



ECOLE SUPERIEURE DE GESTION D'INFORMATIQUE ET DES SCIENCES

**RAPPORT DE  
STAGE**  
EN VUE DE L'OBTENTION DU DIPLOME DE LICENCE  
EN VUE DE L'OBTENTION DU DIPLOME DE LICENCE  
PROFESSIONNELLE  
PROFESSIONNELLE

**Domaine :** Sciences de l'ingénieur

**Mention :** Informatique Réseaux et Télécommunications

**Spécialité :** Architecture des Logiciels (AL)

**Gestion des projets de construction et suivi  
de l'évolution du chantier .**

Rédigé par : ELOHOUNKPON Dewanou Roméo

**Maître de stage:**  
**M. Martial AHADJI**

**Superviseur du rapport:**  
**M. Folly KOUÉVIDJIN**

## ANNEE ACADEMIQUE : 2022-2023

# Remerciements

“Avant tout développement, je tiens à adresser mes sincères remerciements à tous ceux qui, par leurs compétences et leurs soutiens m’ont aidé dans la réalisation de ce document.

Plus particulièrement, ma gratitude va à l’endroit de :

- ➔ **DIEU** le tout puissant de m’avoir donné la vie et la capacité de faire.
- ➔ Ma famille dont mon père M. **ELOHOUNKPOUN Z. Yao**, ma mère Mme **AGBOKODO Gbessito**, mes frères et mes sœurs.
- ➔ M. **Martial AHADJI**, mon maître de stage, Directeur informatique au sein de la société CleanSoft Solutions, qui m’a accueilli comme stagiaire dans son service. Il m’a beaucoup appris sur les enjeux d’un Architecte logiciel et sur ses missions au quotidien.
- ➔ Mon superviseur M. **Folly Chamberlin KOUÉVIDJIN**, Ingénieur logiciel pour le temps consacré à la relecture, aux conseils et recommandations, à l’élaboration et au bon déroulé de mon rapport.
- ➔ M. **Mike NADOR**, développeur logiciel à CleanSoft Solutions qui m’a apporté d’aides intellectuelles, morales et surtout pour son encouragement.
- ➔ M. **Josias ASSOUMANOU**, développeur logiciel à CleanSoft Solutions qui par son soutien et encouragement m’a montré les quelques lignes de vie d’un développeur à sa connaissance.
- ➔ M. **AMAÏZO ékoué Teddy** et Mme **KOGBE Déborah** qui sont respectivement Ingénieur logiciel et Ingénieur sécurité informatique pour le temps qu’ils m’ont accordé et de leur aide précieuse.

Je souhaite également adresser mes remerciements les plus sincères aux équipes pédagogiques et administratives de l’Ecole Supérieur de Gestion d’Informatique et des Sciences (ESGIS) qui ont facilité mes démarches et ont assuré un suivi tout au long de ma formation et mon stage.

Je ne finirai pas sans prendre un moment pour exprimer ma sincère gratitude envers toutes les autres personnes qui ont contribué à mon parcours de leurs manières différentes, même si je n’ai pas énuméré leur nom dans mon rapport initial.

Je vous remercie tous du fond du cœur pour votre implication et votre précieuse contribution à mon parcours professionnel y compris vous cher lecteur actuel.”

# Sommaire

Remerciements .....	ii
Sommaire .....	iii
Résumé .....	v
Abstract .....	vi
Glossaire .....	vii
Liste des figures .....	ix
Liste des tableaux .....	xi
Liste des participants au projet .....	xii
Introduction .....	1
Première partie : Cahier des charges .....	2
Chapitre 1 : Présentations .....	4
I.    Présentation de l'école .....	4
II.    Présentation du cadre de stage .....	5
Chapitre 2 : Thème de stage .....	7
I.    Présentation du sujet et étude du domaine .....	7
III.    Problématique du sujet .....	9
IV.    Intérêt du sujet .....	10
V.    Étude de l'existant .....	12
VI.    Critique de l'existant .....	12
Chapitre 3 : Proposition de solutions.....	14
I.    Spécification des solutions.....	14
II.    Proposition de solutions .....	16
Chapitre 4 : Choix de la solution .....	19
I.    Solution retenue .....	19
II.    Evaluation financière de la solution retenue.....	21
III.    Planning prévisionnel de réalisation .....	23
Deuxième partie : Analyse et conception .....	26
Chapitre 1 : Présentation des outils d'analyse de modélisation .....	28
I.    Présentation de la méthode analyses .....	28
II.    Outils et langage de modélisation .....	29

Chapitre 2 : Étude détaillée de la solution .....	30
I. Étude fonctionnelle.....	30
II.      Étude technique .....	50
III. Diagramme de modèle .....	61
Troisième partie: Réalisations et mise en œuvre .....	64
Chapitre 1 : Mise en œuvre .....	65
I. Choix matériels .....	65
IV. Choix logiciels .....	66
Chapitre 2 : Présentation de l'application .....	73
I. Présentation .....	73
II. Architecture de l'application .....	73
III. Mise en place de la base de données .....	75
IV. Sécurité de l'application .....	77
Quatrième partie 4 : Exploitation et utilisation .....	78
Chapitre 1 : Configuration matérielle et logicielle .....	79
I. Configuration matérielle .....	79
II. Configuration logicielle .....	80
III. Déploiement et suivi de l'application .....	81
Chapitre 2 : Utilisation .....	85
I. Plan de navigation .....	85
II. Présentation de principales interfaces de l'application .....	87
III. Présentation des états .....	91
Conclusion générale .....	94
Bibliographie .....	I
Webographie .....	II
Table des matières .....	III

# Résumé

BuildTracker est une application web et mobile de gestion de projets de construction qui simplifie la gestion et le suivi des projets pour les ingénieurs, entrepreneurs, chefs chantier et clients. Elle permet la gestion de projets avec ses étapes détaillées, le partage de fichiers, la génération de rapports, la communication entre les équipes via des discussions intégrées, et envoie des notifications pour les mises à jour importantes. Elle repose sur une architecture technologique moderne, utilisant des bases de données NoSQL, suivant un modèle MVC, et exploitant des langages de programmation avancé comme HTML, CSS, JavaScript, Node.js, Express.js et Nuxt.js. L'objectif est d'améliorer la gestion, la communication et la transparence des projets de construction, en réduisant les retards et en optimisant la productivité. Actuellement en développement, BuildTracker vise à devenir une solution complète pour la gestion de projets de construction. Bien que certaines limites subsistent, notamment l'intégration et la dépendance à Internet. Les perspectives futures envisagent l'intégration de la gestion des ressources, des analyses prédictives et des améliorations continues de l'interface utilisateur pour renforcer l'efficacité et la convivialité de l'application.

# Abstract

BuildTracker is a web and mobile building project management application that simplifies project management and tracking for engineers, contractors, site managers and customers. It enables project management with detailed milestones, file sharing, report generation, communication between teams via integrated discussions, and sends notifications for important updates. It is based on a modern technological architecture, using NoSQL databases, following an MVC model, and exploiting advanced programming languages such as HTML, CSS, JavaScript, Node.js, Express.js and Nuxt.js. The aim is to improve the management, communication and transparency of building projects, reducing delays and optimizing productivity. Currently under development, BuildTracker aims to become a complete solution for managing building projects. Although some limitations remain, notably integration and dependence on the Internet. Future prospects envisage the integration of resource management, predictive analytics and ongoing improvements to the user interface to enhance the application's efficiency and user-friendliness.

## Glossaire

**API** : Application Programming Interface - Un ensemble de règles et de protocoles qui permettent à différentes applications logicielles de communiquer entre elles.

**CPU** : Central Processing Unit - L'unité centrale de traitement d'un ordinateur, responsable de l'exécution des instructions d'un programme.

**CSS** : Cascading Style Sheets - Un langage de feuilles de style utilisé pour définir la présentation visuelle des pages web.

**DNS** : Domain Name System - Un système de noms de domaine qui traduit les noms de domaine en adresses IP pour permettre la navigation sur Internet.

**ESGIS** : Ecole Supérieur de Gestion d'Informatique et des Sciences

**HTML** : Hypertext Markup Language - Un langage de balisage utilisé pour créer le contenu et la structure des pages web.

**HTTP** : Hypertext Transfer Protocol - Un protocole de communication utilisé pour le transfert de données sur le World Wide Web.

**HTTPS** : Hypertext Transfer Protocol Secure - Une version sécurisée du protocole HTTP utilisée pour les transactions en ligne, généralement identifiable par un cadenas dans la barre d'adresse.

**IDE** : Integrated Development Environment - Un environnement de développement logiciel complet qui offre des outils pour la programmation, le débogage et la gestion de projets.

**JSON** : JavaScript Object Notation - Un format léger d'échange de données souvent utilisé dans les applications web et les services web.

**JS** : JavaScript - Un langage de programmation pour le développement web.

**MVC** : Modèle-Vue-Contrôleur - Un modèle d'architecture logicielle pour la conception d'applications.

**MLD** : Modèle Logique de Données - Une représentation abstraite des données d'un système, montrant les relations entre les différentes entités.

**NoSQL** : Not Only SQL - Un type de base de données flexible et non relationnel.

**OTP** : One-Time Password - Un mot de passe unique généré pour une seule utilisation, généralement utilisé dans le processus d'authentification à deux facteurs.

**PDF** : Portable Document Format - Un format de fichier largement utilisé pour la présentation de documents, qui préserve leur mise en page.

**RAM** : Random Access Memory - La mémoire vive d'un ordinateur utilisé pour le stockage temporaire de données en cours d'utilisation.

**SQL** : Structured Query Language - Un langage de programmation utilisé pour gérer et interagir avec les bases de données relationnelles.

**SSL** : Secure Sockets Layer - Un protocole de sécurité qui assure le chiffrement des données lors de leur transfert entre un navigateur web et un serveur.

**UML** : Unified Modeling Language - Un langage de modélisation standard utilisé pour visualiser, spécifier, concevoir et documenter les systèmes logiciels.

**UI** : User Interface - L'interface utilisateur, qui permet à un utilisateur d'interagir avec un logiciel ou un système.

**URL** : Uniform Resource Locator - Une adresse web qui identifie la localisation d'une ressource sur Internet.

**UX** : User Experience - L'expérience globale de l'utilisateur lors de l'utilisation d'un produit ou d'une application, y compris ses émotions et ses impressions.

**VPN** : Virtual Private Network - Un réseau privé virtuel qui sécurise la connexion Internet d'un utilisateur en acheminant le trafic à travers un serveur.

## Liste des figures

Figure 1:Carte de localisation ESGIS-Avédji .....	4
Figure 2: Carte de localisation CleanSoft-Solutions .....	5
Figure 3: Organigramme de CleanSoft-Solutions .....	6
Figure 4: Diagramme de GANTT .....	25
Figure 5: Logo UML. ....	28

Figure 5: Logo Draw.io .....	29
Figure 6: Analyse stztistique des besoins de buildTracker .....	30
Figure 7: Diagramme de cas d'utilisation général de BuildTracker .....	33
Figure 8: Diagramme de cas d'utilisation _Authentification à 2 facteurs_BuildTracker.....	35
Figure 9:Diagramme de cas d'utilisation_Gérer Utilisateurs .....	37
Figure 10: Diagramme de cas d'utilisation_Gérer projets .....	39
Figure 11: Diagramme de cas d'utilisation_Gérer Etapes .....	41
Figure 12: Diagramme de cas d'utilisation_Gérer Evènements .....	42
Figure 13:Diagramme de cas d'utilisation_Gérer Fichiers .....	44
Figure 14:Diagramme de cas d'utilisation_Gérer Rapports .....	45
Figure 15: Diagramme de cas d'utilisation_Module de Discussion .....	47
Figure 16: Diagramme de cas d'utilisation_Module de Notifications .....	48
Figure 17: Diagramme de séquences_Authentification avec code OTP .....	50
Figure 18: Diagramme de séquences_Gérer Utilisateurs .....	52
Figure 19: Diagramme de séquences_Module de Notification .....	52
Figure 20: Diagramme de séquences Gérer Projets .....	53
Figure 21: Diagramme de séquences_Gérer Etapes .....	54
Figure 22:Diagramme de séquences_Gérer Evènements .....	55
Figure 23: Diagramme de séquences_Gérer Rapports .....	56
Figure 24: Diagramme de séquences_Module de Discussions .....	57
Figure 25: Diagramme d'activités_Authentification .....	58
Figure 26: Diagramme d'activités_Gérer Projet .....	59
Figure 27: Diagramme d'activités_Gérer Etapes .....	59
Figure 28: Diagramme d'activités_Gérer Evènements .....	60
Figure 29: Diagramme d'activités_Gérer Rapports .....	60
Figure 30: Diagramme d'activités_Module de Discussions et Notifications .....	61
Figure 31: Raffinement des classes .....	62
Figure 32: Diagramme de modèle des classes .....	63
Figure 33: Logo MongoDB .....	67
Figure 34: Logo NodeJs et Express Js .....	67
Figure 35: Logo postman .....	67
Figure 36: Logo Nuxt et Vue.js.....	68
Figure 37: Logo d'HTML, CSS, JS .....	69

Figure 38: Logo Tailwind CSS .....	69
Figure 39: Logo Dart et Flutter .....	70
Figure 40: Logo d'android studio. ....	70
Figure 41: Logo Figma .....	71
Figure 42: Logo Adobe Photoshop et illustrator .....	71
Figure 43: Logo VS Code .....	72
Figure 44: Logo GitHub .....	72
Figure 45: Architecture 3 tière de l'application (Vue blog.lecacheur.com) .....	73
Figure 46: Schéma MVC-Model View Controleur (slideplayer.fr/slide).....	74
Figure 47: Base de données MongoDB_ BuildTracker.....	76
Figure 46: Plan de navigation de BuildTracker .....	85
Figure 47: Page de connexion .....	87
Figure 48: Page de vérification .....	88
Figure 49: Tableau de bord .....	88
Figure 50: Page de création projet .....	89
Figure 51: Page de liste des projets .....	89
Figure 52: Page des étapes .....	90
Figure 53: (1) Page de suivi évolution et (2) médias d'un projet.....	90
Figure 54: (1)Pages documents et (2) page de profil .....	91
Figure 55: Erreur lors de la connexion avec un email .....	92
Figure 56: Message de bienvenue lorsque la connexion est établie .....	92
Figure 57: (1) Réception de code par email et (2) absence d'évènement pour un projet. ....	93

## **Liste des tableaux**

Tableau 1: les participants du projet et rapport.....**Error!**  
**Bookmark not defined.**

Tableau 2: Coût matériel.....	21
Tableau 3: Coût humain .....	22
Tableau 4: Coût totaux .....	23
Tableau 5: Planning prévisionnel de réalisation .....	23
Tableau 6: Détails explicatifs du cas d'utilisations général .....	35
Tableau 7: Choix matériels/Desktop .....	71
Tableau 8: : Choix matériels/Laptop.....	72

Tableau 9: Configuration matérielle .....	85
Tableau 10: Configuration logicielle .....	87

## Liste des participants au projet

*Tableau 1: les participants du projet et rapport*

Nom et prénoms	Rôle
<b>M. KOUVIDJIN Folly Chamberlin</b>	Superviseur
<b>M. AHADJI Martial</b>	Maître de stage
<b>M. ELOHOUNKPON Dewanou Roméo</b>	Responsable de la réalisation



# Introduction

L'industrie de la construction, un secteur de par sa complexité et sa diversité, requiert une gestion efficace et précise pour garantir la réalisation réussie des projets. La réalisation de chantiers, qu'ils soient publics ou privés, demande une coordination minutieuse, une communication transparente et une surveillance constante.

L'innovation technologique et l'effervescence numérique qui caractérisent notre époque ont engendré un éventail de défis et d'opportunités sans précédent. Dans ce contexte dynamique, notre projet de gestion de données et de collaboration, baptisé "BuildTracker," s'inscrit comme une réponse innovante aux besoins actuels des entreprises et des professionnels de construction. BuildTracker est bien plus qu'une simple application logicielle, c'est une solution complète qui révolutionne la manière dont les entreprises planifient, exécutent et supervisent leurs projets, du concept à la réalisation. Notre projet vise à répondre à l'exigence croissante de gestion de projets efficace, tout en intégrant des technologies de pointe pour faciliter la collaboration et la communication au sein des équipes.

Ce présent rapport, vise à présenter en détail les différentes facettes de BuildTracker, de son architecture robuste à ses fonctionnalités conviviales en passant par les technologies qui soutiennent son développement. Au long de ce rapport, nous aborderons chaque aspect essentiel du projet, en mettant l'accent sur les résultats obtenus, les défis relevés et les leçons tirées de cette aventure. Nous examinerons également les implications plus larges de notre travail dans le domaine de la gestion de projets et des technologies de l'information.

## Première partie : Cahier des charges

Le cahier des charges est un document essentiel qui définit les objectifs, les attentes et les exigences d'un projet. Il sert de guide pour orienter la réalisation d'une mission ou d'un projet spécifique. Dans cette première partie du cahier des charges, nous présentons les éléments clés nécessaires pour comprendre le contexte de notre projet. Nous commençons par une présentation de l'école et du cadre de stage, en mettant en lumière les détails administratifs, la mission de la structure d'accueil, la situation géographique, les prestations de services, l'organigramme de la société, ainsi que l'environnement matériel et logiciel de cette structure.

Ensuite, nous abordons le thème du stage en expliquant en quoi consiste notre sujet, quels problèmes il vise à résoudre, et pourquoi il revêt une grande importance. Nous analysons également l'état actuel des choses, afin de mieux comprendre le contexte dans lequel s'inscrit notre projet, et nous formulons une critique constructive de cet état existant.

Dans le troisième chapitre, nous passons à la proposition de solutions. Nous spécifions les exigences fonctionnelles et de sécurité de notre solution. Ensuite, nous présentons au moins deux options de solutions : la première, basée sur une solution déjà existante, et la seconde, qui décrit la solution que nous devons mettre en place. Pour chaque option, nous détaillons leur description, leurs avantages et inconvénients.

Enfin, dans le quatrième chapitre, nous sélectionnons la solution qui répond le mieux à nos besoins et objectifs. Nous procédons à une évaluation financière de cette solution, en considérant les coûts matériels, humains et totaux impliqués. De plus, nous établissons un planning prévisionnel pour la réalisation de ce projet, en définissant les principales étapes à franchir.

Cette première partie du cahier des charges pose les bases et les fondations de notre projet, en fournissant une vue d'ensemble complète de son contexte, de ses objectifs et des solutions envisagées pour le mener à bien.

## Chapitre 1 : Présentations

Dans ce chapitre se charge de la présentation initiale de l'environnement dans lequel s'inscrit le projet BuildTracker. Nous débutons en présentant l'institution éducative au sein de laquelle le stage s'est déroulé, éclairant ainsi le contexte académique de ce travail. Ensuite, nous nous plongerons dans les détails du cadre du stage, y compris les aspects administratifs, les missions de la structure d'accueil, la situation géographique, les prestations de services, et la configuration organisationnelle de l'entreprise. Ce chapitre permettra à tous de se familiariser avec les bases essentielles qui ont permis au projet BuildTracker de voir le jour.

## I. Présentation de l'école

L'Ecole Supérieur de Gestion d'Informatique et des Sciences (ESGIS) a été créé en l'an 1994 au Togo, en 2005 au Bénin et en 2012 au Gabon.

ESGIS est une École Supérieure privée agréée qui évolue sous l'égide du Ministère de l'Enseignement Supérieur et de la Recherche du TOGO, du BENIN et du GABON. École de formation théorique et pratique de haut niveau accréditée par le Conseil Africain et Malgache de l'Enseignement Supérieur (CAMES) dans plusieurs filières, elle a pour vocation de mettre à la disposition des entreprises privées et des services publics des pays d'Afrique en général, des cadres compétents, dynamiques, intègres et ayant un sens aigu des responsabilités en gestion comptable et financière, en gestion commerciale, en informatique, et dans les secteurs de technologies de pointe tels que la télécommunication et l'électronique.

Ci-dessous, voici la carte de location du site d'ESGIS Avédji.



Figure 1: Carte de localisation ESGIS-Avédji

## II. Présentation du cadre de stage

### a. Prise de contact et formalité administrative

La prise de contact initiale avec CleanSoft Solutions s'est effectuée par le biais d'une communication téléphonique et d'échanges de courriels. Les formalités administratives ont été rapidement traitées pour officialiser le début du stage.

### b. Mission de la structure d'accueil

CleanSoft Solutions, une startup dynamique et innovante, se positionne comme un acteur majeur dans les domaines de la technologie, de la science et de l'éducation. Sa mission consiste

à créer des solutions novatrices, notamment en développant des sites Internet vitrine et des applications web et mobile pour mettre en avant les marques et les produits de ses clients

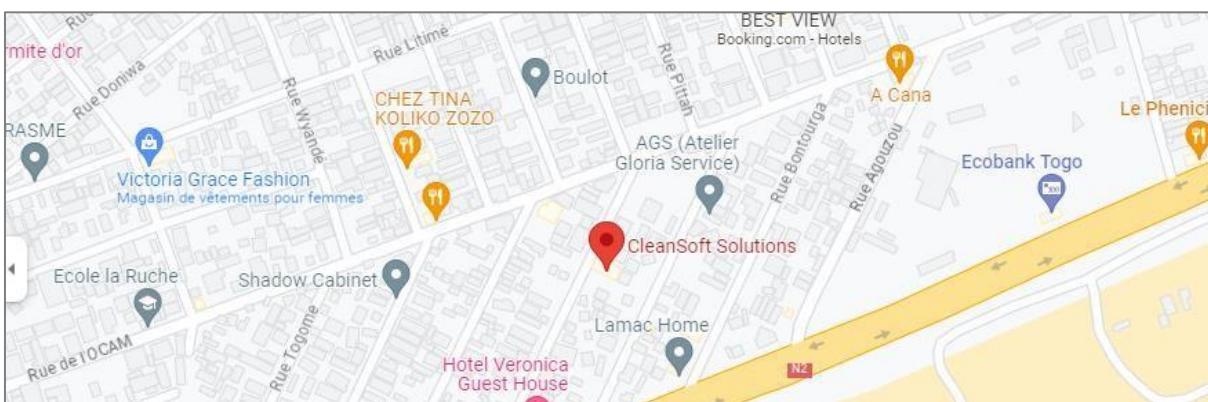
### c. Situation géographique

Située à Ablogamé, quartier Kotokoukondji, à proximité de l'hôtel Veronica, CleanSoft Solutions opère dans la ville de Lomé, au Togo. Son adresse complète est : 192 Rue Apounoukpa LOME TOGO.

Pour toute communication, vous pouvez les joindre : ✉

Au téléphone: +228 92 03 82 32

♦ Sur le site internet : <https://soltg.com/>



*Figure 2: Carte de localisation CleanSoft-Solutions*

#### **d. Prestation de services**

La startup CleanSoft Solutions propose une gamme de services qui incluent la conception, le développement et la mise en œuvre de projets innovants. Leur expertise se déploie également dans la création de plateformes et d'applications mobiles, et dans la sécurité informatique permettant aux clients de promouvoir leurs produits et services de manière efficace.

#### e. Organigramme de la société

Au sein de CleanSoft Solutions, la structure hiérarchique est bien établie. Elle est composée de différents services, chacun jouant un rôle clé dans le fonctionnement de l'entreprise. Voici ci-dessous le diagramme de la société

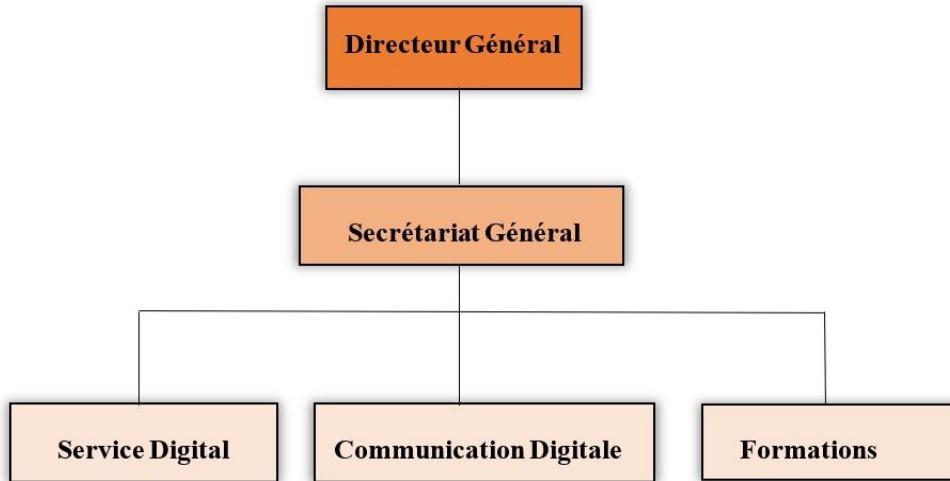


Figure 3: Organigramme de CleanSoft-Solutions

## f. Environnement matériel et logiciel de la structure d'accueil

CleanSoft Solutions travaille avec une infrastructure matérielle et logicielle moderne. Ils utilisent des équipements informatiques performants et des outils logiciels de pointe pour assurer le développement de projets et la création d'applications de haute qualité.

## g. Notre mission et apport dans la société lors du stage

Au cours du stage, notre mission était de contribuer activement aux activités et aux projets de l'entreprise tout en développant nos compétences en tant qu'architectes logiciels concepteurs d'applications. Nous avons assisté les membres de l'équipe dans leurs tâches quotidiennes, participé à des projets spécifiques, et avons cherché à apprendre et à comprendre le fonctionnement global de l'entreprise ainsi que la vie professionnelle. Nous avons également eu l'occasion de proposer des solutions pour améliorer les systèmes existants et avons respecté les politiques et les valeurs de l'entreprise. Notre stage nous a permis d'acquérir une expérience professionnelle précieuse et de développer nos compétences pour notre future carrière.

Le chapitre de présentation a établi les bases du contexte dans lequel le projet BuildTracker a été développé. En comprenant cet environnement initial, nous sommes mieux préparés à explorer en détail le développement et l'impact de ce système dans les chapitres suivants.

## Chapitre 2 : Thème de stage

Le deuxième chapitre de ce rapport plonge dans le cœur du sujet en présentant le thème du stage. Nous explorerons la nature du projet BuildTracker, en commençant par sa présentation

générale, la problématique sous-jacente, et l'intérêt majeur que revêt ce sujet. Cette section jettera les bases pour comprendre les objectifs, les résultats attendus, et l'état actuel de ce système novateur.

## I. Présentation du sujet et étude du domaine a.

### Présentation du sujet

En tant que technicien de chantier ayant une expérience pratique sur des projets de construction publics et privés, ainsi que des collaborations avec des maîtres d'œuvre et d'autres chefs compétents, nous avons aborder le contexte de ce projet en nous basant sur les défis réels que nous avons rencontrés sur les chantiers. Notre expérience nous a permis de cerner les problèmes concrets qui entravent souvent le bon déroulement des projets de construction, et c'est cette expertise terrain qui a motivé la création de notre application de gestion des projets de construction.

Dans ce contexte, nous comprenons les complexités de la gestion de multiples projets simultanément, la coordination des équipes sur le terrain, la planification minutieuse des étapes, la surveillance des ressources, et la communication constante avec les parties prenantes. Les retards, les malentendus, la gestion des fichiers (plans, devis, photos, vidéos...), les informations manquantes et la difficulté à suivre l'avancement global ont été des réalités que nous avons dû affronter.

C'est cette expérience concrète qui nous pousse à développer une solution qui non seulement résoudrait ces problèmes, mais qui apporterait également une efficacité de gestion et d'organisation accrue, une transparence et une meilleure communication tout au long du processus de construction.

Ainsi, notre démarche découle directement de notre expérience sur le terrain et de notre désir de fournir une solution pragmatique aux défis concrets auxquels sont confrontés les professionnels de la construction au quotidien.

### b. Etude du domaine de construction

Le domaine de la construction englobe un vaste éventail d'activités liées à la conception, à la planification, à la réalisation et à la gestion de projets de construction, qu'il s'agisse de bâtiments résidentiels, commerciaux, industriels ou d'infrastructures publiques. Voici une étude détaillée des principaux aspects du domaine de la construction :

#### Conception et Planification

- ❖ **Conception architecturale** : Cela englobe la création de plans, de schémas et de modèles pour les bâtiments, en tenant compte à la fois de l'esthétique et de la fonctionnalité.
- ❖ **Ingénierie structurelle** : Les ingénieurs structurels conçoivent les systèmes de soutien et de stabilité des bâtiments, en utilisant des calculs et des simulations pour garantir leur résistance et leur sécurité.
- ❖ **Conception technique** : Il s'agit de la planification des systèmes mécaniques, électriques, de plomberie et de ventilation d'un bâtiment.
- ❖ **Planification spatiale** : Définir l'utilisation efficace de l'espace à l'intérieur et à l'extérieur du bâtiment, en tenant compte des besoins des utilisateurs.

## Gestion de Projet

- ❖ **Gestion de projet** : Planifier, organiser, coordonner et superviser toutes les phases d'un projet de construction, en tenant compte des délais, du budget et des ressources.
- ❖ **Gestion des coûts** : Estimer, surveiller et contrôler les coûts tout au long du projet.
- ❖ **Gestion des risques** : Identifier et atténuer les risques potentiels pour le projet, qu'ils soient liés aux délais, aux coûts ou à d'autres facteurs.

## Construction

- ❖ **Préparation du site** : Préparer le terrain pour la construction, ce qui peut inclure l'excavation, le niveling et d'autres travaux préliminaires.
- ❖ **Gros œuvre** : La construction de la structure principale du bâtiment, y compris les fondations, les murs, les planchers et les toits.
- ❖ **Second œuvre** : Les travaux intérieurs tels que l'installation électrique, la plomberie, la pose de revêtements de sol, la peinture, etc.
- ❖ **Finitions** : Les touches finales pour rendre le bâtiment fonctionnel et esthétiquement plaisant.

## Sécurité et Réglementation

- ❖ **Sécurité sur le chantier** : Mettre en place des mesures de sécurité pour protéger les travailleurs et les visiteurs sur le site de construction.
- ❖ **Normes et réglementations** : Se conformer aux codes du bâtiment, aux réglementations locales et nationales en matière de construction et d'urbanisme.

## Gestion des Parties Prenantes

- ❖ **Communication** : Entre les différentes équipes de projet, les clients, les sous-traitants et les parties prenantes externes.
- ❖ **Collaboration** : Travailler ensemble pour atteindre les objectifs du projet et résoudre les problèmes éventuels.

## Types de Projets de Construction

- ❖ **Résidentiel** : Maisons, immeubles d'appartements, complexes résidentiels.
- ❖ **Commercial** : Bureaux, centres commerciaux, hôtels, restaurants.
- ❖ **Industriel** : Usines, entrepôts, installations de fabrication.
- ❖ **Infrastructure** : Routes, ponts, tunnels, installations publiques (écoles, hôpitaux) Notons que le domaine de la construction est complexe et exigeant, impliquant une coordination minutieuse entre diverses parties prenantes et une expertise technique approfondie à chaque étape du processus. Les avancées technologiques, l'innovation.

## III. Problématique du sujet

La problématique au cœur de ce projet réside dans la gestion complexe des projets de construction, tant du point de vue des professionnels de la construction que des clients. Les chantiers de construction sont sujets à des défis tels que la coordination des équipes, la gestion des étapes et fichiers, le suivi précis de l'avancement, la communication fluide entre les parties prenantes et la transparence pour les clients.

La question centrale est donc : Comment mettre en place une solution efficace de gestion des projets de construction, à la fois pour les professionnels du secteur et pour les clients, afin d'optimiser la planification, l'exécution et le suivi des chantiers tout en garantissant une communication transparente et une visibilité en temps réel sur l'avancement des travaux ? Cette problématique soulève des enjeux de coordination, de suivi, de communication et d'efficacité globale, nécessitant une solution qui intègre la technologie pour répondre aux besoins spécifiques de ce domaine exigeant.

## IV. Intérêt du sujet

### a. Objectif

Les objectifs de ce projet de développement d'une application de gestion des projets de construction et de suivi du chantier par le client sont les suivants :

**Optimisation de la gestion des projets** : Mettre en place un système permettant une gestion plus efficace et structurée des projets de construction, en facilitant la planification des étapes et la gestion des ressources et fichiers.

**Amélioration de la communication** : Mettre en œuvre des fonctionnalités de communication en temps réel pour les équipes sur le terrain, ainsi qu'une interface de communication fluide entre les professionnels de la construction et les clients, favorisant une meilleure compréhension des besoins et des attentes.

**Transparence pour les clients** : Offrir aux clients une visibilité en temps réel sur l'état d'avancement de leurs projets, en leur permettant d'accéder aux informations, aux rapports d'avancement et aux images du chantier directement via l'application.

**Suivi précis de l'avancement** : Mettre en place un mécanisme de suivi précis de chaque étape du projet, permettant aux parties prenantes de suivre en temps réel les progrès et d'identifier rapidement tout écart par rapport au plan initial.

**Centralisation des données** : Créer une plateforme centralisée pour stocker et gérer toutes les informations liées aux projets, y compris les plans, les documents, photos, les rapports d'inspection et les communications, afin d'assurer une gestion documentaire efficace.

**Personnalisation et flexibilité** : Permettre la personnalisation de l'interface en fonction des préférences de chaque utilisateur, ainsi qu'une flexibilité pour s'adapter aux différents types de projets et aux besoins spécifiques de chaque client.

**Réduction des retards et des erreurs** : Réduire les retards et les erreurs grâce à une meilleure planification, une coordination renforcée et un suivi précis, contribuant ainsi à une livraison plus rapide et à la satisfaction des clients.

**Facilitation de la prise de décisions** : Fournir des données et des analyses en temps réel pour permettre aux professionnels de prendre des décisions éclairées et d'ajuster rapidement les plans en cas de besoin.

**Sécurité des données** : Garantir la sécurité des données sensibles et confidentielles, en mettant en place des mécanismes de protection et d'autorisation pour assurer que seules les personnes autorisées ont accès aux informations.

**Facilitation de la collaboration :** Faciliter la collaboration entre les membres de l'équipe et les différentes parties prenantes en fournissant une plateforme commune pour partager des informations, discuter des problèmes et collaborer de manière efficace.

L'atteinte de ces objectifs vise à transformer la gestion des projets de construction, en apportant une solution moderne et technologique qui répond aux besoins spécifiques du secteur, tout en améliorant l'efficacité opérationnelle, la satisfaction des clients et la qualité des résultats finaux.

## b. Résultats attendus

Les résultats attendus de ce projet de développement d'une application de gestion des projets de construction et de suivi du chantier par le client sont les suivants :

**Amélioration de l'Efficacité Opérationnelle :** La mise en place de l'application devrait conduire à une meilleure gestion, planification, exécution et suivi des projets, ce qui se traduira par une réduction des retards, des erreurs et une optimisation de l'utilisation des ressources.

**Optimisation du Suivi :** Le suivi précis de l'avancement des étapes et des événements de chaque projet permettra aux professionnels de prendre des mesures proactives pour résoudre les problèmes potentiels, évitant ainsi les retards coûteux.

**Centralisation des Données :** Les données liées aux projets seront organisées de manière centralisée, facilitant la recherche, la consultation et la gestion des informations importantes, ce qui améliorera l'efficacité opérationnelle.

**Transparence et Satisfaction Client :** En offrant aux clients une visibilité en temps réel sur l'avancement des projets, l'application renforcera la transparence et la confiance. Les clients seront plus satisfaits grâce à leur implication dans le processus et leur accès direct aux informations.

**Prise de Décisions Éclairées :** Les analyses et les rapports générés par l'application aideront les professionnels à prendre des décisions éclairées basées sur des données réelles et en temps réel.

**Réduction des Coûts :** L'optimisation des processus de gestion et de suivi des projets contribuera à réduire les coûts liés aux retards, aux erreurs et aux inefficiencies.

**Facilitation de la Collaboration :** L'application favorisera la collaboration entre les équipes, en leur fournissant un espace centralisé pour partager des informations, discuter des problèmes et travailler ensemble de manière efficace.

**Communication Fluidifiée :** Les fonctionnalités de messagerie en temps réel et de partage d'informations contribueront à une communication plus fluide entre les équipes de construction, les clients et les autres parties prenantes, favorisant une meilleure collaboration.

**Sécurité des Données :** Les mécanismes de sécurité mis en place garantiront que les données sensibles restent confidentielles et protégées contre tout accès non autorisé.

Ces résultats attendus reflètent l'impact positif que l'application pourrait avoir sur l'ensemble du processus de gestion de projets de construction, en apportant une plus grande efficacité, transparence et satisfaction tant pour les professionnels du secteur que pour les clients.

## V. Étude de l'existant

Les entreprises de construction ont longtemps utilisé des approches anciennes(traditionnelles) pour gérer leurs projets, en se basant sur des méthodes manuelles, des enregistrements dans les cahiers de notes, et des outils physiques et comme logiciel de gestion Excel. Cela impliquait généralement la gestion de documents papier, l'utilisation de tableurs pour le suivi et la planification, ainsi que des communications principalement réalisées via des appels téléphoniques et des réunions en personne.

## VI. Critique de l'existant

L'approche traditionnelle de gestion des projets de construction et de suivi de chantier présente plusieurs lacunes et limitations qui entravent l'efficacité, la transparence et la productivité. Voici quelques critiques majeures de l'existant :

**Communication Fragmentée et Retards :** Les méthodes de communication, telles que les appels téléphoniques, les e-mails et les réunions en personne, entraînent des retards dans la transmission d'informations critiques. La communication fragmentée entre les parties prenantes peut provoquer des malentendus, des erreurs et des retards dans l'exécution des tâches.

**Manque de Visibilité en Temps Réel :** L'absence d'une plateforme centralisée limite la visibilité en temps réel sur l'état d'avancement du projet. Les parties prenantes, y compris les

clients, ont du mal à obtenir des informations à jour sur les étapes de construction, les délais, les problèmes potentiels et conflits.

**Manque de Transparence :** Les clients et les parties prenantes n'avaient souvent qu'un accès limité aux informations sur l'avancement du projet. Cela pouvait entraîner des doutes et un manque de confiance.

**Gestion Manuelle des Données :** La gestion manuelle des données, y compris les plans, les rapports d'inspection et les documents de suivi, entraînent des erreurs, une désorganisation et une difficulté à retrouver rapidement les informations nécessaires. Les archives physiques peuvent également être sujettes à la perte ou à la détérioration des données et documents importants.

**Risque de Pertes de Données :** Les documents et les rapports physiques sont sujets à des risques de perte ou de destruction. Les pertes de données peuvent entraîner des retards, des litiges et des coûts supplémentaires pour récupérer ou reproduire les informations perdues.

**Manque de Suivi Précis :** Le suivi manuel de l'avancement du chantier peut être imprécis et difficile à maintenir. L'absence de mises à jour en temps réel entraîne des incohérences dans la planification et la gestion des ressources avec des retards considérables.

**Perte de Temps et de Productivité :** Les inefficacités causées par les communications manuelles, la recherche de documents physiques et la coordination complexe entraînent une perte de temps et une baisse de la productivité.

Les méthodes utilisées pour la gestion des projets de construction et de suivi de chantier sont confrontées à des limitations majeures qui entravent la collaboration efficace, la gestion des données et la prise de décisions éclairées. Une approche numérique moderne, telle que celle que nous proposons, vise à surmonter ces critiques en offrant une solution intégrée et conviviale adaptée aux défis du secteur de la construction.

Ce chapitre consacré au thème du stage nous a permis d'entrer de plain-pied dans le sujet de notre rapport en dévoilant les enjeux et les objectifs attendus. En identifiant la problématique qui a été à l'origine de ce projet, nous sommes maintenant prêts à explorer plus en profondeur l'état actuel de ce système et à proposer des solutions pour répondre à ces défis.

# Chapitre 3 : Proposition de solutions

Dans ce chapitre, nous allons présenter les différentes solutions proposées pour remédier aux problèmes de gestion des projets de construction. Ces solutions sont basées sur des méthodes modernes et des technologies numériques qui visent à améliorer l'efficacité, la communication et la coordination des projets de construction.

## I. Spécification des solutions a. Fonctionnelle

Notre système de gestion des projets et de suivi des projets construction devra comprendre les fonctionnalités suivantes :

- ♦ **Authentification et Autorisation :** Les utilisateurs devront s'authentifier pour accéder à l'application. Différents niveaux d'autorisation seront définis en fonction des rôles (administrateur, chef de projet, membre d'équipe), permettant un accès contrôlé aux fonctionnalités.
- ♦ **Création et Gestion des Projets :** Les utilisateurs auront la possibilité de créer de nouveaux projets en spécifiant les détails tels que le nom, la description et les dates clés. Ils pourront également attribuer des membres d'équipe aux projets et suivre leur progression.
- ♦ **Gestion des Étapes :** Chaque projet sera divisé en étapes distinctes représentant différentes phases. Les utilisateurs pourront créer, modifier et suivre l'avancement de chaque étape.
- ♦ **Gestion des Événements de Projet :** Chaque projet sera associé à des événements spécifiques tels que des réunions de suivi, des inspections et des livraisons. Les utilisateurs pourront planifier, suivre et gérer ces événements.
- ♦ **Tableau de Bord de Projet :** Un tableau de bord centralisé affichera une vue d'ensemble de tous les projets. Il fournira des informations sur l'état d'avancement, les délais, les tâches en cours et les événements à venir.
- ♦ **Calendrier des Événements :** Un calendrier affichera les dates clés des projets, y compris les jalons, les réunions et les dates limites. Les utilisateurs pourront ajouter, modifier et supprimer des événements.
- ♦ **Notifications :** Les utilisateurs recevront des notifications en temps réel pour les mises à jour importantes, les tâches attribuées, les événements à venir et les étapes achevées.

- ❖ **Messagerie Instantanée** : Une fonctionnalité de messagerie en temps réel permettra aux membres de l'équipe de communiquer facilement, en organisant les discussions par projet.
- ❖ **Gestion de la Documentation** : Les documents liés aux projets pourront être téléchargés, consultés et gérés. Une organisation efficace de la documentation facilitera la collaboration.
- ❖ **Personnalisation de l'Interface** : Les utilisateurs pourront personnaliser l'apparence de l'interface en fonction de leurs préférences.
- ❖ **Profil Utilisateur** : Chaque utilisateur disposera d'un profil où il pourra gérer ses informations personnelles, afficher les projets auxquels il est associé et consulter son historique d'activités.
- ❖ **Sécurité et Confidentialité** : Les données sensibles seront sécurisées, et l'accès aux fonctionnalités sera contrôlé en fonction des rôles et des autorisations des utilisateurs.

Ces fonctionnalités visent à résoudre les problèmes de gestion traditionnelle des projets de construction en offrant une plate-forme centralisée, transparente et collaborative pour toutes les parties prenantes avec une interface intuitive et facile à utiliser pour une expérience utilisateur optimale.

## b. Sécurité

La sécurité est une considération primordiale dans notre application de gestion des projets de construction. Nous mettrons en œuvre plusieurs mesures pour garantir la protection des données et la confidentialité des utilisateurs :

- ❖ **Authentification et Autorisation** : Les utilisateurs devront s'authentifier pour accéder à l'application. Différents niveaux d'autorisation seront définis en fonction des rôles, assurant que seuls les utilisateurs autorisés peuvent accéder aux fonctionnalités appropriées.
- ❖ **Chiffrement des Données** : Les données sensibles, telles que les informations de connexion et les données de projet, seront chiffrées pour empêcher tout accès non autorisé.
- ❖ **Protection contre les Attaques** : Nous mettrons en place des mécanismes de protection contre les attaques telles que les attaques par injection SQL et les attaques de contournement d'autorisation.
- ❖ **Sauvegardes Régulières** : Les données seront sauvegardées régulièrement pour éviter la perte de données en cas de défaillance du système.

- ❖ **Gestion des Sessions :** Les sessions des utilisateurs seront gérées de manière sécurisée pour empêcher les accès non autorisés à partir de sessions volées.
- ❖ **Audit et Journalisation :** Nous mettrons en place des mécanismes d'audit et de journalisation pour suivre les activités des utilisateurs et détecter tout comportement suspect.
- ❖ **Mises à Jour Régulières :** Nous veillerons à ce que tous les composants logiciels de l'application soient à jour avec les derniers correctifs de sécurité.
- ❖ **Test de Pénétration :** Avant le déploiement de l'application, nous effectuerons des tests de pénétration pour identifier et résoudre les vulnérabilités potentielles.
- ❖ **Politiques de Mot de Passe Fort :** Les utilisateurs seront encouragés à créer des mots de passe forts, combinant lettres, chiffres et caractères spéciaux.
- ❖ **Gestion des Identifiants :** Les identifiants et les informations de connexion seront stockés de manière sécurisée pour éviter les fuites d'informations.

En adoptant ces mesures de sécurité, nous visons à garantir que les données des projets de construction sont traitées en toute sécurité et que les utilisateurs peuvent avoir confiance en l'intégrité et la confidentialité de l'application.

## II. Proposition de solutions

### a. Proposition de Solution 1 : Odoo pour la Gestion des Projets

L'utilisation de la plateforme Odoo pour la gestion des projets. Odoo est un système ERP (Enterprise Resource Planning) qui offre une gamme complète de modules pour la gestion d'entreprise, y compris la gestion de projet. Voici les avantages et inconvénients de cette proposition :

#### ❖ **Avantages :**

Odoo propose une suite complète de modules intégrés, permettant de gérer divers aspects de l'entreprise, et la gestion des ressources humaines.

Gestion de Projet : Le module de gestion de projet d'Odoo offre des fonctionnalités telles que la planification des tâches, le suivi de l'avancement, la gestion des ressources et des coûts.

#### ❖ **Inconvénients :**

**Complexité Initiale :** La mise en place d'Odoo peut nécessiter une configuration initiale complexe pour s'adapter aux processus spécifiques de l'entreprise.

**Coûts** : Bien que le coût initial d'acquisition d'Odoo puisse être inférieur à celui d'un développement personnalisé, il existe des frais associés à la configuration, la personnalisation et la maintenance continue.

**Dépendance de la Plateforme** : L'utilisation d'Odoo signifie une dépendance continue vis-à-vis de la plateforme, ce qui peut présenter des risques en cas de problèmes techniques ou de non-conformité.

**Limites de Personnalisation** : Bien qu'Odoo soit personnalisable, certaines fonctionnalités peuvent ne pas répondre aux besoins spécifiques de l'entreprise, nécessitant des adaptations.

**Absence d'Application Mobile Client** : Comme mentionné précédemment, Odoo n'a pas d'application mobile dédiée pour les clients, ce qui limite leur accès et leur suivi en temps réel des projets de construction.

### **b. Proposition de Solution 2 : BuildTracker (Suivie des constructions).**

Nous proposons le développement d'une plateforme de gestion des projets de construction qui sera accessible via une plateforme web et une application mobile dédiée. Cette solution vise à faciliter la gestion complète des projets de construction, depuis la planification initiale jusqu'au suivi en temps réel sur le chantier. Voici les avantages et inconvénients de cette proposition :

#### **† Avantages :**

**Centralisation des Informations** : L'application permettra de centraliser toutes les informations liées aux projets, y compris les détails du projet, les membres de l'équipe, les étapes, les échéances et les ressources.

**Suivi en Temps Réel** : Les clients pourront suivre l'avancement du chantier en temps réel via l'application mobile, ce qui renforcera la transparence et la satisfaction client.

**Communication Améliorée** : L'outil de messagerie intégré facilitera la communication entre les membres de l'équipe et les parties prenantes, réduisant les malentendus et les retards.

**Planification et Gestion des Étapes** : L'application permettra de diviser chaque projet en étapes claires, facilitant ainsi la planification, la coordination et le suivi de l'avancement. **Stockage et Gestion des Documents** : Les documents pertinents tels que les plans, les rapports d'inspection et les devis seront stockés et organisés dans l'application pour un accès facile et rapide.

**Notification et Alertes** : Les notifications en temps réel informeront les utilisateurs des mises à jour importantes, des tâches à accomplir et des événements à venir.

**Personnalisation de l'Interface :** Les utilisateurs pourront personnaliser l'interface selon leurs préférences, améliorant ainsi l'expérience utilisateur.

### ⊕ Inconvénients :

**Complexité de Développement :** Le développement de la plateforme complète peut être nécessite des ressources techniques qualifiées et de niveau.

**Coûts :** La mise en place et la maintenance de l'application peuvent entraîner des coûts initiaux et continus, y compris les coûts de développement, d'hébergement et de support.

**Formation des Utilisateurs :** Les membres de l'équipe et les clients devront être formés à l'utilisation de l'application, ce qui peut nécessiter du temps et des efforts.

**Dépendance Technologique :** L'application dépendra de la technologie et des infrastructures, ce qui peut présenter des risques en cas de pannes ou de problèmes techniques.

Malgré les inconvénients potentiels, les avantages de solution ‘BuildTracker’ l'emportent largement, car elle permettra une gestion plus efficace, une meilleure communication et une visibilité accrue sur les projets de construction. Avec une planification minutieuse et une mise en œuvre soignée, ces inconvénients peuvent être gérés pour assurer le succès de l'application et améliorer l'ensemble du processus de gestion de projets de construction.

Ce chapitre a ouvert la voie à l'exploration de diverses solutions possibles pour répondre à la problématique du sujet posé. En spécifiant ces solutions sous les angles fonctionnel et sécuritaire, nous avons jeté les bases pour un choix éclairé. La présentation de deux propositions distinctes, avec leurs avantages et inconvénients respectifs, ouvre la porte à une analyse approfondie dans le but de sélectionner la solution optimale pour atteindre nos objectifs. Le prochain chapitre nous guidera dans ce choix critique.

## Chapitre 4 : Choix de la solution

Après avoir exploré deux solutions possibles dans le chapitre précédent, nous nous concentrerons ici sur le choix de la solution qui répondra le mieux à notre problématique. Nous examinerons

en profondeur les avantages et les inconvénients de ce choix, car il façonnera la direction future de notre projet. Nous aborderons également l'aspect financier de la solution retenue, en évaluant les coûts matériels, humains et totaux, pour avoir une vue d'ensemble complète de notre plan d'action. Enfin, nous élaborerons un planning prévisionnel de réalisation pour concrétiser notre choix.

## I. Solution retenue

La solution retenue pour la gestion et la suivie des projets de construction c'est le développement d'une application de gestion web et mobile personnalisée que nous avons nommé "**BuildTracker**" (Suivie des constructions).

BuildTracker est une application polyvalente qui s'adresse tant aux professionnels du bâtiment qu'aux particuliers désireux de suivre leurs projets de construction. Avec BuildTracker, nous développons une application web et mobile robuste pour simplifier la gestion de projets de construction, du début à la fin.

**Application Web** : BuildTracker offre une interface web complète pour une gestion détaillée des projets de construction. L'application web permet de créer, de planifier et de suivre des projets en détail, de gérer des équipes et de partager des informations essentielles avec toutes les parties prenantes. Destinée aux administrateurs et du bureau, elle est accessible à partir de n'importe quel navigateur web, offrant une grande flexibilité d'utilisation.

**Application Mobile** : L'application mobile BuildTracker est conçue pour accompagner les utilisateurs sur le terrain et les clients. Elle offre une expérience utilisateur fluide, des fonctionnalités de suivi des tâches en temps réel, la possibilité de capturer des photos et des documents directement depuis l'appareil mobile, bien plus encore et de permettre aux clients de suivre en temps réel l'évolution de leur chantier. Ainsi cette application sera compagnon de poche pour un accès facile aux projets où que l'utilisateur soit.

BuildTracker se distingue par son approche centrée sur l'utilisateur. L'interface est intuitive et conviviale pour permettre aux utilisateurs de prendre rapidement en main l'application.

Ce choix de développer une application personnalisée pour la gestion des projets de construction repose sur plusieurs raisons spécifiques qui tiennent compte des besoins et des contraintes générales dans le domaine de la construction, ainsi que des limitations de l'existant. Voici les principales raisons qui ont conduit à cette décision :

**Répondre aux Besoins Spécifiques :** Chaque projet de construction peut avoir des exigences uniques en termes de gestion. Une application personnalisée offre la flexibilité nécessaire pour adapter les fonctionnalités aux besoins spécifiques de chaque projet.

**Expérience Utilisateur Optimale :** En développant une application sur mesure, il est possible de concevoir une interface utilisateur intuitive et conviviale, alignée sur les attentes des professionnels de la construction. Cela garantit une expérience utilisateur positive et favorise l'adoption de l'application.

**Contrôle Total :** La création d'une application personnalisée donne un contrôle complet sur les fonctionnalités, la sécurité et l'évolution de la solution. Il n'y a pas de dépendance vis-à-vis des fonctionnalités préexistantes d'un logiciel tiers.

**Personnalisation Avancée :** Une application personnalisée permet d'intégrer des fonctionnalités spécifiques et des outils de reporting adaptés aux besoins. Les utilisateurs peuvent obtenir des informations précises et pertinentes pour leurs projets.

**Adaptation aux Changements :** Les besoins et les pratiques dans l'industrie de la construction évoluent. Une application personnalisée peut être mise à jour en conséquence pour s'adapter à ces changements.

**Évolutivité :** L'application pourra être étendue et améliorée au fil du temps pour répondre aux besoins futurs, garantissant sa pertinence à long terme.

**Amélioration de l'Efficacité :** Une application personnalisée peut être conçue pour automatiser les processus, réduire les tâches manuelles et optimiser la gestion des projets, ce qui améliore l'efficacité opérationnelle.

## Avantages pour les Clients

Les clients bénéficieront d'une :

- ➔ Visibilité totale sur l'avancement du projet,
- ➔ Communication transparente avec l'équipe de projet ➔ Gestion simplifiée des étapes et évènements.

Cette application améliore considérablement leur expérience en tant que parties prenantes actives dans le processus de construction.

## Avantages pour l'Entreprise

Les entreprises de construction tireront profit de cette application en termes :

- D'efficacité opérationnelle,
- Une optimisation des processus internes grâce à une meilleure organisation et coordination.
- Une amélioration de la réputation en offrant des services de gestion de projet plus efficaces.
- Une augmentation potentielle de la satisfaction client et de la fidélité.
- Une possibilité de se démarquer de la concurrence en proposant une solution technologique réputation en tant qu'entreprise innovante et axée sur la qualité.

Cette première section introduit de manière complète et engageante l'application de gestion des projets de construction que nous avons développée, en mettant en avant son rôle central dans l'optimisation de la gestion de projets de construction et dans la création d'une expérience client exceptionnelle.

## II. Evaluation financière de la solution retenue

Voici une évaluation financière préliminaire de la solution BuildTracker que nous proposons pour la gestion et la suivie des projets de construction, en tenant compte des coûts matériels et humains :

### a. Coût matériel

Le coût matériel englobe les dépenses liées à l'infrastructure physique nécessaire pour mettre en œuvre notre plateforme. Cela comprend les serveurs, les dispositifs de stockage, les équipements réseau et autres composants matériels.

Tableau 2: Coût matériel

Désignation	Caractéristiques	Prix estimatif (FCFA)
Serveurs d'application Web et mobile	CPU quad-core, 16 Go RAM, 500 Go SSD	522. 000

Serveur pour API & Base de Données	CPU hexa-core, 32 Go RAM, 1 To SSD	650. 000
Solutions de Sauvegarde et Sécurité	Sauvegarde des données, logiciels de sécurité	325. 000
Smartphones et Tablettes de Test	Divers appareils pour tests multi-plateformes	300. 000
<b>TOTAL Coût matériel</b>		<b>2.237.000</b>

## b. Coût humain

Tableau 3: Coût humain

Coût Conception			
Ressources	Durée estimée (Mois)	Coût Mensuel (FCFA)	Montant (FCFA)
Analyste du projet	02	180. 000	360. 000
Développeur Frontends	03	180. 000	540. 000
Développeur Backend	03	200. 000	600. 000
Designer UI/UX	1,5	150. 000	225. 000
Tests et déploiement	02	80.000	160. 000
Sécurité d'application	2,5	175. 000	438. 000
<b>TOTAL Conception</b>		<b>2.323. 000</b>	
Coût de Formation et de Maintenance			
Désignation	Coût horaire	Durée (heure)	Montant estimatif (FCFA)

Formation/Documentation	4.000	30	120. 000
Maintenance et Support	4.000	Indéfinie	80.000(/mois)
<b>TOTAL Coût de Formation et de Maintenance</b>			<b>200. 000</b>
<b>TOTAL Coût humain</b>			<b>2.523. 000</b>

### c. Coût totaux

Tableau 4: Coût totaux

Type de Coûts	Montant estimatif (FCFA)
Total Coût matériel	2. 237. 000
Total Coût humain	2. 523. 000
<b>Coûts totaux</b>	<b>4. 760. 000</b>

## III. Planning prévisionnel de réalisation

Le tableau consigné ci-dessous contient les étapes et tâches avec leur durées délimités

Tableau 5: Planning prévisionnel de réalisation

Etapes	Tâches	Durée (en J)	Début	Fin
Rencontre avec superviseur, pour suivis et validation	A- Rencontre avec superviseur, pour suivis et validation	12	16/06/23	Indéfini

Rédiger rapport d'évolution en continu et présentation au maître de stage	B- Rédiger rapport d'évolution en continu et présentation au maître de stage	58	26/06/23	14/09/23
Initialisation du thème	C- Préparation du thème	02	17/06/23	19/06/23
<b>Étude des besoins</b>		<b>14</b>	<b>19/06/23</b>	<b>02/07/23</b>
	D- Analyse des besoins	07	19/06/23	25/06/23
	E- Élaboration du cahier des charges	07	26/06/23	02/07/23
<b>Conception</b>		<b>28</b>	<b>03/07/23</b>	<b>30/07/23</b>
	F- Modélisation des données	14	03/07/23	17/07/23
	G- Conception UI/UX	14	10/07/23	30/07/23
<b>Développement, Tests et validation</b>		<b>91</b>	<b>31/07/23</b>	<b>02/11/23</b>
	H- Backend et Tests unitaires	56	31/07/23	29/09/23
	I- Frontend et Tests d'intégration	40	25/08/23	02/10/23
<b>Mise en production</b>	J- Mise en production	07	09/10/23	22/12/23
	K- Formation des utilisateurs	14	01/11/23	15/11/23
	L- Suivi et maintenance	Indéfini	À partir de la fin formation	

La figure ci-dessous montre le diagramme de gantt élaboré en fonction du tableau récapitulatif

## Première Partie : Cahier des charges

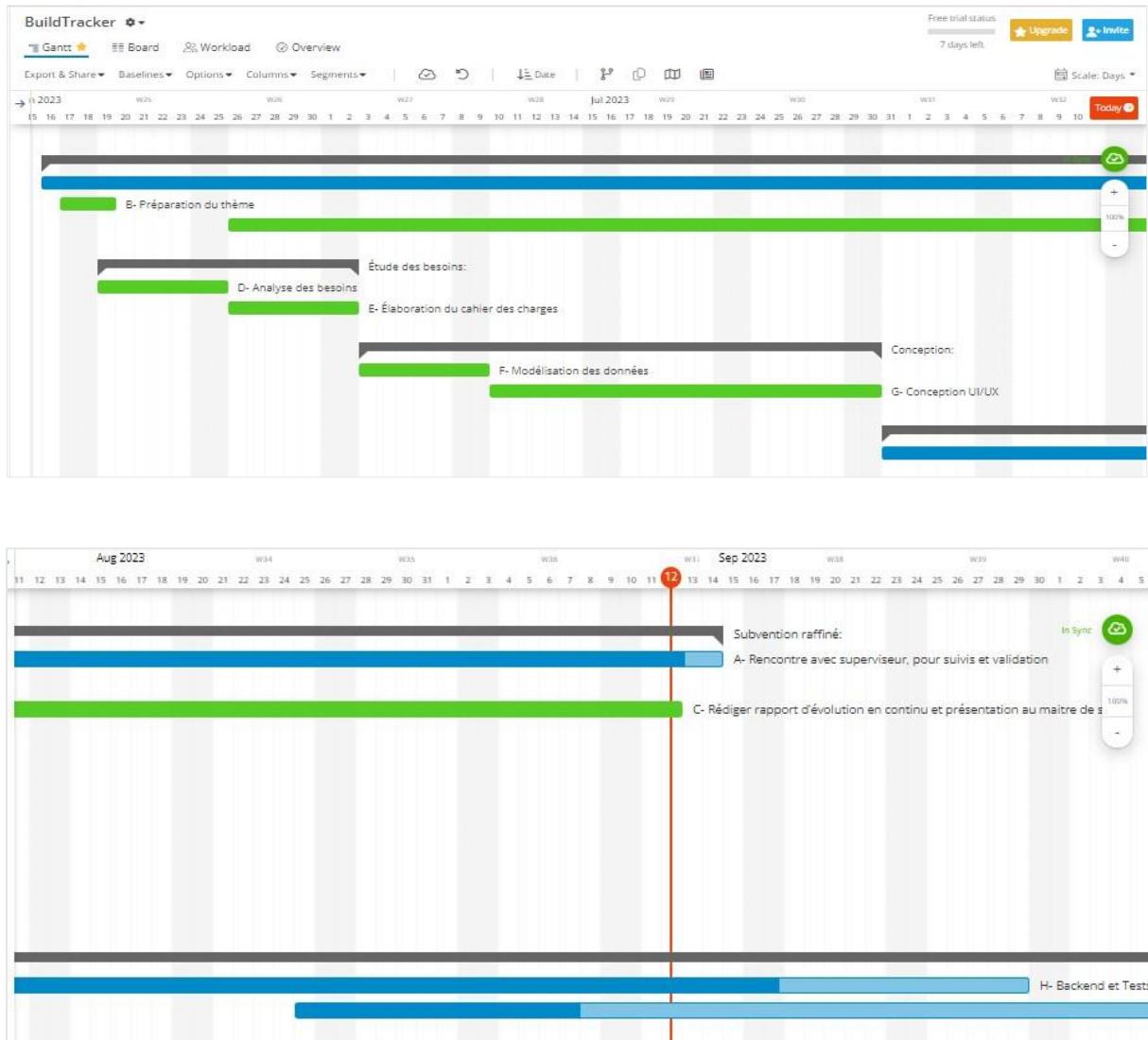


Figure 4: Diagramme de GANTT

Ce diagramme de GANTT ci-dessus mise en place partant du tableau de planning prévisionnel de réalisation, nous a permis de bien répartir les tâches et durées pour un travail bien ordonné.

Ce chapitre a été le théâtre d'une prise de décision cruciale pour notre projet BuildTracker. Après une analyse approfondie des différentes solutions envisageables, nous avons identifié celle qui répond le mieux à nos besoins et objectifs. L'évaluation financière nous a permis de prendre en compte les ressources nécessaires à la mise en œuvre de cette solution. Le planning prévisionnel de réalisation trace la voie que nous allons suivre pour concrétiser notre choix. Ce choix n'est que le début de notre aventure, mais il est fondamental pour la réussite de BuildTracker. Dans le prochain chapitre, nous entamerons la phase de mise en œuvre, mettant ainsi notre solution à l'épreuve du terrain.



## **Deuxième partie : Analyse et conception**

## Introduction

La deuxième partie de notre projet se concentre sur l'analyse détaillée et la conception de notre système de gestion des projets de construction. Cette phase est cruciale car elle permet de définir précisément les besoins fonctionnels et techniques de notre application, ainsi que de créer une structure solide pour son développement. Cette partie du projet est essentielle pour assurer la cohérence, l'efficacité et la réussite globale de notre solution.

Le chapitre 1 de cette partie se penche sur l'analyse, en identifiant en détail les fonctionnalités nécessaires pour répondre aux besoins des utilisateurs et des entreprises de construction. Nous allons examiner chaque aspect de notre application, de la création des projets à la gestion des étapes, en passant par les fonctionnalités de communication, de suivi et de personnalisation. Cette analyse approfondie nous permettra de définir clairement ce que notre application doit accomplir et comment elle doit le faire.

Dans le chapitre 2, nous aborderons la conception technique de notre application. Cela implique de choisir les technologies appropriées pour chaque composant, de créer des diagrammes et des modèles conceptuels pour représenter la structure et les interactions, et de planifier l'architecture générale de l'application. Cette phase est importante pour s'assurer que notre application sera bien conçue, évolutive et conforme aux meilleures pratiques de développement.

Enfin, dans le chapitre 3, nous présenterons un diagramme de classe UML. Ce diagramme fournira une vue d'ensemble visuelle de la structure de notre application, en mettant en évidence les classes clés, leurs attributs et leurs relations. Cette représentation visuelle est essentielle pour la création de la base de données solide pour notre système.

La deuxième partie de notre projet dresse les bases solides pour la création de notre application de gestion des projets de construction. Grâce à une analyse détaillée et à une conception technique minutieuse, nous veillons à ce que notre solution réponde parfaitement aux besoins identifiés et qu'elle soit prête pour la phase de développement.

## Chapitre 1 : Présentation des outils d'analyse de modélisation

Ce premier chapitre de la deuxième partie marque le début de notre voyage dans l'univers de BuildTracker. Pour comprendre et structurer notre projet, nous devons d'abord présenter les outils d'analyse et de modélisation que nous utiliserons tout au long de ce rapport. Ces outils serviront de fondement à notre démarche, nous permettant de visualiser, d'analyser et de planifier chaque étape de développement.

### I. Présentation de la méthode analyses

Pour élaborer une solution de gestion des projets tel que la nôtre, il est nécessaire de disposer d'une méthode d'analyse rigoureuse et cohérente. Cette méthode servira de boussole pour comprendre les besoins, les objectifs et les interactions clés au sein du projet. Dans cette optique, nous avons opté pour le langage de modélisation UML.

† UML (Unified Modeling Language)



Figure 5: Logo UML.

L'UML (Unified Modeling Language), en français "langage de modélisation unifié", est un langage graphique standard utilisé pour spécifier, visualiser, concevoir et documenter les différents aspects d'un système logiciel. Il offre un ensemble de notations graphiques pour représenter visuellement les composants, les interactions, les structures et les comportements d'un système.

L'UML est largement utilisé dans le domaine du développement logiciel pour créer des modèles qui permettent aux développeurs, aux concepteurs et aux parties prenantes de mieux comprendre et communiquer les caractéristiques d'un système. Il englobe plusieurs types de diagrammes, chacun servant à représenter un aspect particulier du système, comme les

diagrammes de classe pour la structure, les diagrammes de séquence pour les interactions, les diagrammes d'activité pour les flux de travail, etc.

En effet il joue un rôle essentiel dans le processus de développement logiciel en fournissant une manière standardisée et visuelle de représenter les idées, les exigences et les concepts, et permet de concevoir des systèmes plus robustes et compréhensibles.

## II. Outils et language de modélisation

Dans la mise en place de notre système, nous avons choisi d'adopter Draw.io comme outil d'analyse afin de modéliser les aspects clés du système.



Figure 6: Logo Draw.io

Cette décision a été prise en raison des avantages spécifiques que Draw.io offre afin de modéliser les processus, les données et les fonctionnalités de notre système de gestion de projets de construction. Cette décision est basée sur plusieurs raisons importantes qui sont spécifiquement adaptées aux besoins et aux objectifs de notre projet :

**Prise en Charge de la Modélisation UML et des Données :** Draw.io nous offre la capacité de modéliser à la fois les processus et les données grâce à une variété de diagrammes UML ainsi qu'à des schémas de bases de données.

**Flexibilité et Personnalisation :** Draw.io permet une personnalisation poussée de nos modèles, ce qui est essentiel pour répondre aux exigences spécifiques de notre projet.

**Documentation et Suivi :** La capacité de documenter nos modèles dans Draw.io nous permettra de garder une trace des décisions prises, des interactions entre les composants du système et de l'évolution de la conception tout au long du processus de développement.

Ce premier chapitre a établi les bases de notre démarche pour l'analyse et la modélisation de BuildTracker. Nous avons présenté les outils essentiels qui seront utilisés pour explorer et comprendre le projet dans son ensemble. Ces outils sont plus que de simples instruments, ils sont notre clé pour décortiquer les complexités du projet et pour présenter clairement nos idées. Le chapitre suivant nous emmènera au cœur du sujet, en présentant une vue d'ensemble de notre projet de gestion des projets de construction.

# Chapitre 2 : Étude détaillée de la solution

Dans ce deuxième chapitre, nous plongeons dans une étude approfondie de la solution proposée, BuildTracker. Après avoir établi les bases dans le chapitre précédent en présentant les outils d'analyse et de modélisation, nous nous concentrons désormais sur la solution elle-même en approfondissant l'analyse des fonctionnalités de notre système. Nous identifierons les besoins spécifiques des parties prenantes, définirons les cas d'utilisation, élaborerons les scénarios détaillés, et créerons des diagrammes pour illustrer ces aspects en fin de répondre de **qui fait quoi ? comment ? et quand ?**

## I. Étude fonctionnelle

### a. Les Fonctionnalités clés du système

Suite aux études statistiques des besoins et fonctionnalités clés, nous avons identifié en fonction de leur niveau de nécessité exprimé en pourcentage, afin de mettre en évidence les fonctionnalités les plus nécessiteuses. Voici le diagramme d'analyse statistique des Fonctionnalités :

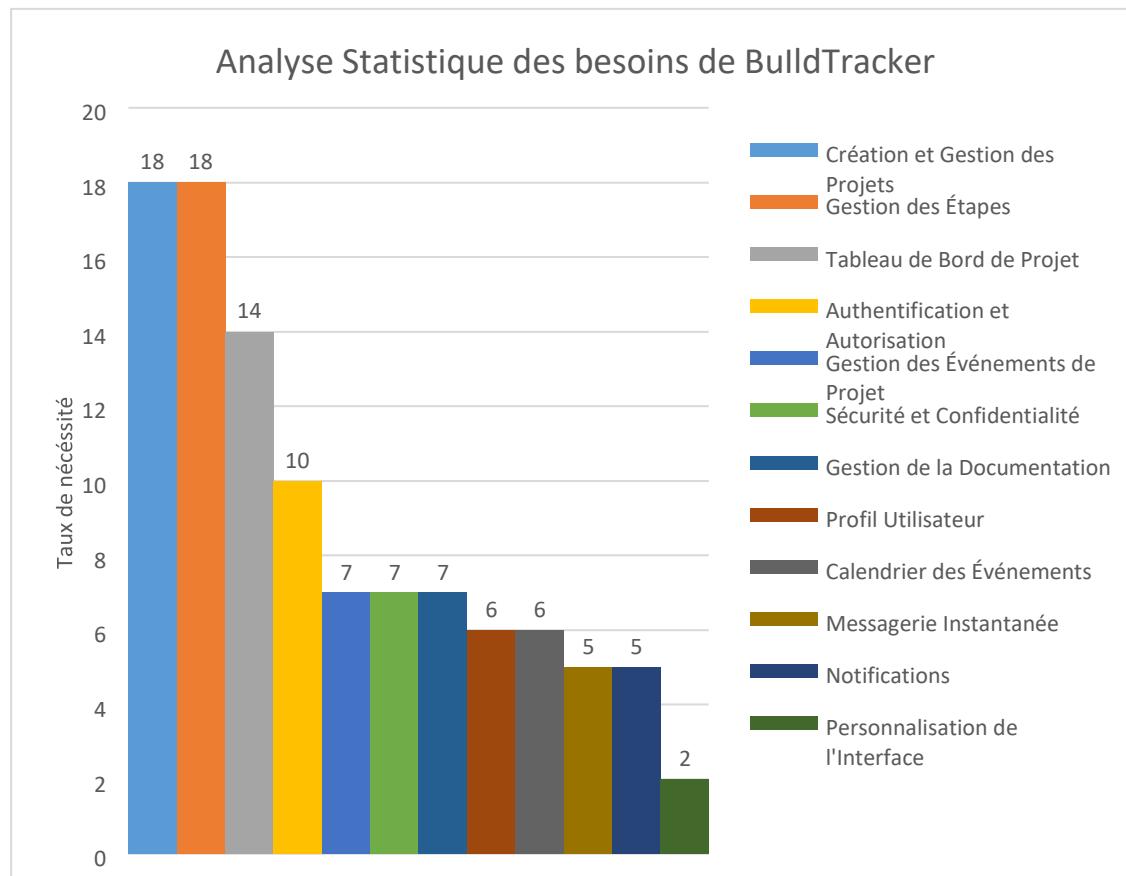


Figure 7: Analyse stztistique des besoins de buildTracker

**Personnalisation de l'Interface (2%)** : Les utilisateurs doivent pouvoir personnaliser l'apparence de l'interface en fonction de leurs préférences.

**Notifications (5%)** : Les utilisateurs doivent recevoir des notifications en temps réel pour les mises à jour importantes, les tâches attribuées, les événements à venir et les étapes achevées.

**Messagerie Instantanée (5%)** : Une fonctionnalité de messagerie en temps réel doit permettre aux membres de l'équipe de communiquer facilement et rapidement, en organisant les discussions par projet.

**Calendrier des Événements (6%)** : Un calendrier doit afficher les dates clés des projets, y compris les jalons, les réunions et les dates limites. Les utilisateurs doivent pouvoir ajouter, modifier et supprimer des événements.

**Profil Utilisateur (6%)** : Chaque utilisateur doit avoir un profil où il peut gérer ses informations personnelles, afficher les projets auxquels il est associé et consulter son historique d'activités.

**Gestion de la Documentation (7%)** : Les documents liés aux projets doivent pouvoir être téléchargés, consultés et gérés. Une organisation efficace de la documentation est nécessaire pour faciliter la collaboration.

**Sécurité et Confidentialité (7%)** : Les données sensibles doivent être sécurisées, et l'accès aux fonctionnalités doit être contrôlé en fonction des rôles et des autorisations des utilisateurs.

**Gestion des Événements de Projet (7%)** : Chaque projet doit être associé à des événements spécifiques, tels que les réunions de suivi, les inspections et les livraisons. Les utilisateurs doivent pouvoir planifier, suivre et gérer ces événements.

**Authentification et Autorisation (10%)** : On peut inscrire les et ils pourront se connecter à l'application. Différents niveaux d'autorisation doivent être définis pour les utilisateurs en fonction de leur rôle (administrateur, chef de projet, membre d'équipe).

**Tableau de Bord de Projet (14%)** : Un tableau de bord centralisé doit afficher une vue d'ensemble de tous les projets, y compris leur état d'avancement, leurs délais et les étapes.

**Création et Gestion des Projets (18%)** : Les utilisateurs doivent pouvoir créer de nouveaux projets, spécifier les détails du projet tels que le nom, la description et les dates clés, et affecter des membres d'équipe aux projets.

**Gestion des Étapes (18%) :** Chaque projet doit être divisé en étapes distinctes, représentant les différentes phases de progression. Les utilisateurs doivent pouvoir créer, modifier et suivre l'avancement de chaque étape.

En organisant les besoins fonctionnels en fonction de leur pourcentage de nécessité, nous mettons en évidence les fonctionnalités les plus cruciales pour répondre aux exigences des utilisateurs, en commençant par celles qui ont le plus élevé pourcentage jusqu'à celles qui ont le plus faible.

### **b. Les acteurs principaux de L'application**

Pour répondre aux besoins, aux exigences d'un large éventail d'utilisateurs et surtout la question de savoir qui fait quoi ? Chacun ayant un rôle spécifique dans le domaine de la construction, nous avons cédé en groupe et chaque groupe d'utilisateurs a des besoins spécifiques en termes de suivi, de gestion, de communication, et de rapports liés aux projets de construction. Voici un aperçu des principaux acteurs concernés par notre application :

#### **SuperAdmin/Admin :**

En tant que chefs ou directeurs d'entreprise, ces utilisateurs ont un rôle clé dans la gestion globale de l'entreprise de construction. Ils sont responsables de la prise de décisions stratégiques, de la gestion des ressources, et de la supervision de multiples projets.

- ➔ Chef ou directeur d'entreprise.
- ➔ Chef chantier, ...

#### **Techniciens :**

Les techniciens sont essentiels à la réussite des projets de construction. Ils se divisent en deux catégories : les techniciens de bureau qui sont responsables de la planification, de la conception, et de la gestion administrative des projets ; et les techniciens de surface.

- ➔ Techniciens de bureau : Architectes, Contrôleurs, banques.
- ➔ Techniciens de surface : Chefs maçons, menuisiers, ferrailleurs, plombiers, électriciens...

#### **Clients :**

Les clients peuvent être des institutions gouvernementales chargées de travaux publics ou des propriétaires privés investissant dans des projets de construction. Ils sont les initiateurs des projets et ont un intérêt direct dans leur bon déroulement.

- ➔ Institut chargée (Cas de travaux publics du gouvernement).
- ➔ Le propriétaire ou l'actionnaire de la construction (Cas de travaux privées).

### c. Diagramme de cas d'utilisation

Les diagrammes de cas d'utilisation que nous avons élaborés pour notre application de gestion des projets de construction décrivent les interactions entre les utilisateurs et le système pour dire court, **qui peut faire quoi ?** Ces diagrammes offrent une vue claire des principales fonctionnalités de l'application et des actions que les utilisateurs peuvent entreprendre, guidant ainsi le développement en alignant les exigences fonctionnelles avec les besoins des utilisateurs.

#### 1. Diagramme de cas d'utilisations général

La figure ci-dessous illustre le diagramme de cas d'utilisation général de notre application :

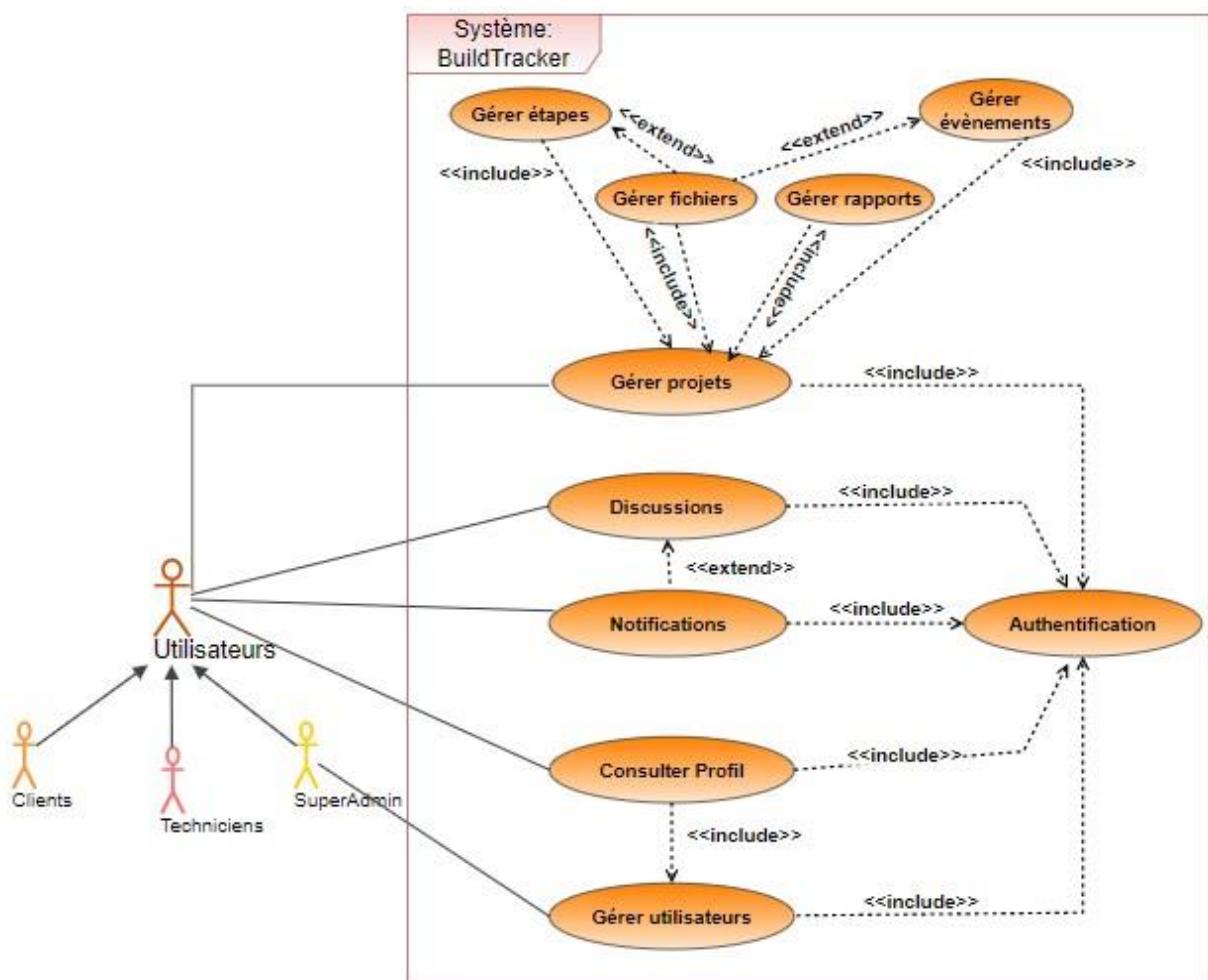


Figure 8: Diagramme de cas d'utilisation général de BuildTracker

Dans le tableau ci-dessous, nous avons consigné les actions de chaque utilisateur en passant par cas d'utilisation de l'application :

*Tableau 6: Détails explicatifs du cas d'utilisations général*

Cas d'utilisations	Actions	Acteurs
Authentification	S'identifier	Tous les utilisateurs
	Connexion	
Gérer utilisateurs	Créer	Admin, chef projet
	Modifier	Admin, chef projet
	Supprimer	Admin et chef projet
	Consulter profil	Tous les utilisateurs et clients (en fonction des droits)
	Affecter au projet ou étape	Admin, chef projet
Gérer projet	Créer projet	Admin, chef projet
	Modifier	Admin, chef projet et techniciens
	Supprimer	Admin et chef projet
	Consulter	Tous les utilisateurs et clients (en fonction des droits)
Gérer étapes	Créer projet	Admin, chef projet
	Modifier	Admin, chef projet et techniciens
	Supprimer	Admin et chef projet
	Consulter évolution	Tous les utilisateurs et clients (en fonction des droits)
Gérer évènements	Créer projet	Admin, chef projet
	Modifier	Admin, chef projet et techniciens

	Supprimer	Admin et chef projet
	Consulter	Tous les utilisateurs et clients (en fonction des droits)
Gérer fichiers	Uploader	Admin, chef projet et techniciens
	Visualiser	Tous les utilisateurs et clients (en fonction des droits)
	Télécharger	Tous les utilisateurs et clients (en fonction des droits)
	Supprimer	Admin, chef projet et techniciens
Gérer rapports	Imprimer ou générer	Admin, chef projet et techniciens
	Consulter	Tous les utilisateurs et clients (en fonction des droits)
	Télécharger	Tous les utilisateurs et clients (en fonction des droits)
	Approuver	Admin, chef projet et techniciens
Discussions	Envoyer message	Tous les utilisateurs et clients (en fonction des droits)
	Consulter	
	Répondre	
Notifications	Recevoir	Tous les utilisateurs et clients (en fonction des droits)
	Consulter	

## 2. Diagramme de cas d'utilisation : “d’Authentification”

Voici le diagramme de cas d’utilisation élaboré pour l’authentification :

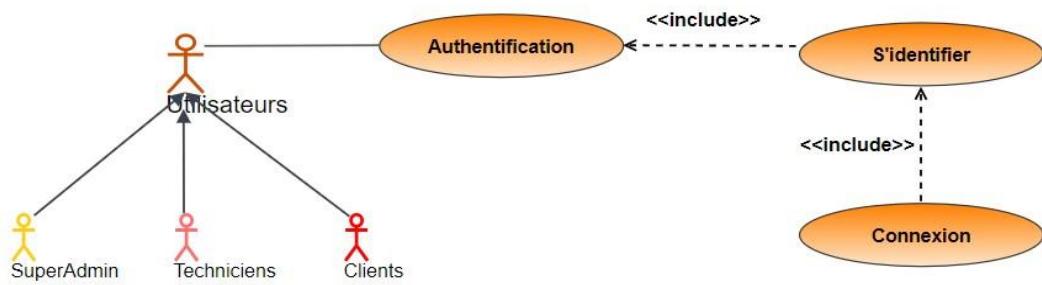


Figure 9: Diagramme de cas d'utilisation \_Authentification à 2 facteurs\_BuildTracker

## Description textuelle

\_\_\_\_\_:

### **Partie 1 : Description.**

#### **Titre : Authentification.**

**Résumé :** Ce cas d'utilisation permet aux utilisateurs de s'authentifier en utilisant un code OTP (One-Time Password) passant par leurs identifiants soit email ou numéro de téléphone.

**Acteurs principaux :** Tous les utilisateurs (Admin, Techniciens, Clients) **Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

### **Partie 2 : Description des scénarios.**

#### **Préconditions :**

L'application est installée et fonctionnel sur un appareil pc ou smartphone.

L'utilisateur est enregistré dans le système avec une adresse e-mail ou numéro de téléphone valide.

#### **Scénario nominale : Authentification à Deux Facteurs**

1. L'utilisateur ouvre l'application.
2. L'application affiche l'écran de connexion demandant à l'utilisateur de choisir entre l'email ou numéro de téléphone pour s'identifier.
3. Le système vérifie les informations d'identification et l'existence de l'utilisateur.
4. Si les informations sont valides, le système génère un code OTP unique et envoie un code OTP par mail ou SMS sur le numéro de téléphone de l'utilisateur.
5. L'utilisateur saisit le code OTP reçu.
6. Le système vérifie la correspondance du code OTP saisi avec celui envoyé.
7. Si le code OTP est correct, l'utilisateur est connecté au système, et redirigé vers la section qui lui convient
8. Le cas d'utilisation se termine. **Scénarios alternatifs :**

#### **Scénario alternatif SA1 : Code OTP incorrect**

Si l'utilisateur saisit un code OTP incorrect, le système affiche un message d'erreur.

L'utilisateur peut réessayer la saisie du code OTP.

Le cas d'utilisation se poursuit au point 7 du scénario nominal.

### Scénario alternatif SA2 : Envoi de code OTP échoué

Si le système ne parvient pas à envoyer le code OTP à l'adresse e-mail de l'utilisateur ou par sms, il affiche un message d'erreur.

L'utilisateur peut demander un nouvel envoi de code OTP ou choisir une autre méthode d'authentification.

Le cas d'utilisation se termine.

### Scénarios d'exceptions :

Scénario d'exception SE1 : Adresse e-mail ou numéro non enregistrée

Si l'adresse e-mail ou le numéro saisi n'est pas associée à un compte utilisateur enregistré, le système affiche un message d'erreur.

Alors L'utilisateur doit saisir une adresse e-mail valide ou numéro.

Le cas d'utilisation se termine.

Post condition : L'utilisateur est connecté au système, et redirigé vers la section qui lui convient.

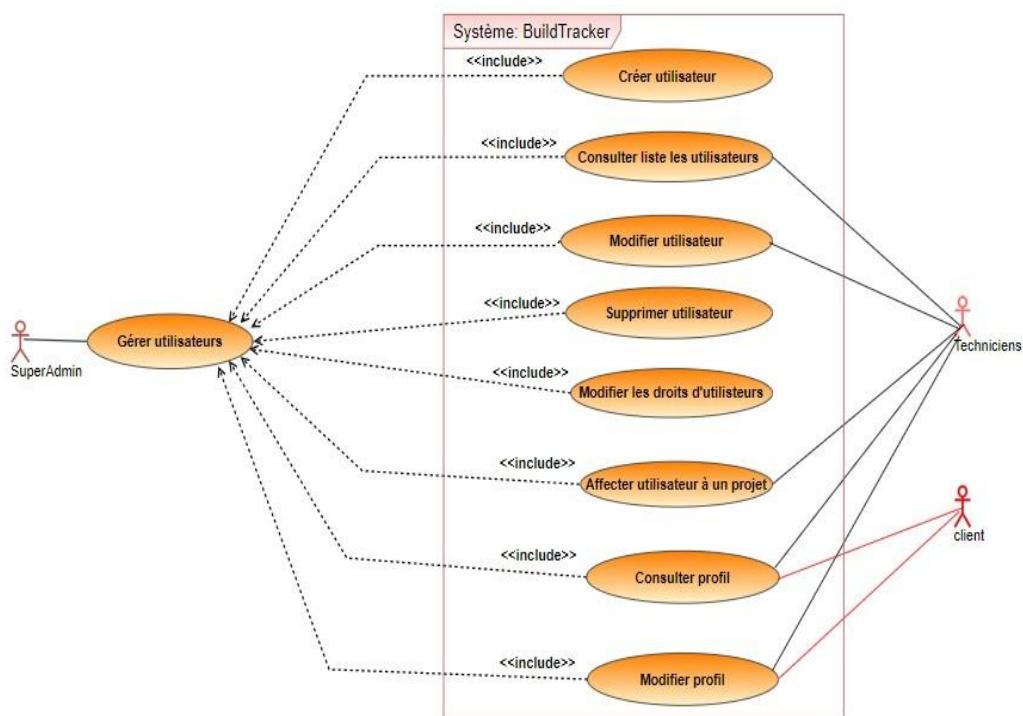
### Partie 3 : Exigences non fonctionnelles :

Les codes OTP doivent être générés de manière aléatoire et ne doivent pas être réutilisables.

Le temps de validité d'un code OTP doit être limité.

## 3. Diagramme de cas d'utilisation : “Gérer Utilisateurs”

Voici le diagramme de cas d'utilisation élaboré pour la gestion des utilisateurs :



## Description textuelle

*Figure 10:Diagramme de cas d'utilisation\_Gérer Utilisateurs*

---

### **Partie 1 : Description.**

#### **Titre : Gérer les utilisateurs.**

**Résumé :** Permet aux administrateurs et aux chefs de projet de gérer les utilisateurs (tous y compris clients) de l'application.

Permet à chaque utilisateur de gérer ses informations personnelles et de consulter ses projets et activités associés.

**Acteurs principaux :** Admin, Techniciens, clients (en fonctions de leurs droits) **Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

### **Partie 2 : Description des scénarios.**

**Préconditions :** Utilisateur authentifié sur le système.

#### **Scénario nominal :**

1. L'administrateur ou le chef de projet accède au module de gestion des utilisateurs.
2. Le système affiche la liste de tous les utilisateurs enregistrés dans l'application.
3. L'administrateur peut rechercher un utilisateur spécifique par nom, adresse e-mail ou rôle.
4. L'administrateur peut afficher les détails d'un utilisateur, y compris ses informations personnelles et son rôle.
5. L'administrateur peut modifier les informations d'un utilisateur, telles que son nom, son adresse e-mail et son rôle.
6. L'utilisateur accède à son profil à partir du menu.
  - a. Le système affiche les informations actuelles de l'utilisateur, y compris son nom, son adresse e-mail et son rôle.
  - b. L'utilisateur peut mettre à jour ses informations personnelles telles que le numéro de téléphone, l'adresse, etc.
  - c. L'utilisateur peut visualiser la liste des projets auxquels il est associé.
  - d. L'utilisateur peut consulter son historique d'activités récentes.
7. L'administrateur peut désactiver ou supprimer un utilisateur si nécessaire

### Scénarios alternatifs

**Scénarios alternatifs SA1 :** Si l'utilisateur recherché n'est pas trouvé, le système affiche un message indiquant que l'utilisateur n'existe pas.

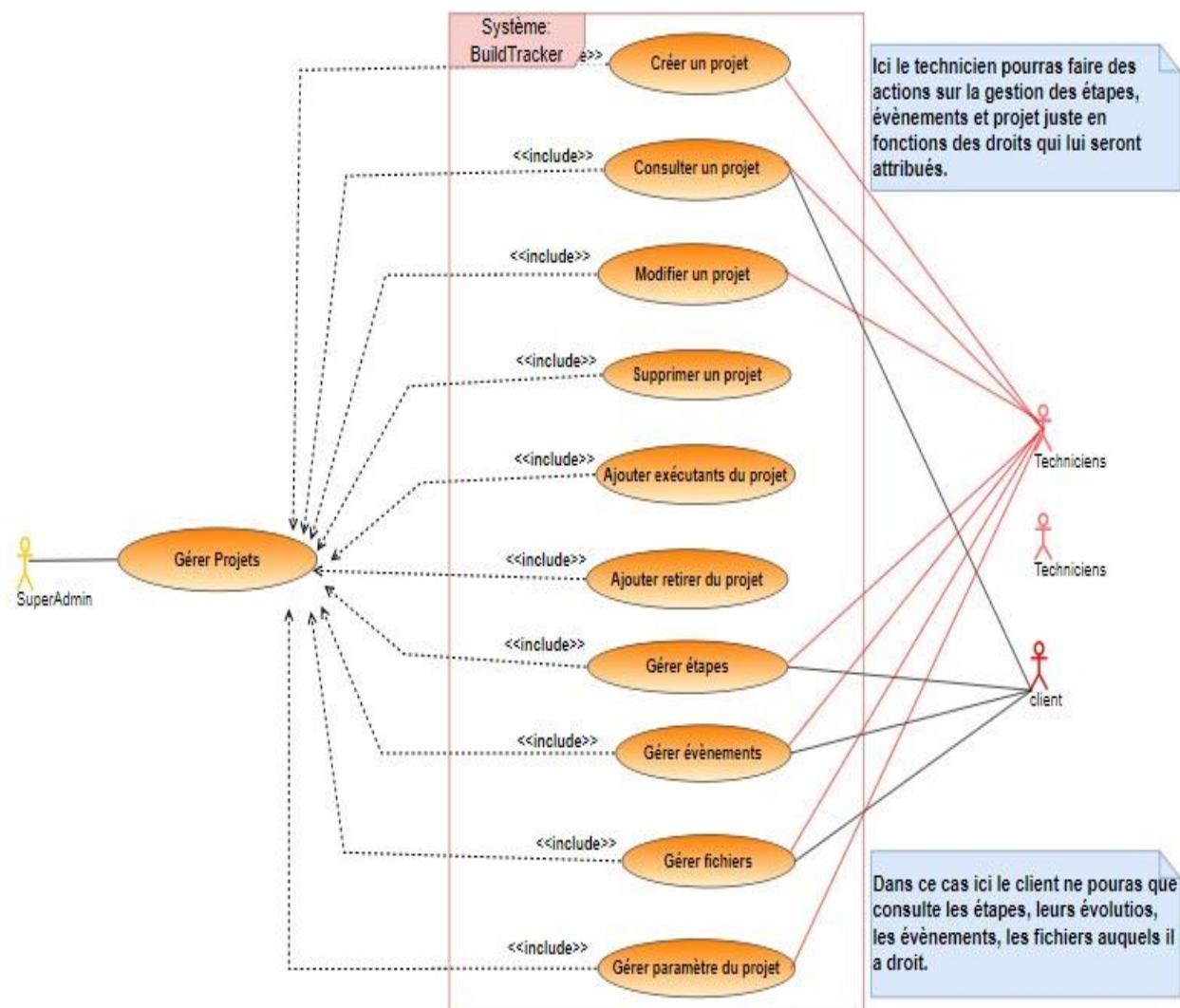
**Scénarios alternatifs SA2 :** Si l'utilisateur ne souhaite pas mettre à jour certaines informations, il peut les laisser inchangées.

**Post-conditions :** Les informations des utilisateurs sont gérées, modifiées ou supprimées selon les besoins.

Les utilisateurs peuvent gérer et consulter leurs informations personnelles, leurs projets et leurs activités depuis leur profil.

## 4. Diagramme de cas d'utilisation : "Gérer projets"

Voici le diagramme de cas d'utilisation élaboré pour la gestion des projets :



## Description textuelle

Figure 11: Diagramme de cas d'utilisation \_Gérer projets

---

### **Partie 1 : Description.**

#### **Titre : Gérer projet.**

**Résumé :** Permet aux utilisateurs de créer un nouveau projet en fournissant les détails tels que le nom, la description et les dates clés. Ensuite de gérer les informations et les détails généraux d'un projet.

**Acteurs principaux :** Tous les utilisateurs (Admin, Techniciens, Clients) en fonction de leur droit.

**Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

### **Partie 2 : Description des scénarios.**

**Préconditions :** Utilisateur authentifié sur le système.

#### **Scénario nominal :**

1. L'utilisateur accède au module de gestion de projet.
2. L'utilisateur choisit un projet existant ou en crée un nouveau.
  - 2.1 L'utilisateur choisit l'option "Créer un Projet".
  - 2.2 Le système affiche un formulaire de création de projet.
  - 2.3 L'utilisateur remplit les détails du projet tels que le nom, la description et les dates.
  - 2.4 L'utilisateur confirme la création du projet.
  - 2.5 Le système enregistre les informations du projet.
  - 2.6 Le système affiche un message de succès.
1. Le système affiche les informations générales du projet, telles que le nom, la description et les dates clés.
2. L'utilisateur peut modifier les détails du projet, y compris le nom, la description et les dates clés.
3. L'utilisateur peut ajouter ou supprimer des membres de collaborateurs associés au projet.
4. L'utilisateur peut suivre l'état d'avancement global du projet.

#### **Scénarios alternatifs :**

### Scénarios alternatifs SA1 :

Si l'utilisateur ne remplit pas correctement les détails du projet, le système affiche un message d'erreur et demande de corriger les champs.

**Scénarios alternatifs SA2 :** Si aucun projet n'est associé à l'utilisateur, le système affiche un message indiquant qu'aucun projet n'est disponible.

**Post-conditions :** Un nouveau projet est créé dans le système. Les détails du projet sont gérés, modifiés et suivis.

## 5. Diagramme de cas d'utilisation : "Gérer Etapes"

Voici le diagramme de cas d'utilisation élaboré pour la gestion des étapes :

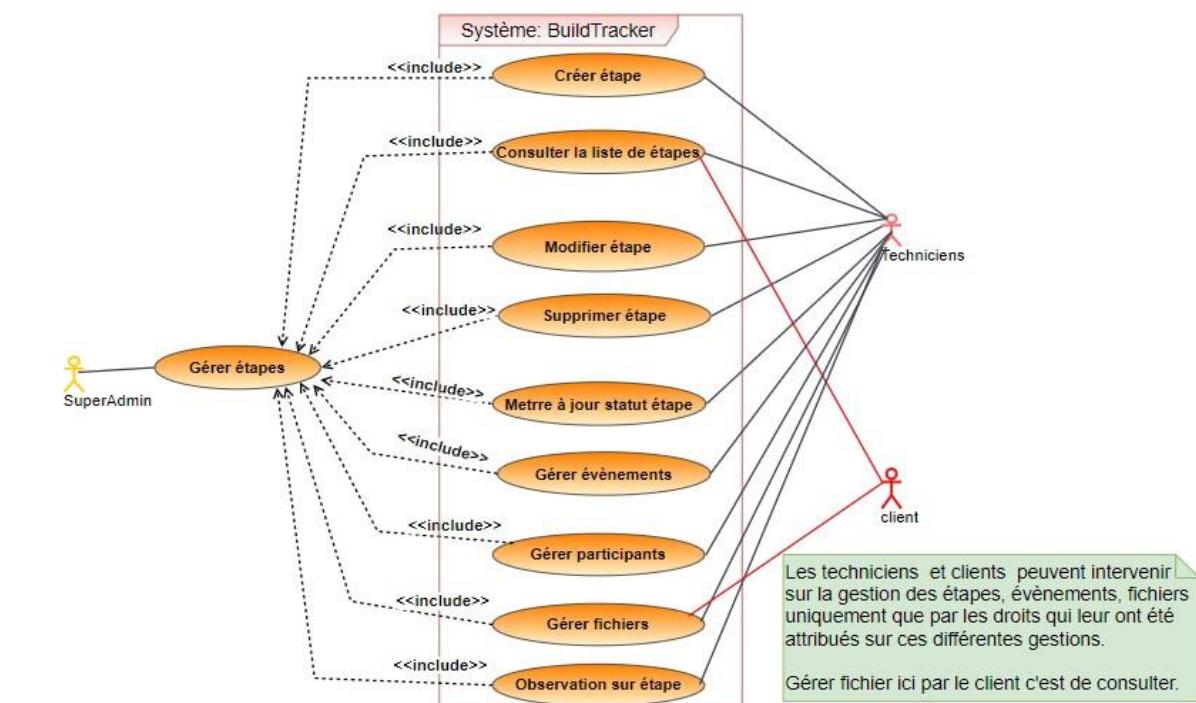


Figure 12: Diagramme de cas d'utilisation \_Gérer Etapes

### Description textuelle

#### Partie 1 : Description.

**Titre :** Gérer étapes.

**Résumé :** Ce cas d'utilisation permet aux utilisateurs autorisés de créer, modifier et suivre les étapes de progression d'un projet.

**Acteurs principaux :** Utilisateur (Chef de Projet, Technicien, Client) en fonction de leur droit.

**Acteurs secondaires :** Le système (BuildTracker) .

**Date :** 11/07/2023

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Description textuelle**

**Version :** 1.0

## **Partie 2 : Description des scénarios.**

**Préconditions :** Utilisateur authentifié sur le système, un projet existe dans le système.

### **Scénario nominal :**

1. L'utilisateur accède aux étapes d'un projet spécifique.
2. Le système affiche la liste des étapes actuelles du projet.
3. L'utilisateur peut créer de nouvelles étapes en fournissant les détails nécessaires.
4. L'utilisateur peut modifier les détails des étapes existantes.
5. L'utilisateur peut marquer les étapes comme achevées au fur et à mesure de la progression.

### **Scénarios alternatifs**

**Scénarios alternatifs SA1 :** Si l'utilisateur ne remplit pas correctement les détails de l'étape, le système affiche un message d'erreur.

**Scénarios alternatifs SA2 :** Si aucune étape n'est associée à l'utilisateur, le système affiche un message indiquant qu'aucun projet n'est disponible.

**Post-conditions :** Les étapes du projet sont créées, gérées et marquées comme achevées. Les utilisateurs peuvent consulter les détails des projets et accéder aux fonctionnalités associées.

## **6. Diagramme de cas d'utilisation : “Gérer Evènements”**

Voici le diagramme de cas d'utilisation élaboré pour la gestion des évènements :

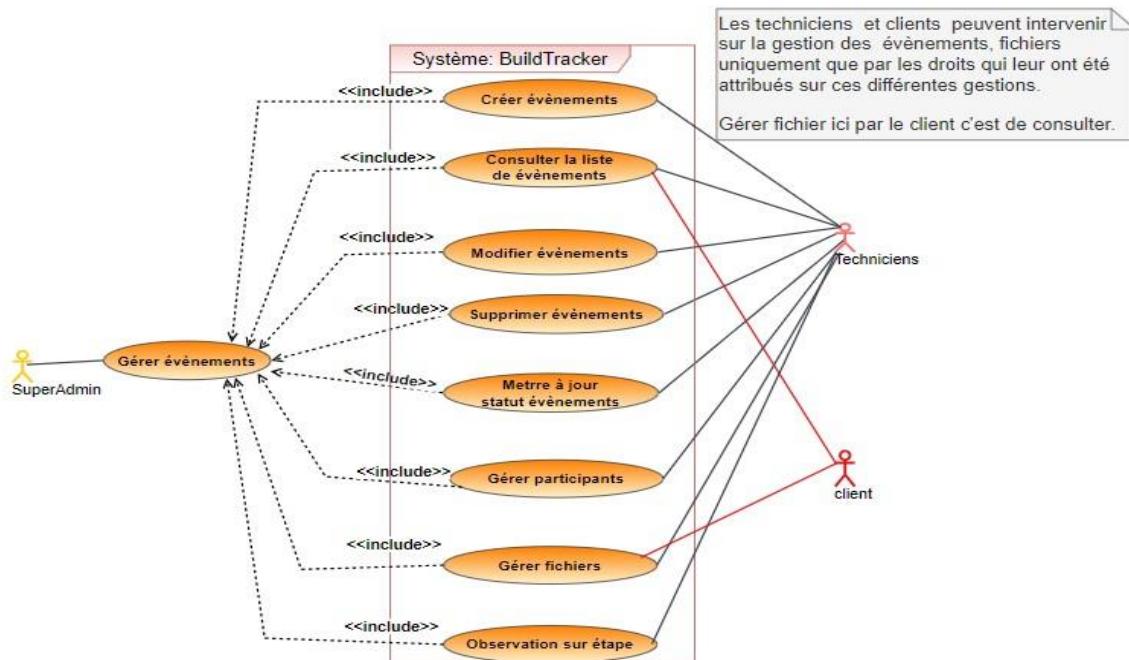


Figure 13: Diagramme de cas d'utilisation \_Gérer Evènements

### **Description textuelle**

## **Partie 1 : Description.**

**Titre : Gérer évènements.**

**Résumé :** Ce cas d'utilisation permet aux utilisateurs autorisés de planifier et de gérer des événements spécifiques liés à un projet.

**Acteurs principaux :** Admin et les Techniciens (chef projet) en fonction de leur droit.

**Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

**Partie 2 : Description des scénarios.**

**Préconditions :** Utilisateur authentifié sur le système, un projet existe dans le système.

**Scénario nominal :**

1. L'utilisateur accède au module de gestion des projets.
2. L'utilisateur choisit un projet existant.
3. L'utilisateur choisit l'option "Gérer les Évènements".
4. Le système affiche la liste des événements planifiés pour le projet.
5. L'utilisateur peut ajouter un nouvel évènement en fournissant les détails.
6. L'utilisateur peut modifier les détails d'un évènement existant.
7. L'utilisateur peut supprimer un évènement.
8. L'utilisateur peut marquer les événements comme planifiés, en cours ou terminés.

**Scénarios alternatifs**

**Scénarios alternatifs SA1 :** Si l'utilisateur ne remplit pas correctement les détails de l'évènement, le système affiche un message d'erreur et lui demande de corriger.

**Post-conditions :** Les évènements du projet sont planifiés, modifiés et gérés et marqués selon leur statut.

**7. Diagramme de cas d'utilisation : "Gérer Fichiers"**

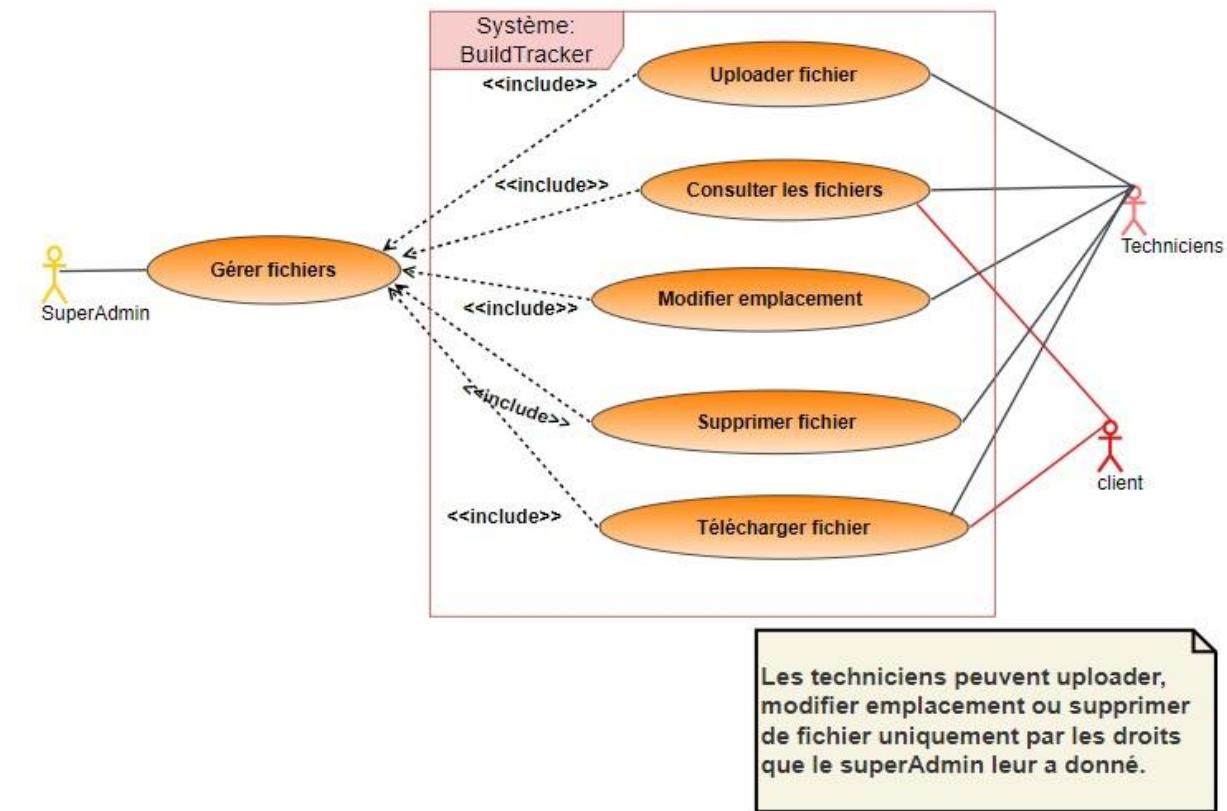


Figure 14: Diagramme de cas d'utilisation \_Gérer Fichiers

## Description textuelle Partie

### 1 : Description.

**Titre : Gérer les fichiers.**

**Résumé :** Permet aux utilisateurs de télécharger, consulter et gérer les fichiers liés à un projet.

**Acteurs principaux :** Admin et les Techniciens (chef projet) en fonction de leur droit.

**Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

### Partie 2 : Description des scénarios.

**Préconditions :** Utilisateur authentifié sur le système, un projet existe dans le système.

### Scénario nominal :

1. L'utilisateur accède au module de gestion des projets.
2. L'utilisateur choisit un projet existant.
3. L'utilisateur choisit l'option "Gérer les Fichiers".
4. Le système affiche la liste des fichiers liés au projet.
5. L'utilisateur peut télécharger de nouveaux fichiers.
6. L'utilisateur peut consulter les fichiers existants.
7. L'utilisateur peut supprimer des fichiers.

### **Scénarios alternatifs**

**Scénarios alternatifs SA1 :** Si l'utilisateur rencontre des problèmes lors du téléchargement ou de la suppression de fichiers, le système affiche un message d'erreur.

**Scénarios alternatifs SA2 :** Si aucun rapport n'est disponible pour le projet, le système affiche un message indiquant leur absence.

**Post-conditions** : Les fichiers du projet sont téléchargés, consultés et gérés.

## **8. Diagramme de cas d'utilisation : "Gérer Rapports"**

Voici le diagramme de cas d'utilisation élaboré pour la gestion des rapports :

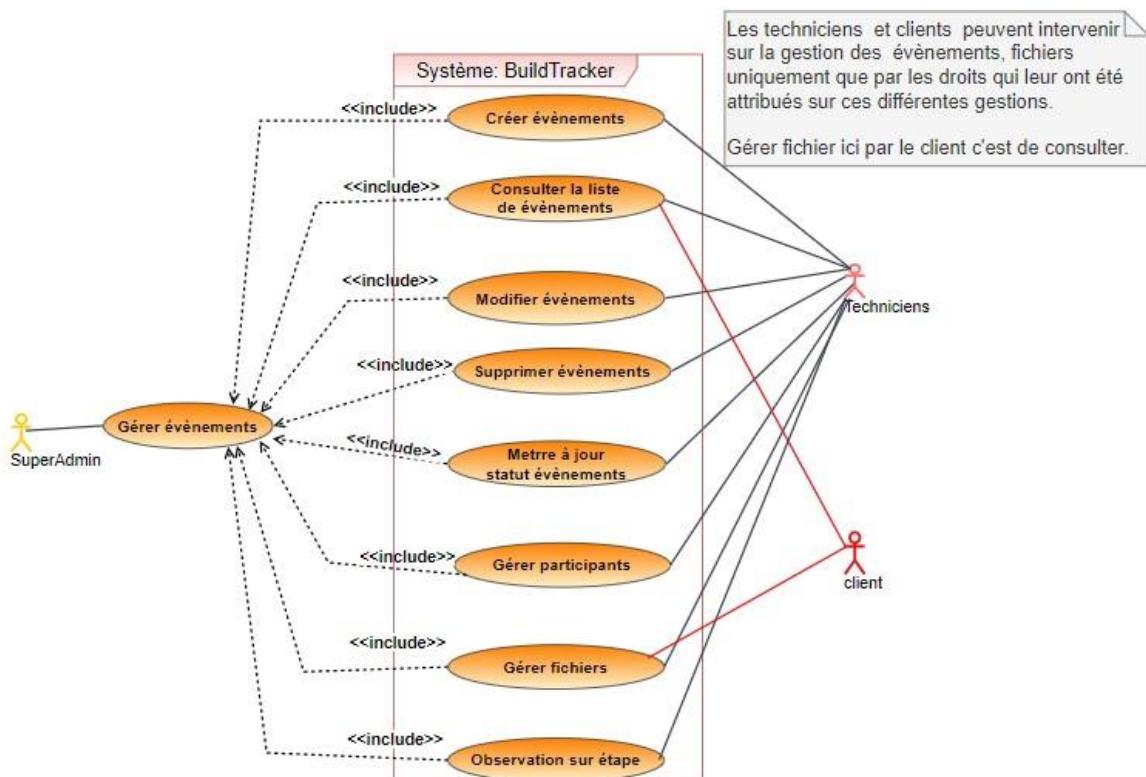


Figure 15: Diagramme de cas d'utilisation \_Gérer Rapports

### **Description textuelle**

## **Partie 1 : Description.**

### **Titre : Gérer Rapports**

**Résumé :** Permet aux utilisateurs de planifier et de gérer les évènements associés à un projet.

**Acteurs principaux :** Admin et les Techniciens (chef projet) en fonction de leur droit.

**Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

## **Partie 2 : Description des scénarios.**

**Préconditions :** Utilisateur authentifié sur le système, un projet existe dans le système.

### **Scénario nominal :**

1. L'utilisateur accède au module de gestion des projets.
2. L'utilisateur choisit un projet existant.
3. L'utilisateur choisit l'option "Gérer les Rapports".
4. Le système affiche la liste des types de rapports disponibles.
5. L'utilisateur choisit un type de rapport.
6. Le système génère le rapport en fonction des données du projet et des critères.
7. L'utilisateur peut consulter le rapport généré.

### **Scénarios alternatifs**

**Scénarios alternatifs SA1 :** Si le système ne parvient pas à générer un rapport en raison de données manquantes, il affiche un message d'avertissement.

### **Post-conditions :**

Les rapports sur l'avancement et les statistiques du projet sont générés et consultés.

## **9. Diagramme de cas d'utilisation : “Module de Discussion”**

Voici le diagramme de cas d'utilisation élaboré pour le module de discussions :

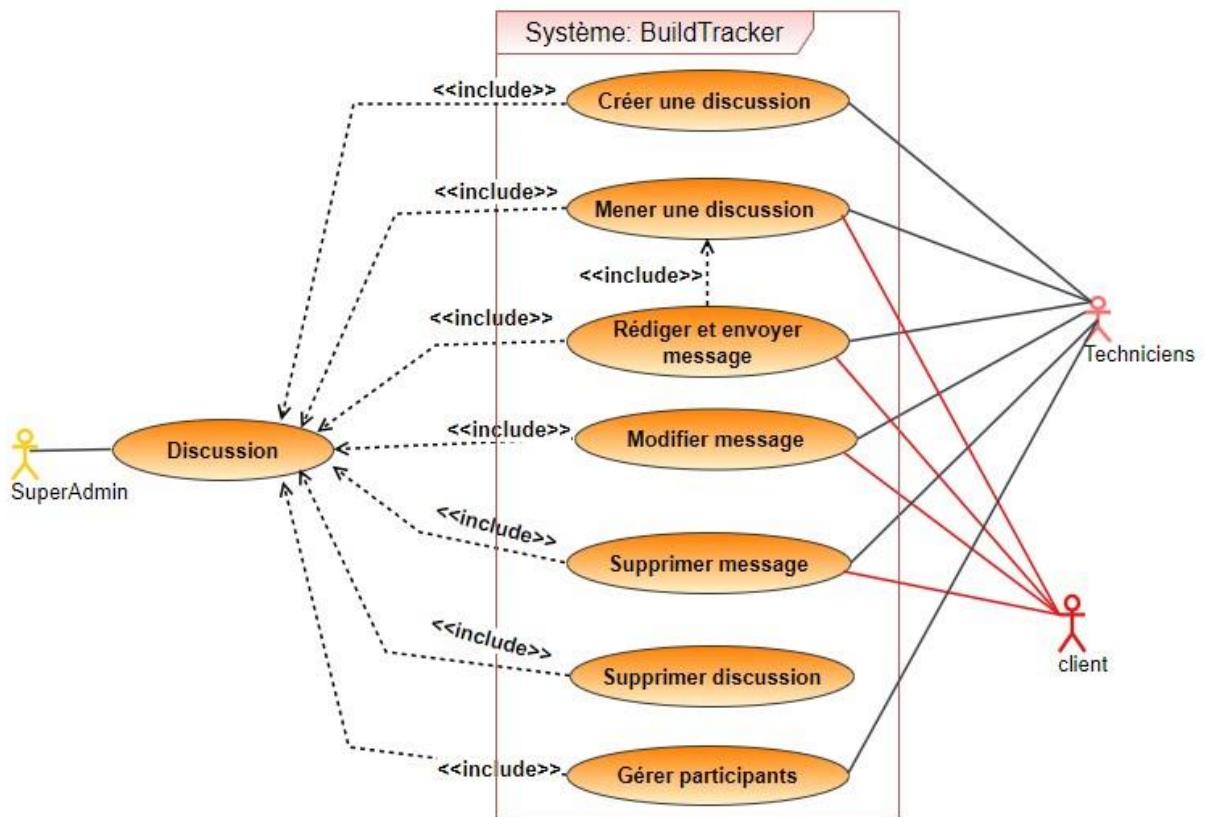


Figure 16: Diagramme de cas d'utilisation \_Module de Discussion

### Description textuelle

#### Partie 1 : Description.

##### **Titre : Gérer discussions.**

**Résumé :** Permet à tous les collaborateurs de communiquer en temps réel via des discussions individuelles et de groupe.

**Acteurs principaux :** tous les utilisateurs (Admin, Chef de Projet, Technicien, Client) en fonction de leur droit spécifiques.

**Acteurs secondaires :** Le système (BuidTracker) .

**Date :** 11/07/2023

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

#### Partie 2 : Description des scénarios.

**Préconditions :** Utilisateur authentifié sur le système, les utilisateurs ou collaborateurs enregistrés, un projet existe dans le système.

### **Scénario nominal :**

1. L'utilisateur accède au module de messagerie "Discussions".
2. Le système affiche la liste des discussions récentes et des contacts.
3. L'utilisateur peut démarrer une nouvelle discussion individuelle en sélectionnant un contact.
4. L'utilisateur peut également créer une discussion de groupe en sélectionnant plusieurs contacts.
5. L'utilisateur peut envoyer des messages texte, des images et des fichiers aux participants de la discussion.
6. Les participants reçoivent les messages en temps réel et peuvent répondre instantanément.
7. L'utilisateur peut voir si les autres participants ont lu ses messages.
8. L'utilisateur peut supprimer ou archiver des discussions.

### **Scénarios alternatifs**

**Scénarios alternatifs SA1 :** Si un participant ne peut pas répondre en temps réel, l'utilisateur peut voir une indication que le message n'a pas encore été lu.

**Post-conditions :** Les membres de l'équipe peuvent communiquer rapidement et efficacement via des discussions individuelles et de groupe.

**10. Diagramme de cas d'utilisation : "Module de Notifications"** Voici le diagramme de cas d'utilisation élaboré pour le module de notifications :

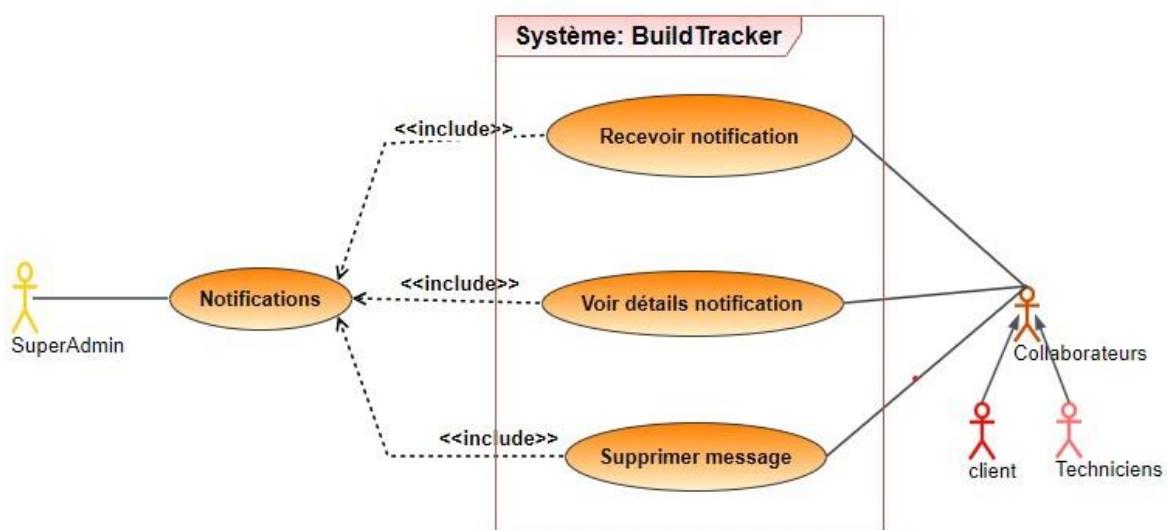


Figure 17: Diagramme de cas d'utilisation \_Module de Notifications

### **Description textuelle**

## **Partie 1 : Description.**

**Titre : Notifications.**

**Résumé :** Informe les utilisateurs des nouveaux messages et des mises à jour importantes.

**Acteurs principaux :** tous les utilisateurs (Admin, Chef de Projet, Technicien, Client).

**Acteurs secondaires :** Le système (BuidTracker) .

**Date : 11/07/2023**

**Responsable :** ELOHOUNKPON Dewanou Roméo

**Version :** 1.0

## **Partie 2 : Description des scénarios.**

**Préconditions :** Utilisateur authentifié sur le système, les utilisateurs ou collaborateurs enregistrés, un projet existe dans le système.

### **Scénario nominal :**

1. Le système surveille les nouvelles activités telles que les nouveaux messages, les invitations aux discussions, les mises à jour de projets, etc.
2. Lorsqu'une nouvelle activité se produit, le système envoie une notification en temps réel à l'utilisateur concerné.
3. L'utilisateur reçoit la notification sur l'application mobile, le site web ou par e-mail.
4. L'utilisateur peut cliquer sur la notification pour accéder directement à la discussion ou à l'activité concernée

### **Scénarios alternatifs**

**Scénarios alternatifs SA1 :** l'utilisateur ne répond pas à la notification, le système peut envoyer des rappels ultérieurs.

### **Post-conditions :**

Les utilisateurs sont informés rapidement des nouvelles activités et des discussions en cours. Tous ces diagrammes de classes définis nous permettent une vision plus claire et précise sur les attentes de notre système.

## **II. Étude technique**

Dans le cadre de notre étude technique, nous avons approfondi la manière dont les utilisateurs interagiront avec notre système, en répondant à la question essentielle de "comment" et "quand" ils pourront accomplir leurs différentes actions. Par les diagrammes nous avons :

### a. Diagramme de séquences

#### 1. Diagramme de séquences : Authentification avec code OTP

Voici le diagramme de séquence élaboré pour l'authentification

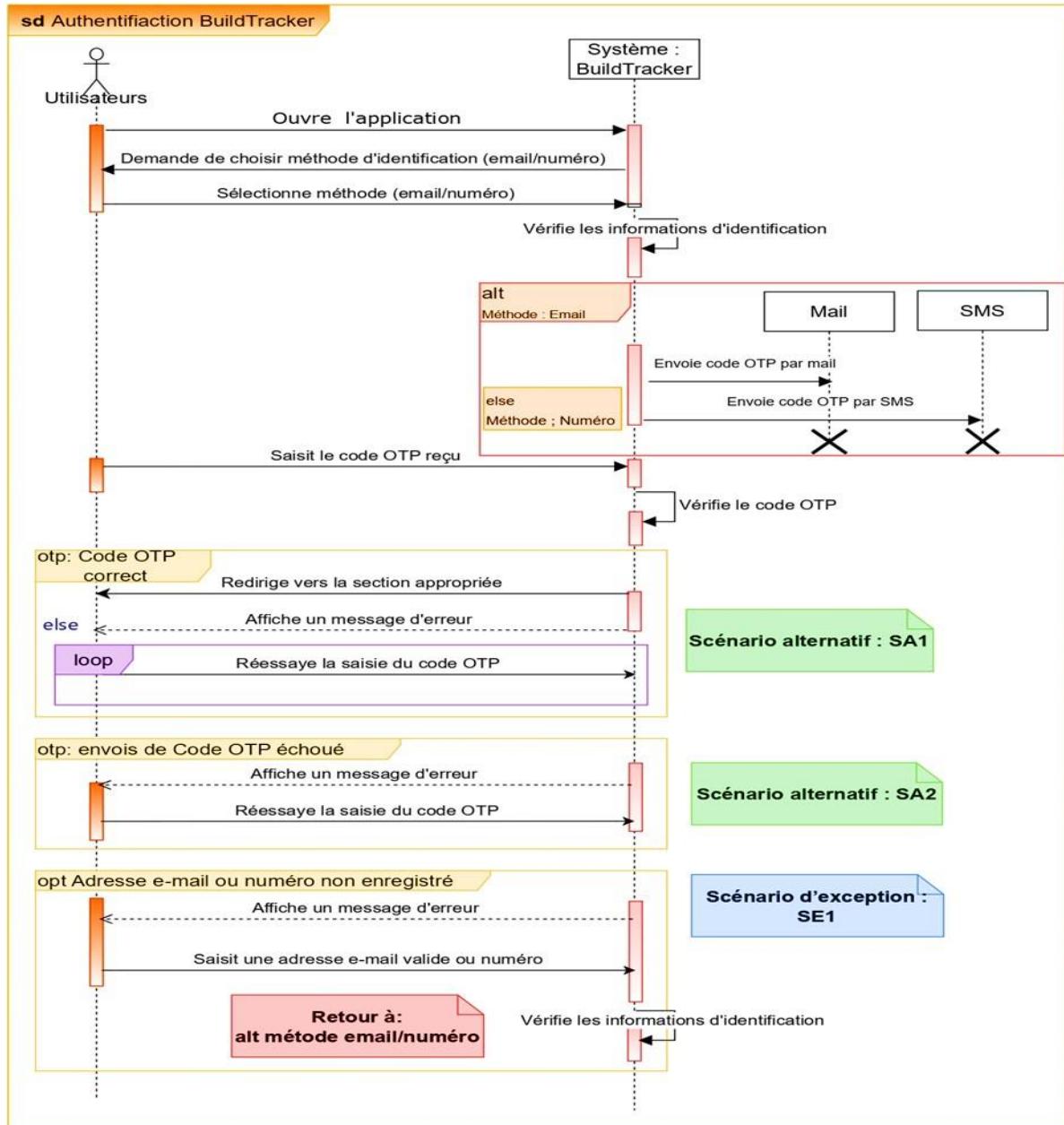
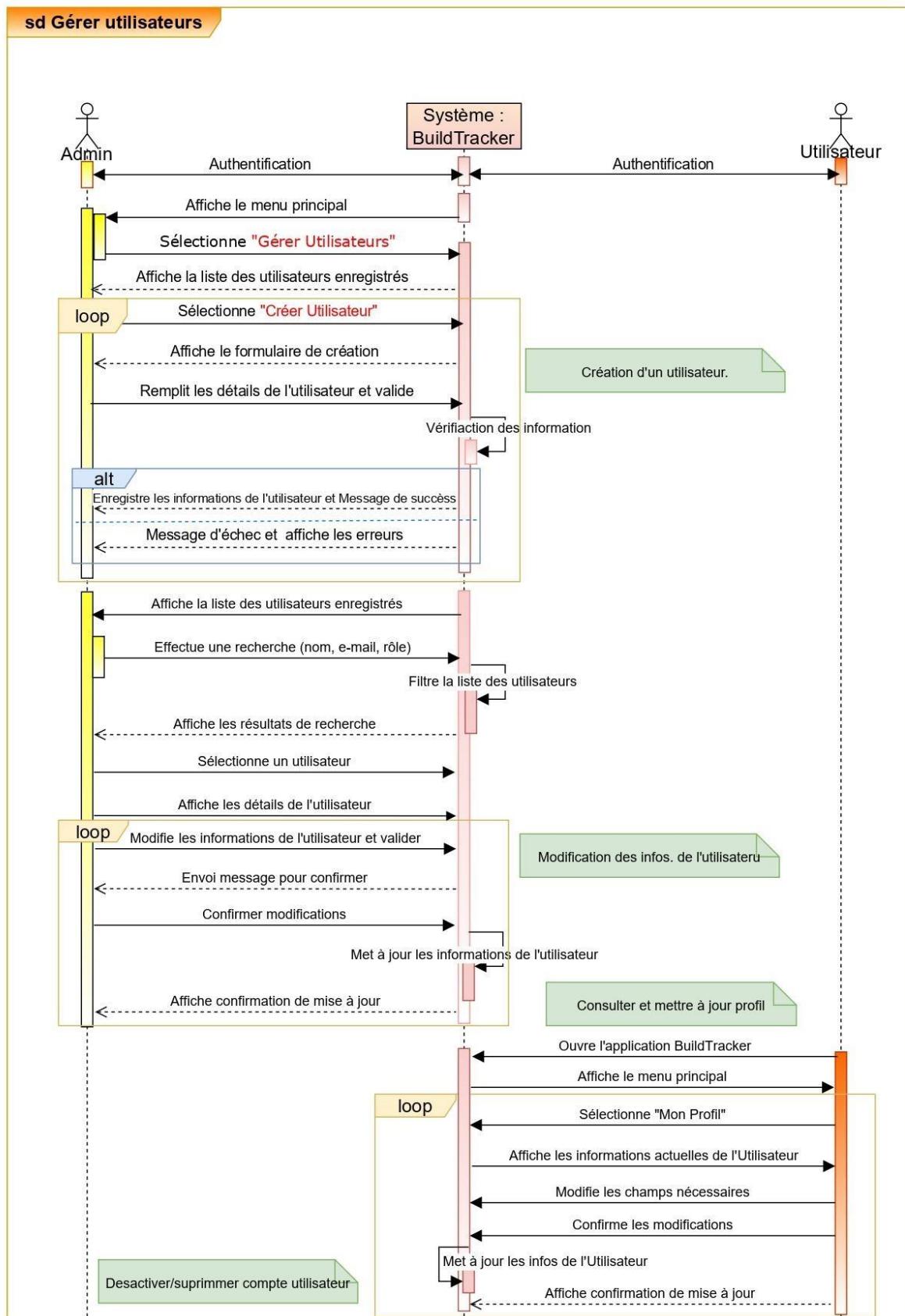


Figure 18: Diagramme de séquences\_Authentification avec code OTP

#### 2. Diagramme de séquences : "Gérer Utilisateurs"

Voici le diagramme de séquence élaboré pour le module de gestion utilisateurs



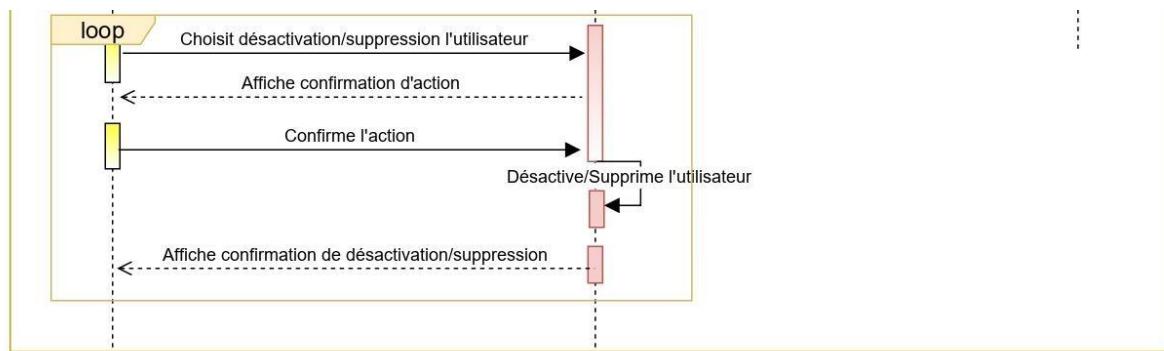


Figure 19: Diagramme de séquences\_Gérer Utilisateurs

### 3. Diagramme de séquences : “Module de Notification”

Voici le diagramme de séquence élaboré pour le module de notification :

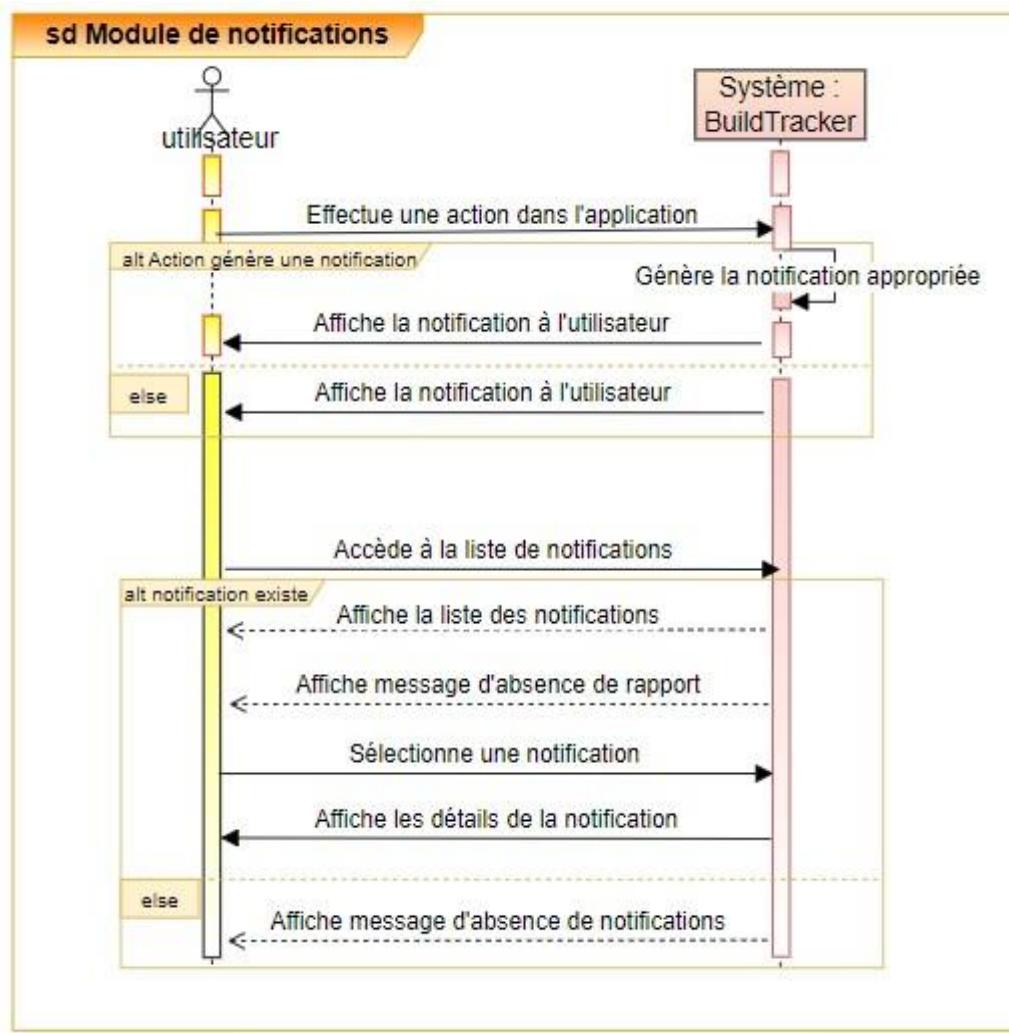
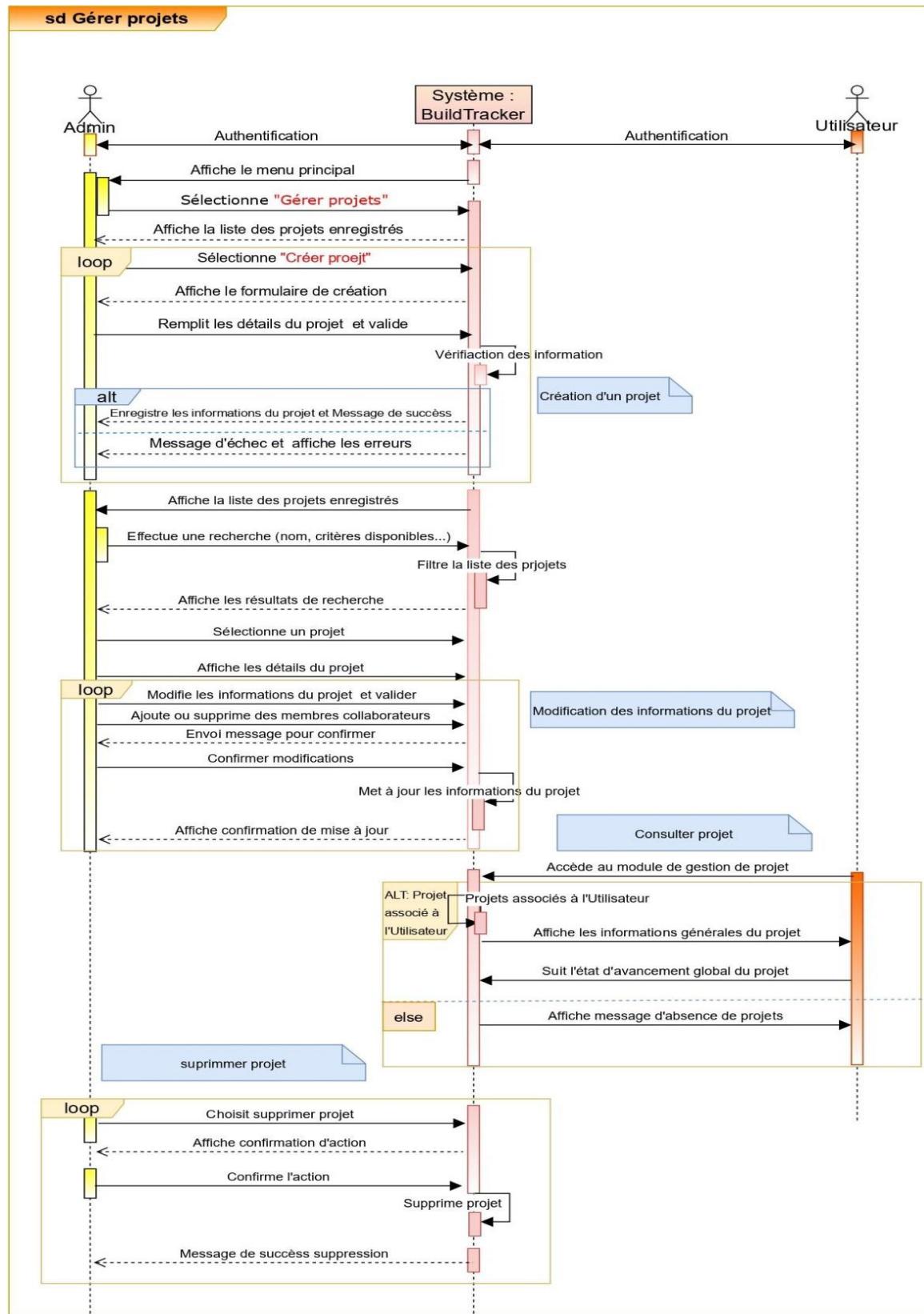


Figure 20: Diagramme de séquences\_Module de Notification

## : "Gérer

### 4. Diagramme de séquences Projets"

Voici le diagramme de séquence élaboré pour le module de gestion des projets :



### : “Gérer

*Figure 21: Diagramme de séquences Gérer Projets*

### **5. Diagramme de séquences Etapes”**

Voici le diagramme de séquence élaboré pour le module de gestion des étapes :

## : "Gérer

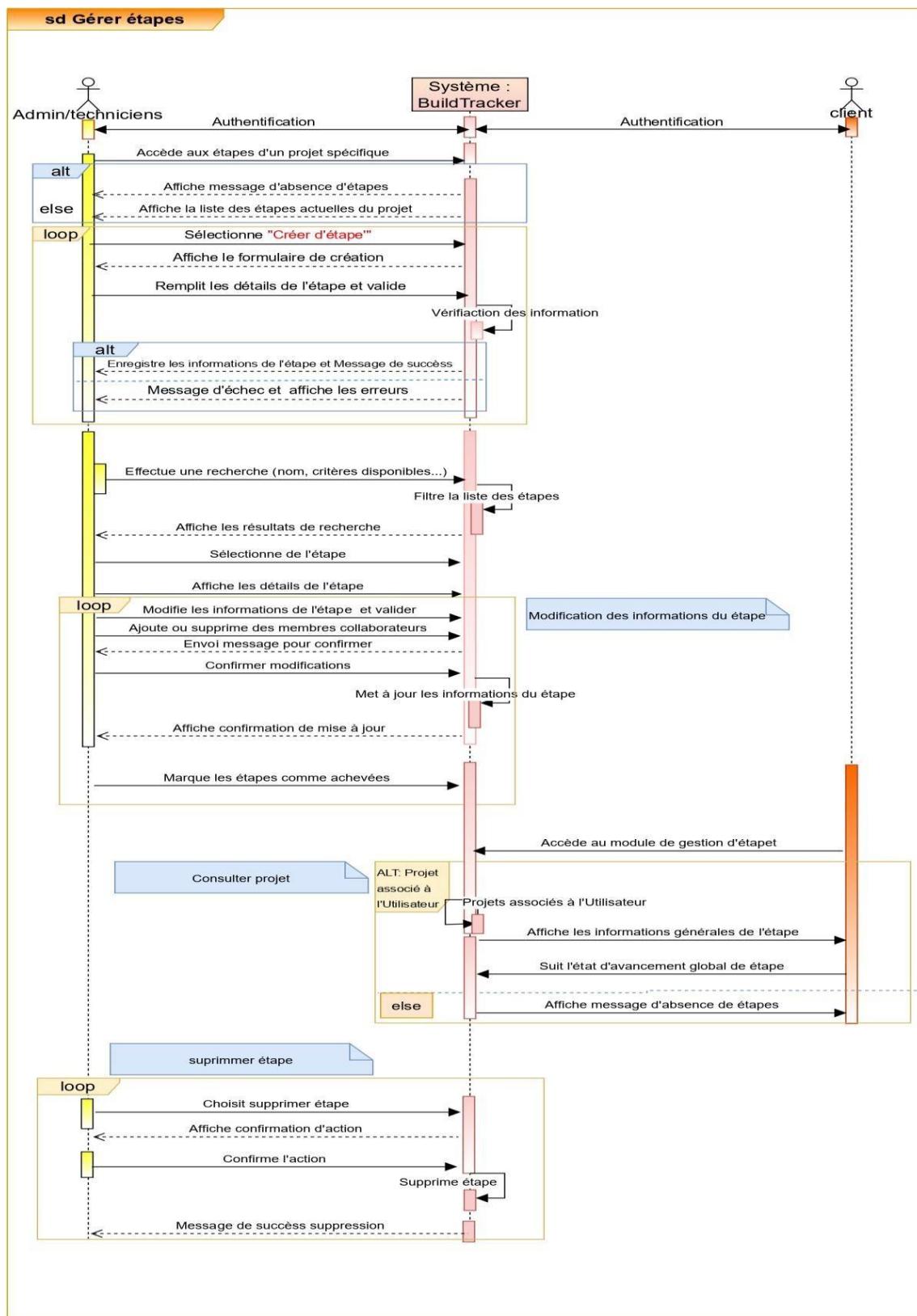


Figure 22: Diagramme de séquences \_Gérer Etapes

## 6. Diagramme de séquences "Evènements"

Voici le diagramme de séquence élaboré pour le module de gestion des évènements :

## : "Gérer

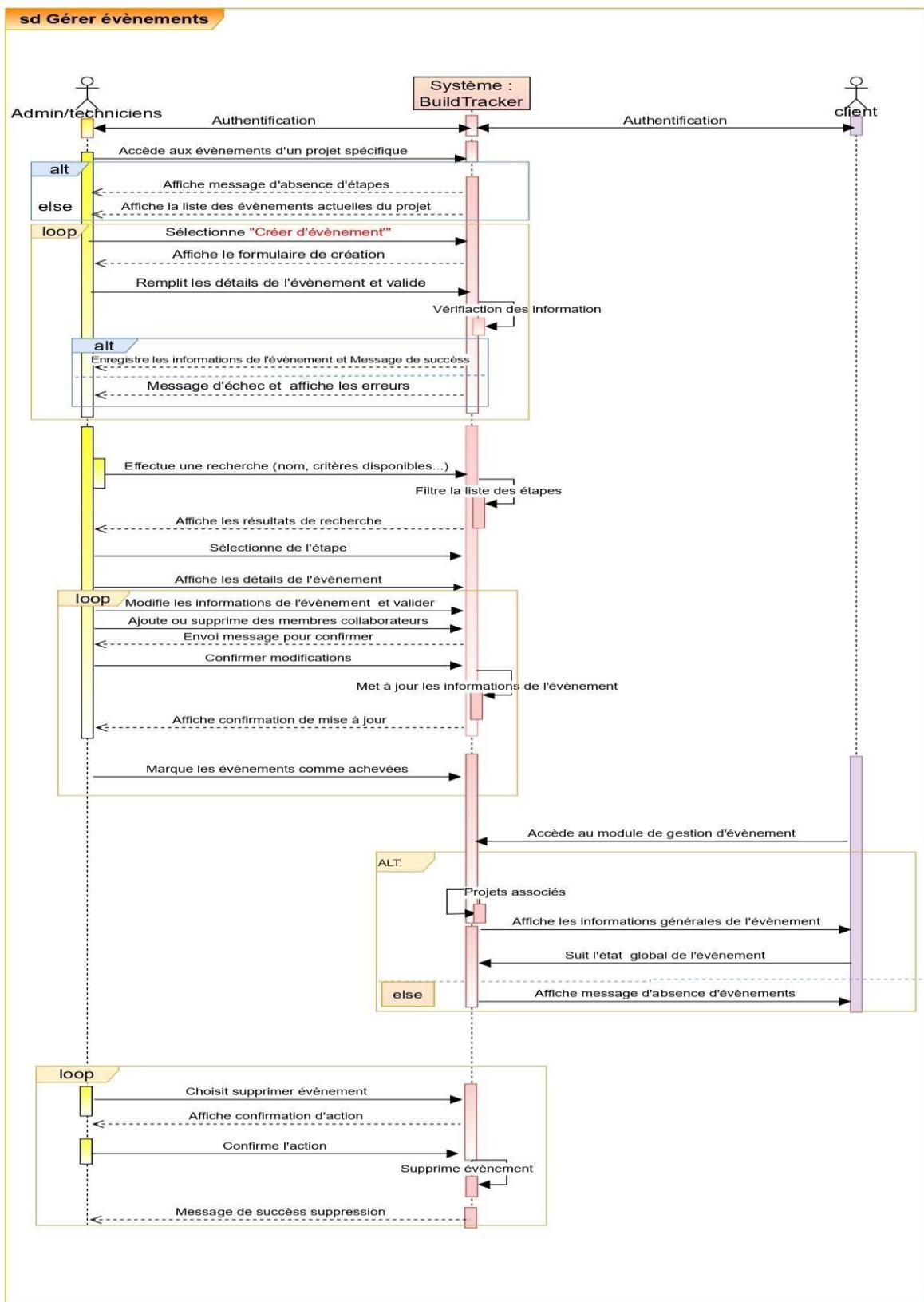


Figure 23: Diagramme de séquences\_Gérer Evènements

## 7. Diagramme de séquences Rapports"

## : "Gérer

Voici le diagramme de séquence élaboré pour le module de gestion des rapports :

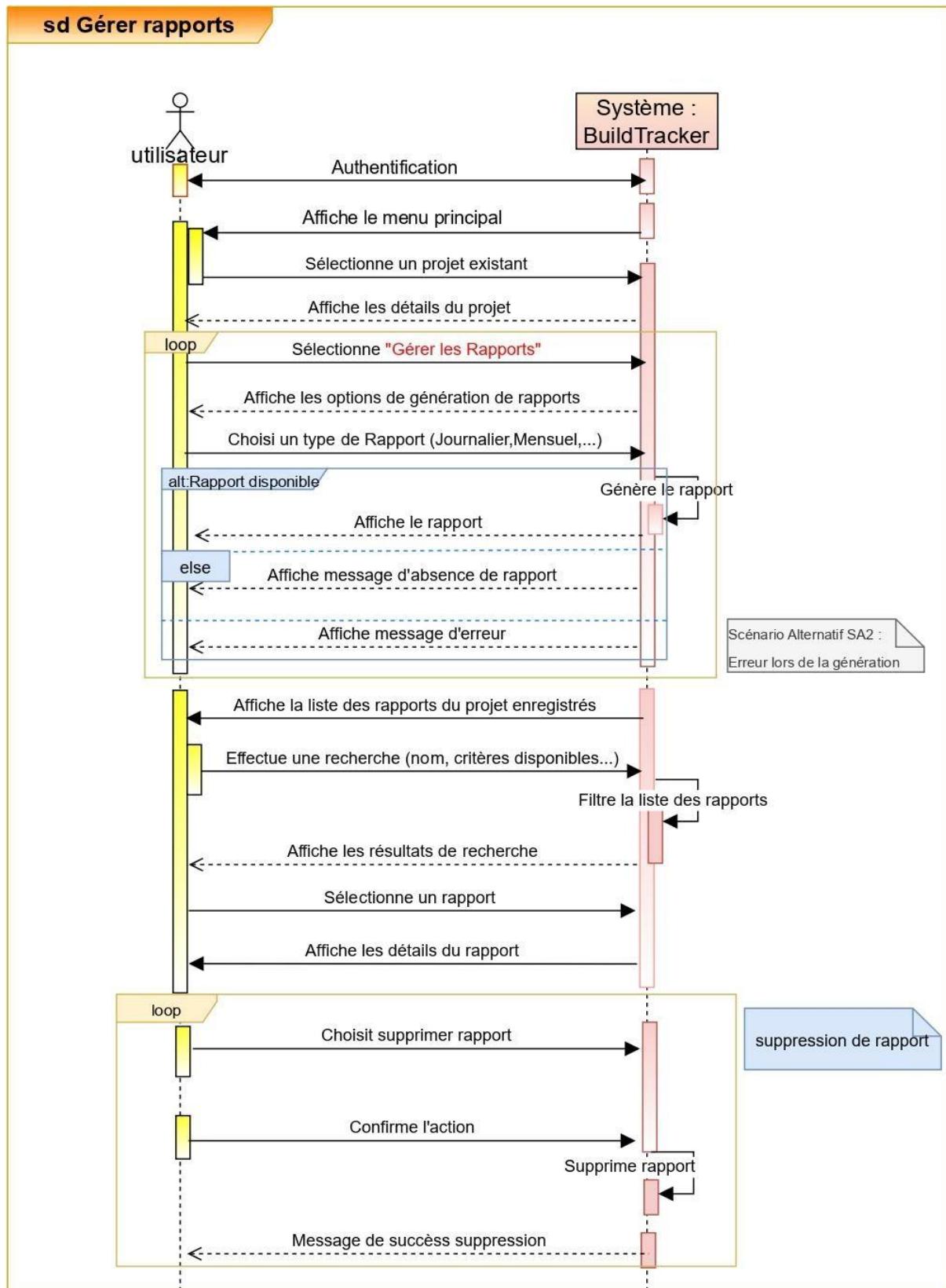


Figure 24: Diagramme de séquences\_Gérer Rapports

## 8. Diagramme de séquences : "Module de Discussions"

Voici le diagramme de séquence élaboré pour le module de discussion :

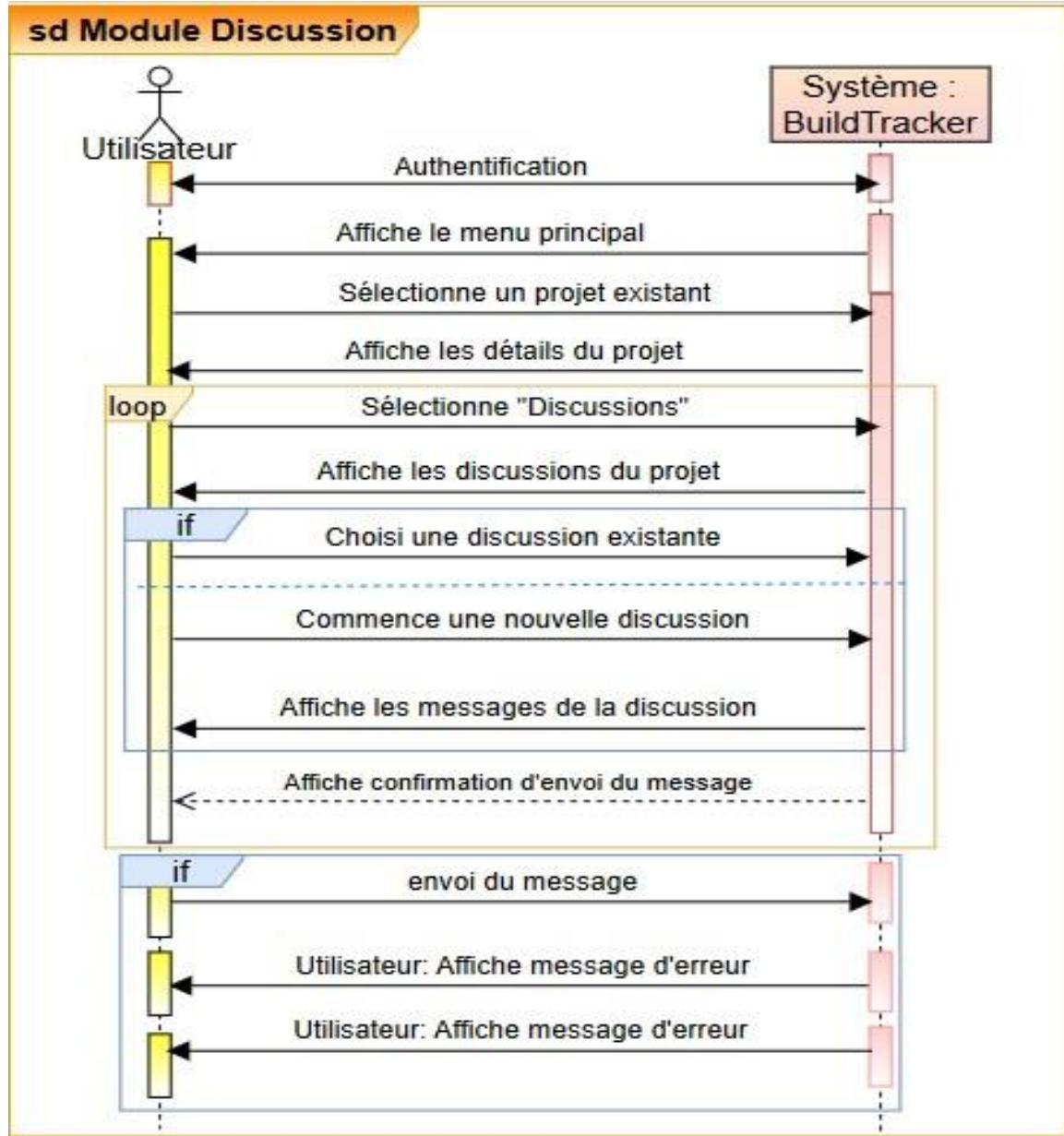


Figure 25: Diagramme de séquences\_Module de Discussions

Ces diagrammes mettent en lumière la chronologie des actions et des communications entre les acteurs (utilisateurs) et le système. Ils nous permettent de comprendre en détail le déroulement des processus, notamment l'authentification des utilisateurs, la gestion de projets, la création d'événements, la communication au sein des discussions, et la réception de notifications. Les diagrammes de séquences sont essentiels pour visualiser le flux des opérations au sein de notre application.

## b. Diagramme d'activités

Les diagrammes d'activités sont utilisés pour représenter les flux de travail, les décisions, et les étapes logiques au sein du système. Ils offrent une vue plus globale des processus, montrant comment les utilisateurs naviguent à travers les différentes fonctionnalités de notre application. Ces diagrammes nous aident à comprendre "comment" les utilisateurs peuvent accomplir leurs tâches, que ce soit pour la gestion de projets, la création d'événements, la gestion de fichiers, ou d'autres activités clés.

### 1. Diagramme d'activités : ‘Authentification’

Voici le diagramme d'activités élaboré pour l'authentification :

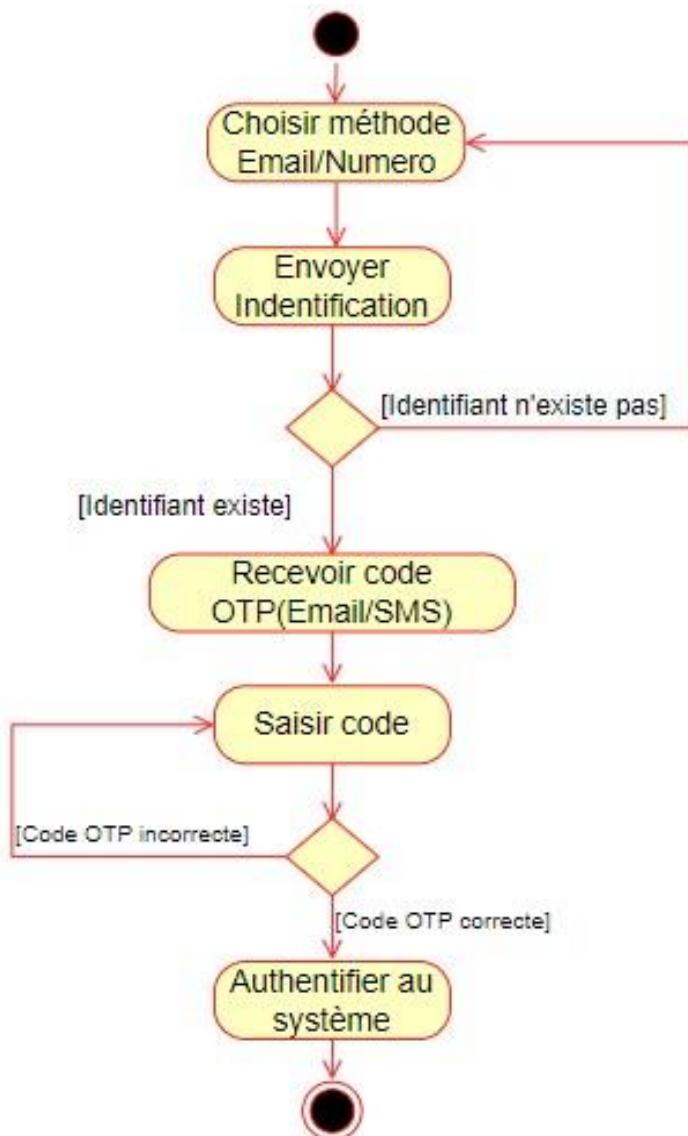


Figure 26: Diagramme d'activités\_Authentification

## Diagramme d'activités 2. : “Gérer Projet”

Voici le diagramme d'activités élaboré pour le module de gestion projet :

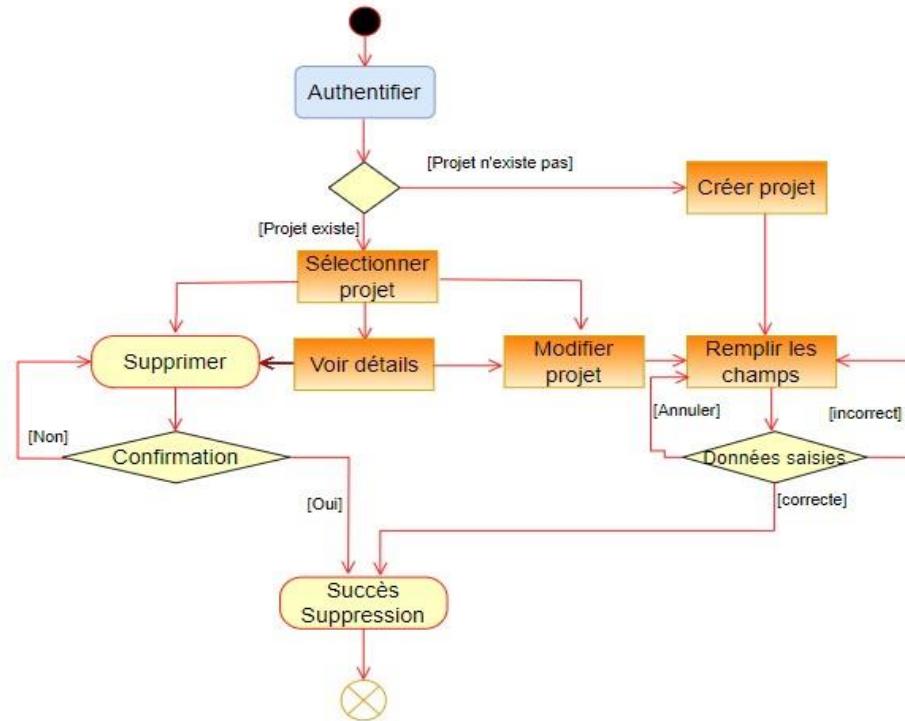


Figure 27: Diagramme d'activités\_Gérer Projet

## 3. Diagramme d'activités : “Gérer Etapes”

Voici le diagramme d'activités élaboré pour le module de gestion étapes :

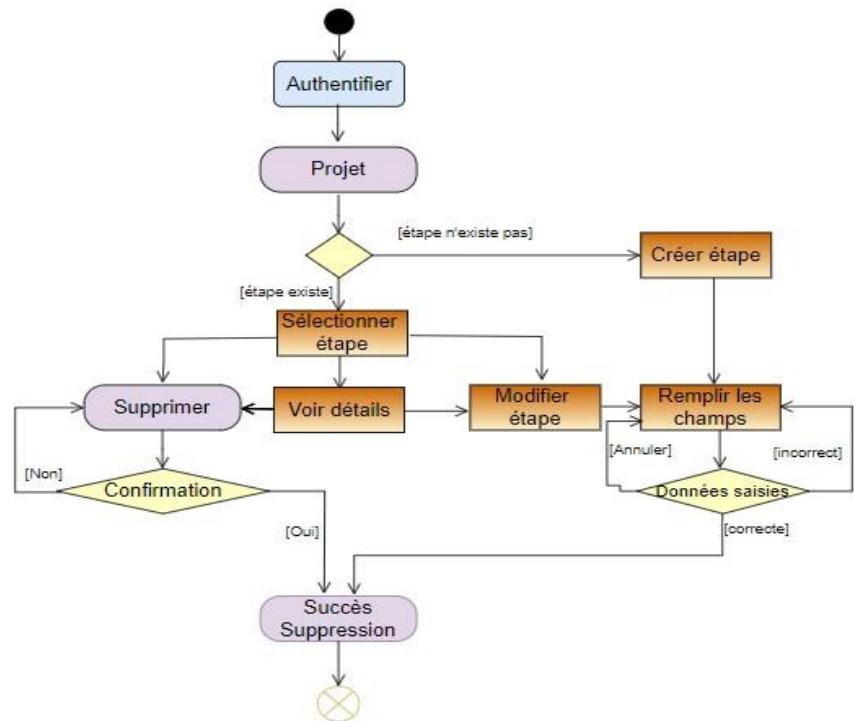


Figure 28: Diagramme d'activités \_Gérer Etapes

## Diagramme d'activités

### 4. : “Gérer Evènements”

Voici le diagramme d'activités élaboré pour le module d'évènements :

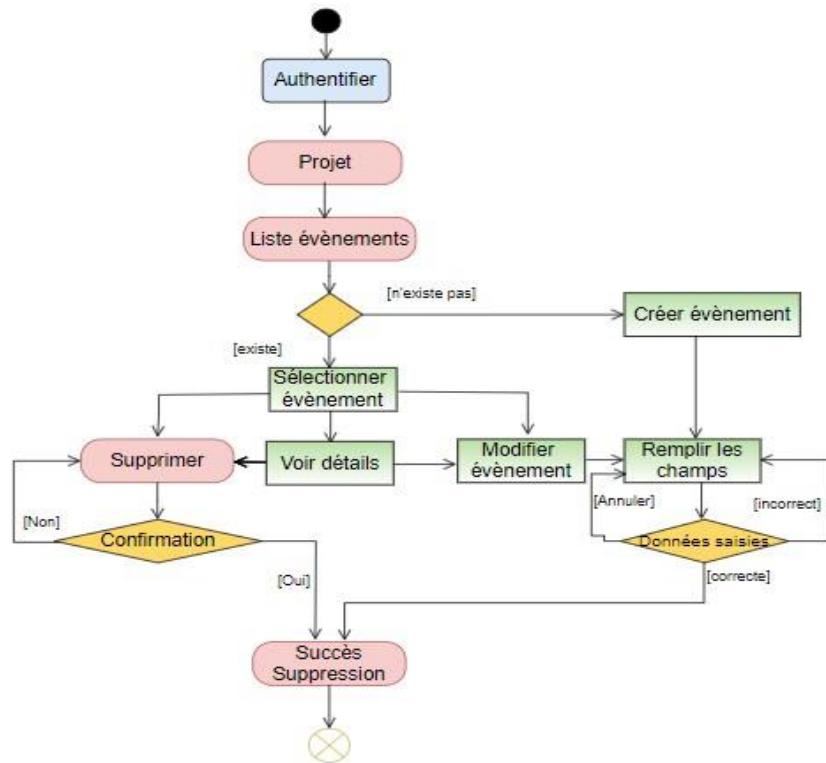


Figure 29: Diagramme d'activités\_Gérer Evènements

## 5. Diagramme d'activités : “Gérer Rapports”

Voici le diagramme d'activités élaboré pour le module de rapports :

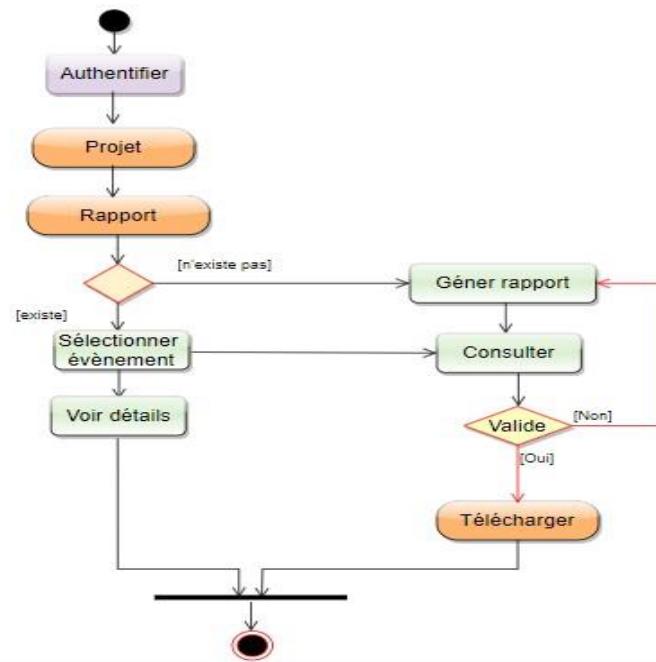


Figure 30: Diagramme d'activités \_Gérer Rapports

## 6. : “Module de Discussions et Notifications”

Voici le diagramme d'activités élaboré pour le module de discussion et notifications :

### Diagramme d'activités

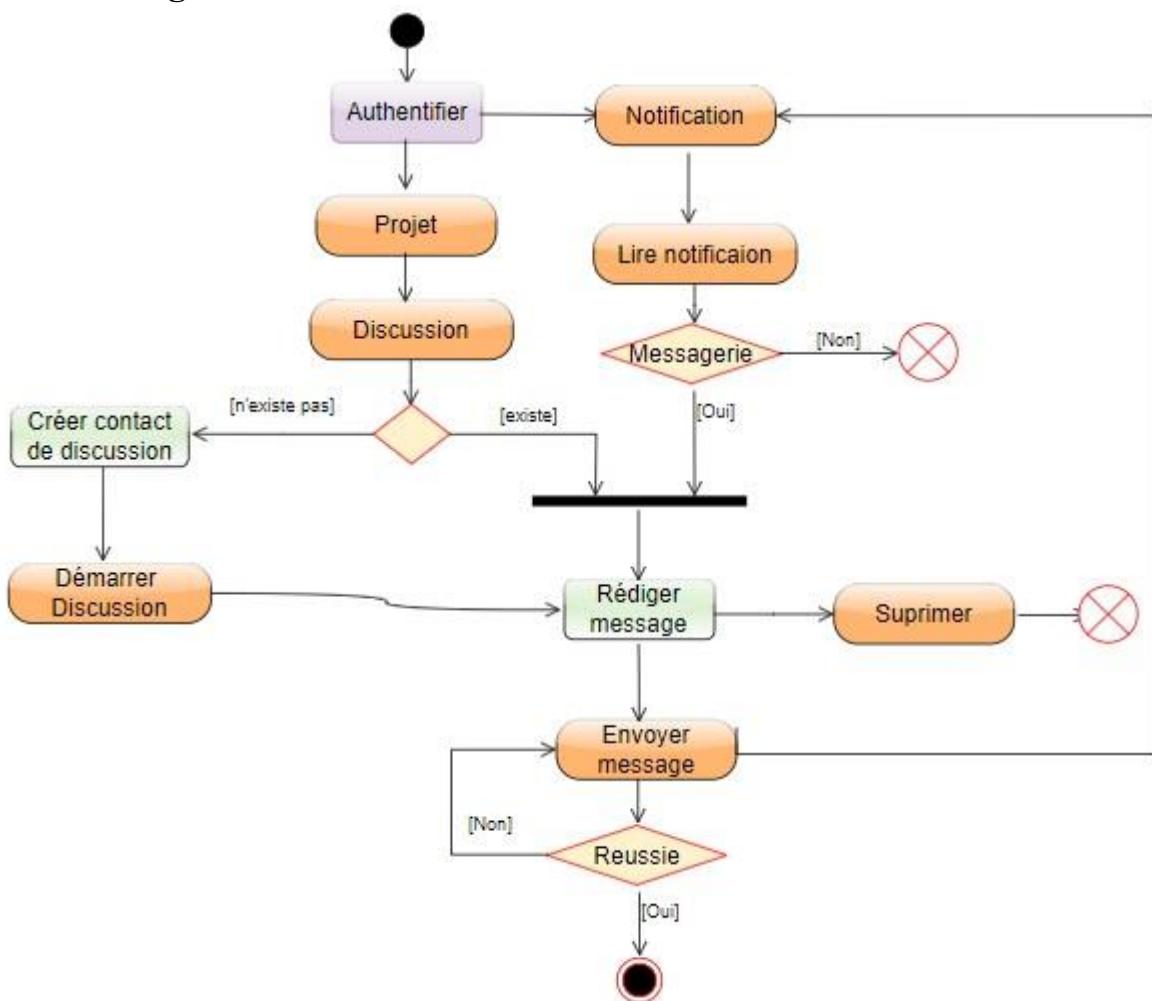


Figure 31: Diagramme d'activités \_Module de Discussions et Notifications

Cette étude fonctionnelle du système nous a permis de d'illustrer par ces diagrammes complémentaires et nous aident à obtenir une vue d'ensemble approfondie de la manière dont notre système répond aux besoins de nos utilisateurs. Ils serviront de guide précieux lors de la phase de développement, en assurant que l'expérience utilisateur soit intuitive et efficace.

### III. Diagramme de modèle

Le diagramme de modèle système représente les classes clés telles que "Utilisateur", "Projet", "Évènement", "Étape", "Fichier", "Rapport", "Discussions" et "Notifications", ainsi que leurs relations et interactions au sein du système de gestion de projets BuildTracker. Ce diagramme offre une vue d'ensemble des entités et de leurs liens, formant ainsi la base de notre conception et de notre implémentation.

## a. Raffinement des classes

Ces classes ci-dessous sont les classes présentes dans notre conception :



Figure 32: Raffinement des classes

## b. Relations entre les classes

Ce diagramme présente en soi les différentes relations et associations entre les classes :

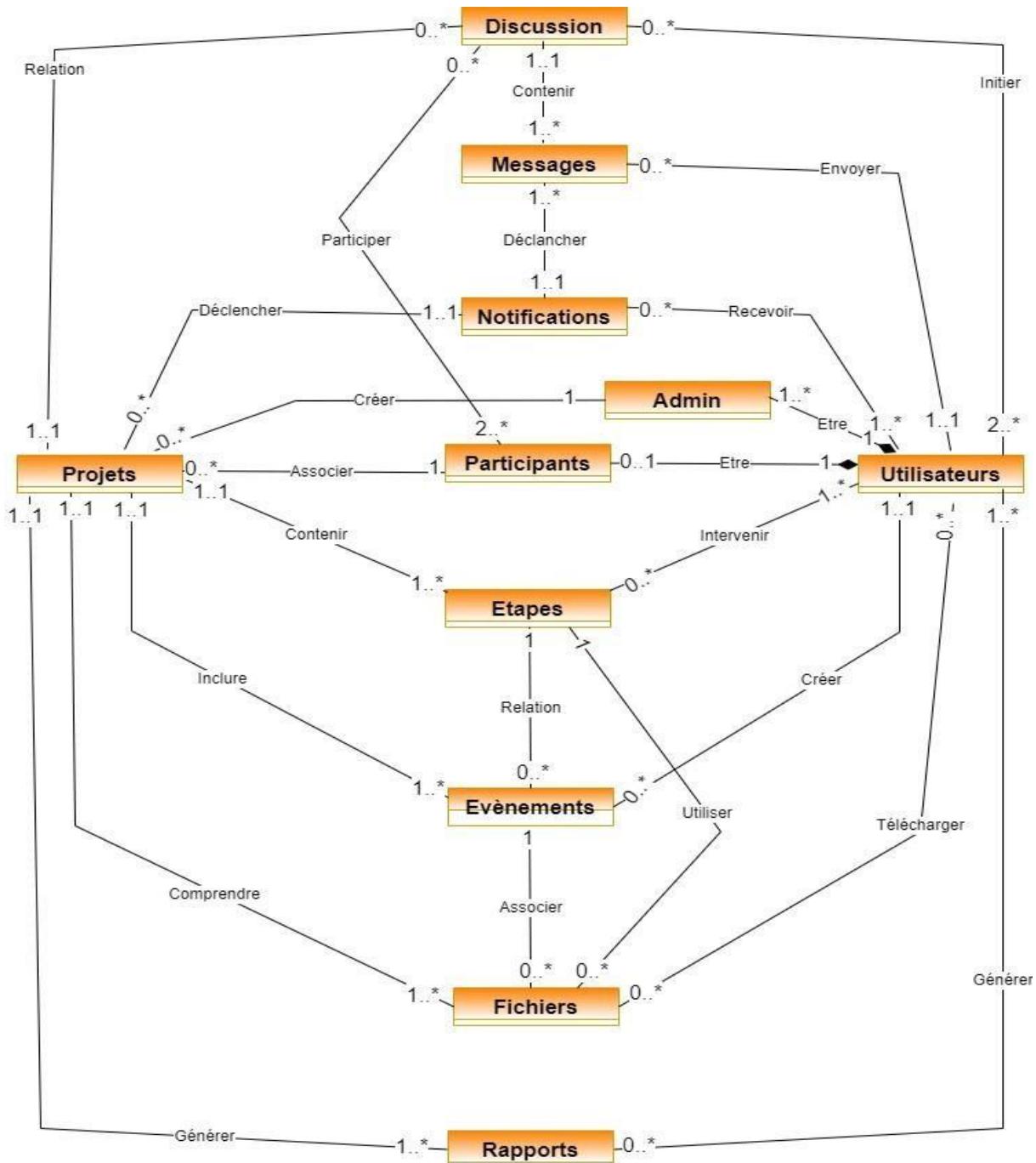


Figure 33: Diagramme de modèle des classes

## Conclusion

L'étude fonctionnelle et technique exhaustive de ce projet a permis de définir précisément les besoins et les fonctionnalités essentielles, tout en identifiant les acteurs clés. Cette étude détaillée nous a permis de mieux comprendre la façon dont les utilisateurs interagiront avec la plateforme et comment elle répondra à leurs besoins. Cette conception modulaire et extensible offre une base solide pour le développement du système, tout en garantissant une flexibilité nécessaire pour l'ajout futur de fonctionnalités.

## **Troisième partie: Réalisations et mise en œuvre**

### **Chapitre 1 : Mise en œuvre**

Ce premier chapitre de la partie "Réalisation et mise en œuvre" plonge dans les aspects pratiques de la création de notre application BuildTracker. Après avoir établi les bases conceptuelles et défini les objectifs dans les chapitres précédents, ce chapitre se penche sur les choix matériels et logiciels qui ont guidé notre processus de développement. Nous examinerons les décisions cruciales concernant l'infrastructure matérielle nécessaire au fonctionnement de l'application, ainsi que les logiciels et les technologies qui sous-tendent son architecture. Cette étape est essentielle pour garantir que notre application puisse fonctionner de manière fiable et efficace, en alignement avec nos objectifs.

## I. Choix matériels

Nous évaluerons les besoins en termes d'ordinateurs, serveurs, d'infrastructures réseau et d'autres équipements techniques nécessaires à une exécution optimale de l'application.

### † Desktop

Ce tableau ci-dessous représente les détails caractéristiques du desktop :

Tableau 7: Choix matériels \_Desktop

Unité centrale		
Stockage	Mémoire	256 Go SSD
Performance	Processeur	Intel(R) Core(TM) i5-6500
	Fréquence du processeur	3,20GHz, 3192MHz
	Nombre de cœur	04
	RAM	16 Go DDR4
Système d'exploitation	Système d'exploitation	Microsoft Windows 10 Professionnel
	Version	10.0.19045 Build 19045
Moniteurs HP et ASUS		
Ecran	Taille	24 Pouces
	Définition	HD 1928 x 1080 pixel

### † Laptop

Nous avons relevé les détails du laptop, consigné dans le tableau ci-dessous :

Tableau 8: : Choix matériels\_Laptop

<b>Système d'exploitation</b>	Système d'exploitation	Microsoft Windows 11 Professionnel
	Version	10.0.22621 Build 22621
<b>Ecran</b>	Tailles	15.6 Pouces
	Définition	1366 X 768 pixel
	Dalle	LCD
<b>Stockage</b>	Mémoire	256 Go SSD
<b>Performance</b>	Processeur	Intel(R) Core(TM) i3-1005G1
	Fréquence du processeur	1,20GHz, 1190MHz
	Nombre de cœur	02
	RAM	8 Go DDR4

## IV. Choix logiciels

Dans le processus de développement de notre système, l'usage de plusieurs logiciels nous ont été d'utilité capital et spécifique en fonction des besoins.

### a. Développement de backend API et la Base de Données † MongoDB

**Description :** MongoDB est un système de gestion de base de données NoSQL orienté document, adapté au stockage de données non structurées.



Figure 34: Logo MongoDB

**Pertinence et comparaison :** Contrairement à MySQL, MongoDB a été choisi pour sa flexibilité dans le stockage de données non structurées. Bien que MySQL soit robuste pour les données structurées, il n'est pas aussi adapté à notre gamme variée de données.

### † Node JS et Express JS

**Description :** Node.js est un environnement de développement basé sur JavaScript, tandis qu'Express est un framework minimal pour la création d'applications web et d'API.



Figure 35: Logo NodeJs et Express Js

**Pertinence et Comparaison :** Node.js et Express sont choisis pour leur évolutivité et leur rapidité d'exécution. Bien que Ruby on Rails soit une alternative, l'utilisation de JavaScript pour le développement backend est privilégiée pour sa cohérence avec le développement frontend.

### † Postman

Bien que des alternatives existent pour tester les API, Postman est devenu un choix naturel en raison de sa facilité d'utilisation et de sa capacité à automatiser les tests et la documentation des API, d'où notre choix pour le test de nos API.



Figure 36: Logo postman

## b. Développement Web † Nuxt JS et Vue Js

**Description :** Nuxt.js 3 est un framework JavaScript progressif pour le développement web, avec une orientation vers le rendu côté serveur (SSR). Vue.js 3 est une bibliothèque JavaScript pour la création d'interfaces utilisateur interactives.



Figure 37: Logo Nuxt et Vue.js

**Pertinence et Comparaison :** Contrairement à React et Angular, Nuxt.js 3 et Vue.js 3 offrent un rendu côté serveur, améliorant les performances et l'expérience utilisateur, leur simplicité d'intégration et de développement ainsi que leur documentation robuste les rendent adaptés aux besoins de notre projet. Les alternatives, bien que populaires, ne répondent pas à notre objectif de rendu côté serveur.

## † HTML5, CSS3 et JavaScript

L'intégration harmonieuse des langages de programmation est un aspect essentiel de notre projet de développement de l'application de gestion des projets de construction. JavaScript, HTML5 et CSS3 sont les piliers fondamentaux qui soutiennent l'architecture et l'expérience utilisateur de notre application.

**JavaScript** est un langage de programmation incontournable dans le développement web moderne. Dans notre projet, JavaScript est utilisé pour rendre les pages interactives et dynamiques. Il permet de manipuler le contenu des pages en temps réel, de gérer les interactions utilisateur et d'intégrer des fonctionnalités complexes. Nous l'utilisons également pour la communication avec les API, ce qui permet une expérience utilisateur fluide et réactive.

**HTML5** est la dernière évolution du langage HTML (HyperText Markup Language) utilisé pour structurer le contenu d'une page web. Dans notre projet, HTML5 est la base structurelle de chaque page. Il nous permet de définir la structure de l'application, d'organiser les éléments et de fournir un cadre cohérent pour le contenu. HTML5 introduit également de nouvelles balises et fonctionnalités qui facilitent l'intégration de médias, d'animations et d'éléments interactifs.

**CSS3** (Cascading Style Sheets) est un langage de feuille de style qui détermine l'apparence et la mise en forme des éléments HTML. Dans notre projet, CSS3 est utilisé pour styliser et personnaliser l'aspect visuel de l'application. Il nous permet de contrôler les couleurs, les typographies, les marges, les bordures et les effets visuels. CSS3 facilite également la création d'interfaces conviviales et est compatibles avec les normes de conception actuelles.

L'utilisation coordonnée de JavaScript, HTML5 et CSS3 garantit une application web moderne, interactive et conviviale pour les utilisateurs.



Figure 38: Logo d'HTML, CSS, JS

### † Tailwind CSS

L'adoption de Tailwind CSS a grandement accéléré le développement front-end en fournissant des classes utilitaires préconstruites pour le stylage.



Figure 39: Logo Tailwind CSS

**Comparaison :** Alors que Bootstrap est couramment utilisé, nous avons choisi Tailwind CSS pour son approche modulaire. Contrairement à Bootstrap, il nécessite moins d'efforts pour styliser les éléments individuels, et cela garantit une personnalisation plus fine et une conception plus légère.

## c. Développement Mobile

### † Dart et Flutter

Les langages de programmation Dart et Flutter jouent un rôle central dans le développement de la partie mobile de notre application de gestion, en particulier pour la création de l'interface utilisateur mobile. Ces technologies offrent des avantages significatifs en termes de performance, de rapidité de développement et de convivialité.

**Dart** est le langage de programmation utilisé dans le développement d'applications Flutter. Il se distingue par sa syntaxe simple et expressive, ainsi que par sa performance élevée. Dart facilite la création de code lisible et maintenable, ce qui contribue à la qualité globale de notre application.

**Flutter** est le framework open-source développé par Google pour la création d'interfaces utilisateur multiplateformes. Il offre un vaste ensemble de widgets prêts à l'emploi, permettant de créer des interfaces visuellement attrayantes et cohérentes sur Android et iOS. Flutter se distingue par sa rapidité de développement, sa réactivité et son hot-reloading, qui accélèrent la création et les tests d'interfaces.

L'utilisation conjointe de Dart et Flutter nous permet de créer une application mobile riche en fonctionnalités et réactive. Ces technologies simplifient le développement cross-platform, réduisant les efforts nécessaires pour maintenir deux bases de code distinctes. Dart et Flutter sont les choix optimaux pour réaliser une expérience utilisateur mobile fluide et uniforme dans notre application de gestion des projets de construction.



Figure 40: Logo Dart et Flutter

**Pertinence et comparaison :** Bien que React Native soit une option populaire, nous avons opté pour Dart et Flutter en raison de leur approche cross-platform plus avancée. Les inconvénients de React Native, tels que la nécessité de plugins pour certaines fonctionnalités natives, ont influencé notre choix.

### ❖ Android studio



Figure 41: Logo d'android studio.

Android Studio est l'IDE officiellement recommandé par Google pour le développement d'applications Android. Il offre un large éventail de fonctionnalités et d'outils qui simplifient et accélèrent le processus de développement. Il est conçu pour travailler de manière transparente

avec les technologies Android, ce qui en fait un choix naturel pour la création d'applications Android natives.

#### d. Conception Design UI/UX † Figma

**Description :** Figma est un outil de conception d'interface utilisateur et d'expérience utilisateur (UI/UX) basé sur le cloud. Il permet la création de maquettes, de prototypes interactifs et de designs collaboratifs en temps réel.

**Pertinence :** Figma est privilégié pour sa facilité d'utilisation, son environnement de travail en ligne qui facilite la collaboration, et sa capacité à créer des prototypes interactifs, ce qui est essentiel pour visualiser les fonctionnalités de l'application avant le développement.



Figure 42: Logo Figma

**Comparaison :** Bien qu'Adobe XD soit une alternative populaire, Figma a été privilégié pour sa collaboration en temps réel, qui dépasse les fonctionnalités offertes par Adobe XD. La possibilité pour plusieurs membres de l'équipe de travailler simultanément sur la conception en fait un choix plus efficace pour une collaboration transparente.

#### † Adobe photoshop et illustrator

Les alternatives de retouche d'images et de création de graphiques existent, mais Adobe Photoshop et Illustrator sont des références incontestables dans le domaine. Leur richesse en fonctionnalités et leur capacité à manipuler des images haute résolution ont été déterminantes pour la création d'éléments visuels attrayants.



Figure 43: Logo Adobe Photoshop et illustrator

#### e. Autres † Visual Studio Code



Figure 44: Logo VS Code

Visual Studio Code (VS Code) est l'environnement de développement intégré (IDE) privilégié pour notre application de gestion. Son interface intuitive, ses extensions puissantes, son intégration Git, ses outils de débogage et de profilage en font un choix judicieux pour améliorer l'efficacité, la collaboration et la qualité du code au sein de notre équipe. VS Code reflète notre engagement envers un développement de haute qualité et facilite la création d'une application robuste et performante.

## ⊕ GitHub



Figure 45: Logo GitHub

Dans le processus de développement de notre application nous avons utilisé GitHub. En tant que plateforme de gestion de code source et de collaboration, GitHub facilite le travail d'équipe, le suivi des versions de notre application, de modifications et la diffusion du code.

Il agit comme un référentiel central où tout le code source de notre application est stocké. Il permet aux membres de l'équipe de travailler sur différentes fonctionnalités et branches, de fusionner les modifications et de résoudre les conflits de manière transparente.

Ce chapitre ressort les bases technologiques de notre application BuildTracker. Les choix matériels et logiciels sont cruciaux pour assurer la robustesse et la performance de notre système. Les décisions prises ont été guidées par notre vision de fournir une solution de pointe à nos utilisateurs. Le chapitre suivant plongera plus profondément dans la présentation de l'application elle-même, explorant son architecture, la mise en place de la base de données et les mesures de sécurité mises en œuvre pour protéger les données sensibles.

# Chapitre 2 : Présentation de l'application

Dans ce chapitre, nous plongeons au cœur de notre projet en présentant en détail notre application de gestion des projets de construction. Nous commençons par une présentation globale de l'application. Ensuite, nous explorons l'architecture sous-jacente qui permet à notre application de fonctionner de manière harmonieuse, en expliquant comment les différents éléments interagissent pour offrir une expérience utilisateur optimale. Nous poursuivons en examinant la mise en place de la base de données, un pilier essentiel de notre système, et en détaillant les choix technologiques et les structures de données adoptées. Enfin, la sécurité de notre application est abordée en détail.

## I. Présentation

BuildTracker est basée sur une approche multiplateforme, avec une application web pour une utilisation depuis un navigateur et des applications mobiles pour les utilisateurs sur le terrain. Les technologies clés qui sous-tendent cette architecture incluent Dart et Flutter pour le développement mobile permettant une expérience utilisateur uniforme sur Android et iOS, ainsi que des solutions web standard pour la partie web.

## II. Architecture de l'application

L'architecture de notre application de gestion des projets de construction est conçue en suivant un modèle en couches, offrant une structure organisée et extensible.

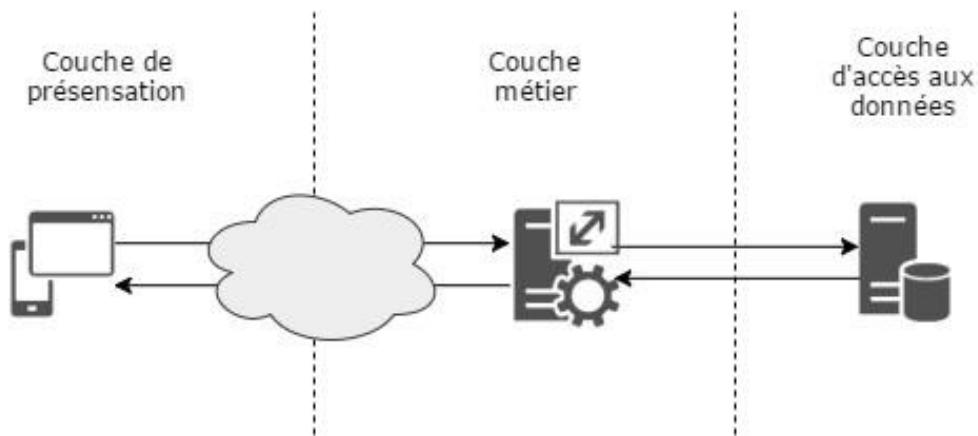


Figure 46: Architecture 3 tierre de l'application (Vue blog.lecacheur.com)

### a. Couche de Présentation

Cette couche est responsable de l'interaction avec les utilisateurs. Elle englobe l'interface utilisateur (UI) et les composants visuels. Nous utilisons Vue.js 3 avec Nuxt.js pour développer une interface utilisateur réactive et dynamique. Cette technologie permet de créer une expérience utilisateur fluide et intuitive.

## b. Couche Métier

La couche métier contient la logique métier de l'application. Elle gère les opérations, les règles et les traitements liés aux fonctionnalités. Nous utilisons Node.js avec Express pour mettre en œuvre la couche métier. Cette combinaison offre une gestion efficace des opérations côté serveur, en traitant les requêtes des clients et en interagissant avec la base de données.

Nous avons opter pour le schéma **Model-Vue-Contrôleur** qui est la base de notre programmation objet.

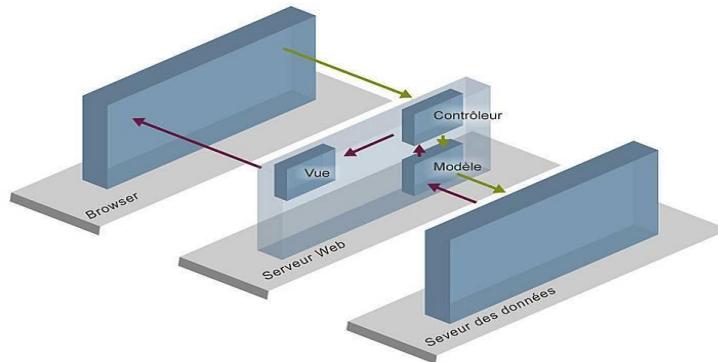


Figure 47: Schéma MVC-Model View Contrôleur ([slideplayer.fr/slide](http://slideplayer.fr/slide))

**Model** : un noyau de l'application qui gère les données, permet de récupérer les informations dans la base de données, de les organiser pour qu'elles puissent ensuite être traitées par le contrôleur.

**Vue** : composant graphique de l'interface qui permet de présenter les données du modèle à l'utilisateur.

**Contrôleur** : composant responsable des prises de décision, gère la logique du code qui prend des décisions, il est l'intermédiaire entre le modèle et la vue. **Quelques avantages du modèle MVC**

- Meilleur organisation du code
- Diminution de la complexité lors de la conception
- Conception claire et efficace grâce à la séparation des données de la vue et du contrôleur
- Possibilité de réutilisation de code dans d'autres applications
- Un gain de temps de maintenance et d'évolution du site
- Une plus grande souplesse pour organiser le développement du site entre différents développeurs

## c. Couche d'Accès aux Données

Cette couche est responsable de la communication avec la base de données. Nous utilisons MongoDB pour stocker les informations relatives aux projets, aux utilisateurs et aux autres données pertinentes. La couche d'accès aux données gère la persistance des données, les requêtes et les mises à jour.

Cette architecture en couches permet une séparation claire des préoccupations, favorisant la modularité et la réutilisabilité du code. Elle facilite également la collaboration au sein de l'équipe de développement et permet des mises à jour spécifiques à chaque couche sans affecter les autres parties de l'application.

### **III. Mise en place de la base de données**

Pour assurer une gestion efficace des données au sein de notre système, nous avons opté pour une approche basée sur une base de données MongoDB. Cette technologie NoSQL est adaptée à la gestion de données non structurées et offre une flexibilité accrue pour s'adapter aux besoins changeants de notre système.

Nous utilisons également Mongoose, une bibliothèque ODM (Object-Document Mapping) pour Node.js, qui facilite l'interaction entre notre application Node.js et la base de données MongoDB. Mongoose simplifie la création de schémas de données, les requêtes et les mises à jour de la base de données, tout en offrant une fonctionnalité pour la validation des données. Ainsi pour la mise en place applicative de la base de données de notre application BuildTracker, nous avons suivi une série d'étapes méthodiques :

#### **Étape 1 : Installation de MongoDB**

Nous avons commencé par télécharger et installer MongoDB sur notre ordinateur nous permettant d'avoir une base de données en local.

#### **Étape 2 : Configuration de MongoDB**

Après l'installation, nous avons configuré MongoDB en définissant les paramètres de sécurité, et en activant l'authentification. Nous avons également configuré MongoDB pour qu'il écoute sur le port approprié.

#### **Étape 3 : Création de la Base de Données et des Collections**

Nous avons créé une base de données MongoDB pour notre application, et à l'intérieur de cette base de données, nous avons défini plusieurs collections pour organiser nos données, notamment celles pour les projets, les étapes, les fichiers, les rapports, les discussions, les notifications et les événements. Voici la structure de notre base de données ci-dessous :

Collection	Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
otps	8.19 kB	0	0 B	2	24.58 kB
projets	20.48 kB	2	276.00 B	1	36.86 kB
services	20.48 kB	1	51.00 B	1	20.48 kB
tokens	20.48 kB	1	289.00 B	2	40.96 kB

Figure 48: Base de données MongoDB \_ BuildTracker

Voici également quelques parties du code permettant de créer des schémas s'ils n'existent pas dans la base de données :

Un exemple de code pour le cas du schéma de l'OTP

```
const mongoose = require('mongoose');
//Define a schema const Schema
= mongoose.Schema;
// Model for otp codes const
OtpModel = new Schema({
code: { type: String,
required: true,
unique: true,
},
phone :
{ type:
String,
required: true,
},
count :
{ type:
Number,
default : 0,
}
},
);
module.exports = mongoose.model('Otps', OtpModel);
```

#### Étape 4 : Gestion des Accès

Pour renforcer la sécurité de notre base de données, nous avons configuré l'authentification MongoDB en créant des utilisateurs et en leur attribuant des rôles avec des autorisations spécifiques pour chaque collection. Cela garantit que seules les personnes autorisées ont accès à certaines données.

En suivant ces étapes, nous avons réussi à mettre en place la base de données de notre projet.

## IV. Sécurité de l'application

Dans le souci de garantir la sécurité optimale de notre système, nous avons mis en place plusieurs mesures de sécurité pour protéger les données, les utilisateurs et les fonctionnalités sensibles.

Tout d'abord, nous avons implémenté un système d'authentification solide. Les utilisateurs doivent se connecter à l'application à l'aide de leurs identifiants uniques soit par email ou téléphone en OTP (One-Time Password), et différentes autorisations sont accordées en fonction de leur rôle dans le projet. Nous utilisons également des mécanismes de hachage et de salage pour stocker les mots de passe de manière sécurisée dans la base de données.

Nous avons également mis en place des contrôles d'accès rigoureux pour s'assurer que chaque utilisateur a uniquement accès aux fonctionnalités et aux données qui lui sont autorisées. Les rôles et les permissions sont gérés de manière centralisée, ce qui permet de limiter l'accès aux zones sensibles de l'application.

Le deuxième chapitre de la partie "Réalisation et mise en œuvre" a permis de présenter BuildTracker en tant qu'application concrète. Nous avons décrit ses fonctionnalités, son architecture, et les décisions prises pour garantir la sécurité des données. Dans les chapitres suivants, nous entrerons dans les détails des aspects techniques de développement et de déploiement.

## **Quatrième partie 4 : Exploitation et utilisation**

### **Chapitre 1 : Configuration matérielle et logicielle**

Sur la base des aspects pratiques de BuildTracker, nous aborderons la configuration matérielle et logicielle requise pour le bon fonctionnement de l'application. De plus, nous explorerons les détails du déploiement et du suivi de l'application, assurant ainsi une mise en service efficace.

## I. Configuration matérielle

Le tableau ci-dessous montre les ressources matérielles minimales et recommandées

*Tableau 9: Configuration matérielle*

Appareil	Ressource Minimale	Ressource Recommandée
<b>Serveur hébergeant le site web</b>	Processeur dual-core, 2,0 GHz	Processeur quad-core, 2,5 GHz ou plus, 8 Go de RAM, SSD pour le stockage
	RAM : 2 Go	RAM : 4 Go ou plus
	Espace de stockage : 50 Go (SSD recommandé)	Espace de stockage : 100 Go ou plus (SSD recommandé)
	Système d'exploitation : Linux (Ubuntu Server, CentOS, etc.)	Système d'exploitation : Linux (Ubuntu Server, CentOS, etc.)
	Certificat : SSL	Certificat : SSL
	Domaine : .com	Domaine : .tg ou .com
<b>Serveur hébergeant la base de données</b>	Bande passante : 100 Mbps	Bande passante : 1 Gbps
	Processeur quad-core, 2,5 GHz, 8 Go de RAM, SSD pour le stockage	Processeur hexa-core ou octa-core, 16 Go de RAM ou plus, SSD NVMe pour le stockage
	RAM : 2 Go	RAM : 8 Go ou plus

<b>Serveur hébergeant la base de données</b>	Espace de stockage : 20 Go (SSD recommandé)	Espace de stockage : 100 Go ou plus (SSD recommandé)
	Système de gestion de base de données (SGBD) : MongoDB	Système de gestion de base de données (SGBD) : MongoDB
	Système d'exploitation : Linux (Ubuntu Server, CentOS, etc.)	Système d'exploitation : Linux (Ubuntu Server, CentOS, etc.)
<b>Smartphone Android et iOS</b>	RAM : 2 Go	RAM : 4 Go ou plus
	Espace de stockage : 16 Go	Espace de stockage : 32 Go ou plus
	Processeur : Quad-core ou supérieur	Processeur : Octo-core ou supérieur
	Système d'exploitation : Android 8.0 (Oreo) ou supérieur	Système d'exploitation : Android 10 ou iOS 14 ou supérieur
<b>Poste Client</b>	Processeur : Dual-core, 1.8 GHz.	Processeur : Intel Core i5 ou AMD Ryzen 5 ou supérieur.
	RAM : 4 Go	RAM : 8 Go ou plus
	Espace de stockage : 126 Go (SSD recommandé)	Espace de stockage : 512 Go ou plus (SSD recommandé)
	Système d'exploitation : Windows 10 ou MacOs 10.15 (Catalina) ou supérieur	Système d'exploitation : Windows 11 ou MacOs 11 (Big Sur) ou supérieur

## II. Configuration logicielle

Le tableau ci-dessous montre les ressources logicielles minimale et recommandées

Tableau 10: Configuration logicielle

Appareil	Configuration minimale requise	Configuration recommandée
<b>Serveur d'hébergement web</b>	Type d'hébergement : cloud	Type d'hébergement : cloud
<b>Système de gestion de base de données (SGBD)</b>	MongoDB 4.0	MongoDB 5.0
<b>Poste client</b>	Système d'exploitation : Windows 10 ou MacOs 10.15 (Catalina) ou supérieur	Système d'exploitation : Windows 11 ou MacOs 11 (Big Sur) ou supérieur
	Navigateur : tout navigateur courant et à jour (chrome, Firefox, Microsoft Edge...)	Navigateur : tout navigateur courant et à jour (chrome, Firefox, Microsoft Edge...)
<b>Smartphone</b>	Système d'exploitation : Android, support 3G.	Système d'exploitation : Android, support 3G ou 4G

### III. Déploiement et suivi de l'application

#### a. Déploiement

Le déploiement de notre application de gestion de projets nécessite plusieurs étapes pour assurer qu'elle soit accessible aux utilisateurs.

Par ailleurs notons que l'application est en cours de développement et nous ne sommes pas encore à l'étape de déploiement de celle-ci. Voici un aperçu des principales étapes du déploiement que nous envisageons mener :

- ♦ **Préparation de l'Environnement Serveur** : nous allons choisir un fournisseur d'hébergement web et de base de données fiable, en l'occurrence AWS, pour garantir la disponibilité et la performance de notre application.

Nous allons configurer un serveur web, Apache, ainsi qu'une base de données MongoDB sur notre serveur d'hébergement.

- ❖ **Déploiement de l'Application :** nous allons transférer les fichiers de notre application sur le serveur web en utilisant SCP, garantissant ainsi un transfert sécurisé et efficace. La configuration de notre application a été soigneusement effectuée, y compris les fichiers de configuration et les clés d'API nécessaires.
- ❖ **Configuration de la Base de Données :** nous allons importer les schémas de base de données nécessaires dans notre système de gestion de base de données MongoDB. Les connexions à la base de données seront configurées dans notre application en utilisant les informations d'identification appropriées.
- ❖ **Tests et Vérifications :** des tests approfondis seront effectués pour nous assurer que l'application fonctionne correctement dans l'environnement de production.  
Nous allons mettre en place des mécanismes de sécurité, notamment des pares-feux et une protection contre les injections SQL, pour garantir la sécurité de notre application.
- ❖ **Domaine et Certificat SSL :** nous allons configurer un nom de domaine pour notre application, assurant ainsi une accessibilité conviviale.  
Un certificat SSL a été obtenu et installé pour sécuriser les connexions HTTPS entre les utilisateurs et notre application.

**Déploiement Mobile :** Pour notre version mobile de l'application, nous allons soumettre nos applications Android et iOS respectivement au Google Play Store et à l'App Store d'Apple.

Nous allons également s'assurer que les applications mobiles sont parfaitement synchronisées avec la version web de l'application.

**Surveillance et Maintenance :** Nous allons mettre en place des outils de surveillance avancés pour suivre en temps réel les performances de l'application, détecter les erreurs, et analyser le trafic.

Une fois ces étapes terminées, notre application sera prête à être utilisée par nos utilisateurs finaux. Nous nous engageons à maintenir une gestion rigoureuse du déploiement pour garantir la stabilité, la sécurité, et les performances de notre application tout au long de son cycle de vie.

## b. Suivi de l'application

Le suivi d'une application après son déploiement revêt une importance capitale pour garantir son bon fonctionnement continu et pour résoudre rapidement tout problème éventuel. Parmi les

principales actions prévues d'être entreprises pour le suivi de notre application, deux aspects majeurs sont mis en avant : le suivi des bugs et la gestion de versions.

## 1. Suivi des Bugs

Le suivi des bugs consiste à identifier, signaler, et résoudre tout dysfonctionnement ou anomalie qui pourrait survenir dans l'application après son déploiement. Cette démarche est essentielle pour maintenir une expérience utilisateur optimale. Pour ce faire, nous allons mettre en place une procédure structurée de gestion des incidents. Les étapes clés de cette procédure incluent :

**Identification des Bugs :** Les utilisateurs, ainsi que les membres de l'équipe de développement, sont encouragés à signaler tout bug ou problème qu'ils rencontrent lors de l'utilisation de l'application.

**Enregistrement et Priorisation :** Chaque incident est enregistré dans un système de suivi des bugs, tel que JIRA ou Trello. Ils sont ensuite évalués et priorisés en fonction de leur impact sur les utilisateurs et sur le bon fonctionnement de l'application.

**Assignation et Résolution :** Les bugs sont assignés à des membres spécifiques de l'équipe de développement pour résolution. Un processus de résolution structuré est suivi, incluant des phases de test et de validation.

**Communication :** Les utilisateurs sont informés des progrès de résolution des bugs, notamment de la disponibilité de correctifs ou de mises à jour.

## 2. Gestion de versions

La gestion de versions est une pratique nécessaire pour maintenir le code source de l'application sous contrôle. Elle permet de suivre l'évolution du code au fil du temps et de garantir la possibilité de revenir à des états précédents en cas de besoin. Pour notre projet, nous avons utilisé Git en conjonction avec le serveur de gestion de développement logiciel GitHub. Les avantages de cette approche incluent :

**Historique du Code :** Chaque modification apportée au code source de l'application est enregistrée avec des commentaires significatifs, permettant de suivre l'évolution de chaque composant.

**Branches de Développement :** Nous utilisons des branches de développement pour travailler sur de nouvelles fonctionnalités ou des correctifs sans perturber la branche principale (main).

**Intégration Continue :** L'intégration continue est utilisée pour automatiser les tests et garantir que le code nouvellement ajouté ne casse pas le fonctionnement existant de l'application.

**Déploiement Facilité :** La gestion de versions facilite grandement le déploiement, car nous pouvons sélectionner des versions stables de l'application pour être déployées en production.

Dans ce chapitre, Nous avons couvert les exigences matérielles et logicielles essentielles, ainsi que les procédures de déploiement et de suivi. La préparation minutieuse de ces éléments garantira une utilisation fluide et optimale de l'application.

## Chapitre 2 : Utilisation

Ce deuxième chapitre de la partie "Exploitation et utilisation" se concentre sur l'expérience utilisateur avec BuildTracker. Nous présenterons un plan de navigation détaillé, mettrons en lumière les principales interfaces de l'application et expliquerons les différents états qui peuvent se produire lors de son utilisation.

## I. Plan de navigation

La figure ci-dessous illustre le plan de navigation modifiable défini pour notre application

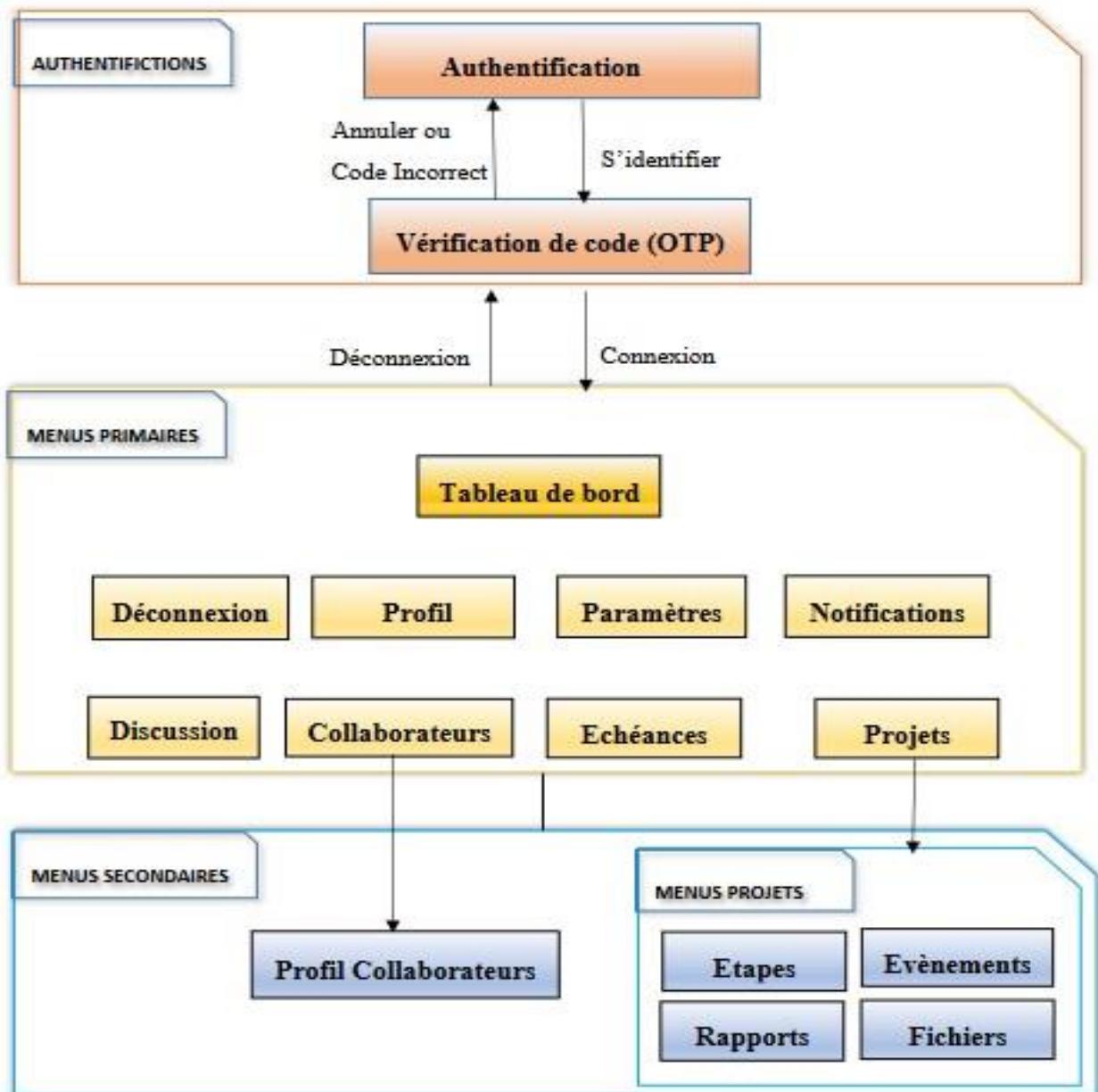


Figure 49: Plan de navigation de BuildTracker

### Explication du plan de navigation

**Authentification :**

- ◆ **Login :** Cette section permet aux utilisateurs de s'identifier à l'application en saisissant leurs informations d'identification soit l'email ou le numéro téléphone.
- ◆ **Vérification de code (OTP) :** Après l'identification les utilisateurs seront dirigés vers une étape de vérification de code à usage unique (OTP) pour renforcer la sécurité, ce code est reçu par email ou téléphone en fonction du choix.

### Menus Primaires:

- **Tableau de bord :** Cette page offre une vue d'ensemble des projets en cours, des étapes en cours à accomplir, des derniers événements, échéances à venir et informations importantes.
- **Projets :** Les utilisateurs peuvent accéder à cette partie pour voir la liste des auxquels l'utilisateur est identifié. Chaque projet est ensuite détaillé avec des sous-menus.
- **Echéances :** Cette section met en avant les dates limites importantes liées aux étapes des projets.
- **Collaborateurs :** Donne accès aux profils des membres de l'équipe, les clients, permettant de voir leurs rôles, responsabilités et informations de contact.
- **Discussions :** Offre un espace pour les discussions en temps réel entre les membres les collaborateurs.
- **Notifications :** Affiche les notifications importantes concernant les mises à jour, les étapes et les événements.
- **Paramètres :** Permet aux administrateurs de personnaliser leurs préférences et les réglages de l'application.
- **Profil :** Cette partie affiche les informations personnelles et les paramètres du profil de l'utilisateur.
- **Déconnexion :** Permet aux utilisateurs de se déconnecter de l'application.

### Menus Secondaires

Ces menus secondaires sont accessibles depuis différents endroits de l'application pour permettre une navigation rapide vers des sections spécifiques pour :

#### Projet

- **Etapes :** Permet aux utilisateurs de suivre et de gérer les différentes étapes d'un projet.

- **Évènements** : Donne un aperçu des événements prévus pour le projet, comme les réunions, les inspections, etc.
- **Rapports** : Offre la possibilité de générer et de consulter des rapports sur l'état d'avancement des projets
- **Fichiers** : Permet de télécharger, visualiser et gérer les documents et fichiers liés au projet.

## Utilisateurs

- **Profils utilisateurs** : Permet d'afficher et de mettre à jour les informations personnelles de chaque membre de l'équipe et clients.

Ce plan de navigation bien structuré que nous avons établi facilite la découverte et l'utilisation de l'application, tout en offrant aux utilisateurs un accès rapide aux fonctionnalités essentielles et en favorisant une expérience utilisateur fluide.

## II. Présentation de principales interfaces de l'application

Voici quelques interfaces principales de notre application

### a. Partie web

Page de connexion de notre application est sur la figure ci-dessous :

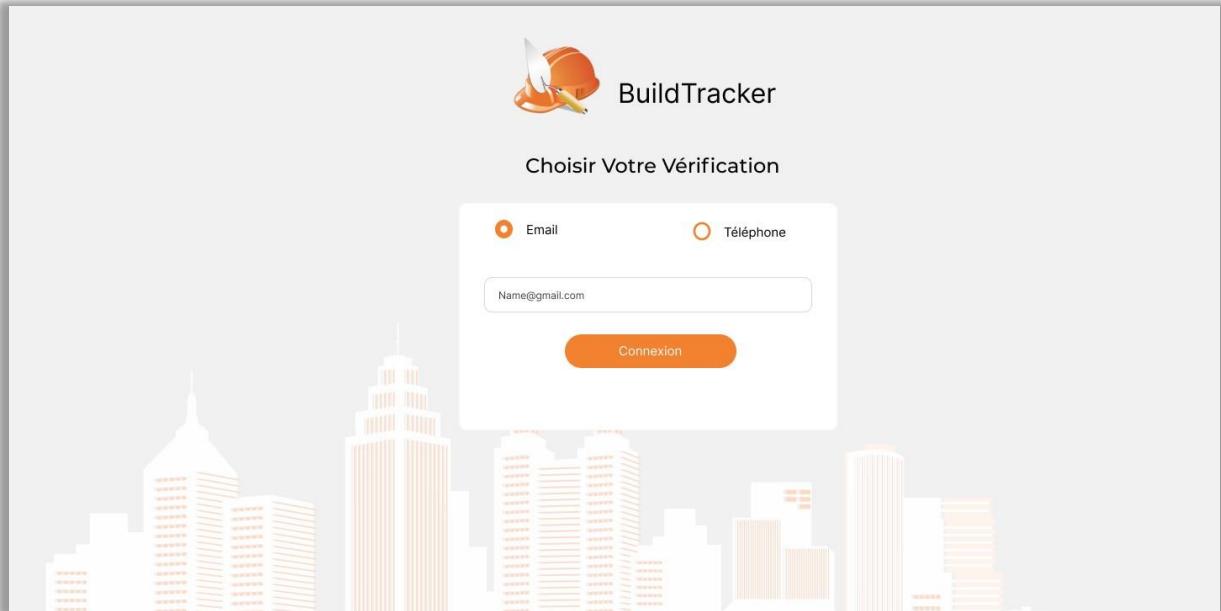


Figure 50: Page de connexion

Après vérification et réception du code, la page de vérification ci-dessous intervient

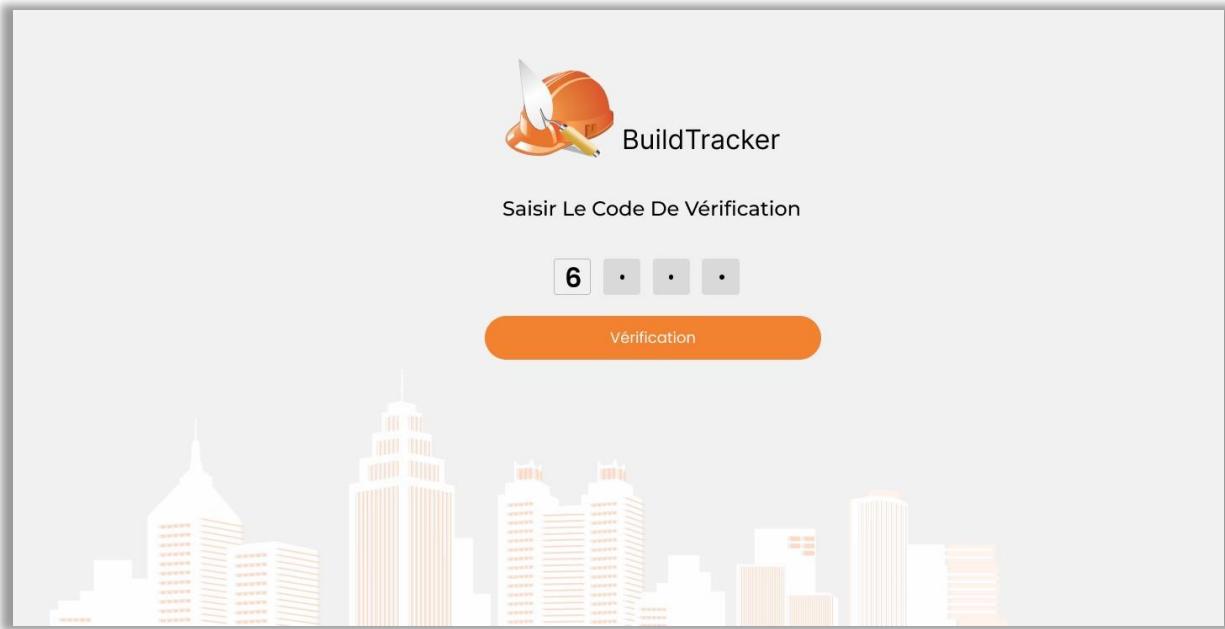


Figure 51: Page de vérification

La connexion étant établie, l'utilisateur sera dirigé sur sa page, ici le tableau de bord

A screenshot of the BuildTracker dashboard. The left sidebar contains a navigation menu with icons and labels: Tableau de Bord, Projets (with a notification badge), Echéances, Collaborateurs, Fichiers, Rapports, Discussions, Email, Calendrier, Paramètres, and Déconnexion. The main area features three large summary cards: "12 Projets" (orange card), "25 Techniciens" (pink card), and "04 Clients" (green card). Below these are three circular progress indicators: "Projets encours" (30 projets) at 22%, "Projets suspendu ou en retard" (12 projets) at 35%, and "Projets terminés" (12 projets) at 43%. A section titled "PROJET EN COURS" lists two projects: "Villa ELDERO" (Description...) with a start date of 12 Mars 2023, a 60% progress bar, and end date of 25 Juillet 2023; and "Rez Bakou" with a start date of 12 Mars 2023, a 95% progress bar, and end date of 25 Juillet 2023. Action buttons for each project include edit, delete, and a small orange icon.

Figure 52: Tableau de bord

Pour créer un nouveau projet, voici la page :

Figure 53: Page de création projet

Cette figure montre la liste des projets

Figure 54: Page de liste des projets

La liste de étapes est affichée ici :

## Quatrième partie : Exploitation et utilisation

Figure 55: Page des étapes

## b. Partie Mobile

L'image ci-dessous montre (1) Page de suivi évolution et (2) médias d'un projet

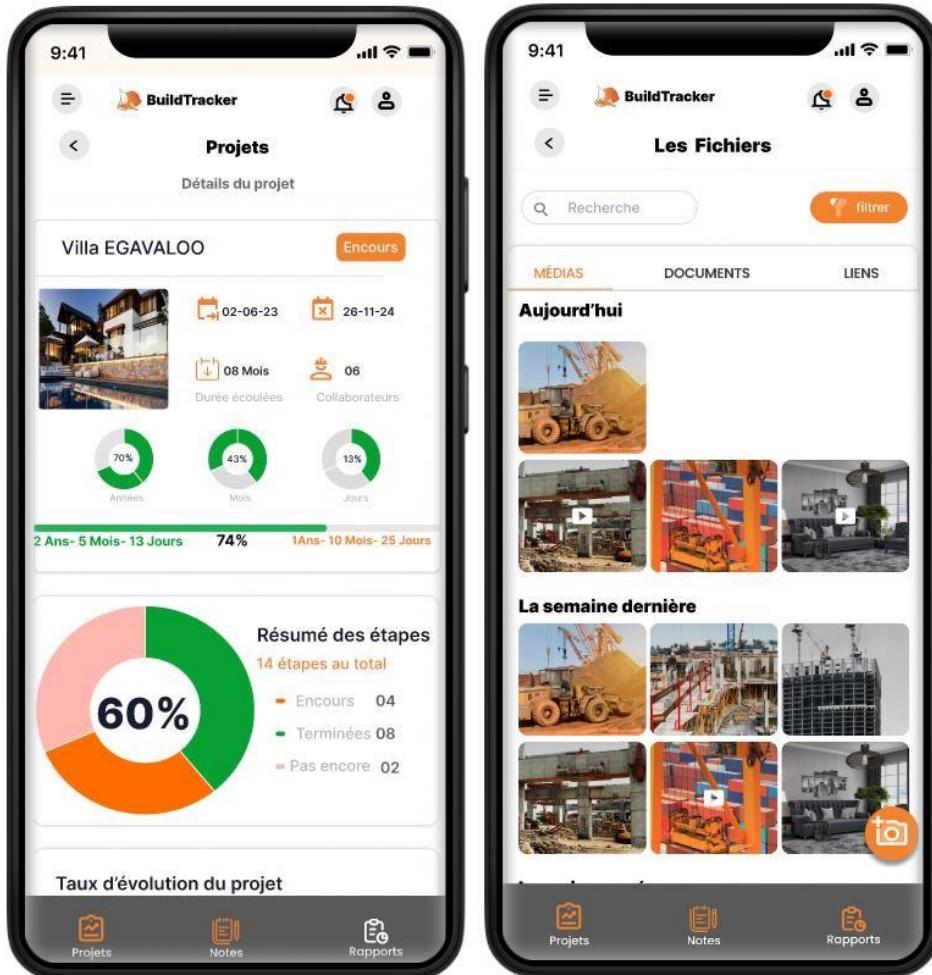


Figure 56: (1) Page de suivi évolution et (2) médias d'un projet

L'image ci-dessous montre (1) Pages documents et (2) page de profil

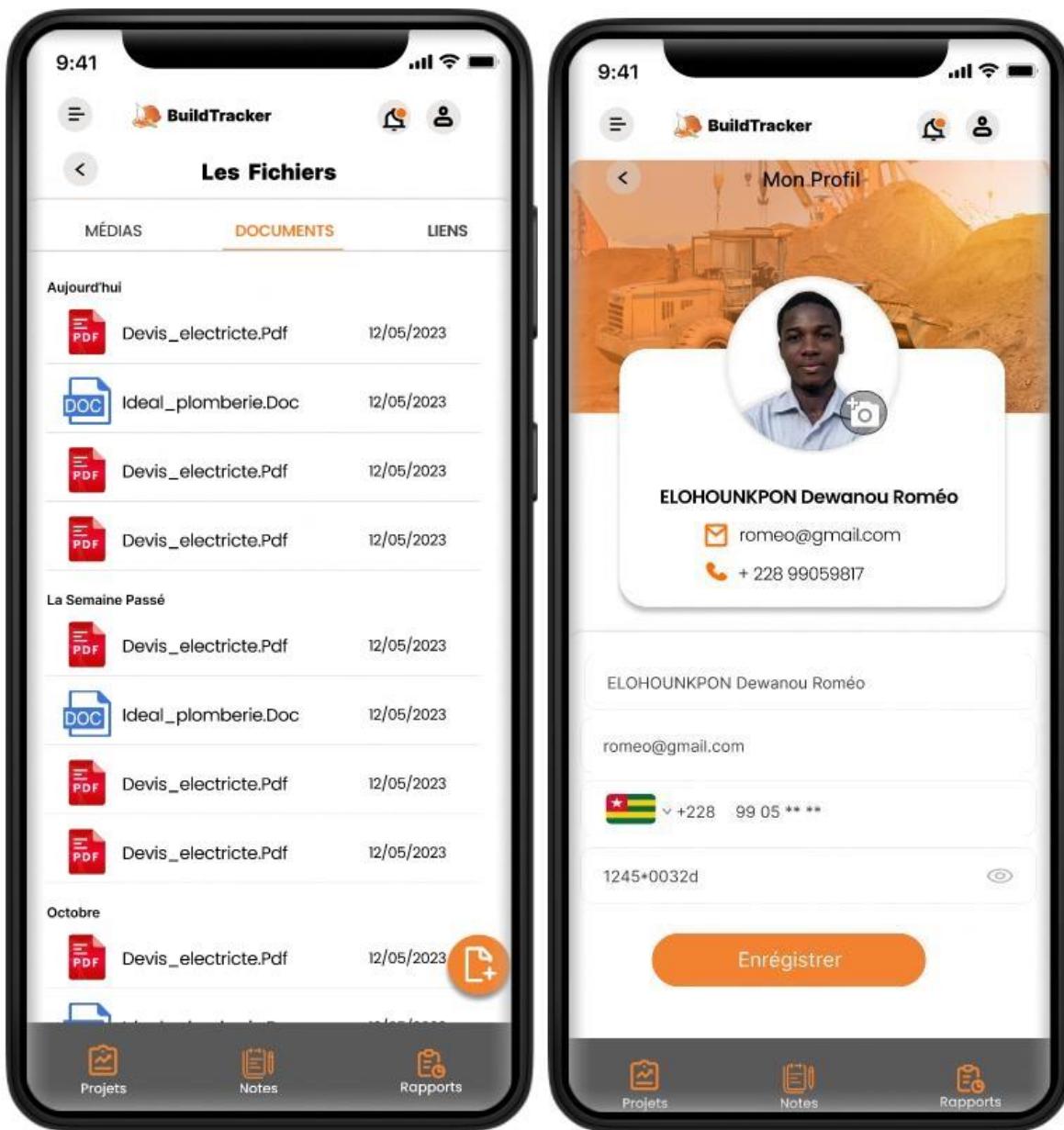


Figure 57: (1)Pages documents et (2) page de profil

### III. Présentation des états

Quelques états d'erreurs et de succès :

Erreur lors de la connexion avec un email inconnu par le système.

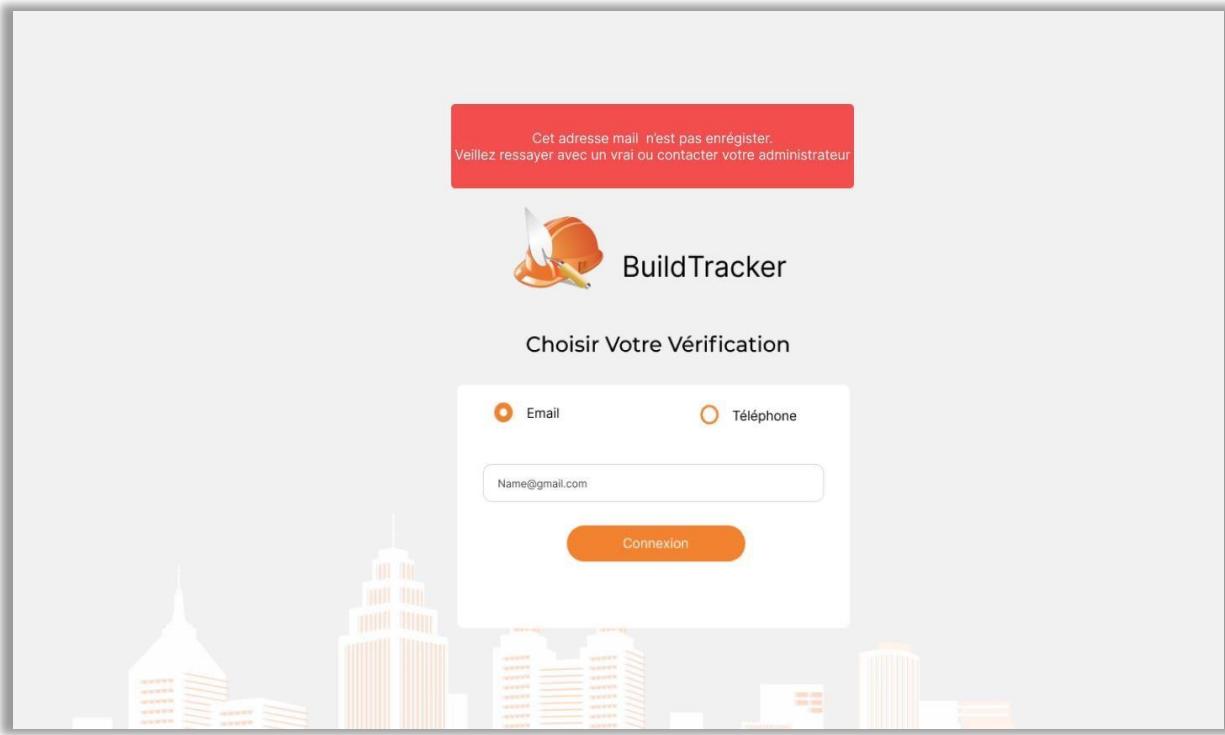


Figure 58: Erreur lors de la connexion avec un email

Message de bienvenue lorsque la connexion est établie

The screenshot shows the 'BuildTracker' dashboard. On the left, a sidebar includes links for Tableau de Bord, Projets (3), Echéances, Collaborateurs, Fichiers, Rapports, Discussions, Email (with 3 notifications), Calendrier, Paramètres, and Déconnexion. The main area features three large cards: '12 Projets' (orange), '25 Techniciens' (red), and '04 Clients' (green). Below these are three smaller cards: 'Projets encours 30 projets' (orange), 'Projets suspendu ou en retard 12 projets' (red), and 'Projets terminés 12 projets' (green). At the bottom, a section titled 'PROJET EN COURS' lists two projects: 'Villa ELDERO' (Description...) and 'Rez Bakou'. Each project card includes a progress bar (60% and 95% respectively), a due date (25 Juillet 2023), and action buttons.

Figure 59: Message de bienvenue lorsque la connexion est établie

(1) Réception de code par email et (2) absence d'évènement pour un projet.

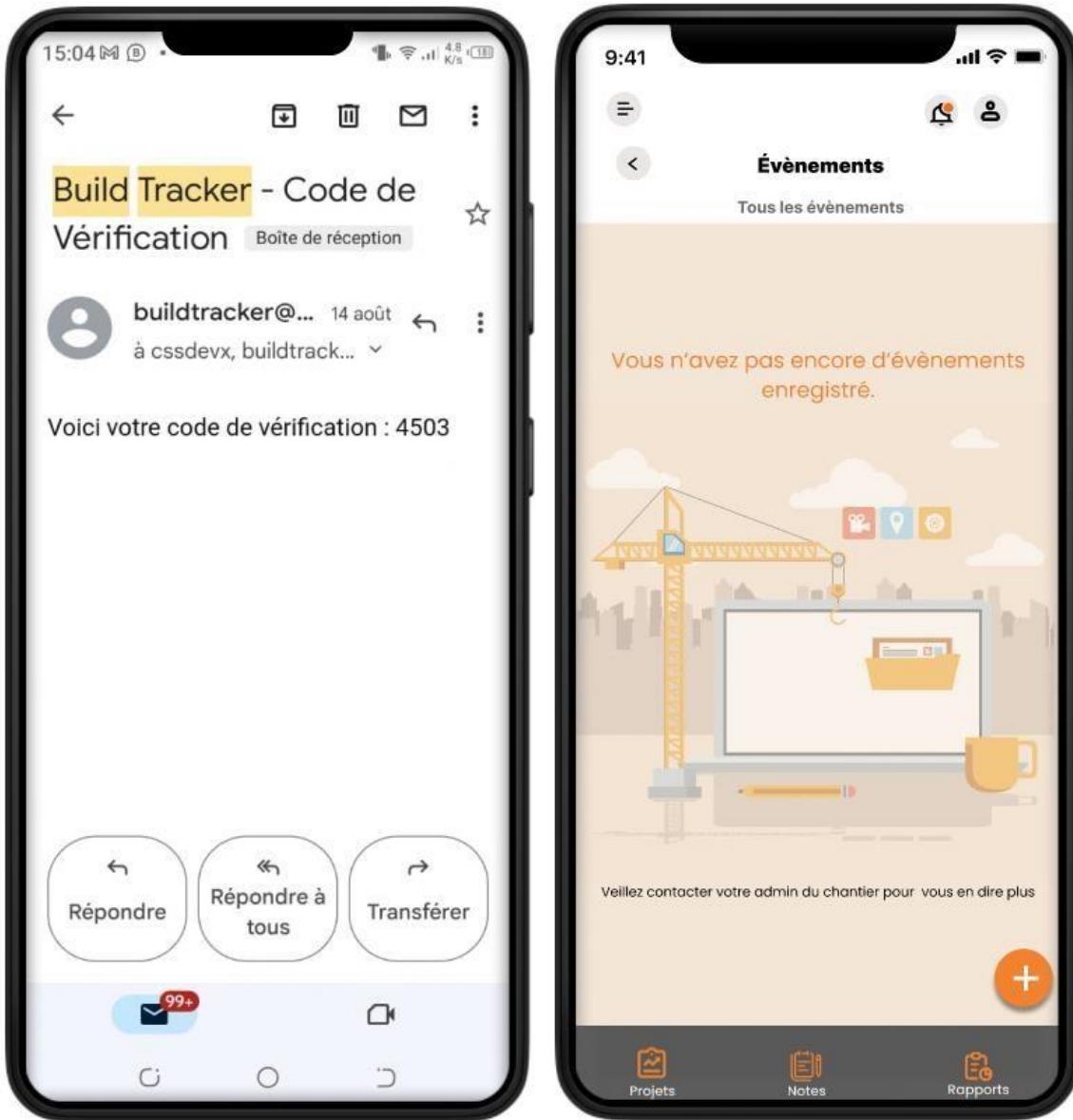


Figure 60: (1) Réception de code par email et (2) absence d'évènement pour un projet.

Ces éléments définissent la convivialité, l'expérience utilisateur et le fonctionnement global de l'application. Ils serviront de base solide pour la mise en œuvre de notre solution, en veillant à ce qu'elle réponde de manière efficace et intuitive aux besoins de nos utilisateurs.

Le deuxième chapitre de la partie "Exploitation et utilisation" a mis en avant l'expérience de l'utilisateur avec BuildTracker. Grâce à un plan de navigation clair et à la présentation des interfaces, les utilisateurs pourront tirer le meilleur parti de l'application. La compréhension des différents états facilitera également une utilisation efficace et productive de l'application.

# Conclusion générale

Ce document retrace notre parcours à CleanSoft Solution, au sein du projet de gestion de construction et de suivi de chantier. Notre travail s'est articulé autour de plusieurs phases clés, à savoir : l'analyse et la conception, la réalisation et la mise en œuvre, l'exploitation et l'utilisation. L'étude approfondie du projet a permis de cerner les besoins, de définir les acteurs majeurs impliqués, et d'établir les bases d'une architecture solide pour notre solution. Ainsi vient le choix des technologies les plus adaptées pour répondre aux exigences du projet.

Cependant, notre recherche n'est pas exempte de limites. Au niveau théorique, bien que nous ayons utilisé des cadres de gestion de projet établis, chaque organisation a ses particularités, et notre modèle peut nécessiter des adaptations pour d'autres contextes. De plus, au niveau empirique, notre projet a été mené en interne, et des études ultérieures avec des organisations externes pourraient fournir des perspectives supplémentaires.

Ce stage a été une expérience formatrice, nous permettant de mettre en pratique nos connaissances théoriques acquises au cours de notre formation. Il nous a permis de développer des compétences essentielles dans le domaine de la gestion de projets de construction. En guise de question pour l'avenir, nous nous interrogeons sur la manière dont l'application que nous avons conçue s'adaptera aux évolutions futures de l'industrie de la construction. Comment cette application pourrait-elle évoluer pour répondre aux besoins changeants et aux nouvelles technologies qui émergeront dans le secteur de la construction et l'intégration de technologies émergentes telles que l'intelligence artificielle ?

# Bibliographie

**M. SEWAVI Maurice** (2020-2023) – Cours d'**UML** 86 pages avec 10 chapitres. Cours dispensé au sein de l'ESGIS (*Ecole Supérieure de Gestion d'Informatique et des Sciences*). *Lomé-TOGO.*

**M. EUSEBIO** (2022-2023) à l'ESGIS (*Ecole Supérieure de Gestion d'Informatique et des Sciences*). Cours d'**UML** 58 pages, *Lomé-TOGO*

**M. ALADJI** (2021-2022) - Cours des langages **HTML, CSS et JavaScript** à l'ESGIS (*Ecole Supérieure de Gestion d'Informatique et des Sciences*), des slides avec 94 pages, *CANADA.*

**M. NOUSSOUGAN** (2022-2023) à l'ESGIS (*Ecole Supérieure de Gestion d'Informatique et des Sciences*). Cours d'**UML** 187 pages avec 5 chapitres, *Lomé-TOGO*

**M. OBANDJE** (2021-2022) - Cours de **Développement Logiciel** à l'ESGIS (*Ecole Supérieure de Gestion d'Informatique et des Sciences*), cas pratique, *Lomé-TOGO.*

**M. AMADJI** (2021-2022) - Cours de **Génie Logiciel** à l'ESGIS (*Ecole Supérieure de Gestion d'Informatique et des Sciences*), 75 pages-4 chapitre cas pratique, *Lomé-TOGO.*

# Webographie

- [1] Android Developer - (Site officiel du développeur Android, offrant une documentation complète et des ressources pour le développement)

d'applications Android) - <https://developer.android.com/> - Consulté en septembre 2023.

- [2] Apple Developer - (Site officiel du développeur Apple, proposant des ressources pour le développement d'applications iOS, macOS, watchOS et tvOS.) - <https://developer.apple.com/> - Consulté le mois de septembre 2023.
- [3] Stack Overflow (Communauté de développeurs) - <https://stackoverflow.com> - Consulté le mois de juin 2023.
- [4] OMG UML Specifications - (Site officiel de l'Object Management Group (OMG)) - <https://www.omg.org/spec/UML> - Consulté en juin 2023.
- [5] GitHub (Plateforme de développement collaboratif) - <https://github.com> - Consulté le mois de juin 2023.
- [6] NoSQL (Concepts et comparaisons des bases de données NoSQL) - <https://www.nosqldatabase.org> - Consulté depuis le mois de juin 2023.
- [7] Procore (Application de gestion architecturale) - <https://www.procore.com> - Consulté depuis le mois de juin 2023.
- [8] Wikipédia - (Site internet de recherche d'information) - <https://fr.wikipedia.org> - Consulté en mois de juillet 2023.
- [9] draw.io (Outil de dessin de diagrammes) - <https://app.diagrams.net> - Consulté le mois de juillet 2023.
- [10] PlanGrid - <https://www.plangrid.com> - Consulté depuis le mois de juillet 2023.
- [11] Instagantt (Outil de gestion de projet Gantt) - <https://instagantt.com> - Consulté depuis le mois de juillet 2023.
- [12] Figma (Outil de conception d'interface utilisateur) - <https://www.figma.com> - Consulté depuis le mois d'avril 2023.
- [13] MongoDB (Base de données NoSQL) - <https://www.mongodb.com> - Consulté depuis le mois d'août 2023.
- [14] Postman (Outil de test d'API) - <https://www.postman.com> - Consulté le mois d'août 2023.
- [15] Chat OpenAI Version 3.5 (OpenAI's ChatGPT) - <https://chat.openai.com> - Consulté le mois de juin 2023.

## Table des matières

Remerciements .....	ii
	102

Sommaire .....	iii
Résumé .....	v
Abstract .....	vi
Glossaire .....	vii
Liste des figures .....	ix
Liste des tableaux .....	xi
Liste des participants au projet .....	xii
Introduction .....	1
Première partie : Cahier des charges .....	2
Chapitre 1 : Présentations .....	4
I.    Présentation de l'école .....	4
II.   Présentation du cadre de stage .....	5
a.   Prise de contact et formalité administrative .....	5
b.   Mission de la structure d'accueil .....	5
c.   Situation géographique .....	5
d.   Prestation de services .....	5
e.   Organigramme de la société.....	6
f.   Environnement matériel et logiciel de la structure d'accueil .....	6
g.   Notre mission et apport dans la société lors du stage .....	6
Chapitre 2 : Thème de stage .....	7
I. Présentation du sujet et étude du domaine .....	7
a.   Présentation du sujet .....	7
b.   Etude du domaine de construction .....	7
III. Problématique du sujet .....	9
IV. Intérêt du sujet .....	10
a.   Objectif .....	10
b.   Résultats attendus .....	11
V. Étude de l'existant .....	12
VI. Critique de l'existant .....	12
Chapitre 3 : Proposition de solutions.....	14
I. Spécification des solutions.....	14
a.   Fonctionnelle .....	14
b.   Sécurité .....	15
II. Proposition de solutions .....	16

a. Proposition de Solution 1 : Odoo pour la Gestion des Projets .....	16
b. Proposition de Solution 2 : BuildTracker (Suivie des constructions). ....	17
Chapitre 4 : Choix de la solution .....	19
I. Solution retenue .....	19
II. Evaluation financière de la solution retenue.....	21
a. Coût matériel .....	21
b. Coût humain .....	22
c. Coût totaux .....	23
Coûts totaux .....	23
III. Planning prévisionnel de réalisation .....	23
Deuxième partie : Analyse et conception .....	26
Chapitre 1 : Présentation des outils d'analyse de modélisation .....	28
I. Présentation de la méthode analyses .....	28
II. Outils et language de modélisation .....	29
Chapitre 2 : Étude détaillée de la solution .....	30
I. Étude fonctionnelle.....	30
a. Les Fonctionnalités clés du système .....	30
b. Les acteurs principaux de L'application .....	32
c. Diagramme de cas d'utilisation .....	33
II. Étude technique .....	50
a. Diagramme de séquences .....	50
b. Diagramme d'activités .....	58
III. Diagramme de modèle .....	61
a. Raffinement des classes .....	62
b. Relations entre les classes .....	62
Troisième partie: Réalisations et mise en œuvre .....	64
Chapitre 1 : Mise en œuvre .....	65
I. Choix matériels .....	65
IV. Choix logiciels .....	66
a. Développement de backend API et la Base de Données .....	66
b. Développement Web .....	68
c. Développement Mobile .....	69
d. Conception Design UI/UX .....	71
e. Autres.....	72
Chapitre 2 : Présentation de l'application .....	73

I.	Présentation .....	73
II.	Architecture de l'application .....	73
a.	Couche de Présentation .....	73
b.	Couche Métier .....	74
c.	Couche d'Accès aux Données .....	75
III.	Mise en place de la base de données .....	75
IV.	Sécurité de l'application .....	77
Quatrième partie 4 : Exploitation et utilisation .....		78
Chapitre 1 : Configuration matérielle et logicielle .....		79
I.	Configuration matérielle .....	79
II.	Configuration logicielle .....	80
III.	Déploiement et suivi de l'application .....	81
a.	Déploiement .....	81
b.	Suivi de l'application .....	82
Chapitre 2 : Utilisation .....		85
I.	Plan de navigation .....	85
II.	Présentation de principales interfaces de l'application .....	87
a.	Partie web .....	87
b.	Partie Mobile .....	90
III.	Présentation des états .....	91
Conclusion générale .....		94
Bibliographie .....		I
Webographie .....		II
Table des matières .....		III