



Lycée Turgot  
69, rue de Turbigo  
75141 PARIS



ONISEP  
12, mail Barthélémy Thimonnier  
77185 LOGNES

**Rapport  
de Stage  
Du 28/05/18 au 29/06/18**

*Stagiaire : Estelle GUYOT*

*Encadrant : Philippe DURAND*

*Service : Informatique*



## **Remerciements**

Mes remerciements s'adressent d'abord au DSI Monsieur Bruno ROITEL, pour m'avoir accueillie au sein du service informatique et également à Philippe DURAND qui m'a encadrée tout au long de mon stage.

Je remercie ensuite le pilote technique gérant le numéro vert PARCOURSUP, Monsieur Jean-Christophe CHAILLAC , pour m'avoir accordé sa confiance dans ce projet et pour toute l'aide qu'il m'a apportée afin de me permettre de mieux cerner le travail à réaliser.

De même, je remercie l'administratrice de bases de données, Madame Nathalie VALLAT pour son suivi et son soutien au cours de ce stage.

Mes remerciements vont également à Madame Sandra DACOSTA, pour m'avoir accueilli et accompagné tout au long du stage.

Enfin, merci à toute l'équipe informatique, pour son accueil, sa disponibilité et sa bonne humeur permanente.

## **Table des matières**

<b>1. Présentation de l'entreprise.....</b>	<b>p.1</b>
1.1. Sa mission et son budget	
1.2. Son architecture Informatique	
1.3. Son service informatique	
1.4. Le principe de la veille technologique	
<b>2. Le contexte de la mission.....</b>	<b>p.4</b>
2.1. Présentation du projet : Parcoursup	
2.2. La mise en place d'indicateurs statistiques	
2.3. L'environnement de travail	
<b>3. La réalisation des différentes tâches.....</b>	<b>p.6</b>
3.1. L'outil	
3.2. Observation et Interprétation du code	
3.3. Écriture d'un fichier csv (export du résultat HTML obtenu)	
3.4. Générer un PDF (issu du résultat HTML obtenu)	
3.5. Les requêtes SQL	
3.6. Une boucle dans une requête SQL	
<b>4. Conclusion.....</b>	<b>p.10</b>
<b>5. Annexes</b>	

## 1. Présentation de l'entreprise

### 1.1. Sa mission et son budget

L'Office National d'Information Sur les Enseignements et les Professions (ONISEP) est un établissement public à caractère administratif (EPA) dépendant directement du ministère de l'éducation nationale.

L'établissement, qui compte environ 550 employés, est organisé en réseau :

- Les services centraux situés à Lognes en Seine-et-Marne.
- 17 délégations régionales réparties sur 28 sites géographiques correspondant aux académies scolaires (y compris les DROM-COM).

Sa mission consiste à diffuser gratuitement aux élèves et à leurs familles des informations sur les métiers et les formations au travers de divers outils et services (brochures, site internet...).

Son budget annuel s'élève à plus de 40 M€.

28 M€ sont financés par le ministère de l'Éducation Nationale. Le reste du financement provient des ressources propres de l'établissement, comme par exemple la vente de brochure sur le thème de l'orientation. En 2018, le budget global était de 42 M€, dont 2 M€ pour l'investissement, 13 M pour le fonctionnement et enfin 27 M pour les salaires. Le budget du département informatique, hors frais de personnel, s'élève à 2,17 M€ en 2018 dont 670 K€ en investissement et 1,5 M€ en fonctionnement.

M€ ⇒ millions d'euros

### 1.2. Son architecture Informatique

L'architecture informatique de l'ONISEP reflète la structure de l'établissement. Chaque site géographique possède son infrastructure avec routeur, serveur, réseau...

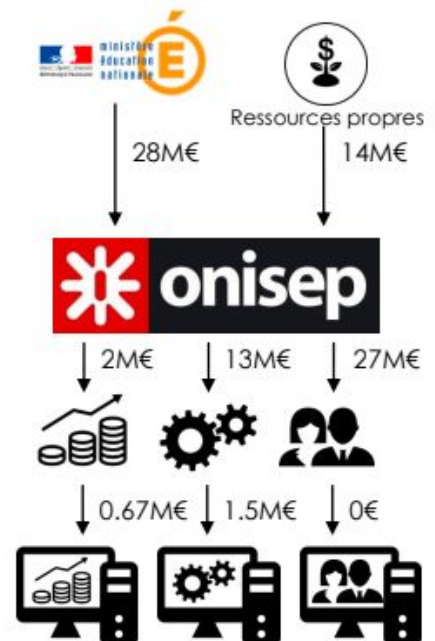
Les services centraux possèdent une architecture virtuelle **VMware** avec plus de 200 serveurs hébergés sur 6 serveurs physiques, des hyperviseurs.

Les systèmes d'exploitation utilisés sur les serveurs sont :

- Debian
- Windows server 2008 R2, 2012 R2 et 2016.

Les systèmes de gestion de bases de données utilisés sont :

- Oracle
- MySQL
- SQL Server.



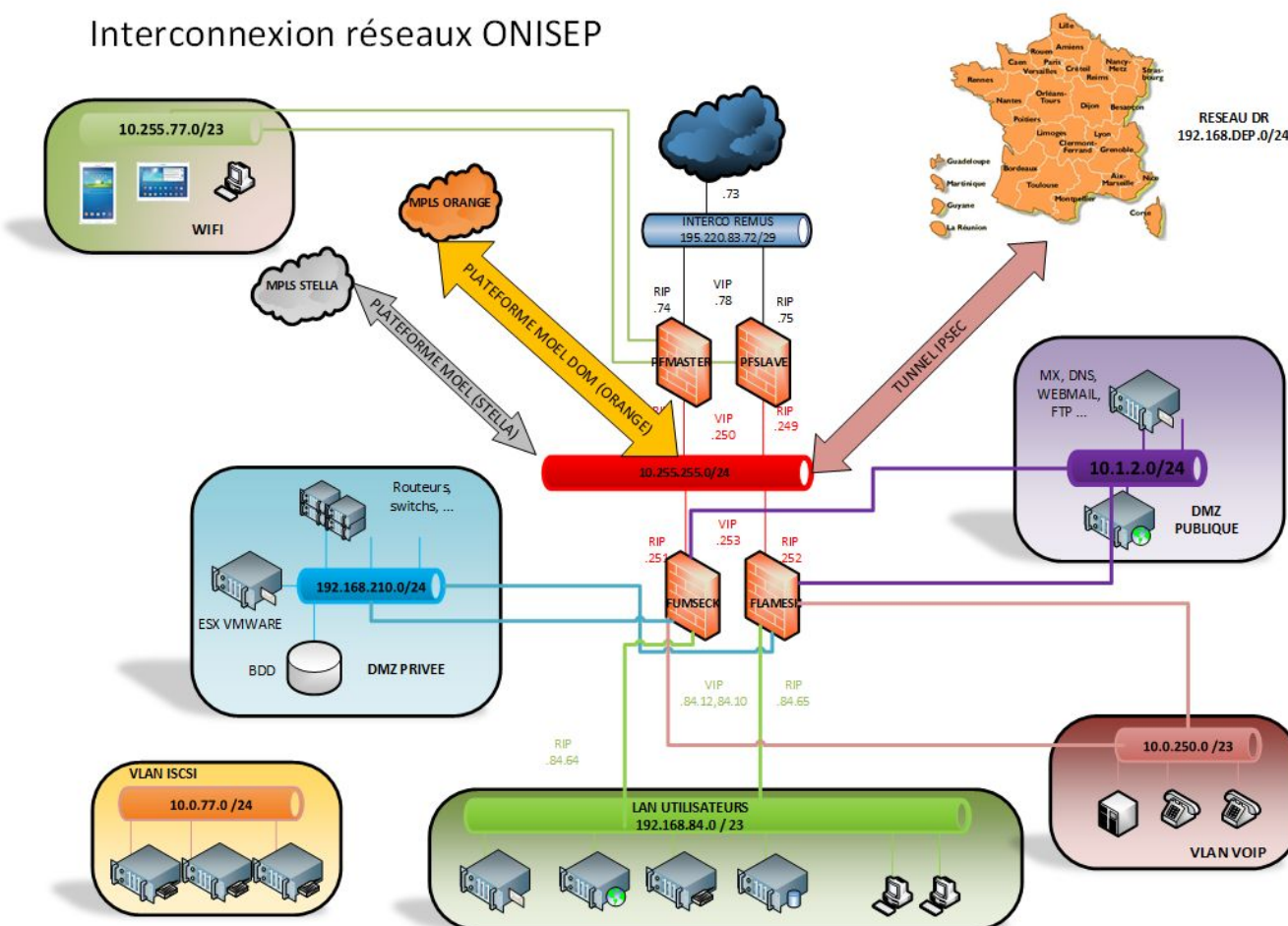
Les sauvegardes du système d'information sont réalisées de façon automatique via un logiciel, **Veeam Backup**.

L'outil de gestion de parc et de gestion d'incidents utilisé est le logiciel Open Source GLPI. Composé de différents modules, **GLPI** permet de gérer l'inventaire des machines et des logiciels (module Fusion Inventory).

L'outil de gestion de déploiement se nomme **SCCM**, il permet d'installer des postes, des applications, des scripts... Les postes informatiques sont déployés par **PXE** à l'aide de **SCCM**. **PXE** permet à une station de travail de démarrer depuis le réseau en récupérant une image de système d'exploitation qui se trouve sur un serveur.

Les postes sont configurés selon les profils métiers de l'utilisateur. Ils sont alors intégrés dans le domaine onisep.fr dans **Active Directory**, outil Microsoft de mise en œuvre des services d'annuaire **LDAP**. Les ordinateurs sont administrés par **GPO** (stratégies de groupe) pour déployer les imprimantes, les lecteurs réseaux ...

### Interconnexion réseaux ONISEP



L'antivirus qui a été mis en place pour les postes de travail est Kaspersky Endpoint Security. La sécurité des postes de travail est gérée via Kaspersky Security Center par l'intermédiaire d'agent d'administration. Ces agents d'administrations effectuent à distance les mises à jour et les analyses des postes.

Une hotline par téléphone et par gestion de ticket GLPI est assurée quotidiennement pour gérer les incidents rencontrés par les utilisateurs.

### 1.3. Son service informatique

Plusieurs corps de métier sont présents dans le service informatique. Les techniciens gèrent le parc informatique. Les administrateurs réseau système configurent tous les serveurs et réseaux. Les chefs de projets mettent en place des applications. Et enfin les administrateurs de bases de données installent, effectuent les maintenances et les sauvegardes des bases de données.

Exemple de projet suivi par un chef de projet : Durant la période de 2015 à 2017, la mise en place d'un référentiel client unique (RCU) a été réalisée dans cette entreprise. Il a été placé au cœur du système d'information et a été interconnecté avec les diverses applications manipulant des données clients (logiciel de gestion commerciale, logiciel marketing, site marchand...).

Les étapes de réalisation du projet ont été tout d'abord :

- Mars - avril 2015 : Rédaction du cahier des charges.
- Juin 2015 : Un appel d'offre a été lancé puis le marché a été attribué à une société prestataire. Les rédactions des spécifications générales ont ensuite été détaillées.
- Développements de l'application.
- Installation des serveurs.
- Livraison des développements.
- Recettes et tests de l'application.
- Une formation des utilisateurs a été organisée.
- Début fin 2016 : Mise en exploitation.
- Depuis 2017, le projet est terminé et l'application est désormais en maintenance et en évolution.

### 1.4. Le principe de la veille technologique

La plupart du personnel informatique de l'ONISEP est de niveau ingénieur ou assistant ingénieur. Le principe de la veille technologique est propre à chacun en fonction de son domaine de compétence. C'est-à-dire que chaque personne par la lecture ou les échanges, via les réseaux sociaux, professionnels ou autres, s'informe et mutualise les nouveautés et en mesure la pertinence sur son domaine d'activité.

## 2. Le contexte de la mission

### 2.1. Présentation du projet : Parcoursup

Depuis janvier 2018, pour le compte du Ministère de l'enseignement supérieur, l'ONISEP gère le numéro vert dédié à PARCOURSUP (le 0800 400 070). Ce numéro est amené à répondre à toutes les questions que les usagers peuvent se poser sur le nouveau dispositif d'accès aux études supérieures. Cette nouvelle fonctionnalité est présente dans toute la France et les DROM-COM (numéros verts distincts de la métropole). Lorsque l'utilisateur appelle ce numéro vert, on lui demande d'indiquer son académie puis de faire un choix compris entre 1 à 3 afin de mieux se rattacher à sa demande.

- Le choix 1 étant un besoin d'information classique d'orientation scolaire.
- Le choix 2 étant un besoin d'information sur la procédure parcoursup.
- Le choix 3 étant un besoin d'action spécifique sur le dossier parcoursup de l'utilisateur.



Si le flux téléphonique est important, la mise en relation avec un conseiller peut ne pas aboutir. Dans ce cas l'utilisateur est soit dissuadé, soit il abandonne de lui-même.

### 2.2. La mise en place d'indicateurs statistiques

Depuis la mise en place de ce dispositif, la direction de l'ONISEP doit rendre compte de l'activité du numéro vert au ministère. À ce titre, le département informatique de l'ONISEP est sollicité pour produire des indicateurs statistiques pertinents. Récemment, il a fallu mettre en place des indicateurs mesurant l'activité en rectorat du choix numéro 3. C'est dans ce contexte qu'il m'a été demandé de travailler sur ces indicateurs.

### 2.3. L'environnement de travail

Dans un premier temps, mon environnement de travail a dû être mis en place. Mon poste client devait posséder une autorisation afin d'accéder aux serveurs contenant les bases de données liées à la gestion du numéro vert. Pour exploiter ces bases de données, l'installation d'outils de requête SQL, DBVisualizer et MySQL Workbench a été nécessaire.

Ensuite, pour mieux comprendre le dispositif technique j'ai eu une mini formation. On m'a présenté le logiciel de gestion de flux téléphonique Ireflet utilisé actuellement par

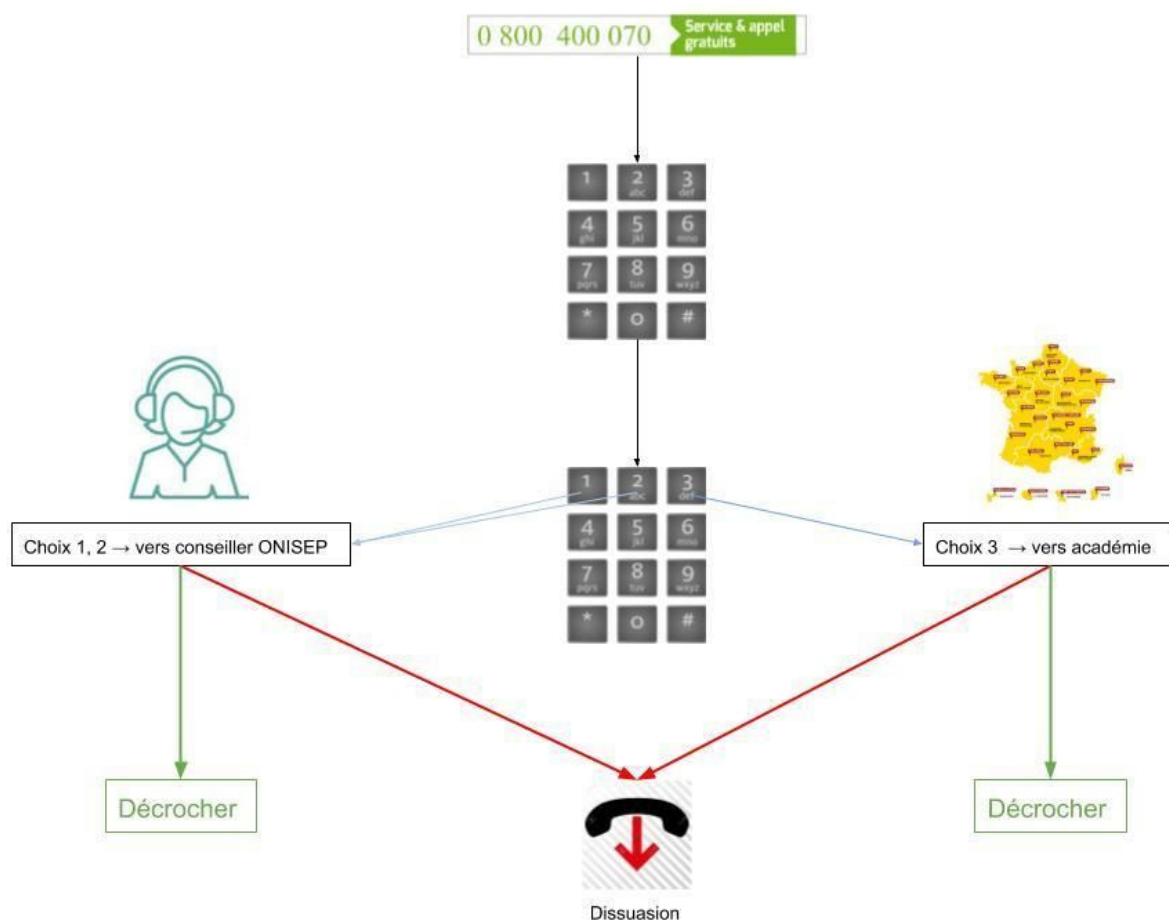


l'ONISEP. On m'a également présenté un outil développé en interne qui permet de présenter et d'extraire des données d'Ireflet sous forme HTML codé en PHP.

Enfin le pilote technique m'a demandé de développer des indicateurs spécifiques selon son besoin à travers l'outil interne HTML/PHP.

Nous avons également eu des réunions en mode projet avec l'ensemble des personnes métier de l'ONISEP gérant le dispositif. Lors de ces réunions, il a été décidé de tenter une consolidation en temps réel des données de tous les serveurs afin de pouvoir fournir des indicateurs en journée. Auparavant la consolidation était effectuée tous les soirs et les chiffres transmis au ministère étaient décalés d'une journée.

Ces tests de consolidation en temps réel ont été concluants, les bandes passantes étant suffisamment résistantes pour le passage de ces différents flux.



### 3. La réalisation des différentes tâches

#### 3.1. L'outil

L'outil sur lequel j'ai dû développer, pour visualiser les indicateurs demandés est un site web codé en PHP, HTML, CSS, JavaScript. Ce site web est hébergé en interne et n'est accessible que par certains employés de l'ONISEP, c'est-à-dire qu'il n'est pas publié sur le web.



#### 3.2. Observation et Interprétation du code

Durant la première semaine, j'ai observé et modifié quelques parties du site afin de mieux comprendre son fonctionnement. La plupart du code était en PHP donc le principe de fonctionnement était assez simple à comprendre. Une petite partie était codée en JavaScript pour un effet dynamique, comme le calendrier ci-contre.

Ceci permet de simplifier la saisie de la date.

Le code est de la forme :

Date début: 25-06-2018

Date fin: 25-06-2018

Type d'acad: 25-06-2018

Mo	Tu	We	Th	Fr	Sa	Su
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1

```
<form action="ParcoursupCalcul-RenduParJour.php" method="POST" class="css" name="form1">
  <fieldset>
    <br />
    <br />
    <label for="datesel">Date : </label>
    <input name="datesel" type="text" id="datesel" size="15" maxlength="10" readonly="readonly" value="<?php echo date("d-m-Y") ?>">
    <script language="JavaScript">
      new tcal({
        // form name
        'formname': 'form1',
        // input name
        'controlname': 'datesel'
      });
    </script>
    <br />
    <br />
    <input type="submit" name="Submit" value="Envoyer">
  </fieldset>
</form>
```

Après la saisie des différents souhaits émis par l'utilisateur on effectue le code suivant sur une autre page :

On récupère à l'aide d'un \$\_POST les différentes variables qui ont été établies à la page précédente. On utilise la fonction : "require" pour

```
require ('accueil.php');
require ("connection_ireflet.inc");
require ('Parcoursup-FonctionRendu.inc.php');
$datesel = $_POST["datesel"];
```

accéder à d'autres codes depuis cette page. La plupart des fichiers où l'on affiche les statistiques ont besoin du fichier accueil.php qui est le menu déroulant et le fichier

connection qui permet de se connecter aux serveurs de la base de donnée. Parfois des fichiers ont besoin d'aller chercher des fichiers contenant des fonctions.

```
17 $sqlDistinctChoix3 = "select count(distinct SessionDigest.CallerAddress) as nombredenumero from SessionDigest where from_unixtime(SessionDigest.Cre
18 $reqDistinctChoix3 = mysqli_query($dbLink, $sqlDistinctChoix3) or die('Erreur SQL !<br>' . $sqlDistinctChoix3 . '<br>' . mysqli_error($dbLink));
```

Pour obtenir les données SQL on indique dans une variable la requête à effectuer. Ensuite on connecte la base de donnée afin de tester le bon fonctionnement de la requête. Pour pouvoir exploiter ces résultats, on a utilisé deux formes différentes :

- Sous forme de tableau, retourne un array :

```
$tabNumDistinctChoix3 = mysqli_fetch_array($reqDistinctChoix3);
```

- Sous forme de boucle, retourne tous types de variables :

À chaque retour de boucle la valeur de la variable actuelle est écrasée et remplacée par la nouvelle valeur. Elle correspond au résultat de la ligne suivante contenu dans la base de donnée.

```
for ($i = 0; $i <= $nbDate; $i++) { // applicati
    $rowmysqliD = mysqli_fetch_assoc($reqDate);
    $datesel = $rowmysqliD['date'];
    $dateJour = $rowmysqliD['jour'];
```

### 3.3. Écriture d'un fichier csv (export du résultat HTML obtenu)

On génère un nom de fichier, si possible non identique à d'autres, d'où la fonction : "time()". Ensuite on ouvre le fichier créé dans un dossier temporaire, la fonction "w" autorise les droits d'écriture dans le fichier. Les données sélectionnées sont alors écrites dans le fichier. Et on ferme le fichier csv.

```
$nomFichier = time();
$filename = $nomFichier . "_export.csv";
$handle = fopen("tempo/" . $filename, "w");

$ligne = "PARCOURSUP|SERVEUR VOCAL AVANT LA

fwrite($handle, $ligne);
$ligne = "";

fclose($handle);
?>
```

### 3.4. Générer un PDF (issu du résultat HTML obtenu)

Lors de mon stage, après la réalisation des statistiques affichées à l'écran en HTML on m'a demandé qu'elles soient imprimables. J'ai pensé que la solution la plus simple serait

de générer un PDF puisque un fichier csv était déjà créé. Pour cela, j'ai trouvé sur le web une classe PHP, se nommant FPDF. (<http://www.fpdf.org/>)

Le PDF sur le web se présente sous la forme :

**Indicateur Académique CHOIX 3 Parcoursup**

Rectorat : Créteil  
Nombre de lignes téléphonique théoriquement dédiées : 2

Vous trouverez ci-dessous un ensemble d'indicateurs sur l'activité du choix3 du numéro vert Parcoursup pour votre académie sur la période du : 04/06/2018 - 28/06/2018

Pour mémoire sur le fonctionnement du choix 3 :

- Lorsqu'un usager sollicite le choix 3 du numéro vert Parcoursup, cet appel n'est transféré au rectorat que s'il y a un « décroché côté rectorat ».
- Actuellement il n'y a pas de différence entre un décroché humain et un décroché automatique. Dans les 2 cas l'appel est transféré.
- S'il n'y a pas de décroché côté rectorat, l'usager entend un message de dissuasion lui signifiant que tous les conseillers sont occupés et qu'ils peuvent soit rappeler ultérieurement, soit passer par la rubrique contact

Date	Nb d'appels ts usagers	Usagers Distincts	Dont usagers transférés	Efficacité du transfert
04/06/2018	405		49	
05/06/2018	478			
06/06/2018	775	54	22	
07/06/2018	75		27	
08/06/2018			24	86%
11/06/2018	146		71	93%
12/06/2018	69		37	
13/06/2018			26	90%
14/06/2018	50			76%
15/06/2018		46	14	
18/06/2018		16		75%
19/06/2018	6	6		
20/06/2018	16		15	
21/06/2018		7		100%
22/06/2018			9	75%
25/06/2018	35	14		
26/06/2018	74	46		
27/06/2018	137		34	
28/06/2018	13	8		

Pour générer le PDF je voulais afficher des variables qui ont été sélectionnées par l'utilisateur auparavant, j'ai pensé faire un formulaire avec l'action \$\_GET. Mais, la fonction : "get" ne fonctionnait pas, sans doute à cause d'un mauvais codage de la balise du formulaire. Pour contrer ce problème, j'ai codé les différentes variables dans l'url. Puis j'ai extrait les variables de la barre URL par un simple \$\_GET afin de pouvoir les retourner dans le PDF. Cette méthode n'est pas sécurisée puisque n'importe quel utilisateur peut modifier la barre url et donc fausser les valeurs. Mais le site étant interne le risque est moins important. Le plus sécurisé aurait été de stocker les variables dans un formulaire du type champ caché, un hidden.

### 3.5. Les requêtes SQL

J'ai appris grâce à l'aide du personnel et d'internet, de nouvelles fonctionnalités dans les requêtes SQL.

```
$sqlnbChoix3 = "SELECT TB.datefield AS Date, IFNULL(count(CallerAddress),0) AS PS_choix_3
FROM
  (select cast(from_unixtime(SessionDigest.CreationDate/1000) as date) as dateSD, CallerAddress from statistic.SessionDigest
  where (cast(from_unixtime(SessionDigest.CreationDate/1000) as date) >= ' . $datedeb . ' '
    and cast(from_unixtime(SessionDigest.CreationDate/1000) as date) <= ' . $datefin . ' '
    and Context like '%choixmenu=dossier%' and Context like '% ' . $typeAca . ' %') as TC
RIGHT JOIN statistic.Calendar as TB ON TC.dateSD = TB.datefield
where TB.datefield between ' . $datedeb . ' ' and ' . $datefin . ' '
group by date";
```



Exemple de fonctionnalités :

- `from_unixtime([nomdelatable].[nomdelacolonne]/1000)`, permet la conversion d'un timestamp en une date plus lisible. Le timestamp (unix) désigne le nombre de secondes écoulées depuis le 1<sup>er</sup> janvier 1970 à minuit UTC précise.
- `cast([expression] as ...)`, est une fonctionnalité qui permet d'extraire une chaîne de caractères dans différents formats (date, heure, ...). Si l'on ne souhaite obtenir seulement la date et non l'heure d'un datetime, le : "as" sera suivi de la valeur date.
- `as [nomdelarequete]`, le : "as" à la fin d'une requête permet de nommer la requête par un alias pour pouvoir effectuer plusieurs requêtes dans une seule requête globale comme ci-dessus.
- `right join`, elle permet de retourner tous les enregistrements de la table de droite sans forcément avoir une correspondance avec celle de gauche.

### 3.6. Une boucle dans une requête SQL

Le seul problème que je n'ai pas réussi à résoudre est le fait qu'une requête SQL est exécutée dans une boucle, ce qui provoque un délai trop long. Ceci est juste un problème de conception de code mais je n'ai pas trouvé le moyen pour le corriger. Le temps d'attente trop long venait du fait que la requête était effectuée pour une variable différente à chaque retour de boucle. Il aurait fallu trouver une requête globale pour ensuite la décomposer par un code PHP, mais nous n'avons pas pu trouver la requête SQL qui convenait.

```

23 => array('Aca' => 'Rouen', 'choix3' => '0', 'choixTrans' =>
24 => array('Aca' => 'Strasbourg', 'choix3' => '0', 'choixTrans
25 => array('Aca' => 'Toulouse', 'choix3' => '0', 'choixTrans' :
26 => array('Aca' => 'Versailles', 'choix3' => '0', 'choixTrans
);
$nbAca = count($tabAca);
for ($i = 1; $i <= $nbAca; $i++) {
    $sqlParAca = "select count(distinct SessionDigest.CallerAddress)
    print($sqlParAca . "<br><br>");
    $reqParAca = mysqli_query($dbLink, $sqlParAca) or die('Erreur S
    $tabParAca = mysqli_fetch_array($reqParAca);
    $tabAca[$i]['choix3'] = $tabParAca['nombredenumero'];
    // var_dump($tabParAca);
    // var_dump($tabAca);

    $sqlParAcaTrans = "select count(SessionDigest.QualificationId)
    $reqParAcaTrans = mysqli_query($dbLink, $sqlParAcaTrans) or die
    print($sqlParAcaTrans . "<br><br>");
    $tabParAcaTrans = mysqli_fetch_array($reqParAcaTrans);
    $tabAca[$i]['choixTrans'] = $tabParAcaTrans['nombredenumero'];
    // var_dump($tabParAcaTrans);
}

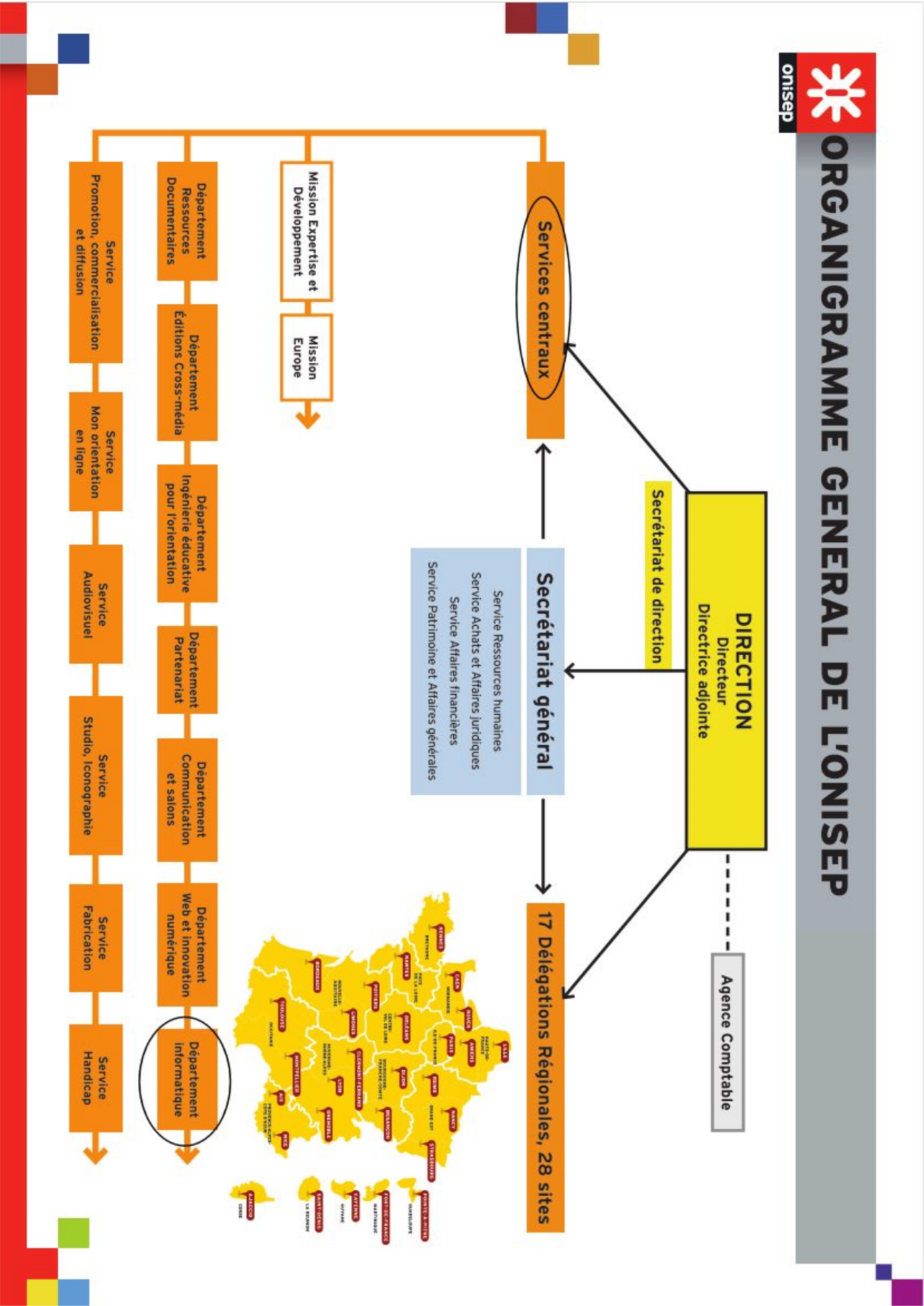
```

## **4. Conclusion**

Ce stage m'a permis d'appliquer les différentes connaissances qui ont été vues durant l'année scolaire. J'ai pu observer la vie en entreprise ainsi que l'organisation des différents services.

Tout au long de mon stage j'ai eu l'opportunité de travailler sur différents aspect du code. Le travail s'est avérée très enrichissant pour mon expérience professionnelle aussi bien dans le domaine technique que celui humain.

5. Annexes



ONISEP - DEPARTEMENT INFORMATIQUE

