versusmind

UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

Développement d'une plateforme de recueil de consentements RGPD

Rapport de stage ST50 - A2019

Nicolas BALLET

Département Génie Informatique
Filière libre

Entreprise Versusmind

30 avenue du Rhin
67000 Strasbourg
versusmind.eu

Tuteur en entreprise
Philippe Didiergeorges

Suiveur UTBM
Vincent Hilaire

utbm
université de technologie
Belfort-Montbéliard

Je tiens tout d'abord à remercier Versusmind et l'UTBM pour m'avoir donné l'opportunité d'effectuer ce stage.

Philippe Didiergeorges pour m'avoir suivi et guidé durant mon stage.

Rémi Benoit et Jacques Lorentz pour m'avoir aidé et épaulé au sein de l'équipe durant ma formation.

Francois Simond et Ahmed Zahri, qui ne faisaient pas partie de mon équipe et qui m'ont tout de même apportés un grand support.

Table des matières

Chapitre 1

Introduction

1.1 Entreprise

Versusmind est un cabinet d'architecture numérique, il est découpée en plusieurs agences, Nancy (centre), Metz, Strasbourg, Paris, Luxembourg.
Il compte aujourd'hui environ 200 employés. Il fournit aux entreprises une expertise ainsi que la réalisation leurs projets.

1.2 Projet

1.2.1 Contexte

Adopté par l'Union Européenne en avril 2016, la date d'entrée en vigueur du RGPD est le 25 mai 2018. Celui-ci oblige les entreprises à identifier les données personnelles en leur possession ainsi que leurs modalités de traitement et de protection et a pour objectifs de :

1. Uniformiser la réglementation au niveau européen
2. Responsabiliser les entreprises
3. Renforcer les droits des personnes

Le non-respect du RGPD peut mener à des sanctions financières importantes ainsi qu'a des sanctions administratives ayant un fort impact sur le fonctionnement de l'entreprise. Il est donc important pour les entreprises d'être en conformité avec le RGPD.
Mais migrer son système d'information pouvant coûter cher, Versusmind propose une solution nommée Central Consent Manager.

1.2.2 Central Consent Manager (CCM)

CCM est système de gestion des consentements centralisé. Il va permettre aux entreprises de se libérer d'un poids en s'assurant que ses utilisateurs ont bien consen-

tis à l'utilisation de leurs données personnelles en centralisant l'enregistrement et la vérification d'authenticité des consentements.

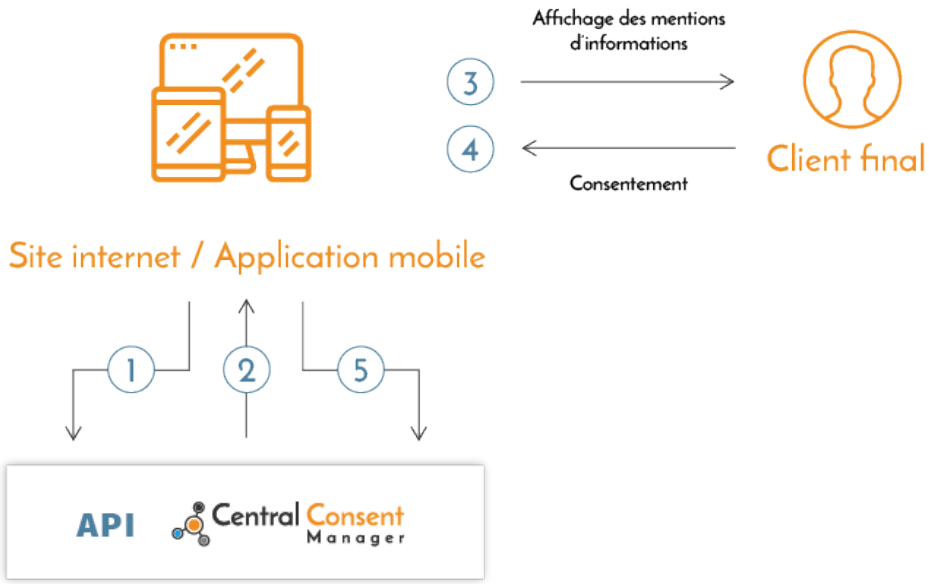

FIGURE 1.1 – Fonctionnement simplifié de l'application CCM

1. Requête sur le service de mentions pour un traitement
2. Envoi des mentions d'informations
3. Demande de consentement (sous forme d'e-mail ou autre)
4. Confirmation
5. Enregistrement du consentement

1.2.3 Ma mission

Dans une équipe de développeurs qui évolue beaucoup, mon rôle a été d'aider au développement de la plateforme CCM, à sa mise en production et à améliorer ses performances.

Chapitre 2

Développement

2.1 Méthodologie Scrum

J'ai été intégré durant mon stage à une équipe utilisant la méthodologie Scrum, qui fait partie des méthodes de gestion de projet agiles. Cela vise à supprimer ou au moins à réduire l'effet tunnel d'une méthode de gestion classique par exemple le Cycle en V.
Le développement est découpé en cycles (de trois semaines dans le cas présent) que l'on appelle "Sprint".
Les Sprints sont regroupés en saisons afin de représenter un objectif général. Par exemple, quand j'ai démarré mon stage, le but de la saison en cours était de terminer la refonte graphique. Au lieu d'un cahier des charges donné au début du projet, on va le découper en User Stories tout au long du projet avec l'aide du client.
Cela permet de rester concentré sur les aspects importants et de ne pas développer de fonctionnalités qui ne seront jamais utilisées.
Un Scrum Master est assigné au projet, son rôle est d'aider l'équipe à bien suivre un fonctionnement agile ainsi qu'a s'organiser. Il est aussi là pour guider le client dans la rédaction des User Stories.

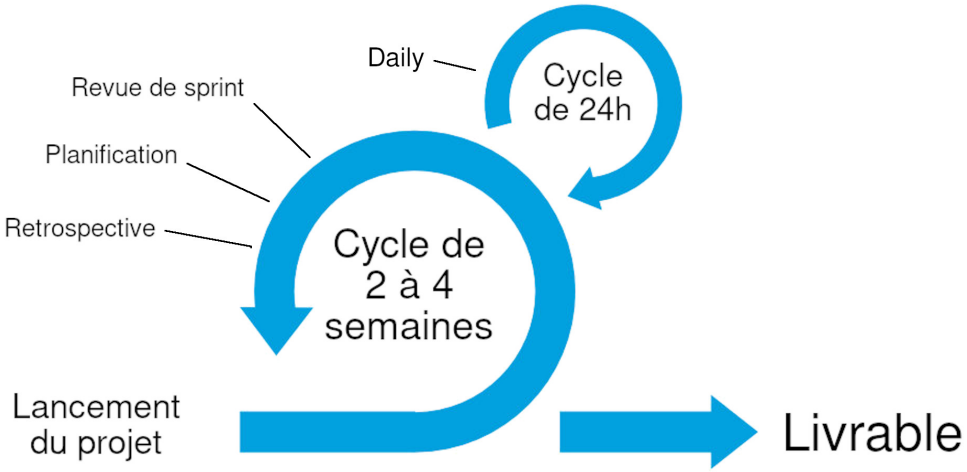

FIGURE 2.1 – Cycle de développement Scrum

2.1.1 Réunions journalières (Daily)

Chaque matin nous avons une petite réunion qui ne doit pas dépasser 15min afin d'informer rapidement le reste de l'équipe de l'avancement des différentes taches, de discuter brièvement des différents problèmes rencontrés mais aussi de choisir ensemble la priorité des différentes taches qu'il reste à faire.
Cela permet de toujours rester conscient de l'état du projet et d'avoir une vision d'ensemble du travail accompli et du travail restant.

2.1.2 Réunions inter-Sprints

Entre chaque Sprint, une série de réunions nous permet de rester dans la bonne direction concernant le développement du projet.

Revue de Sprint

Le but de la revue de Sprint est de montrer à toutes les parties prenantes (Project Owner, Scrum Master, développeurs) l'avancement du projet pendant le dernier Sprint.
L'équipe de développeurs décrit le travail accompli et après une démonstration du résultat, on discute afin de peut-être corriger la direction à prendre pour le prochain Sprint.

Planification

S'en suit la planification, elle se fait avec le Scrum Master ainsi que les développeurs.
On va y sélectionner les User Stories à implémenter durant le prochain Sprint. Cela constituera ce qu'on appelle "L'objectif de Sprint".

Sous forme d'un nombre on va se mettre d'accord sur une complexité pour chaque User Stories. Cela permet de vérifier qu'on en à bien la même compréhension. Si je propose 1 (facile) et qu'un de mes collègue propose 4 (moyen), c'est que l'un de nous deux à mal compris ce qui est attendu.
Ensuite on découpe chaque User Stories en taches concrètes et enfin on estime le temps de travail sur chaque taches.

Rétrospective

Et enfin, encore une fois avec le Scrum Master ainsi que les développeurs, on discute de comment s'est déroulé le dernier Sprint, des pratiques qu'on pourrait mettre en place ou améliorer, mais aussi de comment on l'a vécu et ressenti.
Cela s'inscrit dans un objectif d'amélioration de l'environnement de travail et de l'efficacité.

2.2 Architecture

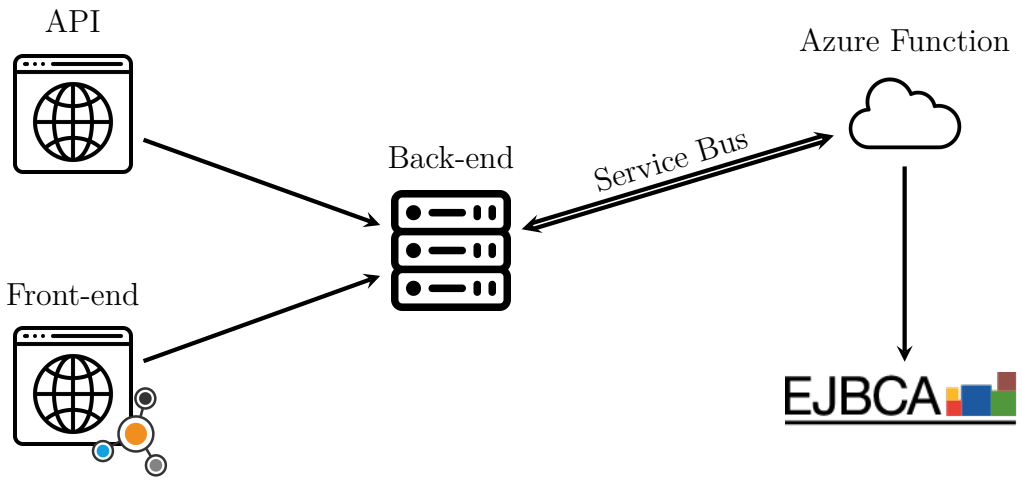

FIGURE 2.2 – Architecture globale simplifiée

Voici un rapide résumé des personnes destinées à utiliser l'application :

- Un administrateur Versusmind, qui pourra, via le Back-Office, accéder aux contrats souscrits avec les entreprises. Il n'a pas besoin de compétences techniques.
- Un administrateurs client, qui pourra, via le Back-Office, ajouter des traitements et des consentements. Il n'a pas besoin d'avoir de compétences techniques.
- Les développeurs clients, qui accédent à l'API du Back-end afin d'intégrer CCM dans leurs solution déjà existante.
- Les utilisateurs finaux, qui donneront leurs consentements au travers de la plateforme.

Tout d'abords je vais rapidement expliquer l'architecture du projet. La solution CCM est composée de différents éléments :

- Une partie Font-end écrite en HTML/CSS/Typescript avec le framework Angular, c'est le back-office d'administration, il sert les différentes pages web à l'utilisateur.
- Une partie Back-end écrite en Java avec le framework Spring Boot, c'est le cœur de l'application, elle peut recevoir des requêtes via le Front-end ou directement par API
- La gestion du chiffrement est en deux parties
  - Une instance EBJCA qui prends le rôle d'autorité de certification
  - Des Fonctions Azure écritent en Java qui chiffrent et vérifient les consentements. Elles communiquent avec le Back-end via des Service Bus Azure (de simples files d'attentes de messages JSON)
- Une instance Redis afin de garder du cache des différentes requêtes faites à la base de données.

2.2.1 Outils

Les différentes parties du stystème sont hébergées sur plateforme cloud Microsoft Azure. On utilise aussi Azure DevOps afin d'informatiser les notions de Sprints et de User Stories mais aussi pour faciliter la gestion du code source, jouer les jeux de tests et automatiser le déploiement des nouvelles versions.

2.2.2 Front-end

J'ai commencé mon stage en me formant sur l'utilisation du framework Angular sur lequel je suis maintenant à l'aise.

J'ai pu ensuite aider à la fermeture d'une saison de refonte graphique en apportant mes connaissances et mon expérience en intégration web à l'équipe.

J'ai aussi participé à l'amélioration et à l'enrichissement de l'interface sur toute le reste de mon stage.

Aussi, tout au long de ma formation, j'ai pu faire de la veille et guider l'équipe vers une restructuration de l'architecture CSS plus maintenable et plus facile à étendre.

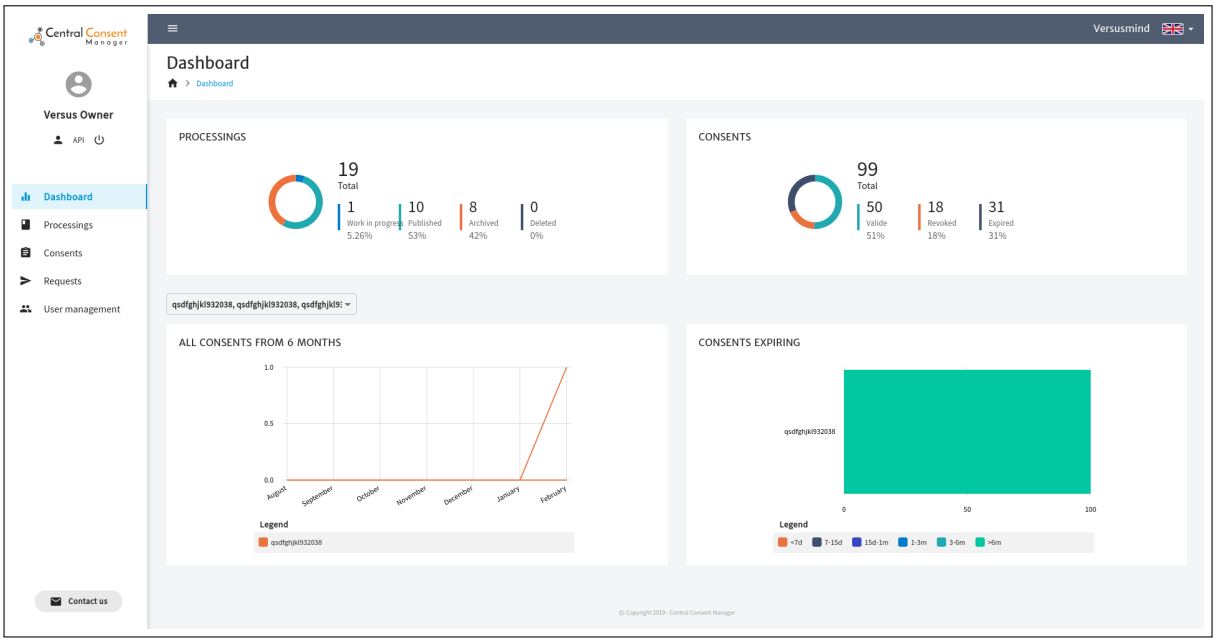

FIGURE 2.3 – Page d'accueil de l'application

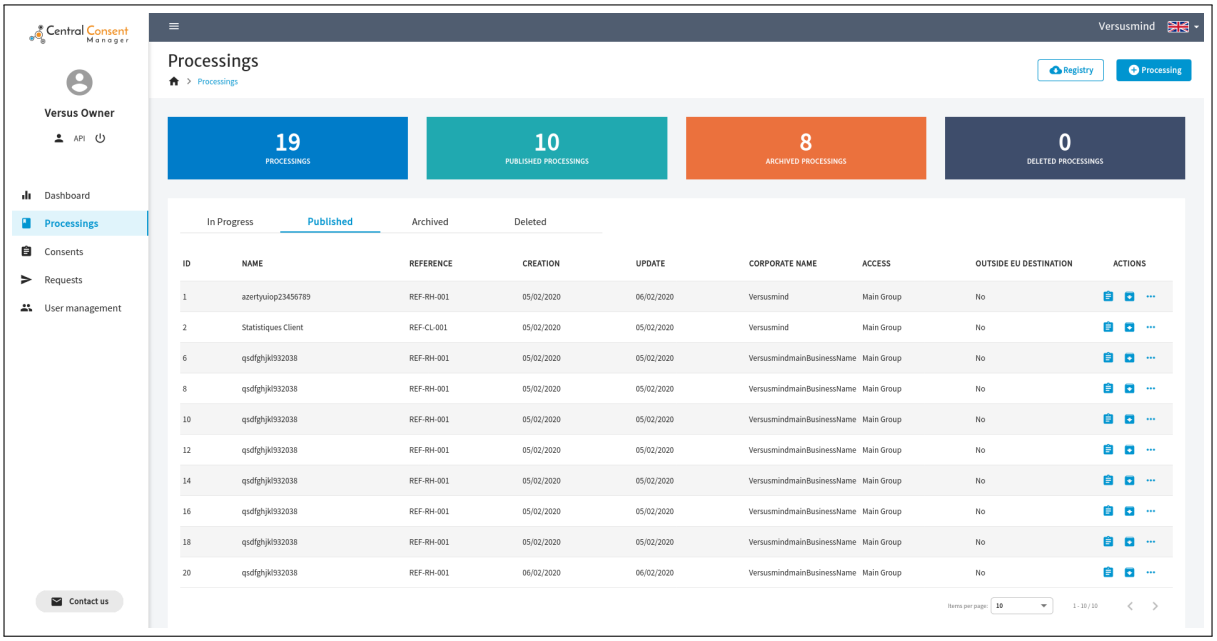

FIGURE 2.4 – Liste des traitements

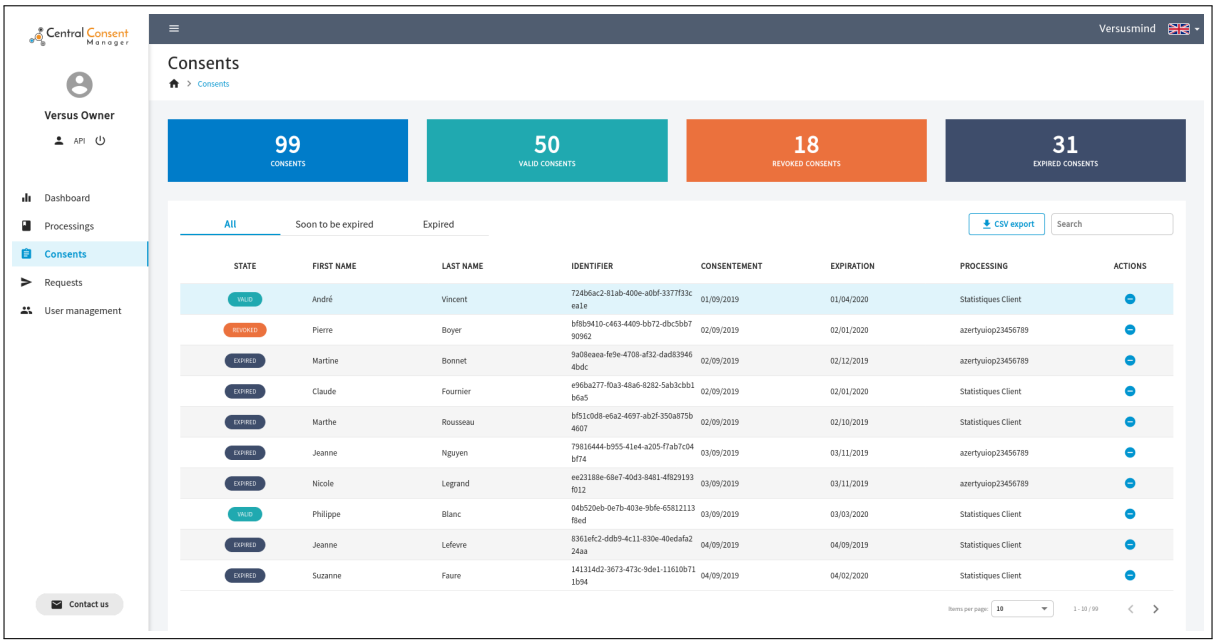

FIGURE 2.5 – Liste des consentements

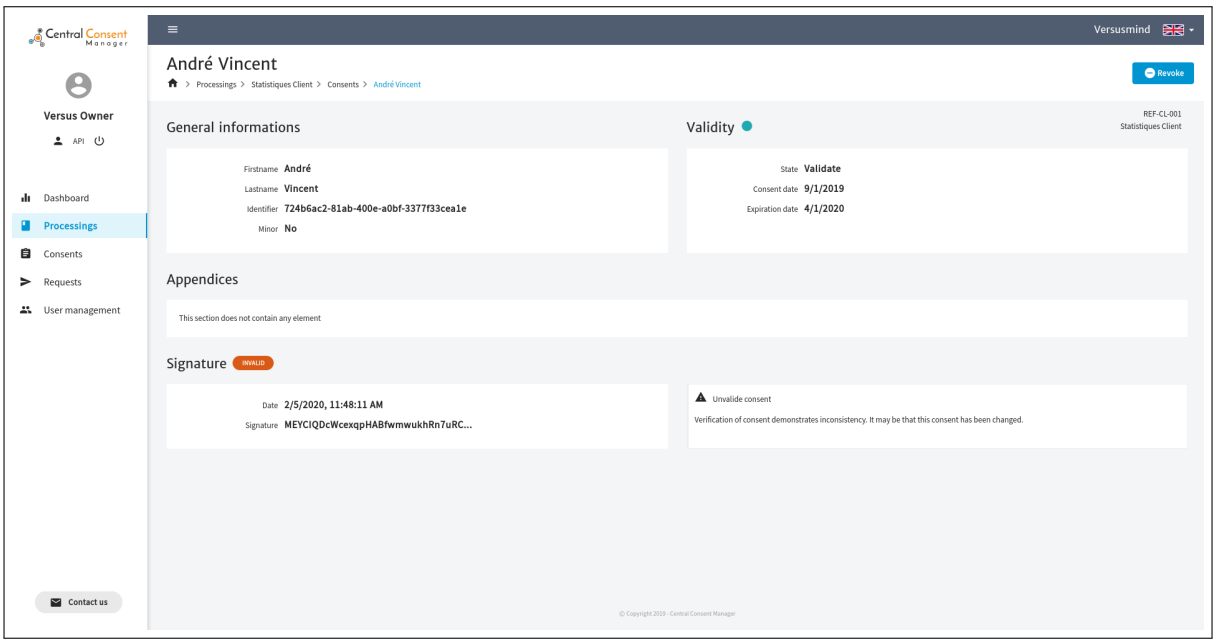

FIGURE 2.6 – Détail d'un consentement

2.2.3 Back-end

J'ai contribué au développement du back-end du projet, sous forme d'ajout de fonctionnalités, correction de failles de sécurité, correction de bugs, etc…

2.2.4 Gestion du cache

Je n'ai pas touché au développement de la gestion du cache côté serveur, mais j'ai participé aux discussions et aidé à la prise de décisions. Nous avons fait face à un problème de mise à jour des données présentes à la fois dans la base de données et dans le cache. La solution vers laquelle nous nous sommes orientés est de simplifier les requêtes faites en base pour unifier la récupération des données (abstraire le cache et la base de données) et de filtrer les données dans une couche applicative.

2.2.5 Certification numérique

J'ai eu à déployer une instance EJBCA assignée à l'instance de production de la plateforme CCM.
Cela comporte de la génération de clés cryptographiques, mais aussi de la manipulation de la plateforme Azure et du stockage de mots de passe sensibles car liés à la production.

2.2.6 Montée en charge

Suite à la refonte graphique, nous sommes entrés dans une saison de mise en production et deux problématiques de performances sont apparues assez vite :

- Lorsqu'un nouveau client veux migrer vers CCM, une base de consentements potentiellement conséquente doit être importée.
- Un client doit pouvoir exporter cette base sous forme de fichier, ce qui implique une vérification de validité d'un grand nombre de consentements.

Suite à des tests de charge, nous avons observé qu'il faudrait environ 3 jours pour importer une base d'un million de consentements. Et plusieurs heures pour générer un export des consentements enregistrés. Ce n'était pas acceptable.

Signature

J'ai eu la responsabilité d'implémenter les fonctions Azure afin de paralléliser chaque création de signature numérique.

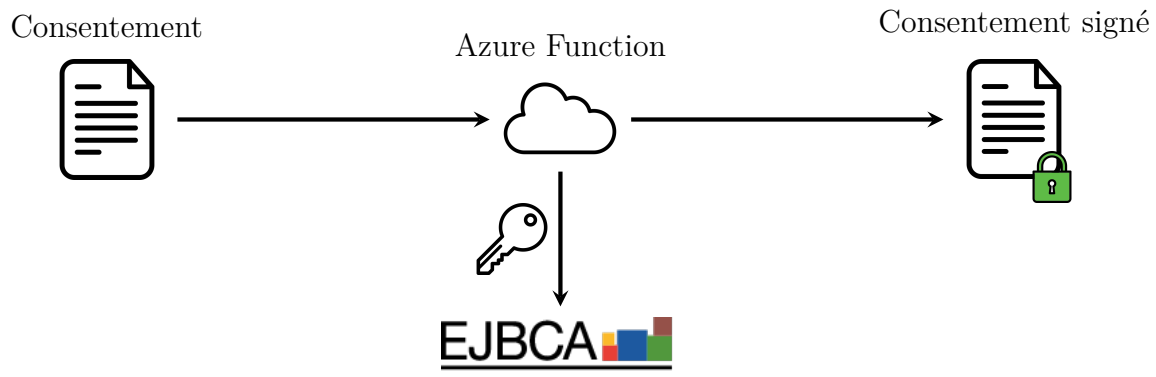

FIGURE 2.7 – Parcours de la signature d'un consentement

Dans la même tâche, j'ai pu implémenter la révocation de consentement en utilisant aussi des fonctions Azure.

Vérification

J'ai aussi eu à travailler sur la vérification des consentements et j'ai pu diviser le temps de signature par 2 tout en augmentant drastiquement le niveau de sécurité.

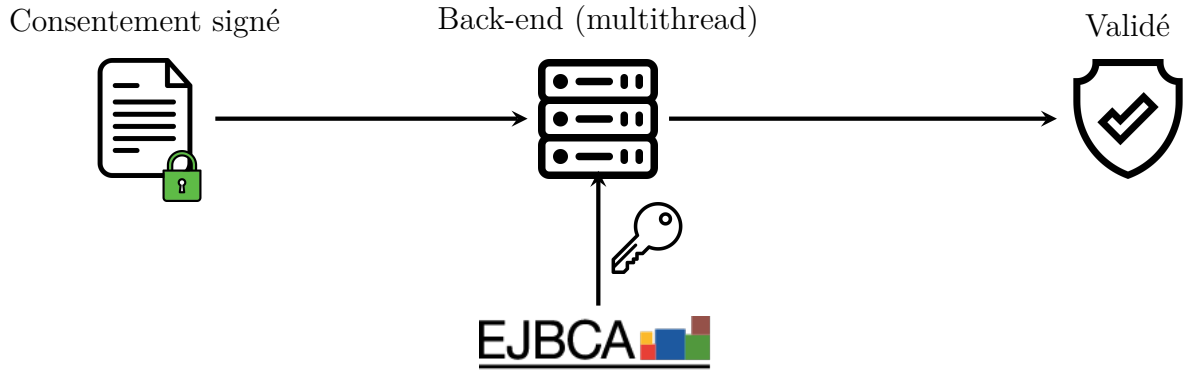

FIGURE 2.8 – Parcours de vérification d'un consentement non altéré

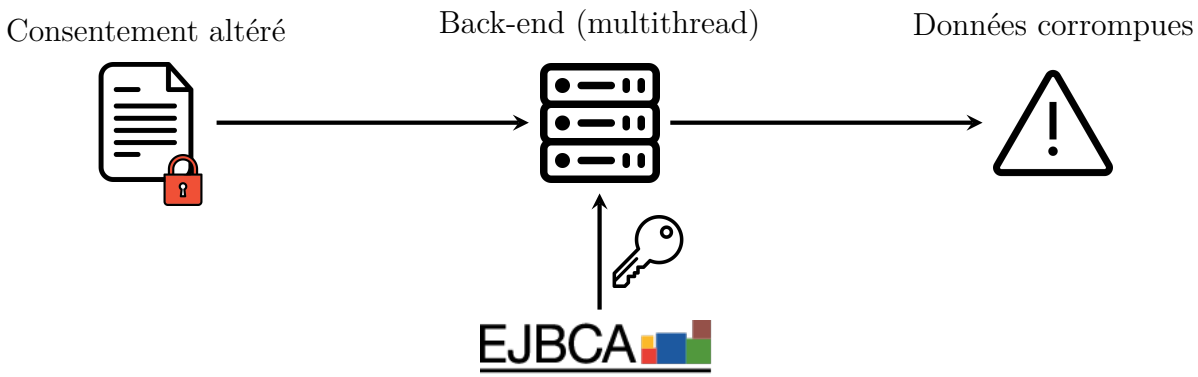

FIGURE 2.9 – Parcours de vérification d'un consentement altéré

2.3 Développement dirigé par les tests (TDD)

Durant mon stage, j'ai appris à implémenter des tests afin de subvenir à plusieurs besoins :

1. Fournir une preuve de fonctionnement du code testé
2. Prévenir d'éventuels bugs futurs
3. S'assurer du bon fonctionnement de l'application avant son déploiement

Le développement suit un cycle circulaire ou nous développons des nouvelles fonctionnalités, ce qui change le comportement de l'application, les tests ne passent donc plus. Nous réparons les test, et le cycle recommence.

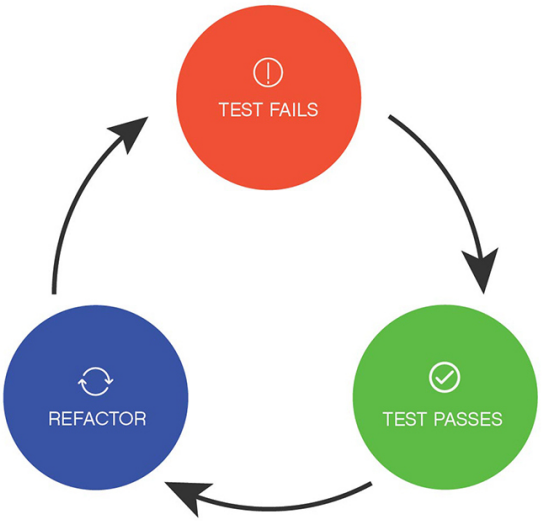

FIGURE 2.10 – Cycle de tests et de développement

Pour cela, on défini le comportement unitaire de chaque composant, ainsi qu'un ou plusieurs parcours d'utilisation critiques utilisant les éléments clés de l'application.
J'ai eu l'occasion de créer différents types de tests :

Les tests unitaires s'assurent que chaque composant de l'application rempli son rôle.

Les tests d'intégrations vérifient un fonctionnement comportant plusieurs composants.

Les test de bout en bout implémentent des User Stories completes, utilisant toutes les couches de l'application.

Et dans une démarche d'intégration continue j'ai participé à la mise en place de systèmes de tests et de déploiement automatisés sur la plateforme Azure DevOps.
J'ai pu utiliser les solutions SonarQube pour le Back-end et Eslint pour le Front-end qui s'occupent d'analyser le dépôt de code source afin d'y trouver des problèmes de sécurité, des bugs, des mauvaises pratiques ainsi que de la duplication de code.
Cela contribue à rendre le code plus stable et plus maintenable.

Chapitre 3

Conclusion

Ce stage m'a beaucoup apporté, d'un point de vue technique mais aussi au niveau organisationnel.
J'y ai appris à suivre un processus de développement plus stable, grâce au développement dirigé par les tests ainsi qu'a l'intégration continue via Azure DevOps.
J'ai aussi appris à développer des tests de charge, à interpréter leurs résultats afin d'en tirer des directions à prendre.

Mots clefs

RGPD - Consentement - Signature numérique - Plateforme Azure - Méthodologie Scrum - Cloud Angular - Spring Boot - Azure Functions

Nicolas BALLET

Rapport de stage ST50 - A2019

Résumé

TODO : Résumé

Entreprise Versusmind
30 avenue du Rhin
67000 Strasbourg
versusmind.eu

utbm
université de technologie
Belfort-Montbéliard

Annexe A

Personas

Les différents rôles dans l'application :

Versusmind

La société éditrice du logiciel

Propriétaire

La personne physique représentant le Responsable de traitement (Directeur général/Président par exemple) et ayant la capacité d'engager l'organisme.

Admin client

Par exemple : Un RSSI et/ou un Délégué à la protection des données (DPO) qui doivent pouvoir avoir tous les droits de lecture/écriture ainsi qu'un droit de gestion des Utilisateurs / Consultant.
Le RSSI et le DPO gèrent conjointement le projet de mise en conformité au RGPD au sein d'un organisme.
Si aucun des deux acteurs n'est nommé au sein d'un organisme et n'a pas vocation à être nommé, il est recommandé de désigner un "Référent données personnelles" ayant en charge le projet de mise en conformité.

Utilisateurs

Par exemple : Le directeur/directrice du service communication de l'organisme qui souhaite créer une newsletter en lien avec les clients de l'organisme ou des prospects ayant consenti sur le site internet de l'organisme.
Cette newsletter nécessite le consentement des clients et doit apparaître au registre des traitements.
Le directeur/directrice du service communication complète donc la partie correspondante dans le registre après que le RSSI et/ou le DPO ou, à défaut, le "Référent données personnelles" lui a octroyé les droits.

Consultant

Un consultant est une personne qui n'a que des droits de lecture sur le registre ou le consentement.
Par exemple : Une personne du service communication qui doit suivre les consentement sans avoir besoin de les modifier.
Ou encore : Une stagiaire qui a besoin d'y accéder pour travailler mais n'a pas soit la responsabilité nécessaire, soit le besoin de bénéficier d'un droit d'écriture.