**Notes au futur développeur**

Il y a 3 listes : la liste des clients, la liste des interventions et la liste des machines. Il est possible d’ajouter, modifier ou supprimer un élément d’une liste.

Il y a deux utilisateurs : l’user et l’admin. L’utilisateur simple ne peut que ajouter un client, une intervention et une machine, mais il ne peut pas modifier ni supprimer quoi que ce soit d’une liste à l’inverse de l’admin.

Ce qui est à modifier est principalement le design du site : je n’ai eu que très peu de temps pour faire le projet et me suis concentrée sur l’aspect pratique plus que sur l’aspect visuel. Il y a aussi un problème avec la déconnexion : si un utilisateur est connecté, qu’il quitte le site sans se déconnecter, puis qu’il revient plus tard, il sera toujours marqué comme connecté mais il l’interface de connexion réapparait tout de même.

LE PROJET FONCTIONNE EN MVC !

**A propos de la page index.php**

c’est une page centrale : elle redirige à chaque fois vers la page concernée.

A la fin du fichier, on voit une ligne « include », elle fait en sorte de conduire à chaque fois vers la page sélectionnée

Elle est ainsi : include(‘ ./views/ ‘.$entite. ‘/ ‘.$action.’.php’ ) ;

$entite fait référence au dossier (ex : client, intervention, connexion, …), et $action faire référence au fichier php concernée (ex : list, modif, ajout, …).

On obtiendra alors une url de ce genre : <http://localhost/newpc/index.php?e=client&a=list.php>

On fait ainsi toujours appel à index.php qui identifie alors le e comme etant client et le a comme étant list.

Le fichier models est réservé aux fonctions php et au css, et chaque liste a sa page de fonctions (client, intervention, connexion, machine). C’est le M de MVC. Les pages avec lesquelles on interfère telles list par exemple sont le V de MVC (soit views) : ce sont elles que l’on voit.

Une grande partie du CSS du projet a été créer à partir de bootstrap.