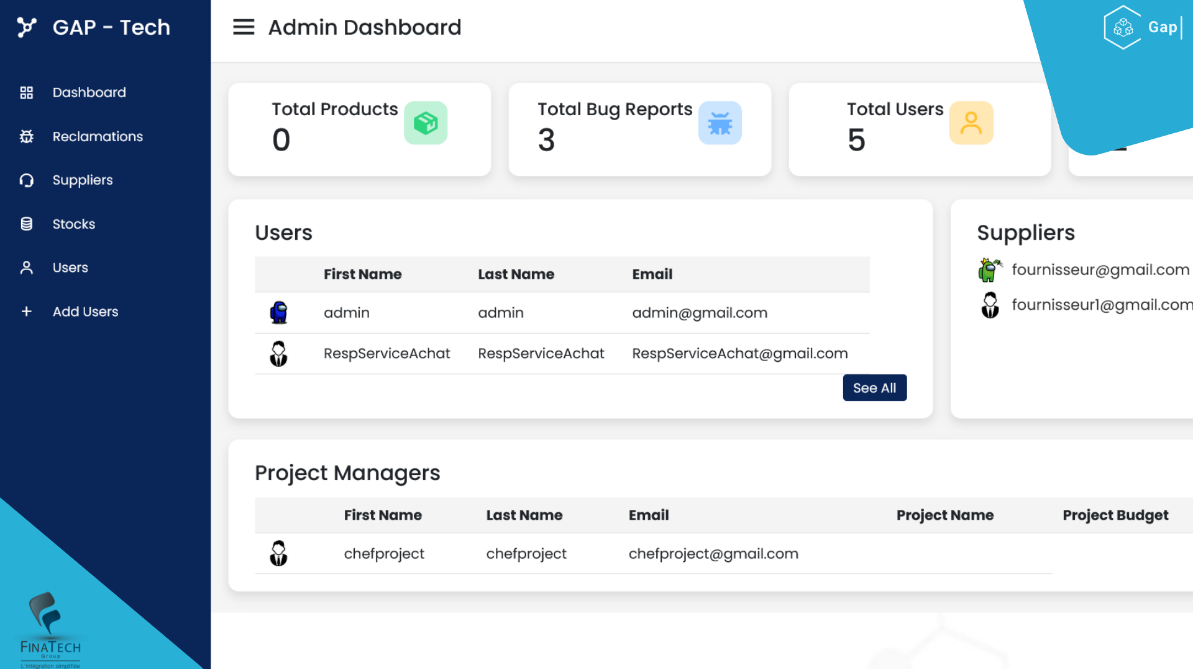
المدرسة المغربية لعلوم المهندس

**École marocaine des sciences de l'ingénieur**

|  |
| --- |
| **Projet de stage :**  **Plateforme Web d'Approvisionnement et d'Automatisation des Achats de Matières Premières** |

****

#### 

#### Réalisé par :

Yahya Rabii

#### Société d’accueil :

FinaTech Group

|  |  |
| --- | --- |
| **Encadrent :** | **Chawki El Balmany** |
| **Superviseur :** |  |

**Remerciements**

**C**e travail n’aurait jamais pu se concrétiser sans l’aide et le soutien de plusieurs personnes que nous souhaitons vivement remercier et à qui nous dédions ce travail.

**M**onsieur **Chawki El Balmany** notre encadrant, qui n'a pas cessé de nous prodiguer ses conseils et qui n'a épargné aucun effort pour contribuer à la réussite de ce travail.

**À** ce titre, je souhaiterais remercier tout particulièrement **M**adame Hajar, qui m’a épaulée et conseillée et qui m’a surtout transmis son expertise dans le domaine

**M**onsieur **‘’Ibrahim Barghout’’**, pour les directives qu’il nous a fournies pendant la durée de stage.

**L**es personnels de l’entreprise « FinaTech group », pour l'expérience enrichissante et pleine d’intérêt qu’ils nous ont fait vivre.

**T**ous nos professeurs et plus particulièrement les membres de jury qui ont accepté de juger notre travail.

**N**otre Institut qui nous a donné l'occasion d'acquérir une formation professionnelle.



Yahya rabii

**Table des matières**

[Introduction Générale 9](#_bookmark0)

[Chapitre 1 : cadre Générale du Stage 10](#_bookmark0)

1. [Présentation de la société 11](#_bookmark0)
   1. [Introduction générale 11](#_bookmark1)
   2. [Logo de FinaTech groupe](file:///C:\Users\yahya\AppData\Roaming\Microsoft\Word\yahya_RAPPORT.docx) 12
   3. FinanceCom 12
   4. [Fiche technique de FINATECH Group](#_Fiche_technique_de) 13
   5. [Activités de FinaTech Groupe 14](#_TOC_250005)
2. Cadre général du travail 16
3. [Présentation du sujet 16](#_bookmark2)
   1. [Présentation du projet 16](#_bookmark3)
   2. [Le but à atteindre 16](#_bookmark4)
   3. [Travail demandé 16](#_bookmark5)
4. Planification du projet 17

[Chapitre 2 : Analyse et spécifications des besoins 18](#_bookmark0)

[Introduction 19](#_bookmark6)

1. [Étude préalable 19](#_bookmark7)
   1. [Étude de l’existant 19](#_bookmark8)
      1. [Description de l’existant 19](#_bookmark9)
      2. [Critique de l’existant 19](#_bookmark10)
   2. [Solutions envisagées 20](#_bookmark11)
2. [Spécifications des Besoins 20](#_bookmark12)
   1. [Les Besoins fonctionnels 20](#_bookmark13)
   2. [Les Besoins non fonctionnels 21](#_TOC_250004)
      1. [Contraintes ergonomiques 21](#_bookmark14)
      2. [Contraintes techniques 21](#_bookmark14)

[Conclusion 21](#_bookmark6)

[Chapitre 3 : Etude conceptuelle 22](#_bookmark0)

[Introduction 22](#_bookmark15)

1. [Cycle de vie de développement de projet 22](#_bookmark16)
2. [Langage UML 25](#_TOC_250003)
   1. [Présentation du langage UML 25](#_bookmark17)
   2. [Intérêt de la modélisation 25](#_bookmark18)
   3. [Les avantages d’UML 25](#_bookmark18)
3. [Conception avec UML 26](#_TOC_250002)
   1. [Outil de modélisation 26](#_bookmark19)
   2. [Modélisation avec les diagrammes cas d’utilisation 26](#_bookmark20)
   3. [Modélisation avec les diagrammes de Séquences 34](#_bookmark20)
   4. [Modélisation avec les diagrammes de Classes 38](#_bookmark20)
   5. [Code MAP 39](#_bookmark20)

[Conclusion 39](#_bookmark15)

[Chapitre 4 : La Réalisation 40](#_bookmark0)

[Introduction 40](#_bookmark15)

1. [Environnement de travail 40](#_bookmark21)
   1. [Environnement matériel 40](#_bookmark22)
   2. [Environnement logiciel 42](#_TOC_250001)
2. [Développement 42](#_TOC_250000)
3. [Système d'exploitation 42](#_bookmark22)
4. [Outil de développement 42](#_bookmark22)
5. [Environnement de développement intégré 44](#_bookmark22)
6. [Outil administration Base de données 45](#_bookmark22)
7. [Outil de conteneurisation 46](#_bookmark22)
8. [Outil de Contrôle de Version 46](#_bookmark22)
9. [Langage de programmation 47](#_bookmark22)
10. Les interfaces graphiques 48
    1. [Interfaces 48](#_bookmark22)
11. Interface d’authentification 48
12. Interface d’inscription 49
13. Interface Accueil 50
    1. [Déploiement 52](#_bookmark22)

[Conclusion Générale 54](#_bookmark0)

**Liste des figures**

Figure 1: Logo de l'entreprise 11

Figure 2: Appartenance au groupe FINANCECOM 12

Figure 3 : Localisation de l'entreprise 13

Figure 4: Le modèle en cascade 25

Figure 5: Le diagramme de cas d'utilisation 28

Figure 6: Le diagramme de séquences « authentification » 29

Figure 7: Le diagramme de séquences « inscription » 30

Figure 8 : Le diagramme de séquences « Supprimer utilisateur » 31

Figure 9: Le diagramme de séquences « passer commande » 32

Figure 10: Le Java Développent Kit architecture 35

Figure 11: Android Studio Logo 36

Figure 12: MySQL Logo 37

Figure 13: Firebase Logo 37

Figure 14: SQLite Logo 37

Figure 15: JAVA Logo 38

Figure 16: XML Logo 38

Figure 17 : Interface Authentification 39

Figure 18: Interface inscription 39

Figure 19: Interface Mot de passe oublier 39

Figure 20: Interface Accueil 40

Figure 21: Interface liste des produits 40

Figure 22: Interface de règlement Cas 1 : Pickup 41

Figure 23: Interface de règlement Cas 2 : Delivery 41

Figure 24: Interface de validation 42

Figure 25: Interface stock (Nos produits) 43

Figure 26: Interface Nos fournisseurs (Nos contacts) 43

Figure 27: processus d'authentification 44

Figure 28: processus d'ajouter produit automatiquement 45

Figure 29: processus d'ajouter produit manuellement 45

Figure 30: processus d'ajouter nouveau contact 46

Figure 31: Code QR 47

## Liste des Acronymes

* **IT SYSTEMS :** Information technology systems
* **UML :** Unified Modeling Language
* **JDK :** Java Development Kit
* **JSON :** JavaScript Object Notation
* **ECMA :** European Computer Manufacturers Association
* **OS :** operating system
* **XML :** extensible markup language
* **SQL :** Structured Query Language
* **QR Code :** quick response code

**Liste des tableaux**

**Tableau 1** : Fiche technique de FINATECH Group 16

**Tableau 2** : planification temporelle 16

**Introduction générale**

Depuis les temps anciens, l'humanité n'a jamais relâché son incessante quête de moyens divers pour transmettre ses messages, établissant ainsi des voies de communication vitales. À travers les époques successives, cette poursuite a mobilisé tant son ingéniosité que son énergie physique, afin de découvrir des méthodes de communication adaptées et efficaces.

Jusqu'au 20ème siècle, une révolution dans le domaine des télécommunications a vu le jour, stimulée par le développement technologique fulgurant et la diffusion généralisée d'Internet. Cette avancée a considérablement accéléré la transmission des informations entre émetteurs et récepteurs. À présent, les flux d'informations se déploient de manière étendue, se diversifient, adoptent des directions bidirectionnelles et deviennent aisément accessibles.

À notre époque, dans un marché des applications web en plein essor, propulsé par la montée en puissance des tablettes informatiques, le nombre d'applications mobiles téléchargées connaît une croissance exponentielle d'année en année.

Dans ce contexte foisonnant, un nombre croissant d'entreprises opérant dans le secteur des télécommunications envisage d'investir considérablement dans ce domaine. Parmi ces pionnières se profile FinaTech Groupe.

Au sein de mon projet de stage, j'ai été confié à la tâche de concevoir une application Android dont la finalité est de connecter le service d'achats de l'entreprise FinaTech avec ses divers fournisseurs.

Ainsi, j'ai organisé mon rapport autour de quatre chapitres :

Dans le premier chapitre « cadre général », j'ai expliqué le cadre de mon projet.

Dans le deuxième chapitre « analyse et spécification des besoins », j'ai détermines les acteurs du futur système, les besoins fonctionnels et non fonctionnels.

Dans le troisième chapitre « Étude conceptuelle », j'ai modélisé une conception des diagrammes de cas d’utilisation, de diagramme des séquences complet.

Dans le dernier chapitre « réalisation », je présente l’environnement matériel et logiciel, ainsi que les interfaces principales de l’application réalisée.



**Chapitre 1**

# Cadre Général

# Du Stage

**Chapitre 1 : Cadre général du stage**

### **Introduction**

Dans ce chapitre je mets mon travail dans son contexte général, Tout d’abord, j'ai écrit l’environnement du stage à travers une brève présentation de l’organisme d’accueil. Ensuite j'ai présenté le cadre général de ce projet.

### **Présentation de la société**

Dans cette partie, je présente l’organisme d’accueil qui a proposé ce sujet.

### **Introduction générale**

**Finatech** Group est un intégrateur technologique de référence ayant vu le jour en 2007 et issu de la volonté du groupe O CAPITAL GROUP de se doter d’un bras technologique capable de déployer des solutions à haute valeur ajoutée auprès d’opérateurs et de donneurs d’ordres stratégiques au Maroc et en Afrique.

**Finatech** Group intervient sur un large scope d’activités à travers ses business units en interne mais également à travers ses différentes filiales.

**Finatech** Group met les enjeux et les challenges de ses clients partenaires au cœur de la réflexion qui lui permet de concevoir des solutions et des infrastructures adaptées aux besoins réels de ses clients partenaires tout en étant durables dans le temps et respectueuses de l’environnement.

**Finatech** Group déploie son expertise sur ses différents métiers à travers un Framework totalement intégré allant de la réflexion avec ses clients partenaires pour la conception de la solution, le déploiement opérationnel et la maintenance aussi bien préventive que corrective.

### **Logo de FinaTech groupe**

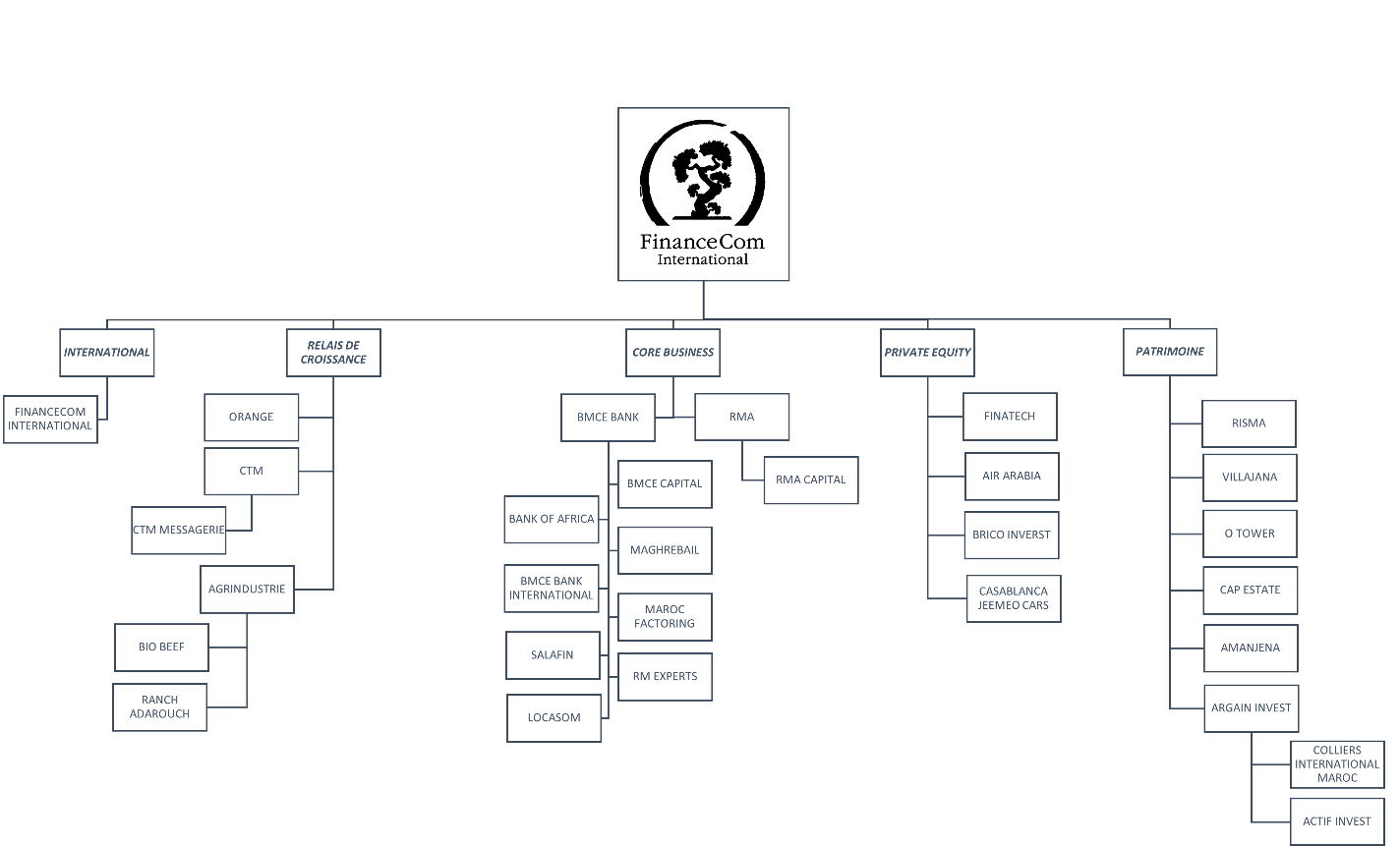


*Figure 1: Logo de l'entreprise*

1. **FINANCECOM**

Finatech Group fait partie intégrante de **O Capital Group** **(ex-Financecom)** depuis sa création en 2007. Finatech Group est le bras technologique du Groupe.

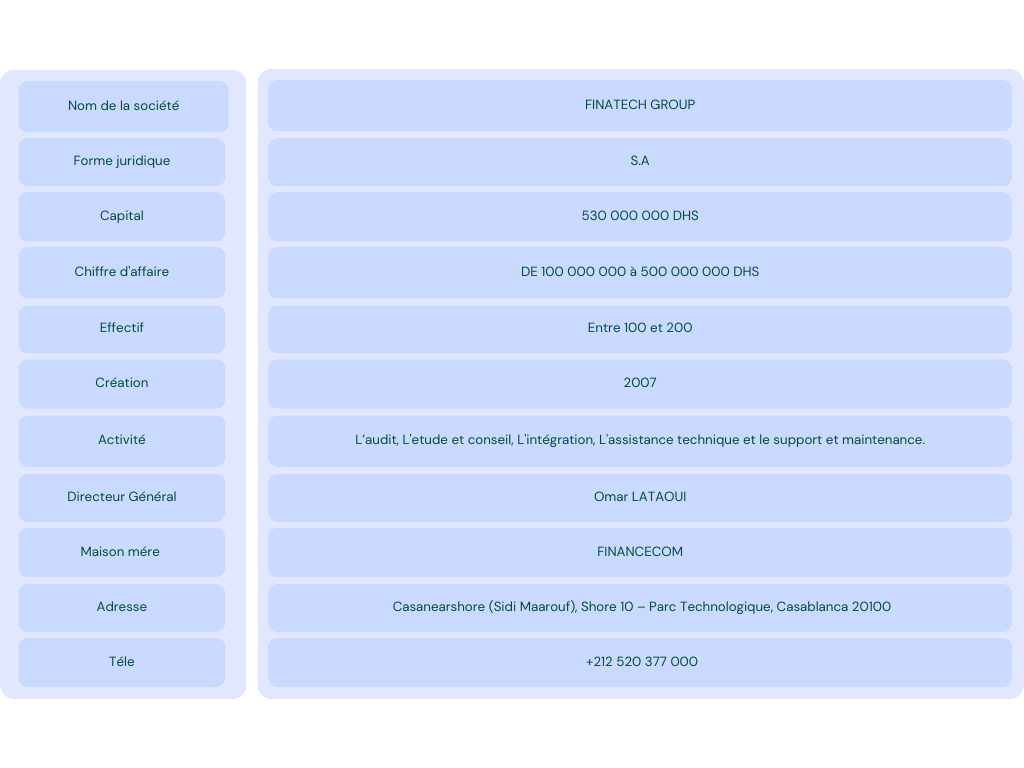
Le groupe Financecom figure parmi les plus grands consortiums économiques et financiers du Royaume Chérifien du Maroc. Le groupe est positionné sur des secteurs stratégiques clés et à fort potentiel (banques, assurances, télécoms, immobilier, hôtellerie, transports, technologie. . .) au Maroc et sur près de 23 pays africains.



*Figure 2:* *Appartenance au groupe FINANCECOM*

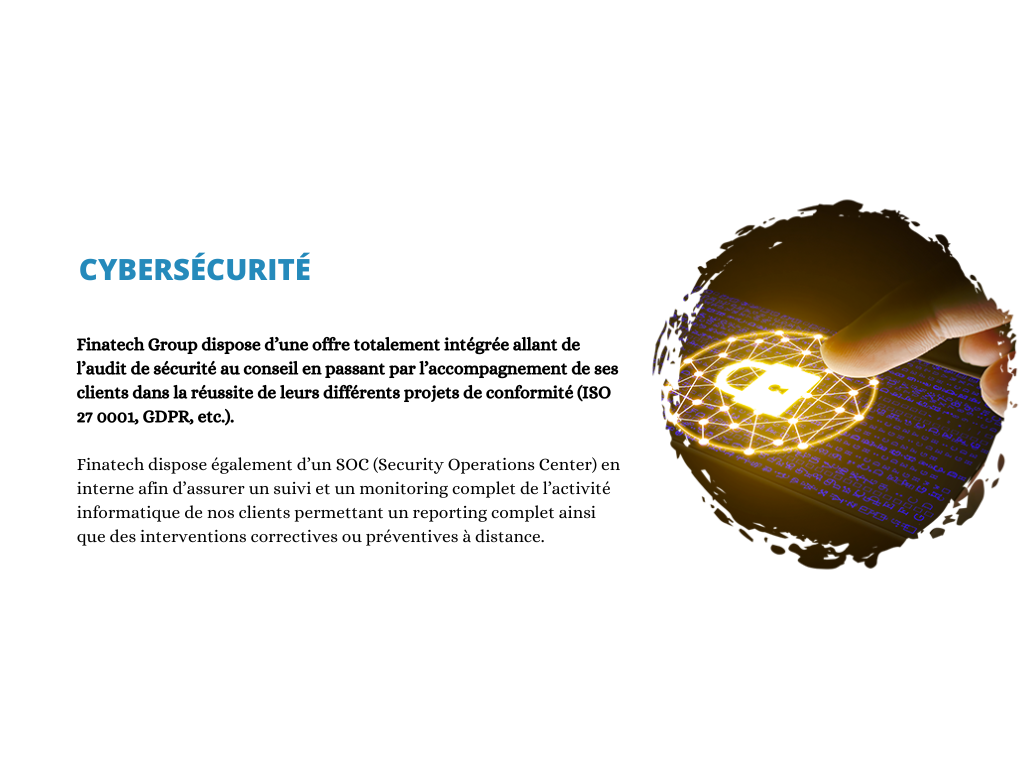
### 

### **Fiche technique de FINATECH Group**

****

*Tableau 1 : Fiche technique de FINATECH Group*

### **Activités de FinaTech groupe :**











1. **Cadre général du travail :**

Pour ce stage, l'objectif consiste à concevoir et développer une plateforme web ayant pour but d'améliorer la communication avec les fournisseurs. Cette initiative vise à faciliter les processus d'achat et de contrôle qualité, contribuant ainsi à accroître la rentabilité de l'entreprise.

### **Présentation du sujet** **:**

### **Présentation du projet :**

Ce travail est effectué dans le cadre d'un projet de stage. La plateforme doit être compatible avec tous les appareils et navigateurs web. Elle offre aux employés de l'entreprise une manière plus simple de communiquer avec les fournisseurs et de tester la qualité des produits achetés, dans le but d'améliorer l'efficacité au travail.

### **Le but à atteindre**

* + - **Faciliter** et **organiser** le travail des employés.
    - Garantir **le bon service** aux clients**.**
    - **Gagner du temps** et atteindre **la satisfaction** des clients.
    - Une application web est bien souvent **plus ergonomique** que l'utilisation des papiers, Elle rend l’expérience de travail bien meilleure grâce à la navigation.

### **Travail demandé :**

Mes responsabilités englobent :

• Contribuer au développement d'une application web dynamique.

• Établir une liaison optimisée entre la plateforme et la base de données.

• Optimiser les performances en vue d'une expérience fluide sur divers types d'appareils.

• Intégrer des fonctionnalités de synchronisation.

• Fournir une documentation exhaustive pour simplifier l'utilisation de l'application.

• Effectuer régulièrement des afin d'identifier et de résoudre d'éventuels problèmes de compatibilité.

### **Planification du projet :**

**Date de début du projet : 20/07/2023**

**Sprint 1:**

* **Objectif principal : Définir les spécifications et les exigences du projet, ainsi que valider la conception initiale.**
* Planification du sprint : 20/07/2023 - 21/07/2023
* Conception et prototypage : 22/07/2023 - 05/08/2023
* Revue interne et amélioration : 06/08/2023 - 10/08/2023
* Revue de sprint et rétrospective : 11/08/2023

**Sprint 2:**

* **Objectif principal** **:** **Finaliser les prototypes et les validations nécessaires et développement de l'application.**
* Planification du sprint : 12/08/2023 - 13/08/2023
* Développement et prototypage : 14/08/2023 - 05/09/2023
* Revue interne et amélioration : 06/09/2023 - 10/09/2023
* Revue de sprint et rétrospective : 11/09/2023

**Sprint 3:**

* **Objectif principal : tester l'application en suivant les spécifications établies.**
* Planification du sprint : 12/09/2023 - 13/09/2023
* Développement et tests : 14/09/2023 - 18/09/2023
* Tests d'acceptation utilisateur (UAT) : 19/09/2023 - 23/09/2023
* Revue de sprint et rétrospective : 24/09/2023

**Déploiement (Lancement en production) :**

* Le déploiement final de l'application est prévu pour le **25/09/2023**

marquant ainsi le lancement officiel du produit pour les utilisateurs finaux.

### **Conclusion :**

**D**ans ce chapitre, j'ai mis ce projet dans son contexte général. J'ai commencé par une description générale de la société « FinaTech ». J'ai exposé son logo, sa localisation puis ses activités et ensuite j'ai présenté le cadre général de ce projet, une brève présentation, le but à atteindre et le travail demandé en finissant avec l’environnement de développement et la planification temporelle.



# Analyse et spécifications des besoins

**Chapitre 2**

**Chapitre 2 : Analyse et spécification des besoins**

### **Introduction :**

Dans ce deuxième chapitre, j'ai commencé par la description de système de fonctionnement existant au sein de la société « FinaTech ».

L’étude réalisée soulignera quelques problèmes au niveau de la communication aux fournisseurs et donc le rendement et de satisfaction totale de clients. Cela me permet de présenter une solution totalement informatisée pour améliorer la qualité de service.

### **Étude préalable :**

Pour s'assurer que les objectifs sont atteints, il est nécessaire de commence par une vue claire et simple des différents besoins escomptés. C’est pour cela qu’une étude du système existant me permettra de justifier et de dégager une solution.

### **Étude de l’existant :**

### **Description de l’existant :**

Le problème est l'absence d'un moyen efficace de communiqué avec les fournisseurs et de contrôler l'entrepôt, créant ainsi un délai qui empêche la satisfaction des clients. Actuellement, la communication se fait par des méthodes traditionnelles telles que les mails électroniques ou le contact téléphonique direct avec le service de vents de chaque fournisseur et en passant le bon de commande (produit et quantité) et une fois la commande est bien passée, une notification est reçue contenant un formulaire (devis) qui doit être rempli et signé par l'employé, qui contient le type de produit et la quantité souhaitée.

### **Critique de l’existant :**

Il est clair qu'il est nécessaire d'assurer l'identification rapide du problème à résoudre et trouvé des solutions efficaces pour assurer le bon service de l'entreprise envers ces clients.

### **Solutions envisagées :**

Pour résoudre les problèmes dégagés et garantir le professionnalisme des services offerts Par **«** **FinaTech** **»**, je vise à :

* Concevoir et élaborer une plateforme web visant à automatiser et informatiser la communication entre les employés de l'entreprise et les fournisseurs, tout en assurant la gestion efficiente de l'entrepôt. Cette initiative vise à garantir la célérité et la sécurité des échanges de services.

### **Spécifications des Besoins**

### **Les Besoins fonctionnels :**

La future plateforme offre les fonctionnalités suivantes aux différents utilisateurs :

* Pour les administrateurs :
* Gestion des utilisateurs : Ajout, modification, suppression, activation et désactivation.
* Gestion des fournisseurs : Confirmation d'activation.
* Consultation de l'historique des produits.
* Consultation de l'historique des fournisseurs.
* Ajout, suppression et modification des réclamations.
* Consultation de l'historique des fournisseurs.
* Pour les chefs de projet :
* Gestion des projets : Ajout, modification, suppression.
* Consultation de l'historique des produits.
* Ajout de produits aux projets.
* Pour les responsables de service achats :
* Création de demandes d'achat.
* Validation des offres de vente : Consultation de l'historique, validation.
* Création de devis d'achat.
* Pour les réceptionnistes du service achats :
* Signalement de l'arrivée des produits achetés.
* Émission de rapports de réception.
* Pour les responsables de service financier :
* Création de factures.
* Pour les responsables de service qualité :
* Création de rapports de test qualité.
* Validation des règlements.

### **Les Besoins non fonctionnels :**

Les besoins non fonctionnels représentent les règles à respecter afin d’assurer une bonne qualité de l’application à achever et le bon fonctionnement du futur système. Quant aux besoins non fonctionnels, ils se récapitulent en :

### **Contraintes ergonomiques :**

L'application vise à accomplir plusieurs impératifs essentiels :

* Assurer une courbe d'apprentissage fluide, permettant aux utilisateurs de se familiariser rapidement avec le contenu et les fonctionnalités.
* Privilégier la simplicité et la clarté dans la conception de l'interface, de manière à garantir une expérience utilisateur intuitive et aisément compréhensible.
* Organiser de façon logique et intuitive les éléments interactifs tels que les boutons, les tableaux et les formes, afin d'optimiser l'ergonomie globale.

### **Contraintes techniques :**

* Hébergement : Cloud Microsoft Azure pour disponibilité et scalabilité.
  + - * Accès (ord) : Réservé aux employés autorisés avec authentification sécurisée.
      * Accès (ex-ord) : Limité à l'administrateur pour contrôle total.
      * Langage : C# avec ASP.NET pour puissance et stabilité.
      * Base de données : SQL Server pour stockage fiable des données.
      * Outil de Conception : Enterprise Architect pour architecture bien structurée.
      * Outil de gestion de version : Git / GitHub pour suivi des modifications et collaboration.
      * Outil de conteneurisation : Docker pour déploiement facile et gestion des dépendances.

**Conclusion :**

Au cours de ce chapitre, j'ai réalisé une étude au niveau du fonctionnement du système existant, afin de trouver une solution acceptable et décrire les besoins fonctionnels et non fonctionnels du nouveau système



**Chapitre 3**

# Étude Conceptuelle

**Chapitre 3 : Etude conceptuelle**

### **Introduction**

"Prévoir avant d'agir, établir des plans avant de construire, conceptualiser en amont et développer en aval : telle est la démarche essentielle à suivre lors de l'élaboration d'une application, ainsi que pour la réussite de tout projet.

En effet, la phase de conception d'un système informatique revêt une importance capitale, exerçant une influence directe sur la qualité et la fiabilité de l'application dans son ensemble.

Initiant ce chapitre, j'ai introduit le modèle de cycle de vie du projet sélectionné, offrant ainsi un cadre méthodologique. Par la suite, j'ai abordé en détail la phase de conception, au cours de laquelle j'ai présenté l'architecture globale de l'application avec précision. Enfin, j'ai mis en lumière les différents diagrammes, à savoir ceux des cas d'utilisation et des séquences, venant compléter cette étape de conception."

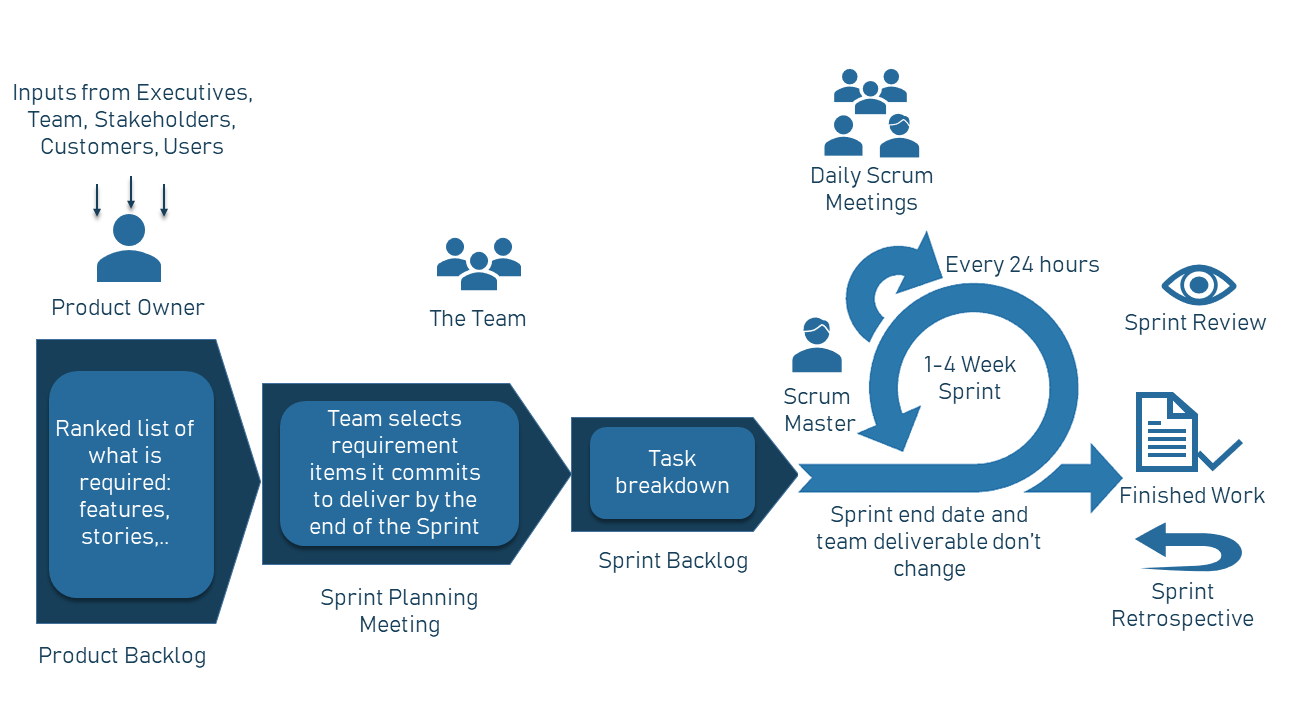
### **Cycle de vie de développement de projet :**

Le cycle de vie d'un logiciel, souvent désigné sous le terme anglais de "software life cycle", englobe toutes les étapes du développement d'une application, depuis sa conception jusqu'à son éventuelle désuétude. Cette structuration vise à établir des étapes intermédiaires pour confirmer le développement logiciel, c'est-à-dire garantir l'alignement du logiciel avec les besoins exprimés, ainsi que pour vérifier le processus de développement en assurant la cohérence des méthodologies adoptées. Dans le but de parvenir à une réalisation optimale, j'ai présenté la méthodologie agile, et plus spécifiquement la méthodologie Scrum, une approche qui a vu le jour dans les années 1990.

Dans le cadre de cette approche, chaque étape est conçue pour être flexible et adaptative. Au lieu d'un découpage rigide en phases, Scrum fonctionne par itérations appelées "sprints". Chaque sprint, d'une durée généralement courte, vise à produire une version fonctionnelle du logiciel. L'approche Scrum favorise une collaboration rapprochée entre les membres de l'équipe, encourageant une communication ouverte et constante.

Au cœur de Scrum se trouve le "Product Backlog", une liste dynamique d'éléments à réaliser qui évolue en fonction des retours des utilisateurs et des changements de priorités. Au début de chaque sprint, l'équipe sélectionne les éléments du backlog à traiter, créant ainsi le "Sprint Backlog" qui guide les efforts du sprint. Les rôles clés de Scrum, tels que le "Scrum Master" et le "Product Owner", travaillent en synergie pour garantir la réussite de chaque sprint et l'atteinte des objectifs.

À la fin de chaque sprint, un "Sprint Review" est organisé pour démontrer le travail accompli aux parties prenantes. Cela est suivi par une "Sprint Retrospective" qui permet à l'équipe d'analyser le sprint passé et d'identifier des améliorations pour les itérations futures.

L'approche Scrum dans le cadre agile offre une souplesse exceptionnelle pour s'adapter aux évolutions et aux exigences changeantes du projet. Elle favorise la transparence, l'engagement de l'équipe et la livraison continue de fonctionnalités, ce qui en fait une méthodologie pertinente pour les environnements complexes et dynamiques de développement logici

*Figure 4: Agile - Scrum*

### **Langage UML :**

### **Présentation du langage UML :**

**UML** (en anglais Unified Modeling Language, « langage de modélisation unifié ») est un langage graphique de modélisation des données et des traitements. C’est une formalisation non- propriétaire de la modélisation objet utilisée en génie logiciel. **UML** spécifie plusieurs objectifs qui font un outil exact de communication :

* Comprendre et décrire les besoins.
* Spécifier un système.
* Établir l’architecture logicielle.

### **Intérêt de la modélisation :**

L’utilisation de la modélisation conceptuelle dans le développement des systèmes d’information permet de prendre en compte les besoins des applications d’une façon plus adéquats et de présenter d’une manière abstraite certains aspects des systèmes physiques et humains.

### **Les avantages d’UML :**

* UML est un langage formel et standardisé.
  + Gain de précision.
  + Motivation à l’utilisation d’outils.
  + Gagne de stabilité et de fixité.

UML normalises les concepts objet, sa notion graphique permet d’exprimer une solution objet, ce qui simplifie la comparaison et l’appréciation des solutions. UML cadre l’analyse objet, il permet non seulement de représenter les concepts objets, mais il sous-entend une démarche d’analyse qui permet de reproduire une solution objet de manière itérative, grâce aux diagrammes, qui supportent l’abstraction.

Un diagramme UML est une représentation graphique, et à chaque vue correspondent des diagrammes qui sont répartis selon leurs aspects statiques ou dynamiques :

* ***Statique :*** 
  + Cas d’utilisation
  + Classes
  + Composants
  + Objets
  + Déploiement

* ***Dynamique :*** 
  + Séquences
  + Activité
  + État-transition
  + Collaboration
* ***Fonctionnel :*** 
  + Cas d’utilisation
  + Collaboration
* Ces diagrammes ne sont pas nécessairement tous produits à l’occasion d’une modélisation. Les plus utilisables sont les diagrammes de classes, de cas d’utilisation, et de séquence.

### **Conception avec UML :**

### **Outil de modélisation :**

**PowerAMC** anciennement connu sous le nom de "PowerDesigner", est un outil de modélisation visuelle et de conception utilisé dans le développement de logiciels et la gestion de bases de données. Il est développé par SAP (Systems, Applications, and Products in Data Processing) et se concentre principalement sur l'aide aux entreprises pour concevoir, modéliser et gérer leurs bases de données, leurs processus métier et leur architecture logicielle.

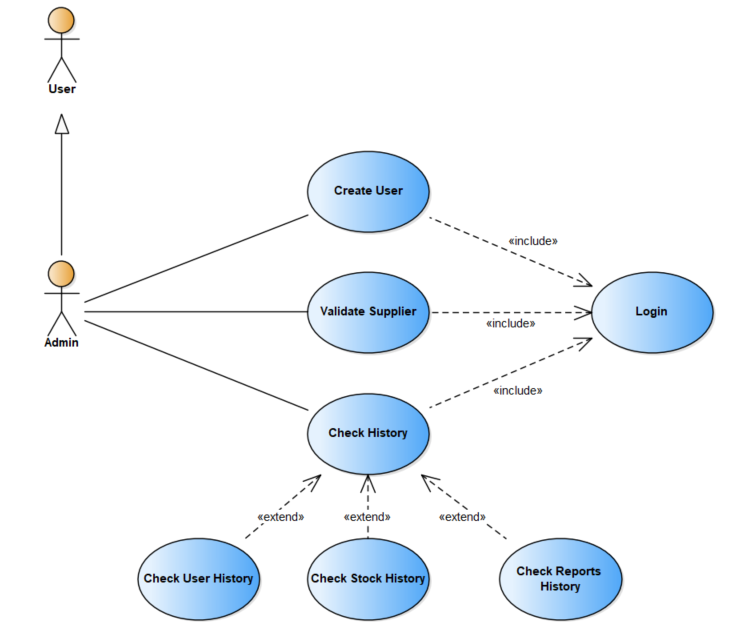
### **Modélisation avec les diagrammes cas d’utilisation :**

Le diagramme de cas d’utilisation permet de déterminer les possibilités d'interférence entre le système et les acteurs, c'est-à-dire déterminer toutes les fonctionnalités que doit fournir le système. Il permet aussi de délimiter ce dernier.

* + - Chaque usage effectué par les acteurs est représenté par un cas d'utilisation.
    - Chaque cas d'utilisation symbolise une fonctionnalité qui leur est offerte afin d’engendrer le résultat attendu.
    - Le diagramme de cas d'utilisation décrit l'interaction entre le système et l'acteur en déterminant les besoins de l'utilisateur et tout ce que doit faire le système pour l'acteur.

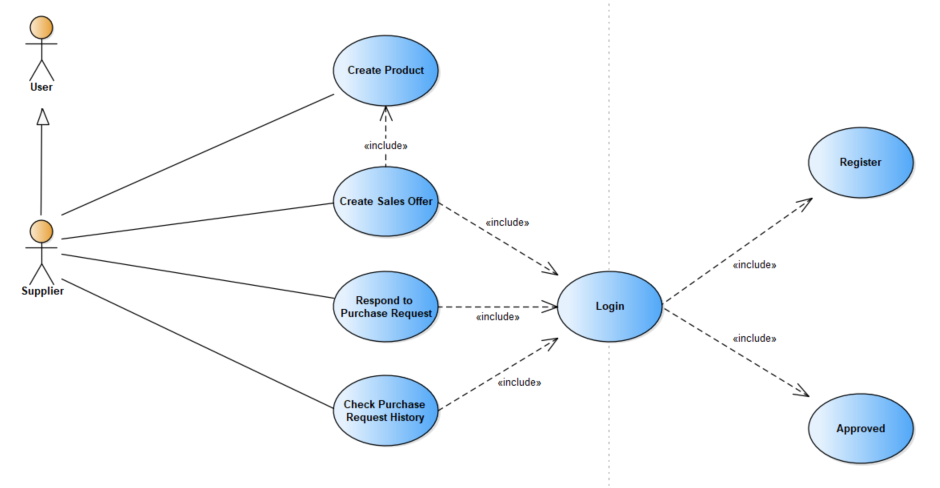
### **Diagrammes de cas d’utilisation :**

* **Admin**



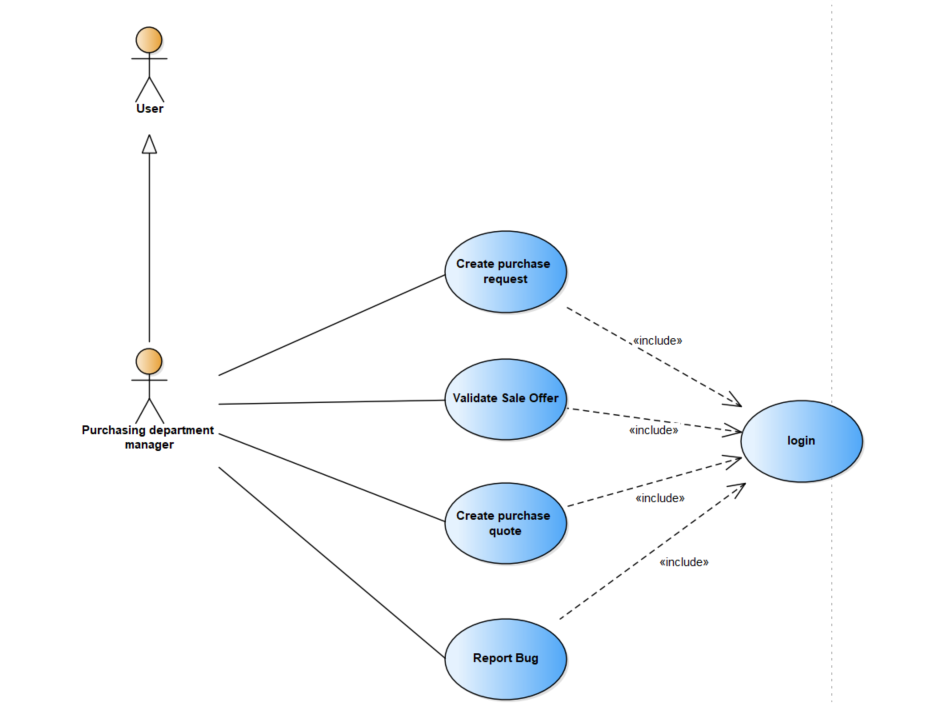
*Figure 5: Le diagramme de cas d'utilisation (Admin)*

* **Supplier**



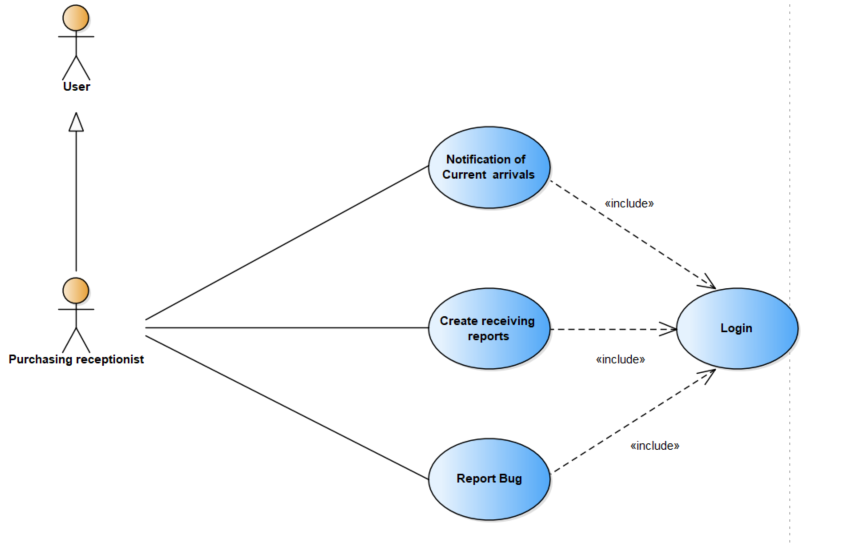
*Figure 6: Le diagramme de cas d'utilisation (Supplier)*

* **User : (****Responsable Service achat)**



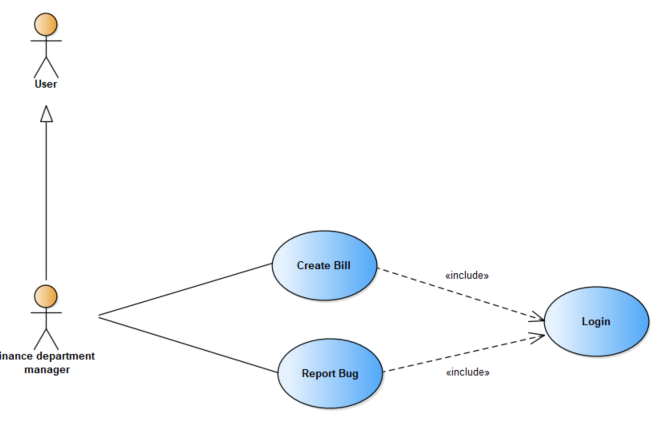
*Figure 7: Le diagramme de cas d'utilisation (Responsable Service achat)*

* **User : (****Réceptionniste Service achat)**



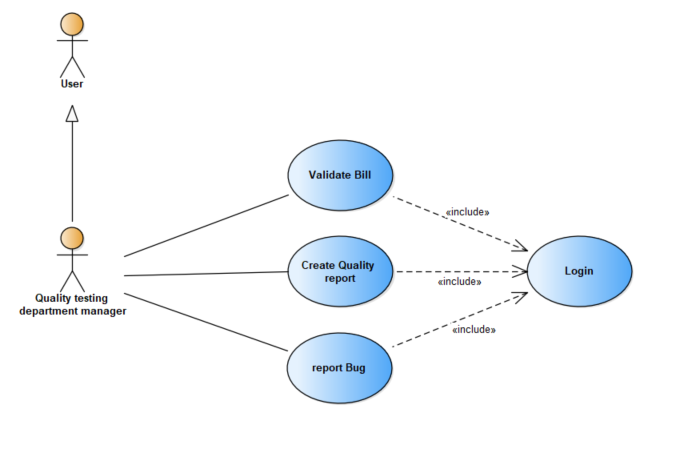
*Figure 8: Le diagramme de cas d'utilisation (Réceptionniste Service achat)*

* **User : (Responsable Service Financière)**



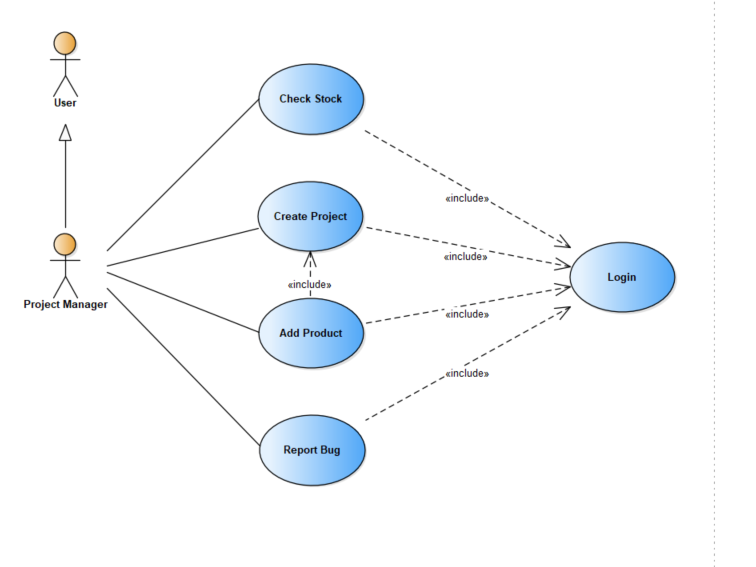
*Figure 9: Le diagramme de cas d'utilisation (Réceptionniste Service achat)*

* **User : (****Responsable Service qualité)**



*Figure 10: Le diagramme de cas d'utilisation (Responsable Service qualité)*

* **User : (****Chef de projet)**

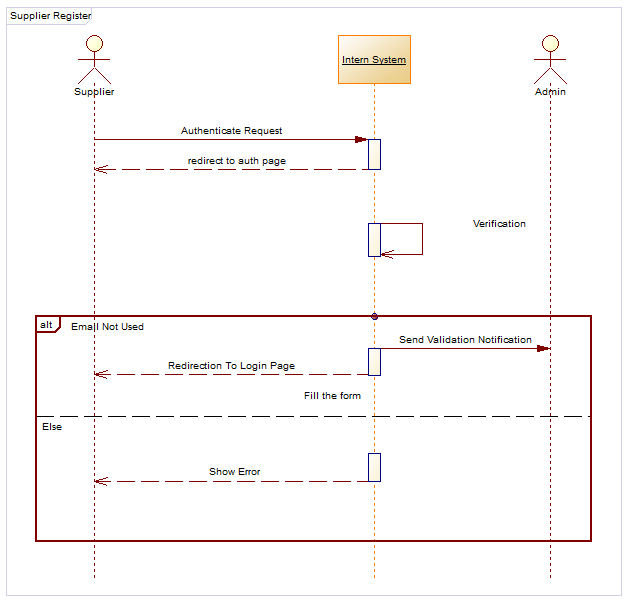


*Figure 11: Le diagramme de cas d'utilisation (Chef de projet)*

### **Modélisation avec les diagrammes de séquences :**

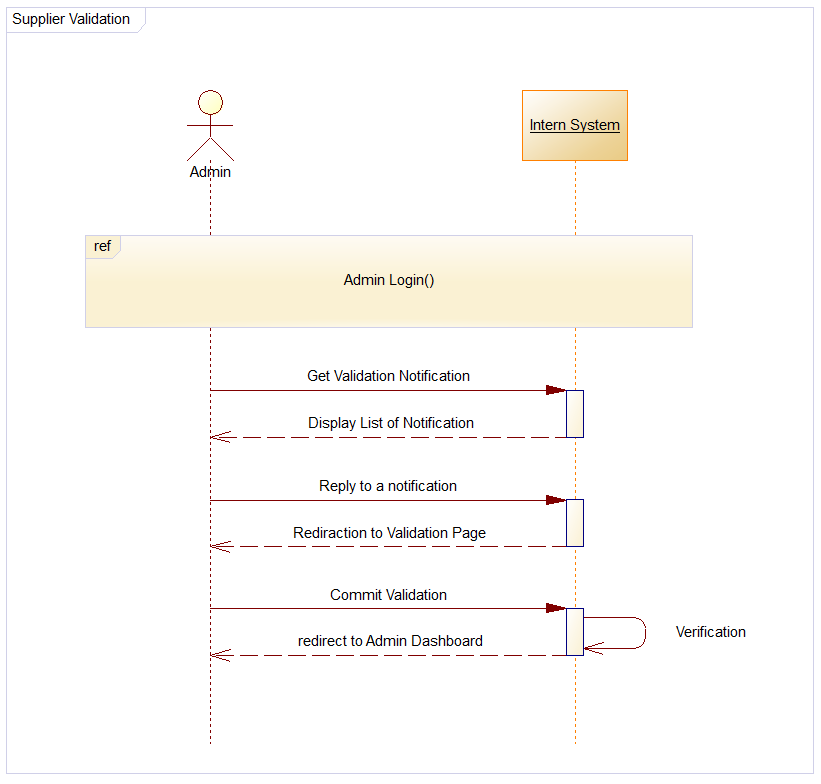
1. **Diagrammes de séquences**

* **Fournisseur Inscription**



*Figure 6: Le diagramme de séquences « Fournisseur Inscription »*

* **Fournisseur Validation**



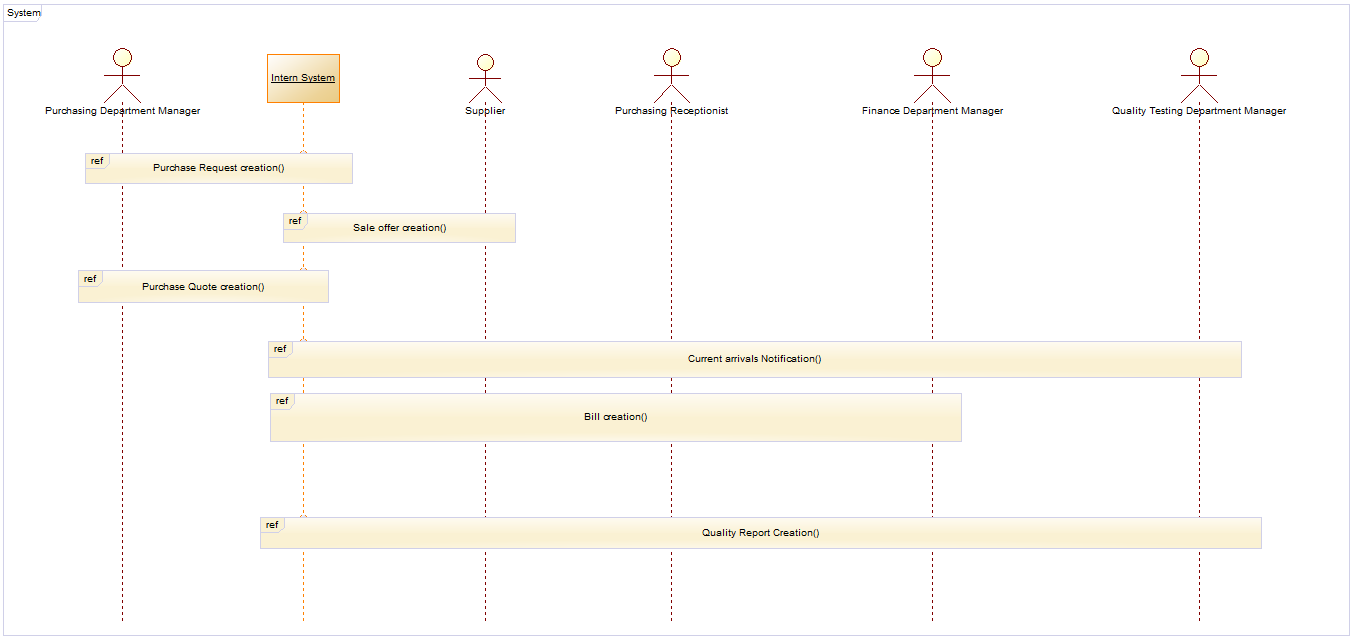
*Figure 6: Le diagramme de séquences « Fournisseur Validation »*

* **Fournisseur Login**



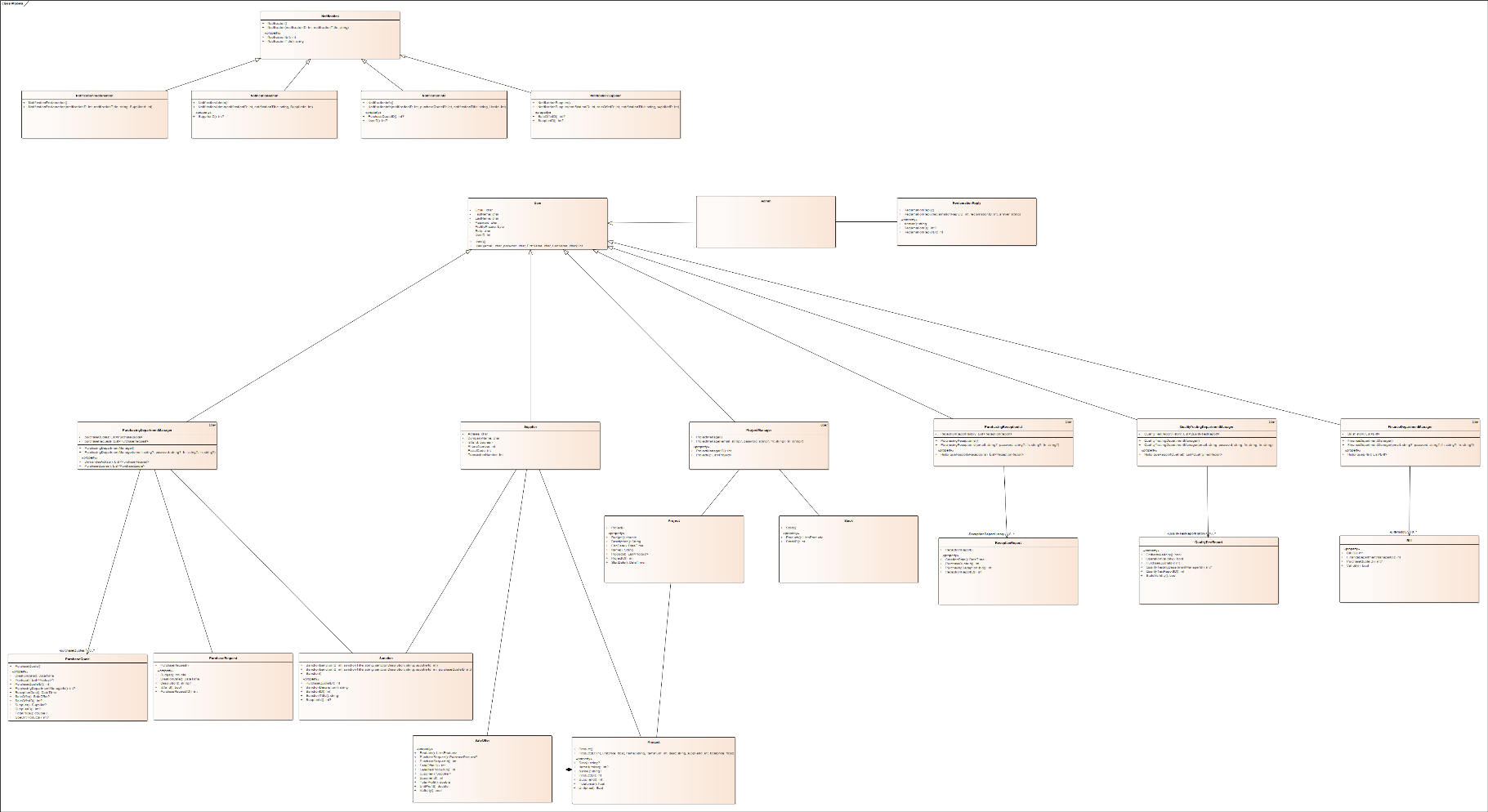
*Figure 7: Le diagramme de séquences « Fournisseur Login »*

* **Diagramme en entier**



*Figure 8: Le diagramme de séquences «Diagramme en entier»*

### **Modélisation avec les diagrammes de Classes :**



### 

*Figure 6: Le diagramme de class « System»*

### **Code MAP :**

### 

*Figure 6: Le code MAP des Modèles*

### Conclusion :

Ce chapitre a été consacré à la modélisation UML du système. J'ai présenté différents diagrammes le diagramme de cas d’utilisation, et de séquences afin de spécifier de façon détaillée les aspects fonctionnels, dynamiques et statiques du système. Dans le chapitre suivant, j'ai présenté la réalisation et la mise à l’essai de ce système.



**Chapitre 4**

# La Réalisation

Chapitre 4 : La réalisation

### **Introduction :**

Je suis maintenant parvenu à la phase ultime de mon projet. Cette dernière étape revêt une importance capitale, car elle concrétise l'ensemble des théories exposées précédemment. Pour commencer, je vais présenter l'environnement logiciel de production. Ensuite, je mettrai en lumière plusieurs interfaces clés de mon application Android, ainsi que quelques scripts que j'ai identifiés comme étant particulièrement cruciaux.

### **Environnement de travail :**

Au sein de cette section, je mettrai en évidence les divers outils matériels et logiciels indispensables au processus de développement de mon application.

### **Environnement matériel :**

Je mentionnerai les caractéristiques de mes ordinateurs sur lesquels j'ai développé l'application parce qu’elles peuvent donner une idée sur les conditions du travail.

L'application a donc été développée sur deux ordinateurs portables dont les caractéristiques principales sont :

* ***1er ordinateur*** 
  + **Nom :** HP
  + **Processor :** Intel® Core™ i9-10210U (1.6 GHz base frequency, up to 4.2 GHz
  + **Mémoire installée (RAM) :** 12,00Go (7.89GO utilisable)
  + **Type de système : :** Système d’exploitation 64bits
* ***2èm ordinateur*** 
  + **Nom :** Asus ROG Strix G15
  + **Processor :** MD Ryzen™ 9 6900HX Mobile Processor (8-core/16-thread, 20MB cache
  + **Mémoire installée (RAM) :** 16,00GB DDR4-4800 SO-DIMM
  + **Type de système : :** Système d’exploitation 64bits

### **Environnement logiciel :**

J'ai énuméré au cours de cette partie les différents outils utilisés tout au long de ce projet pour l’étude et la mise en place de notre application.

### **Développement :**

* 1. **Système d’exploitation :**

J'ai utilisé comme système d’exploitation :

* ***1ère ordinateur :*** Microsoft Windows 11 Professionnel
* ***2ème ordinateur :*** Microsoft Windows 10 Professionnel

### **Outils de développement :**

* + - **ASP (Active Server Pages) :**

ASP, acronyme d'Active Server Pages, constitue une technologie initiée par Microsoft pour concevoir des pages Web à caractère dynamique en exploitant des scripts exécutés côté serveur. En utilisant ASP, les développeurs étaient en mesure d'intégrer des codes de script (généralement en VBScript ou JScript) au sein de pages HTML, en vue de générer du contenu dynamique. Néanmoins, les limites d'ASP en matière de scalabilité, de maintenance et de distinction des préoccupations se sont avérées manifestes.



*Figure 10: ASP logo*

* + - **ASP.NET :**

ASP.NET représente l'évolution d'ASP, arborant un cadre de développement web plus sophistiqué et exhaustif. Cette avancée s'accompagne de diverses améliorations, dont une approche davantage structurée pour la création d'applications web, une meilleure séparation entre le code et la présentation, ainsi qu'un modèle de programmation orienté objet. Parmi les caractéristiques majeures d'ASP.NET, citons l'introduction de Web Forms, qui proposait une interface de glisser-déposer pour édifier des pages web, et une prise en charge élargie pour différents langages de programmation, dont le C# et le VB.NET. En outre, ASP.NET a également instauré un modèle puissant de programmation basé sur les événements, associé à une meilleure gestion des états de l'application.



*Figure 10: ASP.Net logo*

* + - **ASP.NET Core MVC :**

ASP.NET Core MVC représente la dernière avancée du cadre de développement Web de Microsoft, bâti sur la base de la plateforme ASP.NET Core. Voici les distinctions principales en comparaison avec les versions antérieures :

* **Compatibilité Multiplateforme** : ASP.NET Core MVC est polyvalent, fonctionnant aussi bien sur Windows, Linux que macOS. Cette flexibilité le rend adaptable à une variété d'environnements d'hébergement.
* **Open-Source :** ASP.NET Core est open-source et connaît un développement actif sur la plateforme GitHub. Ceci favorise les contributions de la communauté et la transparence dans le processus de développement.
* **Architecture Modulaire** : ASP.NET Core permet aux développeurs d'intégrer uniquement les composants nécessaires, minimisant ainsi les surcharges et rendant les applications plus légères.
* **Performance :** ASP.NET Core est conçu en ayant en tête les performances. Il s'avère plus rapide et plus efficient par rapport aux itérations précédentes d'ASP.NET.
* **Injection de Dépendances :** ASP.NET Core facilite l'injection de dépendances, simplifiant ainsi la gestion et le test des relations de dépendance au sein d'une application.
* **Framework Unifié MVC et Web API :** ASP.NET Core MVC fusionne les frameworks MVC et Web API qui étaient auparavant séparés dans ASP.NET. Cela rationalise le développement et facilite la création d'applications Web et d'API au sein d'un même cadre.

****

*Figure 10: ASP.Net Core MVC logo*

### **Environnement de développement intégré :**

* + - **Visual Studio**

**Visual Studio (VS)** est un environnement de développement intégré (IDE) puissant créé par Microsoft pour la conception d'une large gamme d'applications logicielles. Avec une interface conviviale, il propose des outils complets pour programmer, déboguer et déployer des applications sur diverses plateformes. Les développeurs peuvent coder en langages tels que C#, C++, Python, tout en bénéficiant de fonctionnalités comme l'autocomplétion, un débogage avancé et un contrôle de version intégré. VS prend en charge le développement web, desktop, mobile, cloud et de jeux, et peut être étendu via des plugins, le rendant essentiel pour les développeurs à la recherche d'un environnement riche en fonctionnalités et productif.



*Figure 10: Visual Studio (VS) logo*

### **Outil d’administration de la base de données**

* + - **SQL Server**

SQL Server, conçu par Microsoft, représente un système de gestion de base de données relationnelle (SGBDR) essentiel pour la gestion, le stockage et la restitution de données structurées. Cette solution offre une plateforme solide et évolutive, idéale pour le traitement efficace des données au sein des entreprises et des applications. SQL Server propose une gamme d'éditions adaptées à une variété de besoins, allant des petites entreprises aux grandes entreprises.

* Pour le système de gestion de base de données SQL Server :
* **SQL Server Express :** SQL Server Express est une édition légère et gratuite du système de gestion de base de données SQL Server de Microsoft. Conçue pour des applications à plus petite échelle, elle convient au développement, aux tests ou à des charges de travail de production limitées. SQL Server Express inclut un sous-ensemble des fonctionnalités disponibles dans la version complète de SQL Server.
* **SQL Server Management Studio Express (SSMSE) :** SQL Server Management Studio Express est un outil d'interface utilisateur graphique utilisé pour gérer les bases de données SQL Server Express. Il permet aux utilisateurs de créer, de modifier et de gérer des bases de données, ainsi que d'effectuer des tâches telles que la rédaction de requêtes, la conception de schémas de base de données et la gestion de la sécurité.





*Figure 13:SQL Server Express Logo Figure 14: SQL Server Log*

### **Outil de conteneurisation :**

### **Docker :**

### Est un outil de conteneurisation, révolutionne le déploiement d'applications. Il encapsule applications et dépendances dans des conteneurs isolés. Cette approche résout les problèmes de compatibilité, assurant une exécution cohérente. Les développeurs créent des images conteneurisées contenant tout le nécessaire, accélérant le déploiement. Docker facilite la gestion et le passage en production, tout en offrant une mise à l'échelle souple. Cette méthode modulaire améliore l'efficacité, la portabilité et la cohérence des déploiements.

### J'ai utilisé Docker pour installer et gérer de manière transparente une base de données SQL Server au sein d'un environnement conteneurisé. En exploitant les capacités de conteneurisation de Docker, j'ai encapsulé l'ensemble de l'instance SQL Server ainsi que sa configuration, ses données et ses dépendances dans un conteneur autonome. Cette approche a assuré un déploiement cohérent indépendamment de l'environnement hôte

### 

*Figure 10: Docker logo*

### **Outil de contrôle  de Version:**

### **Git / GitHub :**

### L'outil de contrôle de version **Git** associé à la plateforme **GitHub** représente un pilier essentiel pour la gestion collaborative et la traçabilité du développement logiciel. Git offre un mécanisme sophistiqué de suivi des modifications apportées au code source, permettant aux développeurs de travailler simultanément sur des branches distinctes tout en fusionnant leurs contributions de manière fluide. GitHub agit comme une plateforme centralisée pour héberger, partager et collaborer sur les projets Git. Cela permet aux équipes de suivre l'évolution des codes, d'effectuer des révisions et des commentaires, et de gérer les problèmes en toute transparence. L'association de Git et GitHub favorise un flux de travail agile, une gestion de projet efficace et une historique détaillé des modifications, faisant de cette combinaison un élément essentiel pour tout processus de développement moderne.

### 

### 

*Figure 13:Git Logo Figure 14: GitHub Logo*

### **Langages de programmation :**

* + - **C# :**

C# est un langage de programmation moderne, polyvalent et orienté objet développé par Microsoft. Il est largement utilisé pour créer une gamme diversifiée d'applications logicielles, des applications de bureau aux applications web et mobiles. Réputé pour sa simplicité et sa lisibilité, C# intègre des fonctionnalités de différents langages de programmation, le rendant à la fois intuitif pour les débutants et puissant pour les développeurs expérimentés. Doté d'une forte emphase sur la sécurité des types et la gestion de la mémoire, C# garantit un code solide et fiable. Il fait partie intégrante du framework .NET, offrant une intégration transparente avec d'autres technologies Microsoft.



*Figure 15 : C# Logo*

### **Les interfaces graphiques :**

### **Interfaces :**

L'interface graphique revêt une importance primordiale dans le processus de conception d'une application, car elle procure une expérience agréable à l'utilisateur tout au long de sa navigation. De ce fait, ce critère a le potentiel de distinguer deux applications, même si elles partagent des fonctionnalités similaires. Voici une série de captures d'écran illustrant les points d'entrée clés de l'application :

### **Interface Authentification /** **Inscription :**



*Figure 17 : Interface Authentification*

### **Interfaces Inscription :**



*Figure 17 : Interface Inscription*

La première figure dépeint l'interface d'authentification, au sein de laquelle l'utilisateur est invité à saisir son adresse e-mail ainsi que son mot de passe afin d'accéder à l'espace d'accueil de l'application. En ce qui concerne la seconde figure, elle représente l'interface d'inscription destinée aux fournisseurs.

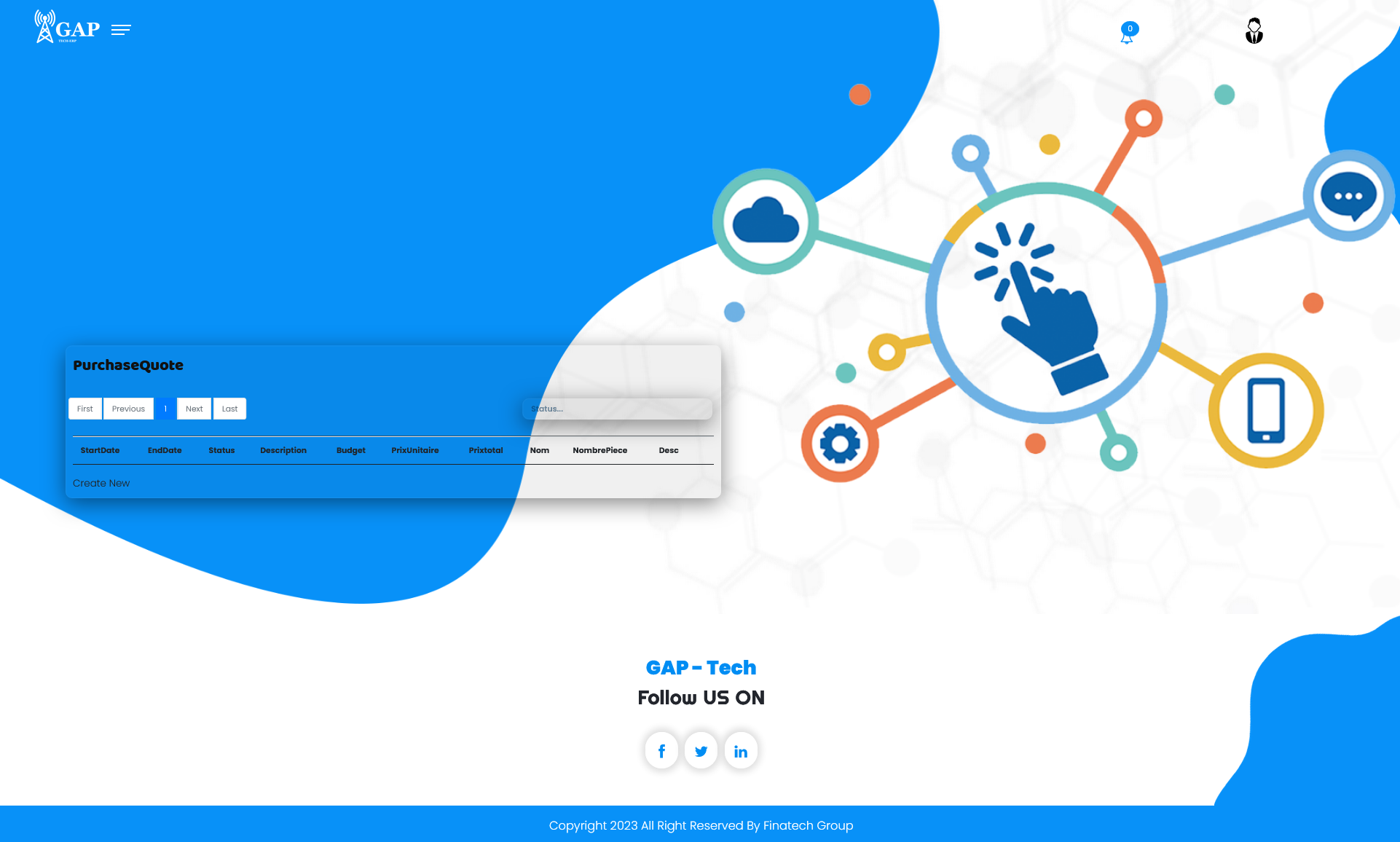
### **Interfaces Accueil :**

* Page accueil Administrateur :



*Figure 20 : Interface Accueil Admin*

* Page accueil Utilisateur

**

*Figure 20 : Interface Accueil Utilisateur*

Une fois que l'utilisateur a saisi son nom d'utilisateur et son mot de passe, l'interface d'accueil s'affiche. Si l'utilisateur est un administrateur, la page du « tableau de bord administrateur » s'ouvrira. Cette page offre une vue d'ensemble du contenu de l'application, des journaux de stock et propose des interfaces pour créer différents types d'utilisateurs. En revanche, si l'utilisateur n'est pas un administrateur, l'interface d'accueil spécifique à l'utilisateur sera présentée. Chaque type d'utilisateur bénéficie d'un contenu unique en fonction de ses responsabilités respectives.

### **Déploiement:**

J'ai réussi à déployer avec succès mon application ASP.NET Core MVC sur Microsoft Azure, en exploitant son infrastructure cloud pour un hébergement et une mise à l'échelle harmonieux. Grâce aux outils de déploiement intuitifs d'Azure, j'ai configuré et publié efficacement mon application, assurant ainsi sa disponibilité à un public plus large. De plus, j'ai intégré le processus de déploiement aux services de base de données gérés d'Azure, en utilisant leur solution de base de données offerte pour stocker et gérer les données de l'application. Cette intégration a non seulement rationalisé le déploiement, mais a également garanti un stockage de données fiable et sécurisé. En tirant parti des capacités d'Azure, j'ai atteint une solution globale de bout en bout, avec mon application ASP.NET Core MVC hébergée et prise en charge par les services cloud d'Azure, et la base de données associée gérée sans effort via leurs services de base de données fournis.

* Voici quelques-unes des étapes que j'ai suivies pour réussir le déploiement :
* **Création de ressources Azure :** Après m'être connecté au portail Azure, j'ai mis en place les ressources nécessaires pour héberger mon application. J'ai configuré un service d'application pour accueillir mon application ASP.NET Core MVC et j'ai sélectionné la pile d'exécution appropriée.
* **Déploiement de l'application :** J'ai utilisé les outils de déploiement d'Azure, tels qu'Azure DevOps ou les fonctionnalités de publication Azure de Visual Studio, pour déployer mon application ASP.NET Core MVC sur le service d'application Azure. J'ai pris soin de configurer correctement l'application, notamment les chaînes de connexion à la base de données.
* **Configuration du service de base de données :** J'ai exploré les services de base de données gérés d'Azure, en particulier Azure SQL Database. J'ai créé une instance de base de données via le portail Azure, ajusté ses paramètres et noté les détails de connexion.
* **Migration de la base de données :** Si vous avez des données existantes, considérez l'utilisation d'outils ou de scripts de migration de base de données pour transférer vos données vers la base de données Azure. Assurez-vous que votre application ASP.NET Core est configurée pour se connecter à la nouvelle base de données Azure.
* **Mise à jour de la chaîne de connexion :** J'ai mis à jour les chaînes de connexion de mon application ASP.NET Core MVC pour qu'elles pointent vers la nouvelle base de données Azure que j'avais créée. Cela garantissait que mon application communiquait avec la bonne instance de base de données.
* **Tests et vérifications :** J'ai minutieusement testé mon application sur la plateforme Azure, en vérifiant son interaction avec la base de données Azure. J'ai validé que la récupération, le stockage et la manipulation des données se déroulaient comme prévu.
* **Sécurité et autorisations :** J'ai pris des mesures de sécurité nécessaires, en établissant des règles de pare-feu pour la base de données Azure et en configurant les autorisations d'accès appropriées pour garantir la sécurité des données.





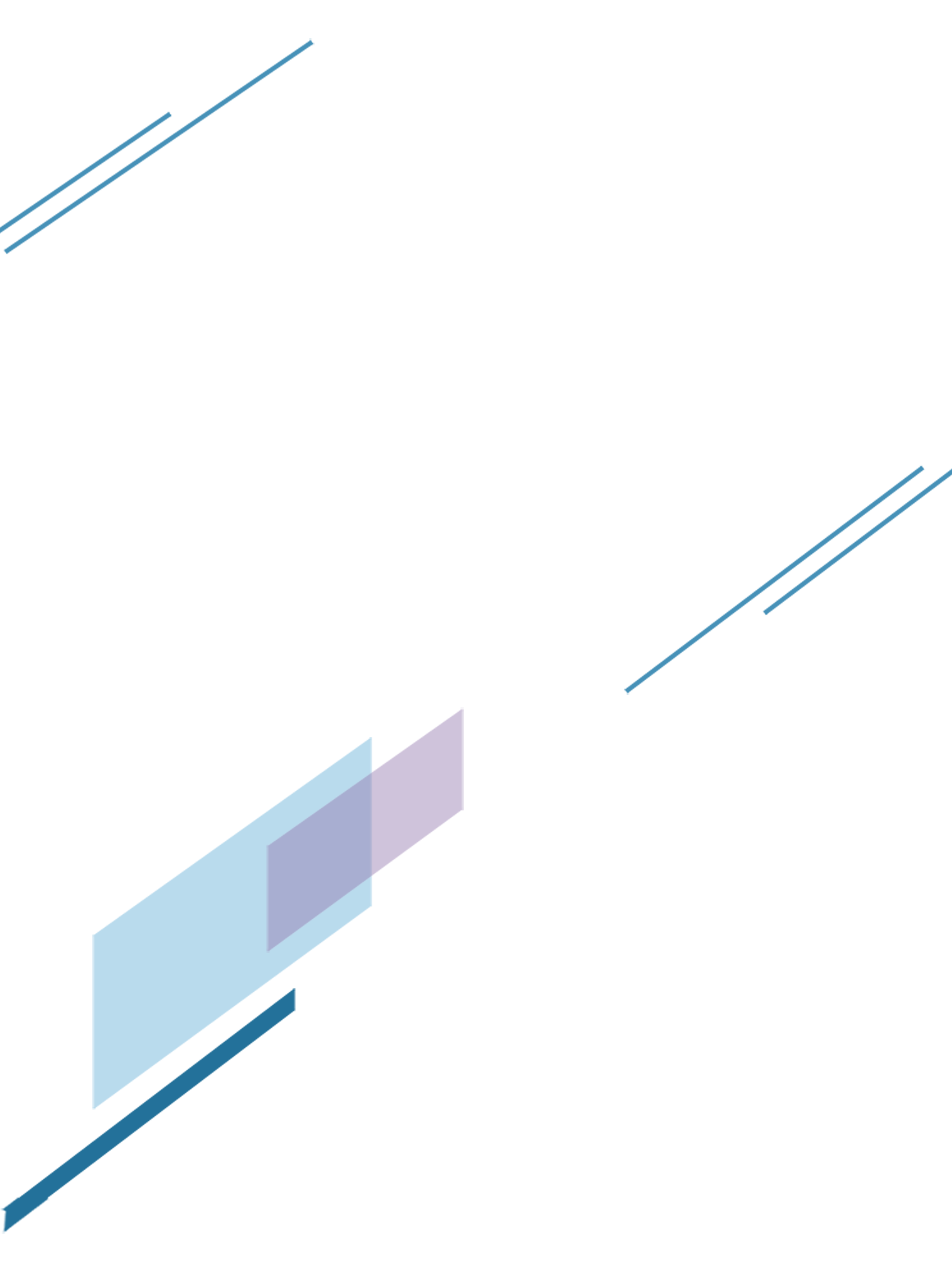
*Figure 31: Microsoft Azure SQL Database Figure 31: Microsoft Azure Portal*

Conclusion :

La dernière partie était dédiée à la navigation dans cette application. Elle constitue le dernier volet de ce rapport, elle a pour objet de présenter l’environnement logiciel et matériel de réalisation, et de décrire les principales fonctionnalités implémentées suite au développement par des captures écrans présentant les principes interfaces des applications.



Contribuer à la création de cette plateforme a été une opportunité exceptionnelle qui m'a permis d'acquérir une expérience professionnelle concrète dans le domaine du développement d'applications Web en utilisant ASP.NET. Cette expérience s'est révélée profondément enrichissante à de multiples niveaux. En mettant en pratique mes connaissances théoriques, ce projet m'a non seulement permis de les consolider, mais également de les ancrer dans un contexte réel et concret. J'ai ainsi pu approfondir mes compétences en langages de programmation et en plateformes, que j'ai explorés avec succès et exploités de manière fructueuse.

Ce rapport détaille de manière exhaustive les différentes étapes indispensables à la conception et au développement de cette plateforme novatrice. De plus, cette initiative a constitué une opportunité inestimable pour acquérir de nouvelles compétences, notamment dans l'utilisation experte de SQL Server, Visual Studio, Docker et Git. En maîtrisant ces outils, j'ai consolidé davantage mes compétences en langages de programmation et en technologies associées, notamment C# et ASP.NET Core MVC.

Il est indiscutable que ces compétences nouvellement acquises joueront un rôle crucial dans mes expériences professionnelles à venir, en me conférant un avantage indéniable sur le marché du travail.

###### En rétrospective, ce stage a été une expérience professionnelle véritablement exceptionnelle. La vie universitaire offre certes un aperçu de notre future carrière, mais il devient rapidement évident que la pratique dépasse souvent les connaissances théoriques, offrant ainsi une formation accélérée et concrète.