

RAPPORT DE STAGE - TN09 - GI

Gestion et optimisation du reporting au sein d'un projet de transport

LUSAIL LRT - THALES GULF (QATAR)

3 SEPTEMBRE 2018 - 15 FÉVRIER 2019 (A18)

Auteur :

Lucas LAMY

Superviseurs :

Vincent ARGENTON

Stéphane GLOAGUEN

Suiveur :

Mohamed SALLAK

Remerciements

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui m'ont aidé lors de la rédaction de ce rapport.

Tout d'abord, je tiens à remercier vivement Vincent ARGENTON, mon maitre de stage, pour le temps qu'ils m'a toujours accordé sans hésitation, ainsi que pour les opportunités et responsabilités qu'il m'a offertes.

Ensuite, je voudrais remercier le manager de l'équipe, Stéphane GLOAGUEN, pour la reconnaissance, le temps et les conseils qu'il m'a accordé.

De plus, je souhaite aussi remercier Kevin HELOUART, qui, avec Vincent, a su m'intégrer à l'équipe très rapidement.

Enfin, je remercie Michelangelo NERI et Milan RADOVIC d'avoir accepté ma candidature.

Je souhaite aussi remercier tout mes collègues pour le temps passé à leurs côtés, ainsi que pour leurs précieux conseils, et plus spécialement Thibault, Bastien et Dominique.

Sommaire

Résumé technique	1
1 Présentation générale	2
1.1 Le groupe Thales	2
1.2 Le projet : Lusail LRT	4
1.3 Présentation de l'équipe T & C	8
2 Mission	10
2.1 Sujet du stage	10
2.2 Planning	10
2.3 Contributions	11
2.4 Technologies	12
2.5 Prise de recul	14
3 Réalisations	15
3.1 Conception d'un processus de production et de gestion des rapports de test	15
3.2 Mise au point d'un outil d'extraction de données	19
3.3 Communication sur l'avancement du projet	26
3.4 Formation et transmission	27
3.5 Tuilage	27
Conclusion	28
Glossaire	29
Bibliographie	
Annexes	

Résumé technique

Lors de mon TN09, j'ai effectué mon stage chez Thales Gulf, à Doha, au Qatar. Le projet auquel j'ai été affecté est un projet ferroviaire de transport léger sur rail (LRT) pour lequel Thales prend en charge la majorité des technologies de télécommunications. L'équipe T&C (test et mise en service), dans laquelle j'ai évolué, est chargée de mettre en service et de tester les différents équipements des nombreux systèmes gérés par Thales. Ma fonction principale était de rapporter l'avancement de la production des rapports de tests, afin, par exemple, de fournir des outils d'aide à la décision, ainsi que des indicateurs de progrès, à mes managers.

Afin de mener à bien cette mission, j'ai utilisé 2 bases de données différentes. L'une regroupait tous les documents produits au cours du projet, l'autre permettait aux différentes équipes d'ingénieurs d'éditer et de stocker leurs rapports de tests.

C'est sur la seconde base de données que la majorité de mon travail s'est concentrée. En effet, les informations indexées dans celle-ci correspondaient à une extraction des données brutes présentes dans les formulaires composant chaque rapport. Ainsi, un de mes projets fut de mettre au point un programme capable d'analyser ces données, les relier aux différentes variables, puis les intégrer au sein d'un fichier Excel afin de pouvoir fournir différentes statistiques et outils d'aide à la décision.

Ce stage m'a amené à découvrir de nombreuses structures ainsi que des domaines d'ingénierie informatiques particulièrement variés, j'ai également assumé différentes responsabilités, qui ont contribué à enrichir cette première expérience professionnelle.

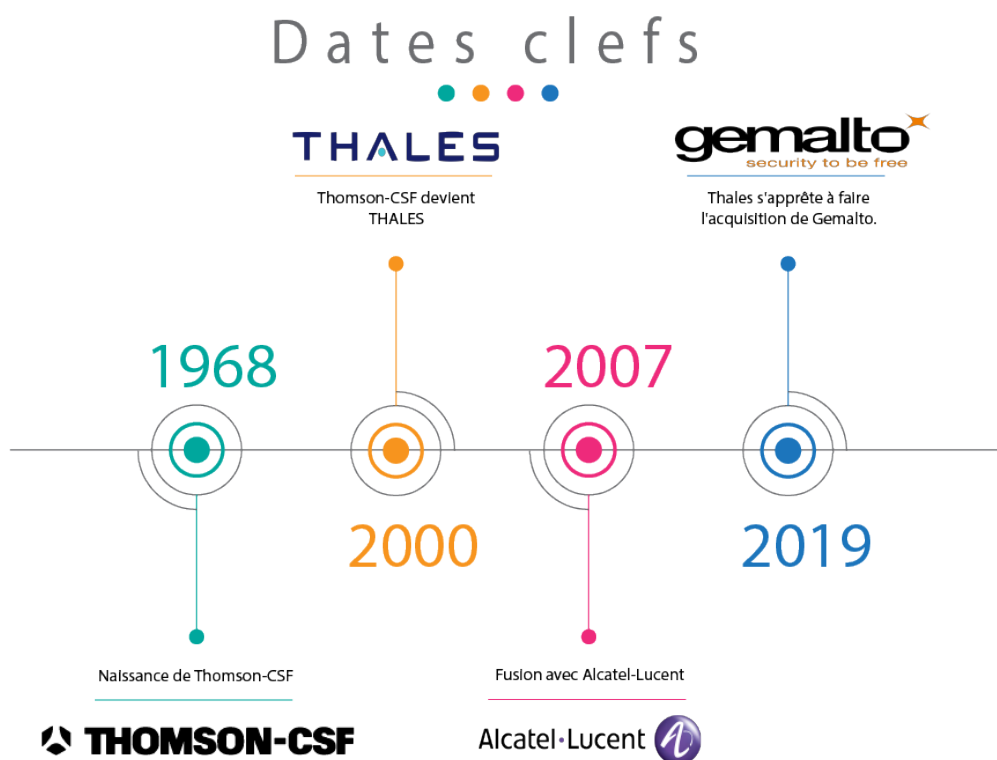
Dans ce rapport, je vous présenterai le contexte de mon stage, la nature de mes missions, mes réalisations et enfin les conclusions que j'en ai tirées.

1. Présentation générale

1.1 Le groupe Thales

1.1.1 Historique du groupe

Le groupe Thales actuel est la résultante d'une évolution complexe, jalonnée par plusieurs fusions dont voici les principales :



- **1968** : La branche électronique de Thomson-Brandt et la Compagnie Générale de Télégraphie fusionnent pour créer Thomson-CSF.
- **2000** : Thomson-CSF fusionne avec Dassault Électronique et devient THALES.
- **2007** : Les activités d'Alcatel-Lucent liées au transport, à l'espace ainsi qu'à la sécurité sont transférées vers Thales par le biais d'une fusion, qui mène aussi à la création de Thales Alenia Space.

- **2019** : Après l'annonce en 2017 d'une offre de rachat de Gemalto, Thales a obtenu de la Commission européenne la validation de l'OPA. Cette dernière devrait avoir lieu au premier trimestre de cette année.

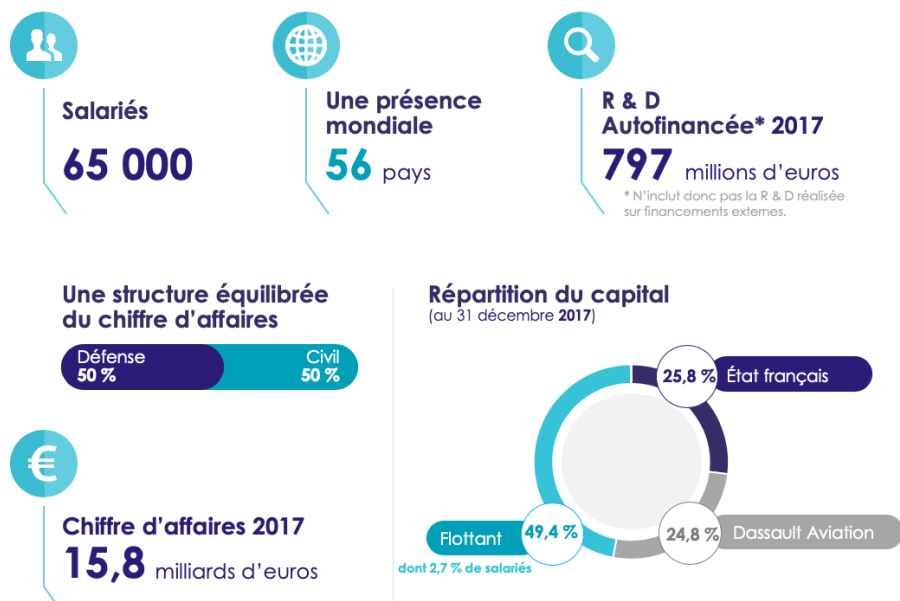
1.1.2 Les domaines d'activités

Thales est implantée dans 56 pays, et cela grâce à un large spectre de compétences, divisé en 5 secteurs d'activité qui sont :

- **La défense** : Thales est numéro 1 mondial dans la défense aérienne avancée et numéro 1 en Europe concernant l'électronique de défense. Les solutions conçues, déployées et maintenues par Thales concernent à la fois l'aérien, le terrestre, le spatial, le maritime ainsi que les communications radio, les systèmes interarmées et la cybernétique.
- **La sécurité** : en 2011, deux sociétés du groupe Thales fusionnent pour former Thales Communications & Security dont les domaines d'activité sont multiples : la sécurité urbaine, la protection d'infrastructures critiques telles que les aéroports, la sécurité des systèmes de communication ainsi que la cybersécurité.
- **L'aéronautique** : Thales domine le domaine de la gestion du trafic aérien, deux tiers des avions possèdent à minima un équipement Thales que ce soit pour la navigation, la génération d'électricité, les solutions de divertissement ou encore les cockpits.
- **L'aérospatiale** : la présence du groupe dans l'aérospatiale civile et militaire est assurée par Thales Alenia Space, un des leaders mondiaux dans la conception de satellites de communication. Cette société propose aussi des systèmes pour l'observation de la Terre, l'exploration, la navigation, le transport ainsi que les infrastructures spatiales (comme l'ISS).
- **Les transports terrestres** : Thales élabore des solutions pour les grandes lignes ferroviaires, mais aussi pour des moyens de transports urbains comme le métro, le tramway ou encore le LRT. Le groupe développe des solutions innovantes, fondées sur des technologies de pointe comme le Système Européen de Contrôle des Trains (ETCS).

1.1.3 Position du groupe sur la scène internationale

Voici les chiffres réalisés par Thales en 2017 :



En ce qui concerne la concurrence, Thales se positionne de la manière suivante :

- **Pour la signalisation ferroviaire** : même si Thales réalise un chiffre d'affaires annuel de 1,7 milliard d'euros, c'est son avance technologique dans le domaine des trains autonomes qui lui permettra de rivaliser avec le géant Siemens Alstom.
- **Dans l'aéronautique** : les trois leaders français dans ce domaine sont Thales, Safran et Zodiac. Ainsi Thales se partage le marché des équipementiers civils, mais possède une position de poids dans le secteur de l'aéronautique militaire et plus particulièrement dans la défense, notamment grâce à la part importante de son capital dédiée à la partie Dassault Aviation du groupe.

1.2 Le projet : Lusail LRT

1.2.1 Le Light Rail Transit

Le Transit Léger sur Rail ou Métro léger est une technologie de transport ferroviaire urbain. Sa vitesse moyenne et sa capacité de transport sont en général supérieures à celle d'un tramway, mais inférieures à celles d'un métro. Le LRT adopte le fonctionnement d'un métro sur les tronçons souterrains et celui du tramway sur ceux en surface. Du point de vue du fonctionnement, la différence principale entre le LRT et le métro réside dans le fait que plusieurs lignes différentes peuvent circuler sur des tronçons de voies communs. De nombreuses villes dans le monde ont déjà implanté un LRT dans leur réseau de transport en commun, telles que Hong Kong, Francfort, São Paulo, Ottawa ou encore Jérusalem.

1.2.2 Le contexte du projet

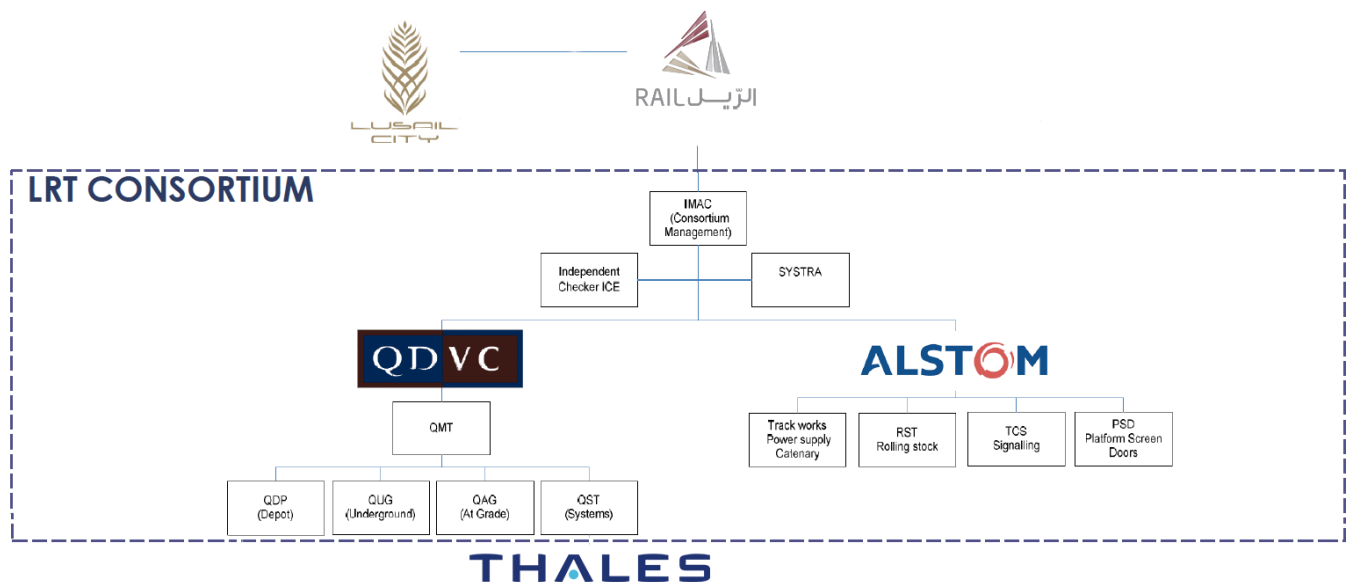
Le projet se déroule dans la nouvelle ville de Lusail au Qatar située en périphérie de Doha, la capitale du pays. Cette ville, lorsqu'elle sera terminée, s'étendra sur 38 km^2

et devrait accueillir 450 000 habitants. Elle tiendra une place centrale dans les activités culturelles, artistiques et sportives du pays avec par exemple le circuit international de Lusail, mais aussi le stade de football qui accueillera le match d'ouverture et la finale de la Coupe du Monde de football en 2022. Dans l'optique de désengorger le trafic routier ainsi que de préparer au mieux l'accueil de la Coupe du Monde, la ville de Doha se verra équipée prochainement de 3 lignes de métro et la ville de Lusail se verra équipée de 4 lignes de LRT. Le projet concerné par ce rapport est celui du LRT de Lusail. Ce dernier comportera 4 lignes, 37 stations (dont 2 reliées aux réseaux du métro de Doha), 28 rames pour une capacité totale de 50 000 voyageurs par jours.



1.2.3 Les acteurs du projet

De nombreuses entreprises sont engagées sur ce projet de transport, puisque celui-ci implique à la fois du génie civil, de l'ingénierie ferroviaire, des télécommunications ainsi que de la sécurité.



- **Qatar Rail** : entreprise créée après le début de la phase de définition du projet, afin de superviser l'ensemble du réseau ferroviaire du pays. C'est l'employeur et donc le client direct du Consortium.
- **IMAC** : entité de management du consortium, elle est composée à la fois d'employés de QDVC et d'Alstom, les deux entreprises principales du consortium et fait l'intermédiaire entre le consortium et le client.
- **ICE** : entité de contrôle dont le rôle est assuré par la société SENER. Elle évolue de manière externe au consortium et donc indépendamment de celui-ci afin de pouvoir certifier la conformité du système par rapport aux exigences du contrat au client Qatar Rails.
- **SYSTRA** : entreprise réalisant pour l'IMAC des missions d'audit concernant le design. Elle est impliquée durant la phase de définition du projet en réalisant les différentes spécifications présentes dans le contrat, mais aussi durant la phase de réalisation afin d'orienter les stratégies d'opération.
- **Alstom** : Alstom prend à charge toute la partie ferroviaire du projet : les rames (RST : matériel roulant), la signalisation (TCS : Système de contrôle du tramway), les portes palières (PSD : portes palières) ou encore les caténaires.
- **QDVC** : QDVC est une filiale à 51% de Qatari Diar et à 49% de VINCI Construction Grands Projets. Elle prend à charge notamment le génie civil pour les bâtiments du dépôt, les bâtiments sous-terrains ainsi que les bâtiments à la surface, que ce soit des stations ou des bâtiments techniques. Cette entreprise sous-traite trois secteurs d'activité :
 - CCS : Systèmes de contrôle et de communication, cette partie est assurée par Thales.
 - TVS : Système de Ventilation des Tunnels.

- **ECS** : Système de Contrôle Environnemental (température, humidité, etc.)
- **RKH** : fruit de la fusion entre le consortium RATP Dev et Keolis (49%) et la société qatarie Hamad Group (51%), RKH assurera l'exploitation et la maintenance du LRT de Lusail ainsi que du Métro de Doha.

1.2.4 Place de Thales au sein du projet

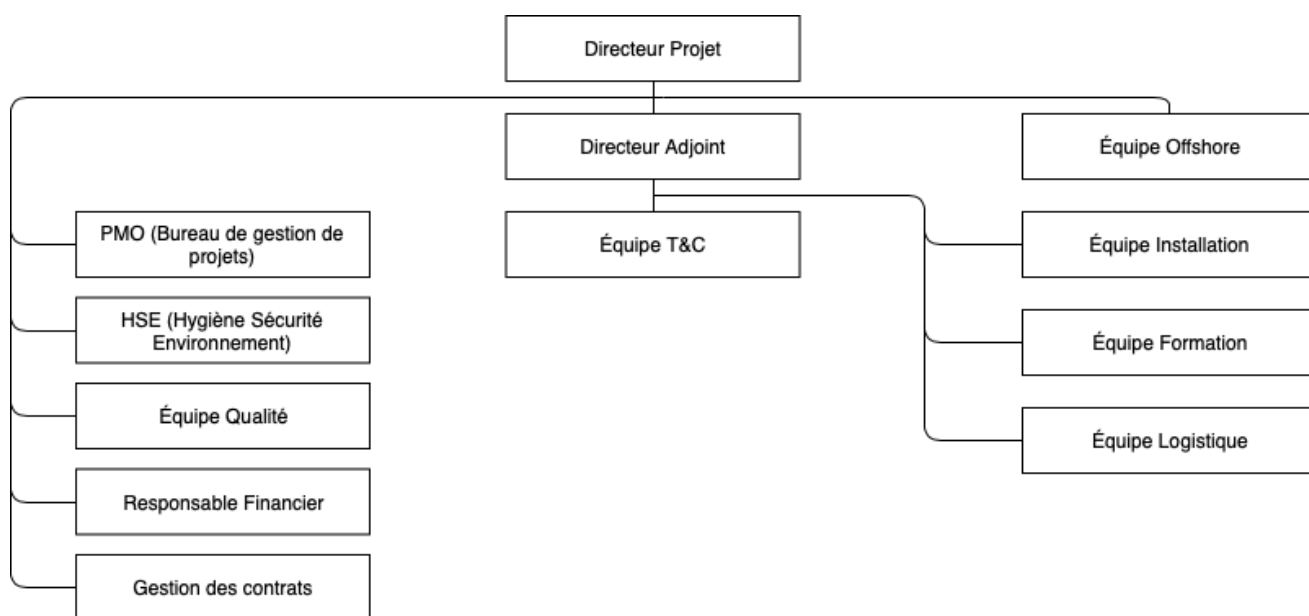
Sur ce projet, l'entreprise est positionnée en tant que sous-traitant de QDVC. Elle est en charge du design, de l'installation, des tests et de la mise en service des systèmes **CCS**. Ces derniers sont organisés comme suit :

- **Télécommunications** : secteur d'activité clé de Thales, il représente une part importante du projet.
 - **DTS** : Système de Transmission Digitale, ensemble des différentes infrastructures réseau.
 - **Téléphonie** : Ce secteur englobe les téléphones présents dans les ascenseurs, dans les centres opérationnels ou encore la gestion de la téléphonie embarquée dans les rames lorsqu'elles circulent à la surface ou sous terre.
 - **TETRA** : Système de radio mobile numérique destiné aux futurs opérateurs du système afin de leur assurer un moyen de communication fiable et sécurisé.
 - **BBRS** : Radio à large bande.
 - **WA** : Système permettant de proposer aux usagers du LRT de bénéficier d'un accès à Internet via un réseau WiFi.
 - **COMTV** : Ensemble des écrans situés en station et à bord des rames ayant pour fonction de diffuser des annonces publicitaires aux usagers.
 - **UPS** : Système d'alimentation sans interruption permettant de garantir la stabilité du réseau électrique, quelles que soient les interactions des différentes entités avec celui-ci.
 - **PIS-PAS** : Systèmes d'information visuelle et sonore destinés aux passagers, mais aussi plus généralement au public.
- **Sécurité** : les trois systèmes de ce secteur sont implantés dans toutes les localisations du projet, c'est à dire les stations, les centres opérationnels, les bâtiments techniques et ceux du dépôt.
 - **ACS-IDS** : Système de Contrôle des Accès ainsi que de Détection des Intrusions.
 - **FDS** : système de détection des incendies.
 - **CCTV** : système de vidéosurveillance.
- **Supervision** : ce secteur est primordial sur les projets ferroviaires urbains, car c'est lui qui permettra aux futurs opérateurs de gérer tout type situation.

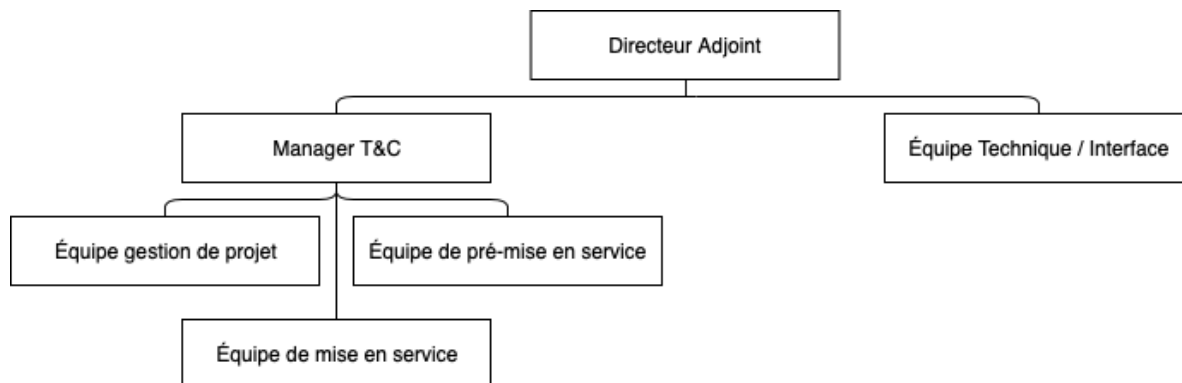
- **SCADA** : Système de Contrôle et d’Acquisition de Données, solution mise au point par Thales.
- **MMS** : Système de Management de la Maintenance.
- **Billettique** : **AFC** : Système de Collection Automatique des Billets

1.3 Présentation de l’équipe T & C

Sur ce projet, Thales fonctionne avec des équipes offshore (Thales France, Thales Italy, Thales Portugal) et une équipe inshore sur place rattachée à Thales Gulf. Ainsi, l’équipe projet est organisée comme ci-dessous :



En ce qui concerne l’équipe T&C, sa fonction principale est de tester et de mettre en service les systèmes et équipements CCS. Son organisation, qui compte environ une trentaine de personnes, est la suivante :



QDVC est le client direct de Thales, mais agit aussi en tant qu’autorité de contrôle durant les différentes phases du projet, en suivant un cycle en V (cf. Annexe I). Concentrons-nous ici sur la phase de test : chaque test nécessite le témoignage et l’approbation de

QDVC et de ICE, l'entité de contrôle indépendante. Dans le but de pouvoir mettre en application ce système d'approbation, le client a fait le choix d'une base de données permettant l'édition dynamique (et donc la signature) de rapports de test au format PDF : SnagR. Afin d'assurer la communication et le contrôle des documents du référentiel projet, une base de données commune est utilisée : Mezzoteam, mais Thales utilise aussi en parallèle une base de données indépendante du client : e-TOL.

Les méthodes utilisées à l'échelle macroscopique du projet sont des méthodes de gestion de projet dites "classiques" : on peut par exemple noter l'utilisation de Primavera, un outil permettant de générer un planning de type Gantt et proposant des outils d'estimation des coûts. Au sein de l'équipe T&C, de nombreux outils sont utilisés (dont certains sont issus des méthodes agiles, mais nous reviendront sur ce point dans la prochaine partie) :

- **Git** : Outil de gestion des versions et configurations logicielles.
- **Doors** : logiciel de gestion des exigences.
- **Giro** : logiciel de gestion des risques.
- **Jira** : solution de gestion des incidents techniques.
- **Palma** : logiciel de gestion des configurations.

Au sein de cette équipe, j'ai été encadré par mon maître de stage, Vincent ARGENTON et par Stéphane GLOAGUEN.

M.Argenton est le manager technique du projet du LRT de Lusail, sur lequel il travaille depuis 4 ans et travaillait auparavant, toujours pour Thales, sur le projet du LRT de Panama. Il est diplômé de l'école d'ingénieur ECE Paris et a également obtenu un master de Management de Projet International à l'ESCP Europe, à Madrid.

M.GLOAGUEN est le chef du département T&C du projet, il occupait la même position sur le projet du Métro de Doha et travaille pour Thales depuis plus de 10 ans.

2. Mission

2.1 Sujet du stage

Le sujet de mon stage était la prise en charge du suivi, de la gestion et de l'optimisation du reporting des différents tests de l'équipe T&C.

Chaque rapport de test se décompose en plusieurs étapes de test, qui sont toutes reliées à des exigences du cahier des charges. Pour chaque test, QDVC et ICE déterminent si oui ou non les résultats permettent de considérer le rapport comme accepté. Ainsi, chaque rapport constitue une preuve nécessaire à la validation des exigences sans laquelle le système ne peut être validé.

Chaque rapport est édité et consigné grâce à SnagR, un outil initialement destiné aux projets de génie civil et à la gestion des problèmes d'installation, dont le choix revient à QDVC.

Or le secteur d'activité de Thales n'étant pas le génie civil, l'utilisation seule de cet outil ne suffisait pas à satisfaire les besoins du projet.

C'est ce qui m'a conduit à me concentrer sur la problématique suivante : comment suivre efficacement la production et la validation des rapports tout en utilisant un outil imposé par le client ?

2.2 Planning

En termes de délais, il m'a fallu incorporer les différentes solutions que j'ai mises au point au fur et à mesure du projet, tout en proposant une version finale environ un mois avant la fin de mon stage afin de pouvoir suffisamment l'éprouver. Le but à court terme était de mettre en place un processus efficace de traitement des rapports et de leurs données puis de l'intégrer au fonctionnement global du projet. Dans un second temps, le but à long terme était d'automatiser une partie de ce processus, notamment l'extraction et le traitement des données des rapports. Enfin, au cours du dernier mois, il m'a fallu former mon remplaçant tant au fonctionnement du reporting qu'à l'utilisation des différents outils que j'ai créés. Chacune de ces trois grandes étapes comporte son lot de difficultés :

- **L'initialisation du processus** : L'ampleur du projet constitue une première difficulté. En effet, la tâche qui m'a été confiée demande une bonne connaissance non

seulement du fonctionnement interne du projet, mais aussi de l'état actuel de son avancement.

- **Son automatisisation** : La base de donnée avec laquelle je travaillais n'avait pas été documentée, ainsi l'extraction de données s'est avérée complexe.
- **Sa transmission** : Former mon remplaçant tout en continuant à travailler a constitué un réel défi pour ma première expérience professionnelle.

2.3 Contributions

Au début du stage, l'installation des équipements et systèmes du CCS était depuis quelques mois à un stade suffisamment avancé pour permettre de débiter les phases de test. Ainsi la production des rapports était amorcée. Mes managers m'ont transmis la documentation nécessaire à l'initialisation de mon projet et m'ont guidé au cours du stage afin de me permettre d'y apporter des améliorations et de nouvelles fonctionnalités. Chaque semaine je devais d'envoyer un rapport sur l'avancement du reporting à différents managers de Thalès ainsi qu'au client (QDVC). Ces rapports avaient pour fonction l'aide à la décision et comportaient de nombreux indicateurs d'avancement (KPI : Key Progress Indicator). Je travaillais conjointement avec :

- Le service de gestion des documents afin de gérer l'indexation des rapports ainsi que leur mise en ligne sur Mezzoteam et e-TOL.
- Le service informatique de QDVC afin de proposer des améliorations de la base de données SnagR, mais aussi pour leur faire remonter les différentes erreurs de l'interface web.
- L'équipe RAMS (en français : Fiabilité Maintenabilité Disponibilité et Sécurité), chargée de la sûreté de fonctionnement, afin d'effectuer le suivi de la production des documents permettant de prouver la conformité du système en termes de sécurité.
- L'équipe d'ingénieurs Système, qui est responsable du design, avec qui j'ai étudié différentes exigences.
- L'équipe d'ingénieurs T&C afin d'identifier les différents facteurs bloquants ralentissant la production des rapports.

En parallèle de ces différentes activités, il m'a fallu automatiser l'extraction des données des rapports. Pour cela j'ai développé un script permettant de se connecter au site SnagR, d'en extraire différents lots de données et de les exporter dans différents fichiers Excel. Il m'a fallu aussi ajouter à cette application une fonctionnalité de sauvegarde complète de rapports, en cas de problème avec les serveurs SnagR, serveurs auxquels Thalès n'avait pas accès physiquement. Ce script est exécutable à travers l'invite de commande.

À la fin de mon stage, la phase de test est avancée à plus de 50% et le processus de production et des suivis des rapports de test est en place et fonctionnel.

Comme nous le verrons plus tard j’ai formé la personne me remplaçant à son utilisation, mais j’ai aussi préparé une documentation.

Ainsi lors du tuilage mon remplaçant utilisait déjà les résultats de mon travail.

2.4 Technologies

Durant mon stage, j’ai donc utilisé :

— **Pour le développement :**

- *Comme langage de programmation* : Python 2.7 puis Python 3.7.
- *Comme environnement de développement* : PyCharm, IDE de la suite IntelliJ, développé par JetBrains.

— **Pour la bureautique** : La suite Microsoft Office, plus particulièrement Excel, pour la communication et l’échange d’informations internes au projet, et TexMaker, un éditeur de documents LaTeX, pour la rédaction de ce rapport.

— **Pour la gestion de mes tâches** : Trello, une solution en ligne de gestion de projet s’inscrivant dans le cadre des méthodes agiles.

— **Pour la gestion des différentes versions du script** : La technologie Git, en utilisant l’hébergeur GitHub.

Au sein de l’équipe T&C, les méthodes agiles ont été utilisées notamment par le biais d’outils comme SnagR ou Jira. En effet, SnagR permet aussi de répertorier les problèmes ou points bloquants que les différentes équipes du projet rencontrent.

SnagR est la solution choisie par le Consortium, mais pour son fonctionnement interne, Thales a choisi d’utiliser Jira Ops , qui permet à l’équipe inshore de faire remonter les différents problèmes rencontrés à l’équipe offshore, sous forme de PCR.

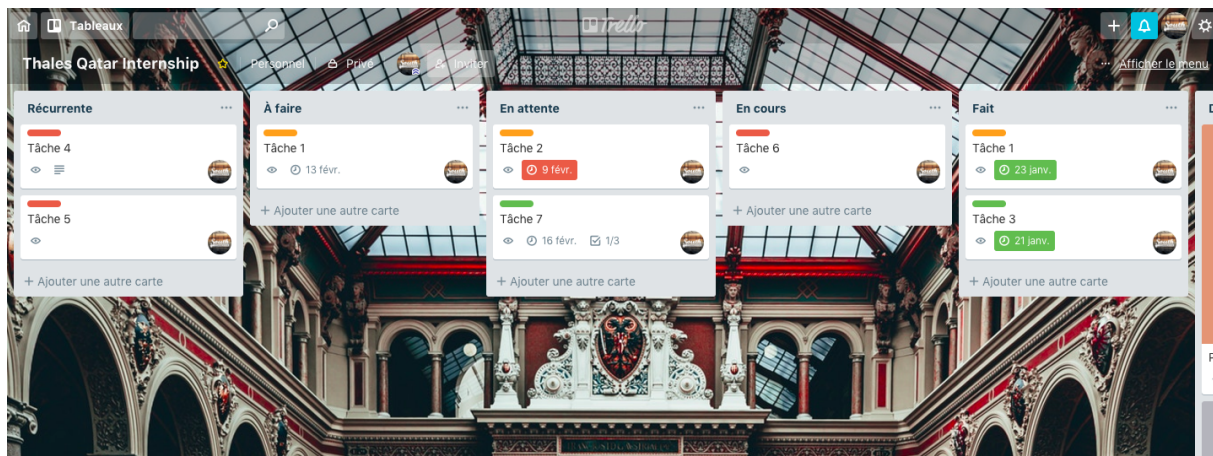
Les trois outils que sont SnagR, Jira et Trello ¹intègrent tous les méthodes agiles, et plus particulièrement la méthode *KanBan*.

La méthode Kanban, est une méthode de gestion de projet où chaque tâche est placée sur une carte et chaque carte doit être placée dans un tableau. Chaque carte peut être assignée à un ou plusieurs utilisateurs et on peut lui attribuer une date d’échéance ainsi qu’une ou des étiquettes permettant de catégoriser la tâche associée. L’avancement d’une tâche est représenté par le déplacement d’une carte d’un tableau à un autre. En ce qui concerne mon organisation personnelle avec Trello, j’ai utilisé 5 tableaux. Une carte passe successivement par le tableau "À faire", "En cours" puis "Fait", mais peut aussi passer par le tableau intermédiaire "En attente" lorsqu’elle nécessite l’achèvement d’une tâche par un autre membre du projet, extérieur à l’équipe. À l’exception des tâches

¹Jira Ops et Trello sont d’ailleurs édités par la même entreprise : Atlassian

portant l'étiquette "Récurrence" (étiquette rouge dans l'exemple ci-dessous) qui une fois achevées, retournent dans le tableau portant la mention "Récurrence".

Voici ci-dessous, l'interface de Trello.



Concernant le processus que j'ai mis en place, que ce soit pour son initialisation, les différentes modifications apportées à certaines étapes ou l'ajout de certaines étapes, le développement se faisait en 4 étapes :

- *Formalisation* : tout d'abord, il fallait analyser le besoin, les ressources dont l'équipe disposait puis chercher une solution optimale et la rendre intelligible afin de pouvoir la communiquer (par le biais d'un diagramme par exemple).
- *Validation interne* : ensuite il me fallait exposer ma proposition à mes managers, c'est à dire au directeur adjoint, au manager T&C, au manager technique et au responsable RAMS. Cette étape n'était pas qu'une simple obligation impliquée par le respect de la hiérarchie, elle s'est avérée cruciale pour deux raisons. Tout d'abord, elle permettait de s'assurer de la conformité du processus au regard des besoins de l'équipe. En effet, ces managers possédaient une connaissance poussée des systèmes déployés, mais aussi du projet ainsi que du fonctionnement des différentes entités de ce dernier. Ensuite elle était indispensable du point de vue de l'attribution des tâches, afin de ne pas surcharger un employé ou une équipe, ce qui dans un projet de cette envergure, se serait avéré lourd de conséquences. En fonction des remarques et conseils de mes supérieurs, j'effectuais les modifications nécessaires.
- *Validation externe* : rendu à ce stade, il fallait ensuite éprouver la solution faite au client. Non seulement parce qu'il fallait obtenir son accord, mais aussi parce qu'il serait lui aussi concerné par les conséquences de cette décision, étant donné que QDVC et ICE possédaient leurs propres équipes intégrées au processus de T&C. J'exposerai plus loin les difficultés rencontrées à cette étape.
- *Application du processus aux équipes T&C* : enfin, après avoir obtenu les deux validations nécessaires, il fallait ensuite mettre en place le processus, propager les

nouvelles pratiques aux membres des différentes équipes T&C puis dans un second temps, il me fallait faire remonter aux managers les premiers résultats impliqués par ce changement (nous verrons plus loin les méthodes utilisées).

En ce qui concerne l'outil d'automatisation de l'extraction des données que j'ai développé, le processus était identique à l'exception de la validation externe, puisque cette solution était à usage purement interne.

2.5 Prise de recul

L'intérêt de mon travail pour l'équipe était de pouvoir obtenir un suivi détaillé de la production des preuves nécessaires à l'ouverture du projet ainsi que de pouvoir s'assurer du référencement de ces preuves dans le référentiel projet (dans la base de données Mezzoteam). Après mon départ, mon remplaçant reprendra la supervision de ces différents processus en utilisant les différents outils que j'ai mis en place.

Certaines parties des processus que j'ai mis en place pourraient encore être automatisées afin de minimiser la charge de travail répartie entre les différents acteurs impliqués dans ce processus. Il y a certains problèmes techniques que je n'ai pas eu le temps de résoudre, mais nous reviendrons sur ce sujet en détail dans la prochaine partie. Concernant la maintenance éventuelle à prévoir sur l'application que j'ai réalisée, elle serait nécessaire dans l'éventualité d'un changement de structure de la base de données de SnagR. Alors il faudrait analyser les changements apportés, déterminer s'ils impactent le bon fonctionnement de l'application, auquel cas des modifications mineures seront nécessaires.

Selon moi, je pense avoir amené certains utilisateurs, acteurs des différents processus, à réfléchir quant à la cohérence des données en les sensibilisant sur l'importance de la validation des données lorsqu'on utilise une base de données par exemple.

3. Réalisations

Le sujet de mon stage étant large, très tôt j’ai dû me concentrer sur plusieurs tâches puis rapidement m’atteler à plusieurs réalisations afin d’assurer au mieux ma fonction. Nous verrons donc dans ce chapitre, pour chacune de ces différentes réalisations, les besoins auxquels il fallait répondre, les choix qui ont été faits et les différentes difficultés rencontrées.

3.1 Conception d’un processus de production et de gestion des rapports de test

3.1.1 État des lieux

Dans un premier temps, une présentation du contexte initial est indispensable. Au sein de l’équipe T&C, il existe différentes phases de test, dont l’organisation correspond au cycle en V du projet (cf. Annexe I) :

- **FAT** : Factory Acceptance Test, test d’acceptation usine consistant à inspecter le système dans l’usine avant livraison. Pendant le FAT, les performances du système sont contrôlées, généralement en collaboration avec le client.
- **iFAT** : Integrated Factory Acceptance Test, test d’acceptation intégré usine consistant à contrôler le fonctionnement des différents équipements ensemble.
- **StAT** : Stand Alone Test, test unitaire, où l’équipement testé est isolé de toute interaction durant le test, afin de restreindre les critères du test à son fonctionnement autonome.
- **SIT** : Site Integration Test, test d’intégration, ayant pour but de détecter les erreurs non détectables pendant les tests unitaires. Ce type de test consiste à assembler différents composants afin de tester leur fonctionnement dans l’ensemble.
- **E2E** : End to End Test, test bout en bout ayant pour but de vérifier si un système se comporte comme prévu du début à la fin. Le testeur doit se mettre dans le rôle d’un utilisateur et effectuer les tests comme s’il utilisait véritablement l’outil mis à sa disposition.

Nous nous concentrerons ici sur les phases StAT, SIT, E2E.

Pour chaque sous-système du projet, 3 procédures de test ont été rédigées, une par phase de test.

Ainsi, si une localisation du projet abrite 3 sous-systèmes nécessitant d'être testés, il y aura 9 tests à effectuer dans cette localisation.

Ensuite, chaque test doit être formalisé par un rapport de test constitué de la manière suivante :

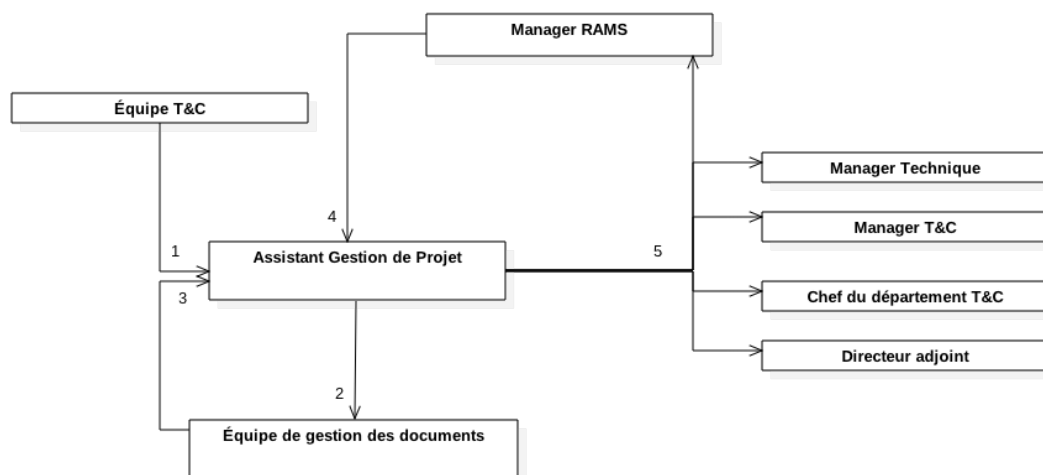
- Une première page que l'on appelle ITR, qui permet d'identifier le rapport (numéro unique), d'identifier le test (localisation, phase, système, date), identifier le statut (accepté avec, sans réserve ou refusé), résumer les observations et réserves des 3 parties et consigner leurs signatures.
- Les pages suivantes sont les Test Cases, en français les scénarios de test, qui permettent de simuler plusieurs situations d'utilisation des systèmes de manière à prouver leur conformité vis-à-vis de toutes les exigences du cahier des charges.
 - Chaque Test Cases est composé de une ou plusieurs étapes qui elles-mêmes sont reliées à une ou plusieurs exigences.

Parmi ces exigences, certaines sont liées à la sécurité, dans le cadre des RAMS, et sont indexées dans un document appelé le Safety Case. Ces exigences liées à la sécurité sont séparées en deux SIL, ou niveaux de criticité : SIL0, le plus faible et SIL2 le plus élevé.

Une fois ce contexte clarifié, nous avons pu entreprendre l'élaboration du processus et c'est le sujet de la prochaine section.

3.1.2 Première version

Après l'analyse exposée précédemment, nous avons travaillé conjointement avec mon maître de stage sur une première version du processus que voici :



- 1. Tout d’abord, l’équipe T&C effectue les différentes inspections et produit les rapports de test en remplissant les formulaires correspondant sur SnagR.
- 2. Ensuite j’extraie les données de SnagR (nous reviendrons sur ce point dans la prochaine partie) et les importe dans un fichier Excel afin de les exploiter. J’exporte les rapports au format PDF et les envoie à l’équipe de gestion des documents.
- 3. Cette dernière importe les rapports sur Mezzoteam afin de rendre ces preuves accessibles à tous les membres du projet et les importe aussi sur e-TOL pour que les membres de l’équipe Thales offshore puissent les consulter. Une fois l’importation terminée, l’équipe me fournit les références de ces documents afin que je les ajoute à ma base de données.
- 4. Le manager RAMS et son équipe me fournissent des mises à jour concernant les exigences liées à la sécurité afin que je puisse actualiser mon fichier, car les rapports devant servir de preuve pour ces dernières nécessitent d’être suivis de manière prioritaire. En effet, le futur opérateur refusera de commencer la période de formation de ses employés à l’exploitation du système tant qu’il n’aura pas la preuve de la conformité du projet vis-à-vis de ces exigences.
- 5. Enfin, il me faut communiquer les résultats aux différents managers sous forme de KPI, courbes et outils d’aide à la décision. Nous reviendrons plus tard sur ce point.

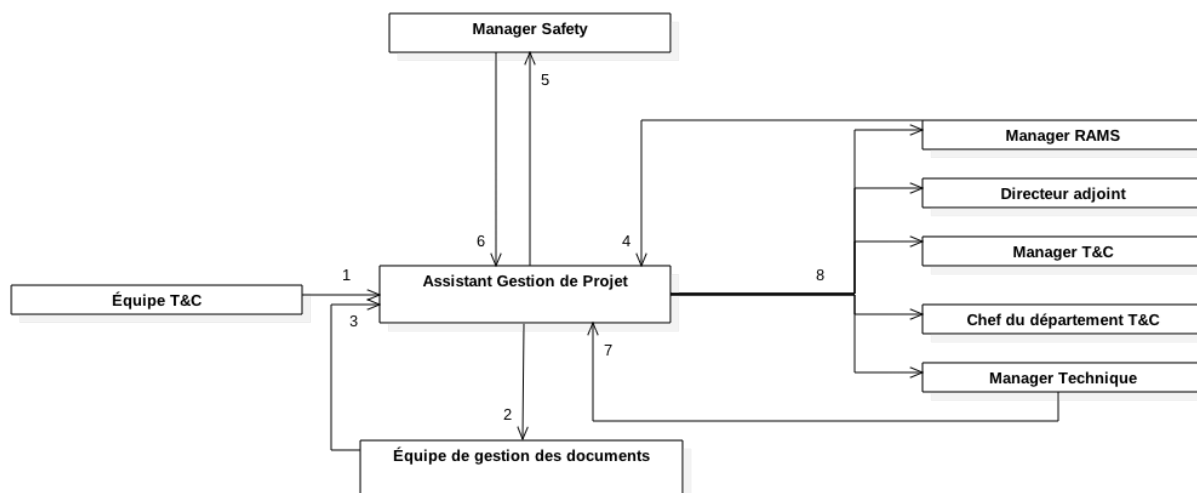
Ce processus fut mis en place et appliqué durant le premier mois de mon stage, mais il fallut rapidement l’adapter.

3.1.3 Évolution du processus

En effet, aux vues des différentes responsabilités que l’on m’a confiées à partir du second mois, j’ai dû intégrer à ce processus de nouvelles étapes. Voici ci-dessous la matrice de responsabilité de notre projet suivant la méthode RACI. Dans cette matrice, le R signifie responsable et désigne la personne qui exécute l’action, le A désigne la personne qui approuve l’action, le C une personne consultée et le I une personne devant être tenue informée.

Activités	Manager Safety	Manager T&C	Manager Technique	Work Package Manager	Qualité	Gestion Projet T&C
Management	R	R	R	A	I	C
Priorisation des tests	I	A/R	C	C	I	C
Exécution des tests	I	A/R	C	I	I	C
Livraison des rapports	C	R	C	I	A	R
Suivi des rapports	I	A	I	I	I	R
PCR Generation	R	R	A	I	C	C
Revue des exigences sécurité	R	R	R	A	R	C
Contrôle des risques	C	C	C	A/R	I	C
Création des KPI	C	C	C	A	C	R

Ainsi, c'est pour cela que j'ai dû choisir d'intégrer au processus de nouveaux acteurs ainsi que de nouvelles interactions.



- **Étapes 1 à 4.** Identiques à la première version du processus.
- **5.** Après avoir vérifié que les exigences liées à la sécurité étaient bien à jour, il me fallait fournir au manager sécurité les données nécessaires à la vérification de la conformité des systèmes. Je devais donc lui fournir, pour chaque exigence déjà vérifiée lors d'un test, la référence du rapport de ce test ainsi que celles des Test Cases permettant de vérifier la conformité de l'exigence.
- **6.** Ensuite, le manager sécurité devait spécifier, pour chaque Test Cases lié à une exigence sécurité, si il était conforme ou non, et le cas échéant, il devait générer une PCR afin de notifier ce défaut aux autres équipes.
- **7.** Après cela, je collectais, auprès du manager technique (et de son équipe), les différents facteurs limitants retardant l'exécution de certains tests, afin de pouvoir, à l'étape suivante, réaliser des outils d'aide à la décision.
- **8.** Comme précédemment, il fallait enfin communiquer les résultats aux managers.

Dans un souci de synthèse, la version du processus final exposée ci-dessus est une version simplifiée, mais les étapes principales y sont représentées. Au cours de la prochaine partie, nous allons nous intéresser à la réalisation de la deuxième étape du processus : l'extraction des données.

3.2 Mise au point d'un outil d'extraction de données

Cette partie constitue le point névralgique du processus, car sans données, aucune des étapes suivantes n'est réalisable. Mais c'est aussi le point le plus complexe et c'est pour cela que cette réalisation fut la plus conséquente de mon stage.

3.2.1 Les motivations

Au tout début de mon stage, je me chargeais de l'extraction de ces données manuellement, puisque le nombre de rapports de test remplis était réduit.

Mais rapidement il fallut envisager une alternative, car sur cette phase du projet, environ 1500 rapports de test seraient produits, soit plus de 15 000 Test Cases (donc autant de formulaires), soit plus de 60 000 étapes de test à analyser.

De plus, le contenu de ces rapports de test étant dynamique (puisque les utilisateurs de SnagR pouvaient éditer le même rapport à maintes reprises), il aurait fallu actualiser cette énorme quantité de données régulièrement.

Ainsi, j'ai essayé, en demandant au client des droits d'accès plus élevés sur SnagR, d'obtenir les données automatiquement en utilisant les fonctions proposées par l'outil.

Mais ces fonctions d'exportation de données présentaient deux défauts majeurs :

- Les champs et les différents lots de données proposés au moment de l'exportation étaient trop restreints, il n'était pas possible d'exporter toutes les données issues des rapports.
- Les données étaient structurées autour d'un découpage basé sur des activités, pour faire simple, ce dernier ne correspondait pas au découpage énoncé précédemment : 1 rapport par localisation pour chaque type de test et pour chaque système. Les données manquaient de cohérence, le nombre total d'activités étant de 6000 alors que seules 1500 de ces activités seraient liées à un rapport. C'est pour cela que les KPI proposées par l'outil SnagR étaient fausses, puisqu'elles se basaient sur un nombre total 4 fois plus élevé que le vrai.

Le problème étant que le client, qui avait choisi cet outil, devait envoyer des tickets techniques au service client SnagR pour chaque incident.

Après en avoir discuté avec mon maître de stage, il a accepté que je me charge d'implémenter une solution permettant d'extraire les données des formulaires présents sur SnagR puis de les réorganiser dans une autre base de données.

Pour cette réalisation, il fallait donc concevoir plusieurs fonctionnalités, c'est le sujet de prochaine section.

3.2.2 La structure de l'outil

Dans cette section, nous verrons les différentes étapes nécessaires au bon fonctionnement de l'outil.

Authentification

Il fallait dans un premier temps fournir au script le moyen de se connecter au site de SnagR pour pouvoir avoir accès aux données.

Extraction des données

Ensuite, l'extraction des données se faisait deux étapes :

- Premièrement seules les données des ITR (première page des rapports) sont extraites, de manière à obtenir la liste de tous les rapports présente dans la base de données, avec leurs informations principales (identifiant, système, type de test, localisation, date du test, statut général, commentaire général et signatures des différents partis).
- Dans un second temps, pour chaque rapport, les données de tous les Test Cases dont il est constitué sont extraites afin d'accéder à un second niveau de détails.

L'utilisateur pouvoir choisir au début de l'exécution du script quel niveau de détails il souhaitait.

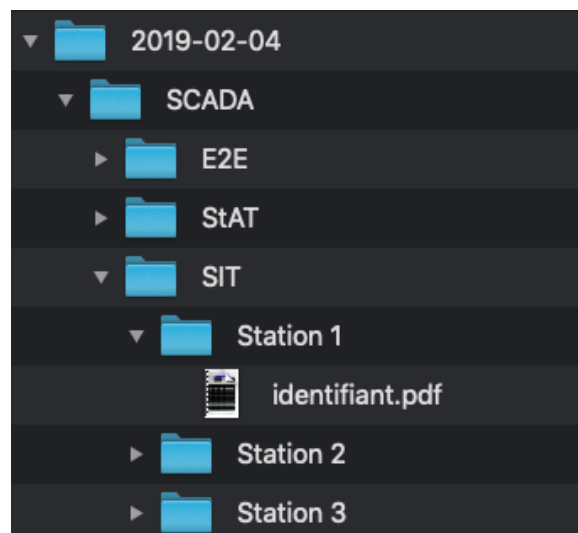
Organisation des données

En parallèle de l'extraction des données, il fallait les organiser au fur et à mesure dans des tables dans le but de les exploiter.

Sauvegarde

À la fin des deux étapes précédentes, le script propose à l'utilisateur d'effectuer une sauvegarde complète de tous les rapports de test dont la complétion avait été entamée.

Si l'utilisateur choisissait d'effectuer une sauvegarde, l'ensemble de ces rapports étaient téléchargés au format PDF avec leur identifiant inscrit dans leur nom de fichier, tout cela organisé dans une arborescence respectant le schéma suivant :



Exportation

Enfin, les données sous forme de tables étaient exportées au format Excel afin d'être intégrées aux différents outils du projet.

3.2.3 Les choix techniques

Dans cette section, pour chaque étape énoncée précédemment, nous exposerons ainsi les éventuelles difficultés rencontrées pour chacune d'elles, les différentes alternatives techniques possibles et nous justifierons nos choix.

Concernant le langage informatique, j'aurais pu utiliser :

- **Node.js** : langage efficace pour extraire des données d'un site web dynamique, mais pas assez stable pour le nombre de requêtes nécessaires à ce projet.
- **C++** : il propose des performances élevées, mais aurait demandé un investissement horaire démesuré.
- **PHP** : ce langage supporte mal le multi-threading et l'ordonnancement de tâches, fonctions pourtant indispensables à ce projet, comme nous le verrons plus bas.
- **Python** : c'est l'alternative la plus utilisée pour ce genre de projet, et cela principalement grâce aux bibliothèques surdéveloppées que la communauté Python a mises au point et enrichies.

J'ai donc choisi d'utiliser Python. Dans un premier temps j'utilisais Python 2.7, mais la fin de vie de cette version étant annoncée pour janvier 2020 j'ai préféré passer, vers la fin du projet, à Python 3.7. Ce choix était aussi motivé par la simplicité de cette opération, grâce à l'existence d'un programme nommé 2to3 qui permet de transformer un script écrit en 2.7 en un script compatible 3.7.

Authentification

Pour cette partie, j'ai défini une session, avec laquelle j'envoyais une requête de type "post" avec les données nécessaires à l'authentification. Cette requête permettait de simuler la saisie manuelle des données dans les champs dédiés de la page d'authentification puis de simuler l'appui sur le bouton "Se connecter". Cependant, une autre solution bien plus simple aurait été d'utiliser la clé d'accès à l'API de SnagR à chaque requête.

En effet, l'outil propose une API, qui est une interface donnant accès à certaines de ses fonctions et de ses données, et cela de manière facilitée.

Malheureusement, je n'ai découvert l'existence de cette API qu'après avoir réussi à implémenter la méthode d'authentification.

Extraction des données

Pour l'extraction des données, j'ai commencé par analyser les différentes requêtes effectuées par SnagR et c'est à ce moment-là que j'ai découvert la présence d'une API (à travers l'URL d'une requête).

Afin de pouvoir l'exploiter au mieux, j'ai voulu me procurer sa documentation. En explorant l'arborescence du site, j'ai découvert une page contenant une documentation non achevée et obsolète puisqu'elle concernait la version 1 de l'API et non la version actuelle (v2).

J'ai donc contacté le service informatique du client (QDVC) puis le service de SnagR, mais une telle documentation pour la version 2 ne semblait pas exister.

Il m'a donc fallu comprendre la structure et le fonctionnement de l'API par l'expérimentation. J'ai analysé de nombreuses pages du site jusqu'à obtenir un lot de requêtes suffisant à l'extraction des données nécessaires.

Les réponses à ces requêtes étaient au format JSON, un format de représentation des données.

J'ai donc choisi de générer ces requêtes au sein du script et de récupérer les données résultantes avec un module Python nommé simplejson.

Le choix de l'extraction des données en deux temps est justifié par deux raisons :

- Laisser le choix à l'utilisateur des données qu'il veut extraire afin d'actualiser sa base de données.
- Optimiser le nombre de requêtes exécutées par le script. En effet, une fois la liste des rapports obtenue, au lieu de chercher à extraire les données de tous les rapports, même ceux encore vierges, on ne s'intéresse qu'à ceux qui sont au minimum entamés. Ici nous avons pris comme critère la présence de la signature de l'ingénieur Thales ayant réalisé le test.

Toujours dans une démarche d'optimisation de cette étape, j'ai ensuite cherché à réduire le temps d'extraction des données. En effet, il fallait une requête pour pouvoir

obtenir les données de chaque Test Cases, ce qui faisait plus de 15 000 requêtes, plus le temps de traitement de chaque requête.

C'est pour ces raisons qu'initialement, le programme prenait environ une heure à s'exécuter.

C'est ce qui m'a amené à devoir choisir entre le multithreading et le multiprocessing. Ce sont deux techniques d'optimisation dont le principe diffère :

- Le premier consiste à exécuter des tâches ou fils de tâches (appartenant au même processus) en parallèle tout en utilisant les mêmes ressources (même processeur).
- Le second consiste à exécuter plusieurs processus en parallèle en utilisant plus de deux processeurs.

J'ai choisi le multithreading, car je ne voulais pas concevoir un outil dont les performances dépendraient grandement de la machine qui l'exécute.

À cela j'ai ajouté un système de queue, pour stocker les requêtes en attente d'être exécutées.

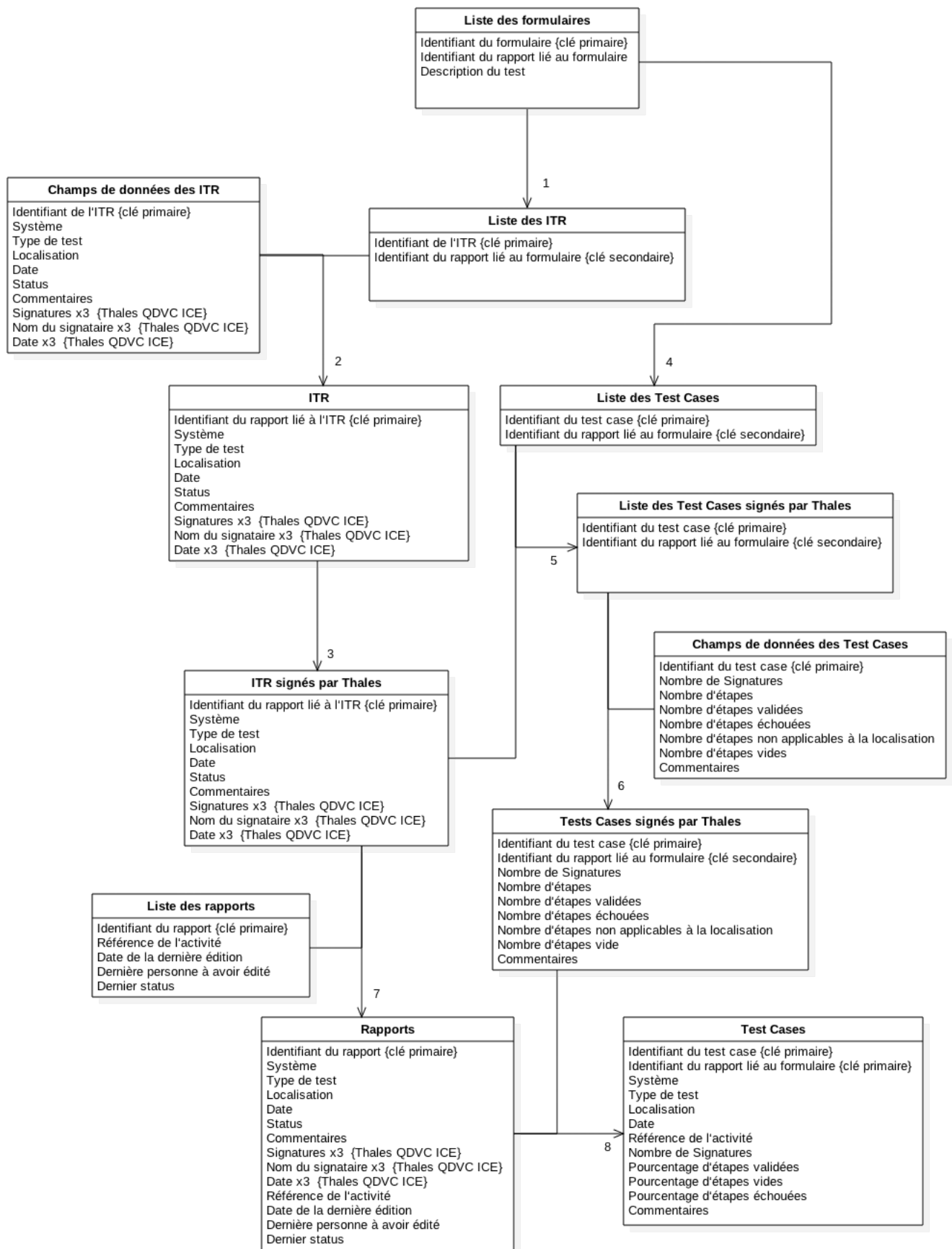
Après avoir mis en place cette solution, le temps d'exécution oscillait entre 5 et 10 minutes, car les serveurs du site n'offraient pas une stabilité optimale, ce qui faisait que par moment, plus de 40% des requêtes échouaient et devaient être effectuées à nouveau.

Organisation des données

Pour organiser les données, j'ai utilisé la bibliothèque Pandas qui est dédiée à l'analyse de données.

La première étape était d'importer les données au format JSON dans des tables. Pour cela j'ai utilisé le module "jsonnormalize" issu de la bibliothèque Pandas.

Cette dernière m'a aussi servie à sélectionner les champs de données que je voulais conserver, filtrer les résultats, joindre des tables et toutes les opérations habituelles lorsqu'on traite avec des tables de données. Voici ci-dessous les différentes tables que j'ai créées afin de concevoir ma base de données finale :



Pour plus de détails, voici les différentes opérations réalisées :

- 1 : on filtre la liste des formulaires en ne retenant que les ITR (première page d'un rapport).
- 2 : on effectue une jointure des deux tables avec l'identifiant du de l'ITR puis on supprime cet identifiant au profit de l'identifiant du rapport lié à l'ITR qui est aussi unique dont éligible en tant que clé primaire.
- 3 : on filtre en ne retenant que les ITR signés par Thales.
- 4 : on filtre en ne retenant que les Tests Cases.
- 5 : on effectue une jointure des deux tables avec l'identifiant du rapport lié à l'ITR.
- 6 : on effectue une jointure des deux tables avec l'identifiant du test case.
- 7 : on effectue une jointure des deux tables avec l'identifiant du rapport lié à l'ITR.
- 8 : on effectue une jointure des deux tables avec l'identifiant du rapport lié à l'ITR.

Sauvegarde

Pour cette fonction, j'ai fait le choix de ne télécharger que les rapports au minimum initialisés par Thales. Chaque sauvegarde est stockée dans un dossier ayant pour nom la date de l'exécution en suivant l'arborescence précédemment énoncée.

J'aurais pus faire le choix de ne créer qu'un seul dossier et de ne télécharger que les rapports qui n'avaient pas été encore sauvegardés, mais le contenu de ces derniers étant dynamique, la seule solution viable était une sauvegarde intégrale et datée à chaque appel de la fonction de sauvegarde.

Exportation

Enfin, pour l'exportation, je me suis posé la question du format d'exportation de ces données que je devais choisir.

La solution optimale, selon moi, aurait été d'héberger ce script sur un serveur, l'exécuter en tâche de fond puis créer une interface web pour la consultation et la communication de ces données.

Cependant, cette solution ne correspond pas aux contraintes du projet.

Premièrement, je ne pouvais pas allouer la totalité de mon temps à cette tâche.

Ensuite, le caractère confidentiel de ces données ne me permettait pas de les héberger sur un serveur personnel. Les héberger sur un serveur Thales aurait de toute façon nécessité une validation en interne puis avec le client ainsi qu'un processus assez long (dans le cas où la proposition aurait été acceptée).

3.3 Communication sur l'avancement du projet

Une fois les données extraites, il faut leur donner un sens afin de pouvoir les exploiter. Sur ce projet, les données issues des rapports étaient indispensables au management ainsi qu'à la communication avec le client. La première étape de cette exploitation est l'analyse des données.

3.3.1 Analyse des données

Si on revient aux raisons qui nous ont poussés à développer l'outil présenté précédemment, une d'entre elles était le manque de cohérence. En effet, le nombre total de rapports sur SnagR étant plus élevé que la réalité, il est, jusque là, impossible de rendre compte de l'avancement, puisque ce dernier s'illustre avec des pourcentages d'accomplissement.

C'est pour cela qu'un travail d'analyse des différentes documentations des systèmes déployés pas Thales a été réalisé avec mon maître de stage ainsi que le manager de l'équipe afin d'obtenir un nombre total de rapports à produire.

Création de différents indicateurs et outils permettant la vérification de la cohérence des données

3.3.2 Communication interne

Liste (exhaustive ou non) des différents graphiques, tableaux et autres éléments présents dans les slides que j'envoyai chaque semaine. Partie expliquant la synchronisation avec le testing (fichier d'Armenio) pour plus de cohérence entre les différents fichiers communiqués.

3.3.3 Communication avec le client

Organisation des données et des indicateurs pour correspondre aux différents scopes proposés par l'équipe RAMS (Pilot, Safety Strategy et Safety Related)

Difficultés personnelles rencontrées face au client (dues à une première expérience)

3.4 Formation et transmission

3.4.1 Sensibilisation des membres de l'équipe

Sensibilisation des utilisateurs SnagR à l'importance de la production des rapports (je vais essayer de tourner ça autrement pour ne pas commettre d'impair, mais l'idée est là)

Création d'une norme de saisie des données et formation (guidelines SnagR, convention de dommage).

Extrait de la norme proposée

3.5 Tuilage

Séance de formation du remplaçant pendant le dernier mois

Délégation du travail progressive

Tutoriel vidéo réalisé pour mon remplaçant.

Documentation fournie.

État de l'avancement (pending tasks, pending issues).

Conclusion

Ici j'exposerais :

- un résumé du rapport
- une liste de travaux que j'aurai pus réaliser si j'étais resté plus longtemps (par exemple proposer de mettre l'outil en production en local sur le réseau Thales)
- Les aspects positifs (expérience, communication, etc) et ceux négatifs (gestion de mes tâches pas optimales, mauvaise gestion de mon temps ce qui me faisait travailler le week-end ou tard le soir) de mon stage. Je vais reprendre ce que tu (Vincent) avais mis dans ma fiche parce que ça résume bien les difficultés que j'ai eues.
- Expliquer en quoi ce stage s'est inscrit dans mon parcours : réutilisation des notions et pratiques étudiées en cours et projet (base de donnée, optimisation, gestion de projet, etc..)
- En quoi ce stage a fait évoluer mon projet professionnel donc par exemple : il a renforcé ma motivation à travailler dans l'informatique des réseaux, celle de m'expatrier (volonté d'effectuer un VIE), ma motivation à travailler pour Thales.

Glossaire

ACS-IDS *Access Control System-Intrusion Detection System*, Système de Contrôle des Accès ainsi que de Détection des Intrusions. 1, 7

AFC *Automatic Fare Collection*, Système de Collection Automatique des Billets. 1, 8

API *Application Programming Interface*, Interface de Programmation Applicative, permettant à un logiciel d'offrir un accès facilité à certaines de ses fonctions, méthodes et classes. 1, 21

CCS *Communication and Control System*, Système de Contrôle et de Communication, département du projet concernant principalement les télécommunications, la sécurité et le monitoring. 1, 6–8, 11

CCTV *Closed-Circuit Television*, Système de Vidéosurveillance. 1, 7

COMTV *Commercial Television*, Télévision commerciale, désigne l'ensemble des écrans situés en station et à bord des rames ayant pour fonction de diffuser des annonces publicitaires aux usagers. 1, 7

DTS *Digital Transmission System*, Système de Transmission Digitale, ensemble des différentes infrastructures réseau. 1, 7

E2E *End to End Test*, test bout en bout ayant pour but de vérifier si un système se comporte comme prévu du début à la fin. Le testeur doit se mettre dans le rôle d'un utilisateur et effectuer les tests comme s'il utilisait véritablement l'outil mis à sa disposition (c.f TestingDigital). 1, 15, 16

ECS *Environmental Control System*, Système de Contrôle Environnemental. 1, 7

FDS *Fire Detection System*, Système de Détection des Incendies. 1, 7

Git logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2. (c.f Wikipédia). 1, 9, 12

IDE *Integrated Development Environment* en français : Environnement de Développement Intégré (EDI) . 1, 12

- KPI** *Key Performance Indicator*, Indicateur clé de performance, indicateur graphique et/ou numérique permettant d'évaluer l'avancement d'un projet, de communiquer, de diagnostiquer les points bloquants ou encore de s'assurer de la continuité du progrès. 1, 11, 17, 19
- LRT** *Light Rail Transit*, forme de transport en commun urbain ferroviaire disposant généralement d'une capacité et d'une vitesse inférieures à celles d'un train ou d'un métro, mais supérieures à celles des systèmes traditionnels de tramway. (c.f Wikipédia). 1, 3, 4, 9
- LaTeX** LaTeX est un langage de description donnant à l'auteur les moyens d'obtenir des documents mis en page de façon professionnelle sans avoir à se soucier de leur forme. La priorité est donnée à l'essentiel : le contenu (c.f OpenClassrooms). 1, 12
- MMS** *Maintenance Management System*, Système de Management de la Maintenance. 1, 8
- Mezzoteam** Base de données regroupant l'ensemble des documents produits par les différents acteurs du projet. 1, 9, 11, 14, 17
- PAS** *Public Address System*, désigne un système d'amplification et de distribution sonore électronique par le biais d'un microphone, amplificateur et de haut-parleurs, permettant à une personne de communiquer un message (pré-enregistré ou en direct) au grand public (c.f Wikipédia). 1, 7
- PCR** *Product Change Request*, requête formulée par ingénieur, souvent par le biais d'un outil de gestion, afin de demander des changements dans le design d'un système. 1, 12, 18
- PIS** *Passenger Information System*, Système d'Information des Passagers automatisé permettant de leur fournir à la fois des informations statiques comme des tables horaires ainsi que des informations dynamiques comme l'attente avant la prochaine rame ou encore les incidents survenus sur le réseau (c.f Wikipédia). 1, 7
- PSD** *Platform Screen Door*, portes palières ou encore façades de quai sont des portes automatiques vitrées situées le long des quais en bordure des voies ne s'ouvrant que lorsque la rame est à l'arrêt en station (c.f Wikipédia). 1, 6
- RAMS** *Reliability Availability Maintainability and Safety*, en français : FMDS (Fiabilité Maintenabilité Disponibilité et Sécurité), entité en charge de la sûreté de fonctionnement au sein d'un projet. 1, 11, 13, 16, 17
- RST** *Rolling Stock*, ensemble du matériel roulant, ici c'est l'ensemble des différentes rames du LRT. 1, 6
- SCADA** *Supervisory Control And Data Acquisition*, Système de Contrôle et d'Acquisition de Données, système de télégestion à grande échelle permettant de traiter en temps réel un grand nombre de télémesures, d'informations visuelles (caméras par exemple), d'alarmes, de contrôler à distance des installations techniques. 1, 8

- SIL** *Safety Integrity Level*, niveau d'intégrité de sécurité, niveau relatif de réduction de risques inhérents à une fonction de sécurité, ou comme spécification d'une cible de réduction de risque. Plus simplement, c'est une mesure de la performance attendue pour une fonction de sécurité. (c.f [Wikipédia](#)). 1, 16
- SIT** *Site Integration Test*, test d'intégration, ayant pour but de détecter les erreurs non détectables pendant les test unitaires (c.f StAT). Ce type de test consiste à assembler différents composants afin de tester leur fonctionnement dans l'ensemble. 1, 15, 16
- Safety Case** document indexant l'ensemble de exigences liées à la sécurité. 1, 16
- SnagR** Solution en ligne permettant l'édition et la gestion des rapports de test ainsi que des différents défauts présents sur les systèmes du projet . 1, 9–12, 14, 17, 19, 21
- StAT** *Stand Alone Test*, test unitaire, où l'équipement testé est isolé de toute interaction durant le test, afin de restreindre les critères du test à son fonctionnement autonome . 1, 15, 16
- TCS** *Tramway Control System*, Système de Contrôle du Tramway. 1, 6
- TETRA** *Terrestrial Trunked Radio*, système de radio numérique mobile professionnel bi-directionnel, spécialement conçu pour des services officiels et pour l'armée. Un réseau de type TETRA offre un canal radio partagé ouvert en permanence, et réservé à un groupe d'utilisateurs. Ceci permet d'établir une communication immédiate entre un utilisateur sur le terrain et un dispatcher, ou un groupe d'utilisateurs (c.f [Wikipédia](#)). 1, 7
- TVS** *Tunnel Ventilation System*, Système de Ventilation des Tunnels . 1, 6
- T&C** *Testing & Commissioning*, département d'un projet dédié à l'inspection et à la mise en service des divers équipements et systèmes du projet. 1, 8–15, 17
- Test Cases** en français : scénarios de test, éléments composant les test, définis au moment du design afin de simuler plusieurs situations d'utilisation des systèmes de manière à prouver leur conformité vis à vis de toutes les exigences du cahier des charges. 1, 16, 18, 19
- UPS** *Uninterruptible Power Supply*, L'Alimentation Sans Interruption (ASI), ou encore un onduleur, est un dispositif de l'électronique de puissance qui permet de fournir un courant alternatif stable et dépourvu de coupures ou de microcoupures, quoi qu'il se produise sur le réseau électrique (c.f [Wikipédia](#)). 1, 7
- WA** *Wifi Access*, Système permettant de proposer aux usager du LRT de bénéficier d'un accès à Internet via un un réseau WiFi. 1, 7
- e-TOL** Plateforme de gestion de projet et de partage de fichiers interne à Thales, permettant par exemple de partager des fichiers entre l'équipe basée en France (offshore) et l'équipe basée à Lusail (inshore). 1, 9, 11, 17

méthodes agiles Les méthodes agiles sont des groupes de pratiques de pilotage et de réalisation de projets. Elles ont pour origine le manifeste Agile, rédigé en 2001, qui consacre le terme agile pour référencer de multiples méthodes existantes. (c.f Wikipédia). 1, 9, 12

reporting Ensemble des processus permettant d'aboutir à la présentation des activités du projet et de leurs avancements. Ici ce terme est utilisé pour désigner la gestion de la production des rapports de test. 1, 10, 11

scraping Le web scraping (parfois appelé harvesting) est une technique d'extraction du contenu de sites Web, via un script ou un programme, dans le but de le transformer pour permettre son utilisation dans un autre contexte. (c.f LuxuryConcept). 1

Bibliographie

- [1] django. *Django Documentation*. URL: <https://docs.djangoproject.com/fr/2.0/>.
- [2] git-scm. *Git Basics Tagging*. URL: <https://git-scm.com/book/en/v2/Git-Basics-Tagging>.
- [3] T. R. Gruber. “A Translation Approach to Portable Ontology Specifications”. In: *Computer Science Department, Stanford University* (1993).

Annexes

Annexe I - Cycle de développement (en V)

