter](#)

```
class UserController extends AbstractController
{
    /**
     * This controller allow us to edit user's password
     */
    * @param User $choosenUser
    * @param Request $request
    * @param EntityManagerInterface $manager
    * @param UserPasswordHasherInterface $hasher
    * @return Response
    */
    #[Security("is_granted('ROLE_USER') and user === choosenUser")]
    #[Route('/utilisateur/edition-mot-de-passe/{id}', 'user.edit.password', methods: ['GET', 'POST'])]
    public function editPassword(
        User $choosenUser,
        Request $request,
        EntityManagerInterface $manager,
        UserPasswordHasherInterface $hasher
    ): Response {
        $form = $this->createForm(UserPasswordType::class);

        $form->handleRequest($request);
        if ($form->isSubmitted() && $form->isValid()) {
            if ($hasher->isPasswordValid($choosenUser, $form->getData()['plainPassword'])) {
                $choosenUser->setPlainPassword(
                    $form->getData()['newPassword']
                );

                $choosenUser->setPassword(
                    $hasher->hashPassword(
                        $choosenUser,
                        $choosenUser->getPlainPassword()
                    )
                );
                $choosenUser->setPlainPassword(null);

                $this->addFlash(
                    'success',
                    'Le mot de passe a été modifié.'
                );

                $manager->persist($choosenUser);
                $manager->flush();
                return $this->redirectToRoute('rdv.index');
            } else {
                $this->addFlash(
                    'warning',
                    'Le mot de passe renseigné est incorrect.'
                );
            }
        }

        return $this->render('user/edit_password.html.twig', [
            'form' => $form->createView()
        ]);
    }
}
```

```
class UserPasswordType extends AbstractType
{
    public function buildForm(FormBuilderInterface $builder, array $options): void
    {
        $builder
            ->add('plainPassword', PasswordType::class, [
                'attr' => ['class' => 'form-control'],
                'label' => 'Mot de passe',
                'label_attr' => ['class' => 'form-label'],
                'constraints' => [new Assert\NotBlank()]
            ])
            ->add('newPassword', RepeatedType::class, [
                'type' => PasswordType::class,
                'first_options' => [
                    'attr' => [
                        'class' => 'form-control'
                    ],
                    'label' => 'Nouveau mot de passe',
                    'label_attr' => [
                        'class' => 'form-label'
                    ]
                ],
                'second_options' => [
                    'attr' => [
                        'class' => 'form-control'
                    ],
                    'label' => 'Confirmation du Nouveau mot de passe',
                    'label_attr' => [
                        'class' => 'form-label'
                    ]
                ],
                'invalid_message' => 'Les mots de passe ne correspondent pas.'
            ])
            ->add('submit', SubmitType::class, [
                'attr' => [
                    'class' => 'btn btn-primary'
                ],
                'label' => 'Changer mon mot de passe'
            ]);
    }
}
```

## Changement de version :

Migration de version Symfony vers la 6.4 qui sera maintenue jusqu'en 2027, suppression des bundles inutiles.

```
11 11 "doctrine/doctrine-bundle": "^2.8",
12 12 "doctrine/doctrine-migrations-bundle": "^3.2",
13 13 "doctrine/orm": "^2.14",
14 - "easycorp/easyadmin-bundle": "^4.9",
15 14 "knplabs/knp-paginator-bundle": "^6.2",
16 15 "phpdocumentor/reflection-docblock": "^5.3",
17 16 + "phpoffice/phpspreadsheet": "^2.0",
18 17 "phpstan/phpdoc-parser": "^1.16",
19 18 "sensio/framework-extra-bundle": "^6.1",
20 19 "symfony/apache-pack": "^1.0",
21 - "symfony/asset": "6.2.*",
22 - "symfony/console": "6.2.*",
23 - "symfony/doctrine-messenger": "6.2.*",
24 - "symfony/dotenv": "6.2.*",
25 - "symfony/expression-language": "6.2.*",
26 25 "symfony/flex": "^2",
27 - "symfony/form": "6.2.*",
28 - "symfony/framework-bundle": "6.2.*",
29 - "symfony/http-client": "6.2.*",
30 - "symfony/intl": "6.2.*",
31 - "symfony/mailer": "6.2.*",
32 - "symfony/mime": "6.2.*",
26 + "symfony/form": "^6.4",
27 + "symfony/framework-bundle": "^6.4",
28 + "symfony/http-client": "^6.4",
29 + "symfony/intl": "^6.4",
30 + "symfony/mailer": "^6.4",
31 + "symfony/mime": "^6.4",
33 32 "symfony/monolog-bundle": "^3.0",
34 - "symfony/notifier": "6.2.*",
35 - "symfony/process": "6.2.*",
36 - "symfony/property-access": "6.2.*",
37 - "symfony/property-info": "6.2.*",
38 - "symfony/runtime": "6.2.*",
39 - "symfony/security-bundle": "6.2.*",
40 - "symfony/serializer": "6.2.*",
33 + "symfony/notifier": "^6.4",
34 + "symfony/process": "^6.4",
35 + "symfony/property-access": "^6.4",
36 + "symfony/property-info": "^6.4",
37 + "symfony/runtime": "^6.4",
38 + "symfony/security-bundle": "^6.4",
```

```
39 + "symfony/serializer": "^6.4",
41 40 "symfony/stimulus-bundle": "^2.14",
42 - "symfony/string": "6.2.*",
43 - "symfony/translation": "6.2.*",
44 - "symfony/twig-bundle": "6.2.*",
41 + "symfony/string": "^6.4",
42 + "symfony/translation": "^6.4",
43 + "symfony/twig-bundle": "^6.4",
45 44 "symfony/ux-autocomplete": "^2.12",
46 - "symfony/validator": "6.2.*",
47 - "symfony/web-link": "6.2.*",
45 + "symfony/validator": "^6.4",
46 + "symfony/web-link": "^6.4",
48 47 "symfony/webpack-encore-bundle": "^2.1",
49 - "symfony/yaml": "6.2.*",
48 + "symfony/yaml": "^6.4",
50 49 "twig/extra-bundle": "^2.12|^3.0",
51 50 "twig/twig": "^2.12|^3.0",
52 51 },
+
+ @@ -94,19 +93,19 @@
94 93 "extra": {
95 94 + "symfony": {
96 95 "allow-contrib": true,
97 - "require": "6.2.*",
96 + "require": "^6.4"
98 97 }
99 98 },
100 99 "require-dev": {
101 100 "doctrine/doctrine-fixtures-bundle": "^3.5",
102 101 "fzaninotto/faker": "^1.5",
103 102 "phpunit/phpunit": "^9.5",
104 - "symfony/browser-kit": "6.2.*",
105 - "symfony/css-selector": "6.2.*",
106 - "symfony/debug-bundle": "6.2.*",
103 + "symfony/browser-kit": "6.4.*",
104 + "symfony/css-selector": "6.4.*",
105 + "symfony/debug-bundle": "6.4.*",
107 106 "symfony/maker-bundle": "^1.50",
108 - "symfony/phpunit-bridge": "6.2.*",
109 - "symfony/stopwatch": "6.2.*",
110 - "symfony/web-profiler-bundle": "6.2.*",
107 + "symfony/phpunit-bridge": "6.4.*",
108 + "symfony/stopwatch": "6.4.*",
109 + "symfony/web-profiler-bundle": "6.4.*",
111 110 }
```

## Mise en prod :

Les deux sites sont configuré sur le même serveur avec Apache2.

Deux alias ont été crée pour rediriger les routes quand on se connecte à l'adresse locale du serveur ( 192.168.1.100 /spanc et /jvspj ).

Le documentRoot renseigné est celui de spanc ( /var/www/html/Spanc/public ) pour permettre à nodeJS (seul spanc utilise nodeJS) d'avoir les chemins relatifs relatif de l'application.

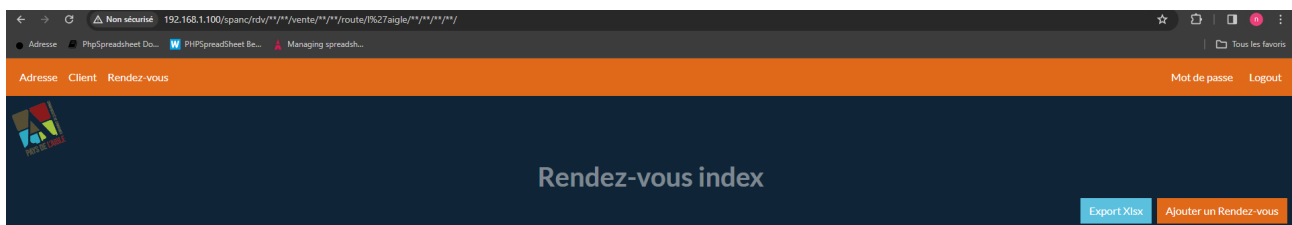
```
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    ServerName jvspj
    ServerAlias jvspj

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/Spanc/public

    <Directory /var/www/html/>
        AllowOverride All
        Order Allow,Deny
        Allow from All
    </Directory>

    Alias /jvspj "/var/www/html/jvspj/public"
    <Directory "/var/www/html/jvspj/public">
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    Alias /spanc "/var/www/html/Spanc/public"
    <Directory "/var/www/html/Spanc/public">
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>
```





Le module d'autocomplexion (ux\_autocomplete) fonctionnant avec le bundle encore, nodeJS a été installé sur le serveur.

```
www-data@jvs-pj:~/html/Spanc$ npm --version
10.2.4
www-data@jvs-pj:~/html/Spanc$ node --version
v21.6.2
```

```
www-data@jvs-pj:~/html/Spanc$ www-data@jvs-pj:~/html/Spanc$ npm run dev
> dev
> encore dev

Running webpack ...

DONE Compiled successfully in 1455ms 9:19:38

6 files written to public/build
Entrypoint app [big] 1.48 MiB = runtime.js 14.6 KiB vendors-node_modules_symfony_stimulus-bridge_dist_index_js-node_modules_tom-select_dist_css_t-a5a5ed.css 34.5 KiB vendors-node_modules_symfony_stimulus-bridge_dist_index_js-node_modules_tom-select_dist_css_t-a5a5ed.js 1.33 MiB app.css 741 bytes app.js 102 KiB
webpack compiled successfully
```

Les deux projet sont fonctionnel et accessible depuis le réseau local de la communauté de communes du Pays de l'Aigles.