**Mémoire du Projet de Fin d’Études**

**Pour l’obtention du titre**

**D’ingénieur d’État en Informatique**

**Génie Logiciel et Développement Mobile**

**Vers une Meilleure Qualité des Données : Développement et Implémentation d'un Outil de Mesure et de Contrôle**

**Soutenu par :**  Encadré par :

Hicham Taib

**Année universitaire : 2023 – 2024**

# Dedication

To those who have waited patiently for the fruits of their good education, their devotion, and their sacrifices

To my beloved parents

To my extended family who have supported me all along

To all my professors, coworkers and friends

To all my dearest,

I Dedicate This Humble Work to You

# **Remerciements**

Il a été très difficile d'écrire ces quelques lignes de remerciement, de peur d'oublier certaines des nombreuses personnes que je dois mentionner pour leur aide, leur accueil et leur soutien. Je voudrais les rassurer de ma profonde gratitude, même si leurs noms ne sont pas inclus ici.

Tout d'abord, je remercie le bon Dieu tout-puissant de m'avoir donné la force, la volonté et le courage nécessaires pendant ces années d'études.

À la fin de ce projet, je tiens à exprimer ma gratitude et mon immense respect envers tous mes enseignants de l'École Nationale des Sciences Appliquées qui ont contribué d'une manière ou d'une autre à l'élaboration de cette thèse ainsi qu'à la réussite de cette merveilleuse dernière année académique.

Je remercie également mes superviseurs, Mme Ilham Oumaira et M. Abdellatif Hassnaoui, pour le suivi de mon travail, leurs précieux conseils, leurs encouragements continus et leur engagement à mener ce travail jusqu'au bout. Je voudrais également remercier ma responsable, Mme Sarrah KHATIBI, pour m'avoir donné cette merveilleuse opportunité, pour sa confiance, son soutien, son accueil et la motivation qu'elle m'a donnée pour réussir ce travail. Je remercie aussi tous les coéquipiers pour leur accueil, les expériences partagées et le partage de leur savoir-faire au quotidien. Mes sincères remerciements vont également aux membres du jury, qui nous ont fait l'honneur d'accepter d'évaluer ce travail.

Je ne pourrais pas conclure sans remercier toutes les personnes qui, d'une manière ou d'une autre, ont contribué à la réalisation de ce travail.

# Résumé

Le présent rapport synthétise le travail que j'ai accompli lors de mon stage à Orange Business Services Maroc dans le cadre de mon diplôme d'ingénieur logiciel.  
  
La qualité des données fait référence au degré de conformité des données aux normes d'une entreprise en termes d'exactitude, de validité, d'exhaustivité et de cohérence. Elle revêt une importance cruciale car des informations précises et opportunes sont nécessaires pour gérer les services et les responsabilités, ainsi que pour rendre compte aux auditeurs et aux inspecteurs chargés d'évaluer notre performance et notre gouvernance.

DQSC (Data Quality Score Card) est un outil qui mesure la qualité des données au sein de l'écosystème d'Orange et présente les mesures sous forme de tableau de bord pour l'utilisateur final. J'ai travaillé sur ce projet lors de mon stage PFE, au cours duquel j'ai eu l'occasion de développer plusieurs KPI et d'ajouter des fonctionnalités intéressantes. J'ai utilisé un ensemble de technologies telles que Spark/Scala, l'écosystème Spring, Angular et MongoDB, dans un environnement Big Data.

**Mots clés** : DQSC, KPI, Spark, Scala, Spring, Angular, MongoDB, SCRUM.

# Abstract

This report summarizes the work I completed during my internship at Orange Business Services Morocco as part of my software engineering degree.  
  
Data quality refers to how well data aligns with a company's standards for accuracy, validity, completeness, and consistency. It is crucial because we rely on accurate and timely information for managing services, accountability, and reporting to auditors and inspectors who assess our performance and governance.  
  
During my internship, I worked on a project called DQSC (Data Quality Score Card). This tool measures the quality of data from Orange ecosystem and presents the metrics in a user-friendly dashboard. I had the opportunity to develop KPIs and add features using technologies such as Spark/Scala, Spring ecosystem, Angular, and MongoDB, all within a Big Data environment.

**Keywords** : DQSC, KPI, Spark, Scala, Spring, Angular, MongoDB, SCRUM.

# Liste des figures

[Figure 1 : OBS logo](#_bookmark12) 3

[Figure 2 : OBS Morocco’s key figures](#_bookmark14) 3

[Figure 3 : Customer Inventory Score Card - CISC](#_bookmark19) 5

[Figure 4 : DQSC vs CISC](#_bookmark21) 5

[Figure 5 : DQSC home page](#_bookmark25) 6

[Figure 7 : Context diagram](#_bookmark31) 9

[Figure 8 : General use cas diagram](#_bookmark37) 12

[Figure 9 : SCRUM life cycle](#_bookmark40) 13

[Figure 10 : Kanban board](#_bookmark42) 14

[Figure 11 : Branching strategy](#_bookmark45) 15

[Figure 12 : Gantt diagram](#_bookmark50) 16

[Figure 13 : Project architecture](#_bookmark52) 17

[Figure 14 : CAP Theorem](#_bookmark58) 18

[Figure 15 : Sprint 39 class diagram](#_bookmark73) 24

[Figure 16 : Sequence diagram : ’Download discrepancy report’](#_bookmark75) 24

[Figure 17 : Profitability KPI in the dashboard](#_bookmark77) 25

[Figure 18 : Active Business Ethernet/Local Ethernet services not billed report](#_bookmark78) 25

[Figure 19 : Resources relationships](#_bookmark85) 27

[Figure 20 : Sprint 40 sequence diagram](#_bookmark89) 28

[Figure 21 : Data transformation flow](#_bookmark91) 29

[Figure 22 : Different Ordered and Polled IP Bandwidth KPI in the dashboard](#_bookmark93)………….29

[Figure 23 : Sequence diagram : ”Generate KPI metrics”](#_bookmark106) 34

[Figure 24 : Sprint 41 class diagram](#_bookmark108) 35

[Figure 25 : SLA KPI in the dashboard](#_bookmark110) 35

[Figure 26 : SLA Data Quality report](#_bookmark111) 36

[Figure 27 : Sprint 42 class diagram](#_bookmark118) 37

[Figure 28 : Sequence diagram : "Solved discrepancies report"](#_bookmark120) 38

[Figure 29 : Resolved discrepancies integrated into the dashboard](#_bookmark122) 39

[Figure 30 : Resolved discrepancies report](#_bookmark123) 39

[Figure 31 : Sprint 43 sequence diagram](#_bookmark129) 41

[Figure 32 : Sprint 43 demo](#_bookmark131) 41

# Liste des tableaux

|  |  |  |  |
| --- | --- | --- | --- |
| [Table 1 :](#_bookmark35) | [Product backlog](#_bookmark35) . | . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 11 |
| [Table 2 :](#_bookmark48) | [sprints planning](#_bookmark48) . | . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 16 |
| [Table 3 :](#_bookmark68) | [Sprint 39 backlog](#_bookmark68) | . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 22 |
| [Table 4 :](#_bookmark70) | [Inspected items for profitability KPI](#_bookmark70) . . . . . . . . . . . . . . . . . | | 23 |
| [Table 5 :](#_bookmark82) | [Sprint 40 backlog](#_bookmark82) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 26 |
| [Table 6 :](#_bookmark84) | [Inspected items for different IP bandwidth KPI](#_bookmark84) . . . . . . . . . . . | | 27 |
| [Table 7 :](#_bookmark86) | [Example of a product characteristics field](#_bookmark86) . . . . . . . . . . . . . | | 27 |
| [Table 8 :](#_bookmark100) | [Sprint 41 backlog](#_bookmark100) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 32 |
| [Table 9 :](#_bookmark102) | [SLA KPI : Inspected items](#_bookmark102) . . . . . . . . . . . . . . . . . . . . . . . | | 33 |
| [Table 10 :](#_bookmark103) | [SLA KPI : Discrepancy items](#_bookmark103) . . . . . . . . . . . . . . . . . . . . . | | 33 |
| [Table 11 :](#_bookmark114) | [Sprint 42 backlog](#_bookmark114) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 36 |
| [Table 12 :](#_bookmark126) | [Sprint 43 backlog](#_bookmark126) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 40 |

# Liste des abréviations

|  |  |
| --- | --- |
| Abbreviation | Signification |
| **ACID** | Atomicity, Consistency, Isolation, and Durability |
| **API** | Application Programming Interface |
| **BO** | Business Owner |
| **CAP** | Consistency, Availability, and Partition |
| **CI/CD** | Continuous Integration/Continuous Delivery |
| **CISC** | Customer Inventory Score Card |
| **CMI** | Customer Marketing Innovation |
| **CPU** | Central Processing Unit |
| **CaaS** | Containers as a Service |
| **DB** | DataBase |
| **DQ** | Data Quality |
| **DQSC** | Data Quality Score Card |
| **DTO** | Data Transfer Object |
| **E2E** | End To End |
| **EDH** | Enterprise Data Hub |
| **EOG** | End Of Game |
| **ETL** | Extract, Transform and Load |
| **FBO** | French Business Operations |
| **FINV** | Federated Inventory |
| **GUI** | Graphical User Interface |
| **HTTP** | Hypertext Transfer Protocol |
| **IP** | Internet Protocol |
| **IR** | Installed Resource |
| **IT** | Information Technology |
| **JAR** | Java ARchive |
| **K8S** | Kubernetes |
| **KPI** | Key Performance Indicator |

**Table des matières**

[Dedication 2](#_Toc168426883)

[Remerciements 3](#_Toc168426884)

[Résumé 4](#_Toc168426885)

[Abstract 5](#_Toc168426886)

[Liste des figures 6](#_Toc168426887)

[Liste des tableaux 7](#_Toc168426888)

[Liste des abréviations 8](#_Toc168426889)

[Introduction générale 11](#_Toc168426890)

[1 Contexte général 12](#_Toc168426891)

[1.1 Présentation de l'organisation hôte 13](#_Toc168426892)

[1.1.1 Orange Business Services : L'entreprise 13](#_Toc168426893)

[1.1.2 Chiffres clés 14](#_Toc168426894)

[1.1.3 Innovation Marketing Client 14](#_Toc168426895)

[1.2 Présentation du Projet 15](#_Toc168426896)

[1.2.1 Besoins Métier 15](#_Toc168426897)

[1.2.2 Solution initiale (CISC) : 15](#_Toc168426898)

[1.2.3 DQSC vs CISC 16](#_Toc168426899)

[1.2.4 Objectifs de DQSC 17](#_Toc168426900)

[1.2.5 Problématique 17](#_Toc168426901)

[1.2.6 Solution 18](#_Toc168426902)

[2 Étude préliminaire 19](#_Toc168426903)

[2.1 Spécification des Besoins 19](#_Toc168426904)

[2.1.1 Identification des Principaux Acteurs 19](#_Toc168426905)

[2.1.2 Exigences Fonctionnelles 20](#_Toc168426906)

[2.1.3 Exigences Non Fonctionnelles 21](#_Toc168426907)

[2.1.4 Product backlog 22](#_Toc168426908)

[Table 1 – Product backlog 23](#_Toc168426909)

[2.1.5 Identification des Cas d'Utilisation Principaux 24](#_Toc168426910)

[2.2 Gestion de Projet 25](#_Toc168426911)

[2.2.1 SCRUM 25](#_Toc168426912)

[2.2.2 Kanban board 26](#_Toc168426913)

[2.2.3 Jira et Confluence 26](#_Toc168426914)

[2.2.4 Workflow GitLab 27](#_Toc168426915)

[2.2.5 Planification du Projet 29](#_Toc168426916)

[2.3 Architecture du Projet 29](#_Toc168426917)

[2.4 Stack Technique 31](#_Toc168426918)

[2.4.1 Spark/Scala 31](#_Toc168426919)

[2.4.2 Spring Web-Flux 31](#_Toc168426920)

[2.4.3 MongoDB 32](#_Toc168426921)

[2.4.4 Angular 32](#_Toc168426922)

[2.4.5 Docker 33](#_Toc168426923)

[2.4.6 Kubernetes 33](#_Toc168426924)

[2.4.7 Redis 34](#_Toc168426925)

[3 Release 1.0 36](#_Toc168426926)

[3.1 Sprint 39 : Ethernet services not billed KPI 36](#_Toc168426927)

[3.1.1 Présentation du KPI 36](#_Toc168426928)

[3.1.2 Sprint Backlog 36](#_Toc168426929)

[3.1.3 Spécification et analyse des exigences 37](#_Toc168426930)

[3.1.4 Conception 38](#_Toc168426931)

[3.1.5 Implémentation 40](#_Toc168426932)

[3.2 Sprint 40: Different Ordered and Polled IP Bandwidth KPI 41](#_Toc168426933)

[3.2.1 Présentation du KPI 41](#_Toc168426934)

[3.2.2 Sprint Backlog 42](#_Toc168426935)

[3.2.3 Spécification et analyse des exigences 42](#_Toc168426936)

[3.2.4 Conception 44](#_Toc168426937)

[3.2.5 Implémentation 46](#_Toc168426938)

[4 Release 2.0 48](#_Toc168426939)

[4.1 Sprint 41 : SLA data quality KPI 49](#_Toc168426940)

[4.1.1 Présentation du KPI 49](#_Toc168426941)

[4.1.2 Sprint Backlog 49](#_Toc168426942)

[4.1.3 Spécification et analyse des exigences 50](#_Toc168426943)

[4.1.4 Conception 51](#_Toc168426944)

[4.1.5 Implémentation 53](#_Toc168426945)

[4.2 Sprint 42 : Divergences Résolues 55](#_Toc168426946)

[4.2.1 Sprint Backlog 55](#_Toc168426947)

[4.2.2 Spécification et analyse des exigences 55](#_Toc168426948)

[4.2.3 Conception 56](#_Toc168426949)

[4.2.4 Implémentation 58](#_Toc168426950)

[4.3 Sprint 43 : Amélioration de l'expérience utilisateur 60](#_Toc168426951)

[4.3.1 Sprint Backlog 60](#_Toc168426952)

[4.3.2 Spécification des exigences 60](#_Toc168426953)

[4.3.3 Conception. 61](#_Toc168426954)

[4.3.4 Implémentation 62](#_Toc168426955)

[Conclusion et perspectives 64](#_Toc168426956)

[Webography 66](#_Toc168426957)

# Introduction générale

Pour améliorer continuellement la performance des entreprises, les propriétaires doivent régulièrement examiner la situation globale de l'entreprise et ses processus. En visualisant les données collectées, on peut identifier les problèmes clés, les risques et les tendances, et ainsi découvrir de nouvelles opportunités. Les données proviennent de diverses sources, comme des individus, des appareils et des applications, et augmentent chaque jour. Les organisations utilisent ces données pour créer des indicateurs de performance clés (KPI).

Cependant, ces données manquent souvent de pertinence et présentent des lacunes, rendant l'analyse difficile. Il est donc essentiel de pouvoir suivre, collecter, décoder et utiliser ces données en temps réel. Les tableaux de bord sont couramment utilisés pour visualiser ces données, fournissant un aperçu de l'activité de l'entreprise avec des graphiques et des pourcentages clés.

OBS, l'entreprise où j'ai effectué mon stage, a créé un outil pour traiter et nettoyer une grande quantité de données, analyser la qualité des données, et produire des rapports sur les divergences, présentés ensuite sur un tableau de bord. Durant mon stage, j'ai développé des indicateurs de performance et ajouté de nouvelles fonctionnalités à cet outil.

Ce rapport résume le travail accompli en cinq chapitres : le contexte général de l'organisation, l'étude préliminaire du projet, le développement des KPI, et la fonctionnalité des divergences résolues intégrée dans l'outil DQSC.

# Contexte général

Dans le premier chapitre, notre objectif est de vous présenter l'entreprise hôte et le projet sur lequel j'ai travaillé pendant mon stage de fin d'études (PFE). Nous débuterons en vous introduisant à Orange Business Services (OBS), puis nous vous donnerons un aperçu du projet, en expliquant la problématique qui l'a motivé ainsi que la solution mise en œuvre. À la fin de ce chapitre, vous aurez une compréhension globale du contexte de mon stage de fin d'études.

## Présentation de l'organisation hôte

### Orange Business Services : L'entreprise

Orange Business Services (OBS) est la branche du groupe Orange qui aide les entreprises et les organisations dans leur transition numérique, avec une présence mondiale et une approche locale. Orange était principalement connu à l'origine en tant qu'opérateur de télécommunications et fournisseur d'accès à Internet. À l'origine, c'était une société britannique de télécommunications qui est devenue une filiale du groupe France Télécom en 2000. Dans le but de s'internationaliser, le groupe France Télécom a progressivement regroupé la plupart de ses marques sous la marque Orange, et en 2013, Orange est devenu le nouveau nom du groupe.

Aujourd'hui, OBS est également présente dans d'autres domaines d'activité. De la connectivité à l'Internet des objets (IoT) et au Cloud, en passant par l'intelligence artificielle (IA), le développement d'applications et la cybersécurité, Orange Business Services accompagne les entreprises à chaque étape de la valorisation de leurs données.

Dans le cadre du plan stratégique quinquennal du groupe Orange, Engage 2025, Orange Business Services vise à réaliser plus de la moitié de son chiffre d'affaires grâce aux nouveaux services de connectivité (SD-WAN, 5G) et aux services informatiques d'ici la fin 2024.



Figure 1 – OBS logo

### Chiffres clés

Chez Orange Business Services, nous nous engageons à connecter, protéger et innover à travers le monde, afin de favoriser une croissance commerciale durable.

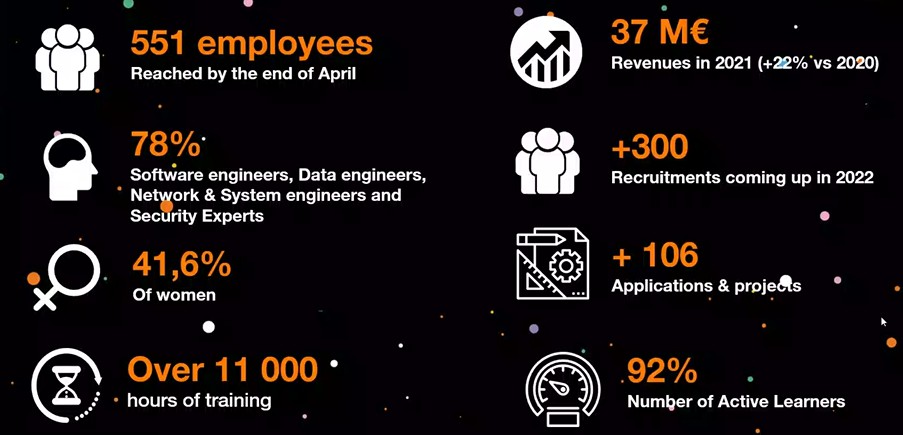


Figure 2 – OBS Morocco’s key figures

### Innovation Marketing Client

Le département CMI, abréviation pour Innovation Marketing Client, se concentre sur le client, est innovant, agile et simple. Il joue un rôle de catalyseur dans la réalisation de l'ambition de l'entreprise, qui est d'accélérer la croissance de son activité et celle de ses clients.

Son objectif principal est de soutenir la croissance et la performance d'OBS en créant un écosystème dynamique et prospère pour l'innovation axée sur les affaires.  
La mission du département CMI est de développer des solutions permettant aux clients d'OBS de se démarquer, de croître et de surpasser leurs concurrents sur leurs marchés respectifs. Pour cela, il mobilise leur expertise sectorielle ainsi que leurs ressources et actifs complets. Il se concentre sur quatre domaines clés, qui sont représentés dans le schéma suivant.

## Présentation du Projet

### Besoins Métier

La qualité des données fait référence à la mesure dans laquelle les données répondent aux attentes d'une entreprise en termes d'exactitude, de validité, d'exhaustivité et de cohérence.

En suivant un processus de gestion de la qualité des données, une entreprise peut identifier les problèmes potentiels qui affectent la qualité des données et s'assurer que les données partagées sont adaptées à une utilisation spécifique. Lorsque les données collectées ne répondent pas aux attentes de l'entreprise en termes d'exactitude, de validité, d'exhaustivité et de cohérence, cela peut avoir des conséquences négatives considérables sur le service client, la productivité des employés et les stratégies clés de l'entreprise.

Afin d'évaluer la qualité des données chez Orange, il est nécessaire de mettre en place des indicateurs clés de performance (KPI) afin de mesurer dans quelle mesure Orange atteint efficacement ses objectifs commerciaux clés. Les KPI sont utilisés pour évaluer le succès dans l'atteinte des objectifs, d'où la nécessité d'un outil de mesure approprié.

### Solution initiale (CISC) :

CISC, Customer Inventory Score Card, est une application qui mesure la qualité des données d'inventaire client en mesurant les indicateurs de performance clés (KPI) à partir de différentes sources de données du hub de données Orange, affichant les résultats dans un tableau de bord selon les filtres clients sélectionnés avec la possibilité de télécharger un rapport Excel pour un KPI, et elle utilise ETL et PowerBI pour générer ces résultats.

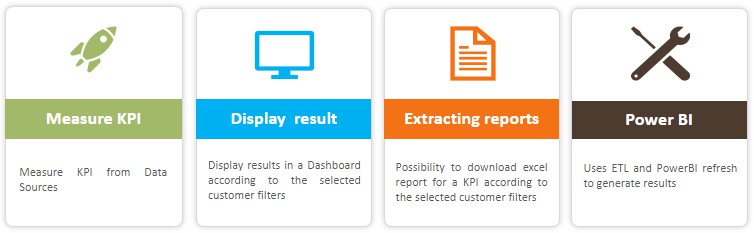


Figure 3 – Customer Inventory Score Card - CISC

### DQSC vs CISC

Data Quality Score Card, est un outil fournissant des niveaux de qualité des données et des rapports de divergence sur plusieurs sources de données. Il s'agit de l'évolution du Customer Inventory Score Card, tant en termes de portée que de technologie. Sur la figure ci-dessous sont présentées les principales améliorations clés ajoutées à DQSC, telles que le rafraîchissement quotidien qui offre des données à jour quotidiennement, il est également basé sur des technologies Big Data telles qu'Apache Spark, ce qui lui donne la capacité de traiter une grande quantité de données et donc une exécution de tâches plus rapide d'environ 6 minutes par rapport à 2 jours pour l'ensemble du processus dans CISC, en plus de l'intégration de nombreux filtres qui ajoutent plus de flexibilité d'investigation et de facilité d'utilisation pour les utilisateurs.

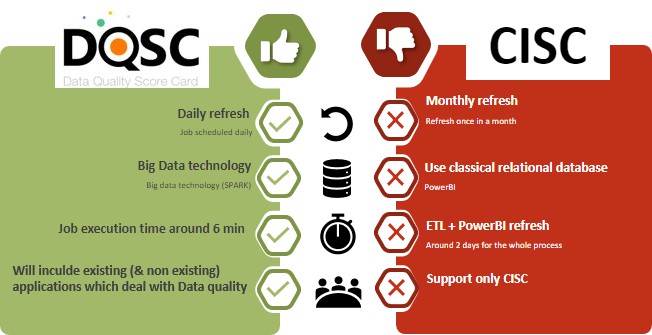


Figure 4 – DQSC vs CISC

### Objectifs de DQSC

Les objectifs de DQSC sont les suivants :

* Mesurer la conformité des données d'inventaire clients fournies via l'Inventaire Fédéré afin de garantir qu'elles soient complètes et cohérentes.
* Respecter les accords de niveau de service (SLA) concernant la qualité des données d'inventaire que nous avons avec certains clients.
* Obtenir une vue d'ensemble de la conformité des données clients, à la fois par client et par produit.

### Problématique

Depuis la sortie de CISC, de nombreux intervenants et partenaires d'Orange se sont intéressés à l'idée qui se cache derrière et aux opportunités qu'elle peut offrir. Par conséquent, de nombreuses autres catégories d'indicateurs clés de performance (KPI) ont été ajoutées, ce qui a entraîné une charge de travail et une quantité de données plus importantes. Ainsi, l'utilisation de bases de données relationnelles classiques et de tâches en cascade n'est plus adaptée. De plus, davantage de sources de données ont été ajoutées, au point où nous avons fait nos premiers pas dans le domaine du big data.

C'est ainsi que DQSC est né, pour remédier aux limitations de CISC, grâce à son architecture robuste et à la pile technologique sur laquelle il repose.

### Solution

DQSC est un outil qui fournit des KPI de données provenant de l'écosystème Orange. Ces KPI sont catégorisés en plusieurs types, chaque type ayant ses propres rapports définis par certaines règles métier.

Il aide à identifier les divergences, à calculer les métriques des KPI, à documenter les règles métier appliquées aux données et à fournir des informations sur la procédure de correction aux propriétaires d'entreprise afin de résoudre les divergences et d'améliorer la qualité des données.

Une divergence fait référence à une incohérence dans les données, par exemple si un routeur a deux adresses IP ou s'il est lié à deux clients différents, etc. Ces types d'anomalies sont considérés comme des divergences.

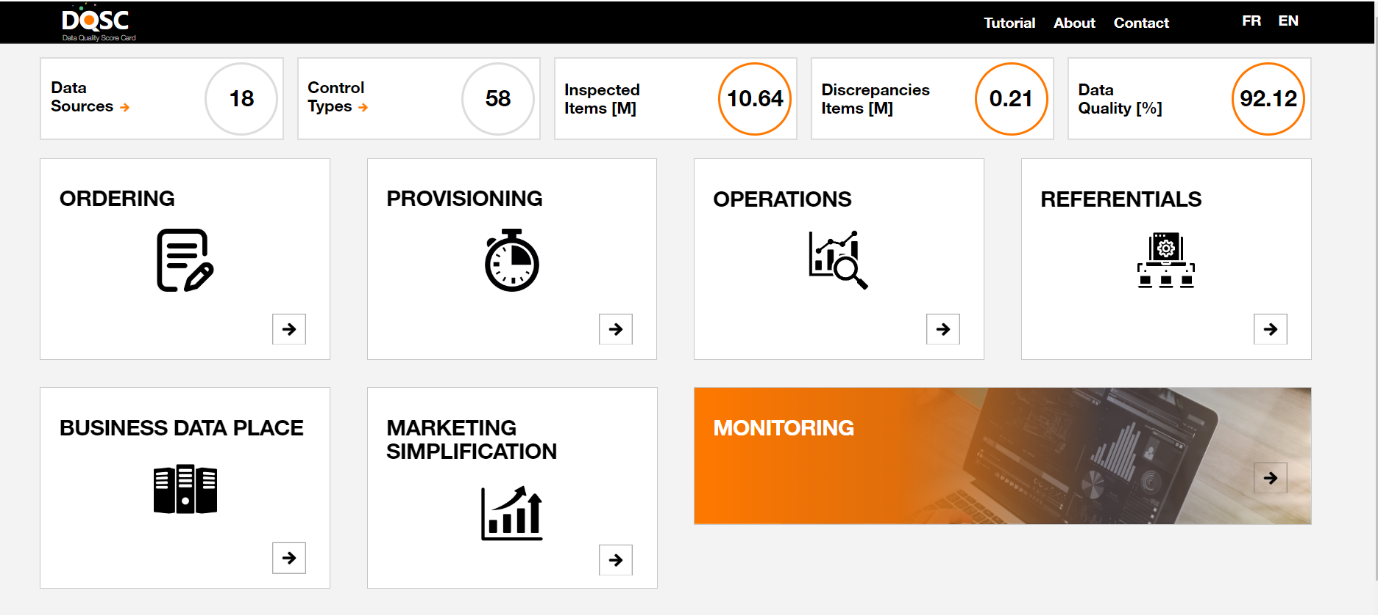


Figure 5 – DQSC home page

**Conclusion**

Cette section avait pour objectif de présenter l'entreprise hôte et le projet afin de vous donner une vue d'ensemble du contexte général du stage. Dans le prochain chapitre, nous entrerons dans les détails concernant l'aspect fonctionnel du projet ainsi que la méthodologie de gestion de projet qui a été suivie tout au long.

# Étude préliminaire

L'analyse des besoins est une étape essentielle pour la réalisation de tout projet informatique. Dans ce chapitre, nous présentons donc les besoins de notre projet. Nous débutons en identifiant les acteurs et leurs principales fonctionnalités, puis nous passons aux spécifications fonctionnelles et non fonctionnelles. Cela nous permet d'établir un extrait du carnet de produit et une planification initiale comprenant plusieurs itérations. Enfin, nous mentionnons les user stories sur lesquelles j'ai travaillé au cours de ces itérations.

## Spécification des Besoins

### Identification des Principaux Acteurs

Le diagramme de contexte est une étape intermédiaire entre les spécifications et le diagramme de cas d'utilisation. Il ne fait pas partie des 13 diagrammes UML, mais il est construit à partir des outils du langage. Son objectif est de définir les différents acteurs qui interagiront avec le système.

Dans notre système, il y a quatre acteurs distincts : FINV, FBO, E2E et SLA chaque acteur peut accéder à ses propres KPI en fonction des sources de données spécifiques à son domaine. En revanche, l'administrateur (ADMIN) a accès à tous les KPI et métriques disponibles dans le système.

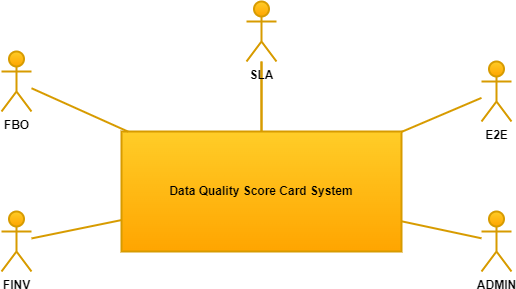


Figure 7 – Context diagram

### Exigences Fonctionnelles

* Mesure et calcul des KPI à partir de plusieurs sources de données.
* Affichage des métriques sur un tableau de bord, classées par catégorie de KPI.
* Filtrage des KPI par client.
* Génération d'un rapport Excel pour un KPI spécifique en fonction des filtres clients sélectionnés.
* Génération d'un rapport Excel pour les divergences résolues depuis le mois dernier.
* Restreindre l'accès en fonction des rôles.

### Exigences Non Fonctionnelles

Les exigences non fonctionnelles, également connues sous le nom de contraintes non fonctionnelles, sont liées à la mise en œuvre et à l'interopérabilité de la solution logicielle. Ces besoins sont identifiés soit par le client lui-même, soit par l'équipe de développement. Les besoins non fonctionnels de ce projet sont énumérés comme suit :

**Performance** : Cette exigence, l'une des plus courantes en matière de NFR/QAR (Non-Functional Requirements/Quality Attribute Requirements), stipule que tous les systèmes doivent être conçus et construits pour offrir des performances acceptables. La performance peut être affectée négativement par des facteurs tels qu'un grand nombre d'appels d'API, de mauvaises procédures de mise en cache, etc. Il est donc important de prendre en compte des domaines tels que la latence, la charge et l'utilisation des ressources.

**Disponibilité élevée** : Il s'agit d'une exigence générale qui englobe d'autres NFR/QAR tels que la fiabilité et la résilience. Pour intégrer cette exigence dans un système, il convient d'éviter les points de défaillance uniques, de mettre en place des délais d'attente et des disjoncteurs, et d'effectuer des déploiements sans interruption.

**Sécurité** : Cette exigence revêt une importance particulière lorsque des informations sensibles, telles que des détails personnels ou des données financières, sont traitées. La sécurité doit également garantir la confidentialité et l'authentification des informations, afin de les protéger de manière optimale.  
  
**Scalabilité** : Ce terme désigne la capacité du système à s'adapter à des volumes plus importants, qu'il s'agisse d'utilisateurs, de débit ou de données, au fil du temps. La scalabilité inclut également des NFR tels que l'élasticité, c'est-à-dire la capacité à s'adapter rapidement en fonction des besoins. De nos jours, la scalabilité peut être plus facilement réalisée grâce aux solutions modernes basées sur le cloud, qui offrent l'infrastructure nécessaire pour s'adapter automatiquement aux besoins.

### Product backlog

L'approche Scrum commence par l'énumération des besoins du client afin de créer product backlog. Ce carnet est une liste des user stories qui comprend tous les éléments nécessaires pour le produit. Il est la seule source de besoins pour toute modification éventuelle du produit.

Ci-dessous, vous trouverez product backlog spécialement conçu pour ce projet.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | User Story | Tasks | sprint |
| DQSC-304 | As Business User I need to have a control on the active orders In order to identify those that are not billed | develop Spark/Scala job  Interceptor  API  Dashboard | SP-39 |
| DQSC-38 | As a Business Owner, I need to compare the values of the ordered IP bandwidth and polled IP bandwidth in order to detect and report the different ones. | develop Spark/Scala job  Interceptor  API  Dashboard | SP-40 |
| DQSC-380 | As an intern working on the SLA DQ KPI monitoring and reporting project, I need to have a new report in order to create SLA data quality controls. | develop Spark/Scala job  Interceptor  API  Dashboard | SP-41 |
| DQSC-51 | As PO, I would like to have the solved discrepancies since last month. | [Interceptor] schedule job to compute resolved discrepancies  [API] generate excel report for solved discrepancies  [Dashboard]add the solved column to that permit downloading solved  discrepancies report | SP-42 |
| DQSC-400 | As a Business Owner, I need to filter by new customer fields : Scope, Segmentation Client, and Billing Account | Add new filters for all Spark/Scala jobs  Interceptor  API  Dashboard | SP-42 |
| DQSC-429 | As PO, I request to calculate the total number of solved discrepancies | [Dashboard] Add the total number of solved discrepancies next to the  excel report. | SP-43 |

### Table 1 – Product backlog

### Identification des Cas d'Utilisation Principaux

Les diagrammes de cas d'utilisation servent à décrire les différentes interactions possibles entre un acteur et le système.



Figure 8 – General use cas diagram

## Gestion de Projet

### SCRUM

Le principe fondamental de la méthode SCRUM consiste à développer un logiciel de manière itérative et incrémentale tout en maintenant une liste claire des demandes d'évolution ou de corrections à implémenter. Grâce à des livraisons fréquentes à intervalles réguliers, le client reçoit un logiciel fonctionnel à chaque itération. Au fur et à mesure que le projet avance, le logiciel devient de plus en plus complet et offre de nouvelles fonctionnalités. Ce processus se répète jusqu'à ce que nous obtenions un produit final complet. La méthode SCRUM est appliquée à des développements itératifs avec un rythme presque constant de deux à quatre semaines en moyenne ‘trois semaines dans notre cas.

La figure ci-dessous illustre le cycle de vie de la méthodologie SCRUM.

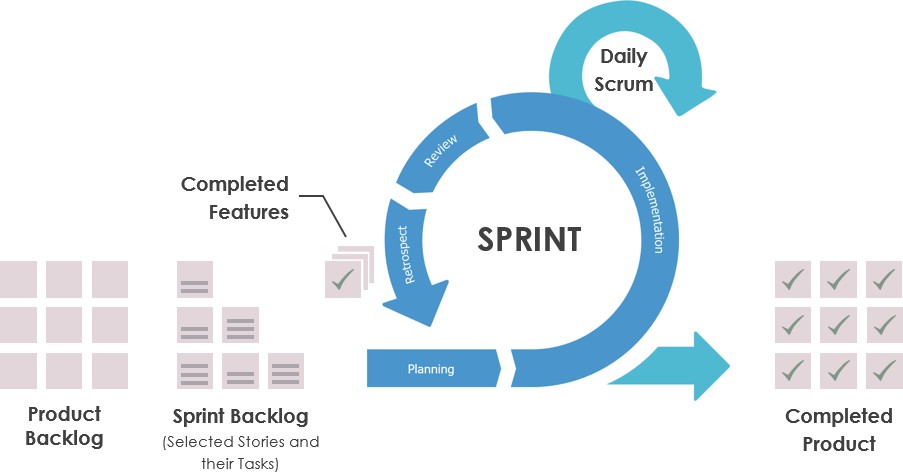


Figure 9 – SCRUM life cycle

### Kanban board

La méthodologie Kanban repose sur la transparence totale du travail et la communication en temps réel de la capacité. Ainsi, le tableau Kanban est la source principale d'information pour l'équipe. Chaque membre de l'équipe travaille en accord avec ce tableau, qui sert à visualiser le travail et optimiser le flux.  
  
Que le tableau soit physique ou numérique, son objectif est de garantir la visualisation du travail de l'équipe, la standardisation du flux de travail, ainsi que l'identification et la résolution immédiate des obstacles et dépendances.  
  
Le tableau Kanban de base comprend six étapes : Analyse, Prêt pour le développement, En cours, En attente, En revue et Livraison. L'exemple du 42e sprint de ce projet est illustré dans la figure ci-dessous.

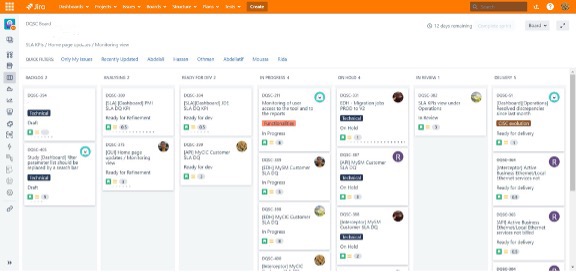


Figure 10 – Kanban board

### Jira et Confluence

**Jira** : Pour les équipes qui pratiquent les méthodologies agiles, Jira Software propose des tableaux Scrum et Kanban prêts à l'emploi. Ces tableaux sont des centres de gestion des tâches où les tâches sont associées à des flux de travail personnalisables. Ils offrent une transparence et une visibilité sur l'état de chaque tâche au sein de l'équipe. Les fonctionnalités de suivi du temps et les rapports de performance en temps réel permettent aux équipes de surveiller leur productivité au fil du temps.

**Confluence** : Confluence est un outil wiki de collaboration utilisé pour faciliter la collaboration et le partage efficace des connaissances au sein des équipes. Avec Confluence, nous pouvons capturer les exigences du projet, attribuer des tâches à des utilisateurs spécifiques et gérer plusieurs calendriers simultanément grâce à l'add-on Team Calendars.

Confluence et Jira se complètent mutuellement. Jira est excellent pour aider votre équipe à planifier et suivre le travail de projet. L'ajout de Confluence vous permet d'organiser toutes vos idées, votre contenu et vos fichiers, donnant ainsi vie à votre vision.

### Workflow GitLab

En ce qui concerne la gestion du code et le versioning, voici notre stratégie utilisant GitLab. Nous avons trois branches principales :

**develop** : C'est notre branche de développement principale, où toutes les fonctionnalités développées sont intégrées. Toutes les modifications dans cette branche sont révisées et fusionnées avec la branche test.

**test** : Cette branche est extraite/fusionnée à partir de la branche develop dans le but de tester les fonctionnalités (test d'intégration et test fonctionnel). Avant qu'une modification ne puisse être déployée en production, elle doit passer des tests QA afin d'assurer la stabilité du code.

**master** : C'est notre branche de pré-production et de production. Elle est extraite/fusionnée à partir de la branche test. Cette branche doit être stable en tout temps et ne permet pas les check-ins directs.

Pendant le cycle de développement, nous utilisons également plusieurs branches de support :

**- feat\_\*** : Ces branches de fonctionnalité sont utilisées pour développer de nouvelles fonctionnalités pour les prochaines versions. Elles peuvent être dérivées de la branche develop et doivent être fusionnées dans cette même branche.  
  
**- bug\_\*** : Ces branches de correction de bugs sont utilisées pour résoudre les problèmes identifiés lors des tests. Elles peuvent être dérivées de la branche test et doivent être fusionnées dans les branches test et develop.  
  
**- hot\_\*** : Ces branches de correction urgente sont utilisées pour résoudre immédiatement des problèmes en production. Elles peuvent être dérivées de la branche master et doivent être fusionnées dans les branches master, test et develop.

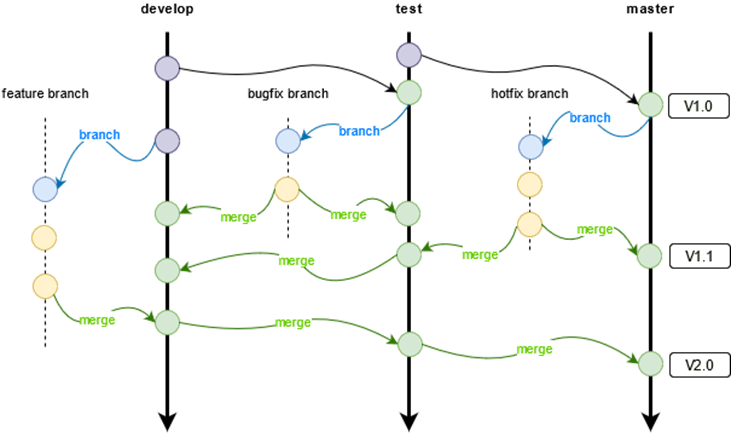


Figure 11 – Branching strategy

### Planification du Projet

#### Sprints planning

Le tableau ci-dessous présente la planification des sprints utilisée dans ce projet, qui comprend cinque sprints, chacun d'une durée de trois semaines.  
  
J'ai rejoint l'équipe au cours du 39eme sprint du projet. Pendant ces Cinque sprints, j'ai travaillé sur l'ajout de nouvelles fonctionnalités et de KPI aux projets existants.

|  |  |  |
| --- | --- | --- |
| Release | Sprint | Time period |
| Release 1.0 | sprint 39 | 05/03/2024 to 26/03/2024 |
| sprint 40 | 27/04/2024 to 16/04/2024 |
| Release 2.0 | sprint 41 | 17/04/2024 to 08/05/2024 |
| sprint 42 | 09/05/2024 to 30/05/2024 |
| sprint 43 | 31/05/2024 to 20/06/2024 |

Table 2 – sprints planning

## Architecture du Projet

Le système est composé de plusieurs éléments, comme illustré dans la figure 13, qui fonctionnent ensemble pour accomplir différentes tâches. Les composants principaux comprennent :

**KPI Job** : Cette partie du système est dédiée au traitement des données massives (big data). Nous y déployons nos jobs Spark/Scala qui nettoient, filtrent, transforment et calculent les données de l'EDH en utilisant des règles métier spécifiques à chaque KPI. Ces jobs sont programmés pour s'exécuter quotidiennement.  
  
**EOG** : "End of Game" (EOG), également connu sous le nom d'exportateur rapide, est un composant de l'EDH qui peut exporter les résultats des jobs Spark/Scala au format ORC vers des fichiers JSON.

**Interceptor** : Il s'agit d'une couche d'accès aux données et d'une implémentation de la responsabilité unique pour accéder à la base de données Mongo et récupérer des données. En outre, il charge les données de l'EDH (résultats de l'EOG) vers le CAAS en les poussant dans notre base de données MongoDB, et les transforme du format JSON en documents stockés dans la collection KPI correspondante.

**Api** : Ce composant comprend tous les points de terminaison exposés sur le tableau de bord. Il utilise l'interceptor pour récupérer les données de la base de données, les traite en effectuant des calculs, et peut également générer des rapports Excel.

**Dashboard** : Il s'agit de l'interface qui permet de visualiser le travail réalisé en arrière-plan pour calculer la qualité des données. Cet outil de visualisation nous permet de présenter les KPI et métriques calculés aux utilisateurs finaux.  
  
La figure 13 illustre l'architecture du projet.

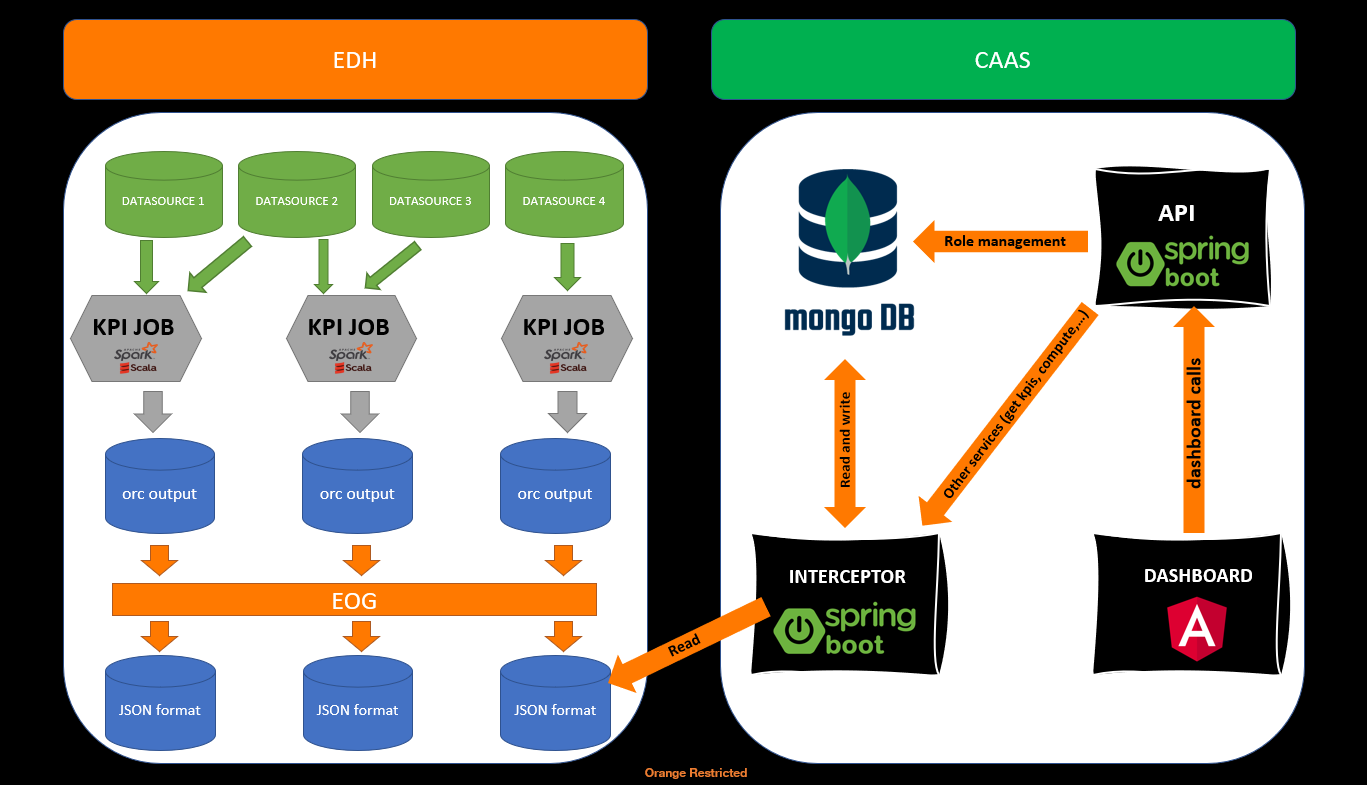


Figure 13 – Project architecture

## Stack Technique

### Spark/Scala

Apache Spark est un cadre open source conçu pour exécuter des applications d'analyse de données à grande échelle sur des ordinateurs en cluster. Il permet de traiter des charges de travail d'analyse et de traitement de données à la fois en temps réel et par lots. D'un autre côté, Scala est un langage de programmation compilé et exécuté sur la machine virtuelle Java (JVM). Il améliore la productivité, l'évolutivité et la fiabilité des applications. En résumé, Scala est considéré comme le langage principal pour interagir avec le moteur Spark Core.

### Spring Web-Flux

Il s'agit d'un cadre web entièrement non-bloquant, basé sur des annotations et construit sur Project Reactor. Il permet de créer des applications réactives au niveau de la couche HTTP. Les systèmes réactifs sont conçus avec un modèle d'architecture réactive qui privilégie l'utilisation de composants flexibles, évolutifs et faiblement couplés. Ils sont également conçus en gardant à l'esprit la résolution des pannes, afin de garantir que la majeure partie du système continuera de fonctionner même en cas de défaillance d'une partie.

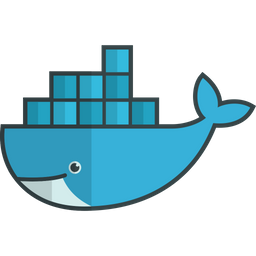
### MongoDB

MongoDB est un système de base de données conçu pour respecter les garanties ACID traditionnelles. Ces garanties sont similaires à celles offertes par les systèmes de bases de données relationnelles, qui privilégient la cohérence par rapport à la disponibilité. En revanche, les bases de données NoSQL sont principalement conçues en se référant à la philosophie BASE, qui préfère la disponibilité à la cohérence.

### Angular

Angular est une plateforme de développement construite sur TypeScript. Elle offre un cadre de travail basé sur des composants pour créer des applications web évolutives. De plus, elle intègre également une collection de bibliothèques couvrant une grande variété de fonctionnalités, telles que le routage, la gestion des formulaires et la communication client-serveur. Enfin, Angular propose une suite d'outils pour aider les développeurs à développer, construire, tester et mettre à jour leur code.  
  
Avec Angular, vous disposez d'une plateforme qui convient aussi bien aux projets individuels qu'aux applications d'entreprise. De plus, Angular est conçu pour faciliter les mises à jour, vous permettant ainsi de bénéficier des derniers développements avec un minimum d'efforts.

### Docker



Docker permet de créer et d'exécuter des applications dans des conteneurs isolés. Ces conteneurs offrent une sécurité et une isolation qui permettent de les exécuter simultanément sur un même hôte. Les conteneurs sont légers et contiennent tout le nécessaire pour exécuter l'application, ce qui signifie que vous n'avez pas besoin de vous appuyer sur les installations existantes sur l'hôte. Vous pouvez facilement partager des conteneurs avec d'autres personnes et être sûr qu'ils obtiendront le même conteneur fonctionnant de la même manière.

Docker propose des outils et une plateforme pour gérer le cycle de vie de vos conteneurs:  
  
- Vous pouvez développer votre application et ses composants de support en utilisant des conteneurs.

- Les conteneurs deviennent l'unité de distribution et de test de votre application.  
- Lorsque vous êtes prêt, vous pouvez déployer votre application dans votre environnement de production, que ce soit sur un centre de données local, un fournisseur de cloud ou une combinaison des deux, en tant que conteneur ou service orchestré.

### Kubernetes



Kubernetes, également connu sous le nom de K8s, est un système open-source qui automatise le déploiement, le dimensionnement et la gestion d'applications conteneurisées. Il rassemble les conteneurs qui composent une application en unités logiques pour faciliter leur gestion et leur découverte.  
Kubernetes définit un ensemble de blocs de construction ("primitives") qui, ensemble, fournissent des mécanismes pour déployer, maintenir et dimensionner des applications en fonction de métriques CPU, mémoire ou personnalisées. Kubernetes est peu couplé et extensible pour s'adapter à différents types de charges de travail. Les composants internes, les extensions et les conteneurs exécutés sur Kubernetes utilisent l'API Kubernetes. La plateforme contrôle les ressources de calcul et de stockage en définissant des objets qui peuvent ensuite être gérés en tant que tels.

### Redis



Redis est une base de données et un cache rapide en mémoire, écrit en C et optimisé pour la vitesse. Il est open source, sous une licence BSD. Le nom de Redis provient de "REmote DIctionary Server".

Redis est souvent appelé un serveur de structures de données car ses types de données principaux ressemblent à ceux que l'on trouve dans des langages de programmation. Il inclut des chaînes de caractères, des listes, des dictionnaires (ou hachages), des ensembles et des ensembles triés. Redis propose également de nombreuses autres structures de données et fonctionnalités pour le comptage approximatif, la géolocalisation et le traitement des flux.

**Conclusion**

Ce chapitre a présenté l'architecture du projet, ainsi que les exigences spécifiées, à la fois fonctionnelles et non fonctionnelles. Nous avons également inclus un diagramme général des cas d'utilisation et le carnet de produits, qui recense les fonctionnalités du système. De plus, nous avons détaillé la pile technologique utilisée et la méthode de gestion de projet que nous avons suivie pour atteindre mon objectif de stage.

Dans le prochain chapitre, nous présenterons la première fonctionnalité ajoutée qui a été développée lors du sprint 39.

# Release 1.0

Le focus de ce chapitre porte sur la combinaison des Sprints 39 et 40. Nous passerons en revue les deux premiers KPI développés, qui appartiennent à différentes catégories, Profitabilityet Finv, regroupés sous la même vue 'Opérations'. Nous commencerons par présenter chaque KPI individuellement, en expliquant les règles métier qui les sous-tendent, puis nous passerons directement aux différents composants, de l'EDH aux données affichées dans le tableau de bord.

## Sprint 39 : Ethernet services not billed KPI

### Présentation du KPI

Le KPI "Ethernet services not billed KPI" est un indicateur qui permet de détecter les services Ethernet d'entreprise et les services Ethernet locaux qui sont inclus dans l'ensemble de données FINV mais qui n'ont pas été facturés après plus de trois mois depuis leur livraison. Cette détection des divergences permet d'éviter une perte de revenus et contribue à améliorer la rentabilité d'Orange.

### Sprint Backlog

Le tableau 3 répertorie les user stories à accomplir pendant ce sprint pour atteindre l'objectif du KPI "Ethernet services not billed KPI".

|  |  |  |
| --- | --- | --- |
| **ID** | **User story** | **Estimate** |
| DQSC-363 | **[BigData]** As Business User I need to have a control  on the active orders In order to identify those that are not billed. | 8 |
| DQSC-364 | **[Interceptor]** As Business User I need to have a control on the active orders In order to identify those that are  not billed. | 3 |
| DQSC-365 | **[Api]** As Business User I need to have a control on  the active orders In order to identify those that are not billed. | 3 |
| DQSC-304 | **[Dashboard]** As Business User I need to have a control on the active orders In order to identify those that are  not billed. | 1 |

Table 3 – Sprint 39 backlog

Il est divisé en quatre briques : EDH, Interceptor, API et tableau de bord. Nous examinerons chacun d'entre eux en détail dans la section suivante.

### Spécification et analyse des exigences

Chaque indicateur clé de performance (KPI) est défini par deux types de règles métier. Le premier type de règle consiste à extraire les éléments inspectés en fonction de certains filtres. À partir de ces eléments inspectés, nous appliquons d'autres filtres pour obtenir les éléments contenant des anomalies, que nous appelons les éléments de divergence.

Pour ce KPI, les règles métiers sur les éléments inspectés sont définies de la manière suivante :

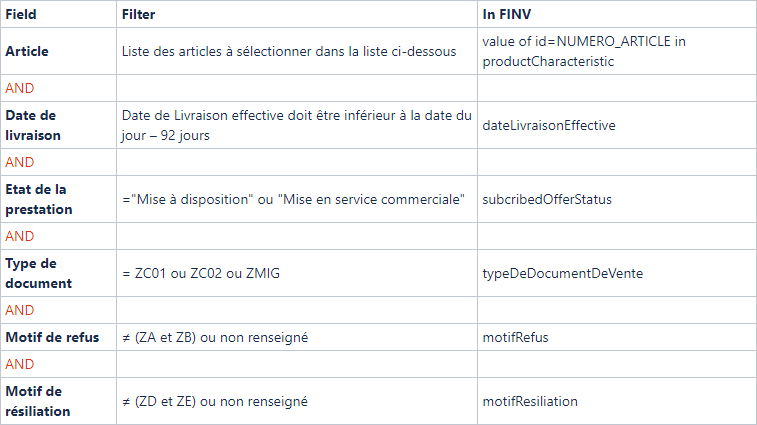


Table 4 – Inspected items for profitability KPI

Pour appliquer ces règles métier, nous transformons ces critères en filtres que nous appliquons aux données provenant de FINV.

Une fois les éléments inspectés obtenus, la deuxième étape consiste à obtenir les éléments de divergence. Cela est réalisé en appliquant le filtre "Date de mise en facturation est vide".

### Conception

#### Diagram de classe

Méthode de conception par fabrique : En programmation orientée objets, le modèle de méthode de fabrique est un modèle de création qui utilise des méthodes de fabrique pour résoudre le problème de la création d'objets sans avoir à spécifier la classe exacte de l'objet à créer. Cela se fait en créant des objets via une méthode de fabrique, qui peut être spécifiée dans une interface et implémentée par des classes enfants, ou implémentée dans une classe de base et éventuellement surchargée par des classes dérivées, plutôt qu'en appelant un constructeur. Dans notre cas, nous utilisons ce modèle pour la génération de rapports. La figure 15 ci-dessous présente le diagramme de classes utilisé pour la création de nouveaux rapports en fonction de leur type, en appliquant le modèle de méthode de fabrique.

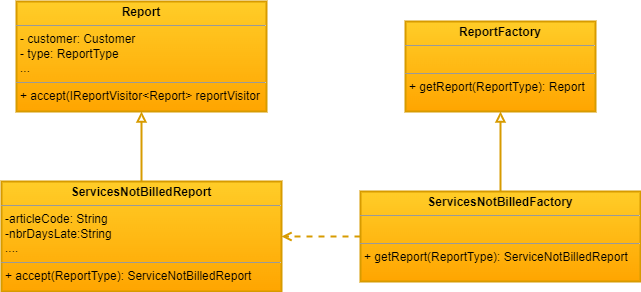


Figure 15 – Sprint 39 class diagram

#### Diagramme de séquence : "Télécharger le rapport de divergence"

La figure 16 présente le diagramme de séquence pour le cas d'utilisation "Télécharger le rapport de divergence".

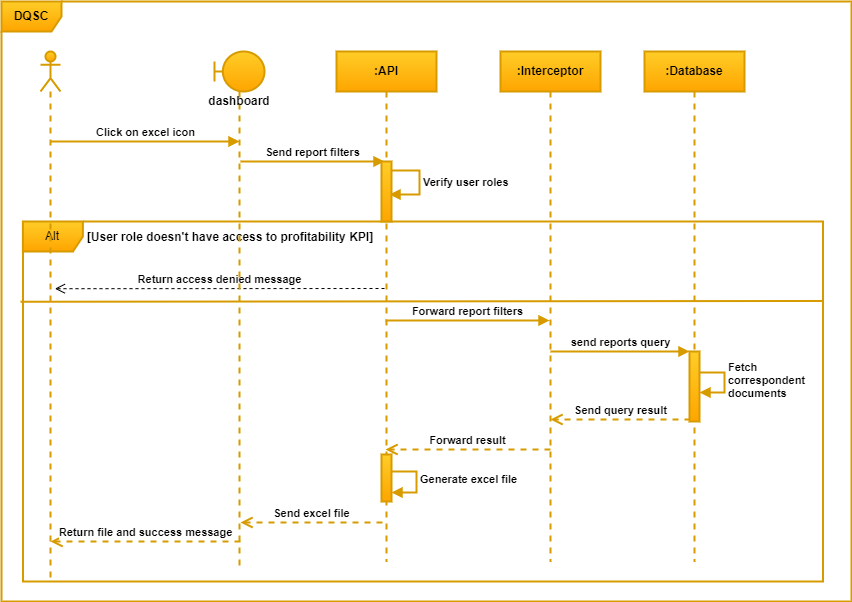


Figure 16 – Sequence diagram : ’Download discrepancy report’

### Implémentation

La figure ci-dessous présente le KPI développé dans la catégorie d’operations . Si vous le souhaitez, vous pouvez sélectionner une date différente de celle affichée par défaut. Une fois la date sélectionnée, les filtres client associés à cette date sont chargés et vous pouvez commencer à filtrer les données. La section de qualité des données globales affiche les métriques pour tous les KPI résultants, y compris le KPI d’operation mis en œuvre.

Des détails supplémentaires sont présentés dans la ligne intitulée "Active Business Ethernet/Local Ethernet services not billed". Vous pouvez vérifier la qualité des données de cette catégorie et accéder au lien de la procédure de correction. Nous fournissons également plusieurs approches pour résoudre les divergences. Vous avez également la possibilité de télécharger le rapport complet au format Excel.

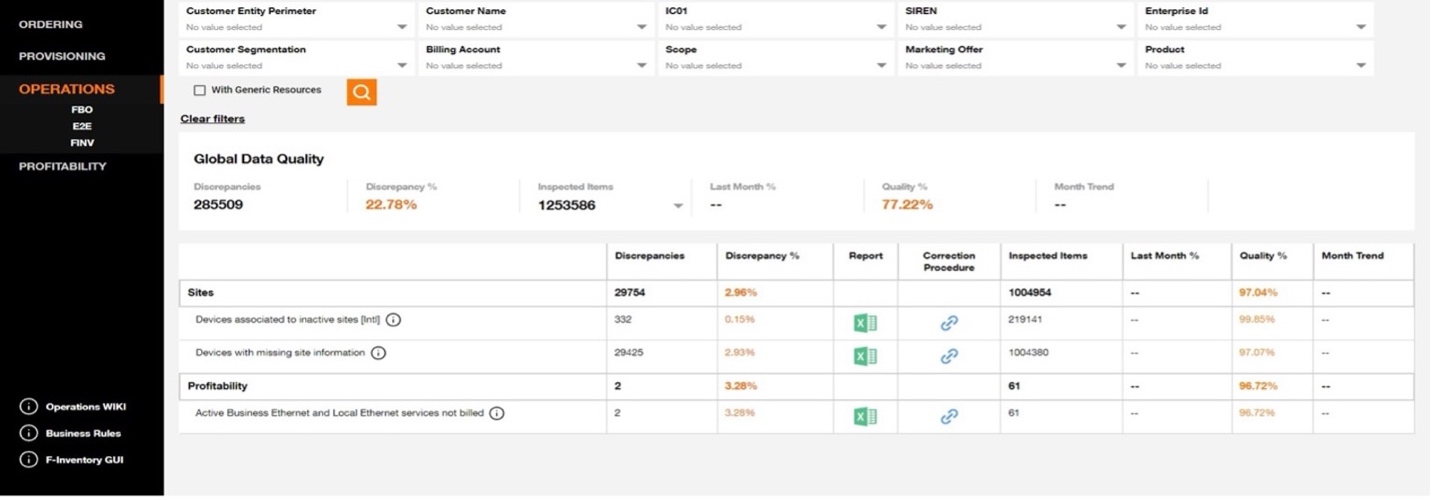
****

Figure 17 – Operations KPI in the dashboard

Voici un extrait du rapport :

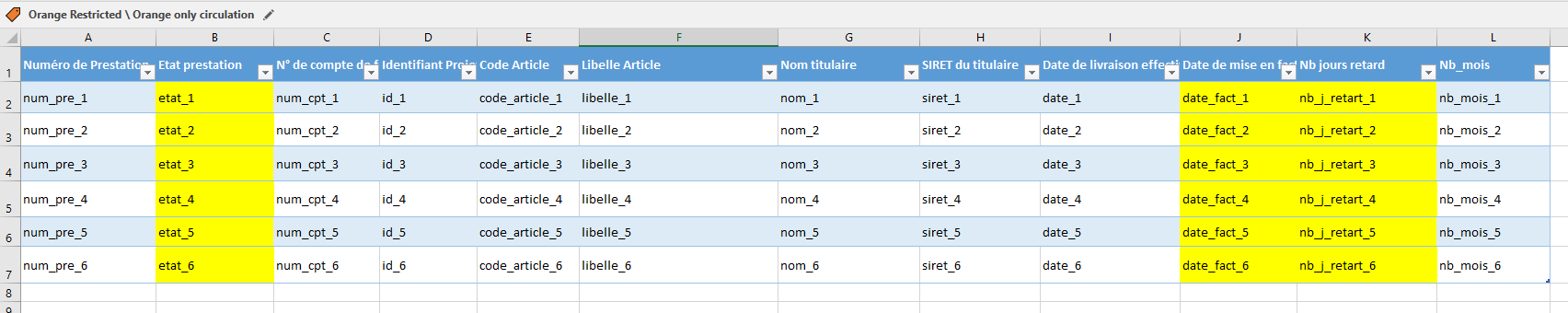


Figure 18 – Active Business Ethernet/Local Ethernet services not billed report

## Sprint 40: Different Ordered and Polled IP Bandwidth KPI

### Présentation du KPI

Le KPI " Different Ordered and Polled IP Bandwidth " est utilisé pour comparer les valeurs de la bande passante IP commandée et de la bande passante IP sondée, dans le but de détecter et de signaler les divergences. Ce rapport met en évidence les écarts entre la bande passante IP commandée et celle obtenue par sondage.

### Sprint Backlog

Le tableau 5 répertorie les user stories à accomplir pendant ce sprint pour atteindre l'objectif du KPI " Different Ordered and Polled IP Bandwidth ".

|  |  |  |
| --- | --- | --- |
| **ID** | **User Story** | **Estimate** |
| DQSC-194 | **[Big Data]** As a Business Owner, I need to compare the values of the ordered IP bandwidth and polled IP  bandwidth in order to detect and report the different ones. | 8 |
| DQSC-195 | **[API]** As a Business Owner, I need to compare the values  of the ordered IP bandwidth and polled IP bandwidth in order to detect and report the different ones. | 3 |
| DQSC-196 | **[Interceptor]** As a Business Owner, I need to compare the values of the ordered IP bandwidth and polled IP  bandwidth in order to detect and report the different ones.. | 3 |
| DQSC-38 | **[Dashboard]** As a Business Owner, I need to compare the values of the ordered IP bandwidth and polled IP  bandwidth in order to detect and report the different ones. | 1 |

Table 5 – Sprint 40 backlog

### Spécification et analyse des exigences

Étant donné que chaque KPI est défini par deux types de règles métier, nous devons d'abord examiner les éléments requis pour la deuxième étape. Ensuite, nous extrairons et calculerons ceux qui présentent les anomalies spécifiées dans la deuxième partie.

**Éléments inspectés :**



Table 6 – Inspected items for different IP bandwidth KPI

**Éléments de divergence :**

La bande passante du service VPN est l'un des éléments d'un des composants de l'offre installée, comme indiqué dans la figure 19. Dans FINV, le lien entre eux est "IO\_is\_composed\_of\_IO". La raison derrière ce type de relation est que la source d'origine était une base de données orientée graphe.

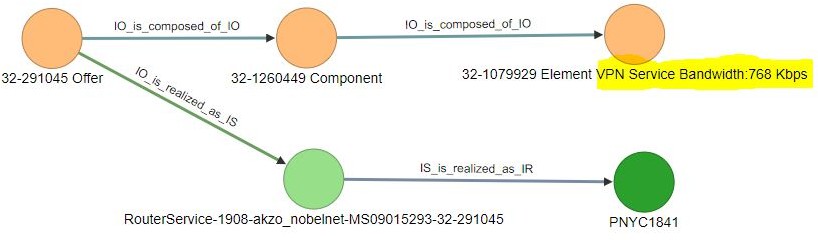


Figure 19 – Resources relationships

L'élément concerné est la " VPN Service Bandwidth " et il se trouve dans le composant appelé " Business VPN Service Plug ". Voici un exemple de champ de caractéristique de production dans le tableau suivant :

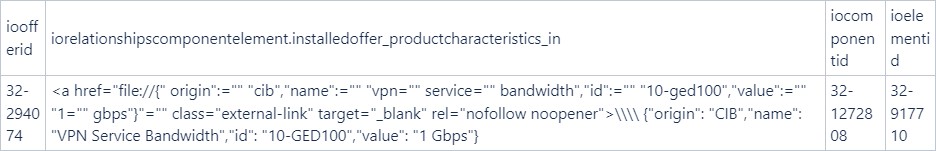


Table 7 – Example of a product characteristics field

L'élément " VPN Service Bandwidth " porte l'identifiant "10-GED100" pour le VPN professionnel d'entreprise. La valeur de la caractéristique du produit "10-GED100" (qui doit être convertie en kbps\*) ne correspond pas à la bande passante locale LLI ±3% (la valeur de la caractéristique du produit "10-GED100" en kbps n'est pas comprise entre 97% et 103% de la bande passante locale LLI).

### Conception

#### diagramme de séquence

Dans cette section, nous présentons le diagramme de séquence du Spark/Scala job qui est responsable du traitement des données. Tout d'abord, le JobLauncher est déclenché pour démarrer l'exécution du DiffBandwidthJob. Ce dernier effectue le traitement en commençant par le Reader, qui lit les ensembles de données utilisés depuis l'EDH. Ensuite, il applique les règles métier et les filtres définis dans le Pipeline. Une fois que le pipeline est terminé et que le traitement est effectué, les résultats sont écrits sous forme de fichiers ORC et stockés dans l'EDH. La figure 20 présente le diagramme de séquence du traitement des données effectué dans le EDH.

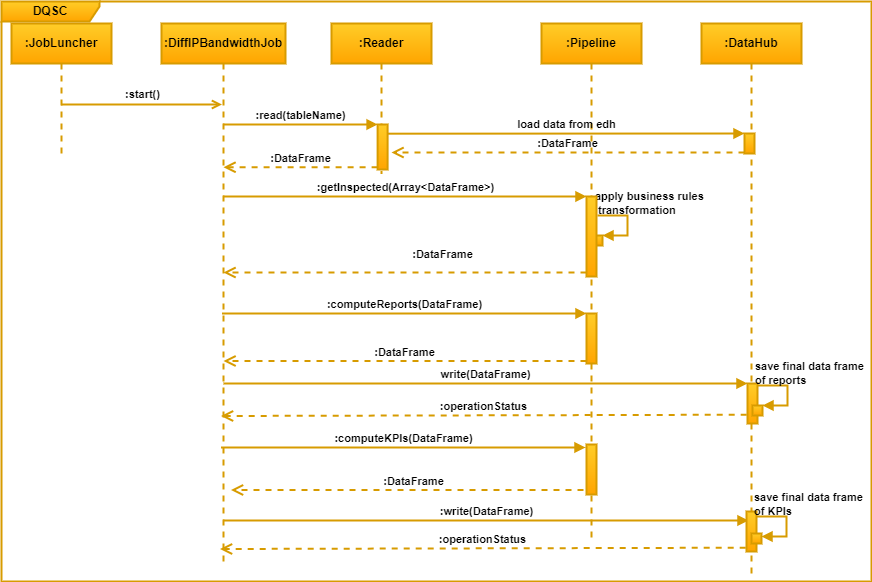


Figure 20 – Sprint 40 sequence diagram

#### Transformation workflow

Après avoir spécifié les besoins métier et les filtres à appliquer aux données, la prochaine étape consiste à convertir la logique métier en un flux de données à implémenter en utilisant Spark/Scala. La figure 21 illustre ce flux de transformations.

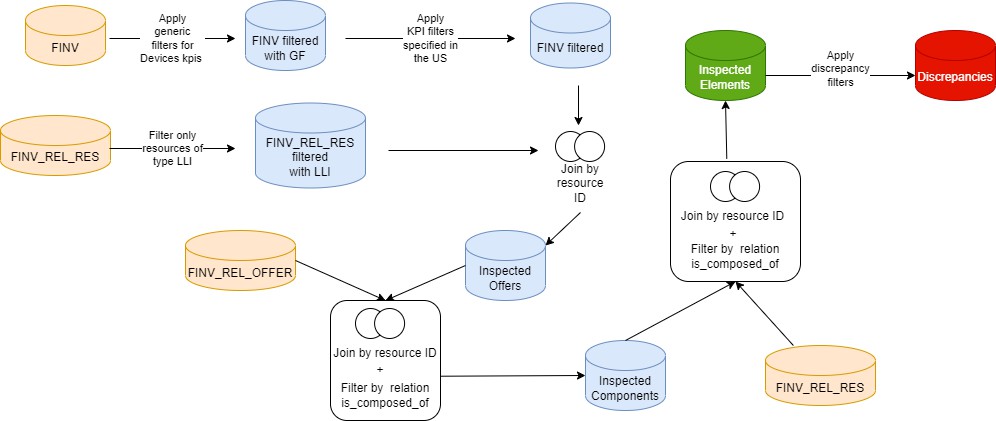


Figure 21 – Data transformation flow

### Implémentation

La figure 22 présente le KPI développé " Different ordred and polled ip bandwidth " dans la vue FINV. Il est possible de filtrer par date d'extraction et filtres client afin de vérifier la qualité des données pour ces filtres.

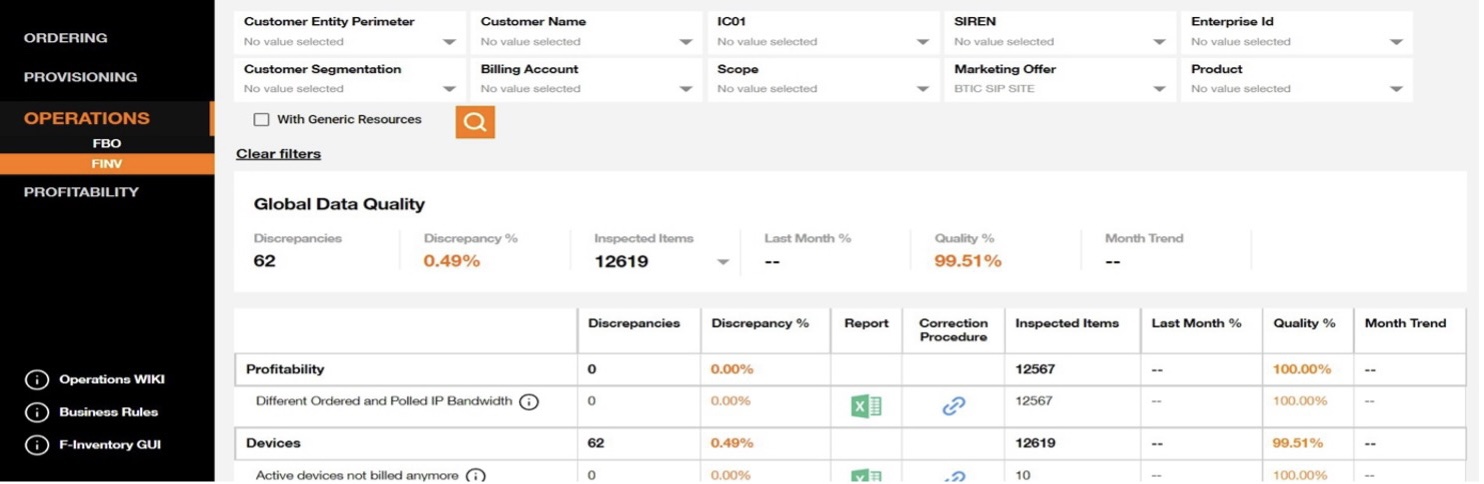


Figure 22 – Different Ordered and Polled IP Bandwidth KPI in the dashboard

Le bloc "Qualité des données globales" fournit des métriques pour presque tous les KPI affichés dans le tableau. Vous pouvez y approfondir les détails de chaque KPI et télécharger le rapport si nécessaire.

**Conclusion**

Dans ce chapitre, j'ai mis en évidence les différentes étapes suivies pour implémenter les user stories du backlog des sprints 39 et 40. L'objectif de ces sprints était de mettre en œuvre les KPI suivants : " Active Business Ethernet/Local Ethernet services not billed KPI " et " Different ordered and polled IP bandwidth ". Pendant chaque sprint, nous avons commencé par une vue d'ensemble du KPI, spécifié les règles métier à appliquer sur la base de données pour obtenir les divergences, puis effectué une analyse et une conception, pour finir par l'implémentation.

Le prochain chapitre présentera les derniers KPI implémentés, sur lesquels nous nous concentrerons davantage sur la génération de métriques. Nous examinerons une fonctionnalité très importante : la capacité pour les utilisateurs d'extraire et de visualiser les divergences résolues depuis le mois dernier, ainsi que l'intégration supplémentaire de filtres client.

# Release 2.0

L'une des réalités fondamentales du développement de produits est que les gens ne comprennent souvent pas vraiment ce dont ils ont besoin avant de commencer à utiliser le produit. La revue de sprint offre régulièrement aux propriétaires d'entreprise et aux parties prenantes l'opportunité d'apprécier ce qui a été construit jusqu'à présent et de collaborer avec l'équipe Scrum pour déterminer les prochaines étapes les plus précieuses.

Dans ce chapitre, nous examinerons d'abord le KPI développé pour les SLA, puis nous nous pencherons sur l'exploration des nouvelles exigences assemblées qui apportent une valeur ajoutée au produit. Ces exigences sont motivées par le besoin de résoudre les divergences du mois en cours, ainsi que par la création de nouveaux filtres permettant aux utilisateurs du produit d'obtenir les données qu'ils souhaitent de manière plus efficace et pertinente.

## Sprint 41 : SLA data quality KPI

service-level agreement (SLA) est un engagement entre un fournisseur de services et un client. Cet accord définit les aspects spécifiques du service, tels que la qualité, la disponibilité et les responsabilités, qui sont convenus entre le fournisseur de services et l'utilisateur. Afin de garantir le respect constant des SLA par Orange, il est essentiel de développer des indicateurs de performance clés (KPI) pour mesurer la qualité des données des SLA.

### Présentation du KPI

Le rapport de divergence de ce KPI indique les dispositifs pour lesquels au moins un des champs obligatoires de l'inventaire client n'est pas rempli. Ce KPI a été créé en raison de l'impact que peuvent avoir ces divergences, notamment des pénalités de SLA si le niveau de qualité minimum de 95 % n'est pas atteint.

### Sprint Backlog

|  |  |  |
| --- | --- | --- |
| ID | User story | Estimate |
| DQSC-389 | [BigData] As an intern working on the SLA DQ KPI  monitoring and reporting project, I need to have a new report in order to create SLA data quality controls. | 8 |
| DQSC-388 | [Interceptor] As an intern working on the SLA DQ KPI monitoring and reporting project, I need to have a  new report in order to create SLA data quality controls. | 3 |
| DQSC-387 | [Api] As an intern working on the SLA DQ KPI  monitoring and reporting project, I need to have a new report in order to create SLA data quality controls. | 3 |
| DQSC-380 | [Dashboard] As an intern working on the SLA DQ KPI  monitoring and reporting project, I need to have a new report in order to create SLA data quality controls. | 1 |

Table 8 – Sprint 41 backlog

EDH, Interceptor, API, et Dashboard sont les quatre briques qui composent le KPI développé ; nous examinerons chacune d'elles en détail dans la section suivante.

### Spécification et analyse des exigences

Étant donné que chaque KPI est défini par deux types de règles métier, nous devons d'abord examiner les éléments requis pour la deuxième étape. Ensuite, nous extrairons et calculerons ceux qui présentent les anomalies spécifiées dans la deuxième partie.

|  |  |
| --- | --- |
|  | Inspected |
| IR\_irType | In [Router, Switch, Wireless access point, Wireless LAN controller] |
| AND |  |
| IR\_provisioning\_status | In [ONLINE, Active] |

Table 9 – SLA KPI : Inspected items

La qualité globale est déterminée en additionnant les erreurs trouvées dans les champs inspectés. Une fois cette somme obtenue, nous pouvons calculer le pourcentage de qualité globale par rapport aux champs inspectés. Ensuite, nous passons à la deuxième étape, qui consiste à identifier les divergences en utilisant les filtres indiqués dans la figure ci-dessous :

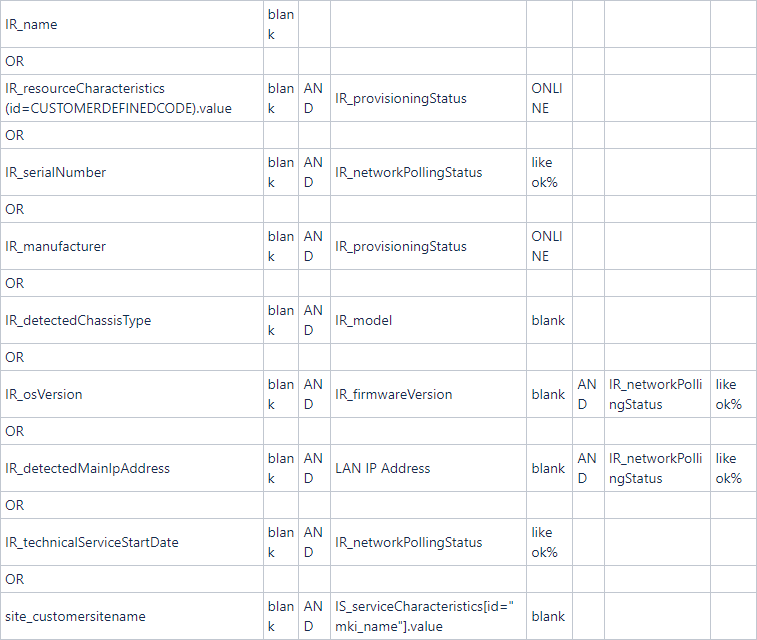


Table 10 – SLA KPI : Discrepancy items

### Conception

Dans cette section, nous allons examiner les métriques du KPI SLA, comment elles sont générées, intégrées dans le tableau de bord et incluses dans le rapport Excel.

#### Diagramme de séquence : "Générer les métriques KPI"

La figure ci-dessous présente le diagramme de séquence pour la génération des métriques KPI. Comme le traitement de la qualité des données est principalement effectué par l'API, nous détaillerons ici sa mise en œuvre.

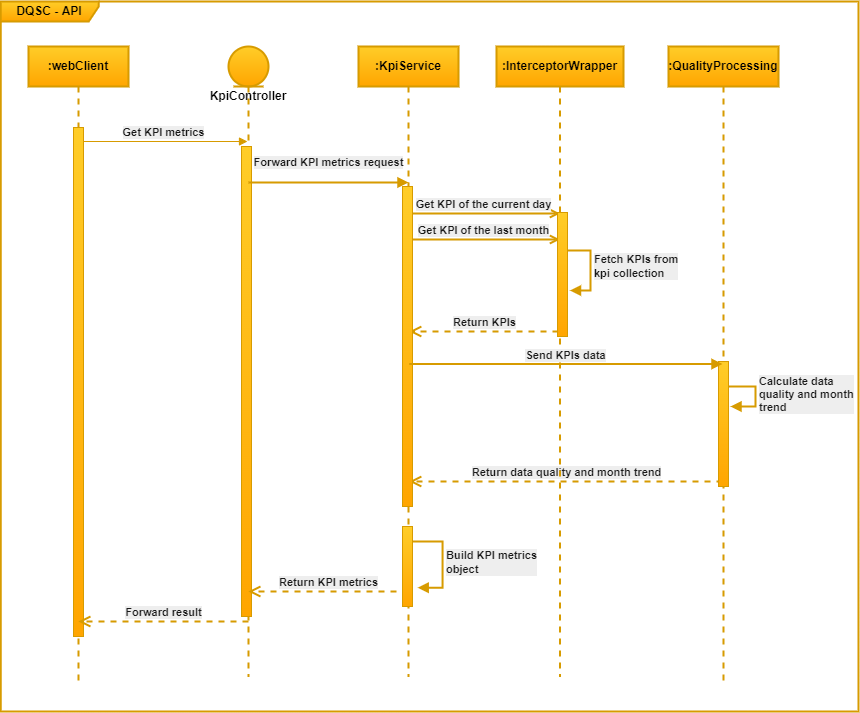


Figure 23 – Sequence diagram : ”Generate KPI metrics”

#### Diagramme de class

La figure 24 illustre le diagramme de classes qui met en évidence les différentes classes qui collaborent pour générer les métriques d'un KPI.

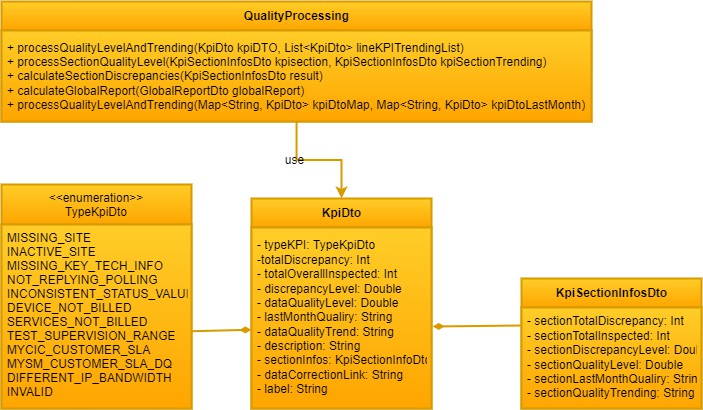


Figure 24 – Sprint 41 class diagram

### Implémentation

Dans le tableau de bord ci-dessous, vous pouvez voir les métriques de qualité des données du SLA. Comme indiqué, il y a le nombre d'éléments présentant des divergences, ainsi que le nombre d'éléments inspectés. En utilisant ces deux données, nous calculons la qualité des données et le pourcentage de divergences.  
  
La tendance mensuelle est calculée en comparant les données du mois précédent à la date actuelle. Enfin, vous trouverez le lien vers la procédure de correction et le rapport. Ce rapport résume les données affichées dans le tableau de bord.

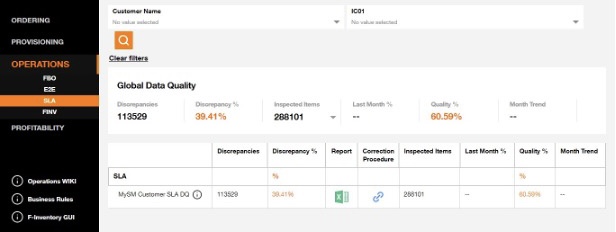


Figure 25 – SLA KPI in the dashboard

Dans le rapport téléchargé, nous avons inclus une feuille qui présente un résumé de la qualité des données, avec les filtres correspondants qui ont été sélectionnés. Cette feuille a aussi été ajoutée à tous les KPI pour simplifier le partage de cette vue d'ensemble.

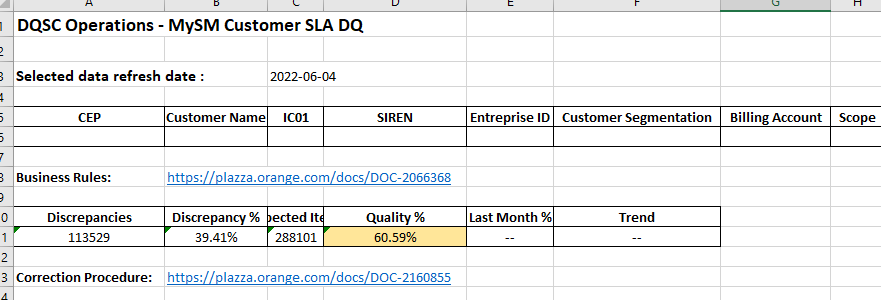


Figure 26 – SLA Data Quality report

## Sprint 42 : Divergences Résolues

### Sprint Backlog

|  |  |  |
| --- | --- | --- |
| ID | User story | Estimate |
| DQSC-376 | [Interceptor] As PO, I would like to have the resolved  discrepancies since last month. | 5 |
| DQSC-377 | [Api] As PO, I would like to have the resolved  discrepancies since last month. | 3 |
| DQSC-51 | [Dashboard] As PO, I would like to have the resolved  discrepancies since last month. | 1 |
| DQSC-400 | As a Business Owner, I need to filter by new customer  fields : Scope, Segmentation Client, and Billing Account. | 5 |

Table 11 – Sprint 42 backlog

Pour cette fonctionnalité, les changements se concentreront principalement sur l'intercepteur et l'API, car nous disposons déjà des données des travaux précédents. À présent, il s'agit davantage de comparer et structurer les données plutôt que de les extraire.

### Spécification et analyse des exigences

Pour mettre en place la fonctionnalité des divergences résolues, les exigences suivantes doivent être respectées :

- Une nouvelle colonne " Solved " doit être ajoutée après la colonne " Month Trend " à la fois dans les Résultats globaux de la qualité des données et dans le tableau des KPI.

- Lorsque le rapport est téléchargé, le nom doit contenir la date sélectionnée dans le titre du rapport, formatée "YYYYMMDD".

- Sous la colonne " Solved ", il doit y avoir un lien vers un rapport fournissant les divergences résolues depuis le début du mois de la date de rafraîchissement des données sélectionnées. Ce rapport affiche les divergences résolues par rapport au dernier jour du mois précédent, du 1er jour du mois jusqu'au dernier jour du mois. S'il n'y a pas de données pour le dernier jour du mois précédent, le dernier jour du mois précédent avec des données disponibles doit être utilisé comme référence.

- Le rapport doit fournir la date à laquelle la divergence a cessé de figurer dans le rapport de divergence associé, ainsi que le nom du rapport de divergence, les champs de filtre et l'ID des éléments qui ont disparu.

### Conception

#### diagramme de classe

Ci-dessous, vous pouvez voir la figure 27, qui présente le diagramme de classes utilisé pour créer le rapport des divergences résolues.

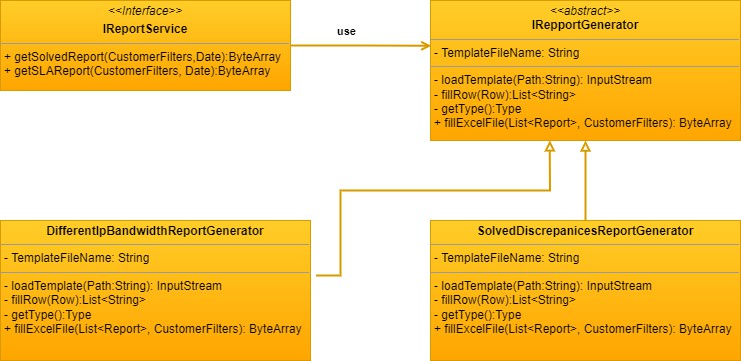


Figure 27 – Sprint 42 class diagram

#### Diagramme de séquence : "Rapport des divergences résolues"

Le diagramme de séquence pour le cas d'utilisation "Rapport des divergences résolues" est présenté dans la figure 28.

Tout d'abord, nous devons mettre en place une tâche quotidienne planifiée qui calcule les divergences résolues en comparant les données du jour présent avec celles du jour précédent. Les ressources manquantes sont celles qui ont été résolues jusqu'à présent.

Ensuite, nous collectons les divergences résolues pour une date de récupération donnée (le mois entier) et des filtres client, puis nous générons un rapport Excel à partir des enregistrements résultants que nous envoyons à l'utilisateur. La réponse de la requête est mise en cache dans le "cluster Redis" pour des raisons de performance.

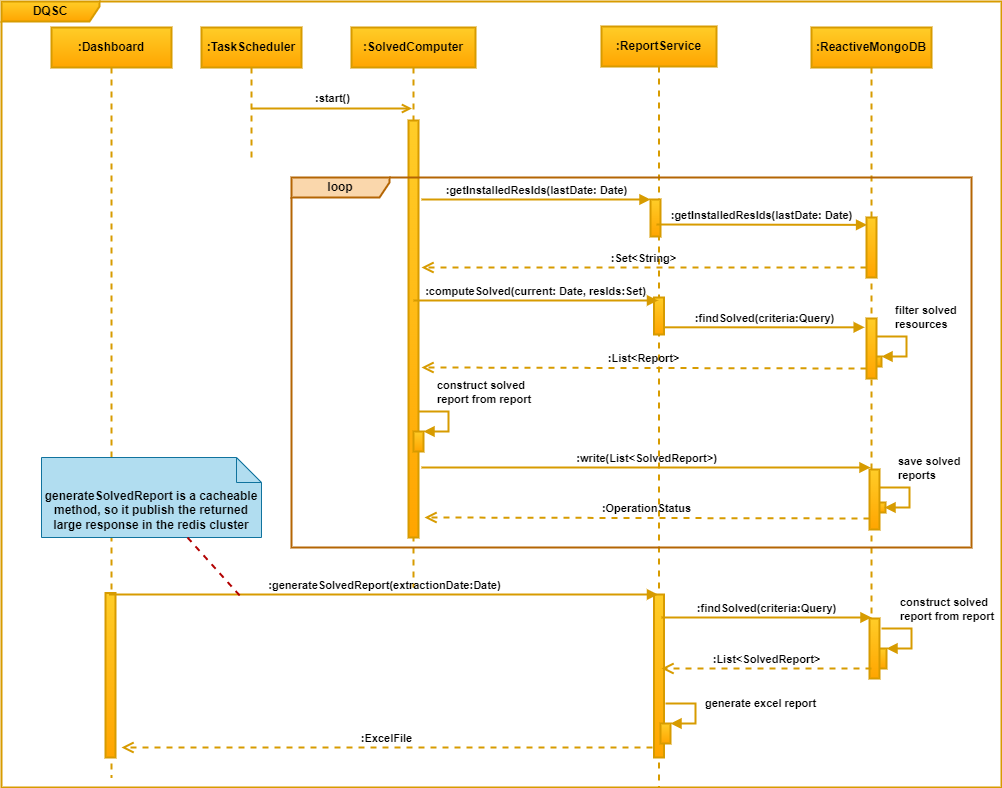


Figure 28 – Sequence diagram : "Solved discrepancies report"

### Implémentation

La figure ci-dessous illustre l'ajout de fonctionnalité dans le tableau de bord : pour chaque KPI, un rapport est maintenant disponible, montrant les divergences résolues depuis le mois dernier. De plus, nous offrons également un rapport global regroupant tous les KPI.

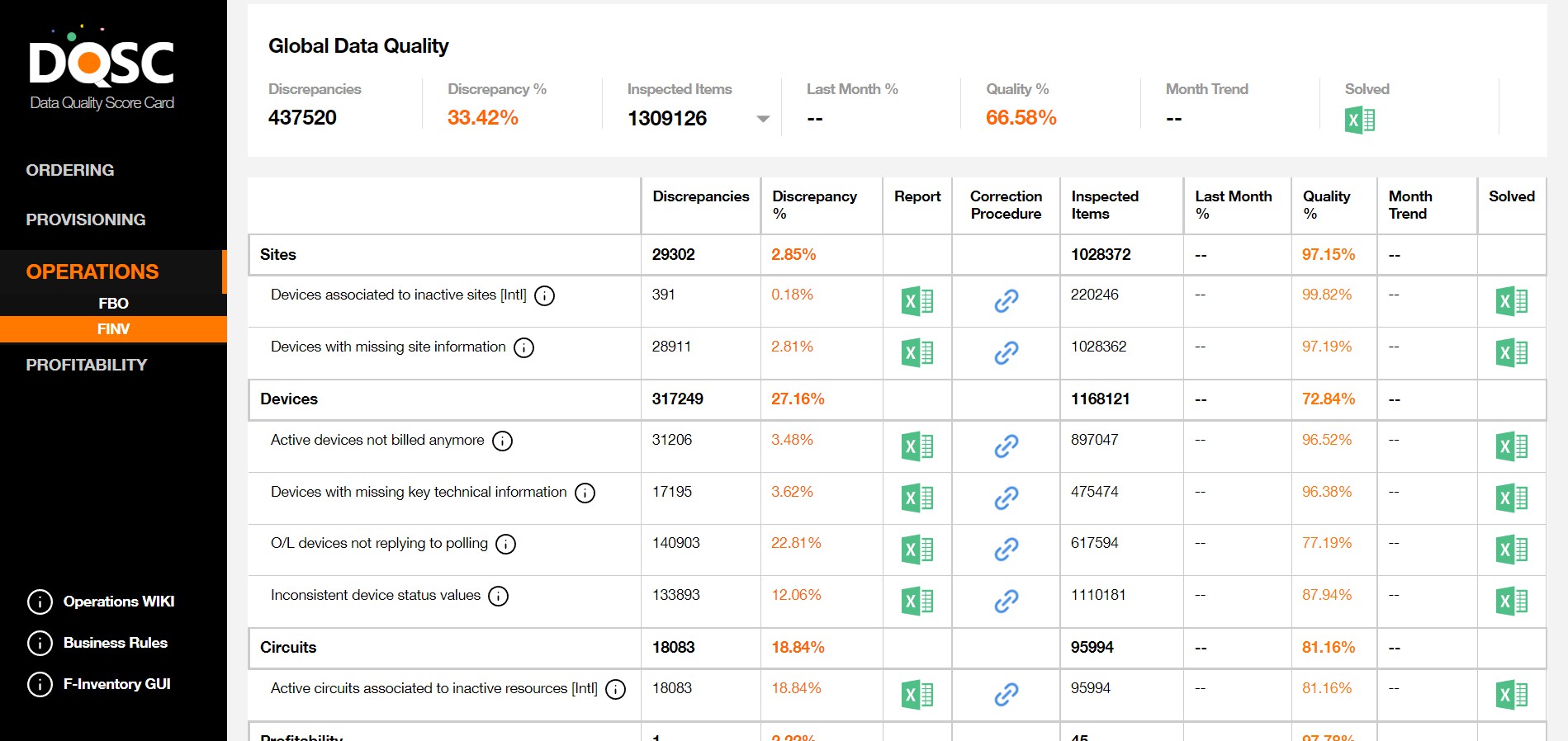


Figure 29 – Resolved discrepancies integrated into the dashboard

Un échantillon du rapport est présenté ci-dessous :

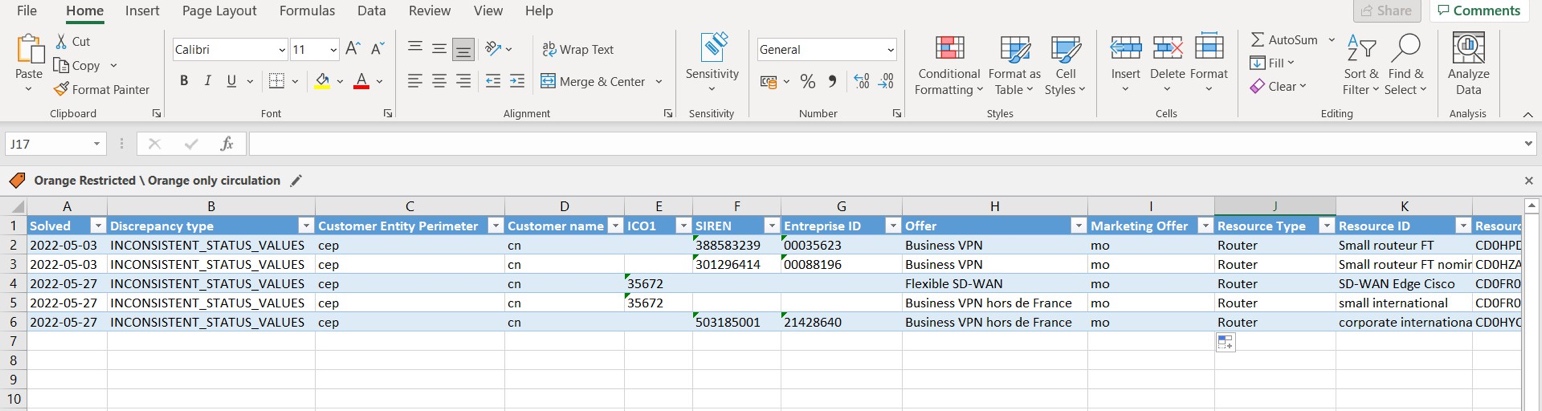


Figure 30 – Resolved discrepancies report

## Sprint 43 : Amélioration de l'expérience utilisateur

### Sprint Backlog

Le tableau ci-dessous répertorie les user stories auxquelles je me suis engagé pour ce sprint. L'objectif est d'améliorer la fonctionnalité de résolution des divergences identifiées lors du sprint précédent, ainsi que de mettre en œuvre les modifications demandées concernant la visualisation sur le tableau de bord.

|  |  |  |
| --- | --- | --- |
| ID | User story | Estimate |
| DQSC-429 | [Dashboard] As PO, I request to calculate the total  number of solved discrepancies and to display it together with the clickable excel report. | 1 |
| DQSC-430 | [Api] As PO, I request to calculate the total number of solved discrepancies and to display it together with the  clickable excel report. | 3 |
| DQSC-391 | [Dashboard] As PO, I request some GUI changes after  sprint 42 demo | 2 |

Table 12 – Sprint 43 backlog

### Spécification des exigences

Afin de mettre en œuvre les user stories, il est essentiel de respecter les critères d'acceptation définis par le PO. En ce qui concerne les changements d'interface utilisateur, nous avons une liste de critères d'acceptation qui guide la mise en œuvre:

- Réorganisez l'ordre des colonnes du tableau global et du tableau KPI en suivant l'ordre indiqué dans la pièce jointe ;

- Mettez à jour les champs de toutes les pages de vue d'ensemble des rapports de divergences pour qu'ils aient le même ordre que dans la vue en ligne, avec "Résolu ce mois-ci" contenant uniquement le total (DQSC-429 ajoute le nombre total de résolutions ce mois-ci) ;

- Centrez verticalement le texte de chaque ligne, et centrez également horizontalement le texte des colonnes "Divergence %" et "Qualité %" ;

- Dans la barre verticale, affichez les noms des sous-vues sous Opérations dans l'ordre alphabétique ;

- Lorsque je sélectionne une valeur SIREN dans le filtre de vue des opérations, si sous IC01 il n'y a pas de valeur, remplacez "Aucun élément trouvé !" par "Aucune valeur trouvée" ;

- Lorsque je sélectionne une valeur IC01 dans le filtre de vue des opérations, si sous SIREN il n'y a pas de valeur, remplacez "Aucun élément trouvé !" par "Aucune valeur trouvée".

### Conception.

Le diagramme de séquence suivant met en évidence la transaction système pour calculer le nombre total de divergences résolues pour chaque KPI, afin de l'afficher à côté de l'icône Excel, pour rendre cette information disponible sans avoir besoin de télécharger le rapport.

À cette fin, l'API obtient toutes les divergences résolues en appelant l'intercepteur pour fournir les données à son avantage, puis regroupe ces divergences résolues par KPI pour en compter le nombre total. Enfin, cette information est injectée dans le KPIDTO transféré au tableau de bord.

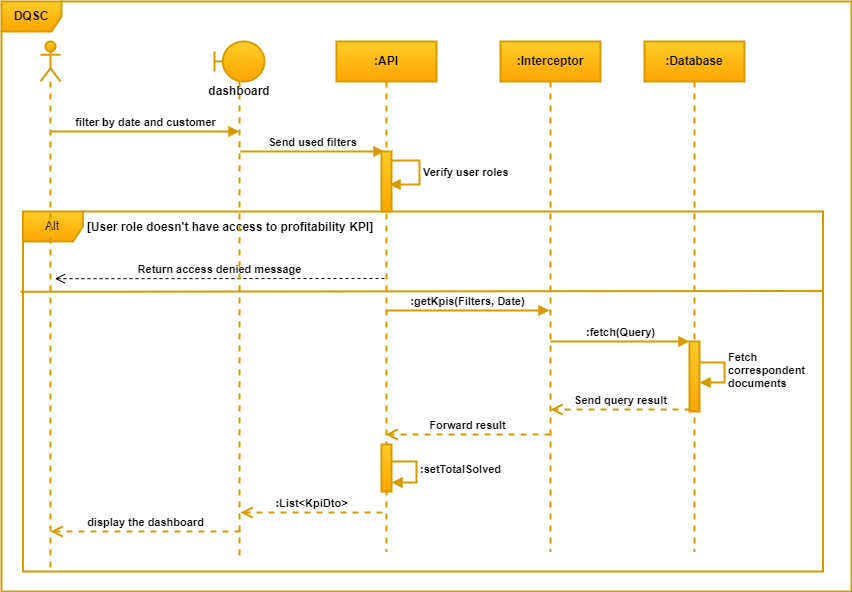


Figure 31 – Sprint 43 sequence diagram

### Implémentation

La figure 32 présente l'implémentation résultante de ce sprint : désormais, nous pouvons voir le total résolu à côté de l'icône Excel, et les changements d'interface utilisateur demandés ont été également mis en œuvre.

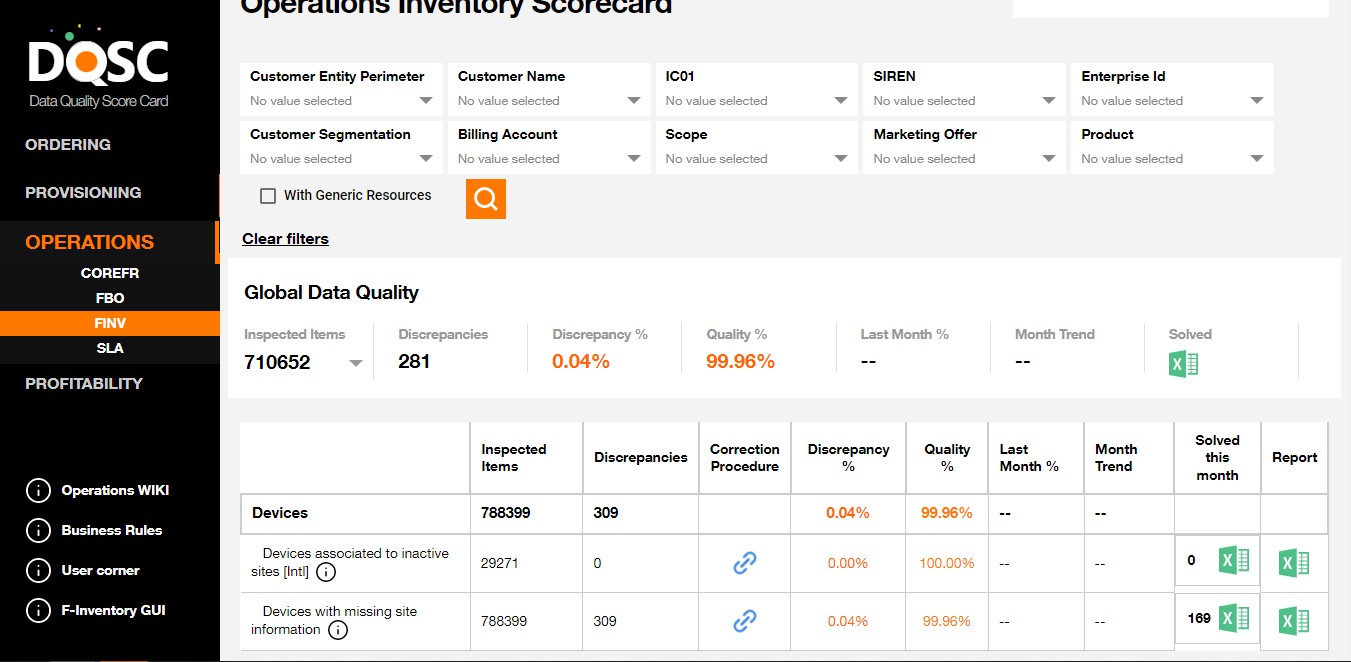


Figure 32 – Sprint 43 demo

**Conclusion**

Tout au long de ce chapitre, j'ai détaillé les différentes étapes pour mettre en œuvre les user stories des sprints 41, 42 et 43. Ces sprints avaient pour objectifs respectivement d'intégrer le KPI "SLA DQ", de mettre en place la fonctionnalité des divergences résolues, et enfin, d'améliorer l'expérience utilisateur et de mettre en œuvre les changements demandés dans la partie visualisation de l'application.

# Conclusion et perspectives

À l'origine, les technologies de l'information étaient principalement axées sur le traitement de grandes quantités de données. Cependant, l'avènement du Big Data a amené une formalisation de l'évolution des volumes, de la vitesse et de la diversité des données, générant ainsi une valeur ajoutée.

Jusqu'à présent, le Big Data est essentiellement une approche stratégique. Nous devons nous demander comment créer de la valeur ajoutée à partir des données que nous avons ou que nous pouvons collecter dans mon entreprise. En d'autres termes, que pouvons-nous construire à partir de nos données ? La stratégie consiste à donner de la valeur à vos données et à les transformer en profit. C'est cette stratégie qui fera fonctionner le Big Data au sein d'une entreprise.  
  
Chez Orange Business Services, nous collectons des données en respectant la vie privée et la réglementation à partir de flux financiers, de services clients, de sites web, de réseaux sociaux, d'objets connectés, et bien plus encore. Nous analysons ces données pour les transformer en informations utiles. DQSC, un outil qui met en œuvre ce processus de données, est le projet sur lequel j'ai travaillé lors de mon stage PFE. J'ai développé trois indicateurs clés de performance (KPI) pour mesurer la qualité des données et j'ai intégré certaines fonctionnalités, comme nous l'avons vu tout au long de ce rapport, tant sur le plan fonctionnel que technique.

En ce qui concerne les perspectives, nous envisageons d'inclure des KPI supplémentaires et d'intégrer davantage d'autres applications de calcul de la qualité des données afin de les regrouper toutes. De plus, nous cherchons à améliorer en permanence les performances de l'application, ce qui est essentiel à son succès.

Dans l'ensemble, le résultat de ce stage est positif. J'ai eu l'opportunité de travailler sur un projet très intéressant, utilisant des technologies modernes, puissantes et évolutives. J'ai beaucoup appris sur le plan personnel et professionnel. Ce stage a également considérablement enrichi mes connaissances en matière de technologies du Big Data.

Je conclurai ce rapport par une citation de Geoffrey Moore : "Sans l'analyse des données volumineuses, les entreprises sont aveugles et sourdes, errant sur le web comme des cerfs sur une autoroute." Ce stage a été mon premier pas pour me débarrasser de ma surdité et de ma cécité.

# Webography

**Angular** , https ://angular.io

**Spring boot,** https ://docs.spring.io/

**Project Management,** https [://www.atlassian.com/](http://www.atlassian.com/)

**Redis,** https ://redis.io/

**Docker ,** https ://docs.docker.com/

**Apache Spark,** https ://spark.apache.org/

**Design patterns,** https ://refactoring.guru/design-patterns/