

L3 MIAHS Informatique
Université Toulouse 2 – Jean
Jaurès



Thales Alenia Space
26 av Jean François Champollion,
31100 TOULOUSE



Développement d'un outil de gestion de production

Thalesmee

ROCHELLI Jonathan

Destinataires :

Luc FERRIE
Pierre-Jean CHARREL

REMERCIEMENTS

Je tiens à remercier tout d'abord, Mme. SEGALIE Pascale, chef du service des moyens d'essais d'environnement pour ses conseils et son enthousiasme me concernant.

M. FERRIE Luc chef du laboratoire des vibrations, pour son implication dans mes projets et son soutien.

Je ne peux tous les citer, mais je tiens à remercier tout particulièrement l'ensemble du service des moyens d'essais pour leur accueil chaleureux, leur bonne humeur, et les multiples conseils pertinents, que j'ai pu recevoir, ainsi que les expériences partagées. Grâce à eux j'ai pu faire le choix de mon avenir professionnel.

Je remercie également M. CHARREL Pierre-Jean, mon tuteur pédagogique, pour ses nombreux conseils sur le rapport d'alternance ainsi que sur la présentation.

Je remercie plus généralement le personnel de Thales Alenia Space avec qui j'ai été en relation, pour le temps qu'ils m'ont accordé ainsi que l'ensemble des personnes qui m'ont permis de réaliser mon alternance.

Sommaire :

Table des illustrations	3
Introduction	4
I/ Présentation de l'entreprise	5
1) Thalès Alenia Space	5
2) Organigramme	7
II/ Contexte de la mission	8
1) Les moyens d'essais	8
2) Thalesmee	9
3) Technologies utilisées	11
III/ Objectif de mon alternance	12
1) Correction de bugs	12
2) Correction et amélioration de certaines fonctionnalités	14
3) Développement de nouvelles fonctionnalités	16
IV/ Méthodes utilisées	31
1) Post-it	31
2) Scrum Board	31
V/ Bilan du travail effectué	33
VI/ Bilan de mon alternance	34
VII/ Annexes	35

Table des illustrations

Figure 1 : Planning de Thalesmee.....	10
Figure 2 : Interface de saisie d'un nouvel instrument..	14
Figure 3 : Interface de saisie d'un nouveau capteur	15
Figure 4 : Tableau récapitulatif des capteurs au format Excel	16
Figure 5 : Affichage d'un retard des moyens d'essais..	16
Figure 6 : Affichage d'une anomalie	17
Figure 7 : Saisie d'une anomalie	17
Figure 8 : Tableau récapitulatif des anomalies	18
Figure 9 : Première série de graphique.....	21
Figure 10 : Seconde série de graphique.....	21
Figure 11 : Graphique des retards de livraison avant le test.....	22
Figure 12 : Graphique de la durée d'attente sur l'étagère avant le test.....	22
Figure 13 : Courbe critère	23
Figure 14 : Saisie des informations générales d'une fiche critère.....	24
Figure 15 : OF concernés par un essai.....	25
Figure 16 : Interface d'ajout d'un capteur.....	26
Figure 17 : Essai comportant deux numéros de série ...	27
Figure 18 : Essai susceptible d'être réalisé en simultané	27
Figure 19 : Vue Hebdomadaire de planning	29
Figure 20 : Vue mensuelle du planning.....	29
Figure 21 : Affectation de la ligne de produit.....	30

Introduction

Dans le cadre de ma dernière année de licence MIASHS (Mathématiques et Informatique Appliquées aux Sciences Humaines et Sociales) à l'université Toulouse Jean-Jaurès (spécialité informatique), j'ai pu réaliser mon parcours en alternance. J'ai donc réalisé mes périodes en entreprise au sein de Thales Alenia Space à Toulouse.

Cette mission s'inscrit parfaitement dans mon parcours puisqu'il s'agissait de développement informatique d'une application interne à l'entreprise. J'ai donc occupé un poste de développeur en utilisant mes compétences acquises à l'université dans le domaine des bases de données et du développement web.

Pour répondre aux attentes de l'entreprise, j'ai dû répondre à des exigences techniques mais aussi développer une méthode de travail rigoureuse et organisée.

Afin de vous présenter mon travail, je vous parlerais donc ,dans un premier temps, de l'entreprise avec une présentation de Thales Alenia Space puis du contexte dans lequel j'ai intégré les équipes.

Ensuite je vous présenterais les différentes tâches qui m'ont été confiées, c'est à dire la correction de bug et l'amélioration de différentes fonctionnalités déjà développées à mon arrivée et le développement de nouvelles interfaces pour satisfaire les nouveaux besoins des utilisateurs.

Je vous présenterais également des différentes technologies que j'ai utilisées.

Ensuite je parlerais également de la méthode agile dont je me suis servies afin d'organiser mon travail avec une méthode de « Post-it » et de Scrum Board.

Pour terminer je ferais un bilan de mon expérience d'un point de vue personnel mais également professionnel.

I/ Présentation de l'entreprise

1) Thalès Alenia Space

Thalès Alenia Space est une coentreprise franco-italienne du domaine de l'industrie spatiale. Elle a été créée en 2007 avec l'accord entre Thalès (France) et Leonardo (Italie) suite au rachat d'Alcatel Alenia Space par Thalès. Elle est implantée dans de nombreux pays tels que : la France, l'Italie, l'Espagne, la Belgique, l'Allemagne ou encore les Etats-Unis.

Thalès Alenia Space intervient dans la conception de solutions pour les télécommunications, la navigation et l'exploration. Elle intervient également dans le développement d'infrastructures orbitales et de systèmes satellites associés aux programmes spatiaux. Elle forme la Space Alliance avec Telespazio. Dans cette alliance, la co-entreprises proposent une offre unique allant du domaine des communications à la navigation en passant par la recherche scientifique et l'observation.

En France, elle est basée à Cannes (siège sociale) et à Toulouse. Le site de Toulouse est dédié au développement et aux tests des équipements. Le site de Cannes est quant à lui, dédié à l'intégration et l'assemblage des sous-systèmes.



Thalès Alenia Space a réalisé un chiffre d'affaires d'environ 2.5 milliards d'euros en 2018 et emploie approximativement 8000 personnes dans 99 pays.

L'organisation de Thalès Alenia Space est basée sur un modèle de multinationales. L'entreprise est gérée par un Président Directeur Général et un Directeur General Adjoint. Pour appliquer cette architecture d'entreprise internationale, deux axes sont utilisés :

- Business Lines
- Meta Competence Center

Les Business Lines sont le côté marketing de l'entreprise. Ils doivent entretenir des relations avec les clients. Plus précisément, ils sont chargés de comprendre le marché et les attentes des clients, créer et maintenir des relations durables avec le client. Ils sont aussi chargés de définir et gérer la stratégie de marketing et de trouver des solutions pour aborder le marché mondial du point de vue commercial. Les Business Lines sont découpés en deux secteurs d'activité :

- BL TL : Secteur Telecom. Ce secteur est chargé des systèmes de télécommunications à bord et au sol.
- BL OEN : Secteur Observation, Exploration et Navigation. Ce secteur est chargé des systèmes d'observation optique et radar, de l'altimétrie, de la météorologie, de l'exploration et de la navigation.

Un Meta Competence Center est en charge de la coordination des investissements industriels et de la coordination des politiques technologiques et industrielles. Il crée aussi des centres de compétences partout où des mises en commun peuvent être réalisées.

Des centres de compétences sont instanciés dans quatre pays : la France, l'Italie, le Royaume-Uni et la Belgique. En France, il existe trois centres de compétences. :

- Electronique
- Solutions logicielles
- Plateformes et Intégration

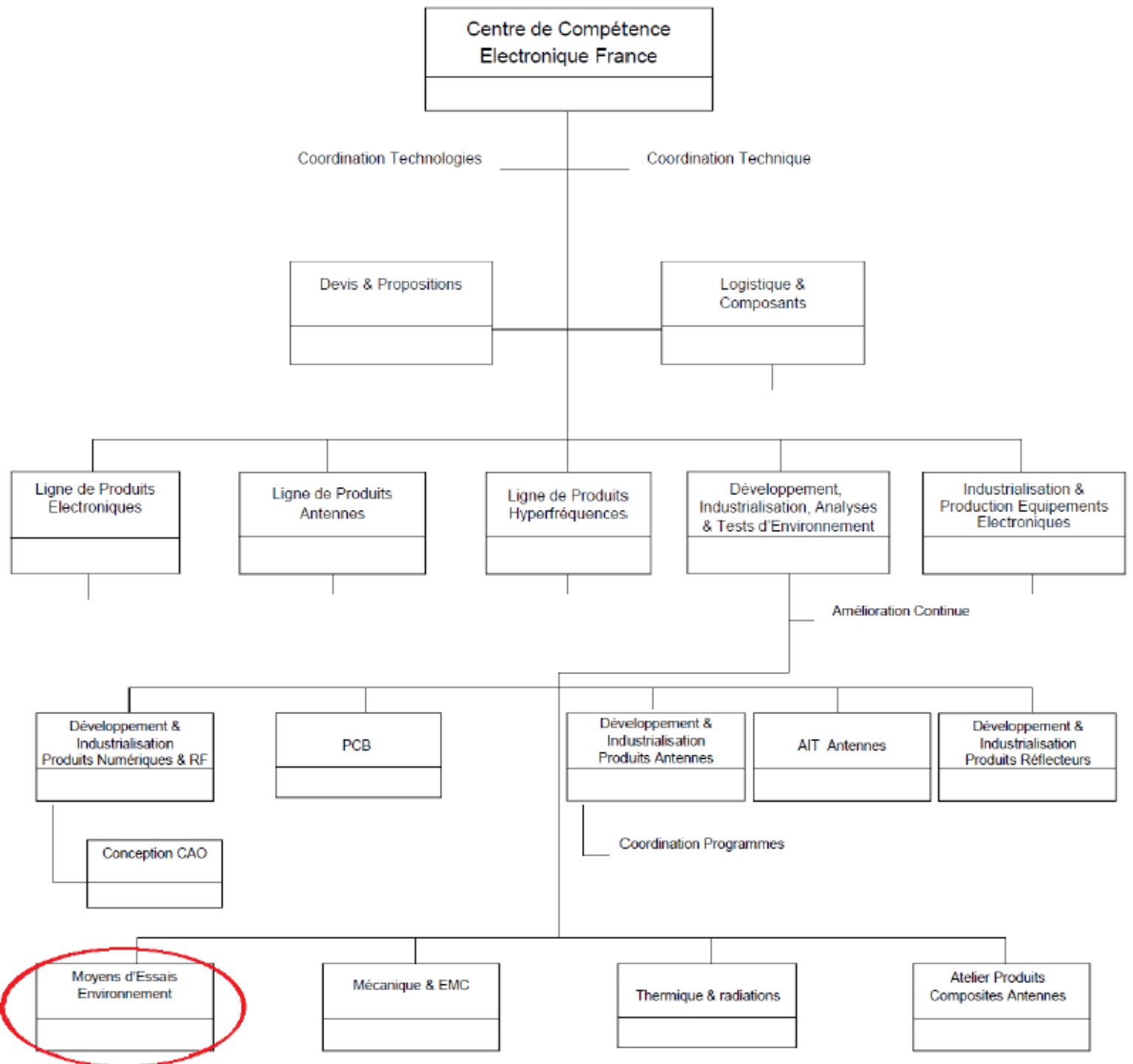
J'ai intégré le centre de compétence Electronique (CCEL). Ce centre de compétence est réparti sur plusieurs pays tels que : la France, la Suisse, l'Italie et la Belgique. En France le CCEL est basé à Toulouse.

Il est divisé en plusieurs catégories :

- LPE : Ligne de produit chargé des produits numériques
- LPA : Ligne de produit chargé des antennes
- LPH : Ligne de produit chargé des produits hyperfréquences
- IPE : Chargé de la production
- DIT : Chargé du développement, des analyses et des tests environnement

Durant mon alternance j'ai intégré le service des moyens d'essais environnement. Ce service fait partie de la branche DIT du CCEL France.

2) Organigramme



II/ Contexte de la mission

1) Les moyens d'essais

Le service des moyens d'essais est en charge des tests qui assurent que les équipements qui composeront les satellites résisteront aux contraintes qu'il subiront tout au long des phase de vie des satellites. En effet, ce service a pour but de vérifier que les équipements résisteront aux contraintes de déploiement des panneaux solaires, de décollage, de l'environnement spatial, etc.

Le service procède à différents tests sur les équipements. Pour cela, il est décomposé en 3 laboratoires :

- ➔ EMC (compatibilité électromagnétique) : Laboratoire chargé des tests sur les perturbations électromagnétiques émises et perçues par l'équipement. Ce laboratoire vérifie que l'équipement n'est pas perturbé par son environnement et ne perturbera pas non plus les autres qui composent le satellite.
- ➔ VIB (essai vibration) : Laboratoire chargé des tests de vibrations engendrés lors de la mise en orbite et du déploiement des panneaux solaire et des réflecteurs du satellite. Ce laboratoire simule donc des vibration et des chocs.
- ➔ VTH (essai vide thermique) : Laboratoire chargé des tests de températures extrêmes et de pression faible. Ce laboratoire vérifie que les écarts de température ($\pm 200^{\circ}\text{C}$) et qu'une faible pression n'endommageront pas les équipements.

Ces trois laboratoires sont composés d'une vingtaine de personnes qui réalisent les essais. Les laboratoires suivent la même organisation d'un point de vue conceptuel, avec des étapes similaires au niveau des essais :



Pour gérer cela le service des moyens d'essais dispose d'un outil de gestion de production : Thalesmee. Cet outil permet de suivre l'avancement de chaque essais, de gérer le parc d'instruments, de lister les instruments associées à chaque tests, etc.

2) Thalesmee

Depuis plusieurs années Thalès Alenia Space se place dans une démarche *LEAN*¹. Cette démarche est déployée dans l'ensemble de l'organisation. Le service des moyens d'essais s'y inscrit, notamment par le développement de logiciel de gestion de production et de suivi. Pour suivre l'avancement de leurs essais les laboratoires utilisent l'outil société Primavera. Primavera renseigne tous les essais que doivent effectuer le laboratoire, à la manière d'un planning. Mais cet outil ne convenait plus aux besoins de Thales Alenia Space. En effet, certaines informations indispensables à la bonne gestion et au bon fonctionnement des laboratoires n'étaient pas présente dans cet outil. Thales a donc du développer ,en interne, un meilleur outil répondant pleinement aux besoins. Après avoir été validé en « *proof of concept* », l'outil Thalesmee a été mis en production.

Pour répondre pleinement aux besoins des services, Thalesmee comporte différentes fonctionnalités telles que :

- Un planning qui permet à l'ensemble des équipes travaillant dans chaque laboratoires de se coordonner pour l'utilisation du matériel de test. Ainsi, cela permet d'améliorer l'organisation globale des différents services mais également de garder une trace des jours d'essais et de pouvoir également communiquer entre les différentes équipes. On peut dire que ce modèle s'inscrit parfaitement dans la démarche LEAN puisqu'il va permettre une meilleure maîtrise des délais.
- Une partie « Métrologie » qui contient une base de données regroupant l'ensemble des capteurs et instruments disponibles. Ce qui s'inscrit, une fois de plus, dans le démarche LEAN puisque l'on pourra gérer les stocks plus facilement. De plus, cette base contient également des informations sur des capteurs et des instruments tel que : les sensibilités, la localisation, etc. Ces informations participent à la gestion du parc des instruments.

Ces deux exemples sont les deux principales fonctionnalités que contient Thalesmee. Nous verrons ces deux parties plus en détail par la suite.

¹ La démarche LEAN est un système d'organisation industrielle qui vient du groupe Toyota dont l'objectif est d'améliorer au mieux les performances des processus à l'aide de :

- La diminution des stocks
- La maîtrise des délais
- La réduction des défauts
- La réduction des coûts

Les laboratoires avaient besoin d'un outil permettant le suivi de l'avancement des essais. De plus, ces essais devaient être partagés avec leurs clients de manière à ce qu'ils puissent eux aussi suivre l'avancement de leurs équipements. Cet outil doit donc être très visuel et ergonomique. Pour cela, le planning des essais dispose d'un système de *drag and drop* et de redéfinition de la taille des barres permettant de manipuler les essais. Chaque essai peut être modifié (date, informations générales, etc.). De plus, l'essai peut évoluer dans cet outil suivant son état (Réservé, En cours, Terminé, etc.), cela permet donc de suivre chaque essais en connaissant la date de chaque état. Les barres disposent aussi de pastilles de différentes couleurs. Ces pastilles apportent des informations supplémentaires sur les essais tels que : les retards enregistrés, les anomalies, etc. Ces indications permettent aux laboratoires de disposer d'informations complémentaires et précises au niveau du suivi du projet.

Tous les essais sont ajoutés dans Thalesmee via un import Excel extrait d'un autre logiciel permettant ainsi de remplir la base de données. Cet export est réalisé à partir des informations présentes dans Primavera. Une mise à jour régulière est donc effectuée par les chef de laboratoires pour mettre à jour le planning.

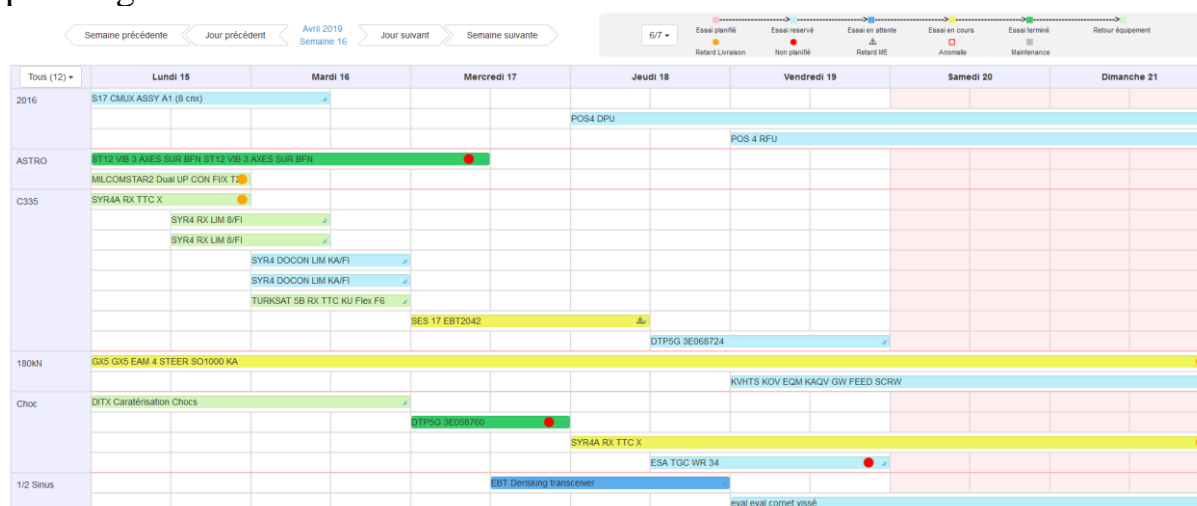


Figure 1 : Planning de Thalesmee

La partie Métrologie de Thalesmee est mise à jour via un import automatique de la base de donnée d'Adveso. Tous les nouveaux instruments sont alors importés dans la base de données de Thalesmee. Cet import permet de renseigner les informations générales des capteurs (numéro de série, nom, etc.). Cependant Thalesmee dispose d'informations complémentaires sur ces instruments. Ainsi, les responsables doivent renseigner les informations telles que : les sensibilités, la localisation, etc. directement sur le site web. La partie Métrologie permet aussi aux utilisateurs d'enregistrer directement un nouvel instrument ou un nouveau capteur via une interface. Le responsable devra donc saisir toutes les informations générales ainsi que la sensibilité si l'instrument enregistré est un capteur.

3) Technologies utilisées

Le projet dont j'ai été chargé durant mon alternance est un projet qui a été réalisé pendant plusieurs années, par plusieurs développeurs avant moi. J'ai donc dû m'adapter aux choix des technologies qui ont été choisies auparavant. Tous les langages de programmations utilisés au cours de mon alternance sont des langages étudiés en cours que j'ai pu approfondir à l'aide de mes projets personnels.

- ❖ HTLM 5 (Hypertext Markup Language) : Langage de balise à la base du développement WEB.
- ❖ CSS : (Cascading Style Sheets) : Langage de programmation utilisé pour gérer le design du site ou de l'application web. Utilisation du Framework Bootstrap.
- ❖ JAVASCRIPT : Langage de script. J'ai utilisé ce langage de programmation, dans un premier temps, pour la validation de formulaire. En exécutant un script juste avant la soumission du formulaire il est possible de vérifier les informations saisies par l'utilisateur. Ainsi je peux éviter que l'utilisateur ne remplisse les champs avec des valeurs aberrantes. Ensuite, j'ai utilisé le JavaScript pour l'affichage dynamique de certains morceaux de la page. En effet, lorsque l'utilisateur clique sur un case à cocher il est possible d'afficher le formulaire correspondant à sa demande. De plus, ce langage m'a permis de mettre en place des requêtes AJAX (Asynchronous JavaScript and XML), qui m'a permis de réaliser des requêtes asynchrones. Cela permet d'exécuter un script PHP et ainsi récupérer les informations de la base de données. Pour terminer, j'ai utilisé le JavaScript pour ne rafraîchir qu'un ou plusieurs éléments de la page sans être contraint de réactualiser la page complète. Cela permet une meilleure réactivité de l'application. Utilisation du Framework JQuery.
- ❖ PHP (Hypertext Preprocessor) : Langage de programmation côté serveur (back end). J'ai utilisé ce langage pour générer du code dynamique suivant le type d'utilisateur, l'identifiant d'un essai, etc. Le PHP m'a été utile aussi pour réaliser des requêtes avec la base de données. Ces requêtes permettent de récupérer, modifier, supprimer des informations dans la base. J'ai aussi utilisé des bibliothèques PHP tel que JpGraph pour générer des graphiques en PHP.

La base de données de Thalesmee est une base de données MySQL. J'ai donc utilisé des requêtes SQL pour gérer la base de données.

La plupart des fonctionnalités de Thalesmee nécessaires aux laboratoires étaient déjà implémentées lors du début de mon alternance. Ma mission aura donc été de développer de nouvelles fonctionnalités ou de corriger des bugs existants.

III/ Objectif de mon alternance

1) Correction de bugs

Dans un premier temps, durant mon alternance il m'a été demandé de corriger des erreurs liées à des fonctionnalités déjà présentes dans l'outil. Ces dernières ne remplissaient pas les besoins attendus. Il m'a fallu pour cela analyser et comprendre le code écrit par mes prédécesseurs. Cette tâche m'a demandé de la concentration et de l'autonomie car le service n'étant pas un service d'informatique j'étais le seul développeur sur le projet. De plus il m'a fallu faire preuve de réactivité pour corriger les bugs le plus rapidement possible.

Quelques exemples ²:

- « *Bug des pastilles* »

Le planning des essais dispose de plusieurs indicateurs, comme la couleur des barres pour indiquer l'état dans lequel est l'essai, les contours en rouge pour indiquer une anomalie (dont nous parlerons plus tard) et les pastilles de couleurs. En effet, les pastilles rouges indiquent que l'essais en question n'est pas planifié et les pastilles orange indique un retard de livraison par le client de son équipement avant le test. Et une troisième qui est le symbole «attention» dont nous reparlerons aussi plus tard.

Le « *Bug des pastilles* » était visible sur les pastilles oranges. La pastille ne se positionnait pas de façon logique. En effet, dans certains cas, la pastille apparaissait alors que l'équipement n'était pas en retard et inversement. Pour cela, j'ai donc modifié la base de donnée de manière à rajouter un champ contenant les informations de la date de début prévue de l'essai ainsi que sa date de fin prévue. Ces informations me permettent donc de connaître la date à laquelle l'équipement devait être livré par le client. Ainsi en modifiant le code de génération du planning des essais (cf. Annexe 1.1) il est désormais possible de savoir précisément si l'équipement est en retard.

² Les exemples cités ci-dessous sont un aperçu des tâches qui m'ont été confiées durant mon alternance. Ces exemples ne sont que les parties les plus importantes de ce que j'ai du réaliser.

De plus, les utilisateurs devaient pouvoir ajouter ou retirer les pastilles manuellement. J'ai donc un fois de plus modifié la structure de la base de données (cf. Annexe 1.2) pour implémenter le choix de l'utilisateur. J'ai aussi insérer une case à cocher dans la page de modification des essais pour que l'utilisateur puisse réaliser la manipulation. Lors de l'arrivée sur la page de modification, si une pastille est appliquée à l'essai les cases correspondantes sont donc déjà cochées. De plus, les actions de l'utilisateur sont prioritaires sur l'affichage de la pastille. En effet, si l'essai est réellement en retard mais que l'utilisateur a décidé d'enlever la pastille alors la pastille n'apparaîtra plus (sauf ré-ajout par l'utilisateur).

- « *L'import Primavera* »

La grande majorité des essais présents dans Thalesmee sont importés depuis un autre logiciel « Primavera ». Les essais sont exportés sous format .xls. Le fichier généré peut ensuite être lu par l'outil et ainsi ajouter ou mettre à jour les essais. Tous les essais peuvent être modifiés via un import seulement si son état est inférieur ou égale à « réservé ». Si l'essai est dans un des états suivants, les dates de début et de fin ne peuvent être modifiées que par l'utilisateur via l'interface.

Mais il arrivait ,dans certains cas, que certains essais (représentés sous forme de barre dans la calendrier de Thalesmee) ne s'importaient pas. J'ai dû comprendre le code précédemment réalisé pour apporter les modifications nécessaires. Après analyse, j'ai remarqué plusieurs problèmes :

- Dans un premier temps, certaines barres disposaient d'identifiant déjà existant. En effet, dans la base de données chaque essais se voit attribuer un identifiant unique. De plus l'essai dispose aussi d'un identifiant unique depuis Primavera. Cependant, certains essais, lors de l'export, étaient associés à un identifiant Primavera déjà associé à un essai renseigné dans la base de donnée. L'essai étant dans un état qui ne permet pas la modification via import, l'essai ne s'importait pas. J'ai donc ajouté un message d'erreur avec toutes les informations liées à l'essai en question pour avertir l'utilisateur que l'import n'est pas conforme.
- Ensuite, l'un des problèmes qui est survenu est que certaines barres de maintenance désignées dans la légende comme étant de couleur grises apparaissaient d'une autre couleur. Ces barres étaient donc traitées comme des essais standards. Une convention de nommage des essais en maintenance a donc été mise en place avec le mot clé MAINT qui permet de reconnaître ces essais.

La correction de bug m'a permis de me familiariser avec le code présent dans Thalesmee. Il m'a donc été demandé par la suite de réaliser des corrections sur certaines fonctionnalités déjà présentes dans Thalesmee mais qui ne répondent plus tout à fait aux besoins des utilisateurs.

2) Correction et amélioration de certaines fonctionnalités

● « Insertion de nouveaux capteurs »

Ajouter un instrument

Informations générales

Numéro de l'instrument

Type/Modèle

Numéro de série

État

Remarque

Ancien immo

Fabricant

Dernière cal: JJ/MM/YYYY

Prochaine cal: JJ/MM/YYYY

Domaine

Désignation

Localisation

Fiche technique :

Parcourir... Aucun fichier sélectionné.

Photo de l'instrument :

Parcourir... Aucun fichier sélectionné.

Capteur

Valider

Annuler

Figure 2 : Interface de saisie d'un nouvel instrument

La partie de Thalesmee permettant la gestion du parc d'instruments permet d'enregistrer de nouveaux capteurs ou instruments à travers différentes méthodes.

La première méthode est l'importation via la base de données Advoso, Une extraction automatique et journalière, permet de mettre à jour les données que nous conservons en local. Ainsi si un instrument est modifié sur Advoso, les modifications sont reportées sur notre base de données. Bien sûr, les informations que l'outil ajoute en plus sur chaque instruments, comme la sensibilité d'un capteur, doivent être modifiées via l'outil de gestion des instruments.

La seconde est la saisie d'un instrument ou d'un capteur via l'interface de Thalesmee. Le laboratoire des moyens d'essais dispose d'une interface permettant de saisir et d'insérer de nouveaux capteurs dans la base de données (cf. Figure 2). L'interface propose de renseigner tous les champs nécessaires à la déclaration d'un nouveau capteur.

En cliquant sur « capteur » (Encadré rouge figure 2) l'utilisateur informe que l'instrument qu'il souhaite remplir est un capteur. Une interface pour la saisie des capteurs est alors affichée (voir Figure 3).

Figure 3 : Interface de saisie d'un nouveau capteur

Lorsque l'utilisateur sélectionne un type de capteur (encadré orange de la Figure 3) l'affichage est dynamique. En effet certains types de capteurs sont des capteurs Mono Axe (un seul axe X) d'autres sont Tri Axes (Axe X, Y, Z). Ces informations étant différentes, l'affichage doit être dynamique au moment du choix du type de capteur. Les nouveaux capteurs qui m'ont été demandés de renseigner sont des bobines. Les bobines sont des capteurs spéciaux possédant une sensibilité et une propriété qui peut être soit Sinus soit Random. J'ai donc dû réaliser une interface qui permet d'insérer ces nouveaux capteurs. J'ai donc modifié le code JavaScript exécuté lors du choix du type de capteur par l'utilisateur pour insérer une condition lorsque le type choisi est un type Bobine. Pour cela, j'ai ajouté une classe CSS à tous les éléments non nécessaires à la saisie d'une bobine pour gérer leurs affichages (cf. Annexe 2.1).

De plus, il m'a fallu modifier le code PHP (cf. Annexe 2.2) pour l'insertion de ces nouveaux capteurs dans la base de données en ajoutant une condition au traitement du formulaire gérant le cas où le type du capteur est de type Bobine. Thalesmee proposant un bouton pour exporter la liste des

instruments au format il a fallu que je modifie ce fichier de manière à afficher correctement les nouveaux instrument dans le classeur Excel (cf. Figure 4).

Type	Marque	Référence	Total	En Service	Calibration	Prêt Cannes	Prêt Intespace	Prêt Mecano	Rebut
Bobine_Tension	LING	Random	1	1	0	0	0	0	0
		Sinus	1	1	0	0	0	0	0
Bobine_Courant	LING	Random	1	1	0	0	0	0	0
		Sinus	1	1	0	0	0	0	0

Figure 4 : Tableau récapitulatif des capteurs au format Excel

A la suite de la phase de correction de bugs et d'amélioration de certaines fonctionnalités il m'a été demandé de développer de nouvelles fonctionnalités. Ces nouvelles fonctionnalités pouvant être variées je vous montrerais les demandes les plus importantes que j'ai dû traiter.

3) Développement de nouvelles fonctionnalités

● « Indicateur retard moyen d'essai » :

L'une des demandes qu'il m'a été soumise aura été l'affichage sur le planning de Thalesmee d'un nouvel indicateur : retard ME. Cet indicateur signifie que le laboratoire a réalisé l'essai en retard. En d'autres termes que le délais n'a pas été respecté. Ces indicateurs prennent la forme d'un symbole « Attention » permettant de reconnaître les essais en retard (cf. Figure 5).

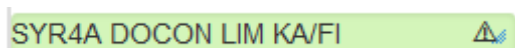


Figure 5 : Affichage d'un retard des moyens d'essais

Ces indicateurs sont générés automatiquement à la manière des pastilles oranges qui signifient un retard dans la livraison au client. En effet, l'affichage de retard ME se fait lorsque la date de fin réel de l'essai est supérieur à la date de fin prévu (cf. Annexe 3).

De plus, d'après une demande des utilisateurs et des chefs de laboratoires un essai possédant la pastille orange, donc étant en retard de livraison par le client, ne peut pas être considéré comme en retard ME. En effet, un essai en retard à cause du client ne peut pas être considéré comme étant une faute de la part du laboratoire.

L'indicateur de retard moyen d'essais étant considéré comme une pastille orange ou rouge il est possible aussi de l'ajouter ou de la supprimer de façon manuelle à l'aide d'une case à cocher via l'interface de modification de l'essai.

- « Gestion des anomalies »

Un autre besoin exprimé par les laboratoires est de gérer les anomalies qui pourraient se produire lors d'un essai et de les renseigner. Ces essais en anomalies sont identifiés sur le planning par un encadré rouge (cf. Figure 6).



Figure 6 : Affichage d'une anomalie

Les besoins étaient clairs. Dans un premier temps les anomalies devaient être saisies et renseignées via l'interface de modification de l'essai. J'ai donc choisi d'implémenter une case à cocher permettant de signaler une anomalie sur cet essai (cf. Figure 7). Lors du clic sur la case à cocher, un champ apparaît comportant toutes les informations à connaître pour la déclaration d'une anomalie.

Anomalie ☒

Traçabilité ok: ☐

Secteur d'origine: ☒ LPE ☐ LPA ☐ LPH ☐ I2PT ☐ I2PA ☐ Autre

☒ QUISS ☐ MPTI / NC

Heure(s) perdue(s): Heure(s) perdue(s) heure(s) Euro(s) perdu(s): KEuro(s) perdu(s) Keuro(s)

Figure 7 : Saisie d'une anomalie

La saisie d'une anomalie devait permettre de renseigner la ligne de produit de l'équipement, le type d'anomalie (QUISS ou MPTI/NC), le nombre d'heures perdues à cause de l'anomalie ainsi que le nombre de KEuros perdus. La saisie d'une anomalie devait comporter aussi un champ de saisie de texte pour permettre au technicien de saisir une phrase pour décrire l'anomalie. La case à cocher « traçabilité ok » permet d'indiquer que l'anomalie a été renseignée dans le logiciel de gestion des anomalies et ainsi que l'anomalie a bien été déclarée. Thalesmee ne pouvant pas communiquer avec ces outils, cette opération est obligatoire.

Pour garder en mémoire les anomalies il a fallu que je modifie la base de données. J'ai donc créé une nouvelle table « anomalie » permettant de stocker les anomalies. Cette table possède une clé primaire qui est l'identifiant de l'essai, le type d'anomalie, la ligne de produit, le descriptif, les heures perdues, les euros perdu ainsi que son statut (tracée ou non). Le schéma mis en place est disponible en Annexe 4.1 . Grâce à ce schéma il est facile de savoir si un essai

comporte une anomalie, il suffit de vérifier si l'identifiant de l'essai est présent dans la table anomalie.

A la suite de ce développement, il m'a été demandé de réaliser une interface permettant de visualiser les essais en anomalie selon une période donnée. Pour réaliser cette interface il a fallu que je récupère dans la base de données tous les essais en anomalie dans l'intervalle de date saisie par l'utilisateur.

Tableau récapitulatif des anomalies

Date de début : 05/04/2019

Date de fin : 05/06/2019

Valider

10

Rechercher :

Date de lancement	Affaire	Equipement	OF	Anomalie	Nom du technicien	Type d'anomalie	Status
27/05/2019	KVHTS	KOV EQM KA USER RFS FEED	PE5392	Plaque non Ok	CALANDRY ALEXIS	QUISS	Non validé
22/05/2019	AMU 7	Assy OMUX KU T1 (10 cx)	PE4722	procedure non figée	PECHEVIS MATHIEU	QUISS	Non validé
21/05/2019	SYR4A	COMSAT - BSB FM	PE1134	- Pas de poignées de transport équipement - Problème de bruit de mesure sur monitoring équipement. (1/2 journée)	PECHEVIS MATHIEU	QUISS	Non validé
16/05/2019	SENTINEL 3 D	RFU	PD9322	vis talon/rondelle interface bloquée -> arrachement de l'insert dans la rondelle interface	BAUZA GÉRARD	QUISS	Non validé
10/05/2019	RUAG	MCPV SADE	RUAG	Annulé pour problème de baie!!!	PELLEGRINI PIERRE	QUISS	Non validé
06/05/2019	RUAG	MCPV SADE	RUAG	problème de baie l'essai est annulé et reporté	PELLEGRINI PIERRE	QUISS	Non validé

Affichage de l'anomalie 1 à 6 sur 6 anomalie

Précédent

1

Suivant

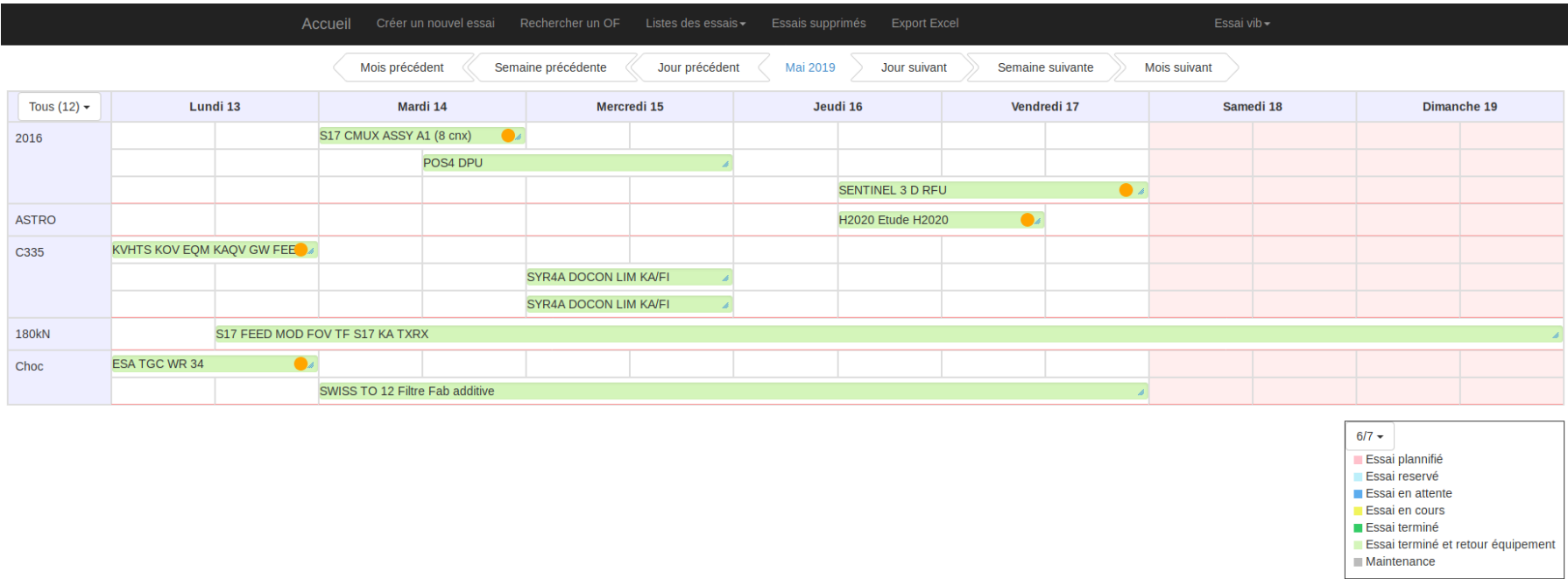
Figure 8 : Tableau récapitulatif des anomalies

Dans ce tableau figure toutes les informations demandées par les utilisateurs en relation avec l'anomalie. En effet, dans ce tableau figure la date de lancement de l'essai, l'affaire, l'équipement en question, le numéro de série concerné, le détail de l'anomalie, le nom du technicien qui a réalisé l'essais, le type d'anomalie et le statut. Le tableau est ordonnable selon n'importe quelle colonne. Il est aussi possible de réaliser une recherche. Cette recherche utilisant du JavaScript qui vise à masquer les lignes qui ne sont pas égales au texte saisi, la page n'est pas rechargée entièrement. Cette utilisation permet de ne pas surcharger le serveur et de rendre l'application plus rapide.

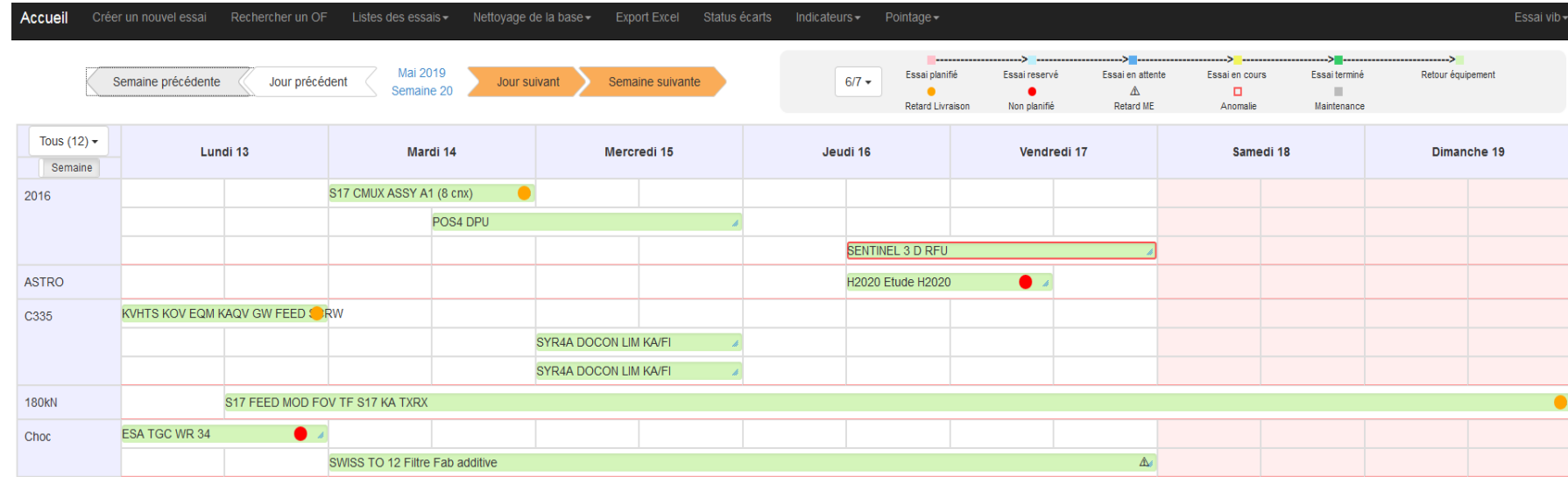
Lors de la saisie de la date par l'utilisateur et de la validation, le formulaire est envoyé en méthode GET (via url), les deux dates sont ensuite récupérées pour générer la page avec les dates souhaitées par l'utilisateur (cf. Annexe 4.2).

Voici un récapitulatif des fonctionnalités précédentes présentées sous la forme d’une comparaison entre l’interface du planning lors de mon arrivée et l’interface actuelle

Interface du planning lors de mon arrivée sur le projet



Interface du planning actuelle



● « Les indicateurs »

L'entreprise, qui se place dans une démarche *LEAN*, doit avoir des indicateurs. En effet, cette démarche impose de réaliser des graphiques pour suivre la performance du laboratoire ainsi que la qualité de réalisation des tests. De plus, faire des graphiques est très chronophage pour les responsables car les graphiques étant réalisés depuis des fichiers Excel, les données ne sont pas toujours disponibles. Cela entraîne, dans certains cas, des résultats qui peuvent être faux.

Une autre mission qu'il m'a été demandé est donc la génération d'indicateur pour les laboratoires.

Les graphiques sont extraits de la base de données. Cela permet d'avoir un indicateur fiable. De plus chaque indicateurs s'actualise en temps réel. En effet, les informations étant extraites de la base de données lorsqu'une modification est effectuée dans Thalesmee, les indicateurs sont automatiquement mis à jour.

Il m'a été demandé de travailler sur ces indicateurs pour, dans un premier temps, corriger les erreurs. En effet, certains graphiques n'étaient pas correct. Les informations affichées n'étaient pas les statistiques réelles du laboratoire. La requête SQL pour le récupération des données ainsi que le calcul pour la génération du graphique étaient faux.

De plus, les données étaient statiques. Les laboratoires devant suivre ses statistiques sur différents intervalles de date, les indicateurs doivent être dynamique en fonction des dates choisie par l'utilisateur. Ainsi, il est possible de choisir la période d'affichage des données. Cela permet au laboratoire de suivre ses statistiques sur toutes bornes temporelles et ainsi comparer entre deux périodes, les différences observées.

L'interface laisse également la possibilité d'exporter les graphiques sous format PDF ou Excel. Cela permet une impression rapide des indicateurs. Lors de l'affichage dynamique des graphiques, il m'a fallu aussi modifier le code de génération des fichiers PDF et Excel pour qu'ils affichent les données correspondantes à l'intervalle de dates.

Pour terminer, le service devait avoir des indicateurs pour le suivie des anomalie. Ainsi, j'ai dû développer 4 graphiques listant le nombre d'anomalies du laboratoire par ligne de produit tel que : le nombre et le nombre cumulé du nombre total d'anomalie en fonction des lignes de produit ainsi que le taux et le taux cumulé d'anomalie par lignes de produit en fonction du nombre total

d'essais. Ces derniers permettent de visualiser parmi un certain nombre d'essais combien sont en anomalie. De plus, ces données devaient être aussi dynamique.

Pour l'affichage de ces graphiques j'ai utilisé la bibliothèque PHP JpGraph qui permet l'affichage de graphiques sous forme d'une image en PHP. (cf. Annexe 5). Les graphiques précédents étant réalisés de cette manière et étant facile à prendre en main, j'ai choisi cette bibliothèque. De plus, ne connaissant pas cette bibliothèque, j'ai pu apprendre à m'en servir. Pour terminer, l'application comportant un nombre important de fichiers, il m'a paru approprié de réutiliser cette bibliothèque plutôt que d'en importer une autre. Cela permet de ne pas surcharger le serveur.

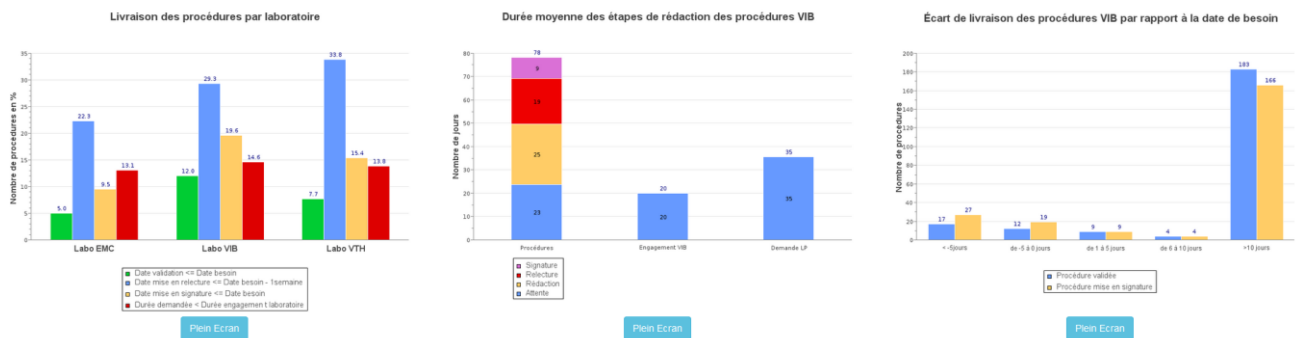


Figure 9 : Première série de graphique

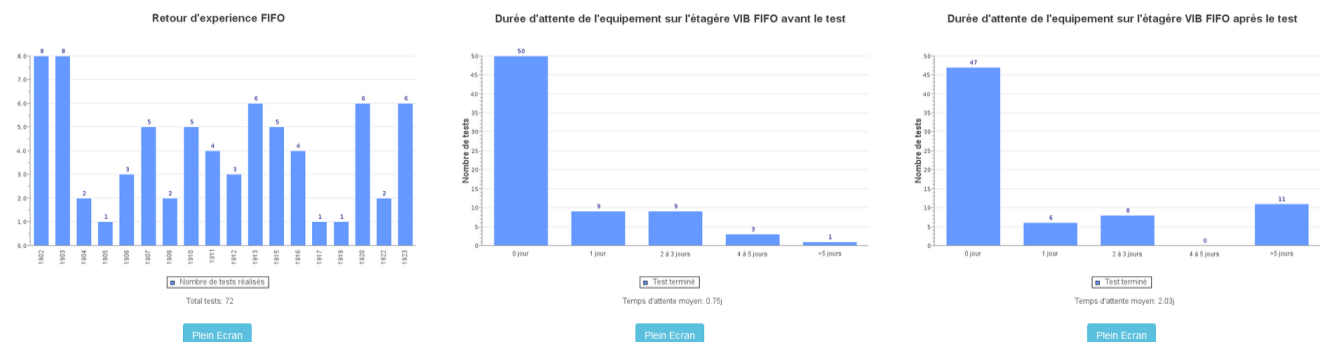


Figure 10 : Seconde série de graphique

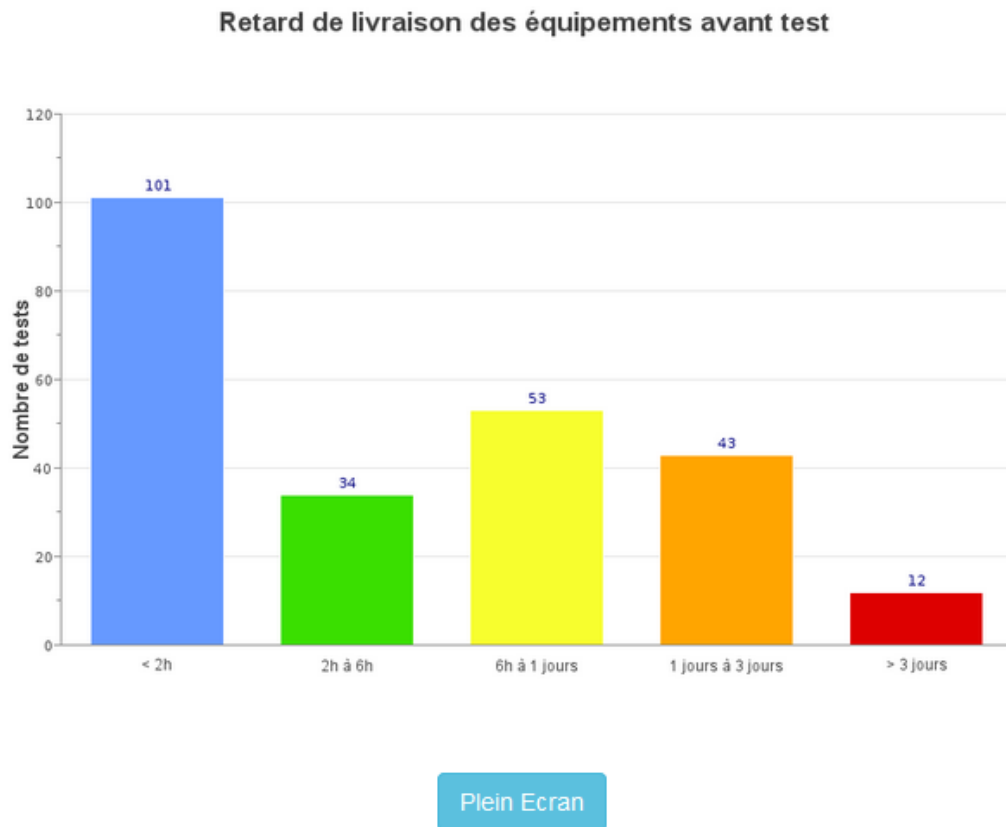


Figure 11 : Graphique des retards de livraison avant le test

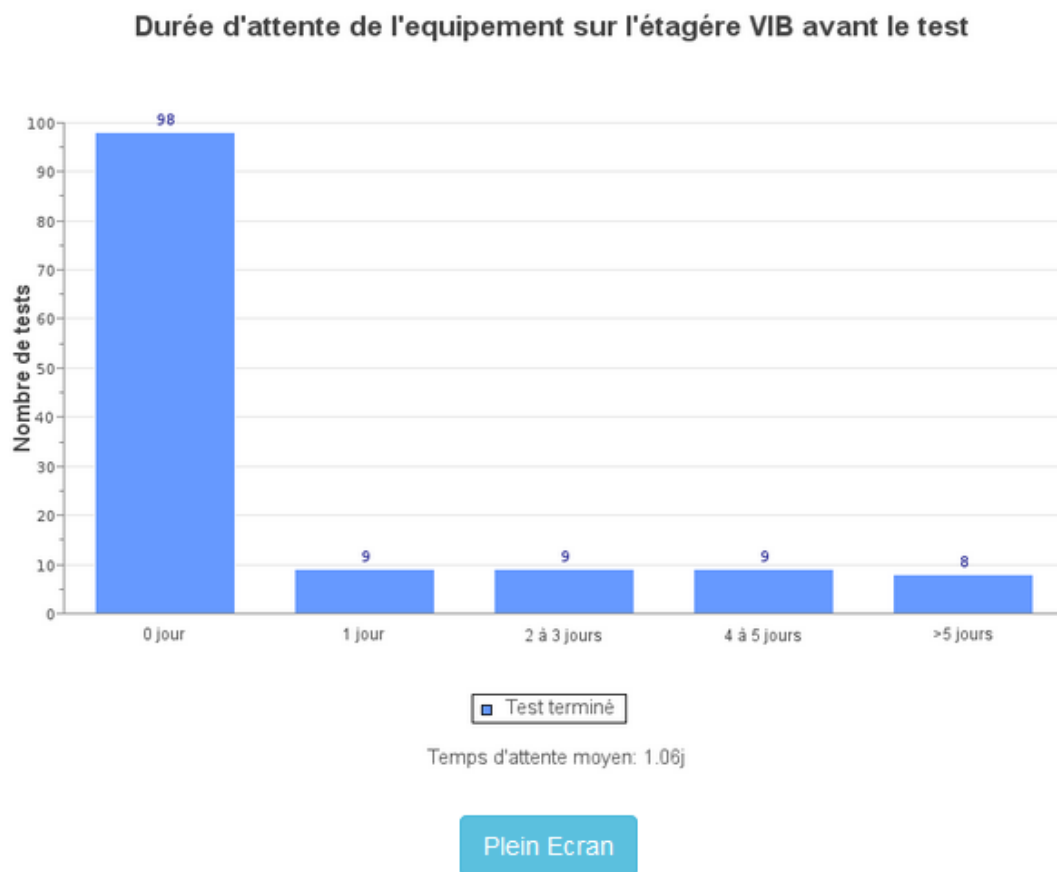


Figure 12 : Graphique de la durée d'attente sur l'étagère avant le test

- « *Fiche critère* »

1. Les besoins :

Le laboratoire des essais vibration réalise ses essais en trois phases. Pour commencer une signature est effectuée en début de test. La signature est une vibration de faible niveau qui est effectuée sur les équipements pour prendre des mesures de comparaison. Ensuite, une qualification est effectuée. Cette phase vise à appliquer des vibrations de niveaux similaire à ceux qui seront appliqués sur l'équipement en phase de lancement. La qualification est destinée à mesurer la capacité de résistance de l'équipement. Pour terminer l'équipement est soumis à nouveau à une vibration de faible niveau permettant de comparer avec les résultat précédent et s'assurer que l'équipement n'a pas été endommagé durant la phase de test. Cette phase est identique à la phase numéro une. A la fin de ces trois phases, des courbes sont générées. Les courbes sont ensuite superposées (cf. Figure 13).

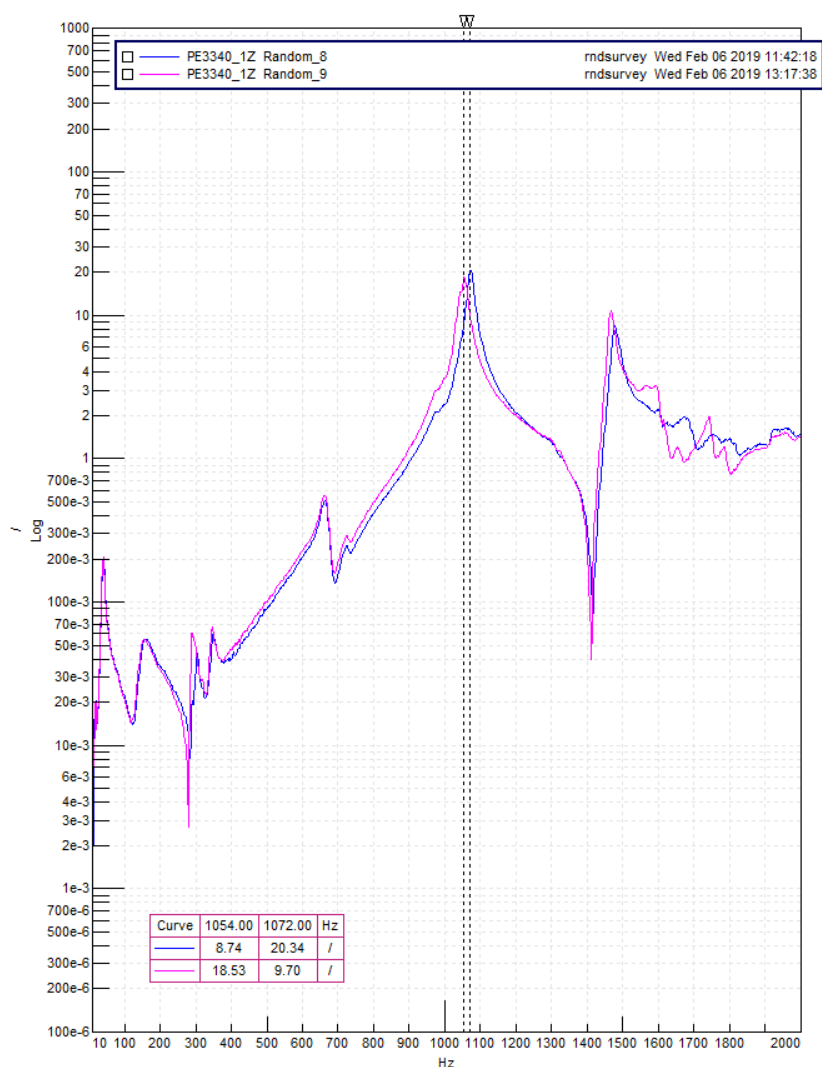


Figure 13 : Courbe critère

Si les deux courbes se superposent idéalement cela atteste que l'équipement n'a pas subi de dégradations sinon il a été endommagé lors de la phase de vibration de fort niveau. Ainsi l'un des besoins exprimé par le laboratoire a été de pouvoir rentrer toutes les informations de mesures des capteurs sur l'outil Thalesmee. Toutes ces données étant rentrées auparavant dans des fichiers Excel, le suivi sur le long terme était donc difficile. Il m'a donc été demandé de réaliser une interface permettant de saisir toutes ces informations. Toutes ces informations, stockées dans la base de données, permettront d'avoir un suivi sur le long terme de ces courbes. En effet, tous les équipements d'une même famille doivent posséder des signatures similaires. Il est possible d'avoir certaines différences entre la phase 1 et la phase 3 de vibration mais, en général, sur l'intégralité des équipements les courbes doivent être homogènes. Ainsi, grâce aux données saisies dans Thalesmee il sera possible de vérifier ces affirmations.

Premièrement l'interface devait permettre de saisir des informations générales telle que le critère pour la fréquence, le critère pour l'amplitude et devait permettre de renseigner si l'équipement a subi une pré-vibration ou non. J'ai donc réalisé un formulaire comportant trois champs (cf. Figure 14).



Critère pour la fréquence: 10

Critère pour l'amplitude: 40

Pré-vibration: ☐ Oui ☒ Non

Figure 14 : Saisie des informations générales d'une fiche critère

Deuxièmement, l'utilisateur devait pouvoir saisir les capteurs selon trois axes (X, Y, Z). De plus, le nombre de capteurs variant selon le type d'essai, le nombre était donc inconnu. Le technicien doit avoir la possibilité de ne renseigner aucun capteur ou d'en renseigner un certain nombre, inconnu à l'avance. S'ajoute à cela le fait que ces capteurs peuvent être différents au sein d'un même essai selon le numéro de série demandé. En effet, lors d'un essai, il est possible de tester plusieurs équipements en même temps (cf. Figure 15).

Ainsi l'interface devait permettre de saisir un nombre inconnu de capteurs sur un nombre inconnu d'équipements.

Figure 15 : OF concernés par un essai

Pour terminer, certains essais sont réalisés en même temps qu'un autre lorsqu'ils sont similaires. Cela signifie que les deux équipements sont testés en même temps mais ne font pas partie du même essai. En effet, les deux équipements n'utilisent pas exactement les mêmes capteurs donc les essais ne sont pas similaires mais sont pourtant réalisés en simultanés. Ainsi pour des questions d'ergonomie et de simplicité d'utilisation, la fiche remplie pour un essai devait être automatiquement remplie pour l'autre essai. Cela permet donc de voir les essais qui ont été réalisés en même temps mais aussi d'éviter à l'utilisateur de remplir deux fois des informations similaires.

L'interface devait donc :

- ✓ Permettre de saisir un nombre inconnu de capteur portant sur un ou plusieurs axes (X, Y, Z)
- ✓ Renseigner un ou plusieurs numéros de série
- ✓ Laisser la possibilité de saisir plusieurs essais à la fois.

2. Travail effectué :

La base de données :

Pour garder en mémoire toutes les informations sur les fiches critères il m'a fallu modifier la base de données afin de l'adapter aux besoins. Pour cela j'ai réalisé un travail de conception (cf. Annexe 6). A la suite de ce travail j'ai créé quatre tables supplémentaires dans la base de donnée :

- ➔ Table « critere » : Cette table permet de renseigner l'identifiant de l'essai, ses numéros de série associés, s'il y a eu une pré-vibration, le critère de la fréquence ainsi que le critère d'amplitude ; Ce qui équivaut aux informations générales saisissables via l'interface de la Figure 14.
- ➔ Table « fiche_critere » : Cette table dispose seulement d'un identifiant permettant de connaître les essais qui sont en lien (réalisés en même temps).
- ➔ Table « capteur_critere » : Cette table dispose d'un numéro de capteur, et de toutes les informations à propos des fréquences et des amplitudes

(initiale, finale et le décalage entre les deux mesures). Elle dispose aussi d'une colonne nom permettant de connaître le nom du capteur, d'une colonne qui indique si le capteur est spécifique ou non et une autre qui renseigne l'axe du capteur.

➔ Table « mesurer » : Cette table permet de lister l'ensemble des capteurs utilisés pour chaque essais et pour chaque numéro de série. Cette table possède le numéro de l'essai, le numéro d'OF et le numéro du capteur.

L'interface :

J'ai donc dû développer une interface ergonomique permettant de renseigner tous les capteurs pour chaque équipements.

Premièrement, l'interface dispose de trois cases à cocher. Une pour chaque axes (Axe X, Axe Y, Axe Z). Cela permet à l'utilisateur d'indiquer les axes qui disposent de capteurs. Lorsque l'on clique sur une des cases à cocher, il apparaît un tableau permettant la saisie de différentes informations du capteur (nom, fréquence, amplitude, etc.). Un bouton permet d'ajouter le nombre de capteurs souhaités pour chaque axe. (cf. Figure 16)

OF concerné : PE4927

Axe X
☒

Spécifique	Nom du capteur	Fréquence initiale	Fréquence finale	Décalage sur la fréquence	Amplitude initiale	Amplitude finale	Décalage sur l'amplitude	Supprimer
<input type="checkbox"/>	<input type="text"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="button" value="✖"/>

Ajouter un capteur

Axe Y

Axe Z

Figure 16 : Interface d'ajout d'un capteur

Deuxièmement, l'interface possédant les trois cases à cocher est multipliée par le nombre de numéros de série. En effet, cette interface sera affichée pour chaque OF de l'essai. Ainsi pour chaque équipement l'utilisateur pourra renseigner les capteurs utilisés. Ils pourront donc être différents entre chaque équipements. (cf. Figure 17)

OF concerné : PE1818

Axe X ☐
Axe Y ☐
Axe Z ☐

OF concerné : PE1819

Axe X ☐
Axe Y ☐
Axe Z ☐

Figure 17 : Essai comportant deux numéros de série

Ensuite, cette page dispose d'une partie proposant une liste de cases à cocher. Cette partie indique les essais qui sont susceptibles d'être passés en simultané. Cette liste contient les essais qui sont actuellement en cours. L'utilisateur peut donc cocher l'essai qu'il souhaite pour le renseigner en même temps et indiquer que ces deux essais ont été réalisés en simultané. Lorsqu'une case est cochée, à la suite de la liste des numéros de série (SN), apparaît le ou les SN concernés par l'essai coché. Il est donc possible de renseigner les capteurs associés. (cf. Figure 18)

Essai susceptible d'être réalisé en simultané

	Numéro	Equipement	Moyen	Affaire	OF
<input type="checkbox"/>	13028	DOCON LIM KA/FI	C335	SYR4B	PD9744
<input type="checkbox"/>	15695	DOCON LIM KA/FI	C335	SYR4B	PE2977

Figure 18 : Essai susceptible d'être réalisé en simultané

« Le planning mensuel »

Thalesmee, après avoir été validé en « *proof of concept* » pour le laboratoire VIB a été mise en service. A la suite des bonnes performances de l'application ainsi que de la preuve de son utilité, il a été décidé de l'utiliser dans les trois laboratoires. Ainsi, à la suite de réunion avec les différents chefs de laboratoires pour l'intégration du site web à chacun, il s'est avéré que certaines fonctionnalités ou affichage ne correspondaient pas aux besoins de tous les laboratoires. En effet, les laboratoires disposant tout de même d'un fonctionnement similaire au niveau des essais, il se peut que certaines différences existent. Thalesmee doit pouvoir remplir les besoins de tous les laboratoires. Il m'a donc été demandé de faire quelques modifications. Par exemple, le laboratoire des essais vide thermique réalise des essais sur plusieurs semaines. Il m'a donc été demandé pour une meilleure lisibilité de réaliser un vue mensuelle du planning de Thalesmee tout en gardant les autres fonctionnalités.

- ✓ Les cases du planning ne représenteront donc plus une demi-journée mais une journée entière.
- ✓ L'interface doit disposer d'une case à cocher pour passer entre les deux affichages.
- ✓ La semaine où se situe l'utilisateur lors de la vue hebdomadaire doit être la première semaine de la vue mensuelle.

Thalesmee disposant déjà d'un code de génération d'un planning j'ai donc apporté quelques modifications à ce code de manière à le réutiliser et non le dupliquer. Pour commencer j'ai dû modifier le code PHP permettant la génération de la page. L'affichage hebdomadaire disposant de deux boutons de navigation destinés à naviguer entre les semaines et les jours, l'affichage mensuel devait disposer de boutons permettant de naviguer entre les mois et les semaines. (cf. Annexe 7.1) De plus, le planning hebdomadaire disposant de 8 colonnes (une pour les entêtes et 7 pour les jours), la version mensuelle devait en disposer de 32 (une pour les entêtes et 31 pour les jours). (cf. Annexe 7.2)

Ensuite, j'ai dû modifier le code PHP permettant de sélectionner les différents essais et les informations correspondantes à la date du planning. Ce code est exécuté via une requête AJAX dans le code JavaScript permettant l'affichage du planning. Auparavant, les essais récupérés n'étaient que sur une semaine, maintenant il est possible de récupérer les essais sur le mois. (cf. Annexe 7.3) Pour terminer, j'ai modifié le code JavaScript pour afficher le planning sur 31 jours. Le code modifié est le code d'affichage des barres de couleurs du planning. En effet, l'affichage au mois nécessite d'insérer des cases vides avant et après le test en nombre plus important que pour la version en semaine. (cf. Annexe 7.4)

Tous (12) ▾	Jeudi 11		Vendredi 12		Samedi 13		Dimanche 14		Lundi 15		Mardi 16		Mercredi 17		
Semaine															
2016	POS4 DPU														
C335									SYR4B COMSAT - BSB FM2						
180kN	EXOMARS PARACHUTE														
Choc	TAS UK ETHM														
Banc traction									S17 FEED MOD LF W S17 KA TXRX						

Figure 19 : Vue Hebdomadaire de planning

Tous (12) ▾		J 11	V 12	S 13	D 14	L 15	M 16	M 17	J 18	V 19	S 20	D 21	L 22	M 23	M 24	J 25	V 26	S 27	D 28	L 29	M 30	M 31	J 1	V 2	S 3	D 4	L 5	M 6	M 7	J 8	V 9	S 10
Mois																																
2016	POS4 DPU																															
									STEEL SWOT SSR																							
													AMU 3 OPF																			
															STEEL calculateur OBC																	
															STEEL MDM 3G																	
ASTRO																																
													S17 DOCON ASSY KA/C A3 24 RF																			
																					S17 DOCON ASSY KA/C A2 24 RF											
																					S17 UPCON ASSY C/KA A1 24 RF											
C335																																
180kN																																
	EXOMARS PARACHUTE																															
Choc	TAS UK ETHM																															
Banc traction																																

Figure 20 : Vue mensuelle du planning

● « Les lignes de produits »

La génération de certains indicateurs nécessite des informations sur les lignes de produits d'équipement. En effet, certains indicateurs indiquant le taux d'essais en anomalie par rapport à la ligne de produit, la ligne de produit doit donc être renseignée pour chaque essais. Nous nous sommes donc aperçus que l'information sur les lignes de produits était manquante. Cependant, les lignes de produits ne sont pas renseignées lors de l'import des essais depuis le fichier Excel de Primavera. Ainsi, nous avons décidé de créer un champ dans l'interface de modification de l'essai pour saisir l'information manquante. Cependant, nous avons vite remarqué que l'utilisateur doit accéder à l'interface de modifications de chaque essais pour modifier ou renseigner la ligne de produit. Cette façon n'étant pas pratique ni rapide pour renseigner un grand nombre d'essai il m'a fallu réaliser une interface pour saisir les lignes de produit de façon optimisé.

Pour cela j'ai choisi de réaliser une interface sous forme de tableau qu'il est possible de trier. Ce tableau possède une série de boutons radios qu'il est possible de cocher pour renseigner la ligne de produit. Au chargement de la page la date par défaut est la semaine actuelle. Il est possible de naviguer entre les semaines de la même manière que pour le planning. En effet, l'interface possède trois boutons (semaine précédente, semaine actuelle, semaine suivante) permettant de naviguer entre les différentes semaines.

Certains essais pouvant déjà être renseignés il faut, de plus, que lorsque la ligne de produit d'un essai est renseignée qu'elle apparaisse dans l'interface. Ainsi si une ligne de produit a été saisi la case sera cochée au chargement de la page indiquant à l'utilisateur la valeur actuelle de la ligne de produit.

Ligne de produit - Semaine n° 24

Semaine précédente

Semaine actuelle

Semaine suivante

50

Rechercher :

Número	Date de lancement	Date de fin	Affaire	Equipement	OF	Ligne de produit					
13312	16/06/2019	19/06/2019	RX NEO ARTES	RX SI KU NEO	EM EM RX NEO	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
13311	14/06/2019	18/06/2019	RX NEO ARTES	RX SI KU NEO	EM EM RX NEO	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
12848	13/06/2019	13/06/2019	SYR4B	DOCON LIM KA/FI	FM PE2978	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
15705	13/06/2019	13/06/2019	SYR4B	DOCON LIM KA/FI	FM PE2970	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
12749	13/06/2019	13/06/2019	SYR4A	RX TTC X	PFM PE0735	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
13474	12/06/2019	18/06/2019	SYR4A	DTP 3G - COMSAT	PFM PE0009	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
15852	12/06/2019	12/06/2019	SYR4	CVSHF	FM PE0284	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
12995	12/06/2019	20/06/2019	KVHTS	RFLA 3500 EQM KOV	EQM PE4521	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
13318	11/06/2019	11/06/2019	MILCOMSTAR2	Dual UP CON FIX	FM PE5143	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
15703	11/06/2019	11/06/2019	MILCOMSTAR2	Dual UP CON FIX	FM PE1597	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre
10835	10/06/2019	13/06/2019	SYR4A	CMT - FM	PFM PE1197	<input type="radio"/> LPE	<input type="radio"/> LPA	<input type="radio"/> LPH	<input type="radio"/> I2PT	<input type="radio"/> I2PA	<input type="radio"/> Autre

Figure 21 : Affectation de la ligne de produit

IV/ Méthodes utilisées

Pour remplir correctement et rapidement les demandes des utilisateurs il a fallu que j'organise mon travail et que je fasse preuve de rigueur. Pour gérer mon développement j'ai utilisé deux méthodes étroitement liées :

1) Post-it

Durant mon alternance, tous les besoins utilisateurs m'ont été transmis sur de



simple Post-it. Cette méthode est simple mais terriblement efficace. Sur ces Post-it les besoins étaient clairs et précis. Ces bouts de papiers finissaient donc sur mon bureau. J'attribuais ensuite un ordre de priorité à chaque besoins utilisateurs de manière à mettre en place un hiérarchie des besoins. Par exemple, les bugs de certaines fonctionnalités

détenaient la priorité la plus élevé. Une fois la priorité déterminée je plaçais ces Post-it dans un Scrum Board.

2) Scrum Board

Le Scrum Board est un outil de la méthode Agile permettant de lister l'état d'une « user stories » (Besoin utilisateur). En effet, ce tableau dispose de plusieurs colonnes. Mon tableau avaient trois colonnes (A faire, En cours, Terminé). Cette catégorisation permet de suivre l'avancement de chaque tâches. En effet, il est possible de voir rapidement dans quel état se situe un besoin. De plus, cette méthode permet une bonne communication. Lors de la demande d'un utilisateur sur l'avancement de sa requête il est rapide de lui répondre et de justifier son état grâce à l'ordre de priorité.

Scrum Task Board Template				
Company name				
Stories	To Do	In Progress	Testing	Done
<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text. Replace it with your own text.</div> <div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text. Replace it with your own text.</div> <div>This is a sample text. Replace it with your own text.</div>
<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text. Replace it with your own text.</div> <div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text. Replace it with your own text.</div>
<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text. Replace it with your own text.</div>

Ces deux outils s'inscrivent dans la méthode Agile. J'ai utilisé cette méthode tout au long de mon alternance.

Dans un premier temps les utilisateurs me prévenaient qu'ils avaient besoin d'une nouvelle fonctionnalité. Je communiquais donc avec eux à propos de ce besoin en prenant des notes et en demandant des précisions si besoin. L'utilisateur ou moi écrivions ce besoin sur le Post-it. A la fin de cette phase, je remplissais le Post-it avec la priorité correspondante. Je plaçais ensuite ce Post-it dans le Scrum Board à l'intérieur de la case « A faire ». Si aucune priorité n'était plus importante je pouvais démarrer le développement. Je plaçais donc le besoin dans la case « En cours ».

Pour commencer j'écrivais les cas d'utilisation du besoin. Je mettais ensuite en place une liste de tests que le programme devait passer. A la suite de cela, je réfléchissais à la méthode que je pouvais appliquer pour répondre à la problématique. S'il fallait que je modifie la base de donnée je passais par une phase de conception afin de l'architecturer. A la fin de cette phase de réflexion, j'écrivais le code correspondant à la fonctionnalité.

Une fois le développement terminé je testais. Si les tests ne passaient pas, je repassais au développement et cela jusqu'à faire passer les tests. Si les tests étaient concluants je mettais en production la nouvelle fonctionnalité. Je déplaçais donc le Post-it dans la colonne « terminé ». Lors de la mise en production je notais la liste des fichiers que j'ai dû modifier pour cette « user stories ». Ainsi en cas d'erreur il m'était facile de retrouver cette liste pour apporter les modifications nécessaires. De plus, lors de chaque mise en production je réalisais une sauvegarde permettant de prévenir tout risque de perte de fichiers ou de données.

Ensuite je retournais voir l'utilisateur pour obtenir des retours sur mon développement. Cette phase est très importante pour un développement rapide et correct. L'utilisateur pouvait ainsi apporter certaines modifications à l'interface.

Si la fonctionnalité ne correspondait pas au besoin j'entamais une nouvelle itération en déplaçant à nouveau le post-it dans la colonne « En cours ».

Si la fonctionnalité correspondait au besoin et que l'utilisateur était d'accord j'enlevais le post-it du tableau et je renseignais le besoin dans un classeur Excel possédant plusieurs informations telles que : la demande, l'explication, la solution et la priorité. Ce tableau me permet d'avoir un suivi de tous les besoins utilisateurs (user stories) que j'ai développé au cours de mon alternance.

V/ Bilan du travail effectué

Durant mes périodes en entreprise j'ai dû répondre à différentes demandes, qu'elles soient de type correction de bug ou création de nouvelles fonctionnalités. Malgré le fait qu'il n'y est pas d'autre développeurs informatiques sur le projet j'ai dû répondre à ces demandes de manière efficaces et rapide. J'ai donc pu corriger toutes les fonctionnalités qui ne répondaient plus aux besoins et développer toutes les améliorations nécessaires aux laboratoires. J'ai même pu présenter ces développements au cours d'une réunion inter-laboratoire et expliquer ces nouvelles fonctionnalités afin qu'elles soient appliquées à tous.

A titre d'exemple, je citerais les informations suivantes, reflétant le travail que j'ai effectué. Afin de vous les présenter, je comparerais le code initial qu'il y avait à mon arrivée et celui que j'ai développé.

Thalesmee en quelques chiffres

Lors de mon arrivée	Actuellement
30 tables dans la base de données	39 tables dans la base de données
137 fichiers pour l'interface du planning (hors bibliothèques et Framework)	163 fichiers pour l'interface du planning (hors bibliothèques et Framework)
Environ 670 lignes de codes pour la génération du planning	Environ 1380 lignes de codes pour la génération du planning
Taille du répertoire /www : 12,7 Mo	Taille du répertoire /www : 13,9 Mo

VI/ Bilan de mon alternance

Durant mon alternance j'ai pu mettre en œuvre toutes les connaissances que j'ai pu obtenir lors de mon cursus scolaire dans un contexte professionnel. Ce fut pour moi une très bonne expérience en matière technique et fonctionnelle.

Lors de cette année j'ai acquis un gain de compétence technique. En effet, cette expérience m'a permis de m'exercer dans le développement web et la gestion de base de données. L'utilisation du langage JavaScript et PHP m'a permis de m'améliorer dans ces deux langages de programmation. De plus, j'ai pu mettre en œuvre des tests pour m'assurer que mon code est juste et réalise correctement ce qu'il doit faire. Ainsi, pour chaque développement d'une fonctionnalité j'ai écrit des tests.

Cette année en alternance m'a permis aussi de gagner en compétence au niveau fonctionnel. En effet, j'ai dû apprendre à m'organiser, à planifier mes tâches et à gérer mon travail. Ainsi j'ai pu utiliser la méthode Agile pour mes développements et ainsi bien comprendre son fonctionnement. Cette méthode m'a permis de gagner beaucoup de temps dans mon travail. De plus en ayant des retours fréquents de la part des utilisateurs j'ai pu réaliser un développement répondant au mieux à leurs besoins.

J'ai aussi dû faire preuve de qualité au niveau de la compréhension des besoins. En effet, il m'a fallu une bonne compréhension de ce que l'utilisateur souhaite. Cette compréhension passe par un travail d'écoute et de communication avec le client. Grâce à cette phase de communication et de dialogue avec l'utilisateur j'ai pu trouver des solutions rapides aux différentes requêtes.

Pour terminer, cette année en alternance, m'a permis de prendre confiance en moi que ce soit en matière de technique avec les progrès que j'ai effectués ou en matière de communication et d'identification des besoins. En effet, étant le seul développeur du département il a fallu que je développe le projet en autonomie.

Thales Alenia Space m'a proposé de rester sur ce projet pour les deux prochaines années de Master ICE. Une proposition que j'ai acceptée car je connais ce projet et je souhaite continuer pour approfondir mon expérience dans ce domaine.

VII/ Annexes

1) Gestion des retards

1.1) Code permettant la mise à jour des retards

```
//Fonction permettant de calculer la différence entre deux dates
function dateDiff( date1, date2 ) {

    var diff = {}                                // Initialisation du retour
    var tmp = date2 - date1;

    tmp = Math.floor(tmp/1000);                  // Nombre de secondes entre les 2 dates
    diff.sec = tmp % 60;                         // Extraction du nombre de secondes

    tmp = Math.floor((tmp-diff.sec)/60);         // Nombre de minutes (partie entière)
    diff.min = tmp % 60;                        // Extraction du nombre de minutes

    tmp = Math.floor((tmp-diff.min)/60);         // Nombre d'heures (entières)
    diff.hour = tmp % 24;                       // Extraction du nombre d'heures

    tmp = Math.floor((tmp-diff.hour)/24);        // Nombre de jours restants

    diff.day = tmp;

    return diff;
}
```

```
/*
DateDebutPrevu : Date prévu pour le début de l'essai
test["pastilleOrange"] : Entier permettant de savoir l'état de la pastille :
    - 0 : Pas de pastille
    - 1 : Pastille présente
    - 2 : Pastille supprimé par l'utilisateur
*/
//Si pas de DateDebutPrevu -> pas de pastille
//Possible pour les essais insérer avant la mise à jour
//Pas de pastille si test["pastilleOrange"] = 2 car l'utilisateur l'a enlevé manuellement
if ( DateDebutPrevu != "null" && test["pastilleOrange"] != 2 ) {

    //Calcul de la différence entre la date de livraison de l'équipement et la date prévu
    var tempsRetard = dateDiff(DateLivraison,DateDebutPrevu);
    //Condition de retard
    //Etat 21 correspondant à un essai réservé. Pour être en retard l'essai doit être au moins dans l'état en attente (22)
    //Vérification que le temps de retard est au moins supérieur à 2 heures
    if (idEtat > 21 && parseInt(tempsRetard.day) < 0 || (idEtat > 21 && parseInt(tempsRetard.day) == 0 && parseInt(tempsRetard.hour) < -2) ||
        (idEtat > 21 && parseInt(tempsRetard.day) == 0 && parseInt(tempsRetard.hour) == -2 && parseInt(tempsRetard.min) < 0 ))
    {

        //Requête AJAX
        var url='majEssaiRetard.php?id='+test["idEssai"];
        $.get(url).fail( function() {

            //Affichage d'un message d'erreur
            var child= document.createElement("div");
            child.className="text-center";
            child.innerHTML="<div class='alert alert-danger'><strong>Erreur Ajax </strong></div>";
            document.body.insertBefore(child, document.body.firstChild);
            window.scrollTo(0,0);

        });

        //Ajout de la pastille orange
        orange = true;
    }
}
```

1.2) Schéma de la table 'essai'

essai
idEssai
idTachePrime
badge
affaire
equipement
os
commentaire
idService_SERVICE
idMoyen_MOYEN
fifo
retard_interne
planifie
idDep_DEPOSITAIRE
date_debut
date_fin
date_debut_prevu
date_fin_prevu
pastilleOrange
pastilleRouge
retardME
ligneProd

2) Gestion des capteurs « Bobines »

2.1) Code JS pour la saisie des bobines

```
}else if (($('#typeC option:selected').text() == "Bobine_Courant") || ($('#typeC option:selected').text() == "Bobine_Tension")){  
  
    $('#bobine').show();  
    $('#nonBobine').hide();  
    $("#type").attr('required', '');  
    $("#nonBobine").removeAttr('required');
```

2.2) Code PHP pour la saisie des bobines

```
}else if (($lg->nomTypeC == "Bobine_Courant" ) || ($lg->nomTypeC == "Bobine_Tension" )){

    $type=htmlspecialchars(mysqli_real_escape_string($bdd,$_POST["type"]));
    $str="Update instrument set modele='$type' where numInstru= '$numInstru'";
    $req=mysqli_query($bdd, $str);
    $axeX=' ';
    $axeY=null;
    $axeZ=null;
    $axeZs=null;
    //Sensibilité de la bobine
    $sensiX=htmlspecialchars(mysqli_real_escape_string($bdd,$_POST["bobine"]));
    $sensiY=null;
    $sensiZ=null;
    $sensiZs=null;
    $unite=htmlspecialchars(mysqli_real_escape_string($bdd,$_POST["unite"]));
    $dateSensi=dateFrToSQL(htmlspecialchars(mysqli_real_escape_string($bdd,$_POST["dSensi"])));
```

3) Retard moyen d'essais

```
/*
DateFinPrevu : Date prévu pour la fin de l'essai
test["retardME"] : Entier permettant de savoir l'état de la pastille :
    - 0 : Pas de pastille
    - 1 : Pastille présente
    - 2 : Pastille supprimé par l'utilisateur
*/
//Si pas de DateFinPrevu -> pas de pastille
//Possible pour les essais insérer avant la mise à jour
//Pas de pastille si test["retardME"] = 2 car l'utilisateur l'a enlevé manuellement
if (DateFinPrevu != "null" && test["retardME"]!=2){

    var tempsRetard = dateDiff(DateTermine,DateFinPrevu);
    //Condition de retard
    //Etat supérieur à en attente (22)
    if (idEtat > 22 && parseInt(tempsRetard.day) < 0 || (idEtat > 22 && parseInt(tempsRetard.day) == 0 && parseInt(tempsRetard.hour) < 0) ||
        (idEtat > 22 && parseInt(tempsRetard.day) == 0 && parseInt(tempsRetard.hour) == 0 && parseInt(tempsRetard.min) < 0 ))
    {
        var url='majEssaiRetard.php?retard=true&id='+test["idEssai"];
        $.get( url).fail(function() {

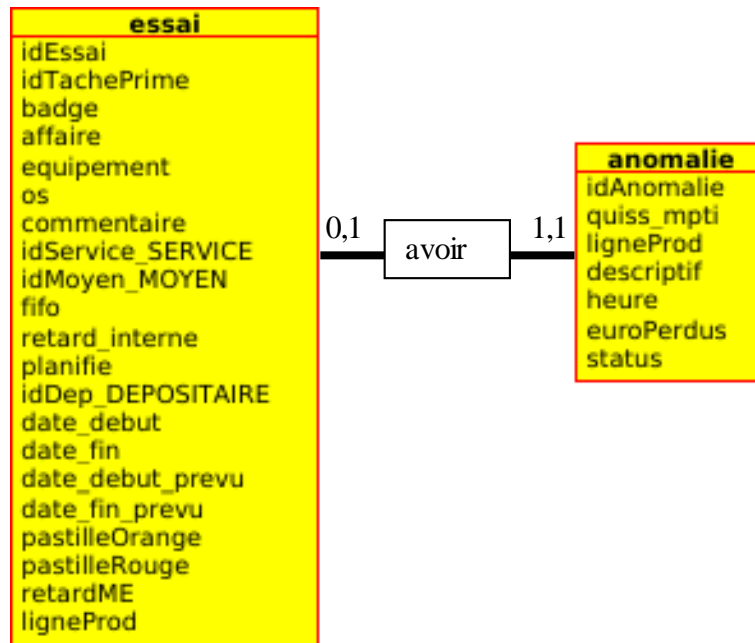
            var child= document.createElement("div");
            child.className="text-center";
            child.innerHTML="<div class='alert alert-danger'><strong>Erreur Ajax </strong></div>";
            document.body.insertBefore(child, document.body.firstChild);
            window.scrollTo(0,0);

        });
        retardME = true;
    }
}
```

4) Gestion des anomalies

4.1) Schéma de la table anomalie

Schéma Entité-Association :



Relationnel :

Les cardinalités des deux côtés étant 0,1 ou 1,1 le modèle relationnel est donc de la forme :

anomalie (idEssai, quiss_mpti, autre, descriptif, heure, eurosPerdus, status)

4.2) Gestion de l'intervalle de date

```
if (isset($_GET["date_deb"]) && isset($_GET["date_fin"])){

    //Récupération des dates du formulaire

    //date_create fonction php permettant de créer un date avec un chaîne de caractère
    $date_deb = date_create($_GET["date_deb"]);
    $date_fin = date_create($_GET["date_fin"]);
    //Récupération de la date au format fr pour le formulaire
    $dateDebForm = date_format($date_deb, "d/m/Y");
    $dateFinForm = date_format($date_fin, "d/m/Y");

    //Récupération de la chaîne de caractère
    $date_deb = $_GET["date_deb"];
    $date_fin = $_GET["date_fin"];

}else{

    //Date de début date (date actuelle moins deux mois)
    $date_deb = strftime("%Y-%m-%d", mktime(0,0,0,date('m')-2, date('d'), date('Y')));
    //Date de fin (date actuelle)
    $date_fin = strftime("%Y-%m-%d", mktime(0,0,0,date('m'), date('d'), date('Y')));

    //formatage pour la valeur des input
    $dateDebForm=explode("-", $date_deb);
    $dateDebForm=$dateDebForm[2]."/". $dateDebForm[1]."/". $dateDebForm[0];
    $dateFinForm=explode("-", $date_fin);
    $dateFinForm=$dateFinForm[2]."/". $dateFinForm[1]."/". $dateFinForm[0];

}
```

5) Code des indicateurs

Code de génération de l'indicateur de calcul du temps d'attente sur l'étagère avant le test.

```
//Initialisation du nombre de jour d'attente à 0
$ج0=0;
$ج1=0;
$ج2_3=0;
$ج4_5=0;
$جsup6=0;
$nbجMoy=0;
$nbTest=0;
//Booleen permettant de voir si les deux états sont entre les dates demandées.
//Dans certain cas il est possible que l'état 23 soit dans l'intervale de date mais pas l'état 22
$correct = false;
while ($لg=$res->fetch(PDO::FETCH_OBJ))
{
    $dateEtat=$لg->dateEtat;
    $idEtat=$لg->idEtat_etat;
    if ($idEtat==23 and $correct == true)
    {
        //dateEtat contient la date de fin d'attente
        //Calcul de la différence
        $nbج=floor((strtotime($dateEtat) - strtotime($datePrecedente))/86400);
        $nbجMoy+=$nbج;
        $nbTest++;
        if ($nbج==0)
        {
            $ج0++;
        }
        elseif ($nbج==1)
        {
            $ج1++;
        }
        elseif ($nbج <4)
        {
            $ج2_3++;
        }
        elseif ($nbج<6)
        {
            $ج4_5++;
        }
        else
        {
            $جsup6++;
        }
        $correct = false;
    }
    else if ($idEtat==22) {
        $correct = true;
        $datePrecedente=$dateEtat;
    }
}
```

Code de génération du graphique à l'aide de la bibliothèque JpGraph en PHP :

```
$graph = new Graph(800,600,'auto');
$graph->SetScale("textlin");

$theme_class=new UniversalTheme;
$graph->SetTheme($theme_class);
$graph->xaxis->SetFont(FF_ARIAL,FS_NORMAL,9);
$graph->yaxis->title->SetFont(FF_ARIAL,FS_BOLD,12);
$graph->title->SetFont(FF_ARIAL,FS_BOLD,16);
if($fifo==0)
    $titre="Durée d'attente de l'equipement sur l'étagère $labo avant le test";
else if ($fifo == 1)
    $titre="Durée d'attente de l'equipement sur l'étagère $labo FIFO avant le test";
else
    $titre="Temps d'attente tout tests confondus avant le test";

$graph->title->Set($titre);
$graph->SetBox(false);
$graph->ygrid->SetFill(false);
$graph->xaxis->SetTickLabels(array('0 jour','1 jour','2 à 3 jours','4 à 5 jours','>5 jours'));
$graph->yaxis->HideLine(false);
$graph->yaxis->HideTicks(false,false);
$graph->yaxis->title->Set("Nombre de tests");

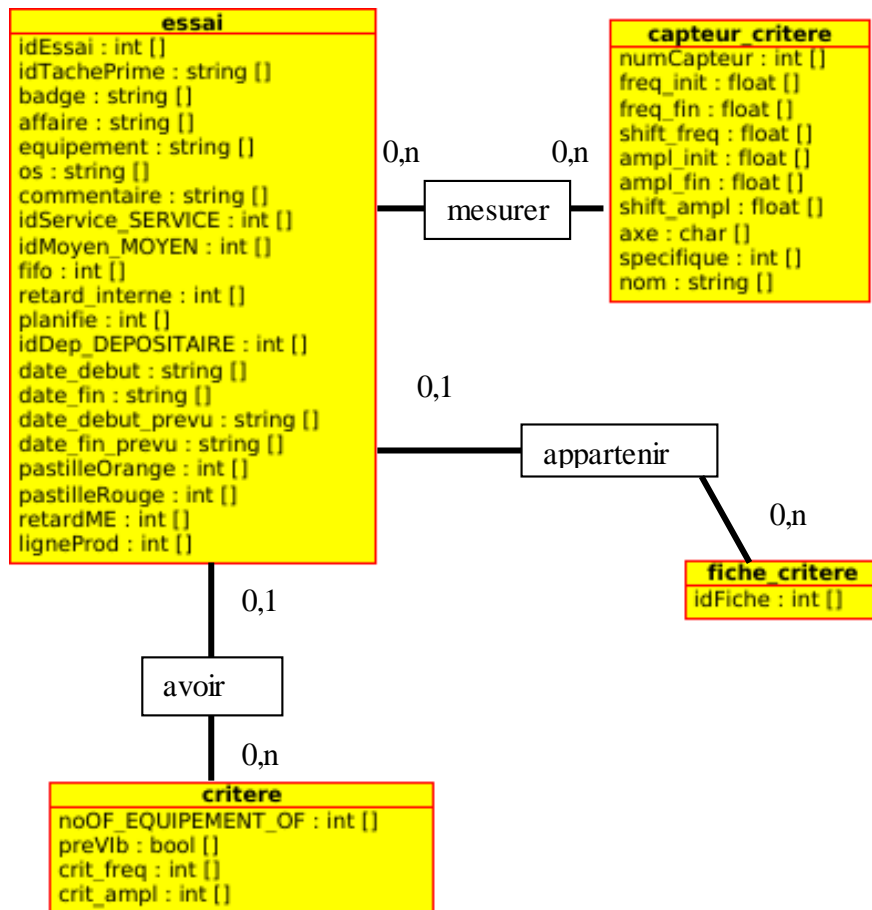
//Creation des barPlot
$b1plot = new BarPlot($data1y);

//Creation du groupe de barPlot
$gbplot = new GroupBarPlot(array($b1plot));

//ajout au graph
$graph->Add($gbplot);

//ajout de la legende
$b1plot->SetLegend('Test terminé');
$graph->legend->SetFrameWeight(1);
$graph->legend->SetColumns(6);
$graph->legend->SetColor('#4E4E4E','black');
$graph->legend->SetLayout(LEGEND_VERT);
$graph->legend->SetFont(FF_ARIAL,FS_NORMAL,11);
```

6) Conception fiche critère



7) Affichage du planning mensuel

1) Affichage des boutons de navigation

```
if ($affichage == "semaine"){

    echo '<button type="button" class="btn btn-default btn-arrow-left" id="sem_prec" >Semaine précédente</button>'
    echo '<button type="button" class="btn btn-default btn-arrow-left" id="jour_prec" >Jour précédent</button>'
    echo '<button type="button" class="btn btn-link" id="infos_date">Link</button>'
    echo '<button type="button" class="btn btn-default btn-arrow-right" id="jour_suiv">Jour suivant</button>'
    echo '<button type="button" class="btn btn-default btn-arrow-right" id="sem_suiv">Semaine suivante</button>';

}else {

    echo '<button type="button" class="btn btn-default btn-arrow-left" id="mois_prec" >Mois précédent</button>'
    echo '<button type="button" class="btn btn-default btn-arrow-left" id="sem_prec" >Semaine précédente</button>'
    echo '<button type="button" class="btn btn-link" id="infos_date">Link</button>'
    echo '<button type="button" class="btn btn-default btn-arrow-right" id="sem_suiv">Semaine suivante</button>'
    echo '<button type="button" class="btn btn-default btn-arrow-right" id="mois_suiv">Mois suivant</button>';

}
```

2) Génération des entêtes du tableau

```
for ($i=0; $i<$nbJours; $i++){

    if ($affichage == "mois"){

        echo '<th rowspan="2"></th>';

    }else{

        echo '<th rowspan="2" colspan="2"></th>';

    }

}
```

3) Requête pour récupérer les essais au mois

```
$idMoyen=$lgMoyen->idMoyen;
$nomMoyen=$lgMoyen->nomMoyen;
$nbJours = intval(date("t",$dateFin));

if($idMoyen=="")
    $t="e.idMoyen_Moyen is null";
else
    $t="e.idMoyen_Moyen='$idMoyen'";

// on recupere les moyens
$str="SELECT e.idEssai, et.idetat_etat, e.date_debut, e.date_fin, e.equipement, e.affaire, e.retard_interne, e.planifie,
e.pastilleOrange, e.pastilleRouge, e.retardME, e.commentaire
from essai e, etatessai et
where
    e.idService_SERVICE='$labo'
and et.idEssai_ESSAI=e.idEssai
and et.idetat_etat in ($typeEssai)
and e.date_fin>='$dateAct'
and e.date_debut < DATE_ADD('$dateAct', INTERVAL ".$nbJours." DAY)
and et.idetat_etat=(select max(idetat_etat) from etatessai where idEssai_ESSAI=e.idEssai)
and (e.planifie=1 or (e.planifie=0 and et.idetat_etat!=22))
and $t
order by date_debut;"
```

4) Affichage du tableau

```
while(i<nbJDeb)
{
    var newCell=newRow.insertCell(-1);
    newCell.className="jours";

    if (affichage == "semaine"){

        var newCell2=newRow.insertCell(-1);
        newCell2.className+="jours";

    }

    if(contains(tabWeek,i+1))
    {
        newCell.className+=" cellWeekEnd";
        if (affichage == "semaine"){

            newCell2.className+=" cellWeekEnd";

        }
    }

    i++;
}
```

```

while(i<nbJour)
{
    var newCell=newRow.insertCell(-1);
    newCell.className="jours";

    if (affichage == "semaine"){

        var newCell2=newRow.insertCell(-1);
        newCell2.className+="jours";
    }

    if(contains(tabWeek,i+2))
    {
        newCell.className+=" cellWeekEnd";

        if (affichage == "semaine"){

            newCell2.className+=" cellWeekEnd";
        }
    }

    i++;
}

```

5) Code JavaScript pour gérer les évènements au clique

```

if (affichage == "mois"){

    $("#mois_suiv").click(function(){
        var url='focusDate.php?mois=1';
        $.get( url, function(data) {
            ladate.setMonth(ladate.getMonth()+1);
            creerTab();
        });
    });

    $("#mois_prec").click(function(){
        var url='focusDate.php?mois=0';
        $.get( url, function(data) {
            ladate.setMonth(ladate.getMonth()-1);
            creerTab();
        });
    });
}

```