

# Testing : Etude de cas COMURLUX

*Alexis BURUCHIAN, Myeongio JUNG, Anchalee LALITURAI,  
Florian LERIDEAU et Li SHAOWEI*

---

## Introduction

Nous sommes l'équipe chargée de l'audit du SI de COMORLUX, et plus précisément de tout ce qui concerne le testing de l'application.

Nous allons étudier la situation actuelle, de tous les points de vue offerts (hiérarchique, organisationnel, technique, etc.) pour en retirer une manière d'améliorer ce qui est existant. Notre objectif personnel est de garantir un service dans un temps de crise, c'est-à-dire en se fixant l'objectif d'offrir les solutions les plus abordables en termes de services et d'outils mis à disposition, en gardant à l'esprit la notion de qualité de service inhérente à notre société.

Pour cela, nous allons revoir toute l'organisation actuelle et repasser au crible les risques actuels, les classer, offrir des axes d'améliorations puis des solutions techniques ainsi que des outils.

Nous ferons évidemment des choix qui seront tous dictés par l'idée directrice d'un budget resserré et d'une exigence d'amélioration à moyen-terme.

Pour bien illustrer la problématique, commençons par souligner les faiblesses actuelles de la situation de COMORLUX vis-à-vis de son application et de son testing.

## Question 1

Le projet nécessite avant tout début de solution un listing un tant soit peu complet de la situation actuelle, pour être sûr de ce que l'on cherche à améliorer.

Pour cela, le but va être d'identifier les risques de l'état actuel du système.

On peut identifier des risques autant sur le développement des correctifs de l'application, l'organisation des équipes que sur les outils nécessaires au testing.

Pour cela, nous allons répartir les risques identifiés en 3 catégories :

- **Risques majeurs** : Les risques qui peuvent affecter de façon immédiate et grave le système. C'est ceux qui doivent être traités en priorité et pour lesquels les investissements en terme de moyens financiers et moyens de personnel peuvent être les plus importants.

- **Risques importants** : Les risques qui, traités individuellement, ne peuvent affecter de façon irréversible le système, mais qui, s'ils s'accumulent, peuvent arrêter le système de façon grave. Il faut donc les considérer de façon sérieuse, notamment en termes de moyens, cependant ils n'ont pas le même critère de priorité que les risques majeurs.

- **Risques mineurs** : Les risques qui ne peuvent pas compromettre le système mais qui peuvent le ralentir considérablement si accumulés. Cependant, les identifier et mettre des

plans en place pour les limiter seraient un bon moyen de s'assurer que nous sommes en bonne position pour améliorer les problèmes actuels.

Nous allons ajouter également des "catégories" aux différents problèmes soulevés, pour regrouper encore plus certains types de problèmes. Les catégories choisies sont :

- *Sécurité* : Comment le système est-il protégé ?
- *Continuité* : Facilité pour retrouver des informations et les transmettre.
- *Connectivité* : Les outils choisis.
- *Contrôle de données* : Peut-on vérifier l'intégrité des données ?
- *Flexibilité* : Facilité d'adaptation des outils en place.
- *Réusabilité/Expérience* : Peut-on bénéficier des résultats antérieurs pour faire plus vite.
- *Timing* : Le processus est-il bien évalué en termes de durée ?

Etudions le listing des risques :

#### **Risques majeurs :**

- La centralisation des résultats : Les résultats sont transmis individuellement au chef de projet. Ce système présente une faille évidente : si pour une raison ou une autre, le chef de projet est absent, il n'y a rien de possible. Le chef de projet est ainsi à un carrefour des informations, et le seul détenteur global de ces derniers. C'est un risque majeur car un élément primordial du système (les résultats) est détenu par un seul individu.

Catégorie : *Sécurité, continuité, connectivité, contrôle des données, flexibilité.*

- La méthode de transmission/sauvegarde des résultats : Les résultats sont transmis oralement au chef de projet, ce qui est une faille majeure dans le système car cela est dépendant de la mémoire et/ou du format de sauvegarde de données choisi par le chef de projet. La moindre erreur à ce niveau là peut considérablement ralentir le processus de testing.

Catégorie : *Sécurité, continuité, contrôle des données, Timing*

- Les tests de non-régression : Parce que ces tests sont effectués de façon manuelle, la seule garantie qu'ils sont faits de façon toujours similaires et donc correctes concerne l'expérience des personnels, dont nous reparlerons ensuite. Le fait que ces tests soient réalisés de manière uniquement manuelle peut créer un grand problème si des erreurs ne sont pas trouvées et se propagent dans les versions suivantes, rendant difficile de retrouver la source.

Catégorie : *Connectivité, Continuité, Contrôle des données, Réusabilité/Expérience*

### **Risques importants :**

- Pas d'historisation des tests : Il n'est jamais fait mention d'aucune procédure d'historisation des tests, ou même de définition concrète. Cela met en jeu la qualité des tests effectués, et donc l'application, mais également le temps nécessaire pour former des nouveaux testeurs et pour effectuer les tests.

Catégorie : *Réusabilité/Expérience, Timing, Contrôle des données*

- Manque de formation du personnel : Comme indiqué, tous les tests sont faits manuellement et aucun personnel n'est formé à la mise en place de tests automatiques. Par ailleurs, aucun processus qualité n'est indiqué. Il n'y a aucun signe d'un guide ou de consignes précises remises à l'accueil de nouveaux membres, il faut donc prendre un temps considérable pour la formation de nouveaux testeurs, sans garantie que la période de début soit réussie en termes de qualité du testing.

Catégorie : *Réusabilité/Expérience, Timing, Continuité*

- Suivi des liens + mouvements de personnels sur Excel : Des tas d'informations sont enregistrées manuellement (aussi bien sur l'organisation de l'application que du personnel) sur Excel par le chef de projet. Cela présente des risques de perte de données, de sécurité et de rigueur. Un outil de suivi(couplé au stockage des données dans un espace disponible aisément aux supérieurs du chef de projet) n'est disponible que partiellement, car il est trop complexe pour être utilisé, et donc il y a un problème de double logique/doublage de données présenté.

Catégorie : *Flexibilité, Contrôle des données, Sécurité*

### **Risques mineurs :**

- Mouvement de personnels trop fréquent : En liaison avec le point sur la formation, si la formation en testing ne semble pas être performante, le fait que les mouvements de personnels soient importants ne facilite pas cette tâche, rendant difficile la mise en place d'un vrai service de testing avec des testeurs expérimentés.

Catégorie : *Timing, Réusabilité/Expérience*

- Organisation qui manque d'un outil de mise en commun : S'il est vrai que les 9 testeurs travaillent sur des catégories différentes, rien n'indique qu'il ne teste pas des objets/fonctions communes. Seul le chef de projet et son suppléant selon leur collaboration peuvent le savoir, il y a donc des risques de double test et double résultat même éventuellement.

Catégorie : *Timing, Réusabilité/Expérience, Contrôle des données*

Il est important de noter que tous ces risques sont relevés pour être sûr que l'amélioration et le cahier des charges des solutions proposées peut à la fois offrir des solutions souples et flexibles (notamment en termes financier) mais aussi être sûr que les mesures permettront d'assurer un service performant en limitant les problèmes coûteux à l'avenir.

## Question 2

Après avoir étudié les lacunes et les risques dans l'organisation mise en place dans la société COMURLUX, nous avons tenté de mettre en évidence différents axes, visant à améliorer les performances globales de l'entreprise. Pour ce faire nous avons suivi les fondamentaux de la méthode TMap Next et avons dégagé plusieurs mesures qui pourraient être profitables à l'entreprise.

Tout d'abord, il nous faut définir une nouvelle stratégie de test, en déterminant quels éléments doivent être testés et de quelle manière, une fois déterminés, il faudra spécifier les composantes nécessaires à la réalisation de tels tests (scénario, cas de test, jeux de données).

Nous pouvons identifier qu'il nous faut améliorer la stabilité de l'application, afin de permettre d'incorporer de nouvelles fonctionnalités (dont certaines liées au processus de test) au système existant. Il nous faut également prévoir quels processus de test peuvent être automatisés, afin de gagner en efficacité. De telles améliorations permettraient de mettre en place une nouvelle configuration visant à pallier les lacunes de la situation actuelles; le processus d'automatisation notamment induirait un grand gain de temps, car, actuellement, ces tests fastidieux sont réalisés manuellement ce qui est relativement peu productif.

Afin de gagner en productivité, il faut effectuer des tests de suivi de l'application lors des procédures de mise à jour. En effet, dans la situation actuelle, les patchs correctifs apportent des anomalies qui doivent être étudiées et éradiquées à chaque fois. Ce problème existant nuit clairement à la productivité car des processus de test doivent être établis à chaque patch, ce qui signifie que les employés ne peuvent utiliser l'application.

Pour résoudre ce problème, des tests de non régression doivent être effectués sur un système annexe, afin de vérifier le bon fonctionnement des patchs à appliquer tout en n'entravant pas la production.

Nous avons également relevé un sérieux problème de compétences, lié à l'utilisation de cette application. En effet, seul le chef de projet informatique est apte à effectuer des modifications liées aux mouvements du personnel. Pour combler cette lacune, il nous faut revoir l'infrastructure de l'entreprise, en proposant des formations et des guides d'utilisations, détaillant clairement la démarche à suivre afin d'effectuer des modifications dans l'application, sans pour autant provoquer de graves conséquences. Cette formation permettrait à l'ensemble du personnel de gagner en compétences techniques et, de ce fait, de réduire la fréquence des anomalies.

Néanmoins, cette mesure, liée au personnel, n'est pas la seule à mettre en place afin de gagner en efficacité. En effet, un grand nombre d'outils, et de fonctionnalités de contrôle doivent être mis en place, pour optimiser les résultats des tests.

Une historisation des problèmes rencontrés permettrait de résoudre plus rapidement des anomalies récurrentes qui surviendraient, suite à l'utilisation de l'application. De plus, la mise en place de métriques définies et précises, faciliterait la qualification des anomalies et

permettrait aux gestionnaires de la stratégie de test, de pouvoir classer les erreurs en fonction de leur degré de criticité. En ce sens, un système de retour sur la qualité des données (et du fonctionnement en général) serait intéressant à mettre en place. Cela permettrait aux employés de rapporter les anomalies qu'ils auraient pu découvrir et ainsi de réduire le temps nécessaire au traitement de ces dernières.

Enfin la gestion de la complexité peut être assistée de deux manières:

Si l'on souhaite conserver l'organisation actuelle, en utilisant le logiciel Excel, alors les données qui sont renseignées dans ce logiciel doivent faire parties de la stratégie de test, définie au préalable. Cela entraînerait des tests qui seraient appliqués dans un environnement extérieur à l'application elle-même, mais qui resteraient vitaux pour le bon fonctionnement du système dans son intégralité.

En revanche, nous pouvons également proposer l'utilisation d'une solution alternative à Excel, gérant la complexité de l'application, qui serait un module indépendant de l'application "mère", mais qui fonctionnerait de manière autonome en cas de défaillance de cette dernière. Un tel outil serait plus efficace car il aura été conçu pour répondre plus précisément aux questions de versionning, de planification et de modification.

Toutes ces propositions d'amélioration, basées pour la plupart sur les fondamentaux de la méthode TMap Next, ont pour but primaire d'améliorer les performances de l'entreprise en comblant les lacunes existantes dans la situation actuelles des choses. Néanmoins, il nous faut définir un catalogue de service détaillé, dans lequel les propositions suggérées ci dessus, seraient quantifiées en termes de coût, de temps de mise en oeuvre et de complexité d'installation.

### Question 3

Dans un souci de réalisme, nous avons choisi de modéliser un catalogue de service tel qu'il pourrait être proposé dans une situation réelle d'entreprise.

Titre de service	<b>Stratégie et planification du test</b>
Le service « Stratégie du test et planification » permet à COMURLUX d'établir une meilleure stratégie et planification du test pour atteindre son objectif et de maximiser le ROI.	
<b>Les critères pour la COMURLUX :</b>	
<p>La COMURLUX a besoin de ce service parce que :</p> <ul style="list-style-type: none"> <li>▪ Le risque et la complexité du SI augmentent et leur impact sur les acteurs dans la COMURLUX augmente également. <ul style="list-style-type: none"> <li>✓ L'application est très sensible aux erreurs et une mauvaise manipulation peut avoir des impacts graves sur l'ensemble du SI.</li> <li>✓ Les patchs pour le progiciel peuvent poser des risques graves.</li> <li>✓ La mise à jour de son organigramme RH est de surcroît un risque majeur.</li> </ul> </li> <li>Donc l'importance des tests augmente aussi.</li> <li>▪ L'efficacité et le ROI des tests peuvent augmenter par le test structuré avec une bonne stratégie et planification des tests.</li> <li>▪ Pour faire les tests dans un temps limité, nous avons également besoin d'une bonne</li> </ul>	

stratégie et planification des tests.
<b>Description du service :</b>
<p>Tout commence avec les objectifs du client et son business. Nous allons trouver et nommer des parties prenantes. Nous allons analyser les processus métier du client et son business autour des parties prenantes.</p> <p>Nous faisons analyser, identifier, évaluer et mesurer les risques par rapport aux objectifs, aux exigences, aux qualités attendues par le client. Nous communiquons toujours et donnons un retour sur les résultats de l'analyse de risque. Nous allons faire hiérarchiser les risques, déterminer l'effort de test, estimer les charges et développer et élaborer la meilleure stratégie pour le client et son objectif du test.</p> <p>Nous allons faire de choix et définir les priorités en considérant le budget, le temps, le niveau de qualité attendu, les compétences requises et gravité de l'impact en cas de dysfonctionnement, erreurs, mauvaise manipulation et etc. Nous allons mettre les processus métier du client et son business au cœur de ce service pour la stratégie de test. Nous allons toujours vérifier l'adéquation par apport aux besoins d'initiaux du client.</p> <p>Pour la planification, nous allons définir les charges du test, l'organisation, les ressources techniques, les ressources humaines, le cycle de vie du test à utiliser, les 'outils, les activités et les environnements pour le test.</p>

**Tableau 1. Stratégie et planification du test**

UO	Nombre de travailleur	Jours d'œuvre	Prix par jour
Consultant Manager Testing	1	10 jours	800 euros
Consultant outillage Testing	1	5 jours	600 euros
Consultant Testing Senior	2	10 jours	600 euros
<b>Total prix</b>			<b>23000 euros</b>

<b>Titre de service</b>	<b>Automatisation des tests</b>
Le service « Automatisation des tests » permet au client d'automatiser des tests et maximiser le ROI des tests en termes de ressources.	
<b>Les critères pour la COMURLUX :</b>	
Les tests fonctionnels nécessitent l'utilisation d'outils d'automatisation, mais les ressources internes de la COMURLUX n'ont aucunes compétences dans ce domaine. Donc elle a besoin de ce service et elle peut aussi maximiser le ROI des tests.	
<b>Description du service :</b>	
<p>Nous allons analyser business du client, les processus métiers, l'objectif du test, les ressources humaines et techniques, l'environnement, le budget et le temps permis pour l'automatisation des tests.</p> <p>Nous allons communiquer les résultats au client pour développer et élaborer la stratégie d'automatisation des tests. En nous basant sur la communication et les résultats de l'analyse, nous allons établir la stratégie d'automatisation des tests.</p>	

Nous allons choisir les tâches des tests que nous pouvons automatiser d'une manière réaliste et pratique. Pour maximiser le ROI, nous allons nous concentrer sur les tâches courantes et répétitives. Le coût d'automatisation des tests peut-être élevé à court terme mais à long terme cet investissement vaut la peine.

Nous allons choisir les outils des tests d'utiliser et modifier ce qui est nécessaire pour l'automatisation des tests. Nous allons développer les cas de tests en script pour automatiser des tests, construire l'environnement. Nous allons former les outils et méthodes pour assurer le client des niveaux appropriés de compétences.

**Tableau 2. Automatisation des tests**

UO	Nombre de travailleur	Jours d'œuvre	Prix par jour
Consultant outillage Testing	1*	**	600 euros
Support Méthode et Outils Senior	1*	**	600 euros
Chef de projet	1*	**	600 euros
Administrateur	1*	**	400 euros
Architecturer	1*	**	600 euros
<b>Total prix</b>			

\*\*Les jours d'œuvre dépendent des exigences de l'automatisation et outils à choisir.

\*Le nombre de personnel nécessaire dépend des exigences de l'automatisation et outils choisis.

Titre de service	<b>Externalisation de ressources du test</b>
Le service « Externalisation de ressources du test » permet au client d'introduire les expertises des métiers du test et renforcer les ressources humaines pour les tests.	
<b>Les critères pour la COMURLUX :</b>	
À cause des contraintes de traitement des rémunérations, les tests doivent être réalisés la première ou la dernière semaine du mois. Donc il peut y avoir un besoin d'aide pour respecter les contraintes.	
<b>Description du service :</b>	
<p>Ce service permet le client de renforcer les expertises et les ressources humaines quand il faut d'une manière flexible et rapide et pendant le temps qu'il veut.</p> <p>Quand le client a besoin de ressources humaines pour le test, nos testeurs qui sont expérimentés en diverses matières de test, sont les ressources que nous mettons à disposition du client. Nous pouvons servir au client pour les activités suivantes :</p> <ul style="list-style-type: none"> <li>- Test d'acceptation</li> <li>- Test de régression</li> <li>- Tests fonctionnel</li> <li>- Tests de charge et performance</li> <li>- Tests d'intégration et interopérabilité</li> <li>- Tests d'utilisabilité</li> </ul>	

**Tableau 3. Externalisation de ressources du test**

UO	Prix par jour
Consultant Testing	400 euros
Chef de projet	600 euros



Service Manager Testing

800 euros

<b>Titre de service</b>	<b>Exécution de Tests</b>
Le service « Exécution de Tests » est un service d'exécution des tests pour le client et permet au client d'attendre l'objectif du test avec nos ressources techniques, humaines et des experts.	
<b>Les critères pour la COMURLUX :</b>	
Même si les risques et la complexité augmentent autour l'application et l'ensemble du SI, la COMURLUX n'a pas d'équipe spécialisée pour les tests. Pour gérer les risques et augmenter le ROI des tests, la COMURLUX a besoin de ce service.	
<b>Description du service :</b>	
<p>Pour exécuter les tests, nous allons faire plusieurs activités et répéter celles-ci. D'abord nous avons besoin de l'infrastructure de tests. Les objets à tester doivent être livrés.</p> <p>Nous allons revoir l'objet à tester et l'infrastructure. Nous allons installer des données, exécuter des tests, comparer et analyser des résultats, enregistrer des résultats, parfois tester à nouveau. Nous allons maintenir des scénarios de test pour s'adapter les tests à l'objectif de test si nécessaire. Pour chaque test, nous allons aussi faire du reporting.</p> <p>Ce service sert divers sortes de tests comme les suivants :</p> <ul style="list-style-type: none"><li>- Test d'acceptation</li><li>- Tests fonctionnel</li><li>- Tests de charge et performance</li><li>- Tests d'intégration et interopérabilité</li><li>- Tests d'utilisabilité</li></ul>	

Tableau 4. Exécution de Tests

UO	Nombre de travailleur	Jours d'œuvre	Prix par jour
Consultant Testing	1*	**	400 euros
Chef de projet	1*	**	600 euros
Service Manager Testing	1*	**	800 euros
Support Méthode et Outils	1*	**	400 euros
Administrateur de plateforme	1*	**	400 euros
Total prix			

\*\*Les jours d'œuvre dépendent des exigences de l'automatisation et des outils à choisir.

\*Le nombre de personnel nécessaire dépend des exigences de l'automatisation et des outils choisis.

<b>Titre de service</b>	<b>Gestion et suivi des anomalies</b>
Le service « Gestion et suivi des anomalies » permet au client de construire un système de gérer et suivre des anomalies identifiées.	
<b>Les critères pour la COMURLUX :</b>	

La COMORLUX n'a pas de système de la gestion et suivi des anomalies. Pour cela, tout se passe par communication orale au chef de projet. Avec cette méthode, nous ne pouvons pas gérer et suivre les anomalies. Ca poserait un risque majeur. Donc elle a besoin de ce service.

**Description du service :**

Nous allons construire un système d'information qui permet d'enregistrer des anomalies avec des informations, analyser et évaluer le degré de gravité des anomalies, mettre en place une priorisation des anomalies et un partage des information entre les personnes autorisés.

Ce système permet des services de reporting et suivi. Nous pouvons mettre à jour de l'état d'anomalies et ajouter des informations supplémentaires. Le reporting montre des informations sur des anomalies par exemple :

- L'état de anomalie
- Les anomalies corrigées
- Les anomalies qui restent en suspens.
- Le temps écoulé et le temps restant.
- Les tests spécifiés, les tests exécutés et les résultats.

**Tableau 5. Gestion et suivi des anomalies**

UO	Nombre de travailleur	Jours d'œuvre	Prix par jour
Support Méthode et Outils Senior	1	2	600 euros
Consultant outillage Testing	1	3	600 euros
Administrateur de plateforme	1	5	400 euros
<b>Total prix</b>			<b>5000 euros</b>

\*Ce tableau est basé sur l'hypothèse que nous allons utiliser un logiciel source libre.

<b>Titre de service</b>	<b>CMDB (Configuration Management Database)</b>
Le service « CMDB » permet le client de gérer les configurations des systèmes informatiques. Il permet d'enregistrer et de comprendre les relations entre les composants des systèmes informatiques, d'enregistrer les configurations et suivre tous les changements sur les composants.	
<b>Les critères pour la COMURLUX :</b>	
Pour gérer le risque et la complexité, le chef de projet actuel utilise des feuilles Excel pour représenter les relations entre les différents applications du SI, tracer les modifications, suivre le versioning, représenter l'organigramme cible, planifier les dates et le contenu des livraisons des patches. Mais cette méthode possède plusieurs failles. Le chef a besoin de ce service pour gérer le risque et la complexité.	
<b>Description du service :</b>	
Le service permet d'identifier, enregistrer et restituer de l'information sur tous les composants du SI incluant leurs versions, les sous-composants et les relations entre eux. Il intègre aussi les matériels, les logiciels, les applications et les documentations associées.	
Le service permet au client de contrôler des éléments de configurations et suivre des changements. Il aussi permet de faire vérification et audits sur la gestion de configurations.	

**Tableau 6. CMDB (Configuration Management Database)**

UO	Le prix par an sans limitation d'utilisateurs	Prix par jour
Service en « Cloud ».	1000 Euros	
Total prix		1000 Euros

Titre de service	<b>TEaaS (Test Environnement as a Service)</b>
Le service « TEaaS » permet au client de construire l'environnement avec les matériels et les logiciels qu'il a besoin pour des tests non pas en achetant mais en empruntant tout cela.	
<b>Les critères pour la COMURLUX :</b>	
Le responsable d'exploitation met à disposition un environnement de tests ayant reçu les derniers patches. Mais en comparaison avec le système opérationnel, le serveur et les logiciels qui composent l'environnement sont peu utilisés. Donc avoir l'environnement de tests n'est pas très efficace et coûte cher. En utilisant le service TEaaS, la COMORLUX peut avoir accès à l'environnement quand elle a besoin avec les logiciels choisis. Avec ce service, elle ne s'occupe pas de l'entretien de l'environnement de test. Avec ce service, elle n'a besoin de faire de maintenance de l'environnement de test régulièrement parce que c'est le principe même du service offert..	
<b>Description du service :</b>	
Nous prêtons les serveurs, les logiciels et les matériels que le client a besoin pour des tests et permettons de construire l'environnement de test. Ce service permet au client d'économiser le coût des tests et de maximiser le ROI des tests.	

[Tableau 7. TEaaS \(Test Environnement as a Service\)](#)

UO	Le prix
Service en « Cloud ».	Ca dépend -du type de serveur et sa puissance et -des logiciels à installer.

## Question 4

Les outils que nous proposons à COMORLUX sont Tuleap ALM Suite et Canoo Webtest. Vous pouvez trouver les informations ci-dessous pour chaque outil.

- Tuleap ALM Suite est un outil open source de gestion de cycle de projet. Il permettra d'améliorer la gestion du projet et rendra la collaboration au sein COMORLUX plus productive et efficace. Avec l'outil, COMORLUX peut améliorer les points suivants :
  - **Planifier et suivre de projet.** Il pourra organiser des tâches, des ressources et suivre les progrès et le travail restant.
  - **Tracer les modifications.** Il pourra suivre les changements sur tout type d'artefacts de projet, par exemple, les risques, les tâches, les anomalies, et etc. De plus, l'outil permet tracer et lier des artefacts vers le code, la version de patch, etc.
  - **Suivre le versioning.** Tuleap permet l'intégration avec des outils centralisés de gestion de configuration comme CVS et Subversion.

- **Planifier les dates et le contenu des livraisons des patches.** Tuleap fournit l'automatisation de la construction de la version du logiciel et aussi des notes de publication.
  - **Collaboration.** Tuleap fournit la communication intra-équipe. De plus, il envoie des notifications par email pour les différents artefacts de projet.
- Canoo Webtest est un outil open source d'automatisation pour effectuer les tests fonctionnels. Il permettra d'économiser des efforts et du temps dans la phase d'exécution. Il est simple et facile pour développer des tests. De plus, il supporte la direction par les données donc COMORLUX pourra tester avec divers jeux de données.
- 

Ces outils sont choisis car ils correspondent aux objectifs fixés depuis le début de notre audit, c'est-à-dire un budget serré, or ceux-ci sont des logiciels open source, ils sont donc libres de droit et gratuit.

Par ailleurs, leur utilisation est simple, et donc la formation l'est également, prérogative obligatoire quand on considère un mouvement de personnel régulier.

Enfin, la communication grâce à ces outils est simple et efficace, contrairement au fiat de transmettre des résultats par oral comme précédemment.

Ensuite, les démarches pour implanter les outils ci-dessus sont comme suit :

1. Définition du processus de test structuré  
Sélectionner le cycle de vie de projet et le processus de test qui sont appropriés pour COMORLUX. Dans ce cas, nous pouvons utiliser le cycle de vie TMAP. Il se compose de phase de planification, préparation, spécification, exécution, clôture, contrôle et infrastructure. Ensuite, nous doivent définir quelles sont les activités nécessaires dans chacune des phases.
2. Planification
  - Définir les rôles et les responsabilités associées au projet. Qui fera quoi et quand ? Après, créer le plan de mettre en œuvre de processus de test.
  - Définir des matériels et des logiciels nécessaires pour le test et acheter s'il manque.
3. Mise en place l'infrastructure
  - Mettre en place l'environnement de test. L'environnement de test doit être séparé de l'environnement de production mais ses configurations doivent être les mêmes ou proche de l'environnement de production.
  - Mettre en place les outils de test. Dans notre cas il y a deux outils pour installer : Tuleap ALM Suite et Canoo Webtest. Ensuite, il faut tester que tous les éléments travaillent bien pendant le processus de test. Un nom d'utilisateur et mot de passe pour la connexion à l'outil doivent être créé avant la formation.
4. Manuels et politiques  
Etablir les manuels d'utilisation des outils et des politiques de test. Communiquer ces informations à tout le monde et faire en sorte qu'elles soient connues. Les retours « qualité » devront être nombreux pour pouvoir établir un manuel simple à remettre aux nouveaux personnels du service. Par ailleurs, un service de mentor sera mis en

place, les plus expérimentés du service formant ainsi les nouveaux arrivants et leur offrant leur expérience et des réponses à leurs questions. L'objectif est de contrer l'effet de ralenti à chaque changement de personnel.

5. Formation

Les formations aideront l'équipe de test de COMORLUX à comprendre le processus de test et les façons d'utiliser les outils. Ils se composent de plusieurs sujets et chaque formation est adaptée pour chaque rôle comme la table ci-dessous :

Formation	Chef de projet	Architecte d'infrastructure	Testeurs
Le processus de test	X	x	x
Utilisation Tuleap dans chaque phase de test. Par exemple, comment organiser des tâches et gérer des anomalies par l'outil.	X	x	x
Construction et exécution des tests automatisés par Canoo Webtest.			x

6. Changement

Pour passer aux nouveaux outils, il faut donner une date et l'heure limite pour mettre à jour les données dans l'ancien système. Ensuite, les informations et documents existantes doivent être transférées aux nouveaux outils, par exemple, les spécifications de test doit être téléchargées dans Tuleap.

De plus, il a besoins de beaucoup d'efforts pour développer des tests automatisés. Il peut-être pas possible d'avoir tous les cas de test en automation donc il faut planifier que quels tests doivent faire en automation d'abord.

7. Evaluation

Pendant le premier et deuxième vrai test, nous fournirons des conseils sur des problèmes et aussi évaluerons la qualité du processus de test et préviendrons d'amélioration.

## Conclusion

Dans le cadre de cette étude de cas, nous avons tenté d'analyser les problèmes de l'entreprise COMURLUX et avons proposé des solutions afin de combler les lacunes existantes dans leur système d'information.

Nous avons établi une marche à suivre, en partant d'un constat actuel et des risques existants. L'audit s'est poursuivi par des axes d'améliorations, pour aboutir sur les outils et le catalogue de service.

Au final, nous avons déroulé le processus logique qui conduit à améliorer un processus, un système, une équipe ou un département dédié au test. On voit que les connaissances ne servent pas seulement à « parler technique » mais également à adopter une manière de penser « testing » qui prend toujours en compte un souci d'utiliser l'expérience pour aller plus vite tout en étant plus fiable.

Ce type de situation peut très bien se retrouver en cas pratique dans les entreprises actuelles, cependant, grâce aux notions que nous avons découvertes et aux méthodologies acquises, nous serions capable d'identifier rapidement les processus possédant des faiblesses et ceux menaçant le bon fonctionnement d'un service, voire de l'entreprise en elle-même.