République **A**lgérienne **D**émocratique et **P**opulaire
Ministère de l'**E**nseignement **S**upérieur et de la **R**echerche **S**cientifique
UNIVERSITE **M**OULOUD **M**AMMERI DE **T**IZI-**O**UZOU

FACULTE DE **G**ENIE **E**LECTRIQUE ET D'**I**NFORMATIQUE
DEPARTEMENT D'**I**NFORMATIQUE

Mémoire de Fin d'Études
de MASTER ACADEMIQUE
Domaine **: Mathématiques et Informatique**
Filière **: Informatique**
Spécialité **: Systèmes informatiques**

Thème : Administration et sécurité d'une plate-forme ERP

Réalisé par : NAIT DJOUDI Anya Amel

Sous la direction du professeur : M.SI-MOHAMMED

Année universitaire : 2019/2020

Remerciements

En premier lieu,je tiens à remercier mon directeur de mémoire, le professeur M.SI-MOHAMMED, pour l'encadrement qu'il a assuré tout au long de ce travail, ses remarques pertinentes ,sa patience et la disponibilié dont il a toujours fait preuve.

Je tiens à remercier également les membres du jury d'avoir accepté d'évaluer ce travail.

Enfin, un grand merci à mes chers parents qui n'ont eu cesse à m'encourager et m'assurer leur présence et affection durant l'élaboration de ce mémoire.

À tous , merci !

Anya.N

Dédicaces

À mes chers parents,
À mon cher frère ,
À ma famille,
À mes amis(es).

Anya.N

Résumé

Une plate-forme ERP permet d'intégrer toutes les facettes de l'entreprise sous une suite d'applications logicielles.Notre travail consiste à administrer et sécuriser cette dernière.
Pour ce faire, nous avons implémenté une application web en utilisant le framework Laravel. Notre application permet la gestion des utilisateurs, des rôles et des permissions selon un modèle de contrôle d'accès basé sur les rôles (RBAC).Elle comprend également une mini application de gestion de la relation client (CRM) à travers laquelle nous illustrons la capacité de l'administrateur à accorder les autorisations nécessaires aux utilisateurs sur des ressources spécifiques.

Mots clés : Administration , sécurité, contrôle d'accès,CRM, ERP, RBAC, Laravel.

Abstract

An Enterprise Resource Planning (ERP) allows you to integrate all facets of your business into a suite of software applications.Through this thesis we aim to provide a solution to administer and secure this platform.
To do so, we implemented a web application using the Laravel framework, which allows the management of users, roles and permissions according to a role-based access control model (RBAC).The application includes a mini customer relationship management (CRM) through which we illustrate the administrator's ability to easily grant the necessary permissions to users on specific resources.

Keywords : security, access management,customer relationship management (CRM), Enterprise resource planning(ERP), RBAC, Laravel.

Liste des acronymes

API	Application Programming Interface
BI	Business Intelligence
CRM	Customer Relationship Management
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
CSPs	Cloud Services Providers
DAC	Discretionary Access Control
DOM	Document Object Model
EPM	Enterprise Performance Management
ERP	Enterprise Resource Planning
ETI	Entreprise de taille intermédiaire
FTP	File Transfer Protocol
HTML	Hypertext Markup Language
HTTP/S	HyperText Transfer Protocol Secure
IMC	Inventory Management and control
IaaS	Infrastructure as a Service
IA	Intelligence Artificielle
MAC	Mandatory Access Control
MRP	Material Resource Planning

MRP II	Manufacturing Resource Planning
MVC	Model View Controler
PGI	Progiciel de Gestion Intégré
PME	Petite ou moyenne entreprise
RBAC	Role Based Access Control
RFID	Radio Frequency Identification
RH	Ressources Humaines
SaaS	Software as a Service
SCM	Supply Chain Management
SGBD	Système de gestion de base de données
SMTP	Simple Mail Transfer Protocol
SoD	Segregation of Duties
SQL	Structured Query Language
SSL	Secure Sockets Layer
TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WAF	Web application firewall
XSS	Cross-site scripting
XML	Extensible Markup Language

Liste des figures

- Figure 1: Evolution des ERPs à travers le temps
- Figure 2 : Composants d'un ERP
- Figure 3 : Unités organisationnelles d'une entreprise
- Figure 4 : Utilisation d'un ERP en entreprise
- Figure 5 : Avantages et inconvénients d'un ERP propriétaire
- Figure 6 : Éléments du Cloud
- Figure 7 : Utilisation du Cloud en entreprise
- Figure 8 : Nombre d'utilisateurs de Smartphones dans le monde de 2016 à 2020
- Figure 9 : Nombre de Smartphones vendus aux utilisateurs finaux dans le monde entre 2007 et 2020
- Figure 10 : ERP et Business Intelligence
- Figure 11 : Pare-feu d'application web
- Figure 12 : Illustration du contrôle d'accès de type DAC
- Figure 13 : Illustration du contrôle d'accès de type MAC
- Figure 14 : Composants du modèle 'Flat RBAC '
- Figure 15 : Composants du modèle ' Hierarchical RBAC'
- Figure 16 : Composants du modèle ' Constrained RBAC '
- Figure 17 : Composants du modèle ' Symmetric RBAC '
- Figure 18 : Architecture d'une application web
- Figure 19 : Les 4 vulnérabilités des applications web les plus répondues
- Figure 20 : Les vues UML

- Figure 21 : Les diagrammes UML
- Figure 22 : Eléments du diagramme de séquence
- Figure 23 : Structure organisationnelle d'une entreprise commerciale
- Figure 24 : Diagramme de cas d'utilisation global
- Figure 25 : Diagramme de cas d'utilisation 'PDG'
- Figure 26 : Diagramme de cas d'utilisation 'Responsable commercial'
- Figure 27 : Diagramme de cas d'utilisation 'Commercial'
- Figure 28 : Diagramme de séquence ' Modifier mot de passe '
- Figure 29 : Diagramme de séquence ' Ajouter un employé '
- Figure 30 : Diagramme de séquence ' Ajouter un rôle '
- Figure 31 : Diagramme de classe
- Figure 32 : Tables relatives à la partie d'administration
- Figure 33 : Tables relatives à la partie CRM
- Figure 34 : Authentification
- Figure 35 : Saisie du code de vérification
- Figure 36 : Accueil
- Figure 37 : Liste des modules
- Figure 38 : Liste des utilisateurs
- Figure 39 : Formulaire d'ajout d'utilisateur
- Figure 40 : Modifier le mot de passe

Liste des tableaux

Tableau 1: Descriptif des solutions ERPs de SAP,Oracle et Microsoft

Tableau 2: Comparaison entre SAP, Oracle et Microsoft Dynamics

Tableau 3: Classement OWSAP des failles de sécurité les plus courantes

Tableau 4: Tableau descriptif des rôles au sein d'une entreprise commerciale et des niveaux d'accès associés

Tableau 5: Descriptif des besoins fonctionnels

Sommaire

Résumé……4
Abstract……...5
Liste des acronymes……....6
Liste des figures……...8
Liste des tableaux …….. 10
Introduction ……... 15
I. **Plate-formes ERP**…….. 17
Introduction …….. .17
A. Généralités sur les ERPs ……..17
1. Définition ……....17
2. Histoire ……....19
3. Architecture interne ……..... ……....20
4. Classification des solutions ……. …….....22
a) Open source et propriétaire …… 22
(1) ERP open source ……..22
(a) Définition ……22
(b) Avantages et inconvénients ……22
(2) ERP propriétaire ……..22
(a) Définition ……22
(b) Avantages et inconvénients ……....23
b) ERP généraliste, ERP vertical et ERP métier……...23
(1) ERP généraliste……23
(2) ERP vertical……23
(3) ERP métier……...23
c) ERP monolithique et approche best of breed ……24
(1) ERP monolithique……24
(a) Définition……24
(b) Avantages et inconvénients……....24
(2) Approche best of breed……25
(a) Définition……..25
(b) Avantages et inconvénients……..25
B. Segmentation du marché des ERPs……26
1. Marché des ERPs Open Source……...26
a) Odoo……26
2. Marché des ERPs propriétaires ……..27
a) SAP , Oracle et Microsoft ……..……28
C. RFID, Cloud , Mobilité , BI et ERPs……..29
1. Radio Frequency Identification (RFID)……..29
2. ERP Cloud ……...30
3. ERP Mobile……32

4. ERP et Business Intelligence……...34
Conclusion ……35

II. **Administration et sécurité des plate-formes numériques**……...37
Introduction ……..37
A. Généralités sur la sécurité informatique……..37
1. Services et mécanismes de la sécurité ……37
a) Services de la sécurité ……37
(1) Identification et authentification ……37
(2) Confidentialité ……37
(3) Contrôle d'accès ……37
(4) Intégrité et non répudiation ……37
(5) Disponibilité……38
b) Mécanismes de la sécurité ……...38
(1) Cryptage ……..38
(2) Signature numérique ……...38
(3) Échange d'authentification ……..38
(4) Horodatage……...39
(5) Traffic padding……39
(6) Détection d'intrusions ……39
(7) WAF……39
B. Sécurité et contrôle d'accès……..40
1. Politique de contrôle d'accès……....40
2. Concepts de base ……40
3. Types de politiques de contrôle d'accès……..40
a) Discretionary Access Control (DAC)……41
b) Mandatory Access Control (MAC) ……42
c) Role Based Access Control (RBAC)……...44
(1) Composants du modèle RBAC……44
(2) Règles de base du modèle RBAC……44
(3) Le modèle NIST-RBAC ……..45
(a) Flat RBAC……....45
(b) Hierarchical RBAC……..46
(c) Constrained RBAC……..46
(d) Symmetric RBAC……46
C. Sécurité et applications web……47
1. Architecture des applications web ……..47
2. Risques liés à la sécurité des applications……47
3. Vulnérabilités communes des applications web……..50
a) SQL Injection……...50
b) Cross Site Scripting (XSS)……..51
c) Information Leakage……52

d) Insufficient Transport Layer Protection……...53
Conclusion ……..54
III. **Analyse et Conception** ……..…….……...56
Introduction ……...56
A. Généralités sur l'UML…….56
1. UML……....56
2. Diagrammes UML……...56
3. Diagramme de cas d'utilisation …….58
4. Diagramme de classes …….58
5. Diagramme de séquence……..59
B. Analyse……60
1. Identification des acteurs ……60
2. Spécification des besoins ……63
3. Représentation des diagrammes de cas d'utilisation……....64
a) Diagramme de cas d'utilisation global ……...…64
b) Diagrammes des cas d'utilisation détaillés …….65
(1) Cas d'utilisation 'PDG'……....65
(2) Cas d'utilisation 'Responsable commercial '……...66
(3) Cas d'utilisation 'Commercial'……..…..67
C. Conception……..68
1. Représentation des diagrammes de séquence……..68
a) Diagramme de séquence du cas d'utilisation 'Modifier le mot de passe'……..68
b) Diagramme de séquence du cas d'utilisation 'Ajouter un utilisateur '……...69
c) Diagramme de séquence du cas d'utilisation 'Ajouter un rôle'.....70
2. Représentation du diagramme de classe……..71
3. Structure de la base de données……...72
Conclusion ……73
IV. **Implémentation et réalisation** ……..74
Introduction ……...……75
A. Technologies de développement frontend……..75
1. HTML5……...75
2. CSS3……..76
3. JavaScript……76
4. JQuery……76
5. Bootstrap 4…….……....77
6. Material design ……...78

B. Technologies de développement backend……...79
1. Laragon ……..79

2. PHP ……79
3. MySQL ……..79
4. Apache Server ……80
5. PhpMyAdmin ……80
6. Laravel ……..81

C. Interfaces graphiques ……82
1. Interfaces d'authentification ……..83
2. Interface d'accueil……...84
3. Interface 'Liste des modules'…….....84
4. Interface 'Liste des utilisateurs'……..85
5. Interface 'Ajout d'un utilisateur'……85
6. Interface 'Modifier le mot de passe '……..86
Conclusion ……86

Conclusion et perspectives……88
Références bibliographiques ……....89

Introduction

En réponse à l'intensification de la concurrence mondiale, de nombreuses entreprises ont mis en place de nouveaux systèmes d'informations (SI), connus sous le nom d'ERP. Les systèmes ERP sont des logiciels capables de "gérer l'ensemble des processus d'une entreprise"[14].

Les entreprises qui mettent en œuvre des systèmes ERP bénéficient de nombreux avantages, notamment l'amélioration de la productivité, l'avantage concurrentiel, en satisfaisant la demande des clients et en augmentant leur capacité de réaction rapide.

Les systèmes ERP permettent aux gestionnaires de contrôler l'ensemble de l'entreprise et d'accélérer la prise de décision. Etant donné que les ERPs fournissent un moyen de centraliser l'information, il est impératif de disposer de mécanismes de contrôle d'accès à ces informations , d'administrer les plate-formes ERP mais aussi de les sécuriser.
Dans cette optique,nous proposons une application web fournissant une interface de gestion des utilisateurs, des rôles et des permissions.

Notre travail consiste à développer une application web qui permette d'administrer et sécuriser un ERP afin de contrôler l'accès aux différents éléments du système.

Pour la réalisation de tout projet, le suivi d'une méthodologie de travail s'impose.
Nous avons donc réparti notre travail sur quatre chapitres.

Chapitre 1: Plate-forme ERP

Ce chapitre traite des généralités des plate-formes ERP ,afin de comprendre leur fonctionnement.

Chapitre 2 : Administration et sécurité des plate-formes numériques

Dans ce chapitre, nous expliquons ce qu'est une politique de contrôle d'accès.Mais aussi les vulnérabilités des applications web afin de prendre conscience de la nécessité de sécuriser les données.

Chapitre 3 : Analyse et conception

À cette étape du projet, nous entrons dans le vif du sujet en analysant les besoins et en définissant le champ d'action du système. Nous modélisons notre solution à travers l'utilisation de diagrammes UML.

Chapitre 4 : Implémentation et réalisation

Ce chapitre est axé sur l'aspect technique du projet, les différents outils utilisés pour réaliser l'application y sont introduits.
Nous partageons également des interfaces graphiques afin d'illustrer l'aboutissement de notre travail.

Chapitre 1 :

Plate-formes ERP

Introduction

Ce chapitre a pour objectif de présenter globalement les plate-formes ERP.

Dans un premier temps, nous définirons le concept d'ERP, son évolution à travers le temps ainsi que l'architecture interne des ERPs.

Par la suite nous classifierons les solutions ERP en plusieurs catégories, et nous parlerons de la segmentation du marché des ERPs.

Pour finir, quelques-unes des technologies contribuant à l'amélioration du fonctionnement des ERPs seront présentées.

A. Généralités sur les ERPs

1. Définition d'un ERP

ERP «Enterprise Resource Planning» - «Planification des ressources de l'Entreprise »-: est un terme anglais apparu au début des années 1990.

"Un ERP désigne une application informatique permettant à une entreprise de gérer et d'optimiser l'ensemble de ses ressources."[9]

" L'ERP se définit comme un système qui permet d'automatiser et de gérer les processus d'entreprise au niveau des finances, de la fabrication, de la distribution, de la chaîne d'approvisionnement, des ressources humaines et des opérations. " [12]

Le terme français **Progiciel de Gestion Intégré (PGI)** ,n'est pas une traduction littérale de l'abréviation ERP. Un PGI est un " logiciel permettant de gérer l'ensemble des processus d'une entreprise , en intégrant l'ensemble des fonctions de cette dernière, comme la gestion des ressources humaines, la gestion comptable et financière, l'aide à la décision, mais aussi la vente, la distribution, l'approvisionnement, le commerce électronique."[14]

Le terme ERP s'apparente davantage à l'aspect fonctionnel tandis que le terme PGI a une approche plus globale. [10]

Bien qu'il y ait une différence entre ces deux termes, ils sont employés comme des termes synonymes.

2. Histoire des ERPs [8]

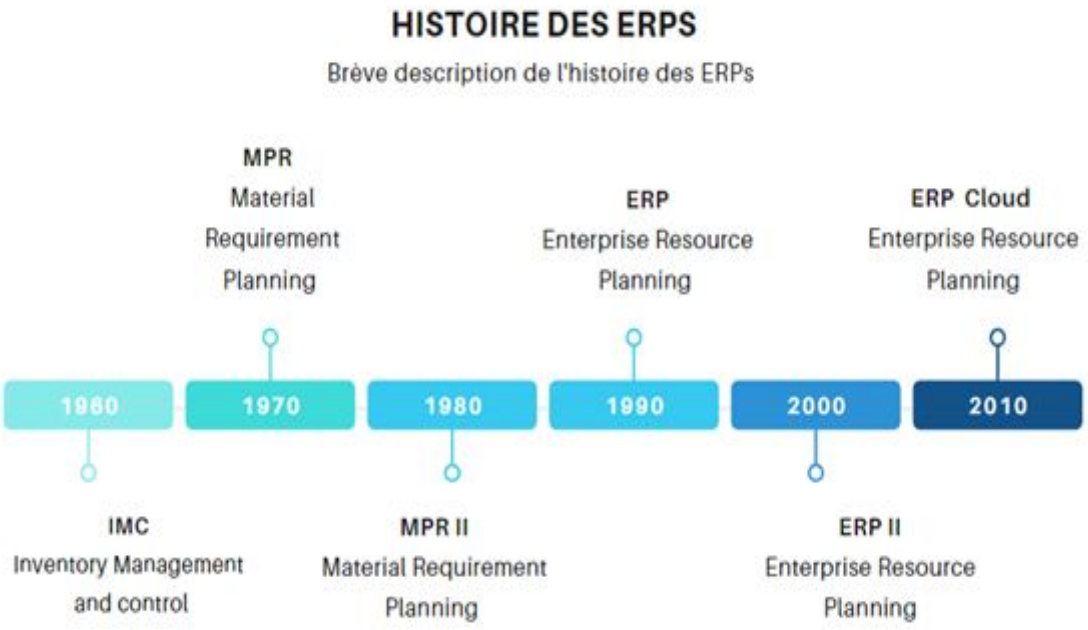


Figure 1: Evolution des ERPs à travers le temps

L'histoire des ERPs remonte aux années **1960**, lorsque les premiers systèmes logiciels basés sur l'automatisation de fonctions individuelles, furent inventés.
Parmi ces systèmes l'on peut citer les logiciels de gestion et contrôle des stocks ou (Inventory Management and control) en anglais.Ces derniers permettent de maintenir le niveau approprié de stock dans un entrepôt.

La planification des besoins en matériaux avec les **MPRs** (Materials Requirement Planning) remonte aux années **1970**. Elle utilise des applications logicielles afin de planifier les processus de production.

En **1980**, les MRPs ont évolué, et une nouvelle appellation fut attribuée à ces systèmes: **MRP II**. Alors que le MRP concernait principalement les matériaux, le MRP II s'intéressait à l'intégration de tous les aspects du processus de fabrication, y compris les matériaux, les finances et les relations humaines.

À l'époque, la plupart des systèmes d'entreprise se concentraient sur un processus d'entreprise particulier. Les entreprises avaient donc recours à plusieurs systèmes qui étaient utilisés uniquement pour un processus spécifique.L'utilisation de systèmes distincts pour la gestion des processus entraîna la nécessité de transférer les données entres ces systèmes afin d'effectuer les traitements nécessaires au niveau de chacun d'eux. Il en résultait de nombreuses erreurs humaines, des processus lents, des informations incohérentes et des rapports inefficaces.

C’est en **1990**, que le terme "**Enterprise Resource Planning**" fut inventé par le Gartner Group pour décrire la prochaine génération de logiciels MRP.
Cette fois l'objectif était d'intégrer toutes les facettes de l'entreprise sous une seule suite d'applications logicielles.

Pendant cette période, les modèles de base des logiciels ERP ont été introduits et intégrés dans la conception. Parmi ceux-ci figurait l'architecture Client/Serveur qui a permit d'héberger des ressources telles que les bases de données, dans des lieux centraux, et de distribuer des ressources, comme l'interface utilisateur à d'autres endroits.

Dans les années **2000**, l’**ERP II** a été mis en place.L’ERP II permit l'accès à d'autres fonctionnalités en dehors de l'entreprise,à savoir la gestion de la relation client (CRM[1]) et la gestion de la chaîne d'approvisionnement (SCM).

En **2010**, le cloud devint plus accessible et permit aux entreprises de profiter de ses services notamment l’IaaS. Ce fut peut-être le plus grand coup de pouce à l'expansion et à l'évolution des capacités des logiciels ERP, et ce grâce à la disponibilité universelle de l'internet et au développement simultané des technologies basées sur le web. Cela a permit de stocker, de gérer et d'accéder à des données, de n'importe où, grâce à une connexion Internet.

L’**ERP cloud** permet de bénéficier de toutes les fonctionnalités qu'un logiciel ERP peut offrir sans avoir à acheter et à entretenir toute une infrastructure informatique.
La flexibilité de ces systèmes permet aux utilisateurs de gérer leurs entreprises à tout moment et en tout lieu.

De nombreuses organisations utilisent déjà le Cloud (SAP, Oracle, Microsoft …) ou ont l'intention de migrer leur ERP dans un avenir proche “...De manière générale, une majorité d’entreprises européennes (57%) pensent qu’elles auront entamé une migration à 100% dans le cloud de leur ERP dans les 12 prochains mois. Elles sont même 71% à considérer le sujet « ERP dans le cloud » comme une priorité des prochains mois.

Selon l’étude, aucune des entreprises européennes ne prévoit d’avoir des déploiements ERP « 100% on-premise » (100% sur site) dans l’année à venir, les applications ERP étant de plus en plus basées sur le Cloud...” Laurent Delattre

Les logiciels ERP ont grandement évolué. Les entreprises de toutes tailles et spécialités ont la possibilité d'adapter ces systèmes à leurs besoins particuliers, ce qui leur permet d'améliorer leur efficacité.

[1] “Logiciel créé expressément pour permettre à une entreprise de fidéliser ses clients et d'accroître sa part du marché, en intégrant à son système informatique la gestion des données relatives aux besoins et aux attentes de sa clientèle.”[14]

3. Architecture interne d'un ERP

Figure 2 : Composants d'un ERP

Un ERP est un système dans lequel coexistent plusieurs modules (comptabilité, ventes, stocks, CRM, RH, etc.) interconnectés à travers une base de données unique, le principe d'un ERP étant de centraliser la base de données de l'entreprise, une modification sur l'un des modules se répercute nécessairement sur les autres modules permettant ainsi de mettre à jour la base de données selon les modifications observées.

Exemple [5]

Habituellement, dans une entreprise lorsqu'un commercial prend la commande d'un client, le workflow[2] suivant en découle :

- (1) Le client prend contacte avec le commercial afin de se renseigner sur la disponibilité du produit.
- (2) Le commercial entre en contact avec le service de gestion des stocks afin de vérifier que le produit est disponible.

[2] "the way that a particular type of work is organized, or the order of the stages in a particular work process." https://dictionary.cambridge.org

- (3) Dans le cas où le produit n'est pas disponible une commande sera passée au service de production.
- (4) Le service de gestion des stocks est à nouveau sollicité pour vérifier la disponibilité de la matière première afin de lancer la production
- (5) Dans le cas où la matière première est indisponible il faudra prendre contact avec des fournisseurs (vendeurs) .
- (6) Ensuite, la production est lancée.
- (7) Une fois la commande prête, elle est envoyée au service des ventes.
- (8) Le service des ventes se chargera de fournir le produit au client.

La figure suivante illustre ce workflow

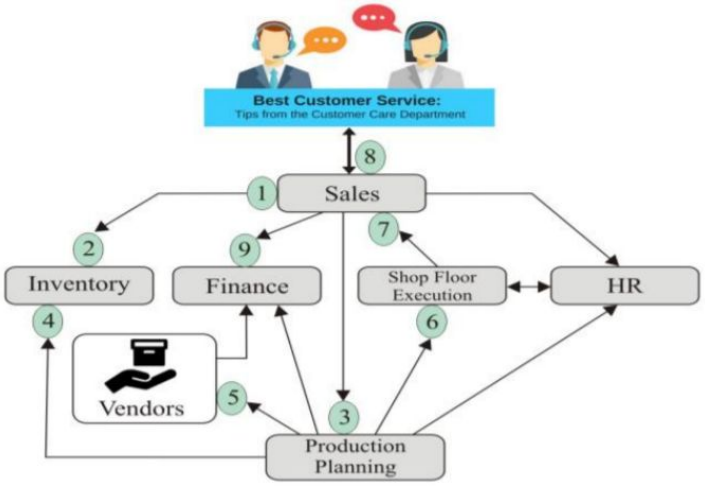


Figure 3 : Unités organisationnelles d'une entreprise [5]

Pour reprendre le workflow évoqué précédemment, dans le cas de l'utilisation d'un ERP et donc d'une solution de gestion centralisée le scénario sera le suivant :

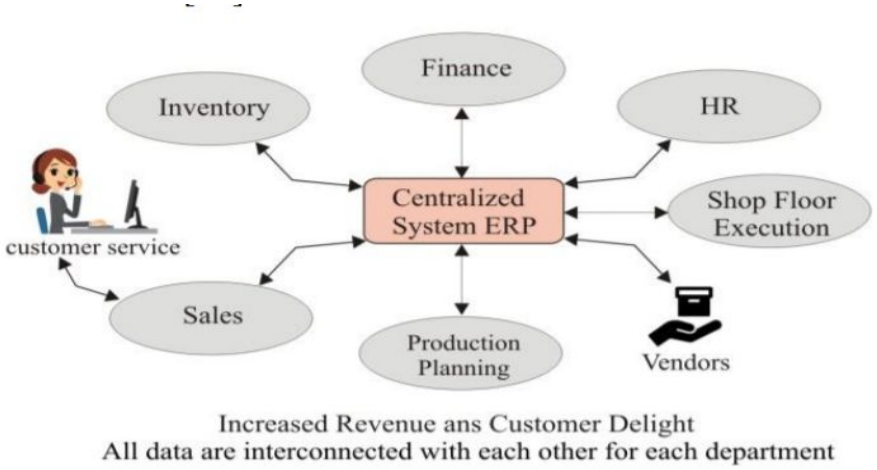


Figure 4 : Utilisation d'un ERP en entreprise [5]

- (1) Quand le client passe sa commande auprès du commercial, le commercial est en mesure de lui fournir une réponse en temps réel et ce grâce au fait qu'il ait accès aux données du service de gestion des stocks .
- (2) Le commercial répond à la demande du client dans les délais, ce qui permet d'améliorer les recettes et la satisfaction des clients.

4. Classification des solutions ERP

a) ERP Open Source et ERP Propriétaire [10]

(1) ERP Open Source

(a) Définition

Un ERP Open Source ou ERP Libre en français, est un ERP dont le code source est libre d'accès. Ce dernier est modifiable c'est à dire qu'il est possible d'ajouter les fonctionnalités manquantes afin de satisfaire les besoins de l'entreprise ou à l'inverse supprimer ce qui l'alourdit autrement dit les fonctionnalités dont l'entreprise n'a pas besoin. Le code source de l'ERP Open source est réutilisable dans les mêmes conditions que la licence d'origine.

(b) Avantages et inconvénients

Contrairement à l'ERP propriétaire qui est payant et qui n'est pas modifiable, l'ERP Open source est gratuit et modifiable.
Le fait que l'ERP Open Source soit modifiable est à la fois un avantage et un inconvénient.Il est possible de personnaliser pour répondre aux besoins de l'entreprise mais cette action nécessite de parcourir plusieurs lignes de code afin de comprendre son fonctionnement. L'entreprise devra donc faire appel à des informaticiens compétents pour faire ceci.Elle devra également prévoir des coûts de formation pour l'utilisation de l'ERP en plus des coûts de développement spécifique.

(2) ERP Propriétaire

(a) Définition

Est opposé à un ERP open source et désigne un ERP qui utilise un code propriétaire et étroitement surveillé. Seuls les auteurs originaux de ce logiciel peuvent accéder, copier et modifier ce logiciel. Dans le cas d'un logiciel propriétaire, Il faut avoir une licence ou un abonnement afin de pouvoir utiliser l'ERP et cette licence/abonnement permet uniquement d'utiliser le logiciel mais ne donne pas accès au code source.

(b) Avantages et inconvénients

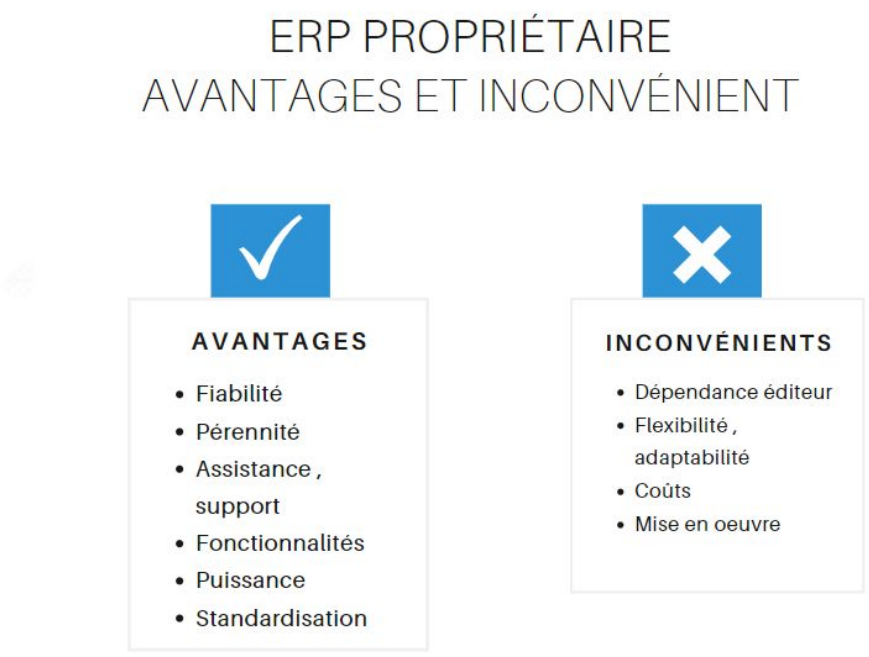


Figure 5 : Avantages et inconvénients d'un ERP propriétaire

b) ERP généraliste ,vertical et application métier [10]

(1) ERP généraliste

Un ERP généraliste est un ERP standard , non modifiable , pouvant être adapté aux besoins de n'importe quelle entreprise.

(2) ERP vertical

L' ERP vertical ou vertical métier est un ERP généraliste disposant de fonctionnalités spécifiques (fonctionnalités métiers) développées par l'intégrateur.

(3) ERP métier

C'est un ERP dédié , conçu par un éditeur spécialisé dans un secteur d'activité particulier. Il permet de répondre aux besoins spécifiques de ce métier.

c) ERP monolithique ou 'Best of breed' [10]

(1) ERP monolithique

(a) Définition

Un ERP monolithique est conçu selon une architecture monolithique, autrement dit sous forme d'un seul bloc. Il peut être défini comme étant une suite logicielle dont les composants sont interconnectés et interdépendants.

C'est une solution autonome qui permet de gérer l'ensemble des processus de l'entreprise.

Avantages et inconvénients

Avantage

Le plus grand avantage de cette approche est de loin l'intégration complète des données entre les différentes fonctions. L'approche monolithique a d'autres avantages, l'on peut citer : interface utilisateur unique ce qui simplifie la familiarisation des employés avec la solution proposée ,soutien d'un fournisseur unique pour faciliter la gestion ...

Inconvénients

L'inconvénient de cette approche est que, très souvent, une seule suite ne peut répondre de manière adéquate à des fonctions essentielles spécifiques. Les inconvénients supplémentaires sont les suivants :

- La personnalisation est plus difficile.
- La viabilité des fournisseurs est un risque d'où l'importance de choisir le bon fournisseur.
- L'entreprise est liée à un seul fournisseur pour l'assistance et la maintenance.

(2) Approche best of breed

(a) Définition

La stratégie "Best-of-Breed" consiste à sélectionner le produit qui répond le mieux aux exigences de l'entreprise .Ce produit étant en général un logiciel spécialisé dans un domaine précis et lié à l'ERP par le biais de connexions statiques ou dynamiques.

Prenons à titre d'exemple le CRM, un CRM peut être un module de l'ERP ou encore un logiciel spécialisé. Les CRMs intégrés disposent en générale de fonctionnalités réduites comparés aux logiciels spécialisés. Une entreprise qui nécessite des fonctionnalités avancées pourra utiliser un logiciel spécialisé en CRM et synchroniser la communication avec L'ERP, le transfert de données entre l'ERP et le CRM peut se faire en statique (nécessite une intervention manuelle) ou en encore de manière dynamique (l'ERP peut échanger directement avec le logiciel CRM et envoyer les données directement dans sa base de données sans intervention humaine.).

(b) Les avantages et inconvénients

Avantages de best of breed

Le principal avantage de cette approche est de garantir le traitement des fonctionnalités spécifiques à travers l'utilisation de logiciels spécialisés.

Outre les traitements des fonctionnalités spécifiques, la gestion des mises à jour des systèmes est plus flexible, un large choix de systèmes et de fournisseurs spécialisés s'offre à l'utilisateur, et la maintenance des applications modulaires est moins fastidieuse.

Inconvénients de l'approche best of breed

Comme toute chose cette solution a également quelques inconvénients, ces derniers sont des problèmes liés à la multiplicité des systèmes, des bases de données et des fournisseurs. L'adoption de cette solution entraîne un coût total de possession élevé en raison des coûts d'assistance multiples des fournisseurs. Il faudra aussi investir dans une formation disjointe entre les services.

B. Segmentation du marché des ERPs

Après avoir classé les solutions ERP en plusieurs catégories nous allons voir quels sont les principaux acteurs des deux catégories suivantes : les EPPs open source et les ERPs propriétaires.

1. Marché des ERPs Open Source :

a. Odoo [20]

[20]

Odoo est une suite intégrée d'applications comprenant de nombreux modules qui permettent de répondre aux besoins de gestion des entreprises.

Odoo, l'ERP open source le plus populaire est écrit en python, il est compatible avec les systèmes d'exploitation suivants : Linux, type Unix, MacOs et Microsoft Windows.
Odoo est basé sur une architecture MVC et utilise une base de données PostgreSQL.

Les 4 applications principales de l'ERP Odoo sont : les Sites web, les ventes, les opérations et les outils de productivité.

Avec 31 modules (Ventes, CRM, RH, comptabilité, Inventaire, etc.) répartis sur 4 applications cette solution dispose d'une large couverture fonctionnelle lui permettant ainsi de s'intégrer à toute taille et toute sorte d'entreprise.

De plus Odoo propose une version community gratuite sous licence LGPLv3, ainsi qu'une version entreprise sous licence propriétaire Odoo Enterprise Edition License v1.05.

2. Marché des ERPs propriétaires

a. SAP , Oracle et Microsoft

Editeur	SAP	Oracle	Microsoft
Cible	*PME, *ETI *Grandes entreprises	*PME *ETI *Grandes entreprises	*PME, *ETI, Grandes entreprises
Produits	*SAP ERP , *S/4 HANNA , *S/4 Cloud *SAP Business One, *SAP Business By Design	Oracle ERP Cloud	Microsoft Dynamics AX, Microsoft Dynamics GP, Microsoft Dynamics NAV, Microsoft Dynamics SL , Microsoft Dynamics C5. Microsoft Dynamics 365 plan (ERP+CRM).
Modules	23 modules, divisés en 3 catégories: *Logistique : Gestion de production, ventes, gestion des stocks, service client... *Comptabilité : Comptabilité financière, control de gestion ... *RH : Gestion des employés, Gestion de la paie...	9 Modules principaux : *Services financiers *Pôle comptabilité *Approvisionnement *Gestion de projet *Gestion des risques *EPM *Applications IA *SCM *Net Suite	11 modules principaux de Microsoft Dynamics 365 : * Ventes *Service client *Service mobile *RH *Finances et opérations *Vente au détail *Automatisation des services de projet *Marketing *IA *Réalité mixte *Business Central
Déploiement	Cloud ERP, On-Premise,Hybride	Cloud: privé, public, hybride	Cloud, On-Premise
SGBD	Microsoft SQL Server, Oracle, My SQL, SAP HANA. Dernière version utilise uniquement:SAP HANNA	Oracle	SQL Server

Tableau 1: Descriptif des solutions ERPs de SAP , Oracle et Microsoft [12] [21] [23]

En 2012, le système logiciel SAP contrôlait environ 25 % de part du marché des ERP. Contre 13% pour Oracle et 5% pour Microsoft Dynamics. [5]

En 2019 , SAP reste le leader des ERPs , Microsoft Dynamics se place en deuxième position et Oracle en troisième position du classement . [5]

	SAP	Oracle	Microsoft Dynamics
store Part	19%	13%	16%
Short-list Rate	38%	18%	31%
Collection Rate When Short Recorded	38%	22%	22%
Application Period	23.1 months	24.5 months	23.6 months
Total Price of Ownership	$2.09 million	$2.38 million	$2.06 million
Reimbursement Period	30 months	29 months	12 months
Disruption at Go-live	44%	42%	41%
Realized 50%+ of Anticipated Business Benefits	34%	21%	26%

Tableau 2 : Comparaison entre SAP , Oracle et Microsoft Dynamics [5]

SAP [5]

- Part importante du marché
- Taux de présélection élevé
- La plus longue période de remboursement

Oracle [5]

- Taux de sélection le plus élevé lors de la présélection
- Durée de mise en œuvre la plus longue
- Le plus grand delta entre la durée prévue et la durée réelle de la mise en œuvre
- Le plus faible pourcentage d'utilisateurs ayant réalisé entre 81 et 100 % des bénéfices

Microsoft Dynamics [5]

- La plus petite part du marché
- Taux de présélection le plus bas
- Durée de mise en œuvre la plus courte
- Le pourcentage le plus élevé d'utilisateurs ayant réalisé entre 80 et 100 % des bénéfices

C. RFID, Cloud , Mobilité et Business Intelligence (BI)

Diverses technologies contribuent à l'amélioration du fonctionnement des ERPs.
Les technologies, telles que l'identification par radiofréquence (RFID), augmentent la quantité de données contenues dans les systèmes ERP.

Les technologies de la BI transforment les données des systèmes ERP afin d'en extraire de précieuses informations et aider à la prise de décisions.

Le Cloud computing ou informatique dans les nuages ainsi que les appareils mobiles modifient le lieu de stockage des données et les rendent plus accessibles.

1. Radio Frequency Identification (RFID) [7]

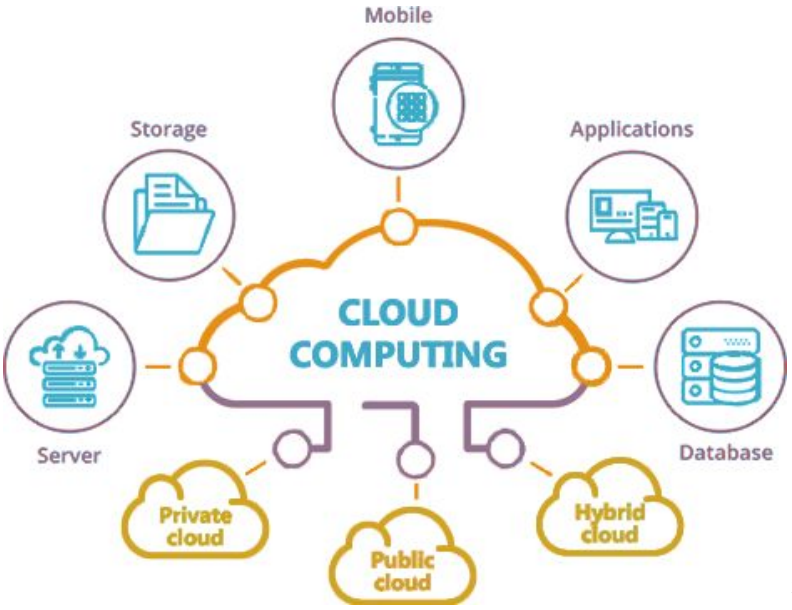
Le terme RFID fait référence à la technologie d'identification par radiofréquence.

C'est un outil permettant de suivre les articles tout au long de la chaîne d'approvisionnement. Le dispositif RFID peut être attaché aux produits.
Le lecteur RFID peut déterminer l'emplacement d'un article à l'aide d'une étiquette RFID. Il émet des ondes radio et reçoit des signaux en retour de l'étiquette.

Parmi les avantages de la technologie RFID, l'on peut citer :

- Ne nécessite pas de connexion en visibilité directe
- Peut résister à la plupart des contraintes environnementales

2. ERP Cloud



[3]

Figure 6 : Éléments du cloud

"Le Cloud Computing, désigne l'utilisation de serveurs distants (en général accessibles par Internet) pour traiter ou stocker l'information. L'accès se fait le plus souvent à l'aide d'un navigateur Web."[3]

Un ERP est dit ERP Cloud s'il satisfait les principales caractéristiques du Cloud Computing.

Un système ERP Cloud doit être accessible via le navigateur de l'utilisateur et sans que l'utilisateur n'ait besoin d'installer ni de configurer l'ERP.

Un ERP Cloud est fourni par le SaaS (Software as a Service). Avec le SaaS les fournisseurs de services Cloud, hébergent et gèrent les applications logicielles et l'infrastructure sous-jacente et gèrent également la maintenance.

Maintes solutions ERP sont proposées sur le marché, mais la solution la plus connue est SAP Business By Design fournie par SAP, leader des éditeurs d'ERP.

Selon l'étude menée par google [11] , le cloud joue déjà un rôle clé dans la technologie d'entreprise, mais les dix prochaines années le verront passer au premier plan, avec un soutien important des dirigeants. Voici comment ces données se répartissent dans le monde entier.

[3] réf : http://www.guide2midipyrenees.com

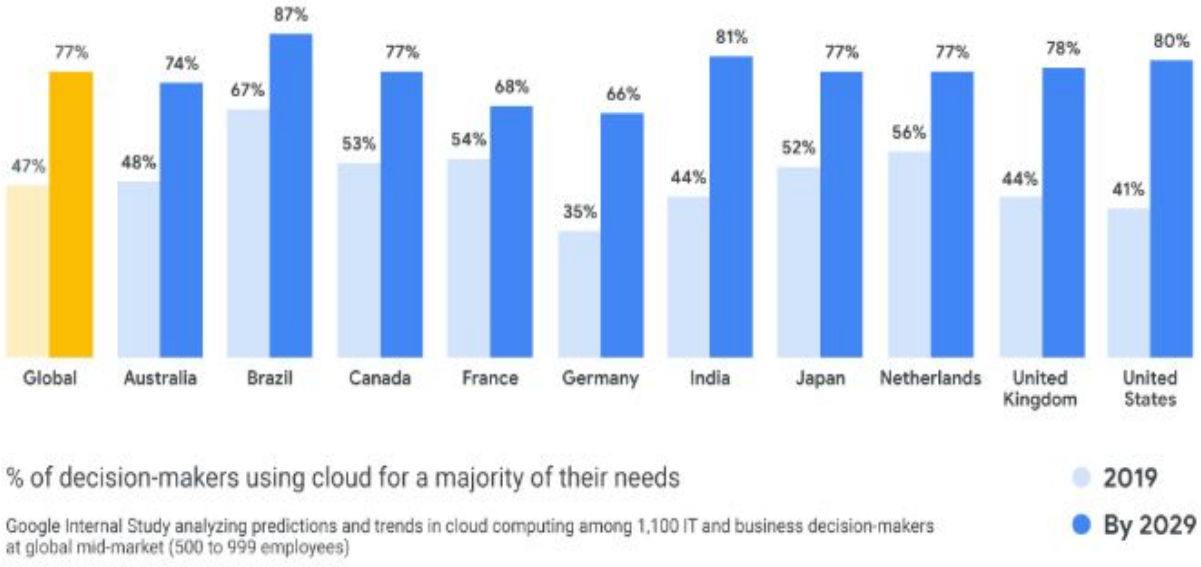


Figure 7 : Utilisation du cloud en entreprise [11]

La solution ERP Cloud présente de nombreux avantages [6], l'on peut citer :

Réduire les coûts initiaux :

Les entreprises n'ont pas besoin de payer pour la construction de l'environnement informatique mais devront souscrire à un abonnement afin d'accéder à cet environnement via internet.

Implémentation rapide :

Les CNPs (Cloud Services Providers) , fournisseurs de services Cloud proposent diverses solutions pouvant satisfaire les besoins des entreprises , c'est pourquoi l'entreprise pourra rapidement mettre en place l'ERP , en choisissant l'offre la plus adéquate.

La scalabilité :

L'un des principaux atouts des services cloud est l'élasticité, les entreprises peuvent augmenter ou réduire les ressources qu'elles utilisent selon leurs besoins.

Se focaliser sur les compétences métiers :

Une solution Cloud ERP prend en charge la mise à jour, ainsi que la mise à niveau de l'ERP selon la demande des entreprises, permettant ainsi aux entreprises de se focaliser sur des tâches plus spécifiques.

Accessibilité, mobilité, utilisabilité :

Un ERP Cloud est accessible à tout moment, que ça soit à l'intérieur ou à l'extérieur de l'entreprise.

Amélioration de la disponibilité du système et de la reprise après sinistre :

CSPs fournissent des politiques et des plans bien définis pour la sauvegarde, la restauration, et toutes les autres fonctions qui concernent la disponibilité et la reprise après sinistre.

Transparence des coûts :

Les entreprises payeront uniquement les ressources qu'elles utilisent (Pay-per-use) .

Utiliser les normes de sécurité :

Certains CSPs appliquent des normes pour le cryptage et le décryptage, épargnant ainsi aux clients (entreprises) de se préoccuper de la sécurité.

3. ERP Mobile

L'ERP mobile est utilisé par les entreprises voulant garantir un accès rapide, sécurisé et en temps réel à leurs données professionnelles en utilisant un terminal mobile.

L'utilisation d'un ERP mobile est bénéfique à tous les niveaux : elle permet de faciliter les relations commerciales, la gestion de l'entreprise et garantir la disponibilité des données nécessaires à l'accomplissement des tâches des employés nomades.

L'impact positif de l'ERP mobile est indéniable, une étude menée par CDW sur 752 sociétés a révélé que 94% de ces entreprises affirment que leurs appareils mobiles les rendent plus efficaces.
Les principaux avantages de l'ERP Mobile sont les suivants :

- Un meilleur pilotage :

  Une solution ERP Mobile facile le pilotage et la prise de décisions et ce en mettant à la disposition des équipes, les informations nécessaires y compris lorsqu'elles ne sont pas au bureau

- Améliorer la performance

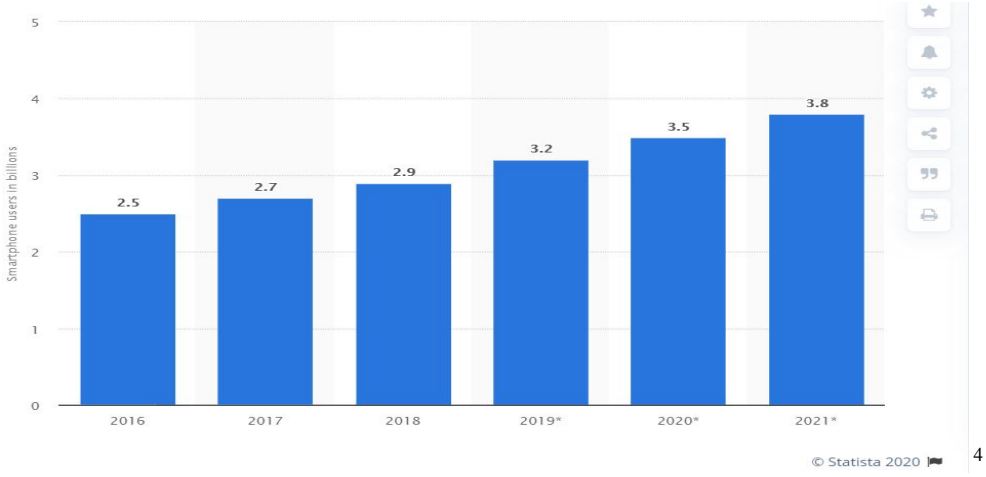
  Les employés peuvent accéder à l'information critique en tous lieux et à tout moment, ainsi ils pourront effectuer des opérations CRUD en temps-réel et ne pas attendre d'être en entreprise pour le faire.

- Implémentation :

  Le nombre d'utilisateurs de Smartphones dans le monde dépasse aujourd'hui les trois milliards et devrait encore augmenter de plusieurs centaines de millions dans les prochaines années. (figure 8)

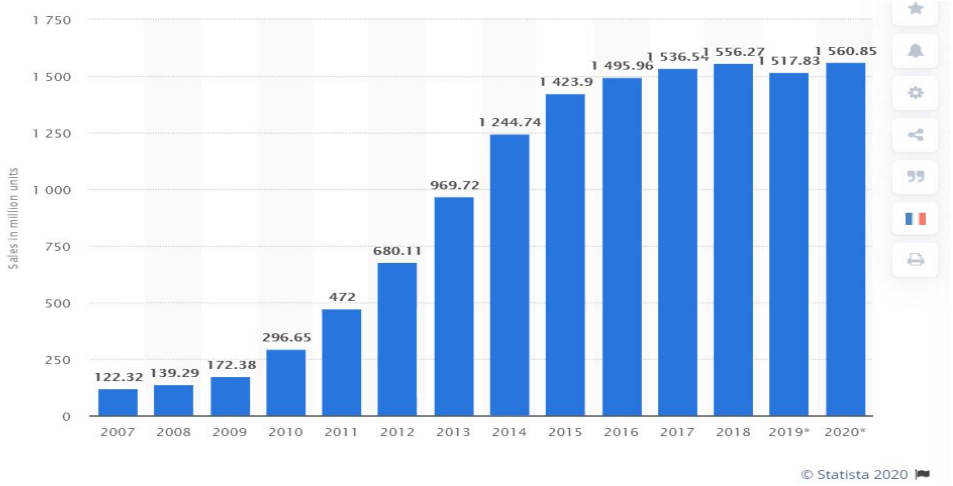
Les statistiques (figure 9) indiquent le nombre de Smartphones vendus aux utilisateurs finaux dans le monde entier entre 2007 et 2020. En 2018, environ 1,56 milliard de Smartphones ont été vendus dans le monde.

L'omniprésence des terminaux mobiles est un élément favorisant l'implémentation de solutions ERP mobiles et leurs adoptions afin d'accroître la productivité.



[4]

Figure 8 : Nombre d'utilisateurs de Smartphones dans le monde de 2016 à 2021

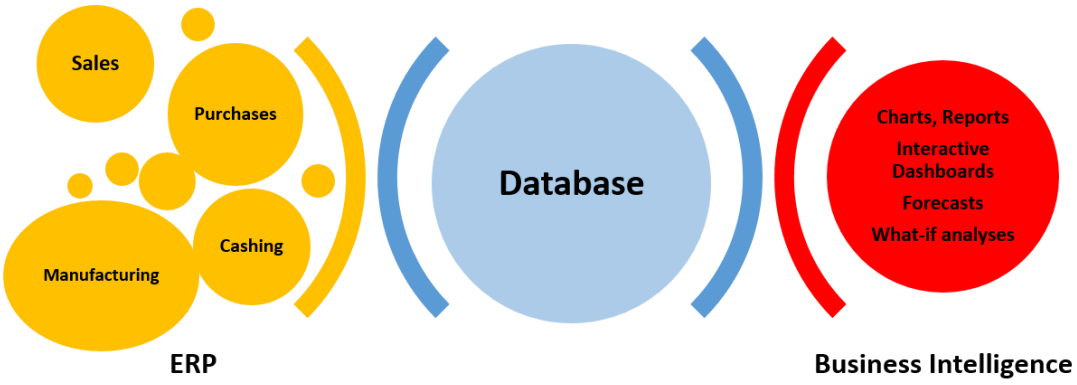


[5]

Figure 9 : Nombre de smartphones vendus aux utilisateurs finaux dans le monde entre 2007 et 2020

4. ERP et Business Intelligence

[4] réf : https://fr.statista.com/
[5] réf : https://fr.statista.com/



6

Figure 10 : ERP et Business Intelligence

Le terme Business Intelligence (BI) désigne les technologies, applications et pratiques de collecte, d'intégration, d'analyse et de présentation de l'information commerciale. L'objectif de la Business Intelligence est de soutenir une meilleure prise de décision commerciale. Essentiellement, les systèmes de Business Intelligence sont des systèmes d'aide à la décision axés sur les données. [21]

Une solution BI, aide les entreprises à regrouper des sources disparates à savoir : les données des systèmes de gestion de la relation client (CRM), les informations sur la chaîne logistique, les tableaux de bord des performances commerciales, les analyses marketing et les données d'appel des centres de contact, en une seule interface unifiée fournissant des rapports, des tableaux de bord et des analyses en temps réel.

De plus, la BI offre aux entreprises un moyen d'analyser et de comprendre les tendances du marché et à repérer les problèmes commerciaux qui doivent être résolus.

La solution ERP Cloud de SAP pour les entreprises moyennes en croissance, unifie toutes les fonctions essentielles dans une suite logicielle.
Avec plus de 500 rapports standards proposés par SAP plus particulièrement à travers sa solution Business ByDesign, les utilisateurs disposent d'une vue claire et des mesures nécessaires à la prise de décisions.[23]

[6] réf : https://www.seniorerp.ro

Conclusion

Ce chapitre nous aura permit d'aborder le concept d'ERP dans sa globalité.
Nous avons identifié les solutions qui s'offrent à nous en matière d'ERP ainsi que les acteurs phares du marché des ERPs Open Source (Odoo) et des ERPs propriétaires à savoir : SAP , Oracle , Microsoft Dynamics , les points forts de ces solutions furent abordés et une comparaison entre elles fut établie.

Enfin, nous avons présenté les technologies RFID, Cloud ,Mobilité et Business Intelligence (BI) et vu comment ces dernières contribuent à l'amélioration du fonctionnement des ERPs.

Chapitre 2 :

Administration et sécurité des plate-formes ERPs

Introduction

Dans ce chapitre, nous commencerons par aborder des généralités sur la sécurité informatique, nous verrons les principaux services et mécanismes de la sécurité.

Dans la seconde partie, il s'agira de définir ce qu'est une politique de contrôle d'accès ainsi que les concepts de base de cette dernière. Nous nous intéresserons également aux types de politiques de contrôle d'accès les plus répondus.

Enfin, la troisième partie sera consacrée à la sécurité des applications web, nous verrons les risques liés à la sécurité de ces dernières ainsi que les vulnérabilités communes de ces applications.

A. Généralités sur la sécurité informatique

La sécurité informatique, c'est l'ensemble des moyens mis en œuvre pour minimiser la vulnérabilité d'un système informatique contre des menaces accidentelles ou intentionnelles [4].

1. Services et mécanismes de la sécurité [4]

a) Les services de la sécurité

Les services de la sécurité sont des services améliorant la sécurité des systèmes informatiques et des transferts d'information. Ils sont conçus pour contrer les attaques de sécurité et utilisent un ou plusieurs mécanismes de sécurité afin d'y parvenir.

Les besoins primaires des applications en terme de sécurité sont les suivants :

(1) Identification et authentification

L'identification est la capacité à identifier de manière unique un utilisateur d'un système ou d'une application qui fonctionne dans le système. L'authentification est la capacité à prouver qu'un utilisateur ou une application est bien la personne ou l'application qu'il/elle prétend être.

(2) Confidentialité

La confidentialité consiste à assurer que seules les personnes autorisées puissent accéder aux ressources. Les mécanismes de confidentialité protègent les informations sensibles contre toute divulgation non autorisée.

(3) Contrôle d'accès

Le contrôle d'accès vise à empêcher l'utilisation non autorisée d'une ressource (serveur, application, etc.) en définissant les droits d'accès à ces ressources ; les conditions d'accès, et les actions pouvant être effectuées par l'entité qui y accède.

(4) Intégrité et non répudiation

L'intégrité a pour objectif de garantir que les données soient bien celles que l'on croit être. La non-répudiation garantit que l'occurrence d'une transaction ne puisse être niée.

Les mécanismes d'intégrité et de non-répudiation des données détectent si une modification non autorisée des données a eu lieu.

(5) Disponibilité

Propriété permettant de maintenir le bon fonctionnement du système en garantissant que le système ou la ressource soit accessible et utilisable suite à la demande d'une entité autorisée.

b) Les mécanismes de sécurité

Les mécanismes de sécurité sont des mécanismes conçus pour détecter, prévenir ou contrer les attaques de sécurité.

Il existe plusieurs mécanismes de sécurité [4], l'on peut citer le cryptage, la signature numérique, l'échange d'authentification, l'horodatage, l'utilisation de pare-feu, etc.

Exemples de mécanismes de sécurité

(1) Cryptage

Consiste à utiliser des algorithmes mathématiques pour transformer les messages en forme inintelligible.

Exemple :

OpenSSL [1] : une implémentation open source des protocoles SSL et TLS. Cette bibliothèque met en œuvre des fonctions cryptographiques de base et fournit diverses fonctions utilitaires.

Au lieu de transmettre les paquets en texte clair, il permet de crypter les informations à l'aide d'un algorithme puissant. La source et la destination effectuent un 'handshake' et se mettent d'accord sur une clé longue et compliquée qui sera utilisée pour décrypter les données lorsqu'elles arrivent à destination en toute sécurité.

Par conséquent, même si un attaquant accède à un dispositif de connexion et lit les paquets qui y transitent, il lui sera très compliqué de les décrypter.

(2) Signature numérique

Consiste à ajouter des informations cryptées à une unité de données afin de prouver la source et l'intégrité de cette dernière.

(3) Échange d'authentification

L'échange d'authentification est un mécanisme assurant l'identité d'une entité à travers un échange d'information.

(4) Horodatage

L'horodatage est l'association d'une date et une heure à un événement une information ou une donnée informatique afin de garder une trace de son occurrence.

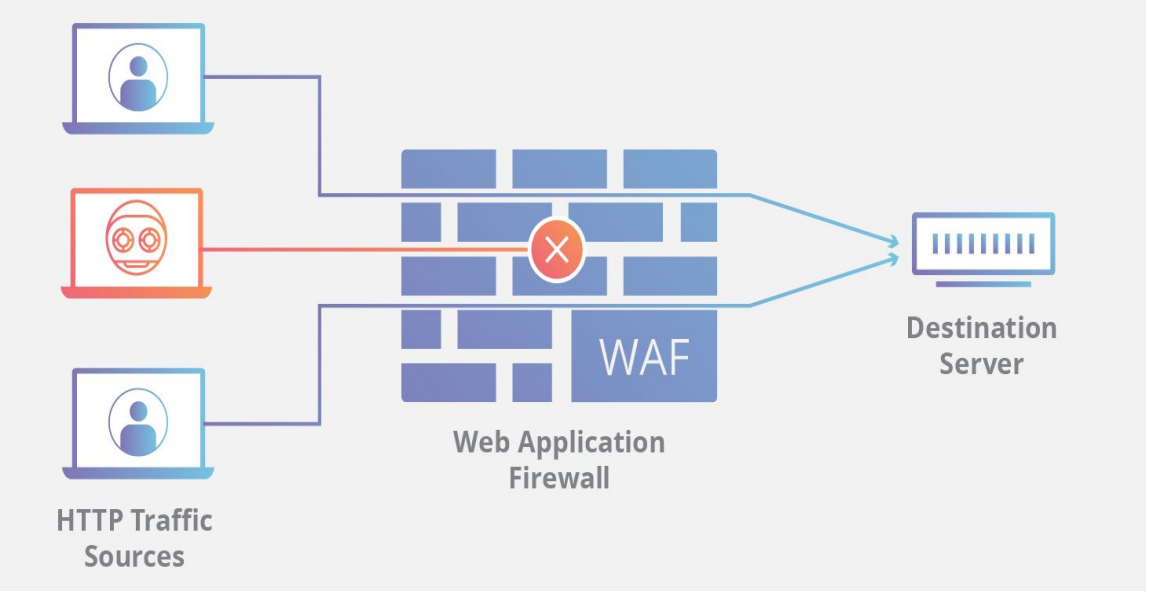
(5) Traffic Padding

Mécanisme qui génère de fausses informations sur le réseau pour rendre l'analyse du trafic plus difficile.

(6) Détection d'intrusions

La détection d'intrusions a pour but de repérer les activités suspectes ou anormales sur le réseau.

(7) Pare-feu (filtrage)



[7]

Figure 11 : Pare-feu d'applications web

Un pare-feu pour applications web (WAF) protège les applications web contre diverses attaques de la couche applicative, telles que le cross-site scripting (XSS), l'injection SQL et l'empoisonnement des cookies. L'utilisation d'un WAF permet de bloquer les attaques qui visent à exfiltrer les données. [13]

[7] réf :https://www.cloudflare.com

Le WAF protège les applications web en filtrant, surveillant et bloquant tout trafic HTTP/S malveillant circulant vers l'application web, et empêche toute donnée non autorisée de quitter l'application[13].

Les WAFs peuvent se présenter sous la forme d'un logiciel, d'un appareil ou d'un service. Les politiques peuvent être personnalisées pour répondre aux besoins uniques des applications web ou ensemble d'applications web.

B. Sécurité et contrôle d'accès

Le contrôle d'accès permet de restreindre l'utilisation du système, par le biais de l'attribution de permissions aux utilisateurs identifiés et authentifiés. Ainsi, à chaque utilisateur, est associé un profil, ce profil représente la liste des permissions qui lui sont attribuées. [13]

Le contrôle d'accès permet de répondre à quelques-uns des besoins de la sécurité à savoir : la confidentialité, l'intégrité et la disponibilité des données.[2]

1. Politique de contrôle d'accès

« L'ensemble des lois, règles et pratiques qui régissent la façon dont l'information sensible et les autres ressources sont gérées, protégées et distribuées à l'intérieur d'un système spécifique » ITSEC .

L'entreprise définie une politique de contrôle d'accès, par la suite , l'administrateur devra la mettre en place et s'occuper de la gestion des utilisateurs et leurs droits d'accès.

2. Concepts de base

Les concepts de base d'une méthode de contrôle d'accès sont les sujets, les objets et les actions.

Les sujets représentent les entités actives du système, c'est-à-dire des utilisateurs ou des processus. Ils accèdent aux objets, entités passives du système (données ou services) à travers l'utilisation de méthodes d'accès aux objets, c'est-à-dire les actions lecture , écriture, exécution….

3. Types de politiques de contrôle d'accès [2]

Nous allons introduire les trois principaux types de contrôle d'accès: DAC , MAC , RBAC .

a) Discretionary Access Control (DAC)

Trusted Computer System Evaluation Criteria (TCSEC) définit le Contrôle d'accès discrétionnaire comme étant : « Des moyens de limiter l'accès aux objets basés sur l'identité des sujets ou des groupes auxquels ils appartiennent. Le contrôle est discrétionnaire, car un sujet avec une certaine autorisation d'accès est capable de transmettre cette permission (peut-être indirectement) à n'importe quel autre sujet (sauf restriction du contrôle d'accès obligatoire). » [13]

Autrement dit, le DAC est un modèle de contrôle d'accès basé sur l'identité qui offre aux utilisateurs un certain contrôle sur leurs données. Les propriétaires des données (ou tout utilisateur autorisé à contrôler les données) peuvent définir des autorisations d'accès pour des utilisateurs ou des groupes d'utilisateurs spécifiques.

Les autorisations d'accès pour chaque élément de données sont stockées dans une liste de contrôle d'accès (ACL). Une ACL comprend les utilisateurs et les groupes qui peuvent accéder aux données et les niveaux d'accès qu'ils peuvent avoir.

Le contrôle d'accès basé sur le modèle DAC fonctionne de la manière suivante :

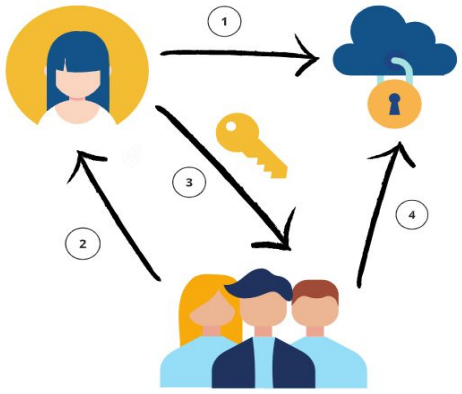


Figure 12 : Illustration du contrôle d'accès de type DAC

- (1) L'utilisateur 1 crée un fichier et devient son propriétaire ou obtient des droits d'accès à un fichier existant.
- (2) L'utilisateur 2 demande l'accès à ce fichier.

- (3) L'utilisateur 1 accorde l'accès à la ressource en question. Cependant, l'utilisateur 1 ne peut pas accorder des droits d'accès qui dépassent les siens. Par exemple, si l'utilisateur 1 ne peut que lire un fichier, il ne peut permettre à l'utilisateur 2 de le modifier.
- (4) S'il n'y a pas de contradiction entre l'ACL créée par l'administrateur et la décision prise par l'utilisateur 1, l'accès est accordé.

b) Mandatory Access Control (MAC) [19]

" means that access control policy decisions are made by a **central authority**, not by the individual owner of an object. User cannot change access rights. An example of MAC occurs in military security, where an individual data owner does not decide who has a top-secret clearance, nor can the owner change the classification of an object from top-secret to secret." NIST

"A means of **restricting access to system resources** based on the **sensitivity** (as represented by a label) of the information contained in the system resource and the formal authorization (i.e., clearance) of users to access information of such sensitivity." NIST

Avec le modèle MAC c'est le système qui fournit aux utilisateurs un accès basé sur la **confidentialité des données** et les **niveaux d'autorisation** des utilisateurs.

Le contrôle d'accès basé sur le modèle MAC fonctionne de la manière suivante :

- L'administrateur configure les politiques d'accès et définit les attributs de sécurité : niveaux de confidentialité, autorisations d'accès aux différents projets et types de ressources.
- L'administrateur attribue à chaque sujet (utilisateur ou ressource qui accède aux données) et objet (fichier, base de données, port, etc.) un ensemble d'attributs.
- Lorsqu'un sujet tente d'accéder à un objet, le système d'exploitation examine les attributs de sécurité du sujet et décide si l'accès peut être accordé.

Reprenons l'exemple évoqué dans la partie 'contôle d'accès de type DAC' , mais cette fois en utilisant le modèle MAC.

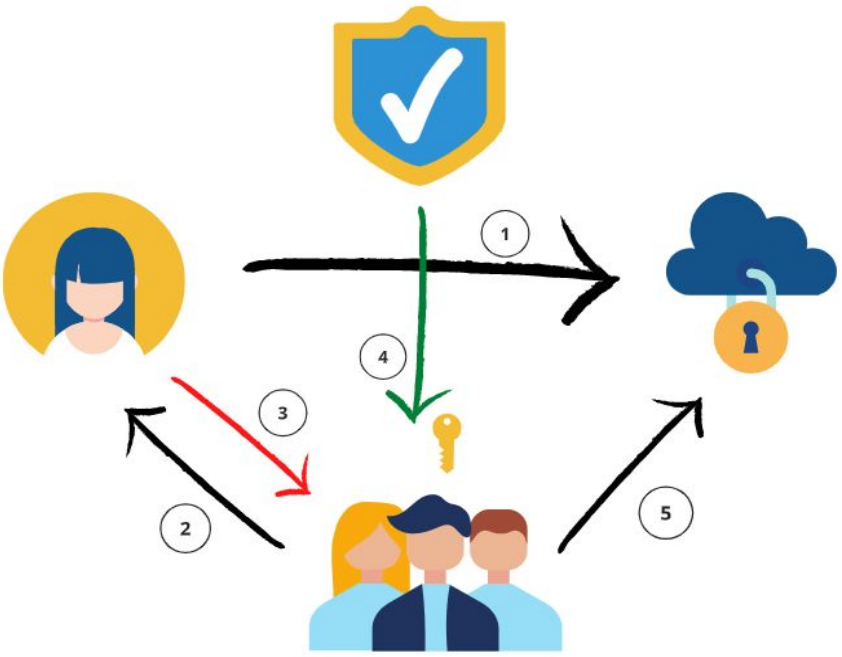


Figure 13 : Illustration du contrôle d'accès de type MAC

- (1) L'utilisateur 1 crée un fichier et devient son propriétaire.
- (2) L'utilisateur 2 demande l'accès à ce fichier.
- (3) L'utilisateur 1 ne peut accorder l'accès à la ressource en question.
- (4) Le système d'exploitation ou le noyau de sécurité donne accès à l'utilisateur 2 après avoir examiné les attributs de sécurité de ce dernier.
- (5) L'utilisateur 2 peut alors accéder à la ressource.

c) Role Based Access Control (RBAC) [19]

Le contrôle d'accès basé sur les rôles (RBAC) est une méthode de contrôle d'accès basée sur la définition des rôles des employés et des privilèges correspondants au sein de l'organisation. Le principe de ce modèle est que chaque employé se voit attribuer un rôle.

Chaque rôle comporte un ensemble d'autorisations et de restrictions. Un employé ne peut accéder à des objets et exécuter des opérations que si son rôle dans le système a les autorisations correspondantes.

Un utilisateur peut être affecté à un ou plusieurs rôles. Lors de l'ajout d'un nouvel employé, l'attribution d'un rôle est facile. Et lorsqu'une personne quitte l'entreprise, nul besoin de modifier les paramètres du rôle.

(1) Composants du modèle RBAC

Utilisateur - un individu ayant accès à un système.

Rôle - une fonction nommée (indique le niveau d'autorité).

Autorisation - droit d'accès aux ressources du système .

Session - une correspondance entre un utilisateur et un ensemble de rôles auxquels l'utilisateur est assigné dans le cadre d'un temps de travail.

Objet - une ressource du système dont l'accès nécessite une autorisation.

Opération - toute action dans le réseau protégé (lecture, écriture ,exécution …)

(2) Règles de base du modèle RBAC

- Un utilisateur ne peut exécuter une opération que si un rôle lui a été attribué.
- L'identification et l'authentification ne sont pas considérées comme des opérations.
- Toutes les activités de l'utilisateur sont effectuées par le biais d'opérations.

(3) Le modèle NIST-RBAC [19]

Le RBAC peut être mis en œuvre à quatre niveaux, selon le modèle NIST-RBAC.Chaque niveau suivant reprend les propriétés du précédent. Examinons-les :

(a) Flat RBAC

Le RBAC plat est une implémentation des fonctionnalités de base du modèle RBAC. Des rôles sont attribués à tous les utilisateurs. Les utilisateurs obtiennent les autorisations dont ils ont besoin en acquérant ces rôles. Il peut y avoir autant de rôles et d'autorisations que l'entreprise en a besoin. Un seul utilisateur peut se voir attribuer plusieurs rôles, et un rôle peut être attribué à plusieurs utilisateurs.

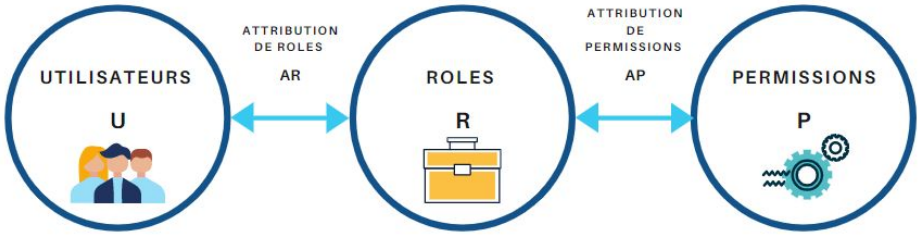


Figure 14 : Composants du modèle 'Flat RBAC'

(b) Hierarchical RBAC

Le RBAC hiérarchique, comme son nom l'indique, met en œuvre une hiérarchie au sein de la structure des rôles. Cette hiérarchie établit les relations entre les rôles. Les utilisateurs ayant des rôles supérieurs acquièrent les autorisations de tous les rôles inférieurs, qui sont attribués à leurs subordonnés. La complexité de la hiérarchie est définie par les besoins de l'entreprise.

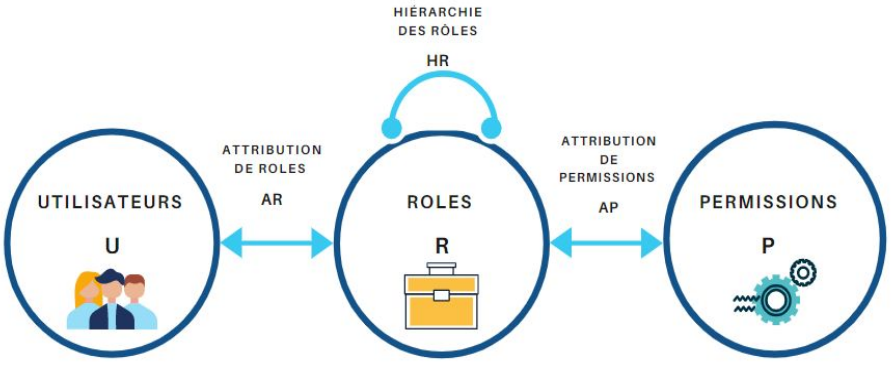


Figure 15 : Composants du modèle ' Hierarchical RBAC'

(c) Constrained RBAC

Le RBAC contraint ajoute une séparation des fonctions (SOD) à un système de sécurité. La **SOD** est une pratique de sécurité bien connue lorsqu'une seule tâche est répartie entre plusieurs employés. Elle est assez importante pour les moyennes et grandes entreprises.

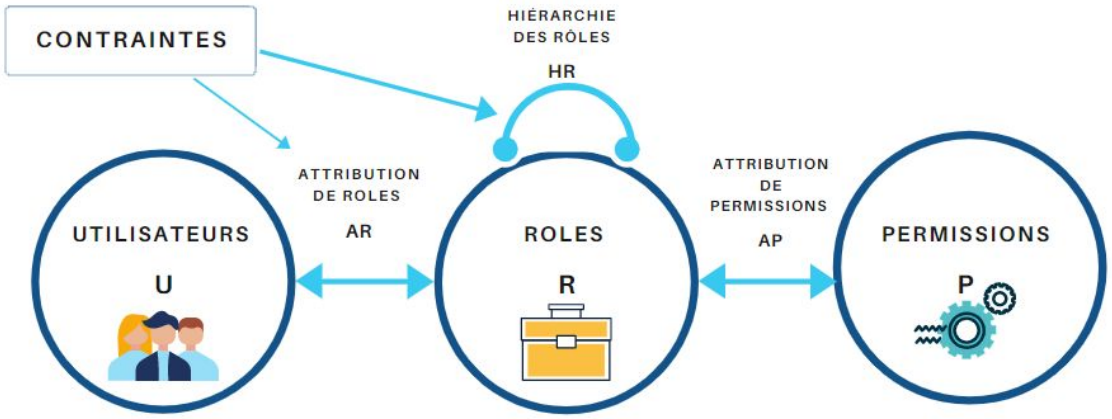


Figure 16 : Composants du modèle ‘Constrained RBAC’

(d) Symmetric RBAC

Permet d'identifier les autorisations attribuées aux rôles existants (et vice-versa). Par exemple, en identifiant les permissions d'un employé licencié, l'administrateur peut révoquer les permissions de l'employé et ensuite réattribuer le rôle à un autre utilisateur ayant le même ensemble de permissions ou un ensemble différent.

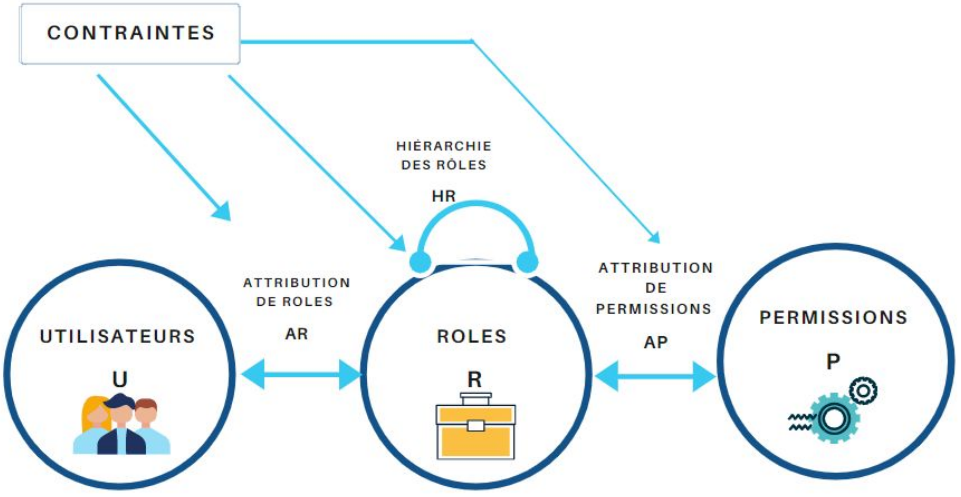


Figure 17 : Composants du modèle ‘Symmetric RBAC’

C. Sécurité des applications web

Une application web est une application hébergée sur un serveur et accessible à travers un navigateur web et dont l'utilisation ne requiert pas d'installation sur les machines clientes. Les applications web sont programmées avec les mêmes technologies que les sites web tout en offrant une expérience utilisateur et des fonctionnalités similaires à celles des logiciels.

1. Architecture d'une application web [1]

L'architecture d'une application web repose sur trois couches essentielles.

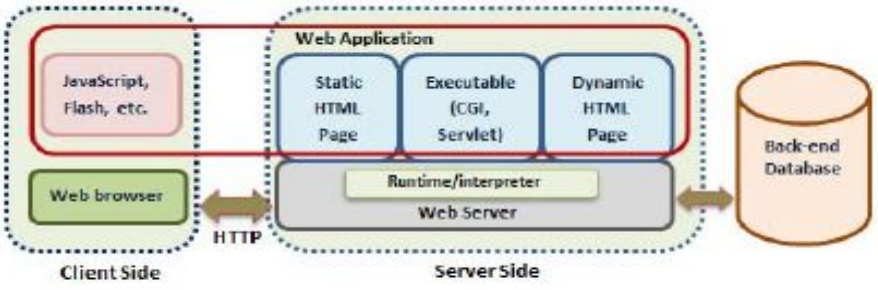
La première couche étant le côté **client** : On y trouve un navigateur web dont le rôle est d'afficher le contenu des pages web aux utilisateurs leur permettant ainsi d'interagir avec les fonctions des applications web.

Le côté client est généralement développé en utilisant du HTML, CSS et JavaScript.

La seconde couche est le côté **serveur** : utilisé , pour générer le contenu dynamique des pages, il est généralement développé en utilisant des technologies et langages de programmation tels que Python, PHP, Java, .NET, Ruby ou encore Node.js

La troisième est la **base de données** : Elle est utilisée pour le stockage des données.

La figure ci-dessous illustre l'architecture d'une application web.



[8]

Figure 18 : Architecture d'une application web

[8] réf : https://ieeexplore.ieee.org

2. Risques liés à la sécurité des applications [1]

L'insécurité des applications web est due à plusieurs facteurs .

Tout d'abord, le web a évolué vers des plateformes d'applications complexes. Malheureusement, les problèmes de sécurité ont augmentés.

Deuxièmement, les pirates informatiques ciblent de plus en plus les applications web depuis que les éditeurs de logiciels sont devenus plus vigilants et écrivent un code sécurisé et veillent à distribuer des correctifs pour contrer les formes traditionnelles d'attaques.

Troisièmement, les technologies de sécurité actuelles, y compris les réseaux, les pare-feu et les logiciels antivirus fournissent une protection sûre au niveau de l'hôte et du réseau, mais pas au niveau applicatif.

Les applications deviennent la cible d'attaques lorsque les réseaux et les points d'entrée au niveau de l'hôte deviennent relativement sûrs.

Open web application Security Project (OWASP) ou le projet de sécurité des applications web ouvertes, en français, est une fondation à but non lucratif qui travaille sur l'amélioration de la sécurité des logiciels.[16]

L'OWASP a répertorié les failles de sécurité les plus répondues dans les applications web afin d'aider les entreprises à améliorer la sécurité de leurs applications web et à réduire les vulnérabilités de ces dernières.

Liste des 10 failles de sécurité des applications web les plus courantes [16]

Risque	**Description**
Injection	Une faille de type injection, comme une injection SQL, une injection de commande système, ou une injection LDAP, se produit quand une donnée non fiable est envoyée à un interpréteur en tant qu'élément d'une commande ou d'une requête. Les données hostiles de l'attaquant peuvent duper l'interpréteur afin de l'amener à exécuter des commandes fortuites ou accéder à des données non autorisées.
Authentification de mauvaise qualité	Les fonctions applicatives liées à l'authentification et à la gestion des sessions sont souvent mal implémentées, ce qui permet aux attaquants de compromettre des mots de passe, des clés ou des jetons de session, ou encore d'exploiter d'autres failles d'implémentation pour usurper temporairement ou définitivement l'identité d'autres utilisateurs.

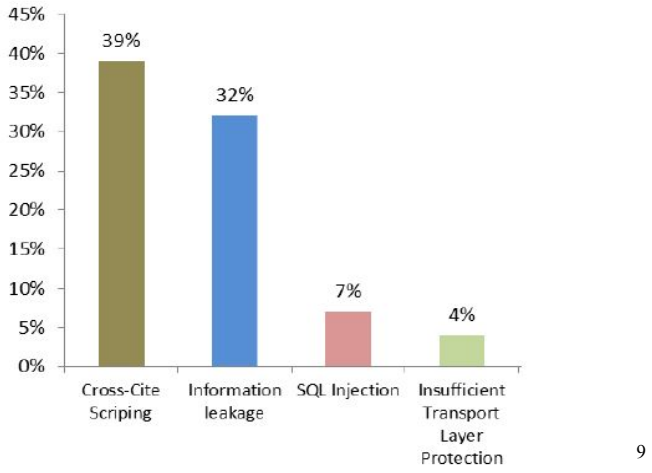
Exposition de données sensibles	Beaucoup d'applications web et d'APIs ne protègent pas correctement les données sensibles (données bancaires,données personnelles d'identification). Les pirates peuvent alors intercepter ou modifier ces données mal protégées pour effectuer une usurpation d'identité par exemple. Les données sensibles méritent une protection supplémentaire ,ainsi que des précautions particulières lors de l'échange avec le navigateur.
XML External Entities (XXE)	De nombreux processeurs XML, anciens ou mal configurés, évaluent les références aux entités externes dans les documents XML. Les entités externes peuvent être utilisées pour divulguer des fichiers internes à l'aide du gestionnaire d'URI, pour faire des partages de fichier interne, pour faire de l'analyse interne de ports, pour exécuter du code à distance, pour faire des attaques par déni de service.
Violation de contrôle d'accès	Les restrictions sur les droits des utilisateurs authentifiés sont souvent mal appliquées. Les attaquants peuvent exploiter ces failles pour accéder à des fonctionnalités et/ou des données non autorisées. Par exemple, accéder aux comptes d'autres utilisateurs, visualiser des fichiers sensibles, modifier les données d'autres utilisateurs, modifier les droits d'accès, etc.
Mauvaise configuration Sécurité	La mauvaise configuration de la sécurité est le problème le plus répandu. C'est généralement le résultat de configurations par défaut non sécurisées, de configurations incomplètes ou ad hoc, d'en-têtes HTTP mal configurés et de messages d'erreur verbeux contenant des informations sensibles. Non seulement tous les systèmes d'exploitation, frameworks, bibliothèques et applications doivent être configurés de façon sécurisée, mais ils doivent également être corrigés et mis à jour en temps et en heure.
Cross-Site Scripting (XSS)	Les failles XSS se produisent lorsqu'une application inclut des données non fiables dans une nouvelle page Web sans les valider, ou encore lorsqu'elle met à jour une page Web existante avec des données fournies par l'utilisateur à l'aide d'une API de navigateur permettant de créer du HTML ou du JavaScript. Cette attaque permet aux attaquants d'exécuter des scripts dans le navigateur de la victime et ainsi de récupérer les données des sessions utilisateurs, ou de rediriger l'utilisateur vers des sites malveillants.
Désérialisation non sécurisée	La désérialisation non sécurisée conduit souvent à l'exécution de code arbitraire à distance. Même si les failles de désérialisation n'entraînent pas l'exécution de code distant, elles peuvent être utilisées pour effectuer d'autres attaques, notamment des attaques par injection.

Utilisation de composants présentant des vulnérabilités connues	Les composants, tels que les bibliothèques, frameworks et autres modules logiciels, fonctionnent avec les mêmes privilèges que l'application. Si un composant vulnérable est exploité par une attaque, cela peut entraîner de graves pertes de données ou la compromission du serveur. Les applications et les API utilisant des composants dont les vulnérabilités sont connues peuvent compromettre leurs défenses et permettre diverses attaques.
Journalisation & Surveillance insuffisantes	La plupart des études de failles montrent que le temps nécessaire pour détecter une faille est supérieur à 200 jours. Généralement, cette faille est détectée par des parties externes plutôt que par une surveillance ou des processus internes.

Tableau 3 : Classement OWSAP des failles de sécurité les plus courantes [16]

3. Les vulnérabilités communes des applications web

Selon le Web Application Security Consortium, les vulnérabilités les plus répondues sont le Cross-Site Scripting, la fuite d'informations, les injections SQL, et la protection insuffisante de la couche transport. La figure 19 montre le pourcentage des vulnérabilités communes des applications web. Dans cette section, nous allons décrire brièvement chaque type de vulnérabilité ainsi que les mécanismes de protection. [1]



[9]

Figure 19 : Les 4 vulnérabilités des applications web les plus répondues

[9] réf : https://ieeexplore.ieee.org

a) SQL injection

L'injection SQL est une technique de piratage web exploitant une faille de sécurité présente dans une application interagissant avec une base de données.

Une attaque par injection SQL consiste à insérer ou "injecter" une requête SQL via les données d'entrée du client dans l'application.

Une SQLi (SQL injection) réussie permet à l'attaquant de lire des données sensibles à partir de la base de données, modifier les données qui s'y trouvent (Insérer/ Mettre à jour / Supprimer) , exécuter des opérations d'administration sur la base de données (telles que l'arrêt du SGBD), l'attaquant pourra également , récupérer le contenu d'un fichier donné présent sur le système de fichiers du SGBD et dans certains cas envoyer des commandes au système d'exploitation.[16]

Mécanismes de protection

- Validation des données saisies par l'utilisateur
- Limitation des privilèges
- Cacher les informations du message d'erreur
- Garder les informations d'identification de la base de données séparées et cryptées
- Utiliser un ORM

b) Cross Site Scripting (XSS)

Le cross-site Scripting est une faille de sécurité des sites web qui permet d'injecter un code malveillant dans une page web , provoquant des actions sur les navigateurs web utilisés pour visiter la page, parce qu'ils pensent que le script provient d'une source fiable, le script malveillant peut accéder aux cookies, jetons de session ou autres informations sensibles conservées par le navigateur et utilisées avec ce site[16]. Ces scripts peuvent même réécrire le contenu de la page HTML. Des failles dans le développement des applications web permettent à ces attaques d'être exécutées principalement lorsqu'il y a un échange de données entre les utilisateurs et les serveurs. Il existe trois formes de XSS, qui visent généralement les navigateurs des utilisateurs : XSS réfléchi ,XSS stocké, DOM XSS.[13]

Mécanismes de sécurité et prévention :

Le développeur doit prendre en considération l'aspect sécuritaire des applications web, voici donc quelques règles [16] à suivre :

- Utiliser des frameworks implémentant des techniques de protection contre les attaques XSS tel que React JS. Il faut aussi penser à lister les limites des mécanismes de protection implémentés par ces Framework et trouver des solutions pour les prendre en charge.
- Appliquer des techniques d'échappement aux données des requêtes HTTP non sûres.

L'utilisateur doit lui aussi adopter les bonnes pratiques afin de se protéger des attaques XSS.

- Choisir le bon navigateur : les attaquants ont tendances à cibler les navigateurs les plus utilisés.
- Booster la sécurité du navigateur : Il existe certains programmes et outils pour améliorer la sécurité des navigateurs tels que NoScript,GoogleToolbar...
- Désactiver certaines fonctionnalités : JavaScript, Java, Active X, JScript, VBScript, Flash et QuickTime sont tous potentiellement dangereux.
- Eviter de cliquer sur les liens inclus dans des emails, se méfier des URL trop longs et faire attention aux liens issus de l'utilisation de link shorteners.

c) Information Leakage [16]

La fuite d'informations permet à une application de révéler des données sensibles tels que les détails techniques de cette dernière, des commentaires de développeurs, l'environnement ou encore des données spécifiques à l'utilisateur. Ces données sensibles peuvent ensuite être utilisées par un attaquant afin d'exploiter l'application cible, son réseau d'hébergement ou ses utilisateurs.

Les informations sensibles peuvent être présentes dans les commentaires HTML, les messages d'erreur, le code source ou simplement laissées à la vue de tous. Bien que la fuite d'informations ne représente pas nécessairement une faille de sécurité, elle donne à un attaquant des indications utiles pour son exploitation future.

Mécanismes de prévention [16]

- Classification des données traitées, stockées ou transmises par l'application.
- Identifier les données sensibles nécessitant d'être protégées.
- Appliquer des contrôles selon la classification.
- Eviter de stocker les données sensibles lorsqu'il n'est pas nécessaire de le faire.

- Choisir les bons algorithmes et générer des clés robustes.
- Chiffrer l'ensemble des données transmises avec des protocoles sécurisés.
- Veiller à désactiver le cache pour les réponses contenant des données sensibles.
- Utiliser des fonctions de hachage puissantes lors du stockage des mots de passe.

d) Protection insuffisante de la couche transport (Insufficient Transport Layer Protection) [16]

Elle est définie comme étant une faiblesse de la sécurité causée par une application ne prenant aucune mesure pour protéger le trafic du réseau.

Durant l'authentification, les applications ont tendance à utiliser TLS/SSL. Mais souvent elles ne l'utilisent pas autre part dans l'application, mettant en périple la sécurité de certaines données de l'application ainsi que les ID de session, rendant ainsi l'interception de ces dernières plus facile pour l'attaquant. L'application devient donc vulnérable aux attaques.

SSL et TLS [16]

Secure Socket Layer (SSL) est le protocole original qui a été utilisé pour assurer le cryptage du trafic HTTP, sous la forme de HTTPS. Deux versions publiques de SSL existaient- les versions 2 et 3 présentent de graves faiblesses cryptographiques, ces deux version ne devraient plus être utilisées. Pour diverses raisons, la version suivante du protocole à savoir (SSL 3.1) a été appelée Transport Layer Security (TLS) version 1.0. Par la suite, les versions 1.1, 1.2 et 1.3 de TLS ont été publiées. Les termes "SSL", "SSL/TLS" et "TLS" sont utilisés de manière interchangeable et, dans de nombreux cas, "SSL" est utilisé pour désigner le protocole TLS, le plus moderne.

Mécanismes de prévention [16]

- Appliquer une couche de cryptage séparée à toute donnée sensible avant de la transmettre au canal SSL, quand cela est possible.
- Utiliser SSL/TLS pour protéger la couche transport et donc améliorer la sécurité de son application web.
- L'activation du drapeau "sécurisé" pour les cookies sensibles.
- L'utilisation d'algorithmes SSL puissants tels que ceux conformes à la norme FIPS 140-2.
- S'assurer qu'un certificat de serveur est valide et qu'il correspond correctement à tous les domaines pour lesquels il est utilisé.

Conclusion

Dans ce chapitre, nous avons commencé par introduire quelques concepts de base de la sécurité informatique.

Ensuite, nous avons abordé la notion de contrôle d'accès, qui permet de restreindre l'utilisation du système, par le biais de l'attribution de permissions aux utilisateurs identifiés et authentifiés.

Les applications web présentent des failles de sécurité, nous avons donc cité les vulnérabilités communes de ces dernières à savoir le cross scripting, la fuite d'informations, les injections SQL et la protection insuffisante de la couche transport.

Enfin, nous avons évoqué divers mécanismes de prévention à implémenter lors de la programmation des applications web.

Ce qui suit est axé sur l'intégration du modèle de contrôle d'accès RBAC dans les diagrammes UML afin de concevoir l'application qui permettra d'administrer et sécuriser la plate-forme ERP.

Chapitre 3 :
Analyse et conception

Introduction

L'objectif de ce chapitre est d'aborder les concepts essentiels à la réalisation du projet. À travers la première partie de ce chapitre, nous ferons un rappel sur les généralités d'UML.

La seconde partie sera consacrée à l'analyse : nous identifierons les principaux utilisateurs du système et les besoins de ces derniers.

Enfin, la troisième partie sera axée sur la conception : les diagrammes de classes ainsi que les diagrammes de séquence et la structure de la base de données seront présentés.

A. Généralités sur L'UML

1. UML

UML est un acronyme qui signifie Unified Modeling Language (langage de modélisation unifié). En termes simples, l'UML est une approche moderne de la modélisation et de la documentation des logiciels. En fait, c'est l'une des techniques de modélisation des processus d'entreprise les plus populaires.

Elle est basée sur des représentations schématiques de composants logiciels.
En utilisant des représentations visuelles, nous sommes en mesure de mieux comprendre les éventuels défauts ou erreurs des logiciels.

2. Diagrammes UML

Un diagramme UML permet de représenter visuellement un système avec ses principaux acteurs, rôles, actions, classes, afin de mieux comprendre, modifier, maintenir ou documenter les informations sur le système.

À travers l'utilisation de diagrammes UML, il est possible d'établir une représentation du logiciel sous plusieurs angles (vues). [18]

Vue des cas d'utilisation : vue des acteurs (besoins attendus)
Vue logique : vue de l'intérieur (satisfaction des besoins)
Vue d'implantation : dépendances entre les modules
Vue des processus : dynamique du système
Vue de déploiement : organisation environnementale du logiciel

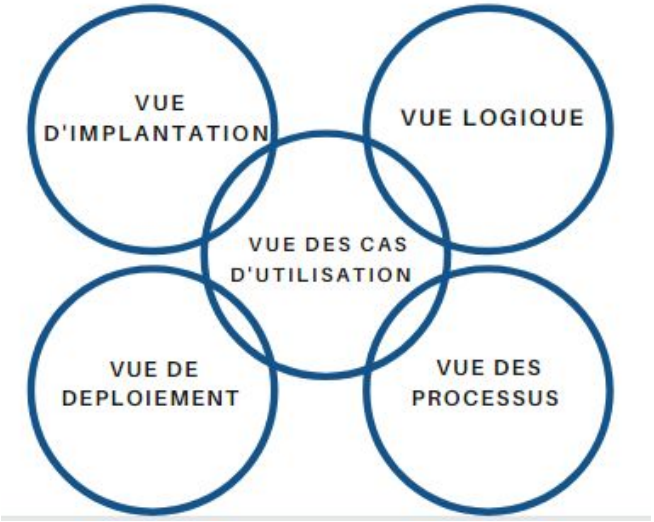


Figure 20 : Les vues UML

Il existe plusieurs types de diagrammes UML et chacun d'entre eux sert un objectif différent, qu'il soit conçu avant la mise en œuvre ou après (dans le cadre de la documentation).

Les trois catégories les plus larges qui englobent tous les autres types sont : la catégorie des diagrammes UML comportementaux, diagrammes UML structurels et diagrammes UML d'interaction. Comme leur nom l'indique, certains diagrammes UML tentent d'analyser et de représenter la structure d'un système ou d'un processus, tandis que d'autres décrivent le comportement du système, de ses acteurs et de ses composants.

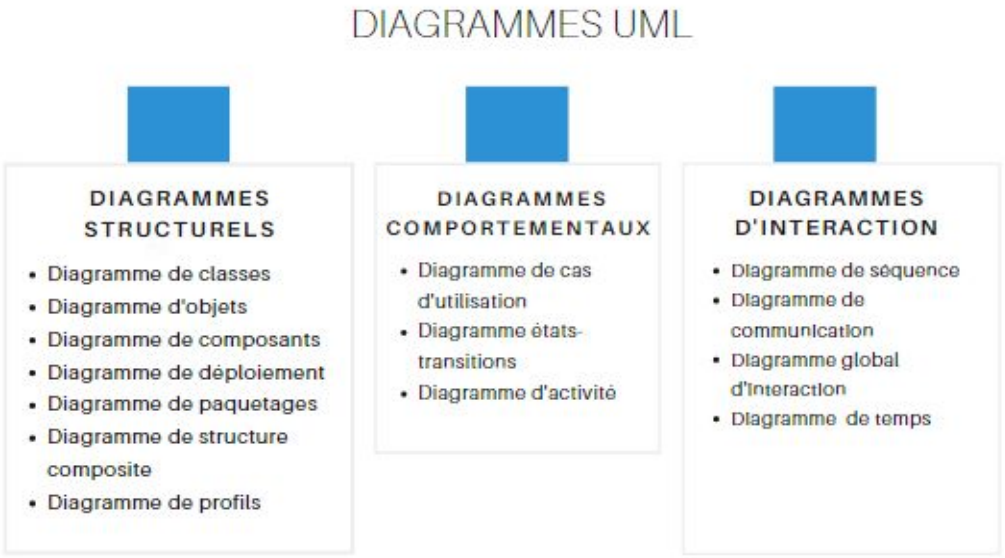


Figure 21 : Les diagrammes UML

Les 14 différents types de diagrammes UML ne sont pas tous utilisés régulièrement pour documenter des systèmes et/ou des architectures. [18]
Les diagrammes les plus fréquemment utilisés dans le développement de logiciels sont les suivants les diagrammes de **cas d'utilisation**, les **diagrammes de classes** et les **diagrammes de séquence.**

3. Diagramme de cas d'utilisation

Les exigences fonctionnelles auxquelles le système répond constituent la pierre angulaire du système. Les diagrammes de cas d'utilisation sont utilisés pour analyser les exigences du système. Ces exigences sont exprimées à travers différents cas d'utilisation. On remarque trois composantes principales de ce diagramme :

Les exigences fonctionnelles : représentées sous forme de cas d'utilisation ; un verbe décrivant une action

Acteurs : ils interagissent avec le système ; un acteur peut être un être humain, une organisation ou une application interne ou externe

Relations entre les acteurs et les cas d'utilisation : représentées par des flèches droites.

4. Diagramme de classes

Le diagramme de classe est le type de diagramme le plus courant pour la documentation de logiciels.L'utilisation de diagrammes de classes pour documenter les logiciel implantés avec la POO est très fréquente du fait que la programmation orientée objet soit basée sur les classes et les relations entre elles.[18]

Les diagrammes de classes contiennent les classes, ainsi que leurs attributs (également appelés champs de données) et leurs comportements (également appelés fonctions). Plus précisément, chaque classe a 3 champs : le nom de la classe en haut, les attributs de la classe juste en dessous du nom, les opérations/comportements de la classe en bas. La relation entre les différentes classes (représentée par une ligne de connexion), constitue un diagramme de classe.

5. Diagramme de séquence [18]

Comme leur nom l'indique, les diagrammes de séquence décrivent la séquence des messages et des interactions qui se produisent entre les acteurs et les objets. Les acteurs ou les objets ne peuvent être actifs qu'en cas de besoin ou lorsqu'un autre objet veut communiquer avec eux. Toute communication est représentée de manière chronologique.

Eléments du diagramme de séquence :

- Acteurs du système
- Objets du système
- Messages (cas d'utilisation, appels d'opération)

Représentation

Les lignes verticales représentent la 'vie' de chaque entité , tandis que les lignes horizontales représentent les échanges de messages avec le système ou au sein du système.

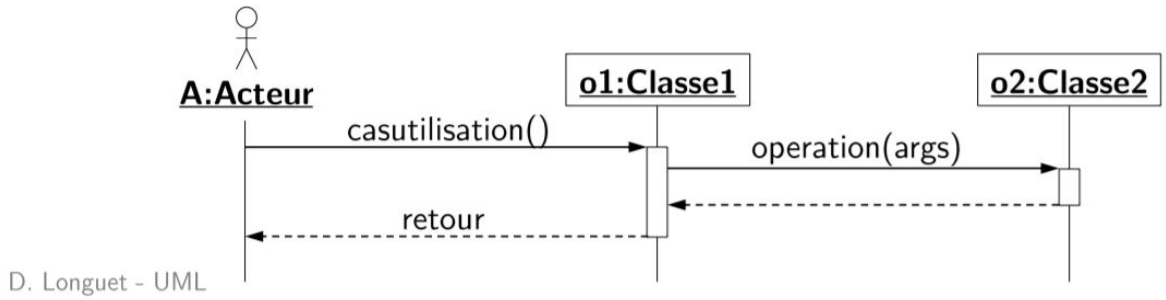


Figure 22 : Eléments du diagramme de séquence [18]

B. Analyse

L'analyse des besoins est une étape importante du cycle de développement d'un logiciel, elle nous permet de définir les besoins fonctionnels et non fonctionnels du client ainsi que les utilisateurs qui interagissent avec notre application.

1. Identification des acteurs

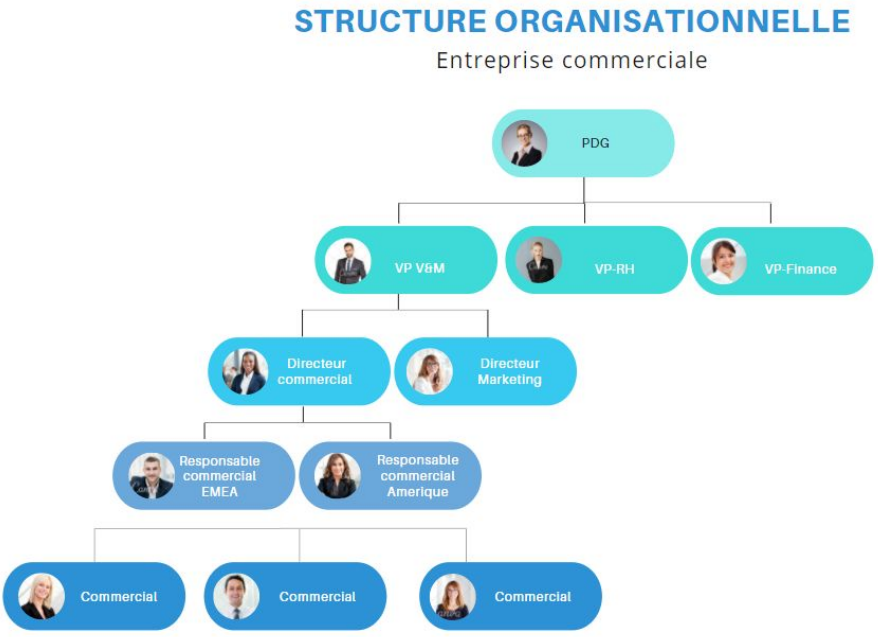


Figure 23 : Structure organisationnelle d'une entreprise commerciale

Le diagramme ci-dessus illustre la structure organisationnelle d'une entreprise commerciale. Pour mieux comprendre le rôle de chacun de ces employés, nous avons établi un tableau descriptif de ces rôles.

Rôle	Définition	Niveau d'accès
Commercial	"Un commercial est une personne dont le métier est lié à la vente. Avec un portefeuille de clients ou de clients potentiels, sur une zone géographique définie, il développe les ventes en respectant la politique commerciale définie par l'entreprise."[13]	Accès aux éléments du CRM
Responsable commercial	"Le responsable commercial doit tenir les objectifs commerciaux fixés et vendre les produits ou services proposés par l'entreprise. Il peut être en charge de la vente d'un ensemble de produits ou services différents, contrairement au chef des ventes qui est plus généralement en charge d'un seul produit ou service." [10]	Commercial + Interface de configuration du CRM
Directeur commercial	"Le directeur commercial d'une entreprise fait partie de l'encadrement supérieur d'un établissement à caractère économique. Il a pour fonction principale la mise en place d'une politique de vente ou de liquidation des stocks tels que les produits finis, les services."[13]	Responsable commercial + Logs et audits (partie commerciale)

[10] réf : https://www.ifcv.fr

VP Vente et Marketing	"Un vice-président, abrégé en VP, est une personne travaillant pour un gouvernement, une entreprise, une institution ou une association, et dont le rôle est d'occuper le poste du président lorsque celui-ci ne peut pas l'exercer." [13] Dans le cas de l'ERP nous pouvons assimiler ce rôle à l'administrateur d'un module particulier. Exemple la partie vente et marketing.	Directeur commercial + Directeur Marketing
PDG	"Le président-directeur général (PDG) est le dirigeant de plus haut rang dans une société."[13 Dans le cas de l'ERP nous pouvons assimiler ce rôle à l'administrateur de l'ERP. De ce fait,il possède un compte 'administrateur' qui lui permet d'avoir accès à l'ensemble des fonctionnalités de l'ERP.	VP vente et Marketing + VP des autres unités de l'entreprise. Autrement dit (Gestion de tous les utilisateurs , gestion des modules , logs et audits sur l'ensemble des composants de l'ERP)

Tableau 4 : Descriptif des rôles au sein d'une entreprise commerciale et des niveaux d'accès associés

Dans le cadre de notre travail, nous allons nous intéresser à la partie relative à la gestion de la relation client. Pour ce faire, nous allons implémenter une mini application CRM.

Lors de l'étape d'analyse et conception, nous focaliserons nos efforts sur les rôles suivants : ´PDG', ´Responsable commercial' et ´Commercial'. Les besoins de ces derniers seront détaillés dans ce qui suit.

2. Spécification des besoins

Besoins fonctionnels

Les rôles PDG , responsable commercial et commercial, illustrent la hiérarchie des rôles au sein d'une entreprise, de ce fait , chacun d'eux aura des besoins spécifiques selon son niveau de responsabilité.

Acteurs	Identification des besoins
PDG	● Gestion des utilisateurs ● Gestion des rôles ● Gestion des permissions ● Gestion des modules
Responsable commercial	● Gestion des clients ● Gestion des projets ● Gestion des transactions ● Paramètres de gestion des clients
Commercial	● Gestion des clients ● Gestion des projets ● Gestion des transactions

Tableau 5 : Descriptif des besoins fonctionnels

Besoins non fonctionnels

La performance : accès rapide à la base de données, temps de réponse court.
La sécurité : les données doivent être sécurisées (chiffrements des données).
La compatibilité : l'application web doit être compatible avec tous les navigateurs et sur tous les supports (ordinateur, tablette et mobile).
L'ergonomie : l'application doit présenter des interfaces graphiques cohérentes et conviviales avec des composants bien structurés.

3. Représentation des diagrammes de cas d'utilisation

a) Diagramme de cas d'utilisation global

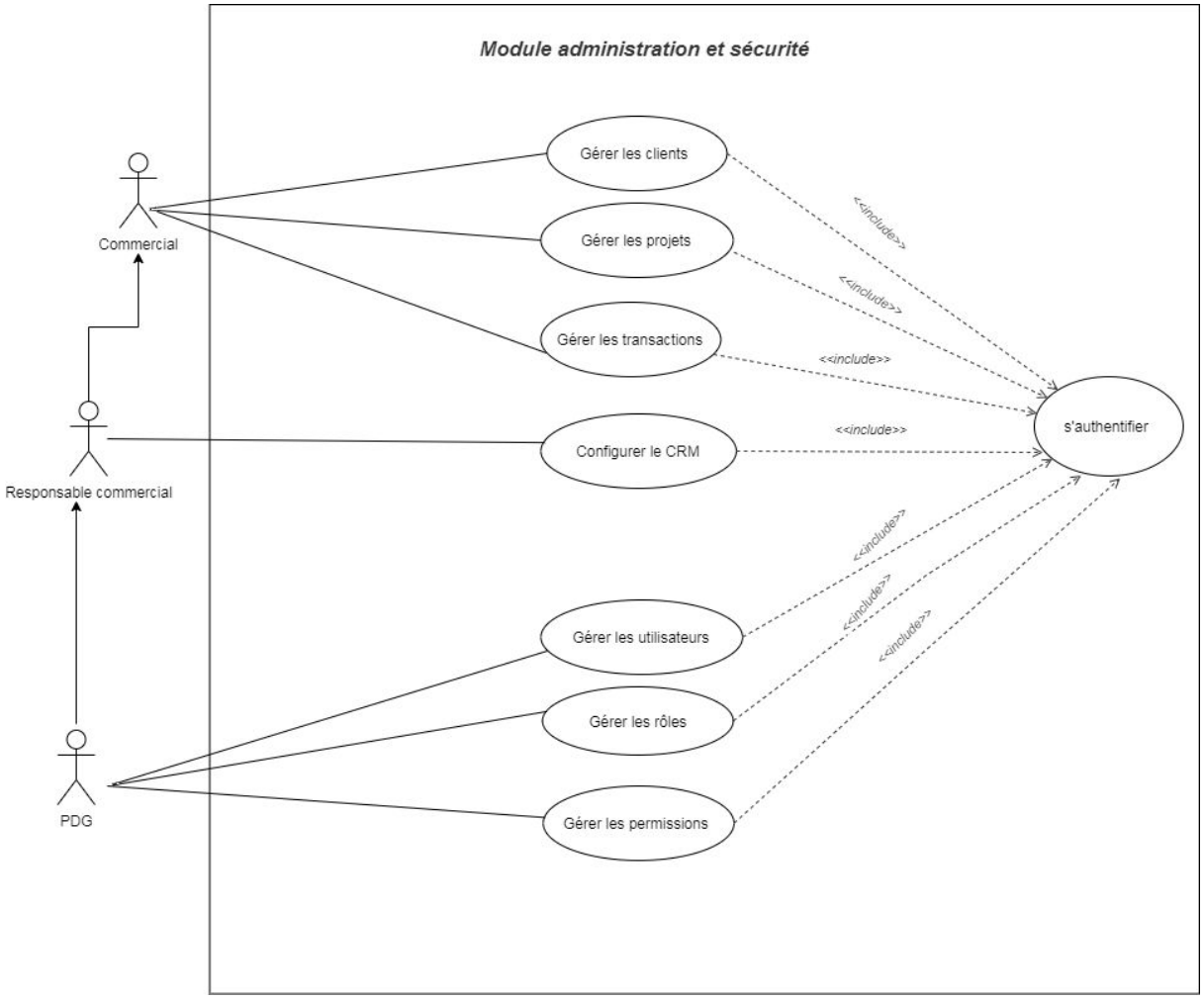


Figure 24 : Diagramme de cas d'utilisation global

b) Diagrammes des cas d'utilisation détaillés

(1) Cas d'utilisation 'PDG'

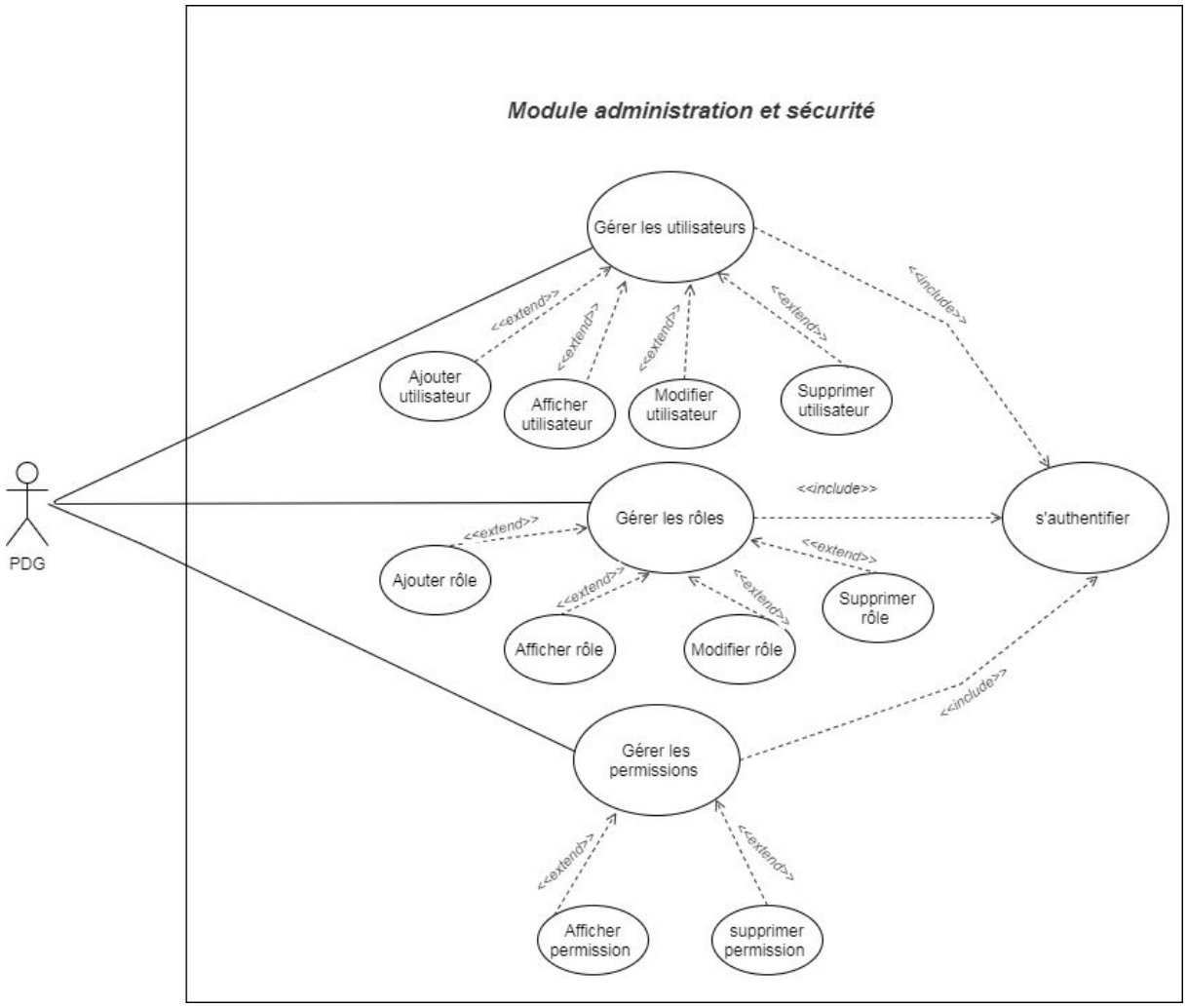


Figure 25 : Diagramme de cas d'utilisation 'PDG'

(2) Cas d'utilisation 'Responsable commercial'

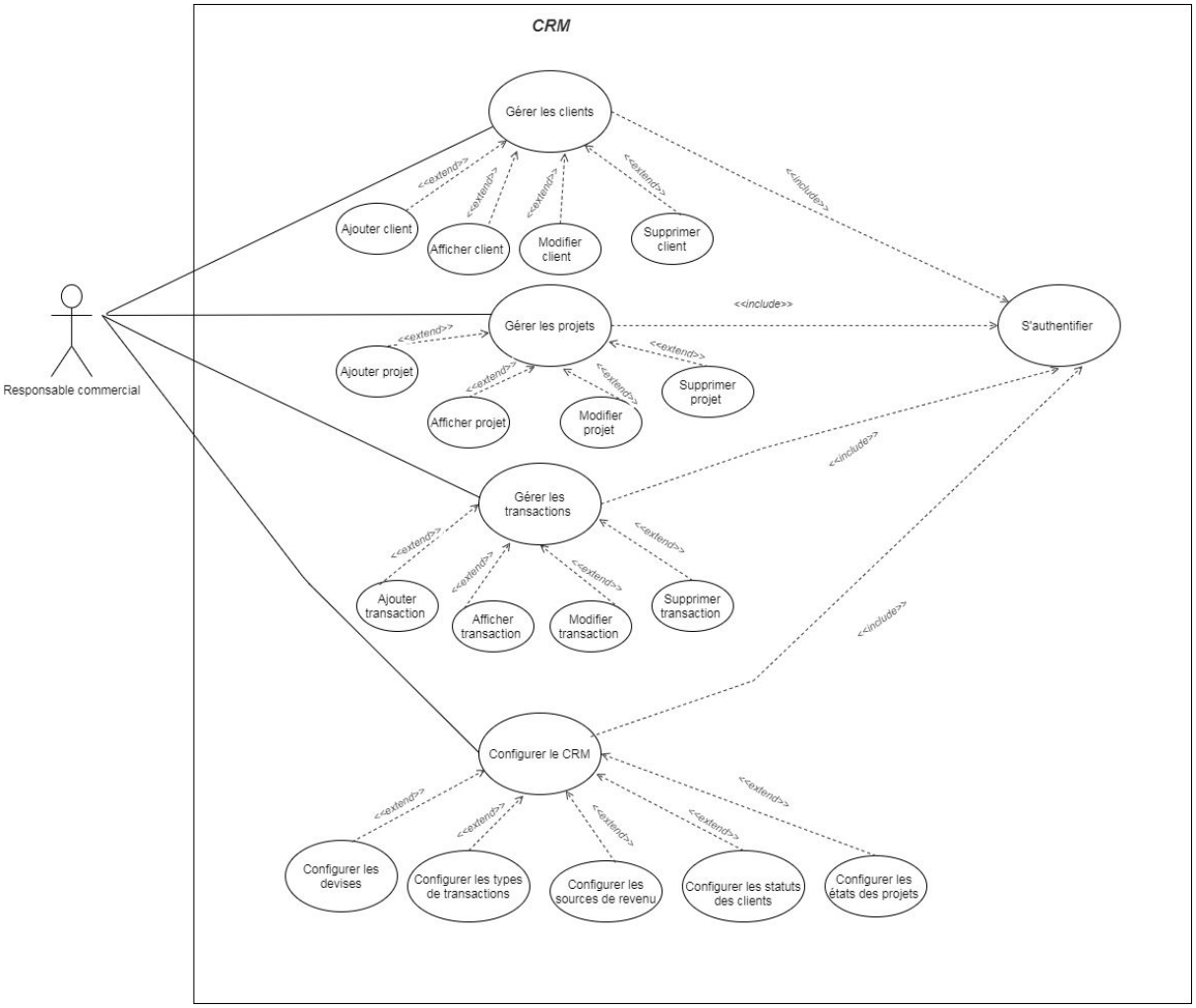


Figure 26 : Diagramme de cas d'utilisation 'Responsable commercial'

(3) Cas d'utilisation 'Commercial'

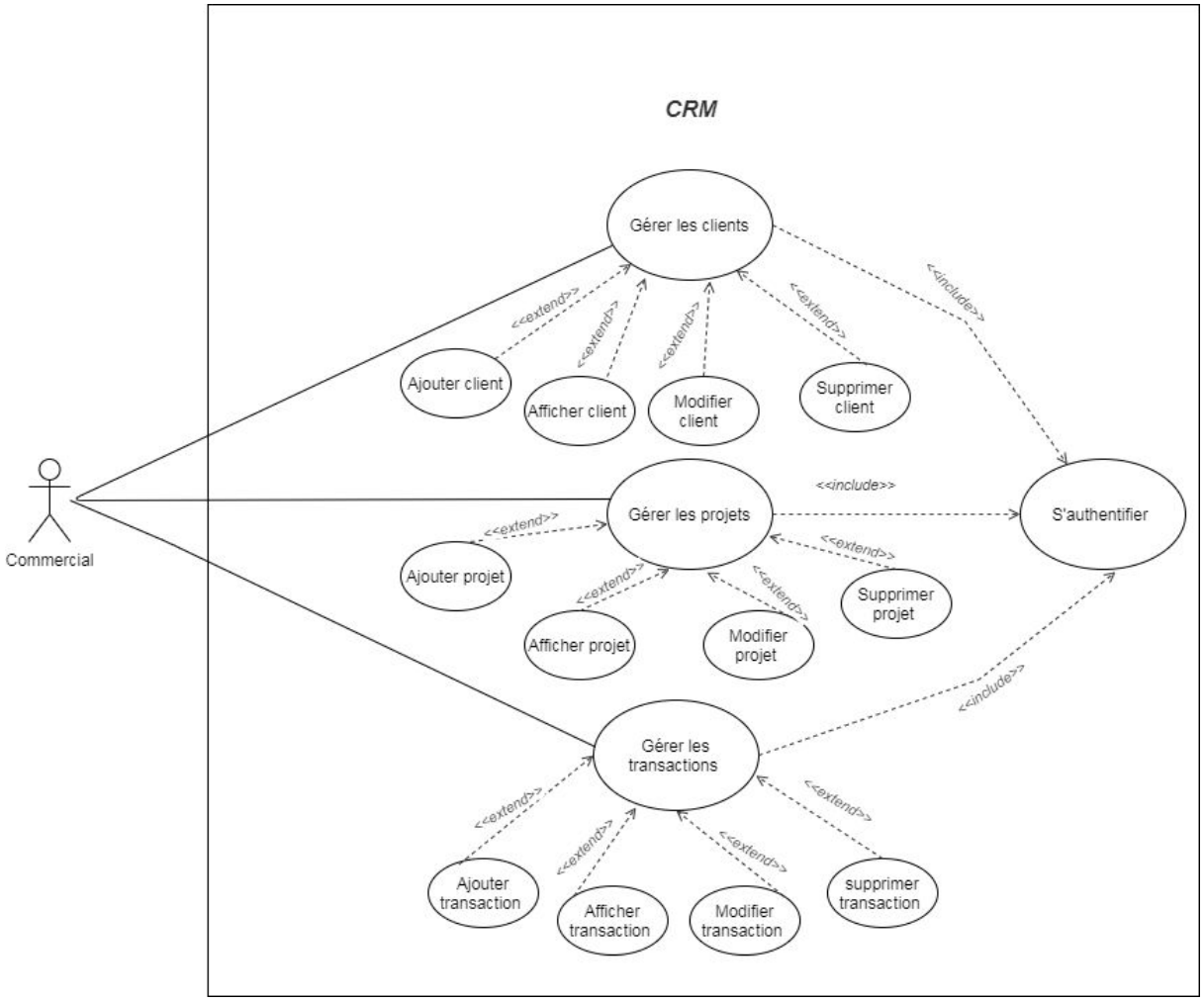


Figure 27 : Diagramme de cas d'utilisation 'Commercial'

C. Conception

Dans un premier temps nous présenterons quelques diagrammes de séquence et le diagramme de classe. Ensuite nous fournirons une description de la structure de quelques tables de la base de données.

1. Représentation des diagrammes de séquence

a) Diagramme de séquence du cas d'utilisation 'Modifier le mot de passe'

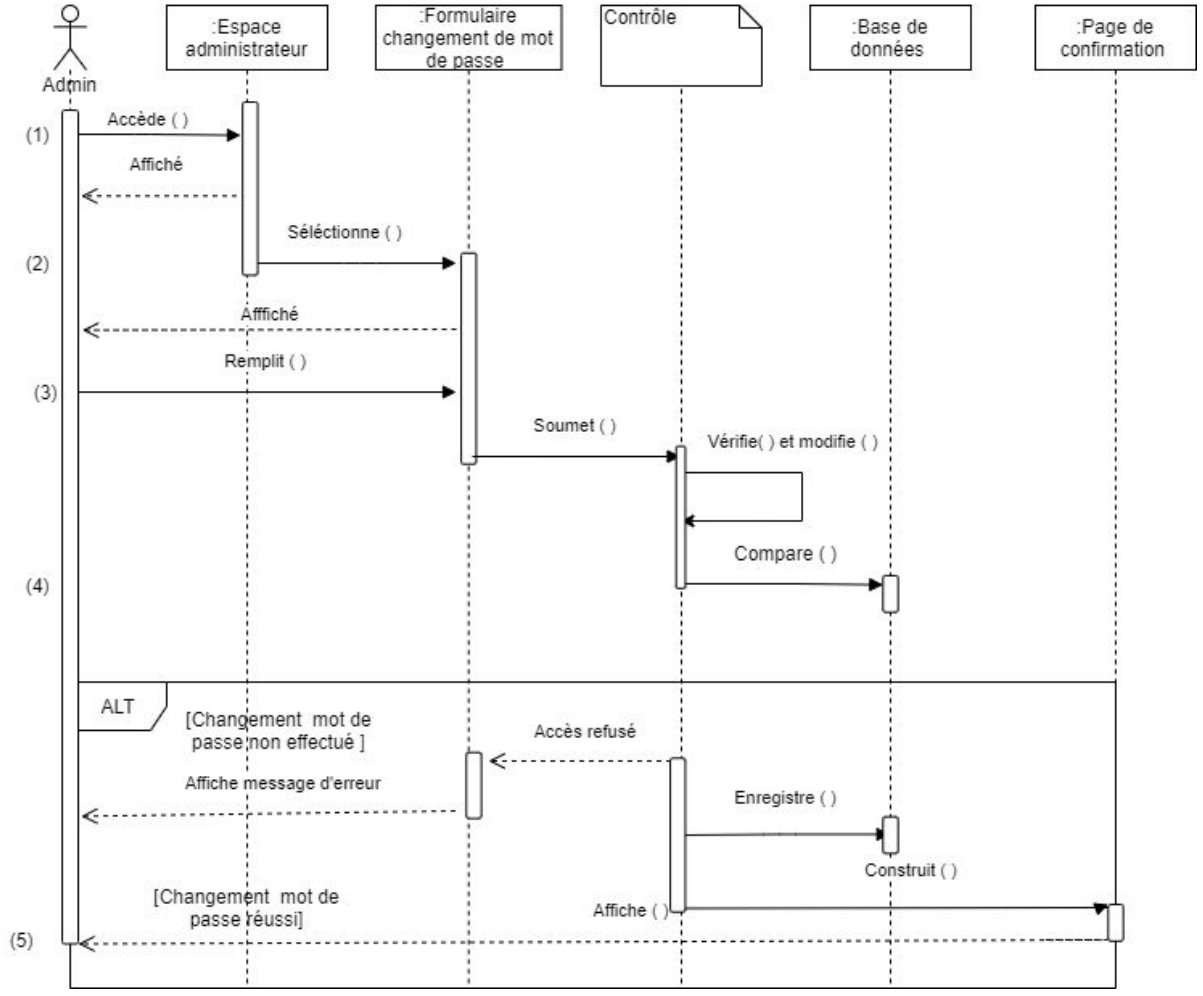


Figure 28 : Diagramme de séquence 'Changer le mot de passe'

- (1) L'administrateur accède à son espace personnel.
- (2) L'administrateur sélectionne le formulaire de changement de mot de passe.
- (3) L'administrateur remplit le formulaire et le soumet au contrôle.
- (4) Le système vérifie le nouveau mot de passe saisi et l'enregistre dans la base de données, il affiche un message d'erreur si l'ancien mot de passe est erroné.

b) Diagramme de séquence du cas d'utilisation 'Ajouter un utilisateur '

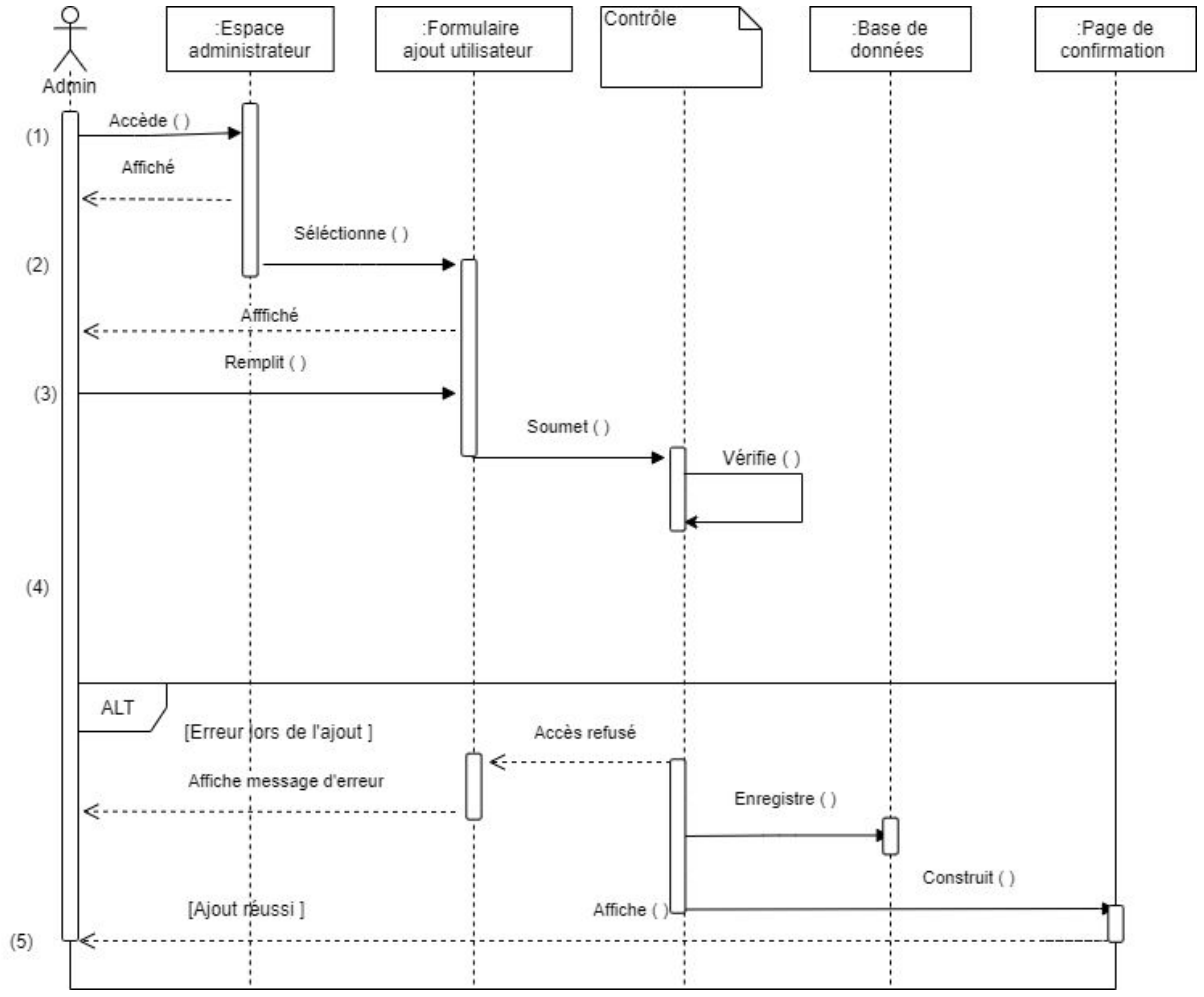


Figure 29 :Diagramme de séquence 'Ajouter un utilisateur'

- (1) L'administrateur accède à son espace personnel.
- (2) L'administrateur sélectionne le formulaire d'ajout d'un utilisateur.
- (3) L'administrateur remplit le formulaire et le soumet au contrôle.
- (4) Le système vérifie la validité des informations saisies et les enregistre dans la base de données, sinon il affiche un message d'erreur.
- (5) Le système affiche la page de confirmation.

c) Diagramme de séquence du cas d'utilisation 'Ajouter un rôle'

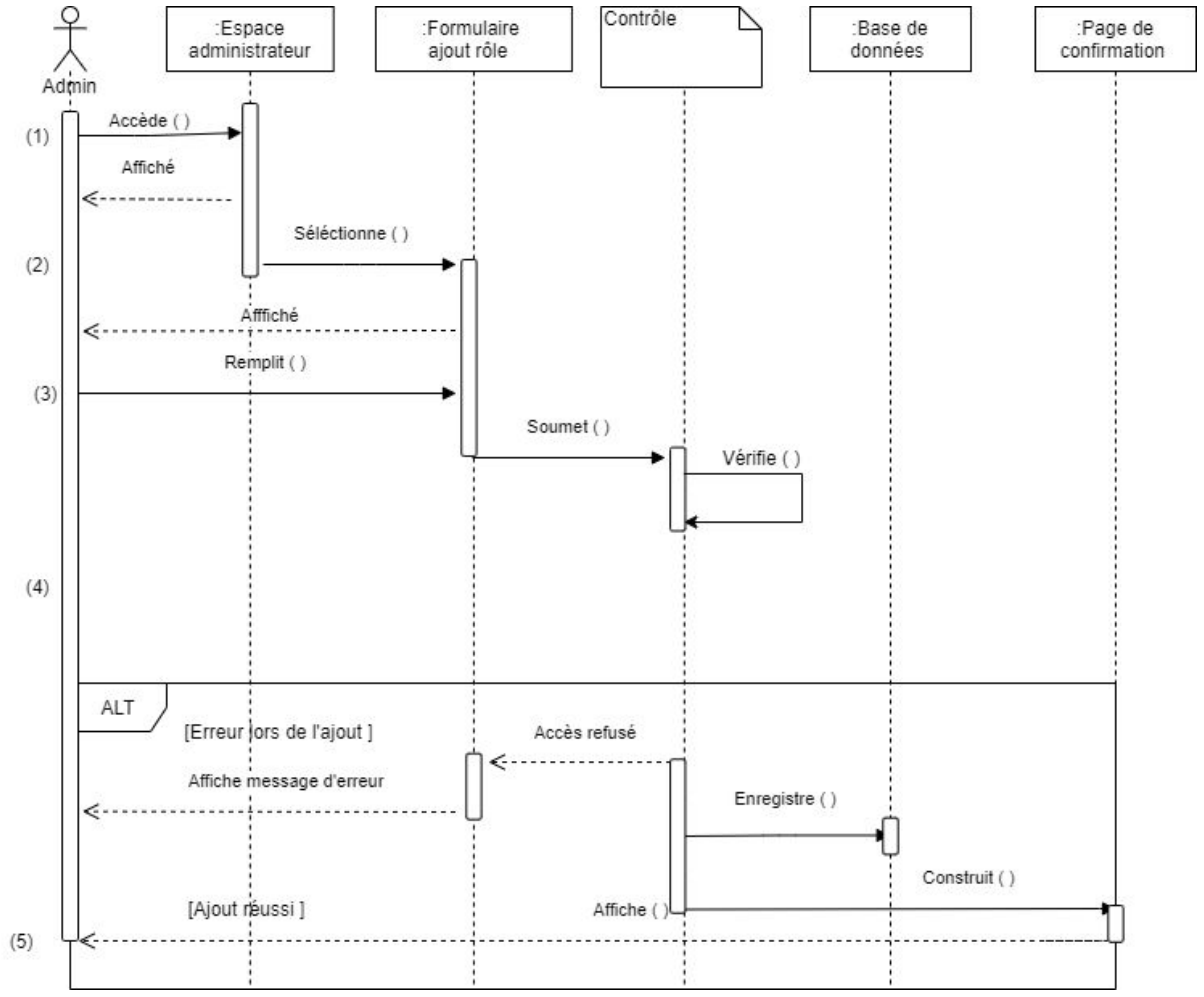


Figure 30 :Diagramme de séquence 'Ajouter un rôle'

- (1) L'administrateur accède à son espace personnel.
- (2) L'administrateur sélectionne le formulaire d'ajout de rôle.
- (3) L'administrateur remplit le formulaire et le soumet au contrôle.
- (4) Le système vérifie la validité des informations saisies et les enregistre dans la base de données, sinon il affiche un message d'erreur.
- (5) Le système affiche la page de confirmation.

2. Représentation du diagramme de classe

Utilisateurs

Un utilisateur est une personne qui se connecte au système.[2]

Rôles

La fonction ou titre de travail d'une personne dans une entreprise peut être appelé un "rôle". Mais un rôle a une définition plus technique dans le RBAC : il s'agit d'un ensemble clairement défini de capacités, ou d'autorisations, à utiliser dans les systèmes de l'entreprise. Chaque utilisateur interne se voit attribuer au moins un rôle, et certains peuvent avoir plusieurs rôles.[2]

Permissions

Dans le contexte du contrôle d'accès, une permission est la capacité d'effectuer une action sur un ou plusieurs objets du système.[2]

Sessions

La session représente une période de temps pendant laquelle l'utilisateur est connecté au système. L'utilisateur se voit attribuer un ou plusieurs rôles (exprimé à travers l'association entre rôles et utilisateurs), qui lui octroient des permissions spécifiques. Les permissions dont bénéficie l'utilisateur durant la session sont la somme des permissions des rôles activés durant cette dernière. [2]

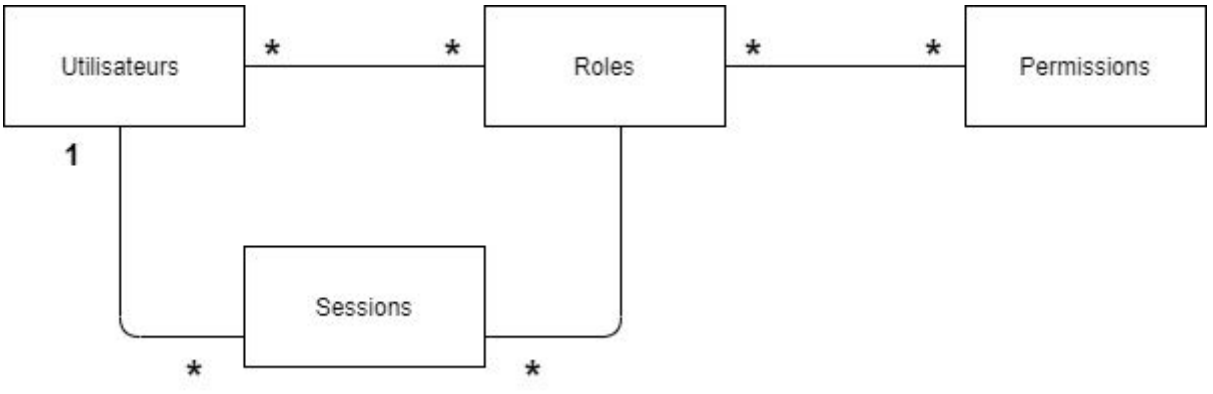


Figure 31 : Diagramme de classe du modèle RBAC

3. Structure de la base de données

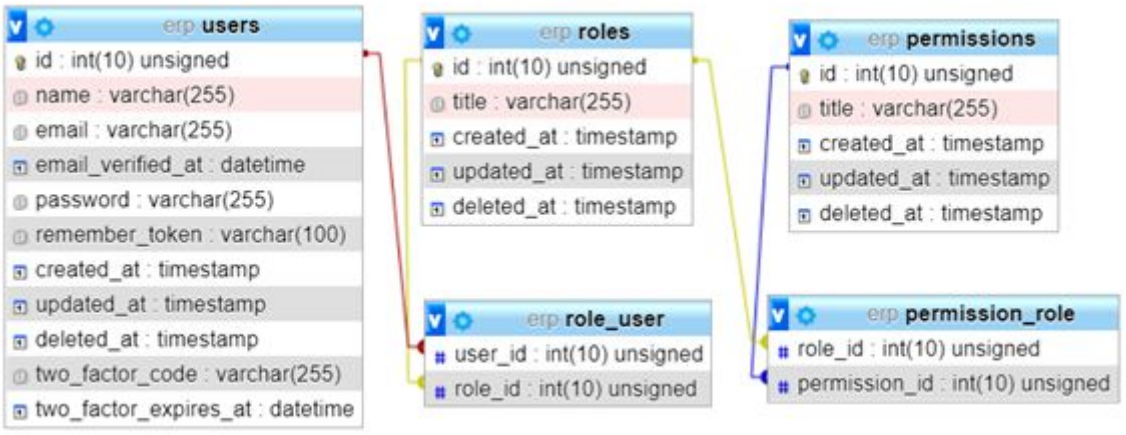


Figure 32 : Tables relatives à la partie d'administration

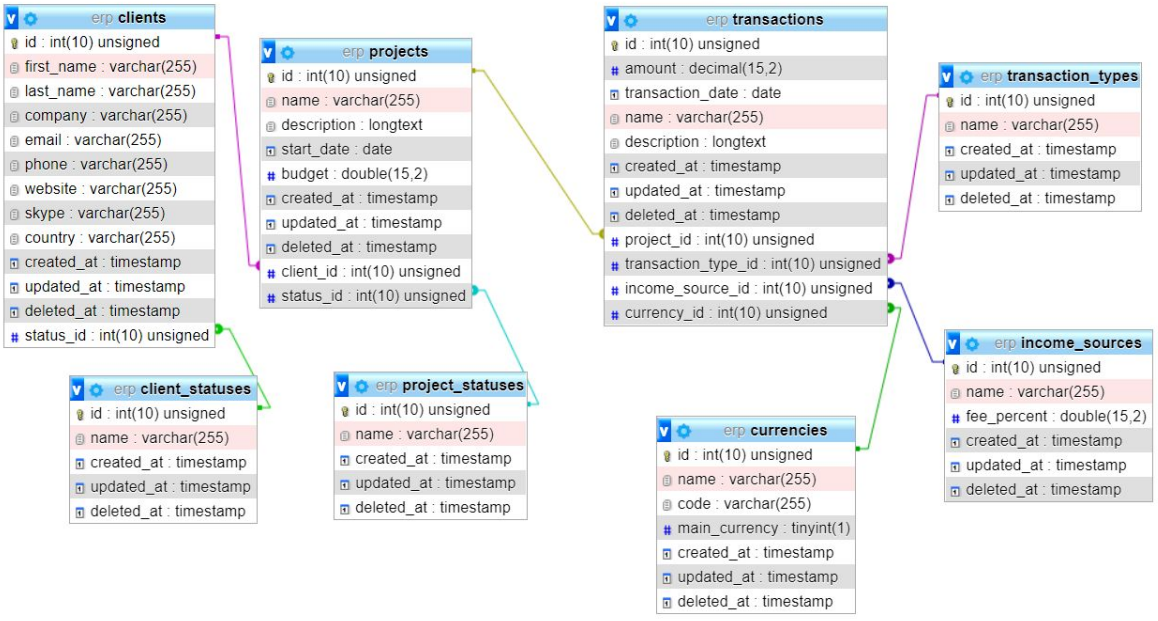


Figure 33 : Tables relatives à la partie CRM

Conclusion

Dans ce chapitre nous avons établi une description graphique du projet à travers l'utilisation du langage de modélisation UML et ses diagrammes.

D'abord, nous avons identifié les principaux rôles au sein de l'entreprise et nous en avons sélectionné trois (PDG, responsable commercial, commercial). À travers l'utilisation des diagrammes de cas d'utilisation, nous avons analysé les exigences du système.

Puis nous avons utilisé le diagramme de classes pour présenter les principales classes du système et les relations entre elles. Les diagrammes de séquence quant à eux, furent utilisés afin de décrire les principales interactions entre les acteurs et le système.

Enfin, après avoir analysé les besoins de l'utilisateur et conçu les diagrammes nécessaires, nous pouvons passer à l'étape suivante qui est la réalisation.

Chapitre 4 :
Implémentation et réalisation

Introduction

Dans ce chapitre nous allons décrire l'environnement de développement de l'application et présenter les différents outils utilisés lors de la réalisation de cette dernière.Dans un premier temps nous exposerons les technologies utilisées pour réaliser le frontend.

Par la suite, nous nous intéresserons aux technologies utilisées en backend.

Enfin nous partagerons quelques interfaces du module d'administration et sécurité de la plate-forme ERP.

A. Technologies de développement frontend :

1. HTML5

[11]

HyperText Markup Language (HTML) est un langage qui permet de créer des pages web et mettre en forme leur contenu en utilisant des balises, d'où le nom langage de balisage.

L'HTML5 fut introduit en 2014 par Word Wide Web Consortium (W3C) est la dernière version disponible à ce jour.

Certaines fonctionnalités n'étaient pas disponibles avec HTML 4.01 et nécessitent l'utilisation de plugins externes tels que 'Adobe Flash'.

Pour palier à ce problème des balises spécifiques ont été introduites : (<section>, <nav>, <article>, <aside>, <header>, <footer> ,<main>), des éléments prenant en charge le contenu multimédia :<audio> et <video>, mais aussi des APIs qui intègrent des objets JavaScript : Canvas , Drag-and-Drop , Offline-Applications, Géolocation.

[11] réf: https://fr.wikipedia.org

2. CSS3

[12]

Langage permettant de définir le design d'une page web html, séparant ainsi la présentation de la structure.

3. JavaScript

[13]

Langage de programmation de script, orienté objet, utilisé pour rendre les pages html interactives.

4. Jquery

[14]

Bibliothèque JavaScript regroupant un ensemble de fonctions permettant d'écrire plus facilement des scripts client-side.

---

[12] réf: https://fr.wikipedia.org

[13] réf : https://seeklogo.com

[14] réf:https://www.programmation-facile.com

5. Bootstrap 4

[15]

Est un framework front-end, utilisé pour développer des sites web ou des applications web responsives.
Bootstrap 5 alpha a été officiellement publié le 16 juin 2020.[17]
Les changements majeurs qu'apporte Bootstrap 5 :

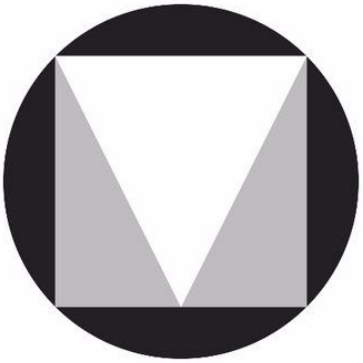
- Bootstrap ne dépend plus de jQuery .
- Le support d'Internet Explorer est supprimé.
- Bootstrap a introduit sa propre bibliothèque d'icônes SVG open-source.
- Tous les plugins JS sont toujours disponibles.
- Ajout de propriétés CSS personnalisées.
- Ils ont élargi leurs palettes de couleurs.

Les autres changements apportés à cette nouvelle version sont :

- Les composants des formulaires ont été mis à jour.
- La documentation a également été améliorée.
- Les contrôles de formulaire ont été redéfinis.
- Nouvelle API pour les services publics pour un meilleur contrôle
- Amélioration du système de grille.
- Nouveau niveau xxl.
- Classe verticale ajoutée
- .gutter remplacé par .g*
- Les colonnes n'ont plus de position : relative par défaut.

[15] réf: https://startbootstrap.com

6. Material Design



16

Langage visuel et interactif créé par Google. Il fut présenté pour la première fois en 2014 lors de l'événement annuel Google I/O.

Material Design fut introduit afin d'unifier le style graphique des applications et plateformes , en suivant les lignes directrices fixées concernant le choix des couleurs , polices ainsi que l'utilisation de mises en page basées sur une grille, des animations et des transitions, des effets de profondeur tels l'éclairage et les ombres, superposition des éléments etc …

Material Design peut être utilisé sur Android/ Web & IOS et vise à être accessible à tous, intuitif , interactif et efficace.

[16] réf: https://commons.wikimedia.org

B. Technologies de développement backend

1. Laragon

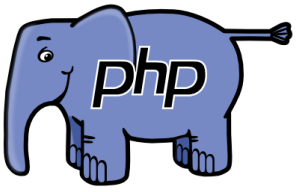
[17]

Laragon est un environnement de développement universel, portable, isolé, rapide et puissant pour PHP, Node.js, Python, Java, Go, Ruby. Il est rapide, léger, facile à utiliser et facile à étendre.

Laragon est idéal pour construire et gérer des applications web modernes. Il est axé sur la performance - conçu autour de la stabilité, de la simplicité et de la flexibilité.

Laragon est très léger. Il n'utilise pas les services de Windows car il possède sa propre orchestration de services qui gère les services de manière asynchrone et non bloquante.

2. PHP



[18]

Langage de programmation permettant le développement de pages web dynamiques.

3. MySQL

[19]

Système de gestion de base de données relationnelles, qui utilise SQL (langage qui permet d'écrire des requêtes pour communiquer avec la base de données).

---

[17] réf:https://laragon.org

[18] réf: https://www.php.net

[19] réf : https://fr.wikipedia.org

4. Apache server

[20]

Est un serveur http, très populaire du World Wide Word.Il permet de délivrer les pages web aux clients.

Plusieurs modules peuvent être pris en charge par Apache, permettant ainsi d'interpréter des langages de programmation tels que PHP , Python , Ruby et Perl. De plus, plusieurs fonctionnalités sont supportées : réécriture d'URL , protocoles de communications additionnels , common Gateway Interface (CGI), serveur proxy …

Le serveur Apache peut être configuré plus facilement en utilisant des interfaces graphiques.

5. PhpMyAdmin

[21]

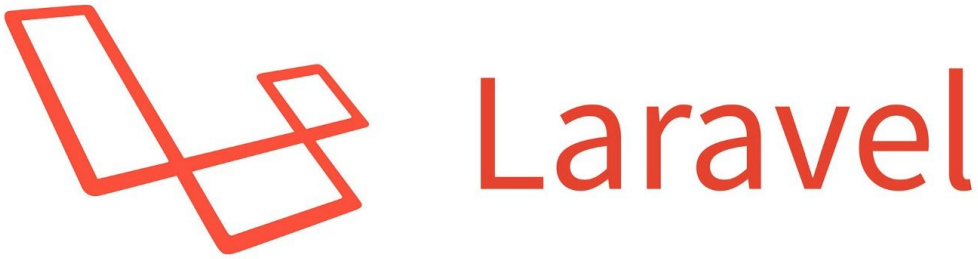
Outil d'administration de base de données utilisé avec le SGBD MySQL. [22] PhpMyAdmin, fut développé en 1998 par' The phpMyAdmin Project' , la dernière version disponible depuis le 21 mars 2020 est la version 5.0.2

Grâce à son interface intégrant un éditeur SQL et un constructeur de requêtes, il est possible d'effectuer des opérations CRUD sur les bases de données , mais aussi d'attribuer ou de révoquer les droits d'accès, et importer ou exporter les données contenues dans la base de données au format SQL, le tout sans écrire de requêtes SQL bien qu'il reste possible de le faire.

---

[20] réf: https://apache.org
[21] réf:https://www.phpmyadmin.net

6. Laravel [15]



Laravel est un framework d'application web à la syntaxe expressive et élégante développé par Taylor Otwell en juin 2011.

Laravel tente de rendre le développement d'applications moins fastidieux en facilitant les tâches courantes utilisées dans la majorité des projets web.

Laravel propose :

- Un moteur de routage simple et rapide.
- Puissant conteneur d'injection de dépendances.
- Stockage des sessions et du cache.
- Migration de schémas de base de données.
- Traitement robuste des tâches en arrière-plan.
- Diffusion d'événements en temps réel.

Ce framework PHP utilise l'architecture MVC pour développer des applications web.

La sécurité est une caractéristique importante lors de la conception d'applications web. Elle assure aux utilisateurs du site web que leurs données sont sécurisées. Laravel fournit différents mécanismes pour sécuriser les applications web. Certaines de ces fonctionnalités sont énumérées ci-dessous.

- Authentification
- Autorisation
- Vérification de l'adresse e-mail
- Cryptage
- Hachage
- Réinitialisation du mot de passe

C. Interfaces graphiques :

1. Interfaces d'authentification

La vérification en deux étapes ou two factor authentication (2FA), est une technique d'authentification qui nécessite une étape supplémentaire comparée à la méthode classique d'authentification.

L'authentification de base requiert un identifiant et un mot de passe.
Tandis que l'authentification en deux étapes nécessite un élément autre que le mot de passe de l'utilisateur après qu'il se soit authentifié. Ce second facteur vise à procurer une mesure de sécurité supplémentaire.

Plusieurs éléments peuvent être utilisés comme second paramètre exemple :
"une clé USB qui supporte le standard Universal Second Factor (U2F), une application utilisant le protocole Time-based One-time Password algorithm (en), une phrase secrète ou encore une empreinte digitale, etc."[13]

Dans le cadre de ce projet, l'authentification en deux étapes fonctionne de la manière suivante: le système demande à l'utilisateur de fournir ses identifiants de connexion (adresse e-mail et mot de passe) dans le cas où ces éléments sont valides,le système envoie un code de vérification à l'adresse e-mail de l'utilisateur, ce code est unique et expire au bout de dix minutes. En fait, le fait de fournir les bons identifiants de connexion ne suffira pas à accéder au système, il faudra impérativement que le code soit lui aussi valide afin de permettre l'accès au système.

L'application comprend deux principales interfaces graphiques pour l'authentification , l'une pour la première étape de connexion (Figure 34) au système et l'autre pour la seconde étape de connexion au système (Figure 35).
Ce n'est qu'après avoir saisi un code valide que l'utilisateur pourra accéder au système (Figure 36).

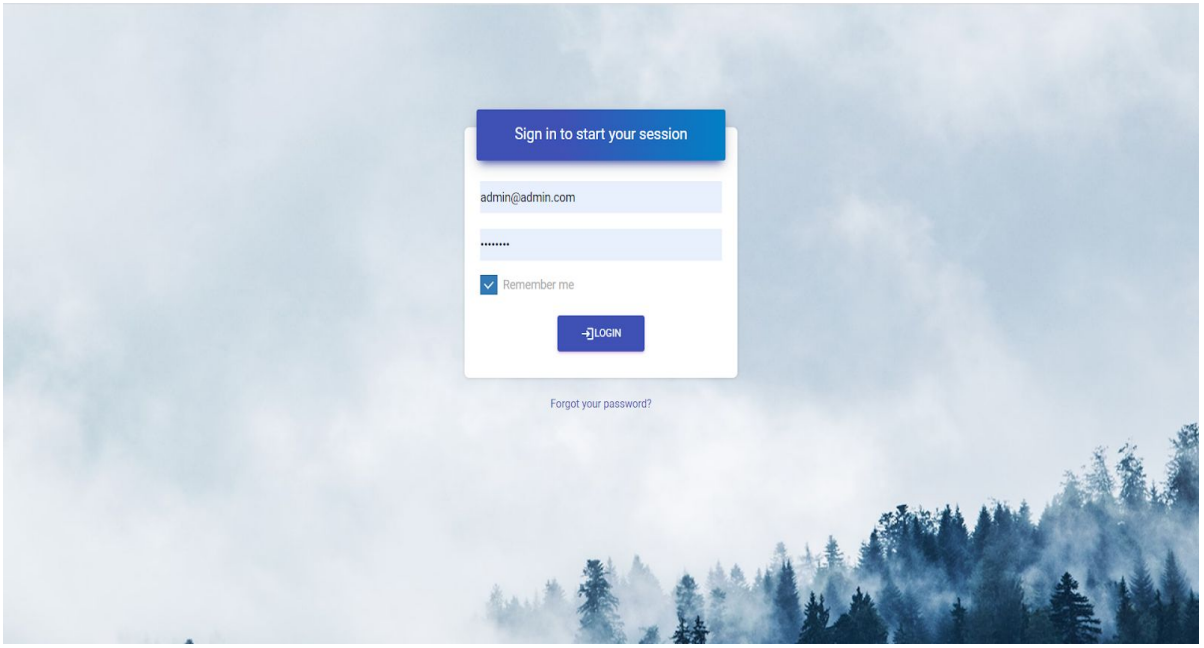


Figure 34 : Authentification

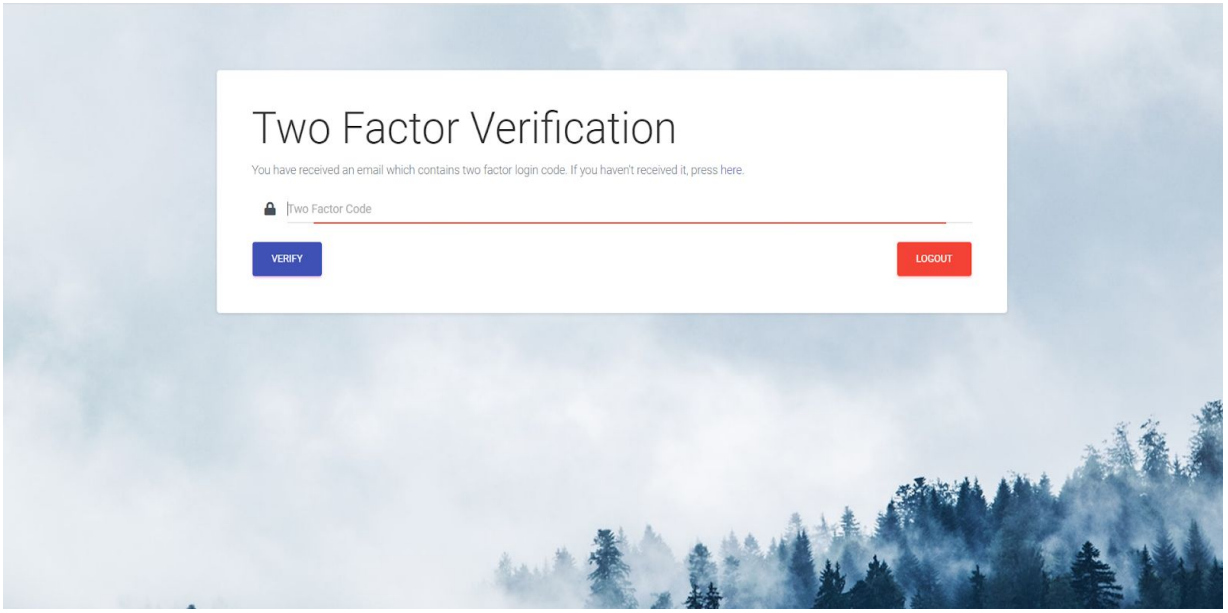


Figure 35 : Saisie du code de vérification

2. Interface d’accueil

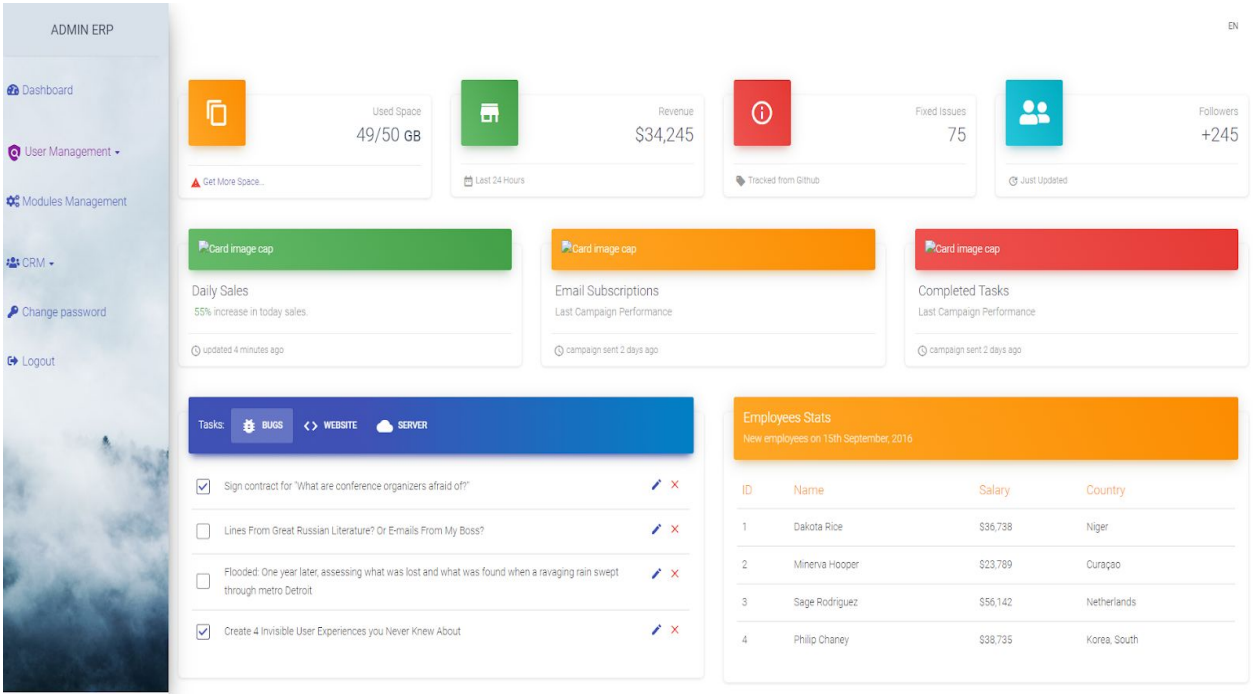


Figure 36 : Accueil

3. Interface ‘Liste des modules’

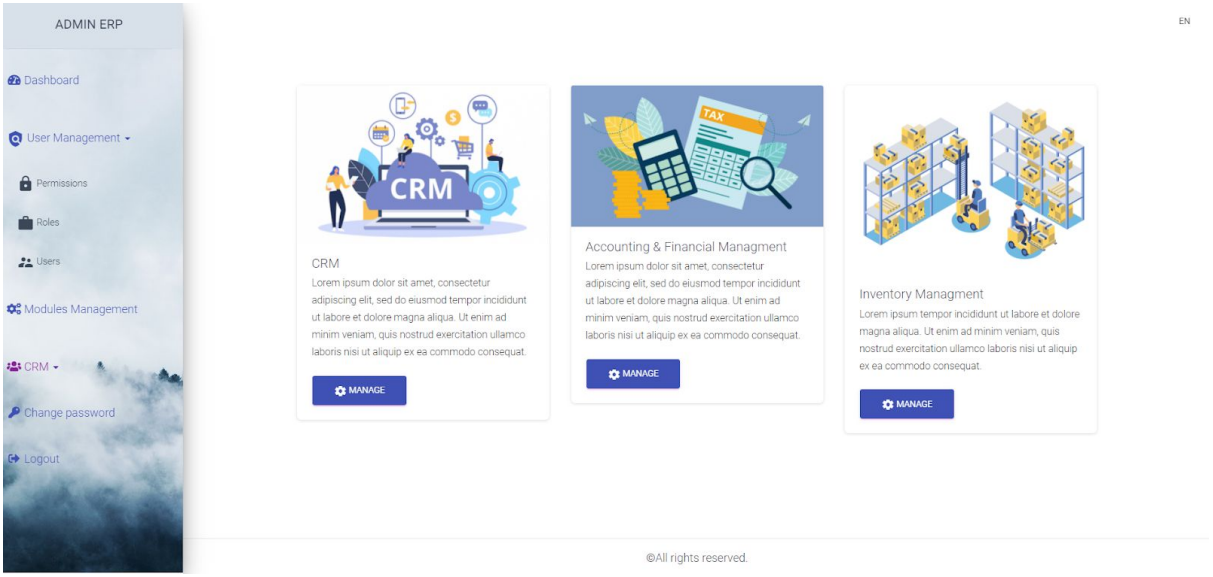


Figure 37 : Liste des modules

4. Interface ‘Liste des utilisateurs’

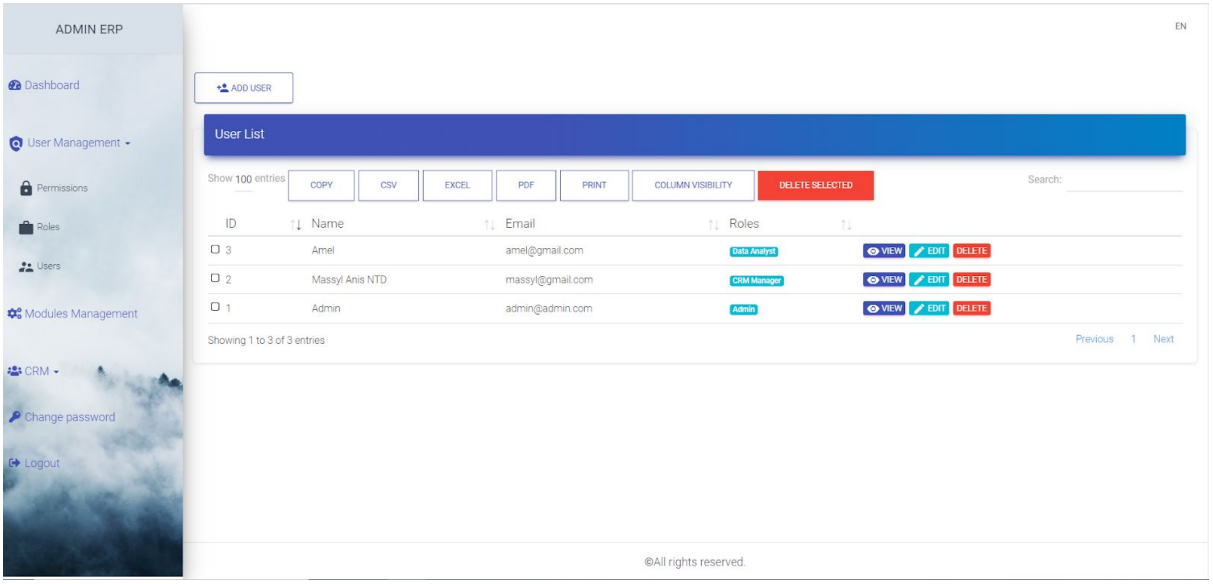


Figure 38 : Liste des utilisateurs

5. Interface ‘Ajout d’un utilisateur’

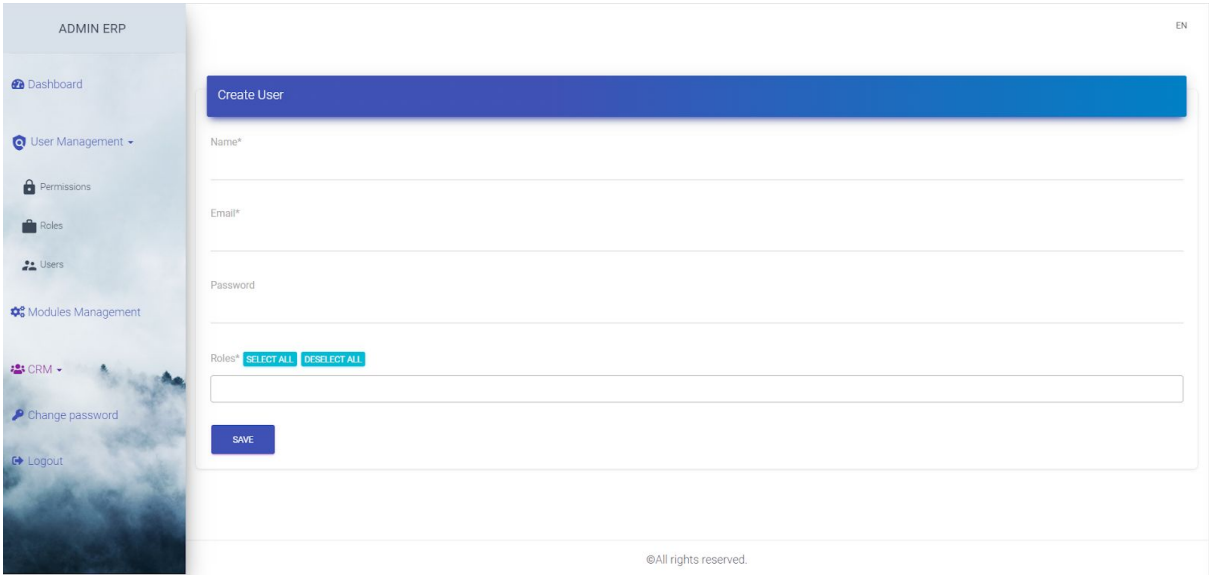


Figure 39 : Formulaire d’ajout d’un utilisateur

6. Interface ‘Modifier mot de passe’

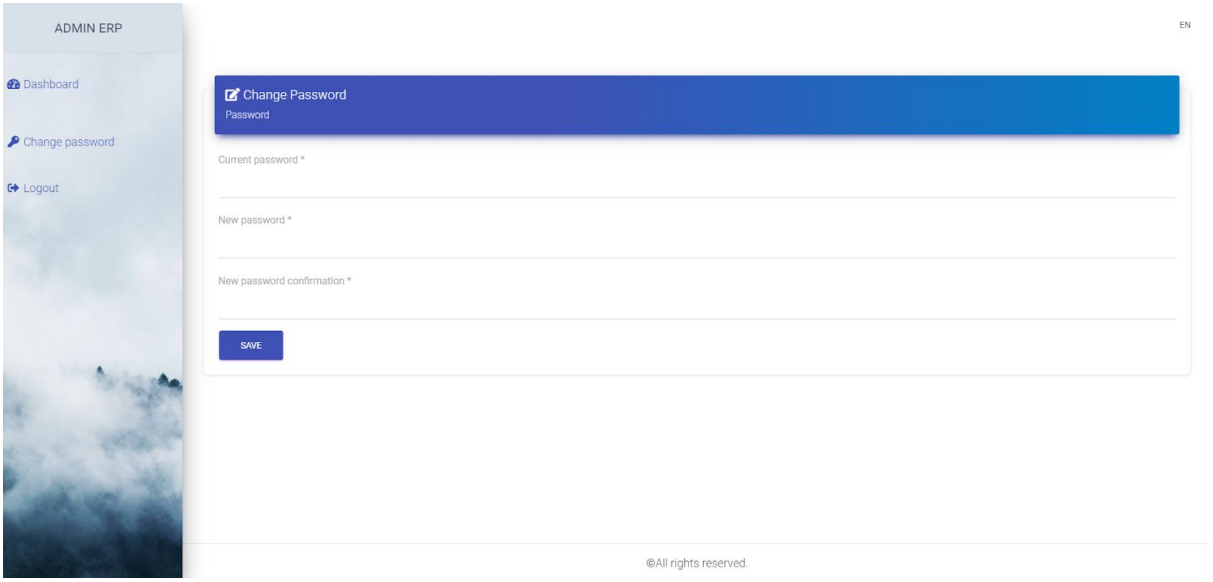


Figure 40 : Modifier le mot de passe

Par mesure de sécurité l’utilisateur nouvellement créé , devra changer son mot de passe après sa première connexion au système.

Conclusion

À travers ce chapitre, nous avons présenté les technologies de développement utilisées pour l’implémentation et la réalisation de notre application.

D’abord, nous avons mis en avant les technologies utilisées en frontend (Boostrap4, material design...).

Ensuite, nous avons exposé les technologies utilisées en backend. Le backend repose essentiellement sur l’utilisation du framework Laravel.

Enfin, nous avons décliné un aperçu de notre application et mis en évidence les principales fonctionnalités offertes par cette dernière.

Conclusion et perspectives

La plupart des organisations du monde entier ont réalisé que dans un environnement commercial en évolution rapide, il fallait des solutions adéquates. Elles adoptèrent l'utilisation de nouveaux systèmes puissants : les ERPs.

Or ces systèmes qui intègrent toutes les données et tous les processus de l'entreprise dans un système unifié, nécessitent d'être sécurisés et correctement administrés.

L'objectif de notre travail était de fournir une solution permettant d'administrer et sécuriser ces plate-formes.

Afin de réaliser ce projet,nous nous sommes d'abord intéressés de près aux ERPs : leur architecture, leur fonctionnement ainsi que les solutions existantes sur le marché des ERPs.

Par la suite nous nous sommes penchés sur l'aspect sécuritaire des plate-formes numériques, en analysant les failles les plus courantes afin de les éviter.

Le contrôle d'accès s'est imposé comme mécanisme de sécurité logique essentiel.

Plusieurs solutions s'offraient à nous concernant les politiques de contrôle d'accès : les modèles DAC, MAC et RBAC.

Notre choix s'est porté sur l'utilisation du modèle de contrôle d'accès basé sur les rôles : le modèle RBAC. Ce choix fut motivé par les raisons suivantes :

- Avec ce modèle, les entreprises n'ont plus besoin d'autoriser ou de révoquer l'accès sur une base individuelle, elles rassemblant plutôt les utilisateurs en fonction de leur rôle. Établir un ensemble de rôles dans une petite ou moyenne entreprise n'est pas difficile. En revanche, la mise en place d'un tel système dans une grande entreprise n'est pas une tâche facile.

- Il permet de garantir le principe du moindre privilège, autrement dit d'accorder uniquement les autorisations nécessaires à l'accomplissement du travail de l'utilisateur.

Ensuite, nous avons modélisé la solution en utilisant le langage de modélisation UML, cette phase fut essentielle et nous a permit de disposer des éléments nécessaires au bon développement de l'application.

Enfin, nous avons implémenté la solution : une application web basée sur le framework Laravel permettant d'administrer et sécuriser une plate-forme ERP.

Le suivi de cette méthodologie de travail a permit d'aboutir à un résultat final répondant aux besoins des entreprises.

Quelques fonctionnalités offertes par l'application :

- Double authentification (Two factor authentication)
- Possibilité de gérer les utilisateurs, les rôles et les permissions
- Mini CRM
- Administration et sécurité du CRM

Cependant, la solution que nous proposons a quelques limites comme l'absence d'une interface graphique pour visualiser les logs et audits (actuellement ceci est possible en backend uniquement).Notre application permet de supprimer plusieurs utilisateurs à la fois , ce qui permet de gagner du temps , en revanche , elle ne permet pas d'affecter un rôle à plusieurs utilisateurs à la fois.

Le module d'administration et sécurité que nous avons implémenté permet de gérer un CRM, or un ERP typique utilisera plusieurs composants de logiciels (modules). C'est pourquoi nous envisageons de poursuivre le développement de ce projet, en répliquant ce processus d'administration et de sécurisation sur l'ensemble des modules de la plate-forme ERP. Le principe étant le même, nous ajouterons des permissions spécifiques à chaque module. Nous envisageons également de regrouper ces permissions sous une suite de rôles, qui seront par la suite attribués aux utilisateurs du système. De nouveaux rôles prédéfinis seront ajoutés au système afin d'aider l'utilisateur à gérer son entreprise tout en ayant la possibilité de créer des rôles personnalisés (option disponible dans la version actuelle).

Afin d'améliorer le système, nous souhaitons ajouter la notion d'exclusion mutuelle entre certains rôles du système.

Références bibliographiques

Articles de recherche ou thèses , cours

[1] Alzahrani, Abdulrahman et al. “Web Application Security Tools Analysis.” 2017 IEEE 3rd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing, (HPSC) and IEEE International Conference on Intelligent Data and Security (IDS) (2017): 237-242..

[2] Aneta PONISZEWSKA-MARAŃDA,Spécification UML du contrôle d’accès dans les systèmes d’information : une approche coopérative de la conception des rôles dans un modèle RBAC, thèse,Université Polytechnique de Łódź,2003.

[3] Cours Cloud computing , S.Fellag

[4] Cours Sécurité informatique, Bourkache

[5] F. M. Elbahri, O. Ismael Al-Sanjary, M. A. M. Ali, Z. Ali Naif, O. A. Ibrahim and M. N. Mohammed, "Difference Comparison of SAP, Oracle, and Microsoft Solutions Based on Cloud ERP Systems: A Review," 2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA), Penang, Malaysia, 2019, pp. 65-70, doi: 10.1109/CSPA.2019.8695976.

[6] Mohamed A.Abd Elmonem,Eman S.Nasr, Mervat H.Geitha, "Benefits and challenges of cloud ERP systems – A systematic literature review ",December 2016.

Ouvrages

[7] Concepts in Enterprise Resource Planning, International Edition by Wagner/Monk - 2013

[8] Enterprise Resource Planning: Global Opportunities & Challenges by Liaquat Hossain, Jon David Patrick and M.A. Rashid x Idea Group Publishing 2002.

[9] Jean-Luc Deixonne.D.(2011).Piloter un projet ERP - 3e édition.Dunod

[10] Jules Rémy.D.(2016).Un ERP dans ma PME.La Ronde des Vivetières.

Sitographie

[11] https://cloud.google.com

[12] https://dynamics.microsoft.com

[13] https://fr.wikipedia.org

[14] http://gdt.oqlf.gouv.qc.ca

[15] https://laravel.com

[16] https://owasp.org

[17] https://startbootstrap.com

[18] https://www.lri.fr/~longuet/Enseignements/17-18/Et3-UML

[19] https://www.nist.gov

[20] https://www.odoo.com

[21] https://www.oracle.com

[22] https://www.phpmyadmin.net

[23] https://www.sap.com