

# Rapport de Stage

---

**Découverte du produit NeoGestion et prise en charge de tickets de maintenance produit sur le framework PHP de Néolink**

---

17 avril - 7 juillet 2023

**Entreprise :**

Néolink

22 rue Christophe Colomb

41000 Blois

**Etudiant :**

Corentin RIO

Corentin.rio41@gmail.com

**Maître de stage :**

Ugo LAFAILLE

Ugo.L@neolink.link

# Sommaire

Abstract .....	1
Introduction .....	1
Travail réalisé .....	2
1) Les outils et l'environnement de travail .....	2
2) Mes missions .....	2
a) Tests End to End .....	2
b) Gestion de tickets de maintenance .....	3
Conclusion .....	5
Annexes .....	6

## Abstract

During my Bachelor's degree, I had the opportunity to do a 12-week internship at Néolink's, a company specialised in web platform design. This firm aims at supporting "RSA" recipients and making their professional integration easier. During my internship I was able to perform End to End tests thanks to the Katalon Studio software and to solve maintenance and development tickets on the Néolink PHP framework. I was able to put my skills in web development at the service of this company. This professional experience allowed me to improve my knowledge about web languages but also to discover new approaches through debugging and testing.

## Introduction

Dans le cadre de ma formation en licence informatique, j'ai eu l'occasion d'effectuer un stage de 12 semaines en milieu professionnel. Lors de cette expérience, j'ai pu mettre en œuvre les multiples connaissances et compétences acquises durant mon DUT MMI et mon année de licence. Cela m'a aussi permis de travailler avec de nombreux professionnels comme des développeurs front-end, back-end et des project managers. J'ai choisi d'effectuer mon stage chez Néolink car c'est une entreprise qui propose des solutions web et mobiles, domaines vers lesquelles je souhaiterais m'orienter après mes études.

Créée en 2012 à Blois, Néolink est un éditeur en web social qui propose des plateformes digitales collaboratives. A sa création, Néolink était une start-up qui comptait 8 employés mais elle s'est très vite développée jusqu'à compter plus de 40 employés aujourd'hui (voir annexe 1).

En 2019, l'éditeur de logiciels Berger-Levrault décide de racheter Néolink pour en faire une filiale. L'objectif de Néolink est de faciliter l'embauche aux allocataires du RSA grâce à ses différentes plateformes web et mobiles. Dans ce rapport, je vais tout d'abord vous expliquer les différents outils que j'ai pu utiliser ainsi que mon environnement de travail. Dans un second temps, je vous détaillerai les différentes missions qui m'ont été confiées.

## **Travail réalisé**

### **1) Les outils et l'environnement de travail**

Lors de ce stage, j'ai utilisé des outils avec lesquels j'avais déjà pu me familiariser lors de différents projets universitaires. Par exemple, lorsque j'ai commencé à travailler sur les tickets de maintenance, j'ai utilisé Visual Studio Code qui est un outil de développement adapté pour coder des applications web en HTML, JavaScript et PHP. Toutefois, j'ai pu découvrir des fonctionnalités supplémentaires de cet outil notamment avec des extensions qui m'ont été proposées par les développeurs. Il y a, entre autres, PHP-CS-Fixer qui permet d'indenter le code PHP lorsqu'on sauvegarde le fichier sur lequel on a apporté des modifications. Il y a aussi l'extension PHP Debug qui permet de faire du pas-à-pas (observer les valeurs associées aux variables dans le code et ainsi identifier l'origine des bugs sur les plateformes web).

J'ai aussi utilisé l'application GitHub Desktop pour interagir avec le dépôt distant qui contient le code des plateformes web. J'ai découvert cette application cette année lors des modules de programmation et je l'ai trouvée pratique puisqu'elle permet d'éviter de taper les commandes de Git à la main.

J'ai aussi découvert de nouveaux outils comme le logiciel en ligne Azure DevOps qui permet d'héberger et de gérer du code. Cet outil est notamment utilisé par les chefs de projets pour gérer les différentes branches du projet et pour attribuer des tickets de maintenance aux développeurs.

Puis, j'ai aussi pu apprendre à utiliser le logiciel Katalon Studio, ce dernier permettant d'effectuer des tests End to End. L'avantage de ce logiciel est qu'il propose une interface graphique pour une utilisation plus conviviale. Enfin, j'ai pu prendre en charge des tickets de maintenance en travaillant sur le framework PHP créé par Néolink qui se nomme Phuton.

### **2) Mes missions**

#### **a) Tests End to End**

Lors de ma première semaine de stage, j'ai pu effectuer des tests End to End sous la supervision d'un développeur. L'objectif de ces tests est de vérifier que tous les composants de la plateforme web sont capables de fonctionner de manière optimale dans des situations réelles (ex : se connecter sur la plateforme, créer un CER...). Ces tests automatiques permettront par la suite de vérifier le bon fonctionnement du site après l'ajout de nouvelles fonctionnalités.

Ces tests, effectués de manière automatique sans qu'on prenne trop de temps pour les effectuer à la main, permettent aussi d'éviter les erreurs de manipulation. Par exemple, un testeur pourrait se tromper d'identifiant de connexion et cette dernière serait alors refusée. On pourrait alors penser que le système de connexion ne marche pas alors qu'en réalité la personne qui a effectué le test s'est juste trompée lors de la manipulation des champs.

Pour réaliser les tests End to End, on va tout d'abord mettre au point des scénarii d'usage avec des actions basiques comme se connecter avec des identifiants ou prendre contact avec un recruteur. Les scénarii ayant déjà été créés avant mon arrivée, je n'ai pas eu besoin d'en mettre au point mais ces derniers m'ont permis de mieux comprendre le fonctionnement d'une des plateformes créées par Néolink : NeoGestion.

Ensuite, on va se rendre sur Katalon Studio afin de capturer les éléments directement sur le site en préproduction (pas encore mis en ligne). Par exemple, on va capturer les champs de formulaire ou les boutons (bouton de connexion, bouton pour contacter un recruteur...) puis les enregistrer sur le logiciel. On pourra ensuite utiliser ces captures comme des variables que le logiciel utilisera lors des tests (voir annexe 2)

Le logiciel s'occupe ensuite de suivre les différentes indications qu'on lui donne :

- Cliquer sur le bouton « Connexion / Inscription »
- Remplir le champ « Nom d'utilisateur » avec la valeur « Admin »
- Remplir le champ « Votre mot de passe » avec la valeur « 1234 »
- Cliquer sur le bouton « Connexion »

Pour chaque scénario, le logiciel va effectuer les différentes étapes automatiquement et nous indiquer s'il a rencontré un problème dans l'optique d'optimiser le fonctionnement du site.

## **b) Gestion de tickets de maintenance**

Durant mon stage, de nombreux tickets de maintenance m'ont été confiés. Certains consistaient à résoudre des bugs tandis que pour d'autres, il fallait développer de nouvelles fonctionnalités. J'ai donc pu travailler sur différents langages de programmation intégrés dans le framework Phuton (PHP, JS et SQL). Les tickets sont créés par le service qualité qui s'occupe de tester à la main les différentes plateformes créées par Néolink (NeoJob, NeoGestion et NeoInsertion). Ils sont ensuite mis en ligne sur la plateforme Azure DevOps afin que les développeurs puissent les prendre en charge (voir annexe 3).

Pour chaque ticket, on crée une nouvelle branche sur laquelle on va pouvoir travailler (effectuer nos debugs, corriger le code ou apporter de nouvelles fonctionnalités). En effet, après avoir résolu notre ticket, on va faire une « Pull Request » qui consiste à faire vérifier par d'autres développeurs nos modifications, et s'il n'y a pas de problèmes majeurs qui risquent d'entraîner des erreurs dans le code, alors nos changements sont acceptés et appliqués à la branche principale. Ainsi, nos modifications ne pourront pas impacter le projet directement.

Je vais maintenant vous présenter la démarche de résolution d'un ticket de maintenance. Ce dernier consistait à développer une nouvelle fonctionnalité sur une carte. Il fallait, après avoir indiqué une localisation dans un formulaire de recherche, centrer la carte sur cette localisation (voir annexe 4).

Tout d'abord, on se rend sur la plateforme concernée en local, qui correspond au site en préproduction. On va ensuite, à l'aide de la console du navigateur, observer les méthodes appelées dans le code ainsi que les paramètres et les réponses envoyées lors des différentes requêtes. Dans le cas de mon ticket, lorsque je complète le formulaire et que je clique sur le bouton « Afficher les résultats », j'observe que la méthode « chercherOffres() » est appelée (voir annexe 5). Je vais donc chercher cette méthode dans le code.

Le framework Phuton se compose d'un système de « modules / briques / fonctions ». Par conséquent, pour l'url ci-dessous, on sait d'emblée qu'on va devoir regarder le code contenu dans la brique « trouver » elle-même contenue dans le dossier de module intitulé « insertion ».

- url : <https://job49.home/insertion/trouver/accueil/1129542/>

Je me rends donc dans le module/dossier « insertion » afin de trouver le fichier controleur.php et la méthode « chercherOffres() » appelée. Une fois trouvée, j'effectue un debug à l'aide de l'extension PHP Debug afin de pouvoir observer les données transmises. Grâce à ce « pas-à-pas » j'observe un tableau qui regroupe différentes données dont un objet qui s'intitule « recherche-point » et qui regroupe des données liées à la localisation entrée dans le formulaire (voir annexe 6).

Pour centrer la carte sur la localisation indiquée dans le formulaire, j'avais besoin des coordonnées (latitude et longitude) contenues dans l'objet « recherche-point ». Ne sachant pas comment récupérer ces données, j'ai demandé de l'aide à un développeur qui m'a orienté vers le fichier dashboard.js qui se trouve dans le module « solutions » où sont créées les cartes.

Une fois dans ce fichier, j'ai pu trouver la méthode « traiterReponseRecherche » (voir annexe 7) qui permet de traiter les données envoyées depuis le formulaire. Elle prend en paramètre un objet « reponse ». Pour observer ce que contient cet objet, j'effectue un console.log() puis j'observe dans la console du navigateur le contenu de cet objet (voir annexe 8). Après cela, je récupère les valeurs qui m'intéressent (latitude et longitude) pour centrer ma carte sur les coordonnées souhaitées.

Néolink utilise Leaflet, une bibliothèque JavaScript libre, pour créer des cartes. Après m'être documenté sur leur site, j'ai pu trouver une méthode nommée « setView() » pouvant prendre en paramètre les coordonnées qui étaient désormais à ma disposition mais après plusieurs tentatives je n'ai pas réussi à utiliser cette méthode. J'ai donc demandé de l'aide à un autre développeur qui avait déjà une certaine expérience concernant les cartes Leaflet et qui m'a redirigé vers un autre fichier qui s'intitule « carte.js ».

Ce fichier gère toutes les méthodes relatives aux cartes comme la méthode « obtenirCarte() » qui permet de récupérer la carte de la page pour ensuite faire des actions dessus ou encore la méthode « centrerSurElement() » pour centrer la carte sur un élément (voir annexe 9).

Ainsi, grâce à ces deux méthodes et aux coordonnées récupérées, j'ai pu développer la fonctionnalité attendue (voir annexe 10).

Cependant, mes modifications sur la plateforme Job49 ont engendré une erreur sur la plateforme Job17. En effet, les cartes sont utilisées par toutes les plateformes NeoJob et, comme les cartes sont gérées dynamiquement à l'aide de fichiers JavaScript, on ne peut pas les thématiser. Des modifications sur le module qui gère les cartes risquent donc d'impacter les cartes de toutes les plateformes.

Par exemple, sur la plateforme Job17, les coordonnées n'étaient pas trouvées par la carte, provoquant alors une erreur de chargement sur cette dernière. En effet, la carte de cette plateforme n'utilise pas de formulaire de recherche donc il est impossible de récupérer une quelconque donnée concernant la latitude ou la longitude (voir annexe 11).

Pour résoudre ce problème, j'ai alors ajouté une condition qui vérifie si on possède les coordonnées (latitude et longitude). Si on les possède alors on peut centrer sur l'élément, sinon on ne fait rien (voir annexe 12).

Ce ticket de maintenance, qui était l'un de mes tout premiers, m'a permis de me familiariser avec le framework Phuton ainsi que le debug en JavaScript. A chaque difficulté rencontrée, je n'ai pas hésité à demander de l'aide autour de moi.

## Conclusion

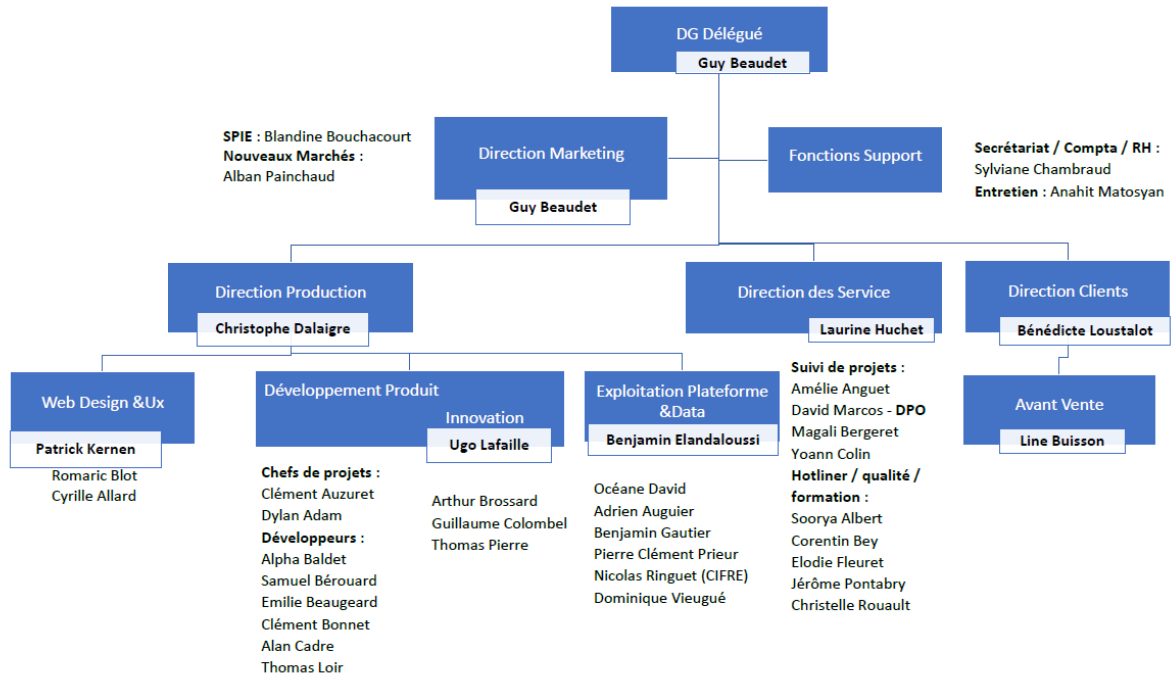
Tout d'abord je tiens à remercier Messieurs Christophe Delaigre et Ugo Lafaille pour avoir retenu ma candidature et me permettre ainsi d'effectuer mon stage au sein de Néolink. Je tiens aussi à remercier tous les développeurs qui m'ont accompagné lors de mes différentes missions et l'ensemble des employés de Néolink pour leur disponibilité. Je voudrais aussi remercier l'ensemble du corps enseignant du Site Universitaire de Blois pour les connaissances qu'ils m'ont apporté tout au long de cette année de formation.

Cette expérience en milieu professionnel m'a permis de découvrir différents métiers dans le domaine de l'informatique (développeur web, testeur, développeur d'applications mobiles...). J'ai gagné en expérience et renforcé mes connaissances dans certains langages de programmation (PHP, JS et SQL). Malgré certaines difficultés au début à appréhender le framework de Néolink, j'ai réussi à réaliser un certain nombre de tickets de maintenance. J'ai ainsi développé mon sens de la réflexion et renforcé mes compétences en développement web. Toutes les différentes missions qui m'ont donc été confiées m'ont alors permis de découvrir certaines des étapes majeures qui constituent un projet informatique (debugs et tests).

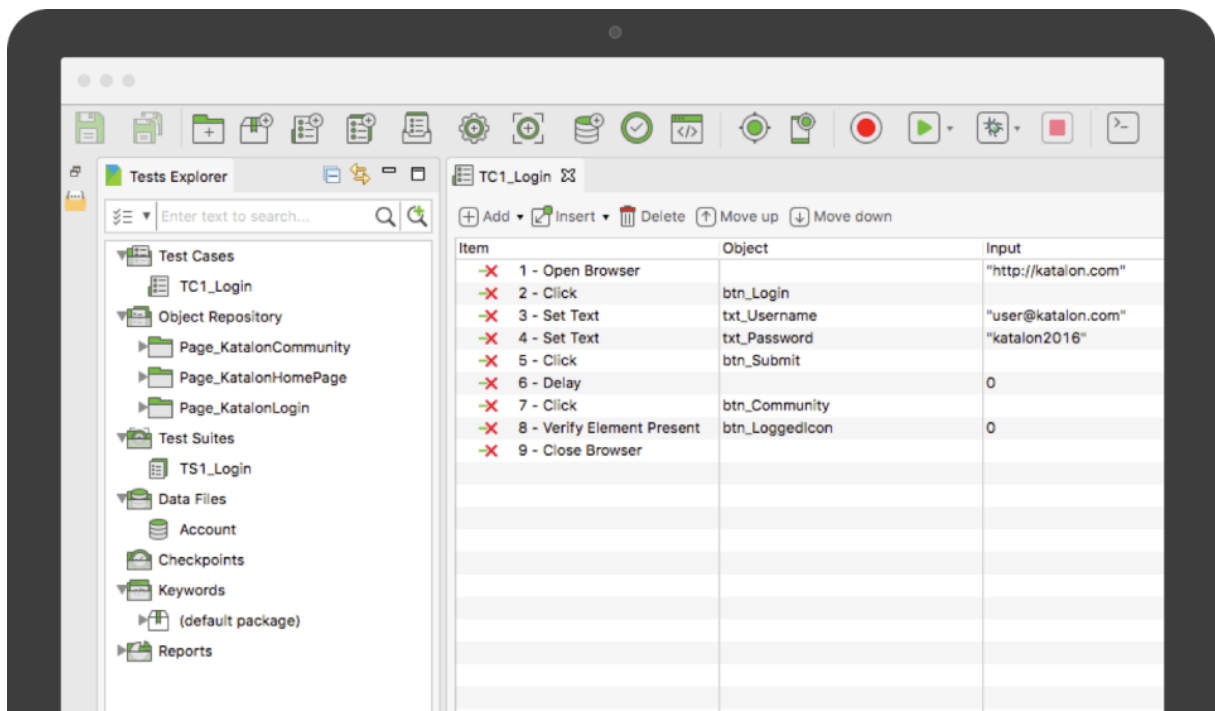
Le développement web est un domaine que je trouve très intéressant mais j'aimerais cependant me spécialiser dans le développement d'applications mobiles. J'ai donc postulé à divers Masters orientés web et mobile pour la rentrée prochaine.

## Annexes

### Annexe 1 : Organigramme de l'entreprise Néolink



### Annexe 2 : Scénario d'un test End to End



## Annexe 3 : Ticket de maintenance sur Azure DevOps

Recently updated | [Back to Work Items](#) 1 of 8

BUG 8720

8720 CD 67 - [TESTCEA2023][CTRSA REORIENTATION] impossibilité de visualiser des odj en préparation

Corentin Riot 0 comments CD 67 X DEMO X gestion\_ep\_sanction X +

Save Follow Settings Refresh Undo Redo

Updated by Corentin Riot: 6h ago

Details History Links (2) Attachments (1)

State: Active Area: Dev Reason: Approved Iteration: Dev\SPRINT S21-S22 2023

Repro Steps

[\[#15955\] \[TESTCEA2023\]\[CTRSA REORIENTATION\] impossibilité de visualiser des odj en préparation : Neolink \(freshdesk.com\)](#)  
  
Bonjour,  
Merci de prendre en compte le rapport d'anomalie ci-après.  
  
**Contexte (rapport initial)**  
Environnement(s) : Démo  
Nature : **Bloquant**  
Profil(s) utilisé(s) : secrétaire insertion (sec\_inser\_cea\_t2 + sec\_inser\_vds\_t2)  
Régression : **Oui**  
URL de l'anomalie : <https://job67-demo-360.neolink.link/ctrsa/default>  
  
**Comportement attendu**  
visualisation des ODJ en préparation créé par un autre profil(ADMIN) du territoire concerné par la secrétaire insertion  
  
**Anomalie(s) constatée(s)**  
impossibilité de visualiser les ODJ en cours du territoire concerné par le profil qui ont été créé par un autre profil (ADMIN)

Planning

Tâche JIRA liée  
NEOL 405  
Story Points  
  
Priority  
1  
  
Effort (Hours)  
Original Estimate  
6  
Remaining  
6  
Completed

Board  
No board found for the current area path

Deployment

To track releases associated with this work item, go to [Releases](#) and turn on deployment status reporting for Boards in your pipeline's Options menu. [Learn more about deployment status reporting](#)

Development  
+ Add link  
dev/crio/8720  
Latest commit mardi  
[Create a pull request](#)

Related Work  
+ Add link  
Parent  
8653 Campagne de Test CEA  
Updated 23/05/2023, Active

System Info

7



## Annexe 4 : Formulaire à remplir pour la carte

maine-et-loire.fr  
**Job49**  
Le Département, acteur de rencontres pros

Contacts Compte

Menu OFFRES POUR L'INSERTION

Vous consultez les informations de  
**Chloe NATIVEL**

**TROUVEZ DES ACTIONS D'INSERTION**

Organisme (Saisissez les premières lettres)  
Saisissez le nom d'une structure  ☐ Avec des places disponibles

Intitulé de l'action

Sous-thème

Où recherchez-vous ?

Autour de chez Chloe NATIVEL Autour de:

Avec quels critères ?

A quelle distance :  Avec ses critères de mobilité : ^

Sa mobilité

Moyen de déplacement :

Sans moyen Vélo Cyclomoteur Moto Transports en commun **Voiture**

Temps de trajet maximum :

**15 mn** 30 mn 45 mn 60 mn 75 mn 90 mn

Afficher les résultats

Actions d'insertion (162)

- FACE ANGERS LOIRE
- RELAIS POUR L'EMPLOI
- RELAIS POUR L'EMPLOI
- RELAIS POUR L'EMPLOI
- RELAIS POUR L'EMPLOI
- CORYLUS Form@tions
- CORYLUS Form@tions
- ADIE

## Annexe 5 : Méthode chercherOffres() affichée dans la console

État	Méthode	Domaine	Fichier	Initiateur	Type	Transfert	Taille	ms
	POST	job49.home	chercherOffres	cdn.js5 (xhr)				0 ms

## Annexe 6 : Objet « recherche-point » observé lors du debug

```

//
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
}

/**
 * Cherche les offres associés au dispositif
 */
public static function chercherOffres()
{
    $retour = ['etat' => 'erreur', 'message' => '<*nok>Aucune offre'];
    $params = self::postRequete();
    $trouver = self::traitement('trouver');

    if (isset($params['parametres'])) {
        $parametres = json_decode($params['parametres']);
        $params['services'] = $parametres->services ?? false;
        $params['espace'] = $parametres->espace ?? false;
        $params['metier'] = $parametres->metier ?? false;
        $params['recherche'] = $parametres->recherche ?? false;
        $params['priorite'] = $parametres->priorite ?? false;
        $params['user_per_clep'] = $parametres->userPerClep ?? false;
    }

    if (!empty($params['adresseParams'])) {
        $tempAdresseParams = $params['adresseParams'];
    }
}

```

```

recherche-point: "{ \"nom\": \"Poitiers 86000\", \"orig_int\": \"nvcv\", \"clep_int\": \"7953\", \"latitude\": \"46.5802596\", \"longitude\": \"0.340196\", \"numero\": \"\", \"voie\": \"\", \"cp\": \"86000\", \"ville\": \"Poitiers\" }"

```

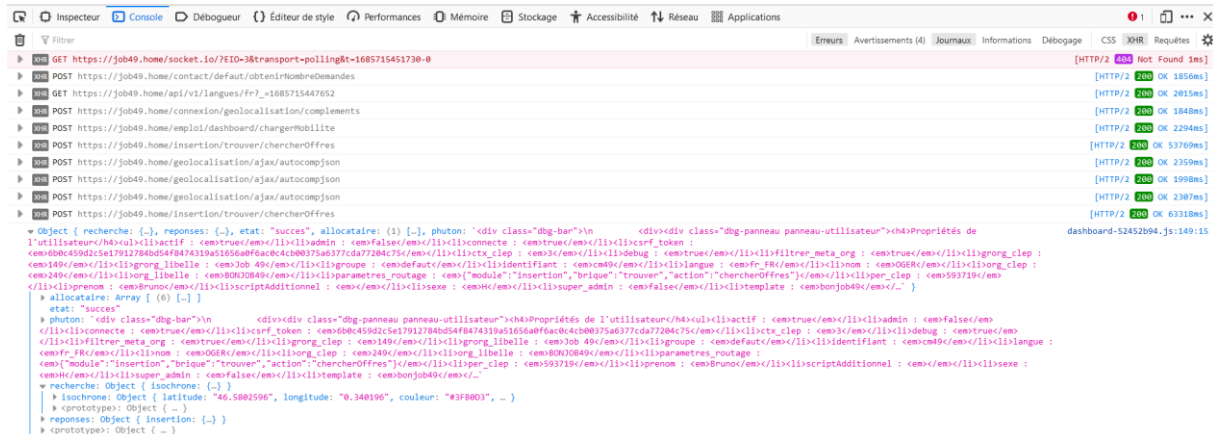
## Annexe 7 : Méthode « traiterReponseRecherche » qui permet de traiter les données envoyées par le formulaire

```

144 //
145 * Traitement de la réponse obtenue
146 * @param reponse
147 */
148 traiterReponseRecherche = function(reponse){
149     console.log(reponse);
150
151     if(cfguti != null && cfg === null){
152         cfg = cfguti;
153     }
154     if(reponse.etat !== 'succes')
155     {
156         let msg = 'Aucun résultat ne correspond à votre recherche';
157         if (typeof reponse.message !== 'undefined')
158         {
159             msg = reponse.message;
160         }
161         Neolink.ui.messageAjax({
162             classe: 'information',
163             message: msg //'Aucun résultat ne correspond à votre recherche'
164         });
165
166         if($('#duoliste').length)
167         {
168             return;
169         }
170     }
}

```

## Annexe 8 : Objet reponse affiché dans la console du navigateur qui contient nos coordonnées (latitude et longitude)



## Annexe 9 : Méthodes pour obtenir la carte et pour centrer sur un élément

```

80  /**
81   * Obtenir la carte avec l'identifiant fournit
82   *
83   * @param identifiant
84   * @returns {*}
85   */
86  obtenirCarte = function (identifiant) {
87    let objet = null;
88
89    if ($("#" + identifiant).length && typeof carte !== "undefined") {
90      objet = carte;
91    }
92    return objet;
93  },

```

```

1553 /**
1554 * Centrer sur un élément
1555 * @param carte
1556 * @param element
1557 * @param zoom
1558 */
1559 centrerSurElement = function (carte, element, zoom) {
1560   let limitesCartes = carte.map.getBounds(),
1561   nord = limitesCartes.getNorth(),
1562   sud = limitesCartes.getSouth(),
1563   amplitude = nord - sud,
1564   decalageLatitude = amplitude * 0.15,
1565   coordonnees = element.getLatLng();
1566
1567   // Zoom sur point
1568   if (zoom) {
1569     carte.map.fitBounds(L.latLngBounds([coordonnees]));
1570   }
1571
1572   // Centrage sur point avec décalage
1573   // Suppression du décalage pour la liste dynamique
1574   carte.map.panoTo({
1575     lat: coordonnees.lat, // + decalageLatitude,
1576     lng: coordonnees.lng
1577   });
1578 },

```

## Annexe 10 : Fonctionnalité attendue

```
746 // Zoom sur la position souhaitée
747 let latitude = reponse.recherche.isochrone.latitude;
748 let longitude = reponse.recherche.isochrone.longitude;
749
750 let marker = new L.Marker([latitude, longitude]);
751
752 let carte = Neolink.Solutions.carte.obtenirCarte('carte_espace_solution');
753
754 Neolink.Solutions.carte.centrerSurElement(carte, marker, false);
```

## Annexe 11 : Problème rencontré sur la plateforme Job17

```
Uncaught TypeError: reponse.recherche.isochrone is undefined
dashboard-c88af2d6.js:749:28
    traiterReponseRecherche https://job17.home/solutions/dashboard-c88af2d6.js:749
    j https://job17.home/cdn.js:3
    firewith https://job17.home/cdn.js:3
    x https://job17.home/cdn.js:5
    b https://job17.home/cdn.js:5
    send https://job17.home/cdn.js:5
    ajax https://job17.home/cdn.js:5
    soumettreRecherche https://job17.home/solutions/dashboard-c88af2d6.js:1152
    d https://job17.home/cdn.js:99
    validate https://job17.home/cdn.js:99
    dispatch https://job17.home/cdn.js:4
    handle https://job17.home/cdn.js:4
    trigger https://job17.home/cdn.js:4
    trigger https://job17.home/cdn.js:4
```

## Annexe 12 : Solution développée pour ne pas impacter les autres cartes

```
746 // Vérifier qu'on a bien nos coordonnées
747 if(reponse.recherche.isochrone !== undefined) {
748   if((reponse.recherche.isochrone.latitude !== undefined) && (reponse.recherche.isochrone.longitude !== undefined)) {
749     let latitude = reponse.recherche.isochrone.latitude;
750     let longitude = reponse.recherche.isochrone.longitude;
751
752     // Récupérer la map et définir un marker
753     let carte = Neolink.Solutions.carte.obtenirCarte('carte_espace_solution');
754     let marker = new L.Marker([latitude, longitude]);
755
756     // Centrer sur la position souhaitée
757     Neolink.Solutions.carte.centrerSurElement(carte, marker, false);
758   }
759 }
```