

TÉLÉCOMMUNICATIONS
PROJET DE FIN D'ÉTUDES
RAPPORT

Développement d'une plateforme web de
géotechnique pour l'analyse des données
géologiques



Étudiant :
Alexandre Baudry (abaudry10@gmail.com)

Tuteur de stage :
Gilles Chapron (gilles.chapron@setec.com)

Tuteur pédagogique :
Charles Consel

Table des matières

1	Introduction	3
1.1	Contexte du projet	3
1.2	Objectifs du projet	3
1.2.1	Objectifs Généraux	3
1.2.2	Objectifs Spécifiques	3
1.2.3	Étendue du Projet	4
1.2.4	Critères de Réussite	4
1.2.5	Bénéfices Attendus	4
1.2.6	Innovations et Défis	5
1.3	Présentation de l'entreprise Terrasol	5
1.3.1	Historique et Expertise	5
1.3.2	Domaines d'Intervention	5
1.3.3	Approche Collaborative	6
1.3.4	Innovation et Technologie	6
1.3.5	Réputation Mondiale	6
1.3.6	Implication Sociale et Environnementale	6
1.4	Présentation de l'application web de géotechnique développée	7
1.4.1	Collecte de Données	7
1.4.2	Visualisation et Analyse	9
1.4.3	Gestion de Projet	10
1.4.4	Modules de Calcul Intégrés	12
1.4.5	Commercialisation et Licences	13
2	Cahier des charges	15
2.1	Description des besoins et des contraintes du projet	15
2.1.1	Besoins du Projet	15
2.1.2	Contraintes du Projet	15
3	Analyse de l'existant	16
3.1	Étude des solutions existantes sur le marché	16
3.2	Intérêt du Développement de l'Application Orbow	16
4	Architecture de l'Application	17
4.1	Frontend : React	17
4.2	Backend : Java avec Spring Boot	17
4.3	Base de Données : Liquibase et Tables AGS	17
4.4	Authentification et Gestion des Accès : Keycloak	17
4.5	Architecture Globale	17

5 Développement de Fonctionnalités Exemplaires	18
5.1 Gestion des Projets Géotechniques	18
5.1.1 Introduction	18
5.1.2 Création de la Table "Projets" dans la Base de Données :	19
5.1.3 Création d'un DTO, d'un Mapper, d'un Controller, d'un Service et d'un Repository	20
5.1.4 Implémentation de l'Interface Utilisateur pour la Gestion des Projets et Projets Supprimés	21
5.1.5 Fichier domainProjectTable.tsx : Gestion de l'affichage des projets	21
5.1.6 Fichier domainProjectTableRow.tsx : Gestion de l'affichage d'une ligne de projet	23
5.2 Développement d'une Interface Intuitive pour la Saisie de Données de Sondage	27
5.2.1 Interface de Saisie de Données de Sondage	27
5.2.2 Personnalisation des Données	28
5.2.3 Validation Rigoureuse des Données	28
5.2.4 Réduction des Erreurs et Garantie de Qualité	28
6 Implémentation	30
6.1 Conception et Développement de la Page d'Accueil	34
7 Respect des échéances	35
7.1 Gestion du planning du projet	35
7.2 Suivi de l'avancement et des délais	35
7.3 Retour sur l'organisation et la gestion du temps	36
8 Retour sur expérience	36
8.1 Analyse des réussites et des difficultés rencontrées lors du développement	37
8.2 Mesure de l'impact de l'application sur la géotechnique et son utilisation	37
8.3 Réflexion sur les améliorations potentielles et les pistes d'évolution	38
9 Conclusion	38
9.1 Bilan du projet et des réalisations	38
9.2 Synthèse des principaux résultats obtenus	38
9.3 Perspectives pour des travaux futurs et des améliorations possibles	39
10 Bibliographie	39
Bibliographie	39
Annexes	39
Glossaire	41

1 Introduction

1.1 Contexte du projet

La géotechnique, en tant que domaine essentiel de l'ingénierie civile, joue un rôle fondamental dans la conception, la construction et la maintenance d'infrastructures durables et sécurisées. Cependant, ce secteur complexe est confronté à divers enjeux tels que la collecte et l'analyse de données géotechniques, la collaboration entre les professionnels, ainsi que la visualisation et la prise de décision basée sur des informations pertinentes.

Face à ces défis, le développement d'une application web dédiée à la géotechnique devient une nécessité impérieuse. Cette application a pour vocation d'améliorer la manière dont les données géotechniques sont collectées, analysées et utilisées, tout en favorisant la collaboration et en fournissant des outils de visualisation avancés pour une prise de décision éclairée.

Les utilisateurs finaux, tels que les ingénieurs géotechniciens, les chercheurs et les entreprises de construction, ont un besoin criant d'outils plus efficaces et conviviaux pour gérer la complexité croissante des données géotechniques. L'application web vise à répondre à ces besoins en simplifiant la collecte des données sur le terrain, en facilitant leur analyse et en permettant une collaboration en temps réel entre les professionnels dispersés géographiquement.

1.2 Objectifs du projet

Les objectifs du projet ont été soigneusement définis pour orienter le développement de l'application web de géotechnique au sein de l'entreprise Terrasol. Ces objectifs ont guidé notre démarche tout au long de la conception et de la réalisation de l'application, en nous permettant de concentrer nos efforts sur des réalisations spécifiques et d'atteindre des résultats mesurables. Les objectifs généraux et spécifiques du projet étaient les suivants :

1.2.1 Objectifs Généraux

L'objectif principal de ce projet était de développer une application web de géotechnique capable de répondre aux besoins spécifiques de Terrasol en matière de gestion et d'analyse des données géotechniques. L'application devait fournir une plateforme intuitive et efficace pour collecter, stocker, traiter et visualiser les données géotechniques liées aux projets de l'entreprise. L'application ayant également pour but d'être commercialisée à d'autres entreprises du secteur, celle-ci devait être modulaire pour s'adapter facilement aux besoins spécifiques de chaque client. L'objectif était de créer une solution polyvalente capable de s'intégrer dans divers environnements géotechniques.

1.2.2 Objectifs Spécifiques

Conception d'une Interface Utilisateur Intuitive : Créer une interface utilisateur conviviale et intuitive, permettant aux utilisateurs de naviguer facilement à travers l'application, d'accéder aux fonctionnalités clés et de visualiser les données géotechniques de manière efficace.

Gestion Centralisée des Données : Mettre en place un système de gestion centralisée des

données géotechniques, permettant aux utilisateurs d'importer, de stocker et de gérer divers types de données, tels que les informations des sondages, les échantillons de sol, les photographies et les rapports.

Visualisation Avancée : Développer des fonctionnalités de visualisation avancée pour afficher les données géotechniques sous forme de graphiques, de tableaux et de cartes interactives, facilitant ainsi l'analyse et l'interprétation des données.

Personnalisation des Rapports : Offrir la possibilité de générer des rapports personnalisés à partir des données géotechniques, permettant aux utilisateurs de créer des rapports conformes aux normes de l'industrie pour les projets spécifiques.

Intégration de Fonctionnalités de Tri et de Filtrage : Mettre en place des fonctionnalités de tri et de filtrage avancées pour permettre aux utilisateurs de rechercher rapidement et efficacement des données spécifiques parmi une grande quantité d'informations.

Gestion des Droits d'Accès : Implémenter un système de gestion des droits d'accès pour garantir que seules les personnes autorisées puissent accéder et modifier les données géotechniques sensibles.

Optimisation des Performances : Assurer des performances optimales de l'application, en veillant à ce qu'elle puisse gérer de grandes quantités de données tout en maintenant une réponse rapide.

1.2.3 Étendue du Projet

La portée du projet a inclus la conception, le développement et le déploiement de l'application web de géotechnique, ainsi que l'intégration des fonctionnalités spécifiques identifiées dans les objectifs du projet. Les limites ont été définies pour exclure certaines fonctionnalités non essentielles qui pourraient être explorées dans des phases ultérieures.

1.2.4 Critères de Réussite

Les critères de réussite ont été établis pour évaluer la satisfaction des objectifs du projet. Parmi les critères clés figuraient la convivialité de l'interface utilisateur, la fonctionnalité complète de la gestion des données géotechniques et la précision des fonctionnalités de visualisation.

1.2.5 Bénéfices Attendus

La conception d'une application web de géotechnique avec une approche de commercialisation et de vente de licences apporte plusieurs avantages à la fois pour notre entreprise et pour nos clients potentiels. En mettant l'accent sur la personnalisation et la diversité des licences, nous anticipons les avantages suivants :

Adaptabilité aux Besoins Spécifiques : Notre application permettra aux entreprises clientes, qu'elles soient internes ou externes, d'acquérir des licences correspondant précisément à leurs besoins en gestion et analyse des données géotechniques. Les fonctionnalités pourront être configurées en fonction des exigences spécifiques de chaque projet.

Optimisation des Opérations Géotechniques : Grâce à des fonctionnalités avancées de

visualisation, d'analyse et de génération de rapports, nos clients pourront optimiser leurs opérations géotechniques. L'accès rapide et efficace aux données stratégiques facilitera la prise de décision et la gestion des projets.

Économies de Temps et de Ressources : L'application contribuera à réduire considérablement le temps nécessaire pour la collecte, le traitement et l'interprétation des données géotechniques, ce qui entraînera des économies de temps et de ressources pour toutes les parties concernées.

Conformité aux Normes de l'Industrie : Les rapports générés par l'application seront conformes aux normes de l'industrie. Cela garantira la qualité et la fiabilité des résultats présentés.

Avantage Concurrentiel : En offrant une solution personnalisée et modulaire, notre application permettra aux entreprises clientes de se démarquer dans un marché concurrentiel. Les fonctionnalités avancées et la flexibilité des licences conféreront un avantage stratégique à nos clients.

Croissance des Revenus : La vente de licences personnalisées à des clients externes ou internes créera une source potentielle de revenus pour notre entreprise. La diversité des licences et des options de personnalisation augmentera notre potentiel de croissance financière.

1.2.6 Innovations et Défis

L'intégration de fonctionnalités avancées de visualisation et de personnalisation des rapports a constitué une innovation majeure dans le projet. De plus, la gestion des droits d'accès et la manipulation de grandes quantités de données ont représenté des défis techniques que nous avons relevés avec succès.

1.3 Présentation de l'entreprise Terrasol

1.3.1 Historique et Expertise

Fondée en 1979, Terrasol est une entreprise de renommée internationale spécialisée dans l'ingénierie géotechnique et géologique. Forte de plus de 44 années d'expérience, Terrasol s'est imposée comme un acteur incontournable dans le domaine, offrant des services de conseil, de conception et d'analyse des sols et des structures.

1.3.2 Domaines d'Intervention

Terrasol intervient dans une variété de projets d'ingénierie civile, de géotechnique et de génie civil. L'entreprise propose des solutions innovantes et sur mesure pour des ouvrages complexes tels que les infrastructures de transport, les barrages, les tunnels, les fondations, les ouvrages souterrains, et bien plus encore. Son expertise couvre une large gamme de secteurs, de l'énergie aux transports en passant par l'environnement.

1.3.3 Approche Collaborative

Terrasol est reconnue pour son approche collaborative, travaillant en étroite collaboration avec ses clients pour fournir des solutions adaptées à leurs besoins spécifiques. L'entreprise s'engage à comprendre les défis uniques de chaque projet et à apporter des réponses techniques pointues et innovantes.

1.3.4 Innovation et Technologie

En tant que leader de l'industrie, Terrasol investit continuellement dans la recherche et le développement de nouvelles technologies. L'entreprise met l'accent sur la numérisation, la modélisation numérique avancée et l'analyse de données pour optimiser la conception, la construction et la maintenance des ouvrages géotechniques.

1.3.5 Réputation Mondiale

Grâce à son expertise, à la qualité de ses prestations et à son engagement envers l'excellence, Terrasol a acquis une réputation mondiale auprès des clients et des partenaires. L'entreprise a participé à de nombreux projets emblématiques à travers le monde, contribuant ainsi à façonner des infrastructures durables et résilientes.

1.3.6 Implication Sociale et Environnementale

Terrasol est également engagée dans des initiatives sociales et environnementales, contribuant activement à la durabilité des projets et des communautés qu'elle dessert. L'entreprise adopte des pratiques responsables et intègre les considérations environnementales dans ses solutions.

1.4 Présentation de l'application web de géotechnique développée



FIGURE 1 – Nom de l'application

L'application web offre un ensemble complet de fonctionnalités conçues pour faciliter la manipulation et l'exploitation des données géotechniques. Parmi les fonctionnalités clés :

1.4.1 Collecte de Données

L'application permet aux utilisateurs de saisir et de télécharger facilement des données géotechniques telles que des données de sondage, des mesures de terrain, des propriétés des sols, des photographies et bien plus encore comme illustré sur les figures 2 et 3. Les interfaces intuitives guident les utilisateurs tout au long du processus de collecte, garantissant l'exactitude et la cohérence des données. Les utilisateurs bénéficient d'une expérience fluide grâce à des mécanismes de vérification intégrés qui s'assurent de la pertinence des données saisies et évitent les erreurs de frappe. Si une saisie incorrecte est détectée, des messages d'erreur informatifs sont affichés, permettant aux utilisateurs de corriger rapidement leurs saisies avant d'enregistrer les données. Cette approche proactive réduit les risques d'incohérences ou d'informations erronées dans la base de données, garantissant ainsi la qualité et la fiabilité des données stockées.

FIGURE 2 – Informations sur la stratigraphie

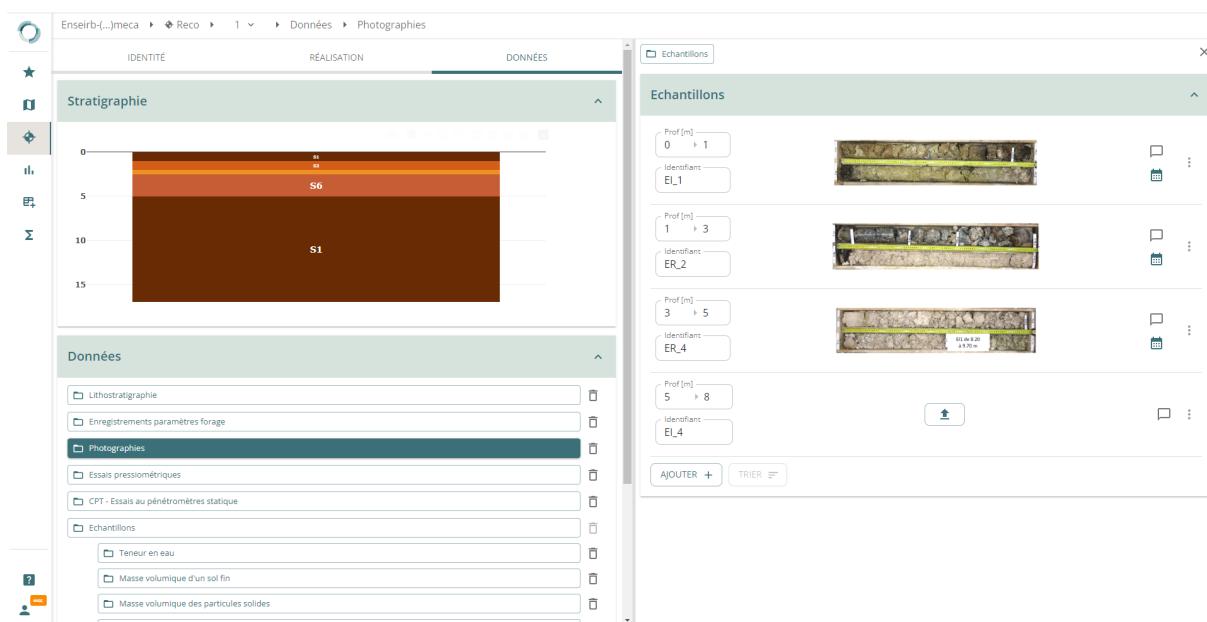


FIGURE 3 – Photographies des échantillons prélevé

De plus, l'une des fonctionnalités clés de l'application développée réside dans sa capacité à importer des données depuis des fichiers et à les traiter de manière efficace pour une visualisation et une caractérisation simplifiées. Cette fonctionnalité offre une solution pratique pour les utilisateurs souhaitant intégrer rapidement et facilement des données géotechniques dans l'application. Cette fonctionnalité est illustrée à la figure 4, qui présente l'importation d'un fichier CSV.

L'application permet aux utilisateurs de télécharger leurs fichiers contenant les données géotechniques. Une fois le fichier téléchargé, l'application effectue un traitement automatique pour extraire et organiser les données pertinentes. Cette étape inclut la validation des données pour s'assurer de leur cohérence et de leur intégrité.

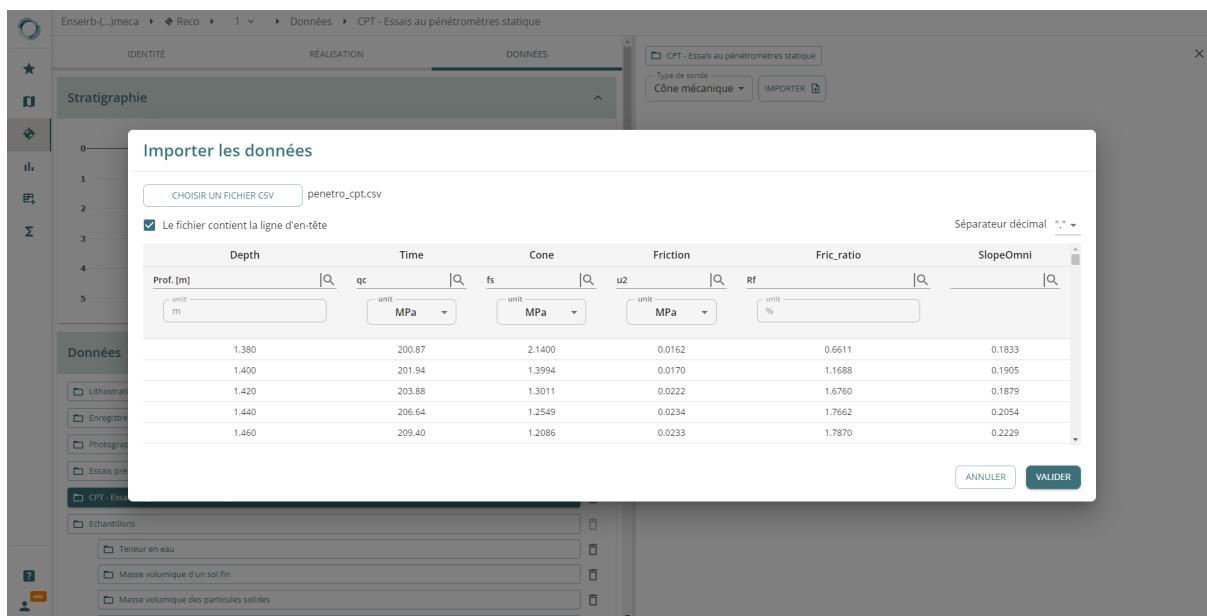


FIGURE 4 – Import de fichiers csv dans l'application

Après le traitement initial, les utilisateurs ont la possibilité d'affiner les données en effectuant des ajustements ou des modifications au besoin tout en bénéficiant d'une interface utilisateur intuitive qui facilite la manipulation des données. L'application offre également des fonctionnalités de visualisation avancée pour les données importées. Les utilisateurs peuvent afficher les données sous forme de tableaux ou de graphiques en fonction de leurs besoins spécifiques. Cette polyvalence de visualisation permet aux utilisateurs de mieux comprendre les tendances, les relations et les schémas au sein des données géotechniques. De plus, l'application propose une variété d'outils pour filtrer, trier et analyser les données importées. Les utilisateurs peuvent appliquer des filtres pour isoler des ensembles spécifiques de données et ainsi obtenir des informations ciblées. Ils peuvent également trier les données en fonction de divers critères pour une analyse comparative plus approfondie.

1.4.2 Visualisation et Analyse

Grâce à des outils de visualisation avancés, les données géotechniques peuvent être explorées sous forme de graphiques, de cartes et de modèles 3D interactifs. Les utilisateurs peuvent analyser les caractéristiques du sol, les contraintes géotechniques, les profondeurs de fouille et d'autres paramètres essentiels, offrant ainsi une meilleure compréhension du site et de ses enjeux. Une partie de ces fonctionnalités est présentée sur les figures ci-dessous.

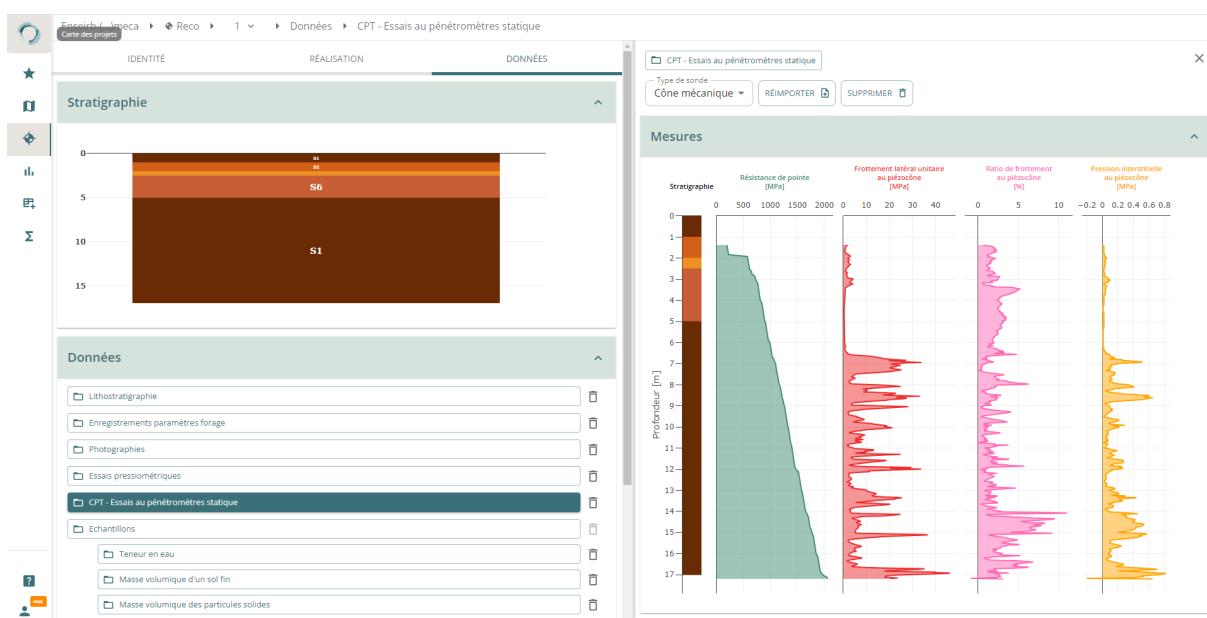


FIGURE 5 – Graphiques des informations des essais au pénétromètre statique

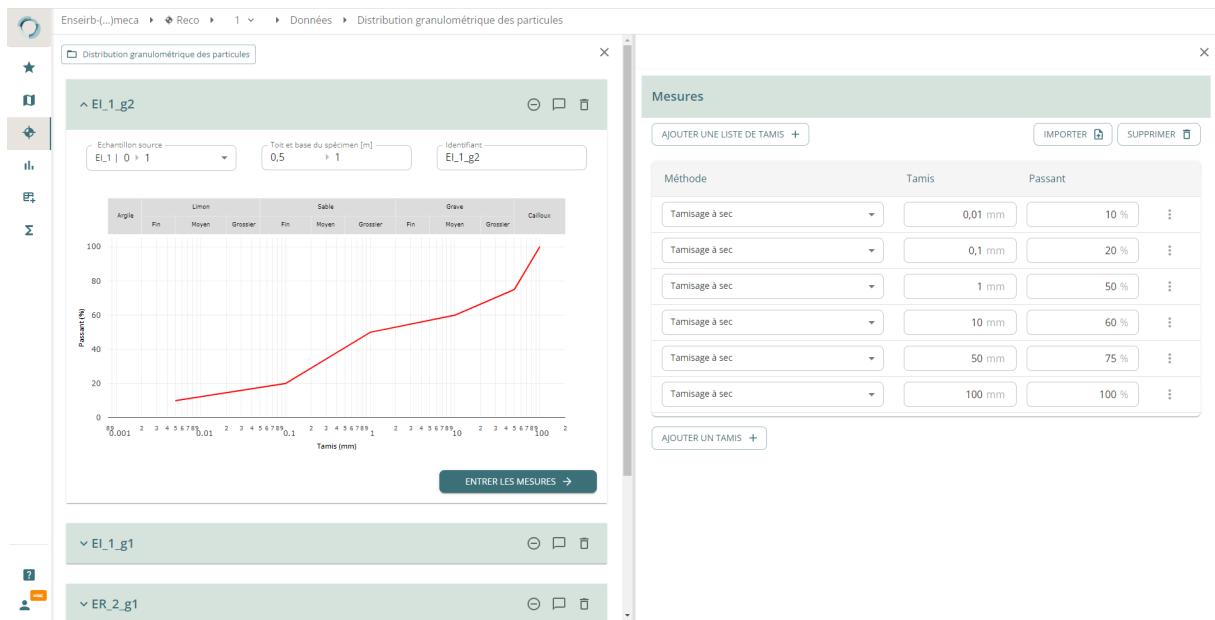
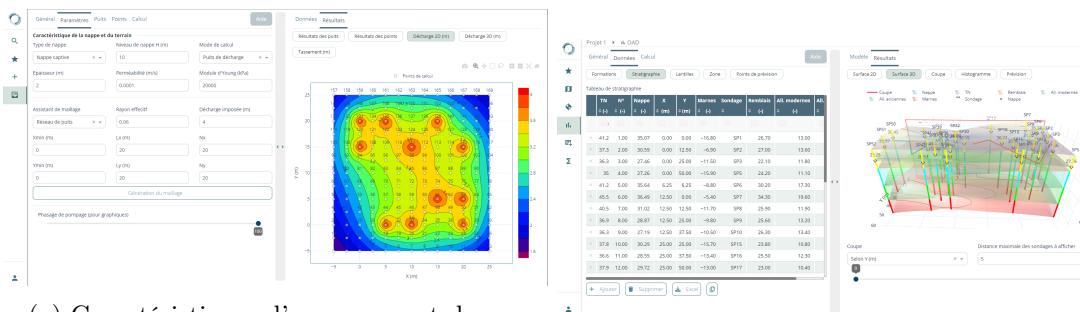


FIGURE 6 – Graphique de la distribution granulométrique des particules



1.4.3 Gestion de Projet

L’application permet la création (Figure 8) et la gestion centralisée de projets géotechniques. Les utilisateurs peuvent organiser et suivre les données spécifiques à chaque projet, ainsi que les documents et les rapports associés. Cette fonctionnalité facilite la collaboration au sein des équipes et optimise la communication entre les parties prenantes. Les figures 9 et 10 illustrent l’interface proposée à l’utilisateur pour gérer les informations générales ainsi que les différents secteurs et ouvrages de son projet.

Nouveau projet

Créer un projet + Importer un projet

Nom du projet *

Centre du projet

Système de coordonnées de référence * EPSG 4326 : WGS 84

Longitude * Latitude *

ANNULER VALIDER

FIGURE 8 – Crédit à la création d’un nouveau projet de géotechnique

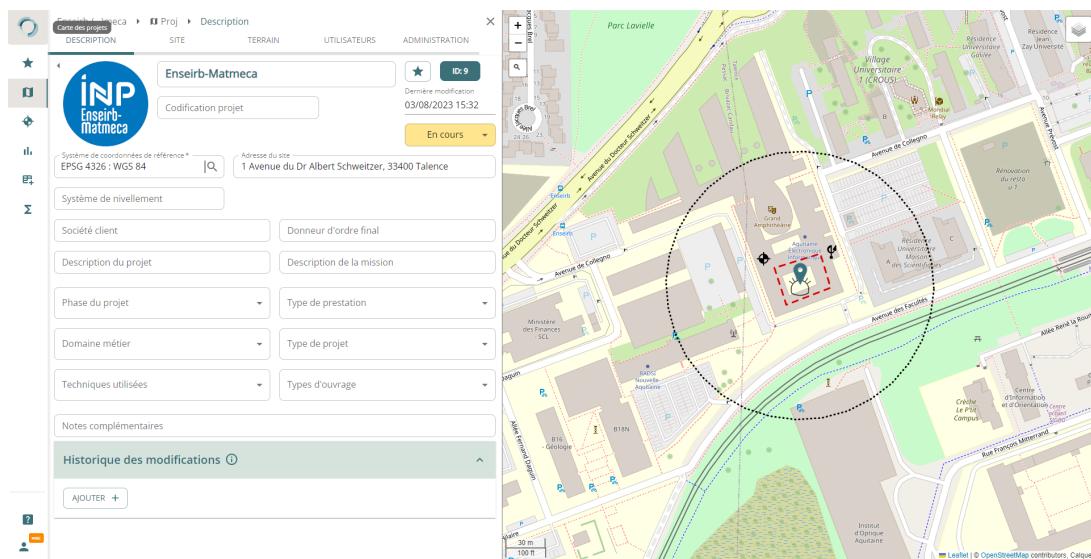


FIGURE 9 – Page de description du projet

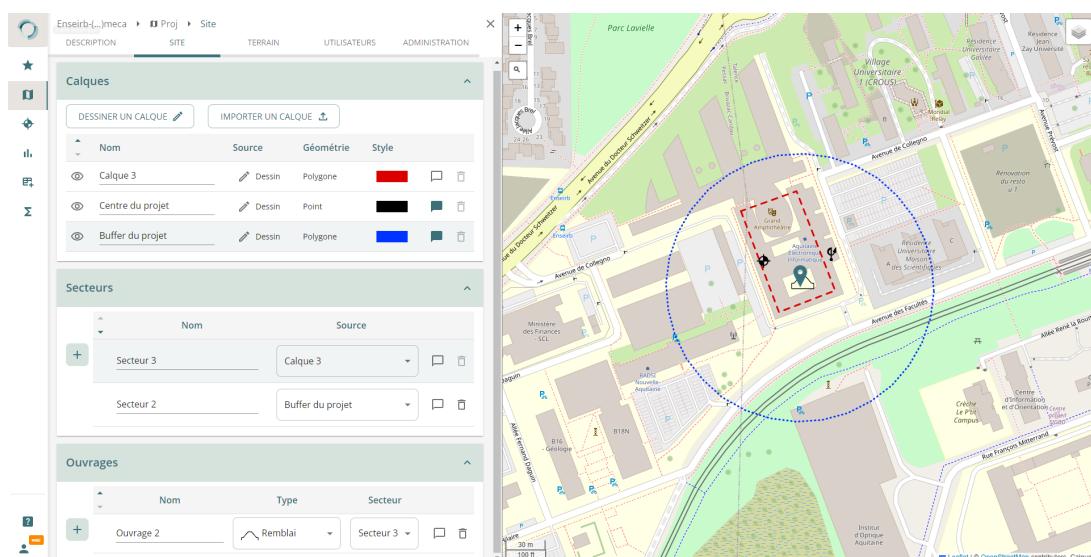


FIGURE 10 – Page permettant de définir le périmètre et la localisation des différents ouvrages

De plus, pour assurer une gestion efficace, chaque administrateur d'un projet a la possibilité de définir des droits spécifiques pour chaque utilisateur participant au projet (Figure 11). Ces droits peuvent être ajustés selon les besoins, allant de l'accès en lecture seule à l'administration complète du projet. Cette flexibilité permet une adaptation fine des niveaux d'accès et de responsabilité, renforçant ainsi la sécurité et la confidentialité des données tout en favorisant une collaboration transparente et ciblée au sein de l'application.

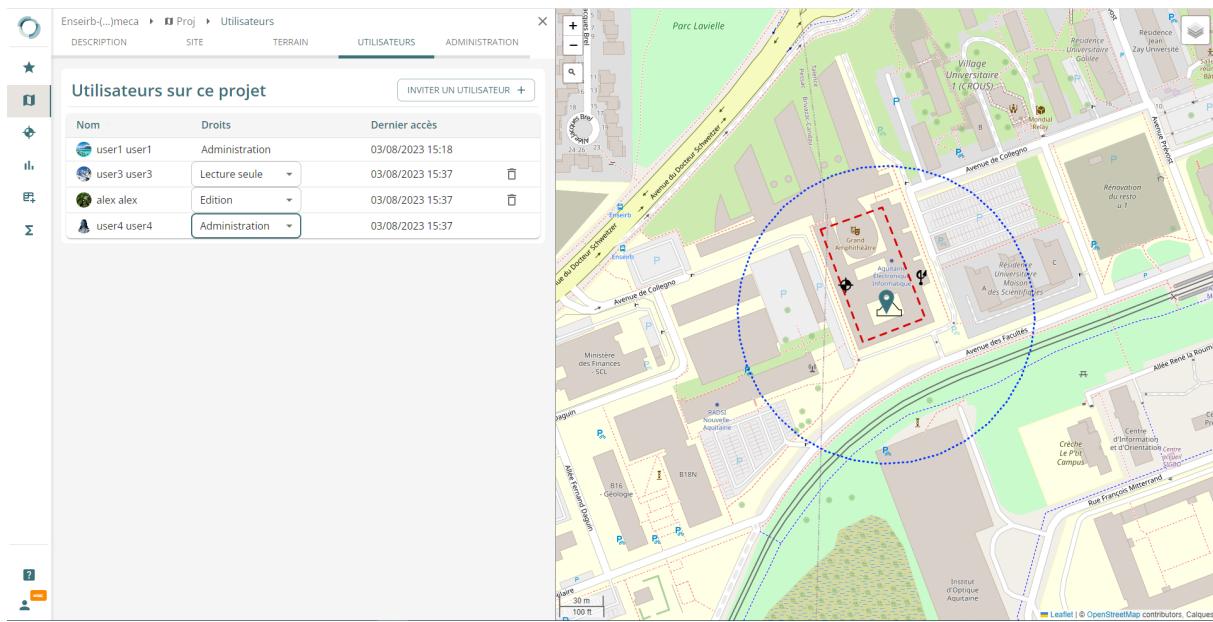


FIGURE 11 – Gestion des droits des utilisateurs sur un projet

1.4.4 Modules de Calcul Intégrés

L’application intègre également des modules de calcul avancés développés en Python, offrant aux utilisateurs des outils puissants pour effectuer des analyses géotechniques approfondies. Ces modules de calcul ont été spécifiquement conçus pour répondre aux besoins de l’industrie géotechnique et sont optimisés pour offrir des résultats précis et fiables. Grâce à cette intégration, les utilisateurs peuvent effectuer des analyses complexes directement au sein de l’application, sans avoir besoin de recourir à des logiciels externes. La liste de ces modules de calcul est présentée dans la figure 12.

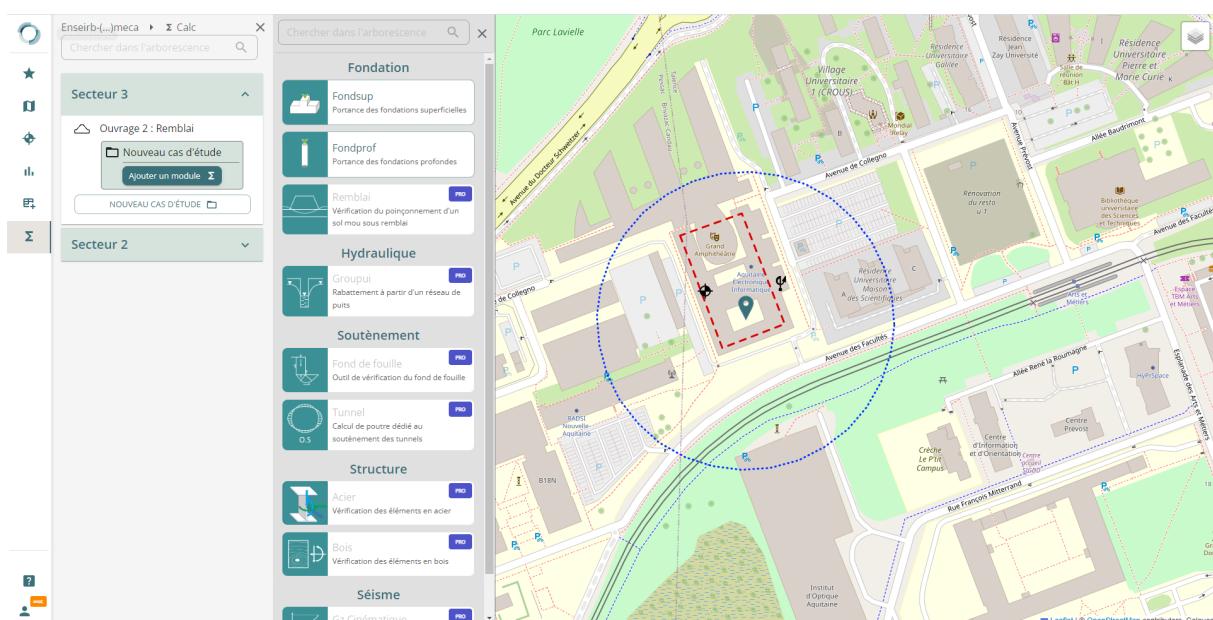


FIGURE 12 – Modules de calcul

1.4.5 Commercialisation et Licences

Outre son utilisation au sein de Terrasol, l'application web de géotechnique est conçue pour être commercialisée auprès d'autres entreprises du secteur. Afin de répondre aux besoins spécifiques de chaque entreprise cliente, notre modèle de licence offre une grande flexibilité. En effet, lors de l'achat, le client acquiert un domaine personnalisé au sein de l'application, et il a la possibilité de choisir le nombre de licences qui correspond le mieux à sa structure organisationnelle. Le client a la liberté de sélectionner un nombre spécifique de licences pour les utilisateurs de son domaine. Trois types de licences sont disponibles, chacun offrant des fonctionnalités de plus en plus avancées. Ces licences sont les suivantes :

Licences "Basic" : Ces licences offrent un ensemble de fonctionnalités essentielles pour la gestion des projets géotechniques. Elles permettent aux utilisateurs d'accéder aux fonctionnalités de base de l'application, telles que la création de projets, la gestion des données géotechniques et la visualisation des résultats.

Licences "Pro" : Ces licences étendent les fonctionnalités offertes par les licences "Basisques". Elles incluent des options avancées pour l'analyse des données, la génération de rapports personnalisés et la collaboration entre les membres de l'équipe. Les licences "Pro" sont destinées aux entreprises ayant des besoins plus complexes en matière de géotechnique.

Licences "Ulti" : Les licences "Ulti" offrent un ensemble complet de fonctionnalités avancées pour la gestion et l'analyse des données géotechniques. Elles conviennent aux entreprises qui exigent un haut niveau de personnalisation, d'intégration et de fonctionnalités avancées pour leurs projets complexes.

En plus de ces licences spécifiques, le client bénéficie d'un nombre illimité de licences "Free". Ces licences peuvent être attribuées par l'administrateur du domaine aux utilisateurs de manière flexible, en fonction de leurs besoins et de leur rôle au sein de l'entreprise. Cela permet à l'administrateur de gérer efficacement l'accès des utilisateurs tout en optimisant l'utilisation des licences.

Ce modèle de licence modulable garantit que chaque client peut personnaliser son expérience en fonction de ses besoins précis, tout en optimisant les coûts et en fournissant un accès adéquat aux fonctionnalités pour chaque utilisateur. Les figures 13 et 14 ci-dessous illustrent respectivement l'interface de suivi de l'utilisation des licences ainsi que l'interface de gestion des licences pour chaque utilisateur au sein d'un domaine.

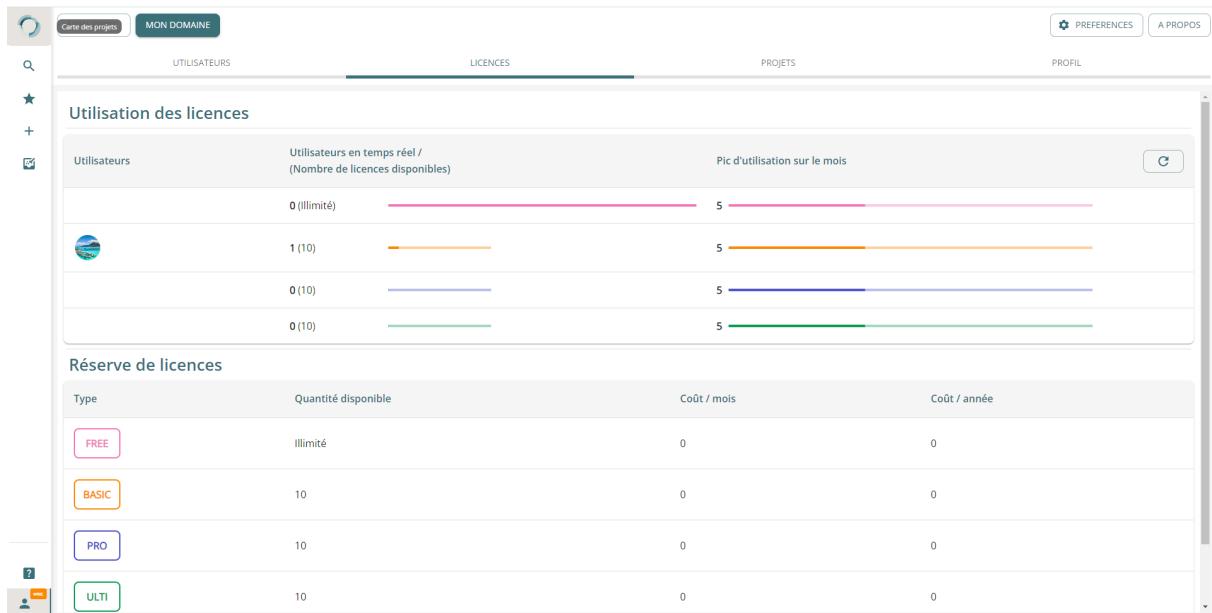


FIGURE 13 – Suivi de l'utilisation des licences

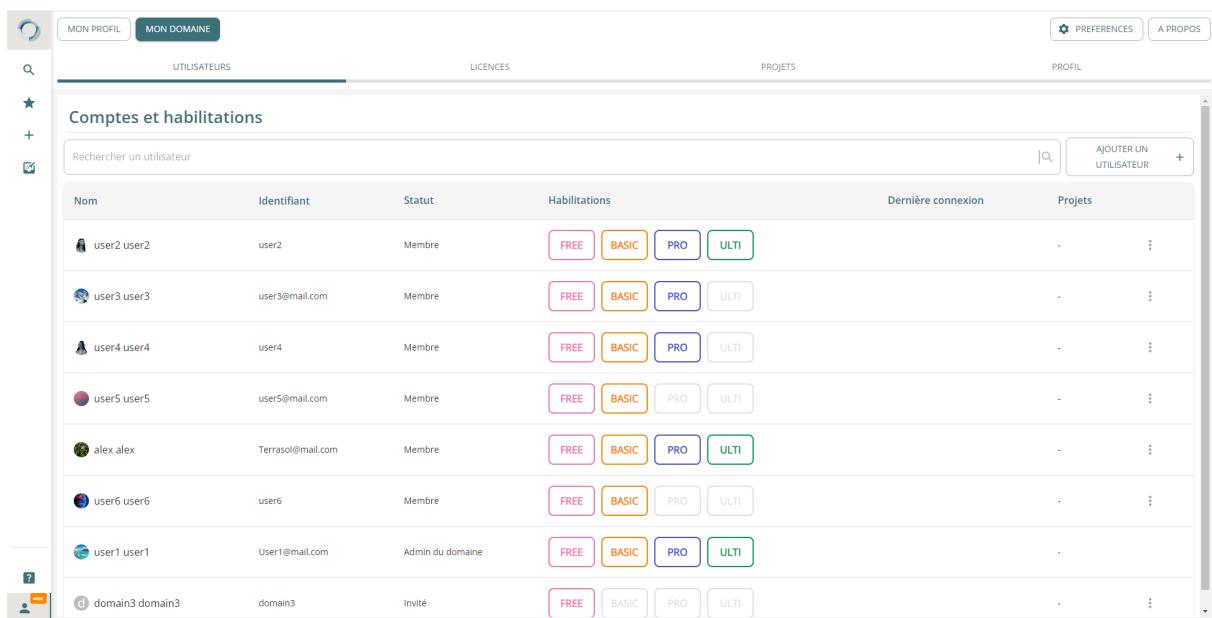


FIGURE 14 – Gestion des licences des utilisateurs

2 Cahier des charges

2.1 Description des besoins et des contraintes du projet

Dans cette section, nous abordons en détail les besoins essentiels ainsi que les contraintes majeures qui ont orienté la conception et le développement de l'application web de gestion géotechnique destinée à être commercialisée par Terrasol.

2.1.1 Besoins du Projet

Les besoins du projet ont été identifiés à partir d'une analyse approfondie des processus actuels de gestion géotechnique, des lacunes existantes dans les solutions disponibles sur le marché et des exigences spécifiques des clients cibles. Les principaux besoins qui ont orienté le développement de l'application sont les suivants :

Gestion Intégrée des Données Géotechniques : L'application doit permettre aux entreprises de gérer de manière centralisée et efficace toutes les données géotechniques pertinentes, notamment les informations des sondages, les échantillons de sol, les photographies et les rapports.

Visualisation et Analyse Avancée : Les utilisateurs doivent être en mesure de visualiser les données géotechniques sous forme de graphiques, de tableaux et de cartes interactives. Cela facilitera l'analyse approfondie des données et la prise de décisions éclairées.

Personnalisation des Rapports : L'application doit permettre aux utilisateurs de générer des rapports conformes aux normes de l'industrie en fonction de leurs besoins spécifiques. Ces rapports personnalisés doivent être faciles à créer et à partager.

Sécurité et Gestion des Accès : Étant donné la nature sensible des données géotechniques, l'application doit mettre en œuvre un système robuste de gestion des droits d'accès. Cela garantit que seules les personnes autorisées peuvent accéder et modifier les informations sensibles.

2.1.2 Contraintes du Projet

Outre les besoins, plusieurs contraintes ont influencé le développement de l'application web :

Compatibilité Multiplateforme : L'application doit être compatible avec divers navigateurs web et dispositifs (ordinateurs de bureau, tablettes, smartphones) pour garantir une expérience utilisateur cohérente.

Performances et Évolutivité : Étant donné que l'application traitera potentiellement de grandes quantités de données, il est essentiel de garantir des performances optimales et une évolutivité suffisante pour répondre à la demande croissante.

Sécurité des Données : La sécurité des données est primordiale. L'application doit mettre en œuvre des pratiques de cryptage et de protection des données pour éviter tout accès non autorisé.

Flexibilité de Personnalisation : Étant destinée à être commercialisée à différentes entreprises ayant des besoins variés, l'application doit être suffisamment flexible pour permettre une personnalisation en fonction des exigences spécifiques de chaque client.

Conformité Réglementaire : L'application doit se conformer aux réglementations et normes

de sécurité de l'industrie géotechnique, en garantissant que les données sensibles sont gérées conformément aux directives en vigueur.

Interface Utilisateur Intuitive : L'interface utilisateur doit être conviviale et intuitive pour faciliter l'adoption par les utilisateurs, même ceux ayant peu d'expérience en informatique.

3 Analyse de l'existant

3.1 Étude des solutions existantes sur le marché

L'exploration des solutions actuellement en utilisation, notamment en France, met en évidence une prédominance des logiciels de géotechnique développés par Terrasol, tels que Talren V6, Foxta V4, K-REA V4 et Straticad V3. Ces outils, reconnus pour leur expertise en géotechnique, offrent une gamme complète de fonctionnalités pour l'analyse, la conception et la modélisation géotechnique.

Ces logiciels sont conçus pour répondre aux besoins spécifiques des ingénieurs et des professionnels de la géotechnique en offrant des fonctionnalités avancées de calcul de stabilité des pentes, de fondations, de murs de soutènement et d'autres éléments géotechniques. Ils permettent également la modélisation numérique des sols et des interactions sol-structure.

Cependant, malgré leur expertise et leurs capacités, ces logiciels présentent certaines limitiations. Leur utilisation peut exiger une période d'adaptation initiale, car ils peuvent être complexes pour les nouveaux utilisateurs. De plus, leur licence et leur coût de maintenance peuvent être élevés, ce qui peut rendre leur accès difficile pour certaines entreprises, en particulier les PME.

3.2 Intérêt du Développement de l'Application Orbbow

Face aux lacunes identifiées dans les solutions existantes, le développement de l'application Orbbow se révèle particulièrement pertinent. Orbbow vise à offrir une alternative moderne et flexible pour la gestion et l'analyse des données géotechniques. Contrairement aux logiciels spécialisés, Orbbow cherche à simplifier la collecte, le traitement et l'interprétation des données, tout en les rendant accessibles à un public plus large.

En complément des logiciels Terrasol, l'application Orbbow permettra aux utilisateurs d'importer, de stocker et de gérer leurs données géotechniques de manière centralisée, tout en offrant des outils d'analyse avancés pour faciliter la prise de décision. De plus, la convivialité de son interface utilisateur, combinée à des fonctionnalités de personnalisation et à des options de tarification flexibles, répond aux besoins variés des entreprises du secteur.

Orbow se distingue également par sa capacité à intégrer des modules de calculs, offrant ainsi la possibilité de réaliser des analyses spécifiques et personnalisées. Cette modularité permet aux utilisateurs d'adapter l'application en fonction de leurs besoins uniques, tout en évitant de dépendre uniquement de solutions préconfigurées.

En résumé, l'application Orbbow représente une opportunité significative pour pallier les limitations des logiciels actuellement en utilisation, en offrant une approche moderne, conviviale et personnalisable pour la gestion des données géotechniques et l'analyse.

4 Architecture de l'Application

L'architecture de l'application que nous avons développée repose sur une combinaison de technologies modernes pour créer une plateforme robuste et évolutive. Cette architecture est conçue pour répondre aux besoins spécifiques de l'application tout en garantissant une séparation claire des responsabilités entre les différentes parties du système.

4.1 Frontend : React

Le frontend de l'application est développé en utilisant React, une bibliothèque JavaScript populaire pour la construction d'interfaces utilisateur interactives et réactives. React permet de créer des composants réutilisables et modulaires, ce qui facilite la conception d'une interface utilisateur conviviale. L'application frontend communique avec le backend via des appels API pour récupérer et afficher les données.

4.2 Backend : Java avec Spring Boot

Le backend de l'application est construit en utilisant Java et Spring Boot. Spring Boot est un framework qui simplifie le développement d'applications Java en fournissant des fonctionnalités prêtes à l'emploi pour la création de services web, la gestion des données, la sécurité et plus encore. Le backend gère la logique métier de l'application, traite les requêtes du frontend et communique avec la base de données.

4.3 Base de Données : Liquibase et Tables AGS

Pour la gestion de la base de données, nous utilisons Liquibase, un outil open source qui permet de versionner et de suivre les changements de schéma de base de données. Les fichiers XML de changelog sont utilisés pour définir les modifications de la structure de la base de données, ce qui permet de maintenir une base de données cohérente et mise à jour en fonction de l'évolution de l'application. Les données géotechniques sont stockées dans des tables AGS (Association of Geotechnical and Geoenvironmental Specialists), qui sont spécifiquement conçues pour stocker des informations géotechniques.

4.4 Authentification et Gestion des Accès : Keycloak

Pour la gestion de l'authentification et des accès, nous avons intégré Keycloak, un système d'authentification et d'autorisation open source. Keycloak permet de sécuriser l'application en gérant l'authentification des utilisateurs, la création de comptes, la gestion des rôles et des permissions. Cela garantit que seules les personnes autorisées ont accès aux fonctionnalités et aux données appropriées.

4.5 Architecture Globale

L'architecture globale de l'application suit le modèle client-serveur, où le frontend communique avec le backend via des API REST pour récupérer, envoyer et manipuler les données. Les

fichiers XML de changelog, gérés par Liquibase, permettent de maintenir la structure de la base de données à jour en fonction des besoins de l'application. L'utilisation de tables AGS offre une structure de stockage adaptée aux informations géotechniques, tandis que Keycloak assure la sécurité de l'application en gérant l'authentification et l'autorisation des utilisateurs.

Cette architecture bien pensée garantit la modularité, la flexibilité et la sécurité de l'application, tout en permettant une évolutivité future pour répondre aux besoins en constante évolution des utilisateurs.

5 Développement de Fonctionnalités Exemplaires

Dans cette section, nous allons examiner en détail quelques fonctionnalités clés que nous avons développées au cours de ce stage. Chacune de ces fonctionnalités répond à des besoins spécifiques des utilisateurs et démontre notre expertise en développement d'applications web.

5.1 Gestion des Projets Géotechniques

5.1.1 *Introduction*

Dans le cadre du développement de l'application, une part importante a été consacrée à la création de fonctionnalités répondant aux besoins spécifiques des administrateurs de domaine. Parmi ces fonctionnalités, nous nous sommes concentrés sur le développement d'une page dédiée à la gestion et à la visualisation des projets géotechniques créés par les utilisateurs d'un domaine (Figure 15). Cette page vise à offrir une vue d'ensemble claire et intuitive des projets, ainsi que la possibilité d'effectuer des actions de recherche, de suppression et de restauration.

L'objectif de cette première partie du développement était de fournir un outil centralisé permettant aux administrateurs de surveiller l'ensemble des projets géotechniques associés à leur domaine. La page de gestion des projets a été conçue de manière à afficher de manière structurée les informations essentielles de chaque projet, à savoir le nom du projet, sa date de modification, son statut (en cours, terminé, supprimé) et la liste des utilisateurs participants au projet, identifiés par leurs avatars. De plus, chaque projet est accompagné d'un identifiant unique (ID) qui sert de lien vers la page détaillée du projet, offrant ainsi un accès rapide et pratique aux détails du projet.

L'une des fonctionnalités clés de cette page est la barre de recherche, qui permet aux administrateurs de filtrer les projets en fonction de plusieurs critères tels que le nom du projet, son ID ou les utilisateurs associés. Cette fonction de recherche vise à faciliter la navigation dans la liste des projets, en aidant les administrateurs à localiser rapidement les informations dont ils ont besoin.

Un aspect important à prendre en compte est la gestion des projets supprimés par leurs créateurs. Bien que les projets puissent être supprimés par les utilisateurs, ils sont toujours stockés en base de données pour des raisons de traçabilité. Ainsi, les administrateurs ont la capacité de restaurer ces projets pour l'ensemble des utilisateurs, ou de les supprimer définitivement de la base de données, en fonction des besoins de l'entreprise.

ID	Nom du projet	Dernière modification	Statut	Utilisateurs
1	Home	03/08/2023 15:08	Terminé	+3
2	user2 project	02/05/2023 15:25	En cours	
3	projet supprimé	01/06/2023 15:06	Supprimé	
4	Enseirb-Matmeca	03/08/2023 15:32	En cours	
5	Eglise	18/07/2023 09:57	En cours	
6	Lycée	04/05/2023 14:01	En cours	
7	London	05/05/2023 16:52	En cours	
8	Bordeaux	24/05/2023 11:38	Terminé	

FIGURE 15 – Page de gestion des projets géotechniques pour les administrateurs d'un domaine

Dans la section suivante, nous détaillerons en profondeur le processus de développement de cette fonctionnalité de gestion des projets géotechniques, en mettant en évidence les choix de conception, les technologies utilisées et les défis rencontrés tout au long de la mise en œuvre.

5.1.2 *Création de la Table "Projets" dans la Base de Données :*

La première étape du développement de la gestion des projets a été de créer une nouvelle table dans la base de données pour stocker les informations spécifiques à chaque projet. Pour cela, nous avons utilisé le concept de "changelog" avec Liquibase, qui nous permet de définir les modifications de la structure de la base de données à l'aide de fichiers XML appelés "changelog". Nous avons créé un nouveau fichier XML de changelog dédié à la création de la table "projets", où nous avons spécifié les colonnes telles que l'ID du projet, le nom, la longitude, la latitude, le statut, etc. Ce fichier a ensuite été ajouté à la liste des fichiers du master-changelog.xml, qui orchestre l'exécution séquentielle de tous les changelogs.

```
<createTable tableName="project">
    <column name="id" type="bigserial" remarks="Identifiant du projet - auto-incrémenté">
        <constraints primaryKey="true" nullable="false"/>
    </column>
    <column name="domain_id" type="bigint" remarks="Identifiant du domaine">
        <constraints primaryKey="true" nullable="false"/>
    </column>
    <column name="name" type="varchar(80)" remarks="Nom du projet">
        <constraints nullable="false"/>
    </column>
    <column name="location" type="geography(point)">
        <constraints nullable="false"/>
        <remarks>localisation en coordonnée géographique longitude/latitude</remarks>
    </column>
</createTable>
```

FIGURE 16 – fichier XML de changelog dédié à la création de la table "projets"

5.1.3 Crédation d'un DTO, d'un Mapper, d'un Controller, d'un Service et d'un Repository

Pour gérer les opérations liées aux projets, nous avons suivi une approche structurée en utilisant le modèle MVC (Modèle-Vue-Contrôleur). Voici le rôle de chaque composant :

DTO (Data Transfer Object) : Un DTO est une classe simple qui transporte les données entre les couches de l'application. Dans notre cas, nous avons créé un DTO "ProjetDto" qui reflète la structure de la table "projets" en termes de champs.

Mapper : Le Mapper est responsable de la conversion entre le DTO et le modèle (entité) de la base de données. Il facilite la transformation d'un ProjetDto en une instance de la classe Projet et vice versa.

Controller : Le Contrôleur est chargé de gérer les requêtes HTTP entrantes et d'appeler les méthodes appropriées du service. Nous avons créé un ProjetController qui expose les endpoints pour la création, la récupération, la mise à jour, la suppression et la restauration des projets.

Service : Le Service contient la logique métier de l'application. Nous avons mis en œuvre un ProjetService qui encapsule la gestion des projets, en appelant les méthodes du Repository et en effectuant des opérations complexes si nécessaire.

Repository : Le Repository est une interface qui définit les opérations de base pour interagir avec la base de données. Dans notre cas, le ProjetRepository propose des méthodes pour effectuer des opérations de CRUD (création, lecture, mise à jour, suppression) sur les entités "projets".

Cette structure modulaire facilite la maintenance, la compréhension et l'extension du code. Chaque composant a un rôle spécifique, ce qui rend le code plus organisé et évite les redondances.

En résumé, le développement de la fonctionnalité de gestion des projets a impliqué la création d'une table dédiée dans la base de données, l'implémentation de divers composants logiciels (DTO, Mapper, Controller, Service, Repository) pour gérer les opérations liées aux projets. Cette approche méthodique a permis d'intégrer efficacement cette fonctionnalité au sein de l'application tout en maintenant un code propre et structuré.

```

@Data
public class ProjectDto {
    private Long id;
    @Size(max = 80)
    private String name;
    private double latitude;
    private double longitude;
    private Long domainId;
    @Size(max = 80)
    private String companyReference;
    private ProjectStatusDto status;
}

@Table(name = "project", schema = "public")
@IdClass(ProjectId.class)
public class Project implements Persistable {
    @Id
    @Column
    private Long id;

    @ToString.Exclude
    @Id
    @ManyToOne
    @JoinColumn(name = "domain_id", updatable = false, insertable = false)
    private Domain domain;

    @NotNull
    @Column
    private String name;
}

```

FIGURE 17 – Crédation d'un Dto et d'un modèle

5.1.4 Implémentation de l'Interface Utilisateur pour la Gestion des Projets et Projets Supprimés

Au niveau du frontend, la fonctionnalité de gestion des projets est mise en œuvre à travers deux fichiers importants qui assurent l'affichage et l'interaction avec les données. Cette partie clé de l'application permet à l'administrateur du domaine de visualiser, trier et interagir avec les projets créés par les utilisateurs du domaine. L'objectif est de fournir une interface conviviale et intuitive pour gérer efficacement l'ensemble des projets, y compris ceux qui ont été supprimés par les utilisateurs. Voici comment ces fichiers sont organisés et interagissent pour créer une expérience fluide pour les administrateurs.

5.1.5 Fichier `domainProjectTable.tsx` : Gestion de l'affichage des projets

Ce fichier correspond au composant DomainProjectTable, qui joue le rôle central dans l'affichage de la page de gestion des projets. Dans ce composant, nous utilisons la fonction map de React pour afficher un composant DomainProjectTableRow (Figure 18) pour chaque projet présent dans la liste des projets triés. Pour cela, nous définissons des constantes (Figure 19) telles que la liste des projets triés et la liste des utilisateurs. Ces constantes sont mises à jour à l'aide de leurs setters associés. Ce fichier gère également l'interaction avec le backend en important les contrôleur nécessaires.

```
<TableBody>
  {sortedProjects.map((project) => {
    return (
      <DomainProjectRow
        key={project.id}
        project={project}
        reloadProjects={loadProjects}
      />
    );
  })}
</TableBody>
```

FIGURE 18 – Appel du composant chargé de l'affichage de chaque ligne associée à un projet

```
const navigate = useNavigate();
const projectApi = new ProjectControllerApi();
const userApi = new UserAccountControllerApi();
const appCtx = useContext(AppContext) as ApplicationContext;
const [projects, setProjects] = useState<ProjectDto[]>([]);
const { currentDomain } = useContext(AppContext) as ApplicationContext;
const [sortedProjects, setSortedProjects] = useState<ProjectDto[]>([]);
const [projectsUsers, setProjectsUsers] = useState<ProjectUserDto[][]>([]);
```

FIGURE 19 – Constantes définies pour la gestion de la page des projets

Au moment du chargement du domaine et de l'arrivée sur la page de gestion des projets, nous utilisons un useEffect pour appeler une fonction init() (Figure 20 et 21). Cette fonction vérifie si le mode administrateur de l'utilisateur est activé. Si c'est le cas, elle charge les projets à l'aide de la fonction loadProjects() présentée à la figure 22.

```
useEffect(() => {
  init();
}, [appCtx.currentDomain, appCtx.isAdminModeActive]);
```

FIGURE 20 – Appel à la fonction d'initialisation des projets

```
const init = () => {
  if (appCtx.isAdminModeActive === null) return;
  if (!appCtx.isAdminModeActive) {
    navigate(UrlHelper.getUrlHome());
  } else {
    loadProjects();
  }
};
```

FIGURE 21 – Fonction d'initialisation des projets

Cette dernière fonction fait appel à FindAllProjectIncludingDeleted() du backend pour récupérer la liste complète des projets de la base de données, y compris ceux qui ont été supprimés par les utilisateurs mais non définitivement supprimés par l'administrateur. Ces projets sont filtrés en fonction de l'ID de domaine pour s'assurer que seuls les projets associés au domaine dont l'utilisateur est l'administrateur sont récupérés. Pour chaque projet, la fonction récupère également la liste des utilisateurs associés.

```

async function loadProjects() {
    if (appCtx.currentDomain?.id != undefined) {
        const projects = await projectApi.findAllProjectIncludingDeleted({
            domainId: appCtx.currentDomain?.id,
        });
        const updatedProjectsUsers: ProjectUserDto[][][] = [];
        for (const project of projects) {
            if (appCtx.currentDomain?.id != undefined && project.id != undefined) {
                const projectUsers = await userApi.findAllUserByProject({
                    projectId: project.id,
                    domainId: appCtx.currentDomain?.id,
                });
                updatedProjectsUsers.push(projectUsers);
            }
        }
        setProjects(projects);
        setSortedProjects([...projects]);
        setProjectsUsers(updatedProjectsUsers);
    }
}

```

FIGURE 22 – Fonction loadProjects permettant le chargement depuis la base de donnée des informations de l'ensemble des projets

5.1.6 Fichier *domainProjectTableRow.tsx* : Gestion de l'affichage d'une ligne de projet

Ce fichier correspond au composant DomainProjectTableRow, qui gère l'affichage de chaque ligne correspondant à un projet. Comme pour domainProjectTable.tsx, nous utilisons le useEffect pour appeler une fonction init() qui peut être observée dans la figure 23. Cette fonction récupère les informations spécifiques à chaque projet (Figure 24), telles que la date de mise à jour, le nom et la liste des utilisateurs. Ces informations sont obtenues à partir du backend en utilisant les fonctions appropriées pour accéder à la base de données.

```

const init = async () => {
    if (project.updatedAt != undefined) {
        const updatedAtDatetime = DateTime.fromJSDate(project.updatedAt).setLocale(
            i18n.language,
        );
        if (updatedAtDatetime.isValid) {
            setUpdatedAt(updatedAtDatetime.toLocaleString(DateTime.DATETIME_SHORT));
        }
    }
    if (appCtx.currentDomain?.id != undefined && project.id != undefined) {
        const projectUsers = await userApi.findAllUserByProject({
            projectId: project.id,
            domainId: appCtx.currentDomain?.id,
        });
        setProjectUsers(projectUsers);
    }
    setIsDeleted(project.deleted);
};

```

FIGURE 23 – Fonction d'initialisation des informations propres à un projet en particulier

```
const appCtx = useContext(AppContext) as ApplicationContext;
const { t } = useTranslation();
const theme = useTheme();
const [anchorEl, setAnchorEl] = useState<null | HTMLElement>(null);
const menuOpen = Boolean(anchorEl);
const [updatedAt, setUpdatedAt] = useState('');
const [projectUsers, setProjectUsers] = useState<ProjectUserDto[]>();
const [isDeleted, setIsDeleted] = useState<boolean>();
const [openDialog, setOpenDialog] = useState(false);
```

FIGURE 24 – Définition des constantes dans lesquelles sont stockés les informations de chacun des projets

De plus, ce fichier gère l’interaction avec la barre de recherche pour filtrer les projets. Lorsque l’on appuie sur la touche entrée, la liste des projets est filtrée en fonction des caractères saisis dans la barre de recherche. Les projets dont le nom ou l’ID contient la chaîne de caractères saisie sont conservés, ainsi que ceux ayant des utilisateurs dont le nom ou le prénom correspond à la recherche. Ce filtrage prend en compte la cast pour garantir une correspondance précise.

```
const handleKeyDown = (event: { key: string }) => {
    if (event.key === 'Enter') {
        setOpen(false);
        setSortedProjects(
            projects.filter((project) => {
                if (project.name !== undefined && project.id !== undefined) {
                    const projectUsers = projectsUsers.find(
                        (users) => users[0].projectId === project.id,
                    );
                    if (projectUsers !== undefined) {
                        return (
                            projectUsers.some(
                                (user) =>
                                    user.firstname !== undefined &&
                                    user.name !== undefined &&
                                    (user.firstname + ' ' + user.name)
                                        .toLowerCase()
                                        .includes(value.toLowerCase()) ||
                                    project.name.toLowerCase().includes(value.toLowerCase()) ||
                                    project.id.toString().includes(value)
                            );
                        }
                    }
                }
            ) || return false;
        );
    }
};
```

FIGURE 25 – Fonction de filtrage des projets

Une fonction de tri a été implémentée (Figure 26) pour permettre le tri des projets par ordre croissant ou décroissant en fonction des colonnes spécifiques. Par exemple, il est possible de trier les projets en fonction de la date de mise à jour, du nom ou de l’ID du projet. Cette fonction de tri offre une expérience utilisateur améliorée en permettant à l’administrateur de visualiser les

projets selon ses préférences.

```
function getSortingFunction(columnIndex: number, direction: 'asc' | 'desc' | undefined) {
    const bigValue = direction === 'desc' ? -1 : 1;
    const smallValue = direction === 'desc' ? 1 : -1;

    if (columnIndex === 0) {
        return (a: ProjectDto, b: ProjectDto) => {
            if (a.id === undefined || b.id === undefined) return 0;
            return a.id > b.id ? bigValue : smallValue;
        };
    } else if (columnIndex === 1) {
        return (a: ProjectDto, b: ProjectDto) => {
            if (a.updatedAt === undefined || b.updatedAt === undefined) return 0;
            return a.updatedAt > b.updatedAt ? bigValue : smallValue;
        };
    } else if (columnIndex === 2) {
        return (a: ProjectDto, b: ProjectDto) => {
            if (a.deleted === undefined || b.deleted === undefined) return 0;
            return a.deleted > b.deleted ? bigValue : smallValue;
        };
    }
}
```

FIGURE 26 – Fonction de tri des projets

Si un projet a été supprimé (indiqué par le champ "isDeleted" en base de données), une option est affichée sur la ligne du projet pour restaurer ou supprimer définitivement ce projet. La fonction handleRestore(), présenté figure 27, est appelée pour la restauration, qui recherche le projet à restaurer, modifie la valeur de "isDeleted" et met à jour le projet.

```
const handleRestore = async () => {
    if (appCtx.currentDomain?.id !== undefined && project.id !== undefined) {
        const projectToRestore = await projectApi.findOneProject({
            id: project.id,
            domainId: appCtx.currentDomain?.id,
        });
        projectToRestore.deleted = false;
        projectToRestore.deletedAt = undefined;
        projectApi.updateProject({ projectDto: projectToRestore, id: project.id });
        setIsDeleted(false);
    }
    setAnchorEl(null);
    appCtx.refreshContext();
};
```

FIGURE 27 – Fonction de restauration d'un projet

Enfin, lorsqu'un projet doit être supprimé définitivement, la fonction deleteProject() du frontend est invoquée (Figure 28). Cela appelle la fonction deleteProject() du backend (Figure 29). Cette fonction du contrôleur effectue d'abord une vérification des droits à l'aide de canPerform() (explicitée à la figure 30) pour s'assurer que l'utilisateur est autorisé à effectuer cette suppression.

sion. Si les droits sont valides, elle appelle DeleteById() du service (Figure 31), qui à son tour supprime le projet de la table des projets en utilisant une fonction du repository. Cette opération inclut également la suppression du projet de la table AGS en utilisant le champ "agsProjId0" pour référencer le projet dans la table AGS.

```
const deleteProject = async () => {
  if (appCtx.currentDomain?.id != undefined && project.id != undefined) {
    await projectApi.deleteProject({ id: project.id, domainId: appCtx.currentDomain?.id });
  }
  setOpenDialog(false);
  reloadProjects();
};
```

FIGURE 28 – Fonction de suppression d'un projet au niveau du frontend

```
▲ BAUDRY Alexandre +4
@DeleteMapping("/{id}")
public void deleteProject(@PathVariable(name = "id") Long id, @RequestParam Long domainId)
    throws RightsException, BadDomainException, ForbiddenAccessException {
  String tokenId = UserInfoUtils.getCurrentUserToken(request);
  checkDomainRight.canPerform(tokenId, domainId, DomainRight.ADMIN, LicenseUtils.LicenseTypeEnum.FREE);
  projectService.deleteById(id, domainId);
}
```

FIGURE 29 – Fonction de suppression d'un projet appelée par le front au niveau du controller

```
public void canPerform(String tokenId, Long domainId, int right, LicenseUtils.LicenseTypeEnum requiredLicense)
    throws RightsException, BadDomainException, ForbiddenAccessException {
  int currentUserRights = hasDomainRight(tokenId, domainId, right);

  if ((currentUserRights & DomainRight.BACK_ADMIN.getId()) != 0) {
    return;
  }

  //If the user has the good rights on the domain, we need to check if he is currently on the domain except for the back admin users
  if (!userSessionService.isUserCurrentlyInDomain(domainId)) {
    throw new BadDomainException();
  }

  if (!userSessionService.hasLicenseRight(requiredLicense)) {
    throw new RightsException();
  }
}
```

FIGURE 30 – Fonction de vérification des droits de suppression d'un projet

```

BAUDRY Alexandre +2
@Transactional
public void deleteById(Long id, Long domainId) throws ForbiddenAccessException {
    ProjectDto projectDto = findByIds(id, domainId);
    Long agsProjId0 = projectDto.getAgsProjId0();
    if (!projectDto.getDeleted())
        throw new ForbiddenAccessException();
    projectRepository.deleteByIdAndDomain_Id(domainId, id);
    agsProjRepository.deleteByIdAgsProjId0(agsProjId0);
}

```

FIGURE 31 – Fonction de suppression d'un projet au niveau du service

Cette structuration des fichiers et cette interaction entre les composants frontend et backend permettent d'offrir une expérience utilisateur fluide et de garantir une gestion cohérente et complète des projets géotechniques au sein de l'application.

5.2 Développement d'une Interface Intuitive pour la Saisie de Données de Sondage

5.2.1 *Interface de Saisie de Données de Sondage*

Une fonctionnalité centrale mise en place est l'intégration d'une interface dédiée à la saisie de données de sondage. Ce composant, intégré à la page de sondage, offre aux utilisateurs la capacité de renseigner des informations spécifiques aux opérations de forage. Ils ont la possibilité de sélectionner différents types de conteneurs d'informations, tels que la progression temporelle du forage, les dimensions du forage en fonction de la profondeur, ainsi que des détails sur les fluides de forage en lien avec la profondeur.

Date et heure	forée [m]	Prof. tubage [m]	Prof. niveau d'eau [m]
06/08/2023 14:54	1	1	<input checked="" type="checkbox"/> Sec <input type="checkbox"/> Min
29/08/2023 15:30	5	2	<input checked="" type="checkbox"/> Sec <input type="checkbox"/> Min
02/09/2023 21:03	10	3	<input type="checkbox"/> Sec <input checked="" type="checkbox"/> Min
14/09/2023 17:13	15	14	<input checked="" type="checkbox"/> Sec <input type="checkbox"/> Min

FIGURE 32 – Page des informations spécifiques aux opérations de forage

5.2.2 Personnalisation des Données

À l'intérieur de chaque conteneur, les utilisateurs ont la possibilité d'ajouter plusieurs entrées d'informations personnalisées. Par exemple, pour suivre la progression du forage, ils peuvent enregistrer diverses données comme la date, la profondeur du forage, la profondeur de tubage, la profondeur du niveau d'eau, ainsi que des commentaires spécifiques à chaque entrée.

Informations particulières ⓘ					
AJOUTER UN TYPE +					
Progression du forage					
Date et heure	Prof. forée [m]	Prof. tubage [m]	Prof. niveau d'eau [m]	Sec	
06/08/2023 14:54	1	1		<input checked="" type="checkbox"/> Sec	<input type="button" value=""/>
29/08/2023 15:30	5	4		<input checked="" type="checkbox"/> Sec	<input type="button" value=""/>
02/09/2023 21:03	10	9	2	<input type="checkbox"/> Sec	<input type="button" value=""/>
14/09/2023 17:13	15	14		<input checked="" type="checkbox"/> Sec	<input type="button" value=""/>
+					

FIGURE 33 – Conteneur de la progression du forage

5.2.3 Validation Rigoureuse des Données

Pour préserver la cohérence des données et assurer une expérience utilisateur sans accroc, nous avons implanté des mécanismes de validation des informations entrées. Ces procédures s'assurent que les données sont cohérentes avant d'être enregistrées dans la base de données. Par exemple, dans le cas du suivi de la progression du forage, nous vérifions que les profondeurs augmentent en fonction des dates et que les valeurs de profondeur du forage et de tubage sont saisies correctement.

5.2.4 Réduction des Erreurs et Garantie de Qualité

L'implémentation de ces mécanismes de validation a notamment restreint les erreurs de saisie potentielles. En cas d'incohérences, les utilisateurs sont alertés, ce qui leur permet de rectifier rapidement les problèmes (Figure 34, 35, 36). Cette approche certifie la justesse des calculs futurs basés sur ces données et contribue à la production de résultats précis.

Cette fonctionnalité de saisie de données de sondage accroît l'efficacité et la précision du processus de collecte d'informations, tout en offrant un environnement convivial aux utilisateurs.

AJOUTER UN TYPE +

Progression du forage

Date et heure	Prof. forée [m]	Prof. tubage [m]	Prof. niveau d'eau [m]	
06/08/2023 14:54	1	1		<input checked="" type="checkbox"/> Sec <input type="button" value=""/> <input type="button" value=""/>
29/08/2023 15:30	10			<input checked="" type="checkbox"/> Sec <input type="button" value=""/> <input type="button" value=""/>
02/09/2023 21:03	5		2	<input type="checkbox"/> Sec <input type="button" value=""/> <input type="button" value=""/>
14/09/2023 17:13	15	14		<input checked="" type="checkbox"/> Sec <input type="button" value=""/> <input type="button" value=""/>
+				

Vérifiez la cohérence dates / profondeurs forées

Profondeur requise

FIGURE 34 – Conteneur de la progression du forage affichant les erreurs de saisie des dates et les oubliés de saisie des profondeurs

Fluide de forage

Prof [m]	Type de fluide	Taux de spoils [%]	Couleur des spoils
1 > 2	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
2 > 3	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
3 > 5	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
4 > 4,5	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
Les recouvrements sont interdits			
+			

FIGURE 35 – Conteneur du fluide de forage affichant des erreurs de recouvrement entre les profondeurs des échantillons

Fluide de forage

Prof [m]	Type de fluide	Taux de spoils [%]	Couleur des spoils
1 > 2	Eau <input type="text"/>	10 > 8	<input type="button" value=""/> <input type="button" value=""/>
Il ne doit pas y avoir de trou		Taux min doit être < à taux max	
2 > 3	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
Il ne doit pas y avoir de trou			
3 > 5	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
Il ne doit pas y avoir de trou			
7 > 9	Eau <input type="text"/>	>	<input type="button" value=""/> <input type="button" value=""/>
Il ne doit pas y avoir de trou			
+			

FIGURE 36 – Conteneur du fluide de forage affichant des erreurs pour des trous entre les profondeurs des échantillons et une incohérence des taux de spoils

6 Implémentation

L'interface destinée à la saisie de données de sondage repose sur une conception ingénieuse et conviviale. Au sein de la page dédiée aux informations spécifiques, un composant appelé "BoreholeExecutionSpecialInformation" est sollicité. Ce composant reçoit en argument une liste appelée "infoTypeList", comprenant les noms des différents conteneurs d'informations importés, tels que "Progression du forage", "Fluide de forage", "Ajout d'eau", etc., ainsi que son setter. Cette liste permet d'afficher uniquement les conteneurs choisis, et chaque ajout ou suppression de conteneur se traduit par une mise à jour de cette liste.

```
<BoreholeExecutionSpecialInformations
    infoTypeList={infoTypeList}
    setInfoTypeList={setInfoTypeList}
></BoreholeExecutionSpecialInformations>
```

FIGURE 37 – Composant des informations particulières d'un sondage

Chaque conteneur de données est associé à une table AGS, où les données saisies par l'utilisateur sont enregistrées. Pour définir quels conteneurs doivent être affichés lors du chargement du composant, la liste "infoTypeList" est préalablement initialisée avec les noms des composants à afficher. Cette initialisation s'effectue via une fonction "fetchTypes()" (Figure 38), qui récupère les informations de chaque table AGS associée à son conteneur, en utilisant les fonctions des contrôleurs correspondants. Ces contrôleurs trient ensuite les éléments de leurs tables AGS respectives, offrant ainsi au frontend des listes pré-triées en fonction de critères spécifiques (Figure 39).

```

async function fetchTypes() {
    if (boreholeContext.agsLoca.agsLocaId0 !== undefined) {
        const agsPtimList = await agsPtimControllerApi.findAllAgsPtimByParentId({
            parentId: boreholeContext.agsLoca.agsLocaId0,
        });
        const agsWaddList = await agsWaddControllerApi.findAllAgsWaddByParentId({
            parentId: boreholeContext.agsLoca.agsLocaId0,
        });
        const agsFlshList = await agsFlshControllerApi.findAllAgsFlshByParentId({
            parentId: boreholeContext.agsLoca.agsLocaId0,
        });
        const agsHdiaList = await agsHdiaControllerApi.findAllAgsHdiaByParentId({
            parentId: boreholeContext.agsLoca.agsLocaId0,
        });
        const infoTypeToRemove = [
            { list: agsPtimList, type: drillingProgress },
            { list: agsWaddList, type: waterAdded },
            { list: agsFlshList, type: drillingFlush },
            { list: agsHdiaList, type: drillDiameter },
        ];
        const updatedInfoTypeList = infoTypeList.filter((type) =>
            infoTypeToRemove.every(
                (infoType) => !infoType.list.length || type !== infoType.type,
            ),
        );
        setInfoTypeList(updatedInfoTypeList);
    }
}

```

FIGURE 38 – Fonction FetchTypes chargée de l’initialisation de la liste des conteneurs

```

▲ BAUDRY Alexandre
@Override
public List<AgsPtimDto> findByParentId(Long id0) {
    List<AgsPtimDto> AgsPtimListDto = agsMapper.mapToDtos(agsRepository.findAllByAgsLocaId0(id0));
    this.sortPtimList(AgsPtimListDto);
    return AgsPtimListDto;
}

1 usage ▲ BAUDRY Alexandre
private void sortPtimList(List<AgsPtimDto> tDtos) {
    List<AgsPtimDto> sortedList = tDtos.stream()
        .sorted(Comparator.comparingDouble((AgsPtimDto p) ->
            if (p.getPtimDtim() == null) {
                return Double.MAX_VALUE;
            } else {
                return p.getPtimDtim().getTime();
            }
        ).thenComparing(AgsPtimDto::getAgsLocaId0))
        .collect(Collectors.toList());
    tDtos.clear();
    tDtos.addAll(sortedList);
}

```

FIGURE 39 – Fonction de tri appelée au moment de la recherche de la liste des informations d’un conteneur

L’ensemble de ces listes de données de conteneurs est stocké dans une liste de listes, puis filtré pour ne conserver que les conteneurs contenant effectivement des données. Pour chaque type de conteneur d’information, un composant distinct a été créé. L’affichage de ces composants est conditionné par la présence du nom du conteneur dans la liste "infoTypeList".

Chaque composant de conteneur dispose d’un sous-composant dédié à l’affichage des lignes de données spécifiques à ce conteneur. Lorsque ce sous-composant est appelé, une fonction "initFields()" est déclenchée via un useEffect. Cette fonction charge l’ensemble des valeurs de la ligne de données du conteneur en question.

Prenons l’exemple du conteneur "Progression du forage", dont le modèle associé est appelé

"agsPtim". Cette fonction "initFields()" récupère la date du sondage, la profondeur, la base, la profondeur du niveau d'eau et le commentaire associé à la ligne. Toutes les lignes de ce conteneur sont stockées dans une liste appelée "orderedAgsPtimList", composée d'éléments "agsPtim".

```
function initFields() {
    if (agsPtim.ptimRem !== undefined) {
        setPtimRem(agsPtim.ptimRem);
    }
    if (agsPtim.ptimDtim !== undefined) {
        setPtimDtim(agsPtim.ptimDtim.toISOString().slice(0, -1));
    }
    if (agsPtim.ptimDpth !== undefined) {
        setPtimDpth(agsPtim.ptimDpth);
    }
    if (agsPtim.ptimCas !== undefined) {
        setPtimCas(agsPtim.ptimCas);
    }
    if (agsPtim.ptimWat !== undefined) {
        setPtimWat(agsPtim.ptimWat);
        setIsChecked(false);
    }
}
```

FIGURE 40 – Fonction d'initialisation des champs d'une ligne de donnée d'un conteneur

Lorsqu'une nouvelle ligne de données doit être ajoutée, un bouton déclenche une fonction "addAgsPtim". Cette fonction vérifie tout d'abord si une ligne non définie (identifiant égal à "undefined") existe déjà dans la liste. Si toutes les lignes sont déjà remplies, un nouvel élément "agsPtim" avec un identifiant "undefined" est ajouté, permettant visuellement à l'utilisateur d'ajouter une nouvelle ligne vide pour la remplir ultérieurement.

```
async function addAgsPtim() {
    const hasUndefinedId = orderedAgsPtimList.some(ptim => ptim.agsttimId0 === undefined);
    if (!hasUndefinedId) {
        const newAgsPtimToAdd = { agsttimId0: undefined };
        setOrderedAgsPtimList([...orderedAgsPtimList, newAgsPtimToAdd]);
    }
}
```

FIGURE 41 – Fonction d'ajout d'un nouvel élément agsPtim à la liste actuelle

Les valeurs saisies par l'utilisateur dans cette nouvelle ligne ou dans les lignes existantes subissent une série de tests pour vérifier leur cohérence. Pour le conteneur "Progression du forage", par exemple, on s'assure de la cohérence des dates par rapport à la profondeur du forage. Une fonction "getIsDateInError()" (Figure 42) est activée chaque fois qu'une valeur de date ou

de profondeur est modifiée. Cette fonction trie d'abord la liste actuelle des éléments "agsPtim" par date, puis vérifie qu'il n'y a pas de situation où la profondeur du forage est supérieure pour une date antérieure. Si une incohérence est détectée, un message d'erreur s'affiche, empêchant la mise à jour ou la création d'une nouvelle ligne de données. En l'absence d'erreur, une fonction "handleUpdateOrCreate()" (Figure 43) autorise la mise à jour ou l'ajout de nouvelles lignes dans la table AGS.

```
function getIsDateInError() {
    const sortedList = [...orderedAgsPtimList];
    sortedList.sort((a, b) => compareByValueAsc(a, b, (x) => x?.ptimDpth));
    for (let i = 1; i < sortedList.length; i++) {
        const current = sortedList[i];
        if (
            (agsPtim.ptimDpth !== undefined &&
                current.ptimDpth !== undefined &&
                agsPtim.ptimDtim &&
                current.ptimDtim &&
                agsPtim.ptimDtim > current.ptimDtim &&
                agsPtim.ptimDpth < current.ptimDpth) ||
            (agsPtim.ptimDpth !== undefined &&
                current.ptimDpth !== undefined &&
                agsPtim.ptimDtim &&
                current.ptimDtim &&
                agsPtim.ptimDtim < current.ptimDtim &&
                agsPtim.ptimDpth > current.ptimDpth)
        ) {
            setDateErrorMessage(t('ptim.error-message.date-error'));
            return true;
        }
    }

    setDateErrorMessage('');
    return false;
}
```

FIGURE 42 – Fonction de test de la validité des dates saisies

```
async function handleUpdateOrCreate(agsPtim: AgsPtimDto) {
    if (agsPtim.agsPtimId0 === undefined) {
        agsPtim.agsLocaId0 = agsLoca.agSLocaId0;
        createAgsPtim(agsPtim);
    } else {
        update(agsPtim);
    }
}
```

FIGURE 43 – Fonction de création ou de mise à jour d'un élément agsPtim

6.1 Conception et Développement de la Page d'Accueil

Au cours de mon stage, l'une des tâches que j'ai entreprises a été le développement de la page d'accueil de l'application web. La conception et le développement de la page d'accueil de l'application ont été abordés avec une approche moins technique et plus concise, tout en garantissant l'efficacité et la convivialité de l'interface. Cette partie du projet, bien que moins complexe en termes de développement, a permis d'explorer davantage le domaine créatif de la conception web. Cette page revêt une importance capitale puisqu'elle est la première interface que les utilisateurs rencontrent lorsqu'ils accèdent à l'application. Mon objectif était de créer une page d'accueil à la fois accueillante, conviviale et informative, reflétant l'identité visuelle de l'entreprise et fournissant des informations clés aux utilisateurs.

Pour ce faire, j'ai utilisé une combinaison de langages et de technologies web, notamment HTML, CSS et JavaScript. L'HTML a été utilisé pour structurer le contenu de la page, en définissant les différentes sections, les en-têtes, les textes et les images. Le CSS a été employé pour styliser la page et lui donner un aspect visuel attractif et cohérent. J'ai conçu des mises en page, défini des typographies et appliqué des effets visuels pour créer une expérience visuelle agréable et engageante.

Le JavaScript a été intégré pour ajouter des fonctionnalités interactives à la page. J'ai utilisé des scripts pour créer des animations, gérer les transitions entre les différentes sections et rendre la navigation plus fluide et intuitive. De plus, j'ai incorporé des éléments interactifs tels que des boutons permettant aux utilisateurs d'interagir avec la page et d'accéder facilement aux différentes fonctionnalités de l'application.

La conception de la page d'accueil a été réalisée en étroite collaboration avec l'équipe de conception graphique pour s'assurer de la cohérence avec la charte graphique de l'entreprise. Des retours réguliers ont été échangés afin d'ajuster les détails visuels et fonctionnels pour répondre aux besoins et aux attentes des utilisateurs.

La page d'accueil que j'ai développée est incluse dans les annexes du présent rapport de stage. Cette partie du rapport met en évidence mon implication dans la création d'une interface utilisateur attrayante et fonctionnelle pour l'application web, soulignant mon aptitude à travailler

avec des langages frontaux tels que HTML, CSS et JavaScript pour réaliser des expériences utilisateur convaincantes.

Passons maintenant à un aspect fondamental de la réussite de ce projet de fin d'études : la gestion des délais et la méthodologie de projet mise en œuvre. L'atteinte des objectifs fixés et la réalisation des développements exemplaires ont été étroitement liées à une approche méthodique et à une planification minutieuse. Dans cette section, nous explorerons en détail comment la gestion proactive du temps et l'application d'une méthodologie rigoureuse ont contribué à l'accomplissement réussi de ces développements, tout en maintenant une qualité optimale. L'analyse de la méthodologie utilisée fournira un éclairage sur les processus de travail, la collaboration interdisciplinaire et les pratiques qui ont permis de garantir le respect des échéances et la réalisation des objectifs fixés.

7 Respect des échéances

La gestion efficace du temps et le respect des délais constituent des éléments cruciaux pour la réussite d'un projet. Durant mon stage, nous avons mis en place une série de pratiques visant à assurer la réalisation des tâches en temps opportun et la livraison des fonctionnalités selon les échéances fixées. Cette section se divise en trois sous-parties pour explorer en détail nos approches : "Gestion du planning du projet", "Suivi de l'avancement et des délais", et "Retour sur l'organisation et la gestion du temps".

7.1 Gestion du planning du projet

Au sein de l'équipe de développement, nous avons adopté une méthodologie agile pour structurer le projet. Nous avons utilisé GitLab (Figure 44) pour la gestion des tâches de développement, où chaque fonctionnalité était représentée par un ticket. Les développeurs avaient la liberté de choisir les tickets sur lesquels travailler, en s'attribuant les tâches qui correspondaient à leurs compétences et à leur disponibilité. Cette flexibilité nous a permis de maximiser l'efficacité individuelle tout en maintenant une vision d'ensemble du projet.

Pour chaque ticket, un processus rigoureux a été suivi. Le développeur se créait une branche distincte pour travailler sur la tâche, évitant ainsi de perturber les autres développements en cours. Une fois la tâche terminée, une "merge request" était initiée, impliquant la revue du code par au moins deux développeurs. Cette approche a favorisé la collaboration, les retours constructifs et la diffusion des connaissances au sein de l'équipe.

7.2 Suivi de l'avancement et des délais

Pour garantir un suivi efficace de l'avancement, nous avons organisé une réunion quotidienne de 15 minutes à midi. Chaque développeur partageait ce qu'il avait accompli la veille et exprimait ses objectifs pour la journée à venir. Cette pratique nous a permis de maintenir une communication fluide, d'identifier rapidement les éventuels obstacles et d'ajuster les priorités si nécessaire.

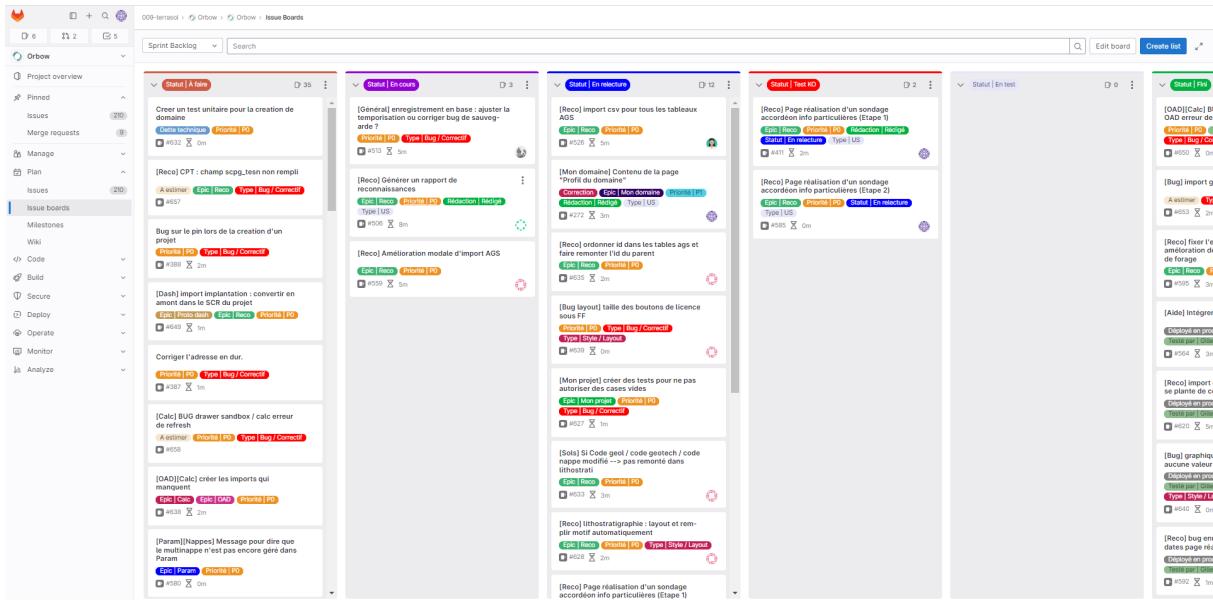


FIGURE 44 – Tableau de bord GitLab des fonctionnalités à implémenter, en cours d'implémentation ou implémentés

De plus, nous avons mis en place des réunions de fin de sprint, organisées toutes les deux semaines. Lors de ces réunions, nous évaluons les nouvelles fonctionnalités intégrées à l'application, nous nous fixions des objectifs pour le prochain sprint et nous surveillions les métriques d'avancement. Parmi ces métriques, nous utilisions des estimations de complexité en temps. Les développeurs participaient à des sessions d'estimation de complexité, en utilisant des nombres de la suite de Fibonacci pour évaluer le temps nécessaire à chaque fonctionnalité. Cette méthode nous a permis d'avoir une estimation réaliste de ce qui pouvait être accompli lors de chaque sprint.

7.3 Retour sur l'organisation et la gestion du temps

La mise en place de ces pratiques a eu un impact significatif sur notre capacité à respecter les échéances. L'utilisation de GitLab pour la gestion des tâches a facilité la distribution équilibrée du travail tout en favorisant la collaboration. La création de branches pour chaque tâche a maintenu un environnement de développement stable, tandis que les revues de code et les merge requests ont amélioré la qualité du code et renforcé l'entraide au sein de l'équipe.

En conclusion, la combinaison d'une méthodologie agile, d'un suivi régulier de l'avancement et de pratiques de gestion du temps efficaces a permis de respecter les échéances du projet. Cette approche a favorisé une collaboration transparente, des ajustements rapides et la livraison réussie de fonctionnalités.

8 Retour sur expérience

L'expérience de développement de l'application web de géotechnique a été marquée par une série de réussites, de défis et d'apprentissages. Cette section se décompose en trois sous-parties

pour analyser ces différents aspects : "Analyse des Réussites et des Difficultés Rencontrées lors du Développement", "Mesure de l'Impact de l'Application sur la Géotechnique et son Utilisation", et "Réflexion sur les Améliorations Potentielles et les Pistes d'Évolution".

8.1 Analyse des réussites et des difficultés rencontrées lors du développement

Lors du développement de l'application, plusieurs réussites ont marqué notre parcours. La mise en place d'une méthodologie agile a permis une distribution équilibrée des tâches, encourageant une collaboration fluide entre les développeurs. L'utilisation de GitLab pour la gestion des tâches et la revue de code a amélioré la qualité du code produit et renforcé l'entraide au sein de l'équipe. De plus, le respect des échéances, telles que la date de sortie de la version alpha à délivrer à Terrasol, et la planification réussie de la version bêta d'Orbow, reflètent notre engagement envers la gestion du temps.

Cependant, des défis ont également émergé. La coordination entre différentes fonctionnalités et composants a demandé des ajustements constants pour assurer leur intégration harmonieuse. La communication régulière au sein de l'équipe a été cruciale pour relever ces défis et maintenir la progression du projet.

8.2 Mesure de l'impact de l'application sur la géotechnique et son utilisation

L'impact de l'application web de géotechnique sur le domaine géotechnique ainsi que son utilisation interne chez Terrasol se révèlent d'ores et déjà positifs, bien que la version actuelle d'Orbow n'ait été mise à disposition que pour les membres de Terrasol. Cette phase de déploiement restreint au sein de Terrasol constitue une réussite en elle-même, offrant des retours d'utilisateurs et des perspectives essentielles pour améliorer et affiner l'application avant son lancement public.

Dans le contexte de Terrasol, l'application a révolutionné la gestion des projets géotechniques. La fonctionnalité de gestion des projets par les administrateurs a permis une visualisation plus approfondie et organisée des projets en cours, ce qui a conduit à une prise de décision plus éclairée. La simplification du processus de saisie de données de sondage a considérablement augmenté l'efficacité du travail, ce qui s'est traduit par des gains de temps significatifs et une réduction des erreurs potentielles.

En termes d'impact plus large sur le domaine de la géotechnique, bien que la version actuelle soit encore limitée à l'usage interne, les perspectives sont encourageantes. L'application répond à des besoins réels et urgents dans la gestion des projets géotechniques, et son développement progressif en tenant compte des retours des utilisateurs de Terrasol est un gage de sa pertinence pour le marché plus vaste.

L'application s'inscrit dans une évolution numérique du domaine de la géotechnique en fournant des outils modernes et intuitifs pour la gestion des projets et la collecte de données. L'impact potentiel sur la productivité, la précision et la collaboration au sein de la communauté géotechnique ne peut être sous-estimé, et le succès initial au sein de Terrasol souligne la pertinence de l'approche adoptée.

La version bêta prévue pour le 15 septembre ouvrira de nouvelles perspectives pour évaluer l'impact de l'application dans un environnement plus large et diversifié. Les retours et les analyses recueillis durant cette phase contribueront à façonner l'application en vue de son déploiement complet, confirmant ainsi son potentiel à influencer positivement la manière dont la géotechnique est pratiquée et gérée dans l'ère numérique.

8.3 Réflexion sur les améliorations potentielles et les pistes d'évolution

Bien que l'application ait été développée avec succès, des améliorations potentielles et des pistes d'évolution sont envisageables. L'ajout de fonctionnalités d'analyse avancée des données pourrait fournir des informations encore plus précieuses aux utilisateurs. La poursuite de l'optimisation de l'expérience utilisateur, reste une voie d'amélioration constante.

En conclusion, l'expérience de développement de l'application web de géotechnique a été enrichissante, combinant des succès et des défis inhérents à tout projet de cette envergure. L'impact positif sur la géotechnique et la gestion réussie du temps et des échéances témoignent de notre engagement envers la qualité et l'innovation. Les améliorations potentielles et les pistes d'évolution tracent une voie prometteuse pour l'évolution continue de l'application, garantissant sa pertinence à long terme dans le domaine de la géotechnique.

9 Conclusion

La réalisation de ce projet de développement de l'application web de géotechnique, baptisée "Orbow", a constitué une expérience riche en enseignements et en succès. Au travers des différentes phases du développement, des fonctionnalités exemplaires, de la gestion des délais et de l'évaluation de l'impact, ce projet a été l'occasion de mettre en pratique des compétences techniques et de gestion tout en répondant aux besoins spécifiques du domaine de la géotechnique.

9.1 Bilan du projet et des réalisations

Le projet a débuté avec la vision de créer une application intuitive, performante et flexible pour répondre aux exigences de la géotechnique. À travers le développement de deux fonctionnalités phares, la gestion de projets géotechniques et la saisie et analyse de données, une base solide a été posée pour répondre aux besoins des utilisateurs. La gestion des projets offre une meilleure visibilité et une prise de décision plus éclairée, tandis que la saisie de données simplifiée améliore l'efficacité opérationnelle. L'intégration des licences personnalisées offre une flexibilité adaptée aux besoins de chaque utilisateur.

9.2 Synthèse des principaux résultats obtenus

L'aboutissement du projet a été marqué par la réussite de la version alpha et la mise en place de la version bêta d'Orbow, conformément au planning établi. Les retours positifs des utilisateurs au sein de Terrasol attestent de la pertinence des choix de conception et des fonctionnalités développées. La création d'une interface intuitive, combinée à une gestion efficace des projets, a

facilité le travail quotidien des utilisateurs tout en contribuant à l'amélioration de la qualité et de la cohérence des données géotechniques.

9.3 Perspectives pour des travaux futurs et des améliorations possibles

Alors que le projet a atteint ses objectifs initiaux, des perspectives d'évolution et d'amélioration demeurent. Tout d'abord, l'élargissement de la portée de l'application au-delà de Terrasol, en la rendant disponible pour d'autres entreprises du secteur géotechnique, ouvre de nouvelles opportunités pour une plus grande utilisation et un impact plus large. L'intégration de retours et de suggestions d'utilisateurs provenant de différentes organisations enrichira davantage l'application.

L'amélioration continue de l'application, notamment l'ajout de nouvelles fonctionnalités en réponse aux besoins changeants du domaine de la géotechnique, est également essentielle. L'optimisation des performances, l'expansion des capacités de visualisation et l'intégration de technologies émergentes pourraient être des axes de développement futurs.

En conclusion, le projet de développement d'Orbow a été une expérience de collaboration fructueuse entre le domaine géotechnique et les compétences techniques. L'application incarne la fusion de la vision de Terrasol avec la mise en œuvre technique réussie. Les réussites obtenues jusqu'à présent et les perspectives d'amélioration et d'extension attestent de la pertinence du projet et de sa contribution à la modernisation du domaine de la géotechnique.

10 Bibliographie

<https://mui.com>

<https://react.dev>

<https://stackoverflow.com>

<https://docs.liquibase.com/home.html>

<https://gitlab.setec.com>

<https://orbow.com>

https://fr.wikipedia.org/wiki/Soci%C3%A9t%C3%A9_d%C2%7C%C3%A9tudes_techniques_et_%C3%A9conomiques

<https://dbeaver.io>

Annexes



A genuine operating system dedicated
to geotechnical engineering

[Discover the platform](#)



From data to calculation: a continuous and smooth process

The digital transition in geotechnics is not just a question of storage and value assigned to survey data. It is essential to link this data to the understanding and design of structures.

terrasol is a strong believer in building on the achievements of engineers. With this in mind, terrasol has developed **orbow**, your new collaborative platform.

AVAILABLE END OF 2022

Why choosing this platform?

— ITS ADVANTAGES —



All-in-one

A smooth & all-inclusive web platform for geotechnical project management, data and calculations.



Your geotechnical home

You need to save time and gain reliability in your everyday work. This is your home, not ours: we create your domain, you manage it and you can invite whomever you want.



Innovative

Our tools meet the needs of our engineers and are constantly improved in line with standards, working groups outcome, and user feedback.



Fiable & Proven

Our calculation tools are the same whether desktop and web versions. Our geotechnical engineers are our main users.



— ITS TOOLS —

PROJECT

Project database
Integrated GIS, search, storage of metadata
Import of background maps
Sectorisation

SURVEY

Import / export of ground investigation data
AGS compatible Interpretation (in situ and lab)
Automated reports

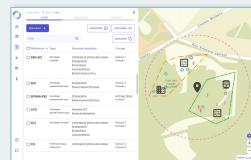
DECISION TOOLS

Decision support tools for representative values
Statistics and graphics
3D sections and stratigraphic blocks
Automated reports

PARAMETERS

Traceability of the interpreted parameters
Application of good engineering guidance practices
Proven calculation methods
Automatic calculations (sensitivity, stratigraphic variation, load cases)
Gateway to desktop software
New evolutionary tools to be included

CALCULATION



Keep up to date with all of orbow's news:

your email address

[Send](#)

A transparent, readable and flexible licensing system



Free

Consult data



Basic

Creating content in Survey module
+ Use simple tools



Pro

Basic + advanced analysis calculation and tools



Ultimate

Pro + support + access to desktop versions of calculation softwares

Please contact us for more information:

orbow@terrasol.com

Glossaire

API : Acronyme pour "Application Programming Interface". C'est un ensemble de règles et de protocoles qui permettent à différentes applications de communiquer entre elles.

Backend : La partie d'une application qui gère le traitement des données et les opérations en coulisses, souvent sans interaction directe avec l'utilisateur.

Branch : Une version parallèle du code source dans un système de contrôle de version comme Git, utilisée pour développer des fonctionnalités sans perturber le code principal.

DTO : Acronyme pour "Data Transfer Object". C'est un objet qui transporte des données entre les couches d'une application, souvent utilisé pour encapsuler les données lors de leur transfert entre le frontend et le backend.

Frontend : La partie d'une application qui interagit avec l'utilisateur, généralement via l'interface utilisateur.

Git : Un système de contrôle de version distribué largement utilisé pour suivre et gérer les modifications du code source.

Interface Utilisateur (IU) : La présentation visuelle et interactive d'une application avec laquelle les utilisateurs interagissent.

Merge Request : Une demande faite par un développeur pour fusionner les modifications d'une branche vers une autre dans un système de contrôle de version.

Méthode Agile : Une approche de gestion de projet qui favorise la flexibilité, l'itération et la collaboration continue pour répondre aux besoins changeants du projet.

Sprint : Une période de temps définie dans la méthodologie Agile pendant laquelle l'équipe de développement travaille sur un ensemble spécifique de fonctionnalités.

Ticket : Une tâche spécifique ou un élément de travail attribué à un développeur dans un système de suivi des problèmes ou de gestion de projet.

Version Alpha/Beta : Les étapes de test d'un logiciel avant sa sortie officielle. L'alpha est généralement testé en interne, tandis que la version bêta est testée par un groupe restreint d'utilisateurs externes.

Web Service : Un service en ligne qui permet à différentes applications de communiquer et d'échanger des données via le Web.