*SISID - 20 BD EUGENE DERUELLE,*



*69003 LYON*

04 65 84 95 59

MAITRE D’APPRENTISSAGE: François AGIER

TUTEUR ENSEIGNANT : JEAN LUC BOREL

COPYRIGHT ©

2024

S

is

ID

-

ALL RIGHTS RESERVED

-

CONFIDENTIAL

SANDRO SPINA

:

4

-

IRC

: 2023

-

2024

Mise en place d’un nouveau système de

Communication client-support et gestion

de la migration du framework

REFERENCE :

RAPPORT

SANDRO SPINA

DATE:

MAI

2024

12

VERSION:

1

UPDATE:

12

MAI

2024

# Remerciements

Je tiens à exprimer ma gratitude envers Sis ID et plus particulièrement François AGIER, mon maître d’apprentissage pour la confiance accordée dans l’exécution des missions et pour ses bons conseils tout le long de ces dernières. J’aimerais aussi particulièrement remercier Nabil LAIDOUNI pour son soutien technique et pour avoir assuré la gestion du projet lors des périodes de cours.

# Résumé

Ce rapport détaille mon projet d'alternance au sein de Sis ID, une fintech française spécialisée dans la sécurisation des virements bancaires. Le projet se divise en deux volets principaux : la migration d'Angular vers une version plus récente et l'intégration de la plateforme de support Zendesk.

La migration d'Angular vise à améliorer la performance et la sécurité de l'application en passant à une version plus moderne. Ce projet sera supervisé par mes soins, et devront être effectuées plusieurs tâches dont un audit de l'application existante, la formation du développeur, l’élaboration de tâches dont l’avancement sera géré en méthode agile grâce à l’outil Jira, la relecture de code et l’assistance technique.

L'intégration de Zendesk a pour but d'améliorer les fonctionnalités de support client de Sis ID. Elle inclut l'analyse des besoins en collaboration étroite avec l’équipe support qui sont les premiers impactés par ce changement, la configuration initiale de Zendesk, son intégration avec les systèmes existants, et la personnalisation de l'interface utilisateur. Le déploiement comprend des étapes de formation des utilisateurs, des tests de non-régression et d’ajout de fonctionnalités.

Malgré des défis, les projets avancent et sont en bonne voie d'atteindre ses objectifs initiaux.

Table des matières

Remerciements 2

Résumé 3

1. Introduction 6



Présentation de

l’entreprise et des services

6

|  |  |
| --- | --- |
| Contexte et objectifs du projet | 7 |
| 1.2.1. Contexte international | 7 |
| 1.2.2. Open banking | 7 |
| 1.2.3. CONFIRMATION OF PAYEE (cop) | 8 |
| 1.2.4. BNP PARIBAS et le lien avec SIS | 9 |
| 2. Environnement de travail | 10 |

10



L’équipe



Environnement technologique



Présentation du connecteur technique

10

11

2.3.1. Architecture 12

2.3.2. Technologies employées 13

3. Conception et méthodologie 14



Les responsabilités

14

 La méthodologie de travail 14

 Etude des besoins et des cas d’utilisation 15

3.3.1. prérequis 15

3.3.2. Cas d’utilisation 16

 Architecture de l’application web 17

3.4.1. Gestion des modules 17

3.4.2. Architecture des services et composants 18

 Exemple concret, développement 20

3.5.1. Création des composants partagés 20

3.5.2. Page d’analyse de l’activite sur l’application 21

3.5.3. Page de gestion des références internes 22 4. Tests et validation 24

Tests 24



|  |  |
| --- | --- |
| 4.1.1. Tests Unitaires | 24 |
| 4.1.2. tests d’intégration et Cypress | 25 |
| 4.1.3. intégration à l’intégration | 25 |
| Validation | 26 |
| 5. Apports et bénéfices du projet | 26 |

6. Conclusion 27

8. Ressources supplémentaires



Bilan du projet et du résultat

27



Bilan personnel

27

7.

Annexes

28



Glossaire

28



Sources

29

30



Informations interessantes

30

# 1. Introduction

## Présentation de l’entreprise et des services



Sis ID est une plateforme Saas ayant pour mission d'aider les entreprises dans leur lutte contre la fraude au virement bancaire.

La fraude au virement bancaire touche 1 entreprise sur 4 en France et représente 6 Trillions de dollars de préjudice à l'échelle mondiale en 2021.

La forme la plus connue et la plus populaire de ces fraudes aux virements bancaires est la fraude au faux fournisseur, qui consiste à usurper l'identité d'un fournisseur de l'entreprise : un fraudeur appelle par exemple le standard de l'entreprise en se faisant passer pour un fournisseur, prétend avoir changé son IBAN et en donne un nouveau. N'ayant aucun moyen de confirmer que cet IBAN est bien rattaché à l'identité de la société fournisseur, l'entreprise va effectuer des virements à l'IBAN frauduleux.

Afin de lutter contre ce fléau, Sis ID propose une plateforme permettant de mutualiser les expériences et les historiques de paiements de milliers d'entreprises et de faire appel à des organismes de références afin de certifier un couple identifiant société/coordonnées bancaires pour s'assurer de faire un virement à la bonne entité et non à un fraudeur.

Les clients peuvent faire des contrôles sur un couple identifiant société/coordonnées bancaires et Sis ID, après une batterie de vérifications, retourne un score (rouge, orange ou vert) au client. Sis ID propose deux produits :



My Sis ID, une plateforme Saas permettant d'effectuer des contrôles.



Sis Inside, un ensemble d'APl Rest permettant aux applications métier des clients d'automatiser les contrôles dans les processus de paiement.

## Description des projets



### 1.2.1.Migration d’Angular

Comme mentionné précédemment, Sis ID dispose d'une plateforme SaaS sous forme d'application web, ce qui évite le besoin d'intégrer Sis Inside dans les ERP des clients. Cette application repose sur diverses technologies, parmi lesquelles :

- **Java Spring** : Sis Inside fournit une API élaborée en Java, utilisant le Spring Framework pour garantir une robustesse et une flexibilité optimales.

- **Angular** : Ce framework front-end utilise TypeScript pour le typage statique, SCSS pour la gestion avancée des styles, et Jest pour la réalisation de tests unitaires afin d'assurer la fiabilité du code.

Le framework Angular reçoit une nouvelle version majeure tous les six mois, une fréquence relativement élevée à l'échelle d'une entreprise. Toutefois, il est crucial de maintenir les versions à jour pour plusieurs raisons. D'une part, les mises à jour régulières garantissent la sécurité du système en corrigeant les vulnérabilités potentielles. D'autre part, elles permettent de bénéficier des dernières optimisations et améliorations de performance, ainsi que des nouvelles fonctionnalités offertes par le framework.

Au début de ce projet, la version d'Angular utilisée était la 14. L'objectif fixé est de migrer vers la version 17. Cette mise à niveau vise non seulement à améliorer la sécurité et la performance de l'application, mais aussi à maintenir la compétitivité et l'efficacité du développement au sein de l'équipe. La migration implique une série de tâches incluant la mise à jour des dépendances, l'adaptation du code existant aux nouvelles API, et la réalisation de tests approfondis pour assurer la stabilité de l'application après la mise à jour.

**Ma mission consiste donc à gérer le projet de migration angular de l’application front-end.**

### 1.2.2 Migration de livechat vers zendesk

#### Analyse de l’écosystème actuel au sein de l’entreprise

Sis ID comporte plusieurs départements. Ceux qui nous intéressent ici, sont les départements informatique, CSM (Customer success management) et support.

Le rôle de ces deux derniers départements est la gestion des demandes externes à Sis ID. Pour ceci, ces deux équipes utilisent en tout 11 outils, dont les utilisations sont très spécifiques. Le fait d’utiliser 11 outils différents au sein d’une même équipe implique une grosse charge d’apprentissage et une difficulté à obtenir les informations souhaitées en temps et en heure.

**C’est ainsi qu’une volonté de centraliser les outils** est née.

De plus, Sis ID ne dispose d’aucun logiciel de Helpdesk, c’est-à-dire de centralisation des points d’entrée des messages clients.

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquementVoici les outils que Zendesk va, de notre point de vue, centraliser

#### Comparatifs de l’existant

Pour accomplir cette tâche, plusieurs outils disponibles sur le marché ont été examinés. Il était donc nécessaire de réaliser un comparatif des solutions afin de choisir celle qui serait la plus viable.

Trois solutions principales ont été considérées et débattues :

Une image contenant cercle, symbole, Graphique

Description générée automatiquement



1. **Maintien de l’utilisation de Hubspot** : Un logiciel CRM puissant et facile à utiliser, offrant un large éventail de partenaires et de solutions pour améliorer l'expérience client.

2. **Freshdesk** : Une plateforme de service client omnicanal en libre-service. Elle propose un tableau de bord de reporting omnicanal, une fonctionnalité de détection de collision d’agents, une répartition de la charge de travail, etc.

3. **Zendesk** : Une plateforme de service client offrant des solutions de support multicanal, une gestion des tickets, des bases de connaissances en libre-service et des outils de collaboration pour améliorer l'expérience client et l'efficacité des équipes.

Les différences entre ces outils ont été compilées dans le tableau suivant :



**En conclusion :**

Dans l'entreprise, nous avions le choix entre conserver Hubspot, pour lequel notre équipe était déjà formée, ou migrer vers Zendesk ou Freshdesk. Nous avons opté pour Zendesk en raison de ses fonctionnalités avancées et de son écosystème riche. Zendesk offre un support multicanal plus complet, des outils de collaboration robustes, et des capacités de reporting et d'analyse supérieures. De plus, sa flexibilité et son évolutivité permettent une personnalisation précise pour répondre à nos besoins spécifiques. Ces avantages nous assurent une amélioration de l'expérience client et une augmentation de l'efficacité de notre équipe de support.

**Ainsi, j’ai été chargé de prendre part à l’équipe d’intégration de Zendesk, en tant que développeur full stack.**

# 2. Migration Angular



L’équipe

Une image contenant texte, capture d’écran, conception

Description générée automatiquementL'organisation de l'équipe pour la migration d'Angular repose sur une structure claire et des responsabilités bien définies, garantissant ainsi une gestion efficace et un suivi rigoureux du projet.

François est chargé du management macro du projet. Il supervise l'avancée globale du projet en se basant sur les rapports fournis par Nabil. Sa responsabilité est de s'assurer que le projet suit les grandes lignes directrices et les objectifs fixés, en intervenant si des ajustements stratégiques sont nécessaires.

Nabil, mon responsable direct, joue un rôle crucial dans le succès de la migration. Il veille à ce que le projet progresse conformément au plan établi et valide ou invalide mes décisions de gestion de projet. En plus de ses fonctions de supervision, Nabil peut également apporter un soutien technique, intervenant directement pour résoudre des problèmes complexes ou guider l'équipe sur des aspects techniques spécifiques.

## En tant que chef de projet de la migration d'Angular, j'assume plusieurs responsabilités essentielles. Mon rôle commence par la gestion de la formation du développeur, Tiavina, au début de son stage. Je suis chargé de l'aider à monter en compétences et à comprendre les spécificités du projet. De plus, je gère le planning sur JIRA, veillant à ce que toutes les tâches soient correctement planifiées et suivies. En collaboration avec Nabil, je rédige les épiques et les tickets pour la migration, en prenant soin de bien identifier et anticiper les breaking changes probables. Une autre de mes responsabilités clés est la relecture des merge requests de Tiavina, assurant ainsi la qualité et la cohérence du code avant son intégration.

## Tiavina, le développeur de l'équipe, a la tâche cruciale de mettre en œuvre la migration d'Angular. Après avoir bénéficié de la formation et du soutien nécessaires, il est chargé de réaliser les modifications et les mises à jour du code selon les directives établies. Son travail est central pour le succès du projet, car il applique concrètement les plans et les stratégies définis par l'équipe.

## Cette organisation claire et cette répartition des rôles permettent une gestion efficace et coordonnée du projet de migration d'Angular, maximisant les chances de réussite tout en assurant une progression régulière et contrôlée.

## Environnement technologique



Présentation de l’environnement technologique de l’application qui impacte cette migration.

## Gestion du projet



### 2.3.1. Formation

Parler du projet de formation élaboré pour former le développeur.

### 2.3.2.Méthodologie employée pour le développement

#### DEVELOPPEMENT

Parler des relectures de MR

TESTS UNITAIRES / TESTS D’INTEGRATION

Parler de la CI / CD

TESTS D’ACCEPTATION

Parler de tests réguliers faits pour vérifier la tenue de route de l’application

#### VALIDATION

Parler de réunions probables

### 2.3.3 Méthodes agiles

Palrer des méthodes agiles mises en place (jira)

#### Présentation du planning

Parler du planning et des probables difficultés à le mettre en place

#### Présentation des épiques

Parler des épiques d’un point de vue gestion de projet (pas trop rentrer dans les détails de l’aspect technique)

## 2.4. Développement

### 2.4.1. Les points d’attention

Parler des breaking changes

### 2.4.2. Problèmes techniques rencontrés

### 2.4.3. Solutions mises en place

### 2.4.4. Rédaction d’une documentation

## 2.5. Résultats de la migration

### 2.5.1. Améliorations apportées

Performances, sécurité, UX

### 2.5.2. Respect du planning

Comparer l’avancement du projet avec les délais souhaités

## 2.6. Conclusion et recommandations

### 2.6.1 Leçons apprises

Parler des leçons apprises sur une telle migration, choses à refaire / ne pas refaire

# 3. Migration chat Zendesk

## 3.1. Objectifs de ce projet

### 3.1.1. Attendus – client

Transparence, ou amélioration

### 3.1.2. Attendus – CSM / Support

Simplicités ajoutées (discuter du contenu de cette partie)

### 3.1.3. Attendus – IT

Suivi des tickets, informations nécessaires pour le fireman etc. ?

## 3.2. Environnement technologique

Par rapport à la partie homonyme dans la migration angular, parler du back ?

## 3.3. Planification

## 3.4. Développement

### 3.4.5. Tâches réalisées par moi-même

### 3.4.6. Problèmes rencontrés

Tous les problèmes rencontrés, sécurité Firefox, génération d’uid, impossibilité de changer les tickets pendant le tunnel d’inscription etc.

### 3.4.7. Solutions apportées

## 3.5. Tests

### 3.5.1. Tests unitaires et d’intégration

### 3.5.2. Tests finaux

Parler des résultats et de la satisfaction de l’équipe produit et support ?

## 3.6. Résultats et bénéfices

### 3.6.1. Pour Sis ID

Bénéfices pour support / CSM / IT

### 3.6.2. Pour moi

## Exemple concret, développement



Mon travail fût notamment de développer des pages complètes de l’application web, ce qui a nécessité la réalisation de plusieurs étapes, accomplies au préalable et à venir.



Validation

La validation approfondie du travail fourni pendant chaque période de sprint était extrêmement rigoureuse. Les points évoqués précédemment étaient intégrés de manière cohérente dans le processus de validation, en tenant compte des tests unitaires et des tests end-to-end effectués avec Cypress pour identifier toute divergence par rapport aux attentes.

Une fois qu'un travail était intégré, la responsabilité incombait au développeur de s'assurer du bon fonctionnement de son code et de son harmonie avec le reste de l'application déjà développée.

En complément, une réunion hebdomadaire était systématiquement tenue à la fin de chaque sprint pour garantir la corrélation entre les attentes et les résultats obtenus. Des tickets étaient ouverts lorsque certaines spécifications n'étaient pas pleinement respectées. De plus, une revue avec le responsable de l'équipe produit était régulièrement planifiée pour recueillir ses commentaires éclairés.

# 6. Conclusion



Bilan du projet et du résultat



Bilan personnel

# 7. Annexes



Glossaire

MERGE REQUEST : Demande de fusion de la branche de travail sur la branche principale.

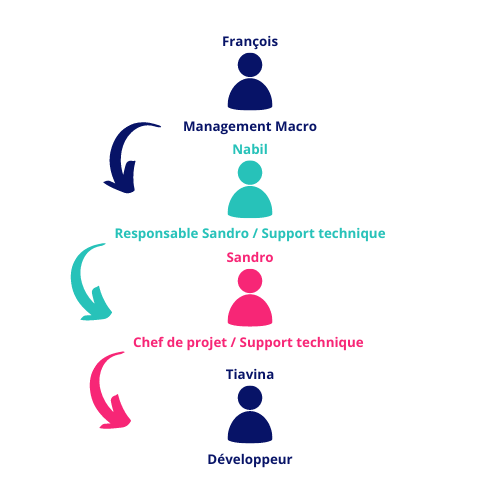
SAAS : Software As A Service, l’entreprise fournit son logiciel en tant que produit/service.

SCRUM : Méthode de travail agile qui consiste à diviser les temps de travail en sprint.

SPRINT : Dans le contexte d’une entreprise fonctionnant en méthode agile, un sprint est une période de travail. Chaque Sprint est identifié par un nom, une date et les bilans sont gardés.

API : Une API (Interface de Programmation Applicative) permet à des logiciels de communiquer et d'échanger des données entre eux.

FRAMEWORK : Une structure logicielle fournissant des outils et des bibliothèques pour faciliter le développement d'applications en offrant des fonctionnalités communes et des conventions de codage réutilisables.

SONUBE : Plateforme d’analyse qui analyse automatiquement le code pour détecter les problèmes de qualité, les violations de règles, la dette technique et améliorer la maintenabilité.

CI/CD : Pratique d’automatisation de la livraison d'applications. Nous intégrons régulièrement le code le livrons rapidement et de manière fiable. Ceci améliorant la qualité et permettant des déploiements fréquents.

#### CRM : Stratégie de gestion des interactions et des données clients pour améliorer les relations et la satisfaction.

#### Omnicanale : Approche intégrée de service client où tous les canaux de communication sont unifiés pour une expérience cohérente et fluide.



Sources

[Sis ID - Solution de lutte contre la fraude aux virements bancaires (sis-id.com)](https://sis-id.com/)

[Angular - Introduction to the Angular Docs](https://angular.io/docs)

// TODO à compléter

# 8. Ressources supplémentaires



Informations intéressantes

