



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHÉMATIQUES ET D'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : GÉNIE INFORMATIQUE.

Par :

MEZIDI Hocine Sadek
MEBKHOUT Saad

Sur le thème

Vers une plateforme multitenant intégrale pour les assistants judiciaires : "eA&Adala"

Soutenu publiquement le 11/06/2024 à Tiaret devant le jury composé de :

Mr ZIOUAL Tahar	MAA Université Ibn Khaldoun - Tiaret	Président
Mr HATTAB Noureddine	MAA Université Ibn Khaldoun - Tiaret	Encadrant
Mr BENATHMANE Lalia	MAA Université Ibn Khaldoun - Tiaret	Examineur

2023-2024

Remerciement

Nous remercions en premier lieu le bon Dieu le tout puissant et le miséricordieux de nous avoir donné le courage et la patience d'achever ce modeste travail.

Nous souhaitons remercier notre encadrant Mr N. HATTAB pour sa disponibilité, sa patience, ses précieux conseils et son accompagnement précieux, qui nous ont amené à la réalisation de ce modeste travail.

Nous exprimons nos remerciements également l'ensemble des enseignants de l'université Ibn Khaldoun –Tiaret- pour leur dévouement et la qualité de leur enseignement, qui nous ont fourni les bases et les outils nécessaires pour réussir ce projet.

Nous souhaitons aussi exprimer nos sincères gratitude à L'ensemble du personnel de l'université Ibn Khaldoun –Tiaret- en collaboration avec Tecno-foster d'organiser la formation et de nous présenter un aperçus général sur la fondation et le développement des start-up.

On tient à exprimer notre profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce mémoire de fin d'études. Leur soutien, leurs conseils et leurs encouragements ont été indispensables tout au long de ce parcours académique.

Enfin, on tient à exprimer nos reconnaissances à nos collègues et amis, pour leur soutien moral et leurs encouragements constants. Leurs discussions constructives et leur camaraderie ont grandement facilité cette période de travail intense.

merci

Dédicace

Nous dédions ce modeste travail

À nos parents, pour leur amour inconditionnel, leur soutien indéfectible et leur patience

infinie

À tous ceux qui nous sont chers, pour leur présence bienveillante et leurs encouragements

constants

À notre directeur de mémoire, Mr N. HATTAB pour sa précieuse guidance, ses conseils

avisés et son engagement tout au long de ce projet.

À tous ceux qui nous avons perdus, vous êtes toujours en mémoire gravé par des souvenirs.

À tous ceux qui croient aux vertus de l'effort, de la patience et de la réussite.

Et enfin à tous ceux qui sont en train de lire ce mémoire



Résumé

Avec la progression rapide de la technologie, de nombreuses architectures ont été remplacées par de nouvelles solutions pour surmonter les limitations de leurs prédécesseurs. Parmi ces nouvelles perspectives, les architectures multi-locataires se sont révélées essentielles pour les entreprises utilisant des systèmes virtualisés tels que le Cloud, le Fog, et l'Edge. Ces avancées ont été particulièrement notables dans des secteurs comme la santé, l'éducation, l'industrie et le système judiciaire.

Le système judiciaire se divise en deux parties : les institutions judiciaires (tribunaux, cours de justice, cours suprêmes, pôles judiciaires) et les assistants judiciaires (avocats, notaires, huissiers, traducteurs, experts). Alors que les institutions judiciaires ont bénéficié d'une numérisation remarquable avec des systèmes de gestion de dossiers complets et uniformes, les assistants judiciaires n'ont pas encore accès à une plateforme unifiée.

Ce projet de fin d'études vise à développer une plateforme SaaS multilocataire nommée "e-A&ADALA" pour gérer efficacement les applications des assistants judiciaires. Contrairement aux applications traditionnelles, cette plateforme sera basée sur le cloud, permettant à un seul code source de servir plusieurs utilisateurs à la fois, réduisant ainsi les coûts de maintenance et de mise à jour. En raison des contraintes de temps, la plateforme se concentrera initialement sur les avocats et les notaires, avec l'intégration des autres assistants judiciaires prévue à court terme.

Mot clés : Système Judiciaire, les assistants judiciaires, Cloud, SaaS, Multi-tenant, Django.

Abstract

With the rapid advancement of technology, many architectures have been replaced by new solutions to overcome the limitations of their predecessors. Among these new perspectives, multi-tenant architectures have proven essential for companies using virtualized systems such as Cloud, Fog, and Edge computing. These advancements have been particularly notable in sectors like healthcare, education, industry, and the judicial system.

The judicial system is divided into two parts: judicial institutions (courts, justice courts, supreme courts, judicial centers) and judicial assistants (lawyers, notaries, bailiffs, translators, experts). While judicial institutions have benefited from remarkable digitization with comprehensive and uniform case management systems, judicial assistants have yet to access a unified platform.

This final project aims to develop a multi-tenant SaaS platform named "e-A&ADALA" to efficiently manage the applications of judicial assistants. Unlike traditional applications, this platform will be cloud-based, allowing a single codebase to serve multiple users simultaneously, thus reducing maintenance and update costs. Due to time constraints, the platform will initially focus on lawyers and notaries, with the integration of other judicial assistants planned in the short term.

Keywords: Judicial System, judicial assistants, Cloud, SaaS, Multi-tenant, Django.

ملخص

مع التقدم السريع في التكنولوجيا، تم استبدال العديد من الهياكل بحلول جديدة للتغلب على قيود أسلافها. من بين هذه المنظورات الجديدة، أثبتت الهياكل متعددة المستأجرين أنها ضرورية للشركات التي تستخدم أنظمة افتراضية مثل السحابة (Cloud) والحوسبة الضبابية (Fog) والحوسبة الطرفية (Edge). كانت هذه التقدّمات ملحوظة بشكل خاص في قطاعات مثل الصحة والتعليم والصناعة والنظام القضائي.

يتقسم النظام القضائي إلى جزئين: المؤسسات القضائية (المحاكم، دوائر القضاء، المحاكم العليا، القطاعات القضائية) والمساعدین القضائيين (المحامين، كتاب العدل، المفوضين القضائيين، المترجمين، الخبراء). في حين أن المؤسسات القضائية استفادت من رقمنة ملحوظة مع أنظمة إدارة الملفات الكاملة والموحدة، فإن المساعدات القضائيين لم يحصلوا بعد على منصة موحدة.

يهدف هذا المشروع المقدم إلى تطوير منصة خدمات برمجية (SaaS) متعددة المستأجرين تسمى "e-ADALA" لإدارة تطبيقات المساعدات القضائيين بكفاءة. على عكس التطبيقات التقليدية، ستكون هذه المنصة قائمة على السحابة، مما يسمح لكود مصدر واحد بخدمة العديد من المستخدمين في وقت واحد، وبالتالي تقليل تكاليف الصيانة والتحديث. نظرًا للقيود الزمنية، ستركز المنصة في البداية على المحامين والموثقين، مع دمج المساعدات القضائيين الآخرين في المستقبل القريب.

كلمات مفتاحية: النظام القضائي، المساعدات القضائيين، السحابة، البرمجيات كخدمة (SaaS)، متعددة المستأجرين،

Django.

Liste des Abréviations et symboles

PaaS: Platform as a Service.

SaaS: Software as a Service.

IaaS: Infrastructure as a Service.

UML: Unified Modeling Language.

PDF: Portable Document Format.

MVC: Model-View-Controller.

VM : Machines virtuelles.

PM: Machine Physique.

CRM: Customer Relationship Management.

AWS : Amazon Web Services.

IBM: International Business Machines.

HTTP: Hypertext Transfer Protocol.

HTML : HyperText Markup Language.

MTV: Modèle-Vue- Template.

XML : Extensible Markup Language.

SGBD : Système de gestion base de données.

IDE : environnement de développement intégré

Liste des tableaux

Tableau 1.1	Cloud Computing Avantages et limites	4
Tableau 1.2	Architecture Multi-Tenant Vs Single-Tenant	9
Tableau 2.1	Les besoins ciblés pas la platform	18
Tableau 2.2	Classes sur lesquelles se porte notre application desktop	24
Tableau 3.1	Les caractéristiques de l'ordinateur	26
Tableau 3.2	Tools et versions	28

Liste des figures

Figure 1.1	Les paradigmes informatiques avant et après La virtualisation	3
Figure 1.2	Cloud public	5
Figure 1.3	Cloud privé	6
Figure 1.4	Cloud hybride	6
Figure 1.5	Modèle des services cloud	7
Figure 1.6	Architecture avec plusieurs bases de données pour une instance d'application	8
Figure 1.7	Architecture avec plusieurs bases de données et plusieurs instances d'applications	8
Figure 1.8	Application SaaS vs Non-SaaS	11
Figure 2.1	Différents composants du système judiciaire	14
Figure 2.2	Diagramme de cas d'utilisation générale	19
Figure 2.3	Diagramme de cas d'utilisation d'avocat	20
Figure 2.4	Diagramme de cas d'utilisation du notaire	21
Figure 2.5	Diagramme de séquence d'avocat	22
Figure 2.6	Diagramme de séquence de service de messagerie.....	23
Figure 2.7	Diagramme de classe générale.....	25
Figure 3.1	Architecture générale	29
Figure 3.2	Architecture MTV	29
Figure 3.3	page d'accueil principale	30
Figure 3.4	Interface d'inscription	31
Figure 3.5	Création de compte	31
Figure 3.6	Interface connexion.....	32
Figure 3.7	Table de tarifs d'abonnement	32
Figure 3.8	page d'accueil de l'application de l'avocat.....	33
Figure 3.9	Gestion des affaires	34
Figure 3.10	Informations divers sur les affaires.....	35
Figure 3.11	Suivie de l'affaire	36
Figure 3.12	Page d'accueil.....	37
Figure 3.13	Les contrats disponibles	37
Figure 3.14	Saisie du contrat	38

Figure 3.15	Gestion des clients.	38
Figure 3.16	Interface de paiement.....	39
Figure 3.17	Gestion des rendez-vous.....	39
Figure 3.18	page d'accueil client.....	40
Figure 3.19	Liste des assistants judiciaires conventionnés	40
Figure 3.20	Messagerie	41

Résumé

Liste des Abréviations et symboles

Liste des tableaux

Liste des figures

Table de matière

Introduction générale1

CHAPITRE 01 : LE CLOUD ET LA MULTI-LOCATION

1	Introduction.....	3
2	Transformation des Paradigmes Informatiques : Avant et Après la Virtualisation	3
3	Définition du cloud computing	4
3.1	Avantages et Limites	4
3.2	Classification des modèles de Cloud Computing	5
3.2.1	Modèle de déploiement	5
3.2.2	Le modèle de services	7
4	Le modèle de service "SaaS"	8
4.1	Fonctionnement de SAAS	8
4.2	Tarifification des solutions SaaS.....	9
4.2.1	Tarifification gratuite ou "freemium"	9
4.2.2	Tarifification par abonnement mensuel ou annuel à plusieurs niveaux.....	9
4.3	Avantages du modèle de logiciel SaaS	10
4.4	Inconvénients du SaaS	10
5	L'Architecture Multi-Tenant "Multi-locataire"	10
5.1	Définition.....	11
5.2	Comparaison avec les architectures single-tenant	11
5.3	Avantages de l'architecture multi-locataire	12
5.4	Inconvénients de l'architecture multi-locataire	13
6	Conclusion	13

CHAPITRE 02 : MODELISATION ET CONCEPTION

1	Introduction.....	14
2	Le système judiciaire : Définition et Organisation	14
3	La modélisation basée sur la démarche UML	15
4	Pourquoi UML ? Comme une approche de modélisation	15
5	La démarche basée sur UML : Les principaux modèles	16

6	Star-UML.....	17
7	Le modèle fonctionnel et l'identification des acteurs	17
7.1	Les tâches des assistants judiciaires dans le système	17
7.2	Diagramme de cas d'utilisation	17
7.3	Diagramme de séquence	21
7.4	Diagramme de classe	23
8	Conclusion	25

CHAPITRE 03 : IMPLEMENTATION ET REALISATION

1	Introduction.....	26
2	Les outils d'implémentation et de réalisation	26
2.1.1	Environnement matériel	26
2.1.2	Environnement logiciel	26
3	Développement de la plateforme e-A&ADALA : Architecture & Interfaces	28
3.1	Architecture Générale.....	28
3.2	Présentation des différentes interfaces de la plateforme.....	30
4	Conclusion	41
	Conclusion Générale et Perspectives.....	42

Introduction générale

1. Contexte

La numérisation du gouvernement, devenue une nécessité après la pandémie de COVID-19, a entraîné une transformation technologique majeure dans de nombreux secteurs, notamment la santé, l'éducation, l'industrie et le système judiciaire.

Le système judiciaire se divise en deux parties : les institutions judiciaires (tribunaux, cours de justice, cours suprêmes, pôles judiciaires) et les assistants judiciaires (avocats, notaires, huissiers, traducteurs, experts). Alors que les institutions judiciaires ont connu une numérisation remarquable avec des systèmes de gestion de dossiers complets et uniformes, les assistants judiciaires n'ont pas encore bénéficié d'une plateforme unifiée.

La popularité croissante des logiciels en tant que service (SaaS) et des architectures orientées services a propulsé le concept de multi-location au premier plan de l'attention dans le domaine des applications cloud. La multi-location dans le cloud est un domaine de recherche prometteur qui continue de susciter un vif intérêt. Cette attention est principalement due à sa capacité à partager des ressources matérielles et logicielles avec plusieurs utilisateurs du cloud, ce qui permet d'accroître la rentabilité des applications cloud existantes. La notion de multi-location est interprétée différemment selon les différents modèles de service cloud. En particulier, les significations de la multi-location varient entre les couches IaaS, PaaS et SaaS du modèle cloud.

2. Objectif

Ce travail se focalise sur la multi-location de type SaaS, qui offre le niveau d'isolement le plus élevé. Dans la multi-location, plusieurs utilisateurs partagent une même instance d'application cloud. Deux modèles principaux se distinguent : Multi-instance Multi-tenant (MIMT) et Single-Instance to Multi-tenant (SIMT). Dans le modèle MIMT, chaque utilisateur dispose d'une instance d'application dédiée. En revanche, dans le modèle SIMT, tous les utilisateurs partagent la même instance d'application. Un avantage commun à ces deux modèles est la réduction des coûts de service grâce au partage des ressources, ce qui conduit à une meilleure utilisation des ressources.

Ce projet de fin d'études vise à développer une plateforme SaaS multilocataire nommée "e-A&ADALA" pour gérer efficacement les applications des assistants judiciaires. Cette

plateforme, contrairement aux applications traditionnelles, sera basée sur le cloud, permettant à un seul code source de servir plusieurs utilisateurs à la fois, réduisant ainsi les coûts de maintenance et de mise à jour.

En raison des contraintes de temps, la plateforme se concentrera initialement sur les avocats et les notaires, les autres assistants judiciaires étant considérés comme une perspective à court terme.

3. Structure Générale

Ce mémoire est structuré en trois chapitres complémentaires qui explorent divers aspects du sujet traité.

Dans le premier chapitre, nous analyserons le paradigme le plus couramment utilisé aujourd'hui dans les environnements virtualisés, à savoir le cloud computing. Nous réaliserons également une recherche bibliographique concise sur ce paradigme, en nous concentrant particulièrement sur le modèle de service SaaS. De plus, nous fournirons un aperçu de l'architecture multi-tenant, en décrivant sa définition, ses avantages et ses inconvénients.

Le deuxième chapitre est principalement consacré à la modélisation et à la formalisation. Nous y donnerons une description détaillée des fonctionnalités et du comportement de la plateforme

proposée en utilisant le langage UML. De plus, nous présenterons quelques modèles conceptuels sous forme de diagrammes UML pour illustrer ces fonctionnalités.

Dans le troisième chapitre, nous aborderons l'implémentation et la réalisation de la plateforme e-A&ADALA, illustrée par un ensemble de fenêtres et d'interfaces. Enfin, pour conclure ce travail, nous fournirons une conclusion générale ainsi qu'un ensemble de perspectives.

Chapitre 01

LE CLOUD ET LA MULTI-LOCATION

Paradigme et Caractéristiques

1 Introduction

La croissance rapide de la technologie Internet a engendré une tendance majeure dans le domaine informatique : le Cloud Computing. Ce paradigme innovant repose sur une approche prometteuse appelée architecture multi-tenant, qui encourage le partage des ressources matérielles et logicielles entre plusieurs utilisateurs, maximisant ainsi les avantages des applications cloud. Dans ce premier chapitre, nous examinerons attentivement ce paradigme, en nous concentrant particulièrement sur le modèle de service SaaS. De plus, nous présenterons une vue d'ensemble de l'architecture multi-tenant, en définissant ses caractéristiques, ses avantages et les défis qu'elle pose.

2 Transformation des Paradigmes Informatiques : Avant et Après la Virtualisation

L'impact révolutionnaire de la virtualisation dans la transformation du paysage informatique au cours des dernières décennies. Avant son émergence, les entreprises déployaient des serveurs physiques dédiés à chaque application ou service, ce qui s'avérait à la fois coûteux et inefficace.

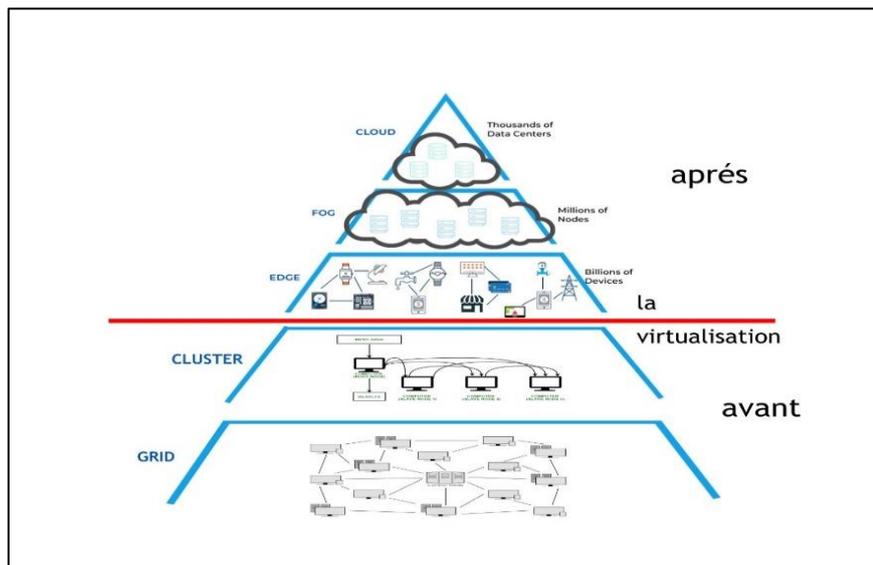


FIGURE 1.1- Les paradigmes informatiques avant et après La virtualisation^[01].

La **Figure 1.1** offre une illustration claire de la différence entre les paradigmes informatiques avant et après l'introduction de la virtualisation des ressources informatiques.

La virtualisation a révolutionné la création de machines virtuelles (VM) sur une seule machine physique (PM), permettant ainsi de séparer les ressources informatiques en instances isolées. Ce changement par rapport aux systèmes non virtualisés tels que le Grid computing et le Cluster computing a apporté des avantages significatifs, tels que la réduction des coûts, une plus grande flexibilité et disponibilité des ressources informatiques. De plus, la virtualisation a facilité

l'émergence de nouveaux paradigmes tels que l'IoT, le Fog Computing, l'Edge Computing et le Cloud Computing ^[02].

3 Définition du cloud computing

Le cloud computing est un modèle qui permet un accès facile, pratique et à la demande à un ensemble partagé de ressources informatiques configurables (tels que les réseaux, les serveurs, le stockage, les applications et les services). Ces ressources peuvent être rapidement allouées et libérées avec un effort de gestion minimal ou une interaction directe avec le fournisseur de services. Ce modèle de cloud se caractérise par cinq éléments clés, ainsi que trois modèles de service et quatre modèles de déploiement.^[03]

3.1 Avantages et Limites

Le tableau 1.1 ci-dessous offre une vue d'ensemble des avantages et des limitations du paradigme du cloud computing.

Tableau 1.1- Cloud Computing Avantages et limites.^[04]

Avantage	Limites
<ul style="list-style-type: none"> -La capacité de partager des données entre les utilisateurs, ce qui encourage la création rapide de plateformes virtuelles collaboratives. -Disponibilité d'une puissance de calcul significative, offrant aux entreprises l'avantage de bénéficier de ressources et de performances optimales sans limites géographiques. -Accès libre et ouvert de n'importe où, sans nécessité de configuration d'un VPN, offrant une connectivité immédiate aux données. -Réduction des coûts grâce à la mutualisation des services entre de nombreux utilisateurs, permettant ainsi une solution financièrement avantageuse. 	<ul style="list-style-type: none"> - Confidentialité et sécurité des données : Lorsque les données sont stockées en dehors de l'entreprise, il existe un risque de vol ou de mauvaise utilisation, ce qui nécessite une confiance accrue dans les mesures de sécurité et de confidentialité mises en place par le fournisseur de services. Dépendance vis-à-vis du fournisseur : Certains fournisseurs peuvent restreindre la flexibilité des fonctionnalités disponibles, limitant ainsi les options pour les utilisateurs. Il est donc important de vérifier les limitations imposées par le fournisseur choisi avant de s'engager.

3.2 Classification des modèles de Cloud Computing

Lorsqu'on aborde le paradigme du cloud computing, il est important de considérer deux aspects fondamentaux : les modèles de déploiement et les modèles de service. ^[05]

3.2.1 Modèle de déploiement : Désigne la manière dont les ressources et les services du cloud sont déployés et gérés. Il existe plusieurs modèles de déploiement couramment utilisés

- a. **Public :** Le modèle de déploiement le plus répandu dans le domaine du Cloud Computing est le cloud public, qui est partagé entre plusieurs utilisateurs sans être spécifiquement dédié à une seule entreprise. Il offre aux utilisateurs la possibilité d'accéder aux services via Internet sans avoir à posséder leurs propres serveurs. Les utilisateurs ne paient que pour les services qu'ils utilisent, ce qui rend ce modèle économiquement avantageux. Le cloud public est couramment utilisé pour des services tels que la messagerie électronique, les applications de bureau, le stockage en ligne, ainsi que pour les environnements de développement et de test. .

L'illustration ci-dessous (**Figure 1.2**) représente visuellement ce modèle de déploiement :

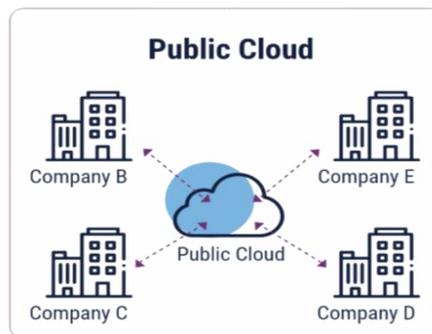


FIGURE 1.2-Cloud public.

- b. **Privé :** Le cloud privé permet à une entreprise ou une organisation de bénéficier d'un accès exclusif à des ressources de Cloud Computing. Il peut être hébergé localement dans le centre de données de l'entreprise ou auprès d'un fournisseur tiers. Ce modèle convient particulièrement aux organisations qui ont besoin d'un contrôle total sur leurs environnements cloud, ce qui garantit une sécurité et une conformité réglementaire optimales. Il est souvent préféré par les organismes publics, les institutions financières et les entreprises de taille moyenne à grande.

L'illustration ci-dessous (**Figure 1.3**) représente visuellement ce modèle de déploiement :

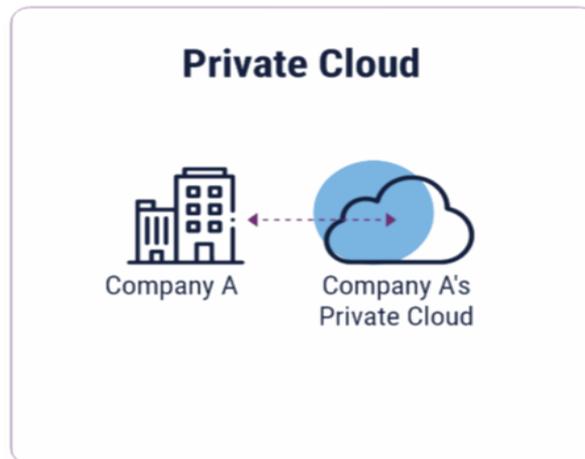


FIGURE 1.3- Cloud privé.

- c. **Cloud hybride** : Le cloud hybride est une combinaison d'environnements de cloud privé et public, parfois avec une infrastructure locale, permettant le déplacement des données et des applications entre ces deux environnements. Ce modèle offre une flexibilité, une réactivité et une sécurité accrues en tirant parti des avantages des deux modèles. Il est idéal pour absorber les pics d'activité ou répondre à des besoins temporaires en ressources. Le cloud hybride est largement adopté par de nombreuses organisations pour répondre aux exigences réglementaires, de souveraineté des données et de faible latence.

L'illustration ci-dessous (**Figure 1.4**) représente visuellement ce modèle de déploiement :

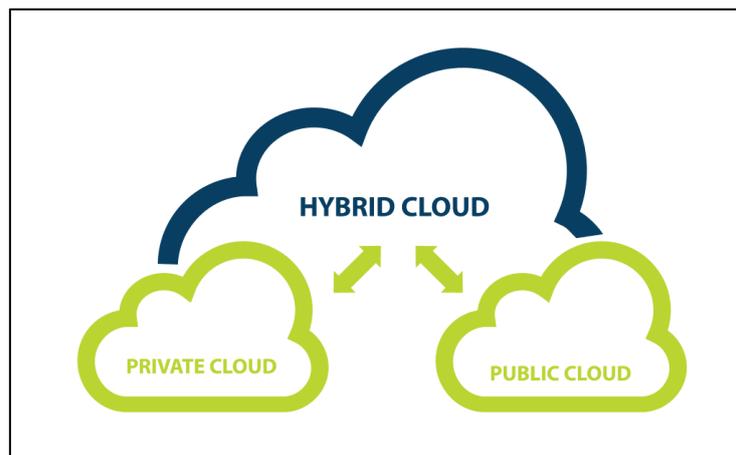


FIGURE 1.4-Cloud hybride.

3.2.2 Le modèle de services dans le paradigme cloud, on trouve trois principaux modèles de services, également appelés "as-a-Service", qui offrent différents niveaux de gestion. Ces modèles sont les suivants :^[06]

- a) **IaaS (Infrastructure as a Service)** Ce modèle permet d'accéder à des ressources d'infrastructure hautement évolutives, telles que des instances de serveur, du stockage

et des services réseau. Il offre a possibilité de configurer rapidement ces ressources en fonction des besoins spécifiques, tandis que la gestion des systèmes d'exploitation, des middlewares et des applications est prise en charge par l'utilisateur.

- b) **PaaS (Platform as a Service)** ce modèle de service libère les équipes de développement des contraintes liées au réseau, au stockage, à la sauvegarde et au système d'exploitation, leur permettant de se concentrer pleinement sur le développement d'applications. Elle offre un environnement de travail où ces aspects sont gérés de manière transparente, permettant ainsi aux équipes de gagner en productivité et en efficacité.
- c) **SaaS (Software as a Service)** Les utilisateurs ont la possibilité d'accéder aux applications via le cloud sans restriction quant au nombre d'utilisateurs. Le fournisseur de services cloud se charge des mises à jour et de la maintenance des applications et de l'infrastructure, offrant ainsi un accès flexible et rentable à une gamme de logiciels d'entreprise.

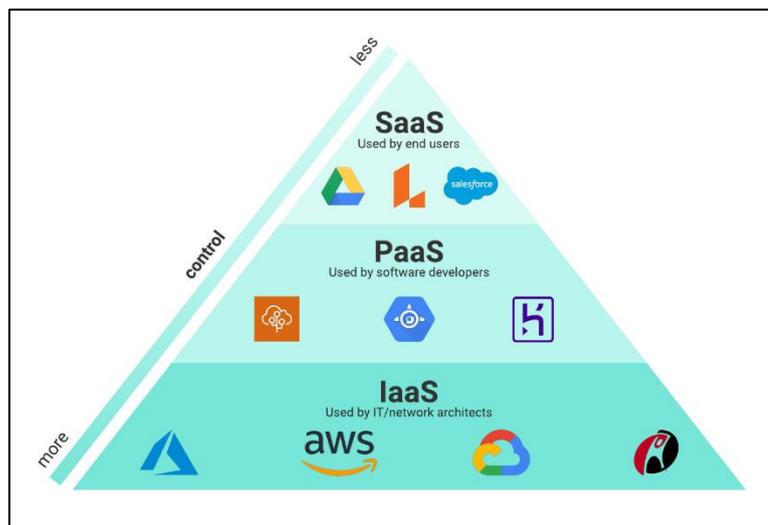


FIGURE 1.5- Modèle des services cloud.

La figure (**Figure1.5**) présente une illustration démontrant des exemples de trois modèles de services dans le cloud. Dans les paragraphes suivants, nous aborderons brièvement le modèle de service SaaS, qui occupe une place centrale dans ce mémoire.

4 Le modèle de service "SaaS" ^[07]

Le modèle de logiciel en tant que service (SaaS) est une méthode de distribution de logiciels dans laquelle l'application est hébergée sur le cloud et accessible via Internet. Le fournisseur de services SaaS gère l'infrastructure et la maintenance, tandis que les utilisateurs paient des frais d'abonnement pour accéder au logiciel. Ce modèle a gagné en popularité à la fin des années 1990 avec des services tels que Salesforce CRM. Aujourd'hui, il est largement utilisé pour une variété d'applications, offrant aux entreprises de toutes tailles des avantages tels que des coûts prévisibles, une mise en œuvre rapide et une réduction des frais de gestion.

L'illustration ci-dessous (**Figure 1.8**) présente une représentation distinctive des logiciels SaaS et des logiciels non SaaS.

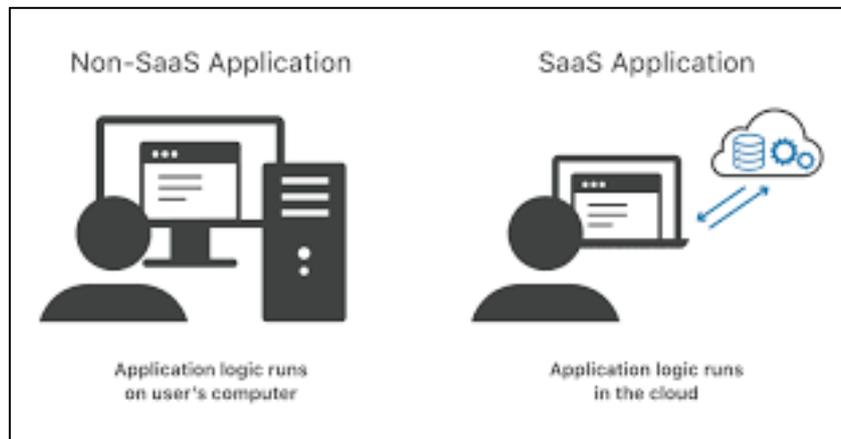


FIGURE 1.6- Application SaaS vs Non-SaaS.

4.1 Fonctionnement de SaaS ^[08]

Les applications SaaS sont spécifiquement conçues pour être hébergées sur le cloud. Le fournisseur de logiciels SaaS peut choisir d'héberger l'application sur son propre infrastructure cloud ou de s'appuyer sur un fournisseur de services cloud tel que Amazon Web Services (AWS), Google Cloud, IBM Cloud ou Microsoft Azure. L'hébergement auprès d'un fournisseur de services cloud établi permet au fournisseur SaaS d'offrir une évolutivité et une accessibilité mondiales, répondant ainsi aux besoins spécifiques de certains clients.

Toute personne disposant d'une connexion Internet et d'un appareil connecté à Internet, tel qu'un ordinateur, un téléphone mobile ou une tablette, peut accéder aux applications SaaS. Ces applications s'exécutent généralement dans n'importe quel navigateur Web. Sur les appareils mobiles, elles peuvent fonctionner de manière plus efficace en utilisant une application mobile ou une application spécifique à la tablette. Certaines applications SaaS, telles qu'Adobe Acrobat, peuvent offrir ou nécessiter un client léger dédié que les utilisateurs doivent télécharger et installer sur leurs ordinateurs.

Les applications SaaS utilisent une architecture multi-locataire, dans laquelle une seule instance de l'application sert chaque client. Pour des raisons de sécurité et de confidentialité des

données, les données de l'application de chaque client, les données utilisateur, les données système et les configurations personnalisées sont séparées de celles des autres clients. ^[09]

4.2 Tarification des solutions SaaS

La tarification des applications SaaS diffère des licences de logiciel traditionnelles et comprend les éléments suivants :

4.2.1 Tarification gratuite ou "freemium" : Les fournisseurs de SaaS proposent souvent une version gratuite pour une période limitée ou une version "freemium" qui offre des fonctionnalités limitées. Cette approche vise à inciter les utilisateurs gratuits à passer à une version payante.

4.2.2 Tarification par abonnement mensuel ou annuel à plusieurs niveaux : Les produits SaaS offrent souvent des plans d'abonnement basés sur différents niveaux de fonctionnalités ou d'utilisation, ce qui permet aux petites et moyennes entreprises de choisir un plan adapté à leurs besoins. Les grandes entreprises peuvent bénéficier de niveaux "entreprise" offrant des fonctionnalités avancées et des conditions de support spécifiques.

Cette tarification à plusieurs niveaux offre aux clients la possibilité de choisir le plan qui correspond le mieux à leurs besoins et à leur budget. Elle inclut souvent des réductions pour les paiements annuels et permet des ajustements de prix en fonction des changements de niveau pendant la durée de l'abonnement.

4.3 Avantages du modèle de logiciel SaaS^[34]

Les avantages et les bénéfices des applications SaaS peuvent être mieux compris en les comparant aux logiciels traditionnels, qui sont installés et gérés sur une infrastructure sur site. Voici les principales différences :

- **Adoption et mise en œuvre plus rapides :** Le SaaS permet une adoption quasi instantanée et une mise en œuvre plus rapide. Les clients peuvent acheter et commencer à utiliser les applications SaaS immédiatement, parfois en quelques minutes, avec des coûts initiaux minimes (principalement le coût de l'abonnement du premier mois). En revanche, les logiciels traditionnels nécessitent l'achat de serveurs, l'installation du logiciel sur chaque appareil utilisateur et l'acquisition de licences complètes pour chaque utilisateur.
- **Accès aux nouvelles fonctionnalités et versions :** Le SaaS offre un accès immédiat aux nouvelles fonctionnalités et versions dès qu'elles sont disponibles. Les fournisseurs SaaS mettent souvent à jour les fonctionnalités et ajoutent de nouvelles fonctionnalités plusieurs fois par semaine, sans causer de perturbations pour les clients. Ils peuvent même améliorer l'interface et l'expérience utilisateur sans interrompre le travail des clients. En revanche, les mises à jour périodiques des logiciels sur site sont souvent coûteuses et perturbatrices, ce qui fait que les clients peuvent attendre plusieurs mois

pour bénéficier des fonctionnalités d'une nouvelle version (à moins qu'ils ne choisissent de sauter certaines mises à jour).

- **Scalabilité rentable et à la demande** : Les clients peuvent faire évoluer les applications SaaS en fonction de leurs besoins en augmentant ou en diminuant les niveaux d'utilisation ou en achetant plus de capacité. En revanche, les logiciels traditionnels nécessitent l'achat de capacités supplémentaires en prévision de pics d'utilisation, ce qui entraîne souvent une sous-utilisation et un gaspillage de capacité jusqu'à ce qu'elle soit réellement nécessaire.
- **Coûts prévisibles et frais généraux considérablement réduits** - Avec le SaaS, il n'est pas nécessaire de prévoir un budget pour l'infrastructure sur laquelle faire fonctionner le logiciel, pour les mises à niveau périodiques des logiciels et l'infrastructure pour les supporter, et surtout pour le personnel informatique interne chargé d'installer, de mettre à niveau et de maintenir le logiciel. Presque toutes les dépenses sont directement liées à l'utilisation du logiciel.

4.4 Inconvénients du SaaS^[11]

- **Sécurité des données** : Les données étant stockées dans le cloud, cela peut poser des problèmes de sécurité pour certains utilisateurs. Le cloud computing n'est pas plus sécurisé que les solutions sur site.
- **Contrôle réduit** : Les applications logicielles internes offrent un plus grand contrôle aux propriétaires d'entreprise par rapport aux solutions SaaS.
- **Dépendance à Internet** : Les logiciels SaaS étant hébergés en ligne, ils ne peuvent pas être utilisés sans connexion Internet.
- **Vitesse d'exécution** : Les applications SaaS fonctionnent à des vitesses légèrement inférieures à celles des applications client/serveur.

5 L'Architecture Multi-Tenant "Multi-locataire"

5.1 Définition : L'architecture multi-tenant est un modèle dans lequel une seule instance d'une application logicielle, y compris sa base de données et son infrastructure sous-jacente, est partagée par plusieurs utilisateurs. Chaque utilisateur, qu'il s'agisse d'un individu ou d'une organisation, dispose de ses propres données isolées et sécurisées. Ce modèle est largement utilisé dans les services SaaS et les hébergements cloud partagés.^[11]

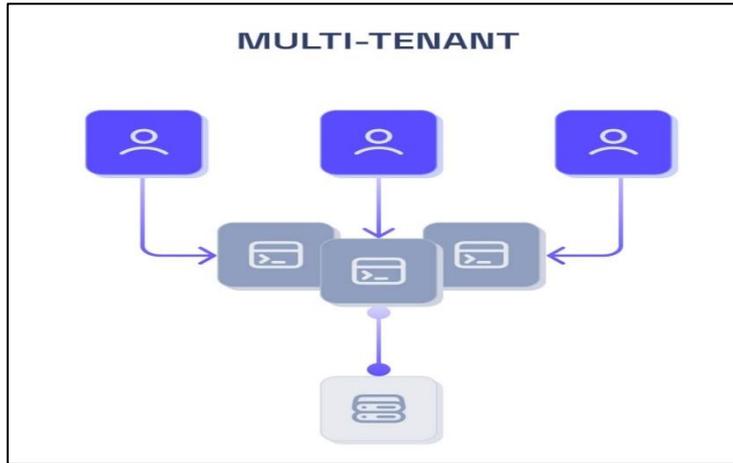


FIGURE 1.7- Architecture avec plusieurs bases de données pour une instance d’application.

5.2 Comparaison avec les architectures single-tenant ^[13]

Avant de comparer les architectures single-tenant et multi-tenantes, voici une explication simple de l'architecture single-tenant : il s'agit d'une architecture où une seule instance d'une application logicielle et de son infrastructure de support est dédiée à un seul client. Maintenant, passons à la comparaison entre les deux architectures

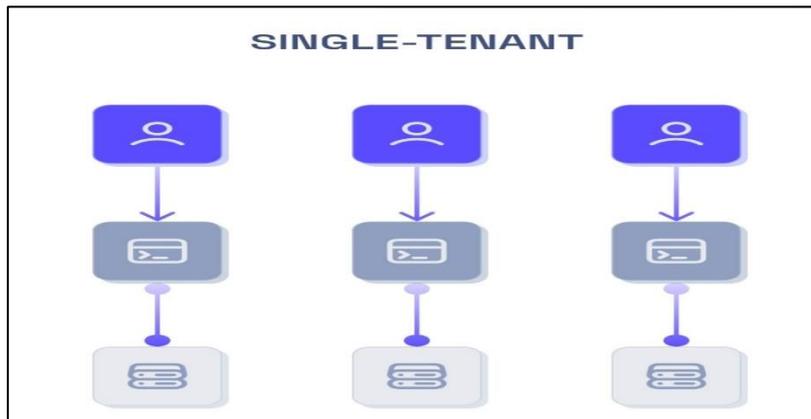


FIGURE 1.8- Architecture avec plusieurs bases de données et plusieurs instances d’applications.

Tableau 1.2- Architecture Multi-Tenant Vs Single-Tenant.

Critère	Single-Tenant	Multi-Tenant
Sécurité	Données totalement isolées des autres clients, réduisant les risques de vols de données.	Données partagées dans le même environnement, mais strictement isolées par des protocoles de sécurité.
Flexibilité - Personnalisation	Personnalisation complète des fonctionnalités et des mises à jour	Personnalisation limitée, car les configurations sont partagées entre les clients.

Flexibilité -Scalabilité	Facilité d'extension des capacités (mémoire, stockage) sur demande	Scalabilité intégrée, mais dépend de l'efficacité de la gestion des ressources partagées.
Coût	Coûts d'infrastructure et de maintenance plus élevés, car non mutualisés.	Coûts réduits grâce à la mutualisation des ressources entre les clients.
Facilité d'usage	Mise en place plus longue nécessitant une infrastructure dédiée.	Mise en place rapide et accès quasi instantané aux logiciels.
Mises à jour	Mises à jour personnalisées, planifiées en fonction des besoins du client.	Mises à jour automatiques effectuées par l'éditeur, profitant à tous les clients.

5.3 Avantages de l'architecture multi-locataire ^[13]

Les avantages de l'architecture multi-tenant sont importants tant pour les fournisseurs de logiciels que pour les clients locataires :

- **Coûts réduits** : En gérant plusieurs locataires à partir d'une seule instance d'application et d'infrastructure de support, les coûts permanents sont généralement inférieurs à ceux d'une monolocation. Les logiciels SaaS sont proposés avec un abonnement mensuel ou annuel prévisible, basé sur le nombre d'utilisateurs, le niveau d'utilisation ou les volumes de données gérés.
- **Évolutivité** : Les locataires peuvent facilement mettre à l'échelle leurs besoins selon la demande. Les nouveaux utilisateurs accèdent à la même instance logicielle moyennant une augmentation progressive de l'abonnement.
- **Personnalisation sans codage** : Les solutions SaaS multilocataires sont hautement configurables, permettant à chaque locataire de personnaliser l'application selon ses besoins sans nécessiter de développement personnalisé coûteux et risqué.
- **Mises à jour et maintenance continues** : Les fournisseurs de logiciels multilocataires sont responsables des mises à jour et des correctifs, ce qui garantit des fonctionnalités à jour sans intervention du client.

- **Amélioration de la productivité** : En déchargeant les locataires de la gestion de l'infrastructure et des logiciels, ces derniers peuvent se concentrer sur des tâches plus importantes, augmentant ainsi leur productivité globale.

5.4 Inconvénients de l'architecture multi-locataire ^[14]

- Mécanisme d'authentification complexe** : En raison de la nature partagée de la répartition des ressources dans le multi-tenant, il existe des protocoles stricts pour les points d'accès aux données. Cela peut entraîner une complexité accrue dans la gestion des mécanismes d'authentification, ce qui peut nécessiter des efforts supplémentaires pour assurer la sécurité des données et garantir un accès approprié aux utilisateurs.
- Réflexion du serveur** : L'architecture multi-tenant expose les locataires au risque de réflexion du serveur, ce qui signifie que tous les locataires deviennent non fonctionnels chaque fois que le serveur tombe en panne pour une raison quelconque. Cela peut entraîner une interruption du service pour tous les utilisateurs, ce qui peut avoir un impact significatif sur l'expérience utilisateur et la continuité des activités.
- Risque de confidentialité** : Dans une architecture multi-tenant, il existe un risque élevé de violation de la confidentialité des données. Étant donné que les données de tous les locataires sont stockées dans une base de données partagée, toute intrusion de tiers pourrait compromettre la sécurité des données et entraîner une corruption ou un vol de données. Cela suscite de graves préoccupations en matière de confidentialité et de conformité réglementaire pour les entreprises qui utilisent ce type d'architecture.

6 Conclusion

En conclusion, le Cloud Computing et l'architecture multi-locataires ont révolutionné le domaine de l'informatique en offrant des avantages significatifs aux utilisateurs. Le partage des ressources matérielles et logicielles a permis de maximiser l'efficacité des applications cloud, notamment dans le modèle de service SaaS. Malgré ses nombreux avantages, l'architecture multi-locataires présente également des défis en termes de complexité de l'authentification, de risques de réflexion du serveur et de confidentialité des données. Cependant, avec une gestion appropriée, cette approche continue de jouer un rôle essentiel dans la transformation numérique des entreprises.

Chapitre 02

Modélisation et Conception

1 Introduction

Ce chapitre est centré sur le système judiciaire, en particulier sur les assistants juridiques qui jouent un rôle essentiel pour assurer le bon fonctionnement de l'ensemble du système. Dans cette partie, nous exposons notre approche conceptuelle pour modéliser ce système, en adoptant principalement la démarche basée sur le Langage de modélisation unifié UML.

2 Le système judiciaire : Définition et Organisation^[15]

Lorsque l'on parle du secteur judiciaire en Algérie, il peut être segmenté en deux composantes principales (**Figure 2.1**). La première composante concerne les différents types et structures de tribunaux, y compris les tribunaux ordinaires, les cours suprêmes et les pôles judiciaires. La deuxième composante concerne les secteurs de soutien des tribunaux, qui comprend des avocats, des huissiers de justice, des traducteurs, des et des notaires.

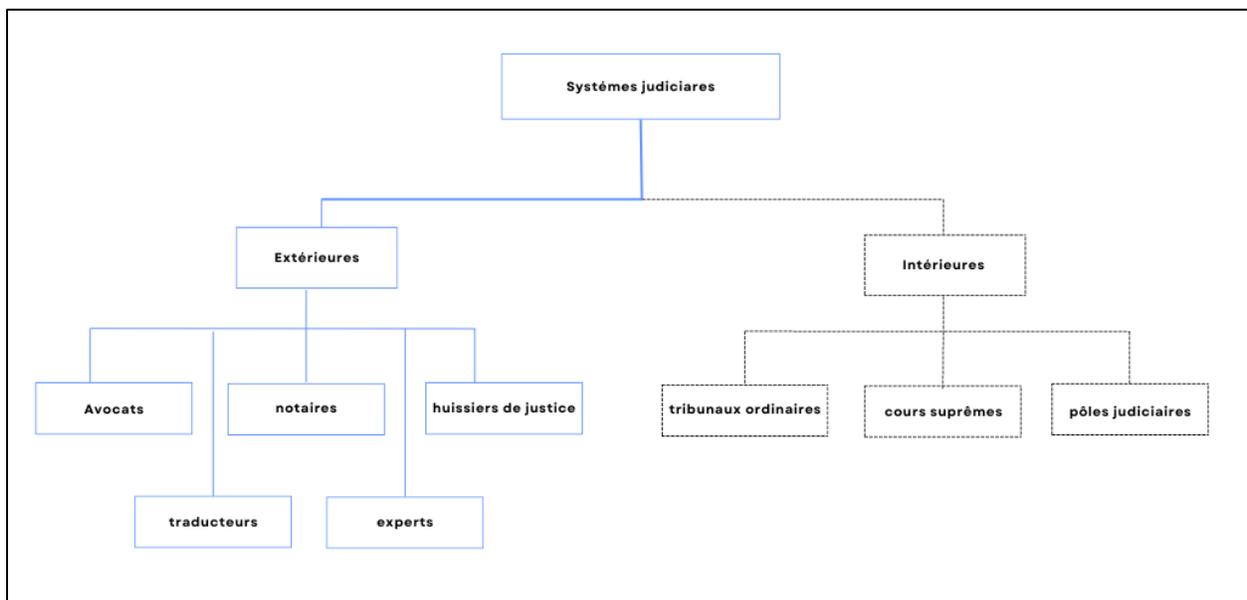


FIGURE 2.1-Différents composants du système judiciaire.

La composante initiale, axée sur les tribunaux, a été considérablement avancée et gérée par le Département pour la modernisation de la justice. Ce département a mis en place un système national de base de données judiciaire pour superviser efficacement toutes les juridictions légales.

En ce qui concerne la deuxième composante, qui est l'objectif principal de ce projet, il a été constaté qu'il n'existe actuellement aucun système ou plate-forme complet qui aborde simultanément les cinq aspects de cette composante. Au contraire, les applications existantes ont tendance à répondre à des rôles individuels tels que les avocats, les huissiers de justice ou les notaires, au lieu de fournir une plate-forme intégrée pour tous les assistants juridiques.

Dans le domaine des assistants juridiques ou judiciaires, des distinctions sont faites entre les différents rôles, notamment ceux d'avocats, d'huissiers de justice, de notaires, de traducteurs et d'experts en justice.

En raison des défis liés à la collecte des exigences, en particulier les exigences fonctionnelles de tous les assistants judiciaires, nous nous sommes concentrés sur deux assistants de premier plan, en particulier les avocats et les notaires. Les trois assistants restants seront abordés comme une future perspective à court terme de ce projet. En revanche, en raison des contraintes de temps liées à la rédaction de ce mémoire de fin d'étude, tous les besoins des assistants juridiques choisis pour ce travail initial (avocats/notaires) n'ont pas été pleinement satisfaits. Les besoins abordés seront restreints et développés plus en détail à l'aide de diagrammes UML dans les sections suivantes, conformément à l'approche de modélisation adoptée.

3 La modélisation basée sur la démarche UML ^[16]

La modélisation est une démarche essentielle dans le développement de systèmes informatique tels que le système judiciaire, impliquant une représentation conceptuelle à l'aide de l'UML. La modélisation aide à comprendre la structure, les interactions et les fonctions du système, ce qui permet d'analyser les besoins, de détecter les erreurs et d'identifier les améliorations. Il représente visuellement le système, améliore la communication avec les parties prenantes, partage clairement les informations et favorise la collaboration.

Les modèles contribuent à la maintenance et à l'évolution du système et à la simplification des modifications, mises à jour et ajouts futurs. En conclusion, la modélisation améliore méthodiquement la compréhension, la communication, la prévision des problèmes et l'évolutivité du système, contribuant ainsi au succès global.

Le langage de modélisation unifié (UML) est un langage de modélisation visuelle omniprésent conçu pour l'architecture, la conception des systèmes informatique, englobant à la fois leur configuration et leur fonctionnement. Sa création visait à établir un langage commun, riche en sens et en syntaxe. UML découvre des applications qui transcendent le domaine du développement logiciel, en particulier dans la modélisation des flux de processus industriels.

L'UML ressemble aux plans utilisés dans d'autres domaines et englobe diverses catégories de diagrammes. Essentiellement, les diagrammes UML permettent de délimiter les limites, la configuration et la conduite du système et des entités existant au sein du système. Ils fournissent une représentation visuelle précise et détaillée des divers composants du système, facilitant ainsi la compréhension et l'interaction entre les parties prenantes.

4 Pourquoi UML ? Comme une approche de modélisation ^[16]

L'UML est largement utilisée comme approche de modélisation en raison de ses nombreux avantages. Il s'agit d'un langage de modélisation visuelle standardisé, utilisé dans le domaine du

logiciel, qui facilite la communication et la collaboration entre les membres de l'équipe de développement et les parties prenantes. UML offre une variété de diagrammes pour représenter les différentes facettes d'un système, tels que sa structure et son comportement, de manière organisée. Il est également adaptable à différents types de systèmes et compatible avec les outils de modélisation et les environnements de développement. En résumé, UML est choisi pour sa standardisation, sa représentation graphique claire, sa flexibilité et sa capacité à favoriser une approche méthodique et structurée de la modélisation.

5 La démarche basée sur UML : Les principaux modèles ^[17]

Une fois que nous adoptons l'approche UML, le développement d'un système doit être basé sur trois modèles complémentaires et ordonnés, qui sont les suivants :

- **Le modèle fonctionnel** : Il est représenté par des diagrammes de cas d'utilisation qui décrivent la fonctionnalité du système du point de vue de l'utilisateur. Ces diagrammes mettent en évidence les interactions entre les acteurs (utilisateurs) et les cas d'utilisation (fonctionnalités) du système.
- **Le modèle dynamique** : Il est représenté par des diagrammes d'interaction, des diagrammes d'états-transitions et des diagrammes d'activités. Ces diagrammes sont utilisés pour décrire le comportement interne du système, en mettant l'accent sur les interactions entre les objets, les transitions d'états et les processus.
- **Le modèle objet** : Il est représenté par des diagrammes de classes qui décrivent la structure du système en termes d'objets, d'attributs, d'associations et d'opérations. Ces diagrammes permettent de visualiser les différentes classes du système, leurs relations et leurs caractéristiques.

Ces modèles de système peuvent être visualisés à l'aide de deux types distincts de diagrammes:

- **Les diagrammes structurels** Ils permettent de représenter la structure statique du système, tels que les diagrammes de classes, les diagrammes d'objets et les diagrammes de déploiement. Ils mettent l'accent sur les entités du système et leurs relations.
- **Les diagrammes comportementaux** Ils permettent de représenter le comportement dynamique du système, tels que les diagrammes de séquence, les diagrammes d'états-transitions et les diagrammes d'activités. Ils mettent l'accent sur les interactions, les flux d'exécution et les changements d'états du système.

En utilisant ces différents modèles et diagrammes, il est possible d'avoir une vision complète et détaillée du système, en capturant à la fois sa fonctionnalité, sa structure et son comportement. Cela facilite l'analyse, la conception et la mise en œuvre du système de manière cohérente et rigoureuse.

6 Star-UML ^[35]

C'est un outil logiciel de modélisation UML (Unified Modeling Language) puisant utilisé principalement pour le développement de logiciels. Il permet aux développeurs et aux architectes logiciels de créer des diagrammes UML qui représentent graphiquement la structure et le comportement d'un système logiciel.

Dans les sections suivantes, nous présentons de manière concise ces modèles et les diagrammes correspondants :

7 Le modèle fonctionnel et l'identification des acteurs ^[18]

- **Avocat** : La profession d'avocat est une profession libérale et indépendante qui œuvre pour le respect et la sauvegarde des droits de la défense. Elle concourt l'œuvre de justice et au respect du principe de la primauté du droit. (**Art 2.**)
- **Notaire** : Le notaire est un officier public, mandaté par l'autorité publique pour accomplir ses missions. (**Art.3.**) Avant d'entrer en fonction, le notaire prête à l'audience de la Cour du lieu d'implantation de l'office.
- **Huissier de justice** : L'huissier de justice est un officier public mandaté par l'autorité publique, chargé de la gestion d'un office public pour son propre compte et sous sa responsabilité, son office est placé sous le contrôle du procureur de la République du lieu d'implantation de son office.
- **Traducteur-interprète officiel** : Les traducteurs-interprètes officiels ont la qualité d'officiers publics. Ils sont nommés à leur office par arrêté du ministre de la Justice.
- **L'expert judiciaire** : Les experts judiciaires, lors de leur première inscription sur les listes des Cours, prêtent le serment prévu par l'article 145 du code de procédure pénale.

7.1 Les tâches des assistants judiciaires dans le système

- Créer Un Compte ;
- Effectuer Un Abonnement ;
- Gérer Leur Application ;
- Faire Des Conversations Avec Les Plaideurs.

7.2 Diagramme de cas d'utilisation ^[19]

Le diagramme des cas d'utilisation (Use Case Diagram) constitue la première étape de l'analyse UML en : Modélisant les besoins des utilisateurs ; Identifiant les grandes fonctionnalités et limites du système ; Représentant les interactions entre le système et ses utilisateurs ;

Le tableau suivant expliquera les différents besoins qui seront ciblés.

Tableau 2.1- Les besoins ciblés pas la platform.

L'authentification	Permet aux assistants de justice de créer un compte, de se connecter et de choisir le plan d'abonnement qui leur convient.
Gestion de la clientèle.	Permet aux avocats et notaires d'ajouter, de modifier, de supprimer et de rechercher des clients dans le système.
Gestion des rendez -vous	Permet aux avocats et notaires d'ajouter, de modifier, de supprimer un rendez-vous dans le système.
Gestion de la messagerie	Permet aux avocats et notaires d'accepter un descution avec des plaideurs, d'envoyer des messages, de recevoir des messages et même de supprimer des messages.
Gestion des affaires	Le système permet aux avocats de gérer les affaires en ajoutant, modifiant, supprimant, recherchant et imprimant . Pour chaque affaire, ils peuvent : ajouter, modifier, supprimer, imprimer et rechercher des seances, ajouter, supprimer et imprimer des paiements, ajouter, supprimer et télécharger des documents, ajouter des décisions.
Gestion des contrats	Le système permet aux notaires de rédiger des contrats selon leurs besoins et de les exporter au format Word.

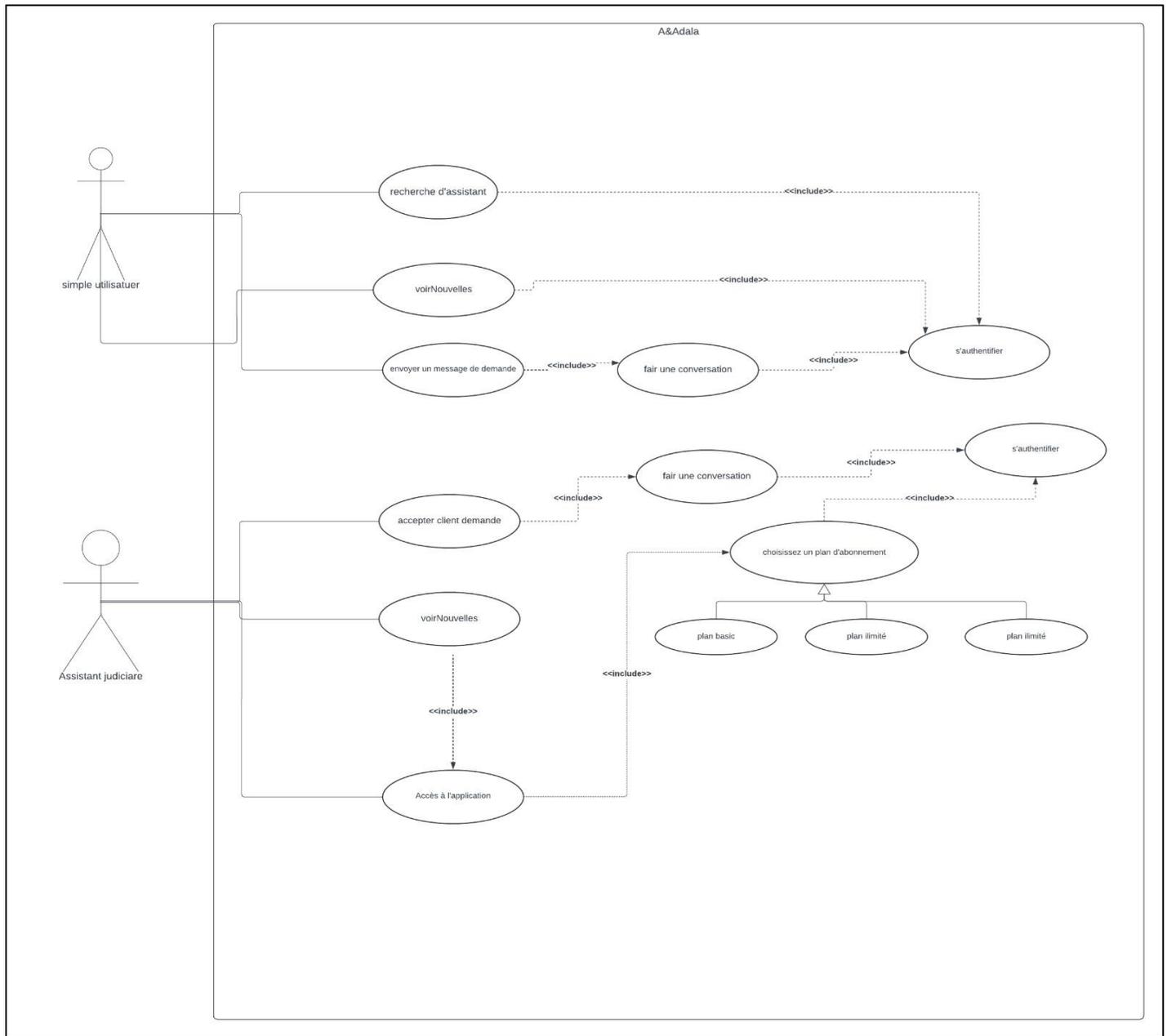


FIGURE 2.2-Diagramme de cas d'utilisation générale.

Model1::UseCaseDiagram1

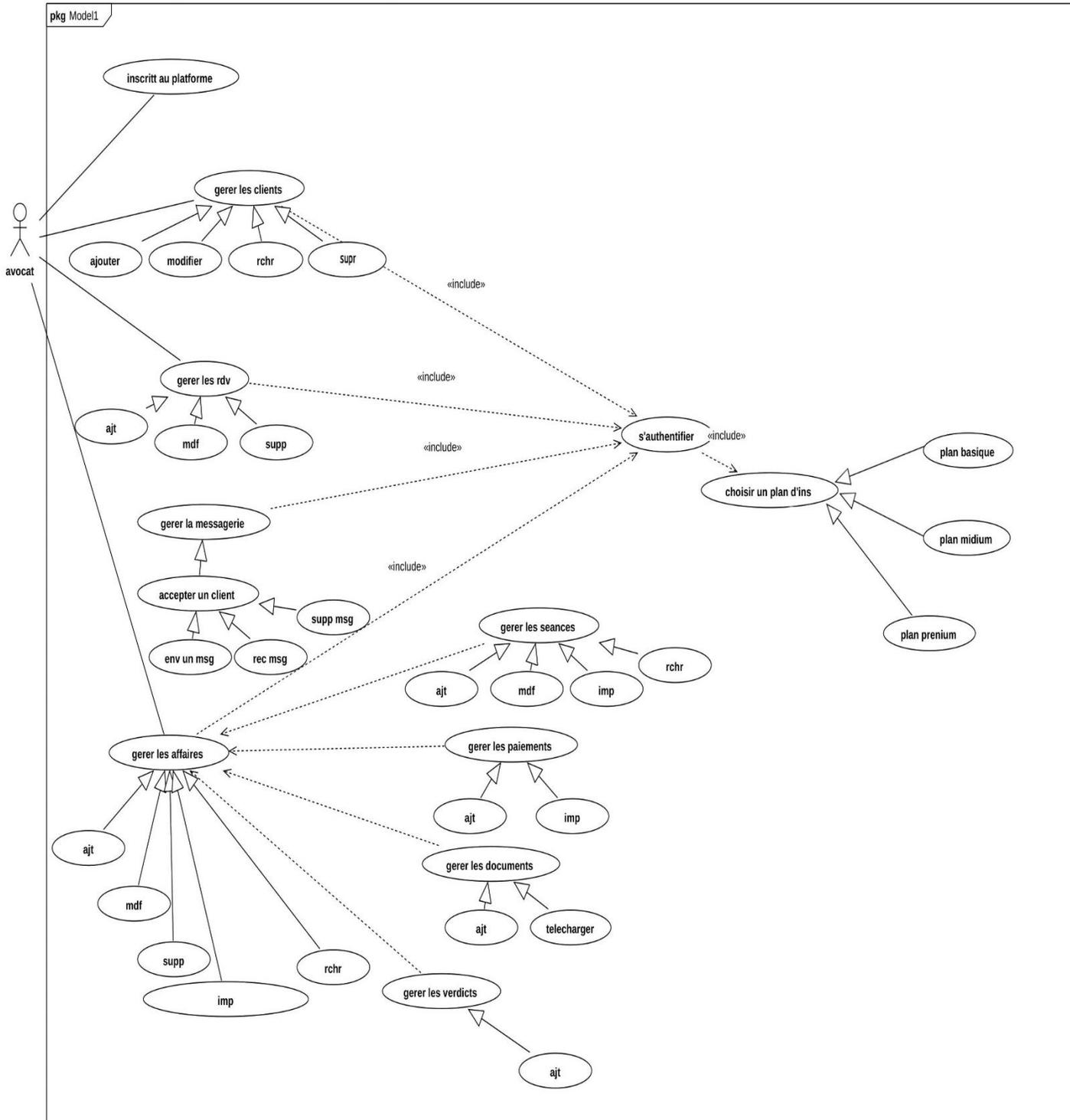


FIGURE 2.3-Diagramme de cas d'utilisation d'avocat.

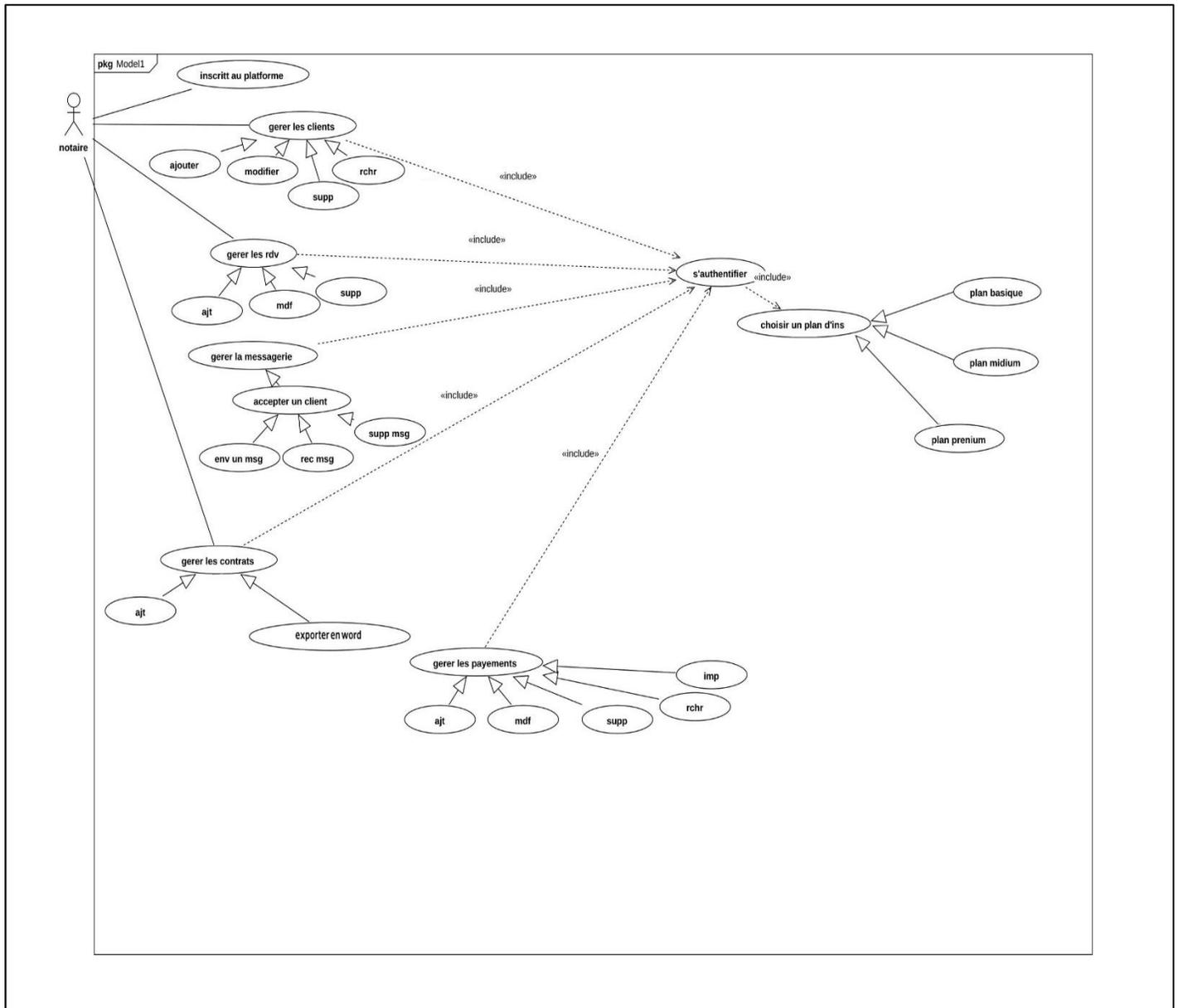


FIGURE 2.4-Diagramme de cas d'utilisation du notaire.

Les figures 2.3 (avocat) et 2.4 (notaire) présentent respectivement un aperçu des différentes fonctions des systèmes d'avocat et de notaire, représentées par des diagrammes de cas d'utilisation.

7.3 Diagramme de séquence [20]

Un diagramme de séquence est un type de diagramme d'interaction car il décrit comment plusieurs objets fonctionnent ensemble et dans quel ordre.

Dans notre système, nous allons illustrer deux scénarios principaux :

- **Le scénario de l'authentification et l'abonnement** Une fois que l'assistant judiciaire (dans notre cas, un avocat) s'authentifie, il doit choisir un plan d'abonnement qui lui convient. Après avoir fait son choix, il reçoit un message de confirmation dans sa boîte email contenant les informations sur l'abonnement. Il peut ensuite accéder aux fonctionnalités du système.
- **Le scénario de messagerie entre plaideur et assistants judiciaires** Lorsqu'un plaideur recherche un assistant judiciaire, dans notre cas un notaire, il peut lui envoyer un message. Le notaire peut accepter ou rejeter ce message. Si le message est accepté, ils peuvent alors commencer une discussion en s'envoyant et recevant des messages. [21]

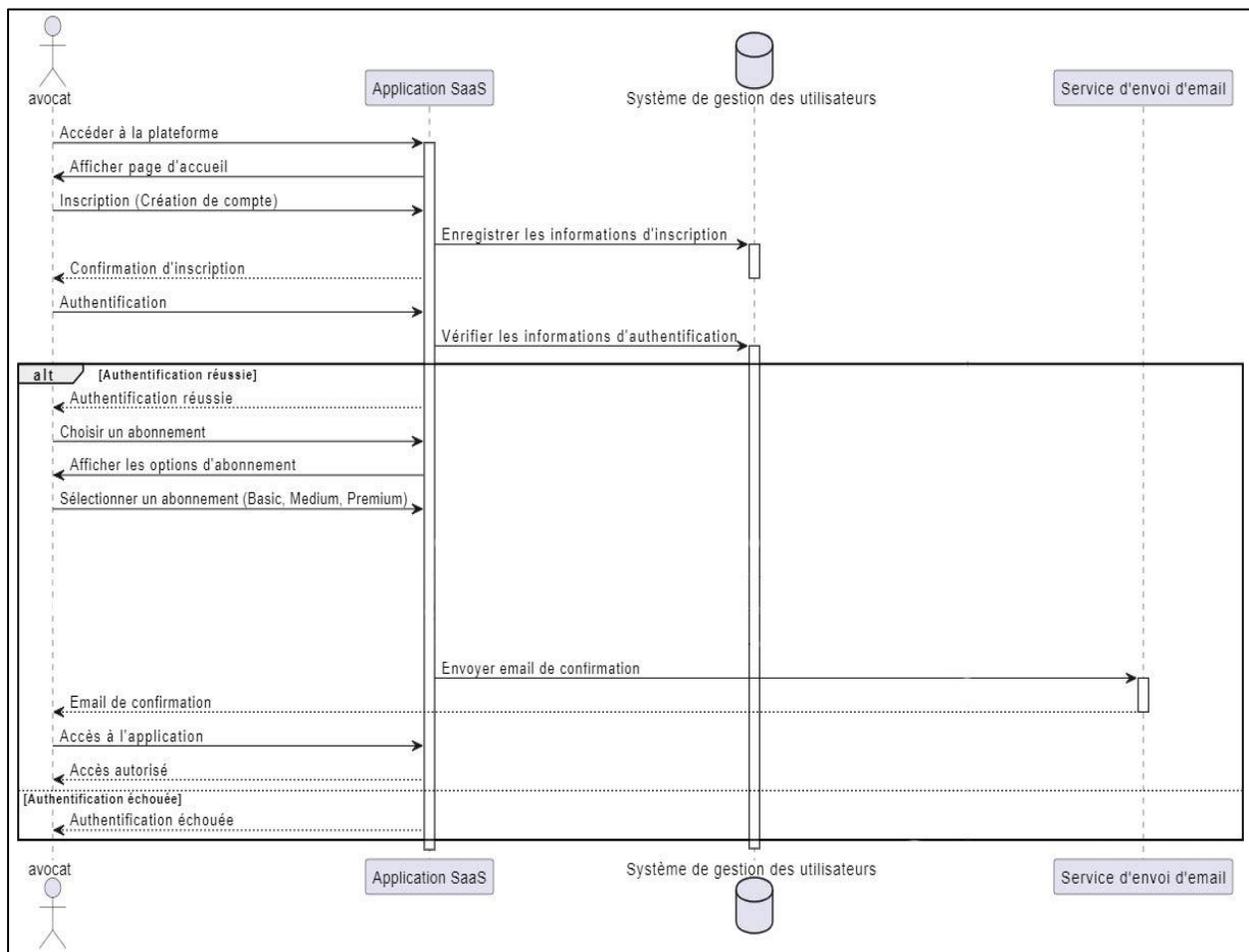


FIGURE 2.5-Diagramme de séquence d'avocat.

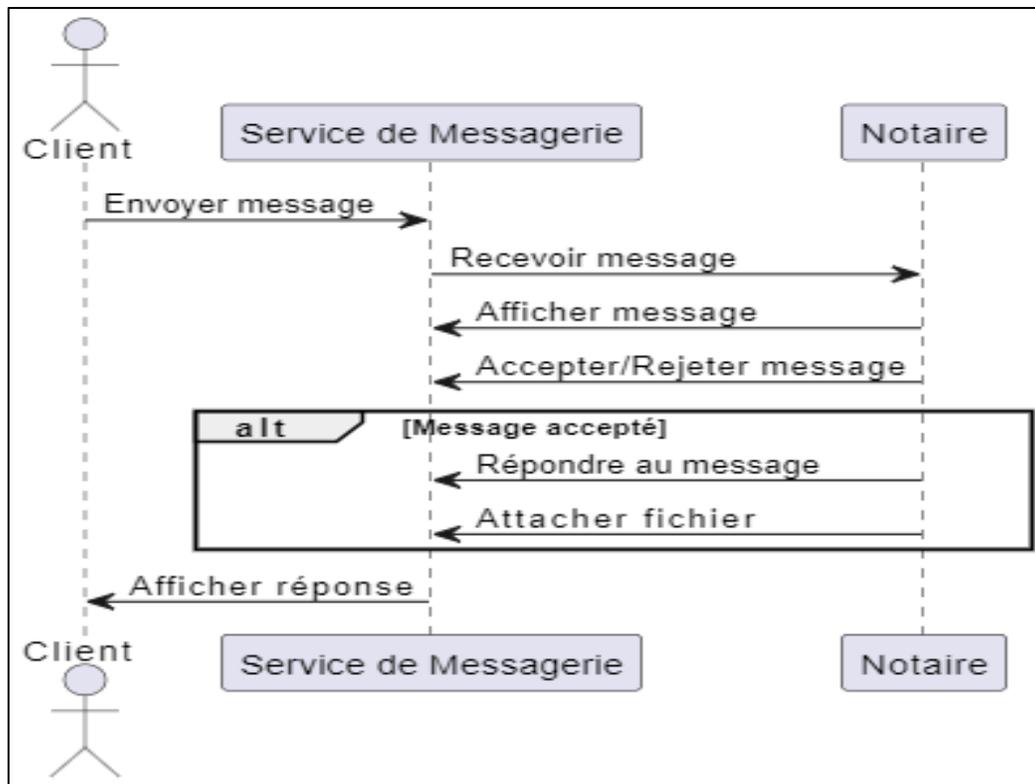


FIGURE 2.6-Diagramme de séquence de service de messagerie.

7.4 Diagramme de classe ^[21]

Il représente les classes intervenant dans le système. Le diagramme de classe est une représentation statique des éléments qui composent un système et de leurs relations

- **Le concept de la classe**

Une classe est une description d'un groupe d'objets partageant un ensemble commun de propriétés (les attributs), de comportements (les opérations) et de relations avec d'autres objets (les associations et les agrégations).

Les classes sur lesquelles se porte notre solution sont :

Tableau 2.2- Classes sur lesquelles se porte notre application desktop.

Classe	Attribut		Méthode
	Champ	Type	
Avocat	Id avocat prénom nom nom utilisateur avocat licence region pratique annees experience numero telephone email mot de passe	Numérique Caractère Caractère Caractère Caractère Caractère Numérique Numérique Caractère Caractère	Gerer les clients Gerer les rendez-vous Gerer la messagerie Gerer les affaires Gerer les seance Gerer les verdicts
Notaire	Id_notaire prenom nom nom_utilisateur notaire_licence region_pratique annees_experience numero_telephone email mot_de_passe	Numérique Caractère Caractère Caractère Caractère Caractère Numérique Numérique Caractère Caractère	Gerer les clients Gerer les rendez-vous Gerer la messagerie Gerer les contrats Gerer les paiements
Simple utilisateur	Id_notaire prenom nom nom_utilisateur email mot_de_passe	Numérique Caractère Caractère Caractère Caractère Caractère	Recherche des assistants Envoyer des messages

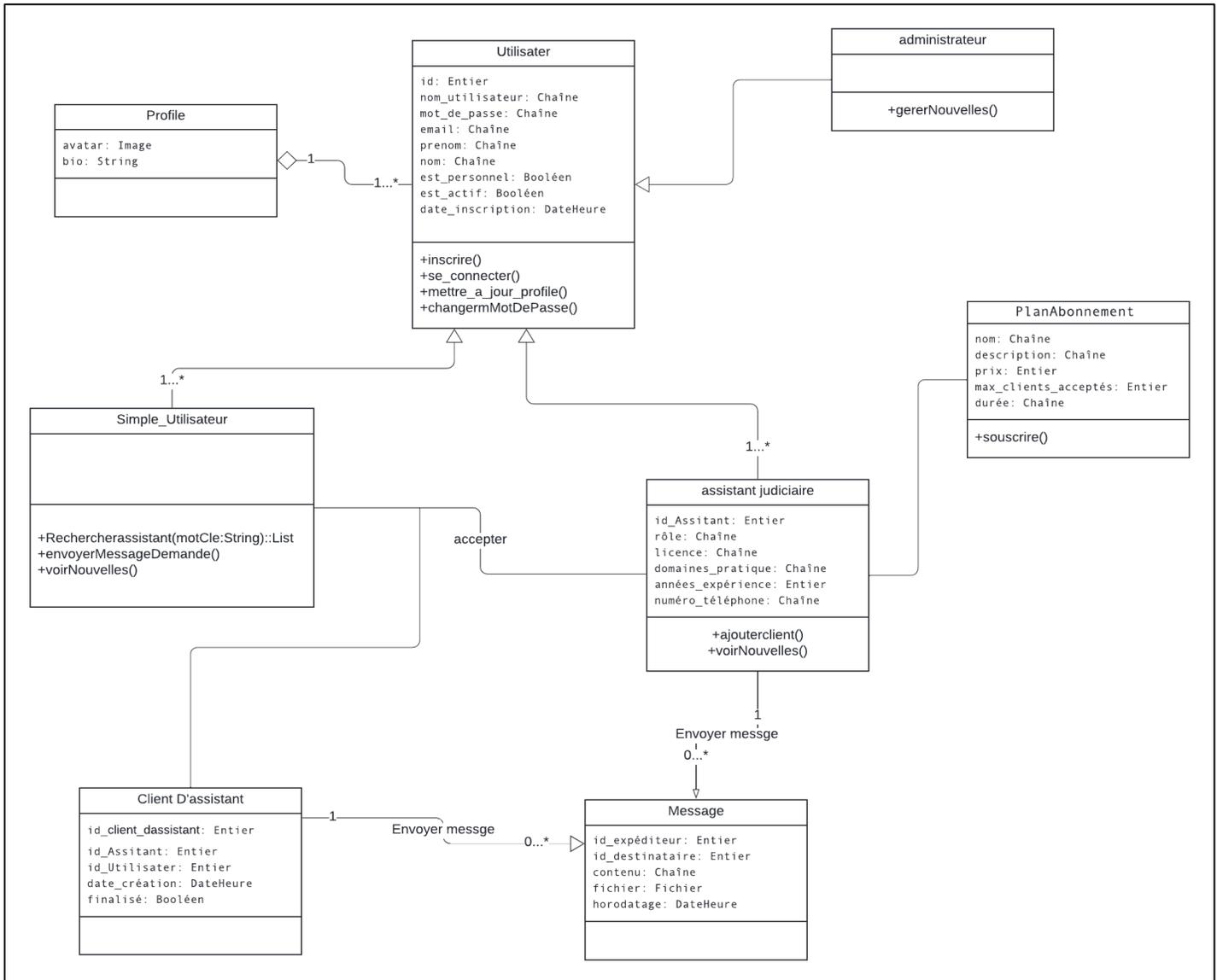


FIGURE 2.7-Diagramme de classe générale.

8 Conclusion

Dans ce second chapitre, nous décrivons brièvement la partie conception du système judiciaire étudié, en mettant l'accent sur le rôle crucial joué par les assistants juridiques tels que les avocats et les notaires. Notre modélisation repose principalement sur l'utilisation du langage UML afin de saisir de manière précise les différents aspects et éléments de ce système. Cette démarche rigoureuse vise à obtenir une compréhension approfondie et une représentation conceptuelle du système judiciaire, ce qui facilitera son analyse, sa conception et sa mise en œuvre future, qui seront abordées dans le prochain chapitre.

Chapitre 03

Implémentation et réalisation

1 Introduction

La croissance indéniable de l'utilisation du cloud computing et du modèle SaaS dans le domaine de l'informatique est évidente. Cependant, mettre en place une plateforme à la fois multiservices et multi-emplacements avec des stratégies d'abonnement représente un défi complexe, tant sur le plan de la conception que de la programmation. Pour relever ce défi, différents frameworks et bibliothèques ont été adoptés afin de faciliter la réalisation de la plateforme ciblée avec un caractère d'abonnement. Dans ce chapitre, nous présentons en détail l'architecture adoptée pour rendre cette plateforme typique compatible avec la multi-location et l'abonnement. Nous mettrons également l'accent sur les outils matériels et logiciels utilisés, en mettant en avant les différentes interfaces de cette plateforme, communément appelée e-A&A.

2 Les outils d'implémentation et de réalisation

2.1.1 Environnement matériel

Nous avons mis en œuvre la plateforme e-A&A en utilisant un ordinateur "Dell Précision 3480" avec une configuration détaillée dans le tableau suivant :

TABLE 3.1 – Les caractéristiques de l'ordinateur

Processeur	Processeurs Intel® Core™ i7 de 13 ^e génération
Disque dur	2 To
Mémoire Vive	32GB 4800 MHz ;
Système d'exploitation	Windows 11
Carte graphique	RTX A500

2.1.2 Environnement logiciel

Au cours de cette section, nous présenterons succinctement les divers outils logiciels utilisés lors de la mise en œuvre de notre plateforme.

a) Framework ^[23]

Le Framework ou infrastructure logicielle facilite le travail des développeurs. Il prend en quelque sorte la forme d'un canevas ou d'un squelette de programme qui pourra ensuite être enrichi afin de répondre aux besoins du client, il existe des frameworks pour tout type de projet qu'il s'agisse du développement d'un site web, d'un jeu, d'une application mobile ou encore la création d'un module d'extension.

Cette infrastructure logicielle offre un ensemble de composants logiciels et d'outils (codes prédéfinis, grille CRUD, arborescence, bibliothèque, etc.) capables d'uniformiser et de faciliter le développement d'applications, sites web, etc.

- **Django** ^[24] un framework web Python open source de haut niveau qui encourage un développement rapide et une conception propre et pragmatique. Conçu par des développeurs expérimentés, il prend en charge une grande partie des problèmes liés au

développement web, ce qui permet de concentrer sur l'écriture des applications sans avoir à réinventer la roue.

- **Bootstrap** ^[25] un framework open-source développé par Twitter. Il est principalement utilisé pour la création de sites web et d'applications web réactives et mobiles. Bootstrap est constitué d'une collection de syntaxes qui remplissent des fonctions spécifiques (html, css, js). Bootstrap avec des propriétés CSS et des éléments HTML peut fonctionner parfaitement, il a besoin de jQuery pour créer un design réactif. Sinon, vous ne pouvez utiliser que les parties nues et statiques du CSS.

b) Langages informatiques

- **HTML** ^[26] (Hyper Text Markup Language) dérivé du Standard Generalized Markup Language est un langage de balisage qui définit la structure de votre contenu, il permet de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias et des programmes informatiques. Il se compose d'une série d'éléments, utilisés pour entourer, ou envelopper, les diverses parties du contenu pour les faire apparaître ou agir d'une certaine façon. Il est souvent utilisé conjointement avec des langages de programmation (JavaScript) et des formats de présentation (feuilles de style en cascade).
- **CSS** ^[27] (Cascading Style Sheets) est l'abréviation de Cascading Style Sheets qui signifie feuilles de style en cascade. Le mot cascade rappelle ici que les styles peuvent être classés selon différents degrés d'importance. CSS permet donc de contrôler de manière précise comment un site web est présenté, il correspond à un langage informatique permettant de mettre en forme des pages web (HTML ou XML). Ce langage est donc composé des fameuses « feuilles de style en cascade » également appelées fichiers CSS (.css) et contient des éléments de codage.
- **JAVASCRIPT** ^[28] est un langage de développement informatique, et plus précisément un langage de script orienté objet. On le retrouve principalement dans les pages Internet. Il permet de créer des fenêtres spécifiques, contrôler les données saisies dans les formulaires, redimensionner certains objets, rediriger des liens entre autres, ainsi qu'introduire des petites animations ou des effets sur une page web ou HTML.
- **Python** ^[29] créé par Guido van Rossum en 1991 est un langage de programmation interprété de haut niveau, utilisé largement dans divers domaines tels que le développement web, l'analyse de données, l'intelligence artificielle, l'automatisation, les applications scientifiques, etc. Python se distingue par sa syntaxe claire et concise, qui favorise la lisibilité et la facilité de compréhension du code, ces caractéristiques clés de Python incluent la Lisibilité, Polyvalence, Vaste écosystème, Portabilité et Communauté active.

c) Editeur et Système de gestion base de données (SGBD).

- **IDE** ^[30] (**Visual Studio Code**) un éditeur de code simplifié, qui est gratuit et développé en open source par Microsoft. Il fonctionne sous Windows, mac OS et Linux. Il fournit aux développeurs à la fois un environnement de développement intégré avec des outils permettant de faire avancer les projets techniques, de l'édition, à la construction, jusqu'au débogage.
- **SGBD** ^[31] (**SQLite3**) créé au début des années 2000 par D. Richard Hipp, est l'utilitaire en ligne de commande fourni avec la bibliothèque du système de gestion de base de données (SGBD) SQLite. Il repose sur une écriture en C, un langage de programmation impératif, et sur une accessibilité via le langage SQL.

Le tableau ci-dessous résume les différents outils ainsi que leurs différentes versions.

TABLE 3.2 – Tools et versions

Tools	Versions
Django	3.2.7
Bootstrap	3.3.7
HTML	5
CSS	----
JAVASCRIPT	ES5
Python	3.11.5
Visual Studio Code	1.89.1
SQLite3	3.31.1

3 Développement de la plateforme e-A&ADALA : Architecture & Interfaces

3.1 Architecture Générale

La figure suivante présente de manière exhaustive l'architecture générale de la plateforme e-A&ADALA de manière claire et précise. Initialement, lorsque les utilisateurs émettent une requête, celle-ci est transmise au framework Django. Ensuite, à l'intérieur du framework, l'architecture bien connue du Modèle-Vue-Contrôleur (MVC) a été adoptée, avec une variante spécifique appelée MTV.

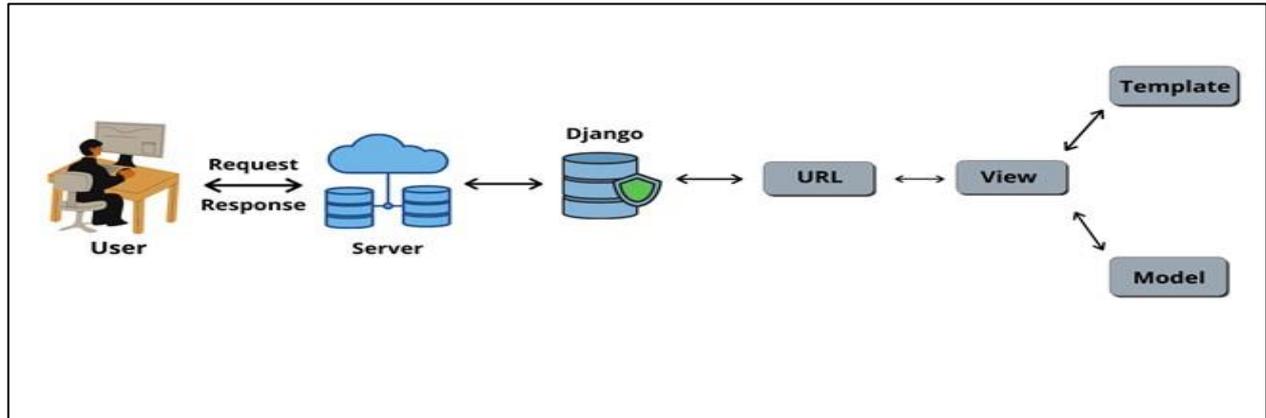


Figure 3.1-Architecture générale.

a. Architecture MTV ^[33]

Il s'agit d'une variante du modèle MVC où le mot-clé "Template" remplace le "Controller". Bien que le modèle ne fonctionne pas exactement comme le contrôleur et possède des propriétés différentes, il joue un rôle essentiel dans l'architecture. Le modèle contient la structure logique du projet et agit comme un intermédiaire entre la base de données et la vue. Il gère les données et fournit une définition de la manière dont les formats de données provenant de la vue sont stockés dans la base de données, et vice versa. En d'autres termes, il assure la conversion et le transfert des informations entre la base de données et la vue dans un format affichable.

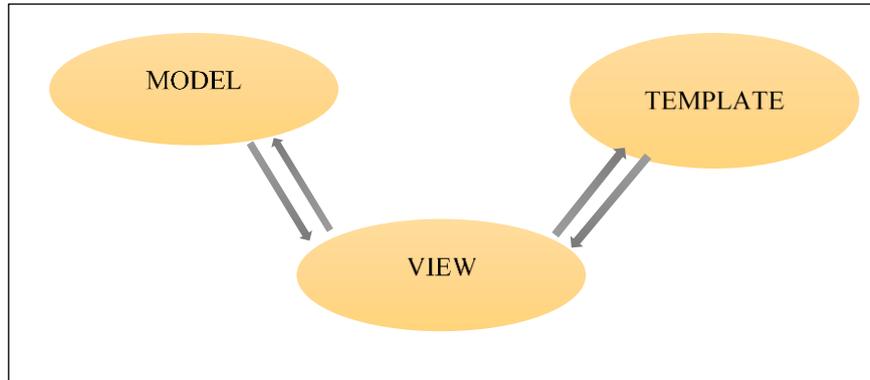


Figure 3.2 -Architecture MTV.

b. Le processus de fonctionnement des requêtes

Lorsqu'un client envoie une requête HTTP, celle-ci est traitée par le contrôleur. Le contrôleur, à son tour, communique avec le modèle, qui interagit avec la base de données pour récupérer les données requises. Une fois les données récupérées, elles sont transmises à la vue, qui se charge de les afficher en HTML dans le navigateur. Ainsi, le processus implique une communication entre le client, le contrôleur, le modèle et la vue pour traiter la requête et présenter les résultats à l'utilisateur.

3.2 Présentation des différentes interfaces de la plateforme

En raison de contraintes de temps, nous avons traité et développé uniquement deux assistants judiciaires lors de l'élaboration de la plateforme e-A&A, à savoir l'assistant avocat et notaire. Les trois autres assistants seront envisagés comme des perspectives à court terme pour ce projet.

Les interfaces qui seront présentées par la suite sont exclusivement dédiées à la création de compte, à la souscription et à la gestion des affaires spécifiquement pour les deux assistants, à savoir l'assistant avocat et l'assistant notaire.

a-Page d'accueil

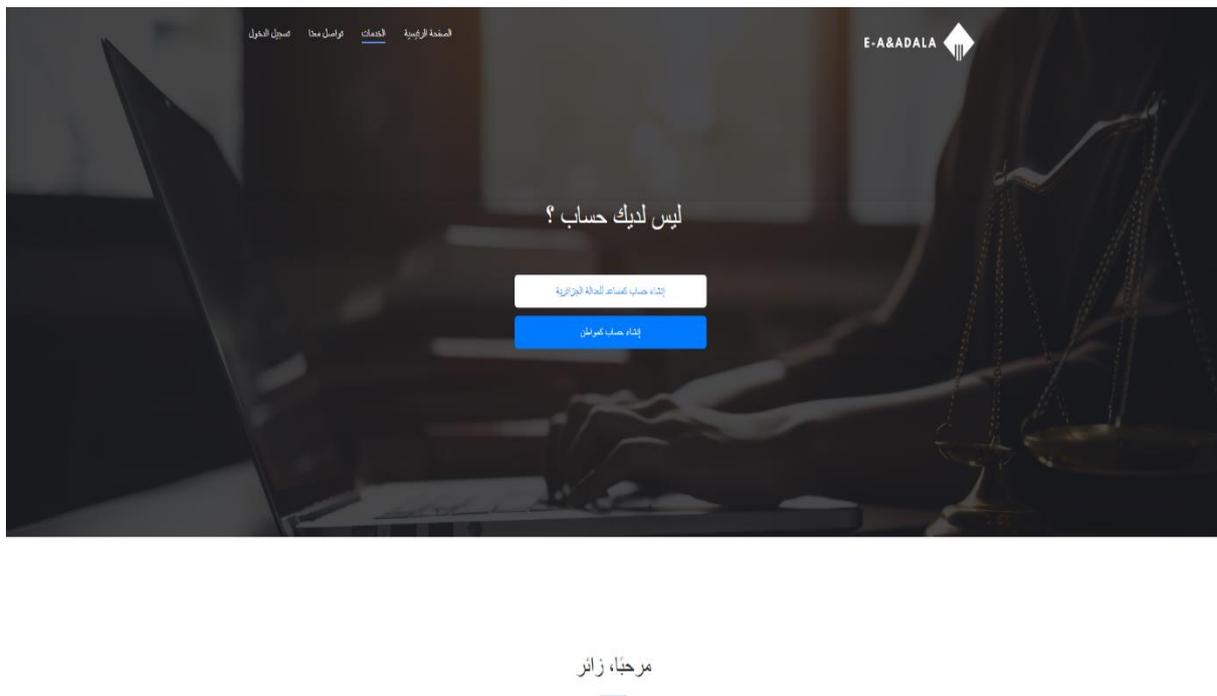


Figure 3.3- page d'accueil principale.

b-Création de compte

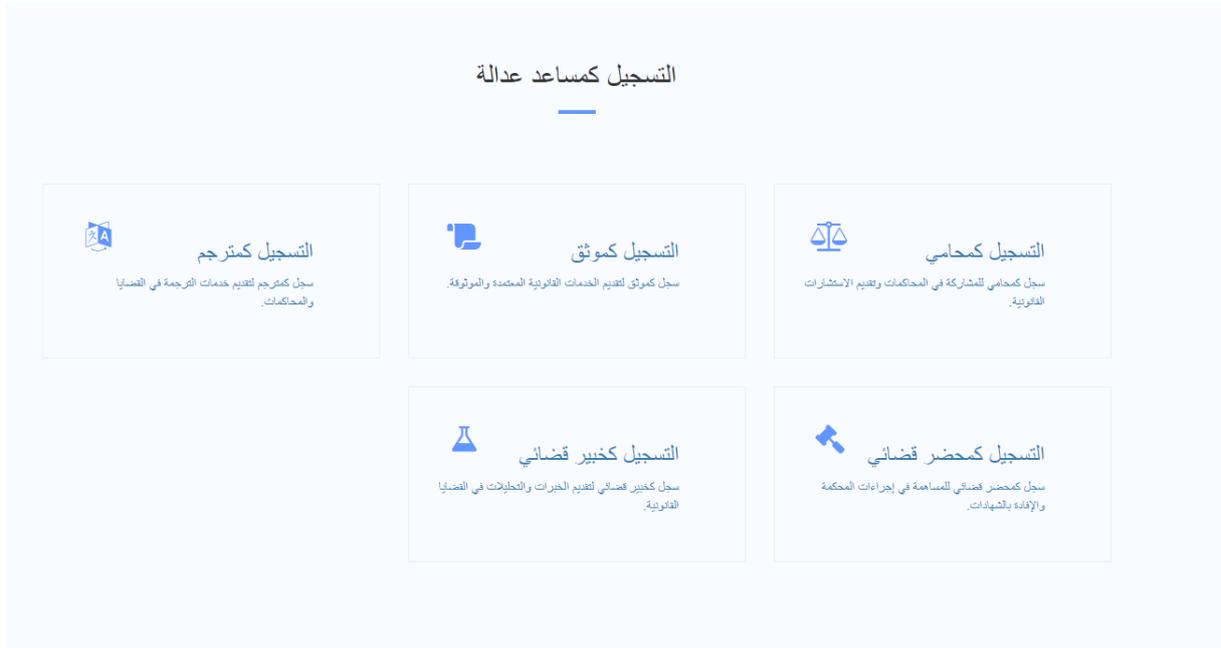


Figure 3.4-Interface d'inscription.

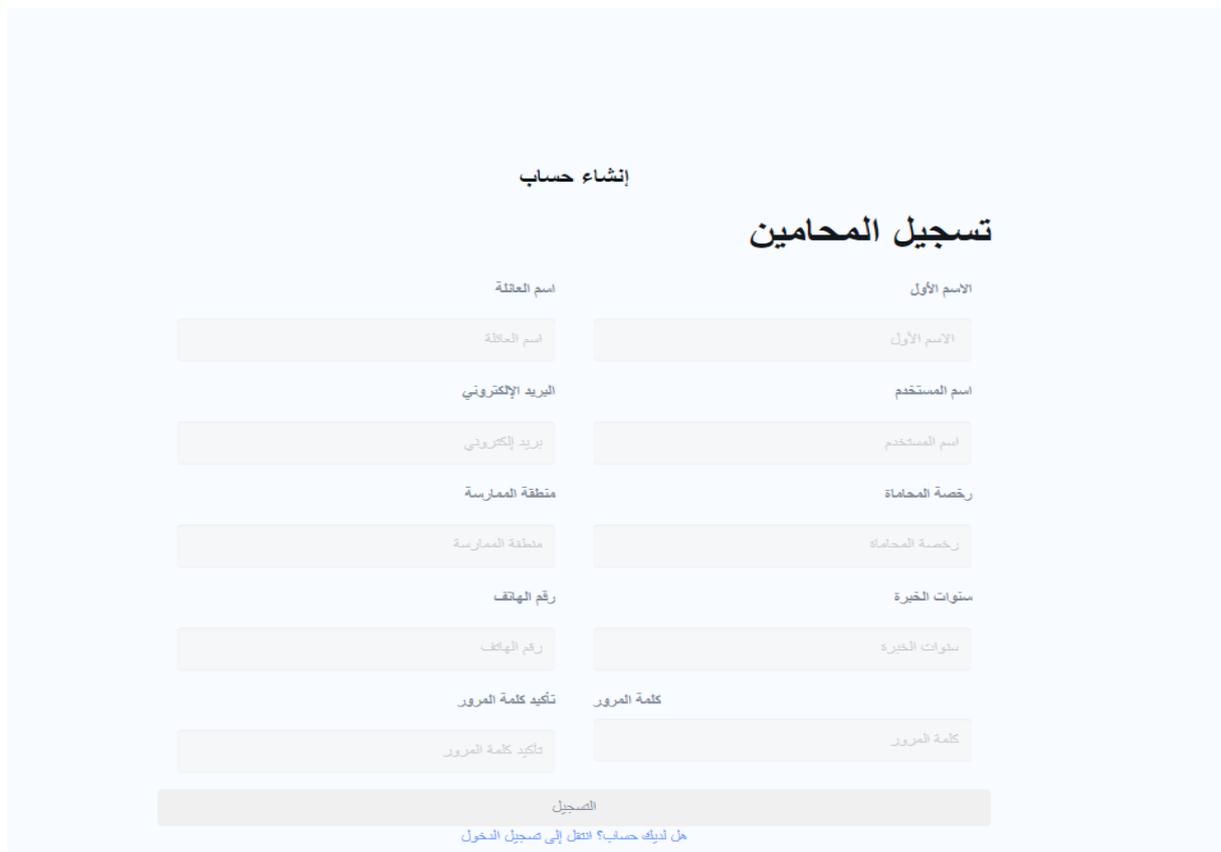


Figure 3.5- Création de compte.

c-Login

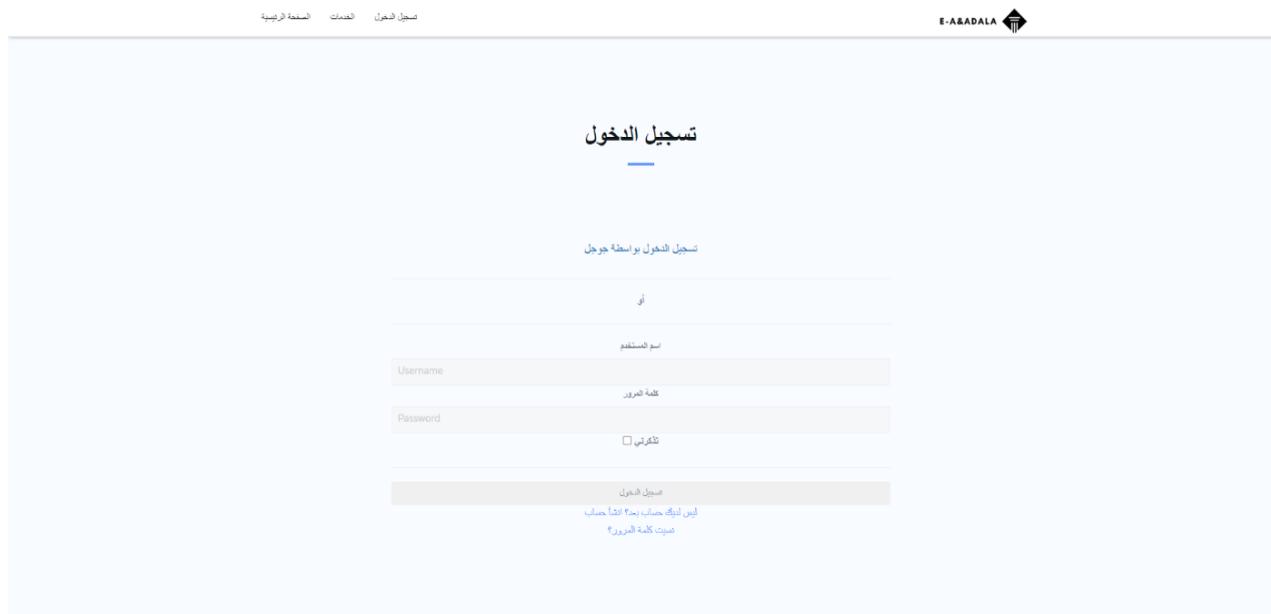


Figure 3.6- Interface connexion.

d-Interface pour les Plans d'abonnement et Tarification

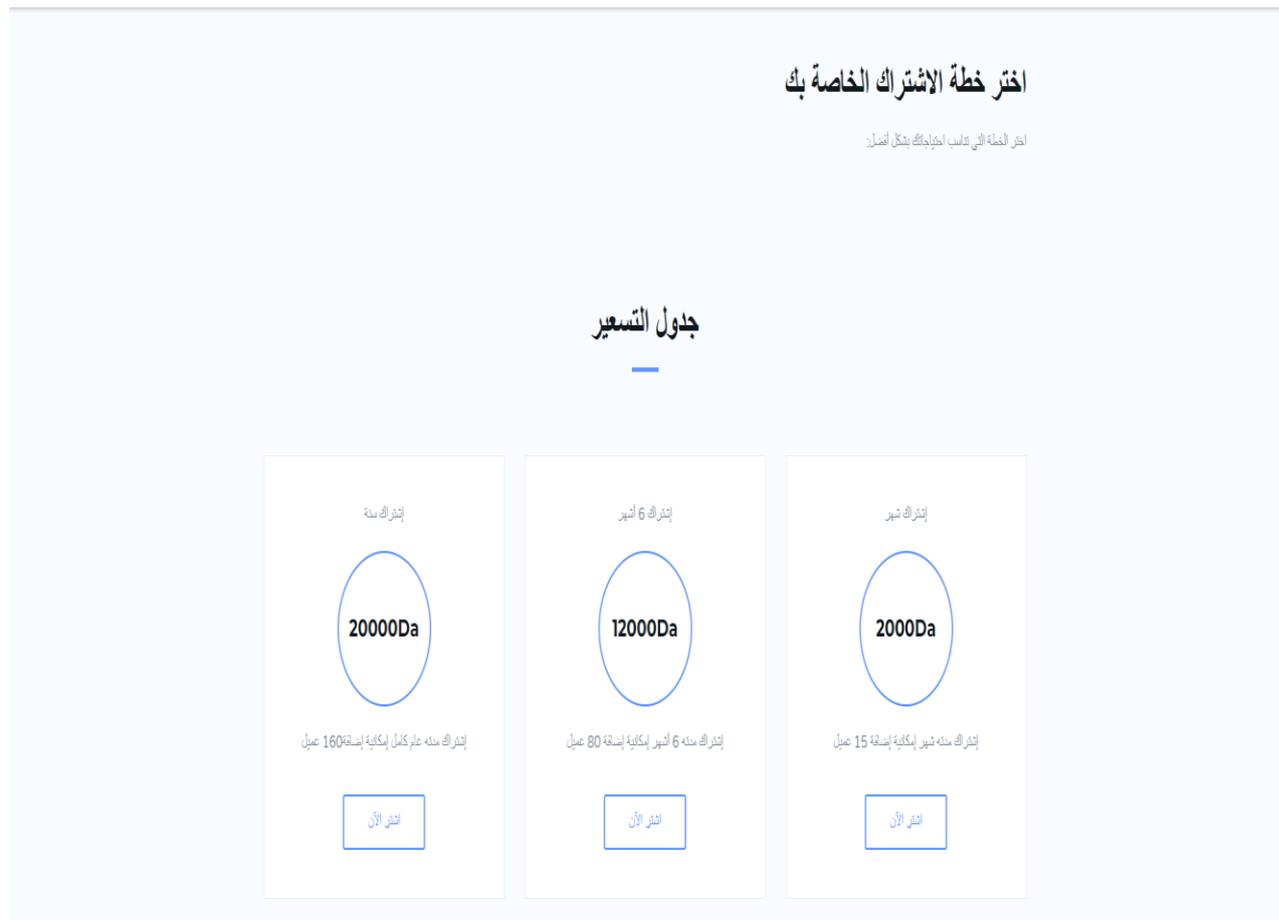


Figure 3.7 Table de tarifs d'abonnement.

e -Page d'accueil (Assistant judiciaire (Avocat))

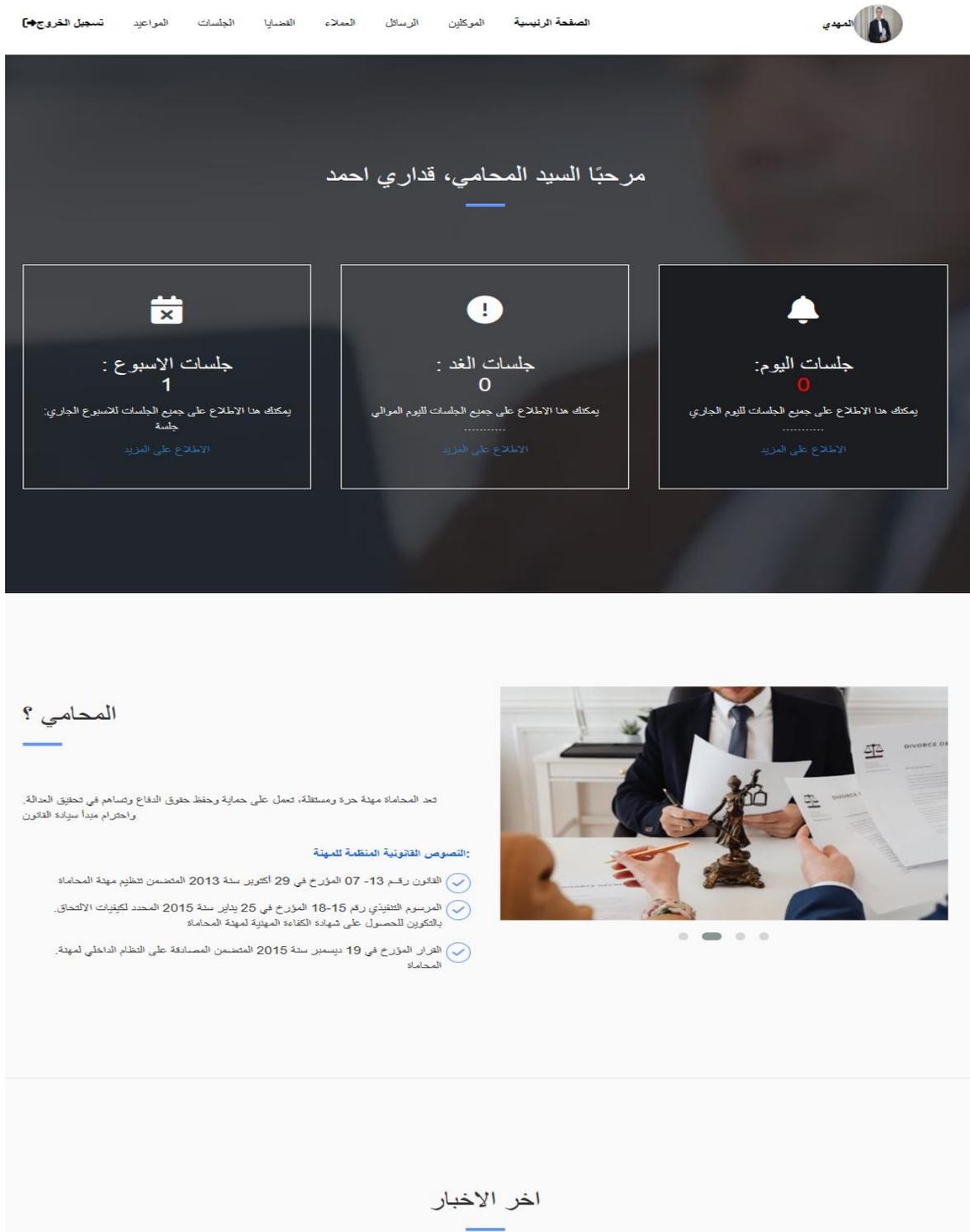


Figure 3.8- page d'accueil de l'application de l'avocat.

f-Gestion des affaires

تمت الحذف بنجاح !

صفحة الرئيسية الموكلين الرسائل العملاء القضايا الجلسات المواعيد تسجيل الخروج

طباعة الجدول

اضافة قضية +

Copy CSV PDF Search:

رقم القضية	القضية	الموكل	الخصم	مخامي الخصم	مقر المحاكمة	اجمالي اتعاب القضية	تاريخ القيد	العمليات
298	الموضوع: عتف اسري : المرحلة : نقض الحالة: تحت الدراسة	بوعلام قادة جاني	الاسم الكامل: اشرف شريفي fd 653467653	الاسم الكامل: اشرف شريفي	مجلس قضاء الشلف محكمة الشلف / القسم : الغرفة: إداري	1233333.0 دينار جزائري	2024-04-28	عرض التفاصيل
24	الموضوع: موضوع القضية : المرحلة : جزئي الحالة: معلقة	بوعلام قادة جاني	الاسم الكامل: اشرف شريفي wrwr 424424242	الاسم الكامل: اشرف شريفي	مجلس قضاء أدرار محكمة رقان / القسم : الغرفة: جنائي	1233333.0 دينار جزائري	2024-05-03	عرض التفاصيل
24	الموضوع: موضوع القضية : المرحلة : ابتدائي الحالة: معلقة	جيلالي امحمد مدعى عليه	الاسم الكامل: hf wrwr 4444444444444444	الاسم الكامل: اشرف شريفي	مجلس قضاء أدرار محكمة رقان / القسم : الغرفة: جنائي	1233333.0 دينار جزائري	2024-05-03	عرض التفاصيل
24	الموضوع: عتف اسري : المرحلة : جزئي الحالة: منتهية	بوعلام قادة مدعى	الاسم الكامل: اشرف شريفي wrwr 9904221	الاسم الكامل: اشرف شريفي	مجلس قضاء أدرار محكمة رقان / القسم : الغرفة: جنائي	1233333.0 دينار جزائري	2024-05-04	عرض التفاصيل
24	الموضوع: موضوع القضية : المرحلة : نقض الحالة: تحت الدراسة	بوعلام قادة مدعى	الاسم الكامل: اشرف شريفي wrwr 9904221	الاسم الكامل: اشرف شريفي	مجلس قضاء الشلف محكمة تنس / القسم : الغرفة: جنائي	1233333.0 دينار جزائري	2024-05-26	عرض التفاصيل
20212	الموضوع: موضوع القضية : المرحلة : ابتدائي الحالة: تحت الدراسة	جيلالي امحمد مدعى	الاسم الكامل: اشرف شريفي تيارت 9904221	الاسم الكامل: اشرف شريفي	مجلس قضاء أدرار محكمة تميمون / القسم : الغرفة: جنائي	1233333.0 دينار جزائري	2024-05-11	عرض التفاصيل

Showing 1 to 6 of 6 entries

Previous 1 Next

Figure 3.9- Gestion des affaires.

g-Informations sur les affaires

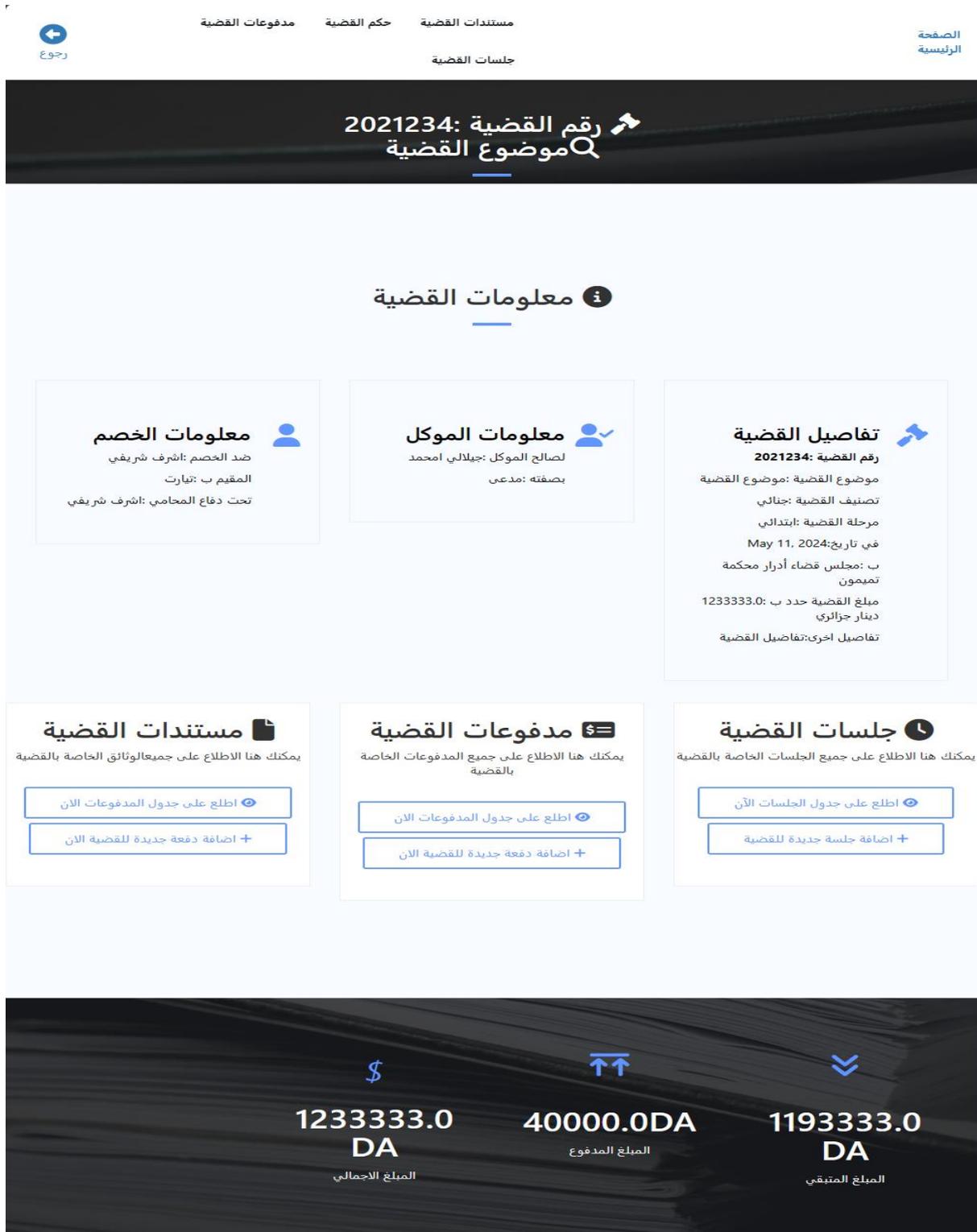


Figure 3.10- Informations divers sur les affaires.

h-Suivre des affaires

\$
1233333.0
DA
المبلغ الاجمالي

↑↑
40000.0DA
المبلغ المدفوع

⇩
1193333.0
DA
المبلغ المتبقي

جلسات القضية

+ اضافة جلسة جديدة

عنوان الجلسة	تاريخ الجلسة	تفاصيل الجلسة
جلسة تصالح	10:09 2024-05-17	تقديم الادلة

+ اضافة دفعة جديدة

المبلغ المدفوع	تاريخ و توقيت الدفع
40000.0 دينار جزائري	18:10 2024-05-11

+ اضافة ملف جديد

الملف
وثيقة الأثبات تحميل الوثيقة
الوثيقة الثانية تحميل الوثيقة

الاحكام الصادرة

+ اضافة حكم للقضية

لم يتم تسجيل اي حكم حتى الان .

Figura 3.11- Suivre de l'affaire.

i. Page d'accueil (Assistant judiciaire (Notaire))

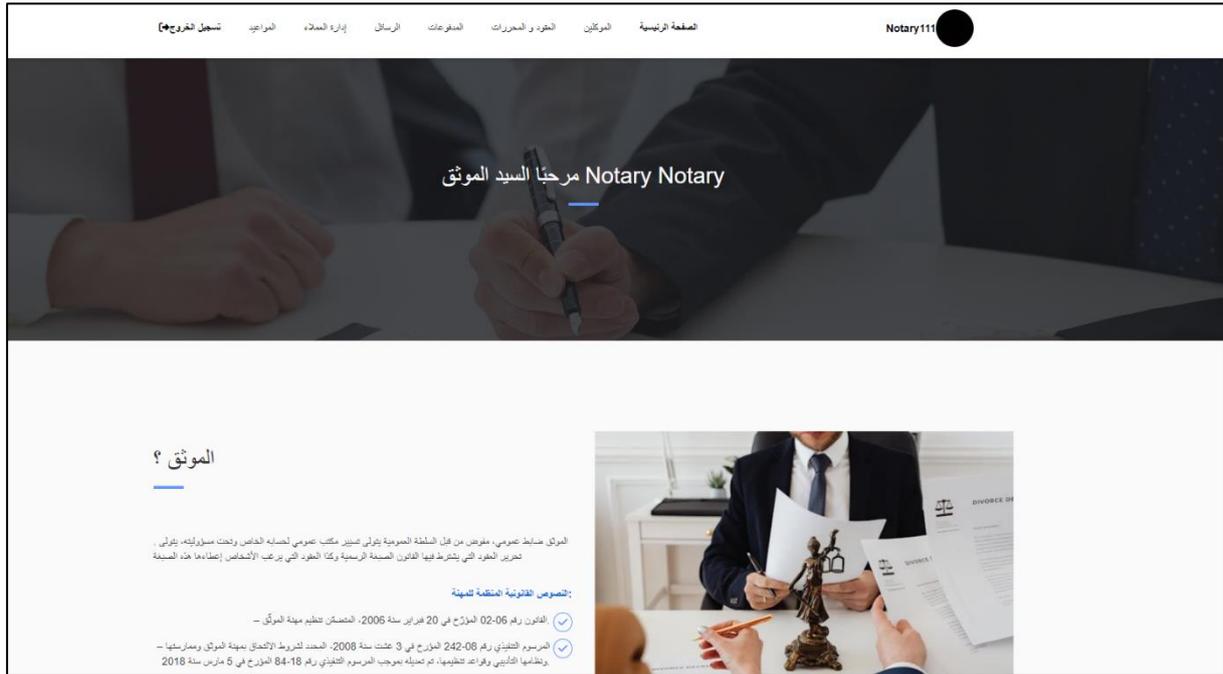


Figure 3.12 : Page d'accueil.

j. Contrats disponibles

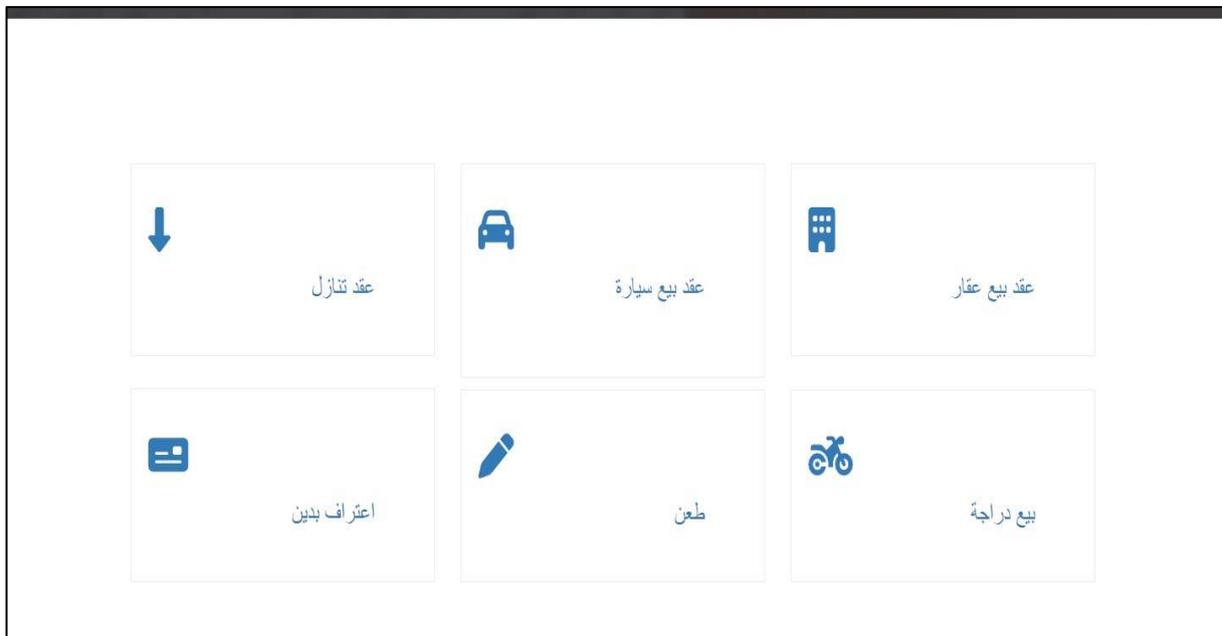


Figure 3.13- Les contrats disponibles.

k. Saisie du contrat

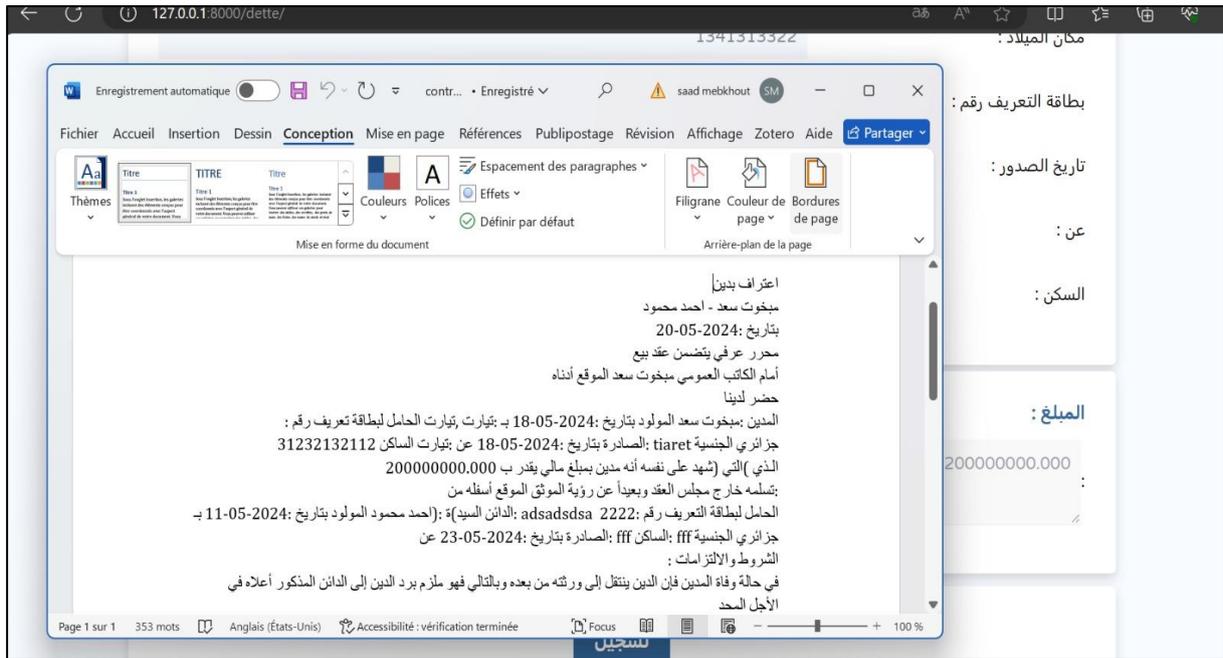


Figure 3.14- Saisie du contrat.

e. Gestion des clients

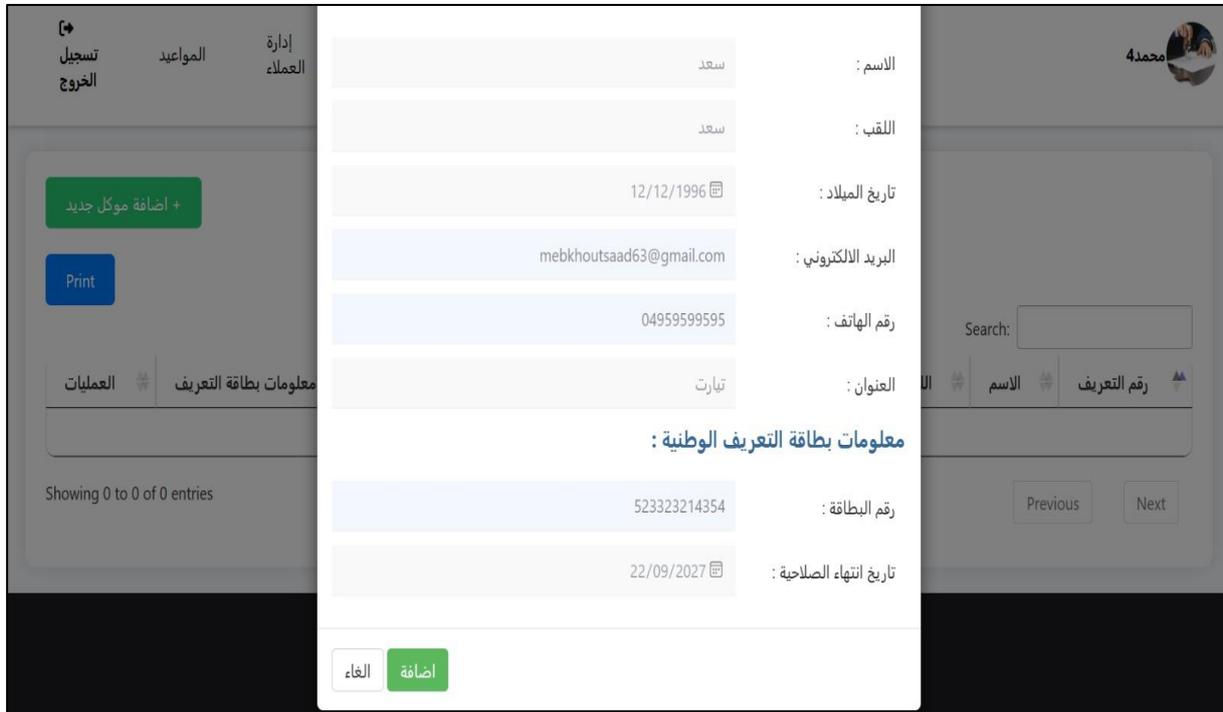


Figure 3.15- Gestion des clients.

1. Payment



Figure 3.16- Interface de paiement.

m. Gestion de rendez-vous



Figure 3.17- Gestion des rendez-vous.

n. Page d'accueil client de l'interface Plaideur standard.

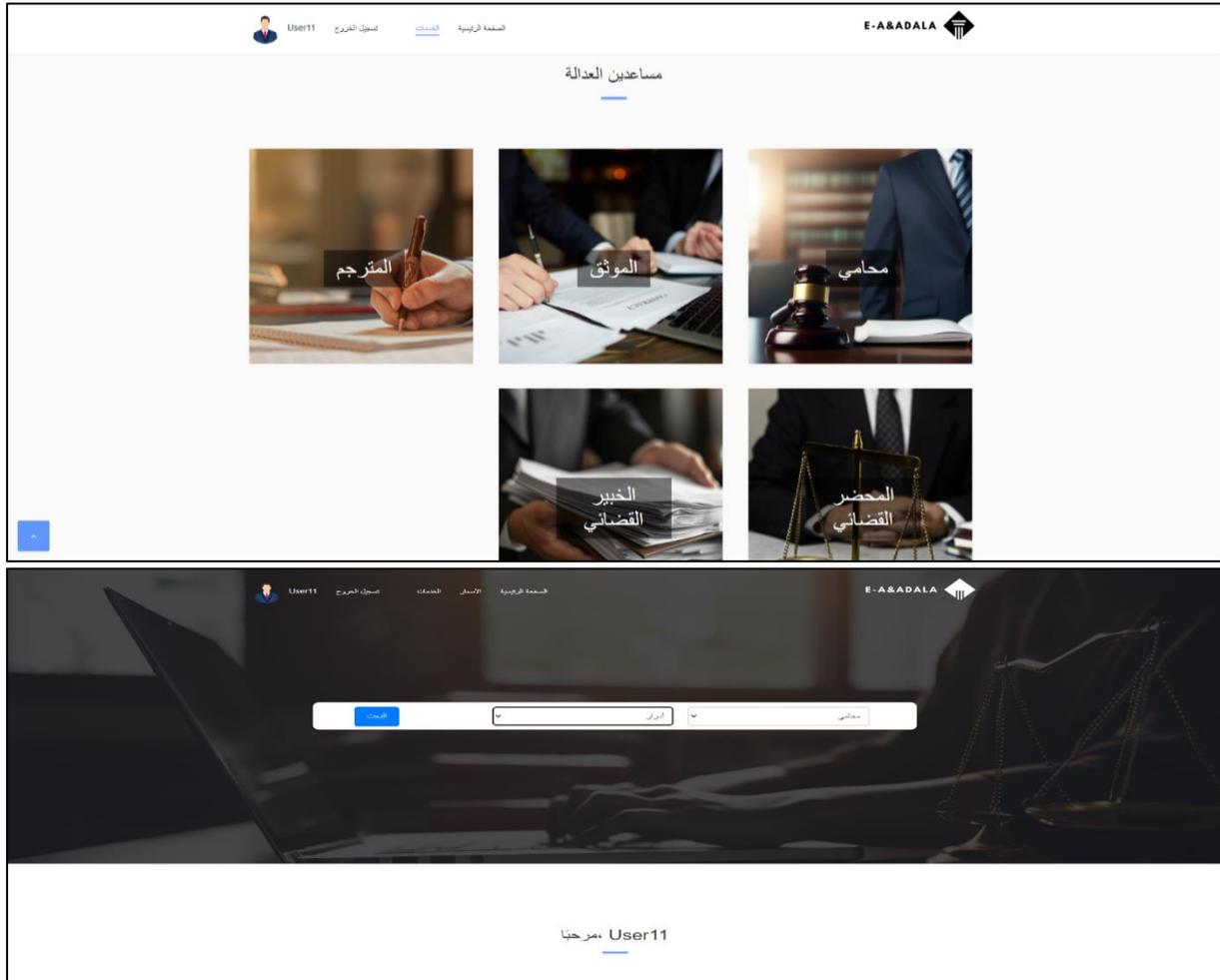


Figure 3.18- page d'accueil client.

o. Résultat de recherche sur un assistant judiciaire

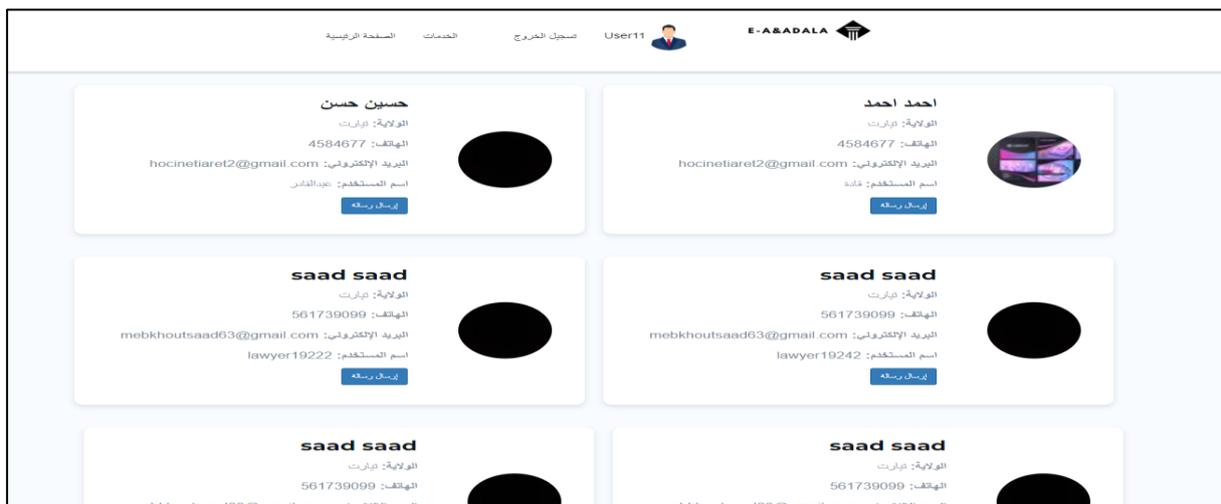


Figure 3.19- Liste des assistants judiciaires conventionnés.

p. Interface de Messagerie

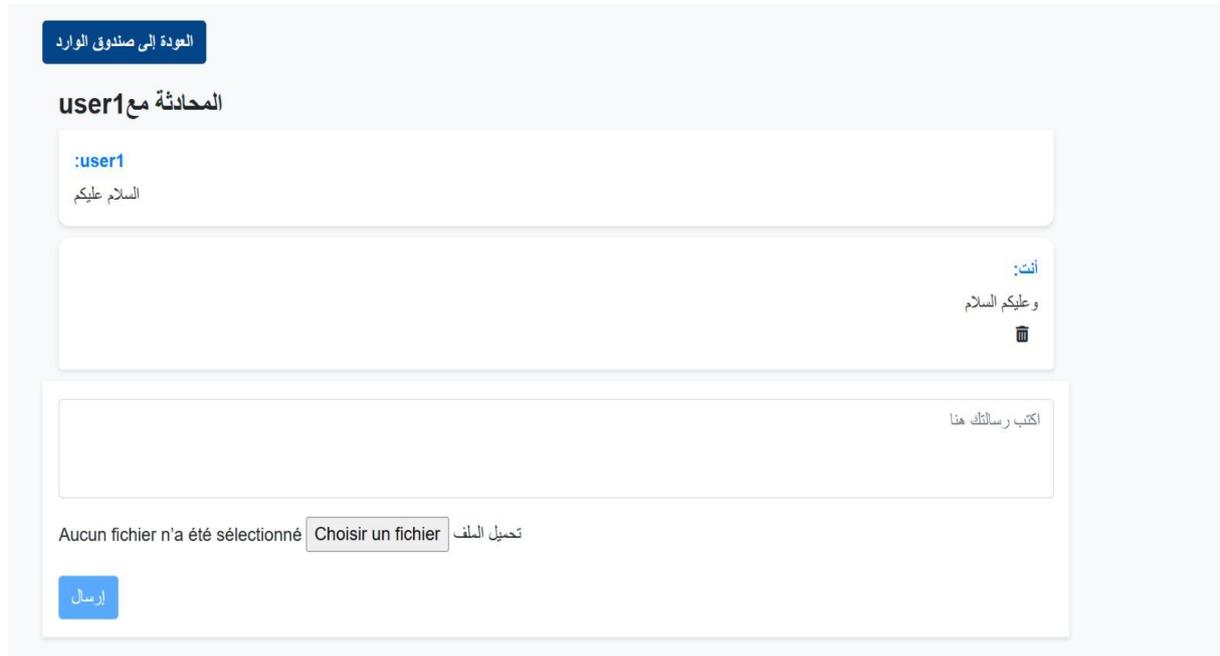


Figure 3.20- Messagerie.

4 Conclusion

Dans ce dernier chapitre, nous avons fourni une description détaillée de l'implémentation et de la réalisation de la plateforme e-A&ADALA. Nous avons présenté l'environnement matériel et logiciel utilisé pendant l'implémentation, ainsi qu'une description de l'architecture du système d'abonnement ciblé par e-A&ADALA. De plus, nous avons exposé les différentes interfaces des deux assistants judiciaires intégrés à la plateforme, à savoir les avocats et les notaires.

Conclusion Générale & Perspectives

L'impact de l'intégration du modèle de service SaaS et de l'architecture multi-tenant sur les applications hébergées par le cloud est facilement sous-estimé, mais peut soulever des défis majeurs en matière de spécification et de mise en œuvre. Cependant, cela peut entraîner des défis majeurs en termes de spécification et de mise en œuvre.

Dans le cadre de ce travail startup, nous avons documenté les efforts entrepris pour créer et mettre en place une application d'abonnement multi-locataire en utilisant le Framework Django, avec une approche de base de données unique.

L'objectif principal de ce travail était de décrire les étapes suivies pour la conception et surtout de la réalisation d'une plateforme d'abonnement holistique destiné à tous les assistants judiciaires.

Une partie significative de ce mémoire s'est concentrée sur :

- Une revue concise de l'état de l'art des paradigmes basés sur la virtualisation, y compris le cloud computing, ainsi que de leurs modèles de déploiement et de service.
- La description, l'explication et la distinction de plusieurs modèles de maturité et de réalisation d'architectures multi-locataires dans le cloud computing.
- La présentation et l'apprentissage du Framework Django, qui offre d'excellentes bibliothèques pour le modèle SaaS et l'architecture multi-locataire.
- La conception et mise en place d'une plateforme d'abonnement SaaS typique nommé "e-A&ADALA" avec le Framework Django.

Le travail présenté dans ce mémoire ouvre de nombreuses perspectives prometteuses à explorer.

À court terme, il est prévu d'intégrer divers assistants judiciaires (huissiers, traducteurs, etc.) et de les incorporer à la plateforme e-A&Adala

À moyen terme, il est également envisageable de développer la même plateforme sur divers frameworks tels que Symfony, Codegniter, et Laravel, entre autres. De plus, il est prévu d'intégrer d'autres bibliothèques supportant la multi-location, comme "Tenancy for Laravel."

Pour l'expansion de cette plateforme, il est prévu de considérer non seulement l'approche d'une base de données unique, mais aussi celle de multiples bases de données. Par ailleurs, il est crucial de ne pas se limiter aux bases de données SQL (comme MySQL) mentionnées dans ce mémoire, mais également d'envisager d'autres types de bases de données. Notre objectif est

d'élargir cette perspective en incluant des bases de données comme OracleDB et PostgreSQL, ainsi qu'en explorant d'autres types de bases de données NoSQL, telles que MongoDB, Cassandra, et Google Datastore.

À long terme, l'architecture multi-tenant décrite dans ce mémoire a pour objectif de s'intégrer dans un environnement virtualisé basé sur le cloud computing en tant que paradigme de recherche. Toutefois, dans une perspective future, nous prévoyons de proposer des implémentations similaires dans d'autres environnements virtualisés. Cela implique que nous pourrions explorer l'implémentation de la multi-location dans des domaines tels que le Fog computing, l'Edge computing, et d'autres encore..

Le travail présenté dans ce mémoire constitue une première étape vers la compréhension et l'implémentation d'une solution holistique pour tous les assistants judiciaires, dans divers environnements virtualisés, tels que le Cloud, le Fog, l'Edge, et bien d'autres. Toutefois, plusieurs extensions n'ont pas pu être abordées en raison des contraintes de temps liées à la rédaction de ce mémoire. Ces extensions incluent des aspects cruciaux tels que la sécurité, la tolérance aux pannes, l'équilibrage de charge, la gestion des ressources, et bien d'autres encore, liés à la multi-location..

Références bibliographiques

- [01] RF Wireless World. Difference between edge, fog, and cloud computing. <https://www.rfwireless-world.com/images/Difference-between-Edge-Fogand-Cloud-Computing.jpg>, Accessed June 5, 2023.
- [02] « TH.M.INF.2023.38.pdf ». Master 2 Informatique. Université de Tiaret. 2023-2024. Consulté le : 21 mai 2024. [En ligne].
- [03] P. Mell et T. Grance, « The NIST Definition of Cloud Computing ».
- [04] samuel, « Le cloud computing : les avantages et les inconvénients », Petite Entreprise. Consulté le: 21 mai 2024. [En ligne].
- [05]<https://www.techniques-ingenieur.fr/base-documentaire/technologies-de-l-information-th9/gestion-de-contenus-numeriques-42311210/cloud-computing-h6020/les-trois-modeles-de-services-h6020niv10004.html>
- [06] Guerid Hachem. Cours : "Cloud computing". Master 2 RSID Informatique. Université des Sciences et de la Technologie d'Oran. 2019-2020
- [07] « What Is Software as a Service (SaaS)? | IBM ». Disponible sur <https://www.ibm.com/topics/saas>
- [08] <https://www.sylob.com/fr/blog/saas-definition>
- [09] <https://www.cegid.com/fr/faq/quest-ce-que-le-saas/>
- [10] <https://www.wizishop.fr/blog/avantage-saas>
- [11] Ali Ghaddar. Une contribution à la gestion des applications SaaS mutualisées dans le cloud: approche par externalisation. Informatique [cs]. Université de Nantes (UN), FRA., 2013. Français.
- [12] <https://hackernoon.com/fr/Comment-lib%C3%A9rer-des-informations-pour-tous-%3F-Architecture-multi-locataires-pour-l'analyse-int%C3%A9gr%C3%A9e>.
- [13] <https://geekflare.com/fr/single-tenant-vs-multi-tenant-cloud-computing/>
- [14] Jaap Kabbedijk, Cor-Paul Bezemer, Slinger Jansen, and Andy Zaidman. Defining multi-tenancy : A systematic mapping study on the academic and the industrial perspective. Journal of Systems and Software, 100 :139–148, 2015.
- [15] Abderrazak Henni, Modernisation de la justice, Conférence Nationale sur La Réforme De La Justice, palais Des nations club des pins Alger, 28-29 Mars 2005, p97.24 مجلة الفكر القانوني (عصرنة قطاع العدالة ")المجلد السادس العدد الثاني (2022) (ص ص : 268، 282)ISSN: (والسياسي 1620-2588 "في الجزائر
- [16] <http://dspace.univ-tlemcen.dz/bitstream/112/6325/3/chapitre2.pdf>.

- [17] Sana Cherif. Approche basée sur les modèles pour la conception des systèmes dynamiquement reconfigurables : de MARTE vers RecoMARTE. Modélisation et simulation. Université des Sciences et Technologie de Lille - Lille I, 2013. Français
- [18] <https://www.mjustice.dz/fr/avocat-2/>
- [19] <https://fr.slideshare.net/slideshow/chp2-uc/31735817>.
- [20] <http://remy-manu.no-ip.biz/UML/Cours/coursUML5.pdf>.
- [21] www.uml-sysml.org.
- [22] https://docwiki.embarcadero.com/RADStudio/Athens/fr/D%C3%A9finition_des_diagrammes_de_classes_UML_1.5.
- [23] <https://www.wizishop.fr/lexiqueecommerce/framework#:~:text=Le%20framework%20peut%20se%20traduire,logiciel%20d'entreprise%2C%20etc>.
- [24] <https://www.djangoproject.com/>
- [25] <https://getbootstrap.com/>
- [26] <https://www.futura-sciences.com/tech/definitions/informatique-uml-3979/>
- [27] <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-cas-utilisation>
- [28] <https://www.ibm.com/docs/fr/rsm/7.5.0?topic=uml-sequence-diagrams>.
- [29] <https://www.python.org/>
- [30] <https://visualstudio.microsoft.com/fr/>
- [31] François Exertier. Extension orientée objet d'un SGBD relationnel. Autre [cs.OH]. Université JosephFourier - Grenoble I, 1991. Français.
- [32] <https://towardsdatascience.com/working-structure-of-django-mtv-architecture-a741c8c64082>
- [33] <https://towardsdatascience.com/working-structure-of-django-mtv-architecture-a741c8c64082>
- [34] « SaaS : définition, signification, avantages, inconvénients ». Consulté le: 8 juin 2024. [En ligne]. Disponible sur: <https://www.guru99.com/fr/software-as-a-service.html>
- [35] Consulté le: 8 juin 2024 : <https://staruml.io/>

BMC

business model canvas



La fiche technique du projet البطاقة التقنية للمشروع

Mezidi Hocine Sadek Mebkhout Saad	الاسم و اللقب prénom et nom
E-A&ADALA	الاسم التجاري للمشروع Intitulé de votre projet
Société à Responsabilité Limitée (SARL).	الصفة القانونية للمشروع statut juridique
0656426417 0561342802	رقم الهاتف numéro de téléphone
hocinetiaret2@gmail.com mebkhoutsaad63@gmail.com	البريد الالكتروني Votre adresse e-mail Your email address
Tiaret-Tiaret	مقر مزاولة النشاط (الولاية - البلدية) Votre ville ou commune d'activité Your city or municipality of activity
Un projet commercial .	طبيعة المشروع Nature du projet
Une solution Cloud-SaaS pour les assistants judiciaires.	القيمة المقترحة أو العرض المقدم valeur proposé



تحديد المشكل الذي يواجهه الزبون (Identification du problème rencontré par le client)

<p>Mises à Jour Individuelles des Sites/Logiciels :</p> <p>Chaque assistant judiciaire utilise son propre site ou logiciel, ce qui nécessite des mises à jour individuelles, souvent chronophages et coûteuses. La gestion des mises à jour de sécurité, des correctifs de bugs et des nouvelles fonctionnalités devient complexe et incohérente</p> <p>Personnalisation des Outils et Tarification :</p> <p>Les solutions actuelles manquent de flexibilité pour s'adapter aux spécificités de chaque utilisateur, limitant ainsi leur efficacité et leur adoption. De plus, il n'existe pas de tarification modulable permettant aux utilisateurs de choisir et de payer uniquement pour les fonctionnalités dont ils ont réellement besoin.</p>	<p>ما هي المشكلة التي تريد حلها؟</p> <p>Quel est le problème que vous souhaitez résoudre ?</p>
<p>mjustice.dz</p>	<p>ما هي البيانات المتوفرة لديك التي تدل على وجود المشكلة المحددة؟</p> <p>Quelles sont les données dont vous disposez qui indiquent l'existence du problème identifié ?</p>
<p>À notre connaissance, il n'existe pas de solution similaire.</p>	<p>ما هي المشاريع الأخرى التي استهدفت نفس المشكلة والتي جرى تنفيذها؟</p> <p>Quels sont les autres projets qui ont ciblé le même problème et qui ont été réalisés ?</p>



<p>- Création d'une solution informatique innovante en tant que service (SaaS) pour gérer de manière personnalisée les besoins de tous les assistants judiciaires, avec une option de souscription distinctive.</p>	<p>ماهي أهداف مشروعك و/أو نتائجه المتوقعة؟ Quels sont les objectifs de votre projet et/ou les résultats attendus ?</p>
---	--

القيمة المقترحة وفق المعايير التالية (La proposition de valeur selon les critères suivants)

<p>-Disposer d'une solution SaaS Polygnote, adaptable avec des options de facturation flexibles.</p>	<p>القيمة المبتكرة أو الجديدة La valeur innovante ou nouvelle</p>
<p>-La tarification de la solution proposée est basée sur les personnalisations recommandées par les assistants.</p>	<p>القيمة بالتخصيص La valeur personnalisée</p>
<p>- Nous adoptons la caractéristique principale du Cloud SaaS : le paiement à l'usage. Nous proposons trois plans de souscription différents.</p>	<p>القيمة بالسعر La valeur par le prix</p>
<p>Nul besoin d'une expertis approfondie de la part des assistants pour accéder à la solution. Un ordinateur, une connexion Internet et un lien URL/compte unique fourni par le fondateur de la solution suffisent.</p>	<p>القيمة بالتصميم La valeur dans la conception</p>



<p>Cette solution unique est spécifiquement conçue pour chaque type d'assistant et ses personnalisations spécifiques sont intégrées .</p>	
<p>- La performance de la solution sera principalement influencée par le débit de la connexion Internet, tandis que les autres ressources seront garanties par le fournisseur de services, y compris les aspects de sauvegarde et de sécurité.</p>	<p>القيمة بالأداء العالي</p> <p>La valeur par la haute performance</p>
<p>-Le fournisseur garantit la maintenance continue de la solution pour tous les assistants, y compris les mises à jour et la sécurité, après la location.</p>	<p>القيمة بالخدمة الشاملة</p> <p>La valeur par le service complet</p>
<p>-Plateforme intégrée :</p> <p>En intégrant des technologies récentes et innovantes pour créer une solution tout-en-un.</p> <p>–(django)Utilisation des technologies avancées pour offrir des fonctionnalités innovantes et automatisées, améliorant l'efficacité et la productivité.</p>	<p>القيمة المبتكرة أو الجديدة</p> <p>La valeur innovante ou nouvelle</p>



Customer Segments شرائح العملاء أو الزبائن

العملاء شرائح الزبائن أو	Comportement	Psychographique	Démographique (B2B)	Démographique (B2C)	Géographique
Usage (استخدام)	Utilisation régulière du logiciel.	Classes sociales moyennes à élevées.	Secteur juridique.	25-55 ans.	Algerie.
Loyauté (الوفاء)	Forte fidélité des clients à long terme.	Niveau de vie confortable.			Algérie,.
Intérêt (اهتمام)	Intérêt pour l'efficacité et l'organisation.	Valeurs de justice et d'équité.		Revenus annuels moyens à élevés.	
Passion (و الهواية و شغف)	Passion pour la technologie et le droit.	Personnalités méthodiques et organisées.		Toutes les catégories.	
Sensibilité (حساسيات)	Sensibles à la sécurité des données.	Convictions en matière de confidentialité.	Propriétaires ou actionnaires.	Tous niveaux d'études.	Districts judiciaires.
Habitude de consommation (عادة الاستهلاك)	Habitude de consommer des services SaaS.	Présence numérique .	.	Professions juridiques.	
Mode de paiement (الدفع طرق)	Préférence pour les abonnements mensuels ou annuels.	Centres d'intérêt pour les nouvelles technologies.	Modèles basés sur des abonnements.	Les cartes de paiement.	
Connaissance (المعرفة)	connaissance des outils numériques.		Secteurs juridiques et connexes.		
Nature de la demande (الطلب طبيعة)	Besoin de solutions personnalisées.		Technologies de pointe utilisées.		

Channels قنوات التوزيع

<p>-Site web et portail en ligne : Le principal canal de location où les clients peuvent s'abonner directement à la plateforme via une interface conviviale. Cela inclut des informations détaillées sur les</p>	<p>المبيعات المباشرة</p> <p>Les ventes directes</p>
---	---



<p>fonctionnalités, les plans tarifaires, et les témoignages de clients.</p> <p>–Équipe de location interne : Une équipe dédiée pour les locations directes, capable de contacter les prospects par téléphone ou par email, offrant des démonstrations personnalisées et répondant aux questions spécifiques.</p>	
<p>-soit localement via le contact de Monsieur SG de la Cour de Tiaret, soit à des niveaux régional ou national.</p>	<p>تجار الجملة Les grossistes</p>
<p>- on peut envisager d'avoir des représentants dans chaque tribunal de la Cour de Tiaret.</p>	<p>الموزعون Les distributeurs</p>
<p>-On peut envisager de fournir des jetons aux représentants intra-tribunaux, lesquels pourraient être échangés contre des paiements monétaires.</p>	<p>توزيع التجزئة La distribution au détail</p>

العلاقة مع العملاء Customer Relationship

<p>–Support technique dédié :</p> <p>Offrir un service de support technique disponible par téléphone, email et chat en direct pour aider les clients à résoudre leurs problèmes techniques et</p>	<p>كيف تدير علاقاتك مع العملاء؟ Comment gérez-vous vos relations avec les clients ?</p>
--	---



à utiliser correctement la plateforme.	
<p>Microsoft Dynamics</p> <p>HubSpot CRM</p>	<p>ماهىة أهم البرامج التي ستعتمد عليها في ادارة العلاقة مع الزبون</p> <p>Quels sont les programmes les plus importants sur lesquels vous comptez pour gérer la relation client ?</p>

الشركاء الأساسيون Les partenaires clés

طبيعة الشراكة nature de la partenariat	معلومات حول الشركاء informations sur les partenaires	الشركاء Les partenaires
-Partenariats et accréditation pour promouvoir la plateforme auprès des assistants juridiques.	-Ministère de la Justice	الشريك الأول Le partenaire principal
-Collaborations pour promouvoir la plateforme auprès des assistants juridiques.	-Associations professionnelles	الشريك الثاني Le deuxième partenaire
-Experts pour garantir que la plateforme répond aux besoins spécifiques de l'industrie juridique.	-Consultants juridiques	الشريك الثالث Le troisième partenaire



La structure des coûts هيكل التكاليف

200 000,00 DZD	تكاليف التعريف بالمنتج أو المؤسسة Frais d'établissement
45 000,00 DZD	تكاليف الحصول على العدادات (الماء - الكهرباء) (..... Frais d'ouverture de compteurs (eaux-gaz-....)
1500 000,00 DZD	Droits d'entrée تكاليف الحصول على تكنولوجيا او ترخيص استعمالها
60 000,00 DZD	Droit au bail الحق في الإيجار
100000,00 DZD	Frais de notaire ou d'avocat تكاليف الموثق-المحامي-.....
20000,00 DZD	Enseigne et éléments de communication تكاليف التعريف بالعلامة و تكاليف قنوات الاتصال
51500000,00 DZD	Matériel (cluster avec ses coûts associés)+autres
400000,00 DZD	Matériel de bureau تجهيزات المكتب



2000000,00 DZD	<p>Trésorerie de départ</p> <p>التدفق النقدي (الصندوق) الذي تحتاجه في بداية المشروع.</p>
----------------	--

55825000,00 DZD = المجموع

نفقاتك أو التكاليف الثابتة الخاصة بمشروعك

400000,00 DZD	<p>Assurances</p> <p>التأمينات</p>
20000,00 DZD	<p>Téléphone, internet</p> <p>الهاتف و الانترنت</p>
100000,00 DZD	<p>Autres abonnements</p> <p>اشتراكات أخرى</p>
45000,00 DZD	<p>Eau, électricité, gaz</p> <p>فواتير الماء - الكهرباء - الغاز</p>
35000,00 DZD	<p>Fournitures diverses</p> <p>لوازم متنوعة</p>
10000,00 DZD	<p>Entretien matériel et vêtements</p> <p>صيانة المعدات والملابس</p>
30000,00 DZD	<p>Nettoyage des locaux</p> <p>تنظيف المباني</p>
350000,00 DZD	<p>Budget publicité et communication</p> <p>ميزانية الإعلان والاتصالات</p>



المجموع = 990000,00 DZD

Revenue Stream مصادر الإيرادات

30000000,00 DZD	1 (Investisseur) ممول
20000000,00 DZD	2 (Investisseur) ممول
10000000,00	3 (Investisseur) ممول

المجموع = 60000000,00 DZD



رقم الأعمال

Votre chiffre d'affaires de la première année بيع المنتج في السنة الأولى

متوسط أيام العمل في الشهر	بيع المنتج في السنة الأولى
20	200 000 DZD
20	250 000 DZD
20	300 000 DZD
20	350 000 DZD
20	400 000 DZD
20	450 000 DZD
20	500 000 DZD
20	550 000 DZD
20	600 000 DZD
20	650 000 DZD
20	700 000 DZD
20	750 000 DZD

المجموع = 5700000DZD



بيع المنتج في السنة الثانية Votre chiffre d'affaires de la deuxième année

متوسط أيام العمل في الشهر	بيع المنتج في السنة الثانية
20	800000 DZD
20	850 000 DZD
20	900 000 DZD
20	1000 000 DZD
20	110 0000 DZD
20	1200000 DZD
20	130 0000 DZD
20	140 0000 DZD
20	160 0000 DZD
20	180 0000 DZD
20	200 0000 DZD
20	220 0000 DZD

المجموع = 1615 0000 DZD



بيع المنتج في السنة الثالثة Votre chiffre d'affaires de la troisième année

متوسط أيام العمل في الشهر	بيع المنتج في السنة الثالثة
20	220 0000 DZD
20	225 0000 DZD
20	230 0000 DZD
20	235 0000 DZD
20	240 0000 DZD
20	245 0000 DZD
20	250 0000 DZD
20	255 0000 DZD
20	260 0000 DZD
20	265 0000 DZD
20	270 0000 DZD
20	275 0000 DZD

المجموع = 2970 0000 DZD

تطور حجم رقم الأعمال في السنة

- النسبة المئوية للزيادة في حجم الأعمال بين السنة 1 والسنة 2: 183.33 %
- النسبة المئوية للزيادة في حجم الأعمال بين السنة 2 والسنة 3: 83.84.....%



حاجتك لرأس المال العامل

30 يوم	متوسط مدة الاعتمادات الممنوحة للعملاء بالأيام Durée moyenne des crédits accordés aux clients en jours
30 يوم	متوسط مدة ديون الموردين بالأيام Durée moyenne des dettes fournisseurs en jours

رواتب الموظفين و مسؤولين الشركة

2685000,00DA	رواتب الموظفين Salaires employés
500000,00DA	صافي أجور المسؤولين Rémunération nette dirigeant

Business Model Canvas

Designed for:

Designed by:

Date:

Version:

Key Partners

Who are our Key Partners? Who are our key suppliers? Which Key Resources are we acquiring from partners? Which Key Activities do partners perform?

MOTIVATIONS FOR PARTNERS: Optimization and economy, Reduction of risk and uncertainty, Acquisition of particular resources and activities

Key Activities

What Key Activities do our Value Propositions require? Our Distribution Channels? Customer Relationships? Revenue streams?

CATEGORIES: Production, Problem Solving, Platform/Network

Key Resources

What Key Resources do our Value Propositions require? Our Distribution Channels? Customer Relationships? Revenue Streams?

TYPES OF RESOURCES: Physical, Intellectual (brand patents, copyrights), Human, Financial

Value Propositions

What value do we deliver to the customer? Which one of our customer's problems are we helping to solve? What bundle of products and services are we offering each Customer Segment? Which needs are we satisfying?

CHARACTERISTICS: Newness, Performance, Customization, "Get the Job Done", Design, Brand/Status, Cost Reduction, Risk Reduction, Accessibility, Convenience/Usability

Customer Relationships

What type of relationship does each Customer Segment expect us to establish and maintain with them? Which ones are we established? How are they integrated with the rest of our business model? How costly are they?

Channels

Through which Channels do our Customer Segments want to be reached? How are we reaching them now? How are our Channels integrated? Which ones are best? Which ones are most cost-effective? How are we integrating them with our routines?

Customer Segments

For whom are we creating value? Who are our most important customers? Is our customer base a Mass Market, Niche Market, Segmented, Diversified, Multisided Platform?

Cost Structure

What are the most important costs inherent in our business model? Which Key Resources are most expensive? Which Key Activities are most expensive?

BUSINESS MODELS: Cost Driven (leanest cost structure, low price value proposition, mass automation, extensive outsourcing), Value Driven (focused on value creation, premium value proposition).

CHARACTERISTICS: Fixed Costs (salaries, rents, utilities), Variable costs, Economies of scale, Economies of scope

Revenue Streams

For what value are our customers really willing to pay? For what do they currently pay? How are they currently paying? How would they prefer to pay? How much does each Revenue Stream contribute to overall revenues?

TYPES: Transactional sale, Usage fee, Subscription Fees, Lending/Renting/Leasing, Licensing, Brokerage fees, Advertising

SAMPLES: FIXED PRICING: List Price, Product feature dependent, Customer segment dependent, DYNAMIC PRICING: Negotiation (bargaining), Yield Management, Real-time-Market

Business Model Canvas

Designed for:

Designed by:

Date:

Version:

Key Partners

- Ministry of Justice :Partnerships and accreditation to promote the platform to legal assistants.
- Professional associations: Collaborations to promote the platform to legal assistants.
- Legal consultants: Experts to ensure the platform meets the specific needs of the legal industry.

Key Activities

Creation of an innovative software solution as a service (SaaS) to manage the personalized needs of all legal assistants, with a distinctive subscription option.

Key Resources

- Partnerships with cloud service providers for platform hosting and security.
- Collaborations to promote the platform to legal assistants.
- Experts to ensure the platform meets the specific needs of the legal industry.

Value Propositions

A Cloud-SaaS solution for legal assistants.

Customer Relationships

Dedicated customer support: Online and phone assistance to help users resolve technical and functional issues.

Channels

- Telephone contact, Social media channels (Facebook, Messenger, Instagram...)
- Collaborations with groups and associations of legal assistants to promote the platform.

Customer Segments

- Legal assistants: The primary users of the platform, seeking to optimize office management and improve communication with litigants.
- Litigants: Indirectly affected by the platform's use, benefiting from better interaction and management of their cases.

Cost Structure

- Software development and maintenance: Costs related to developer salaries and technical infrastructure.
- Marketing and sales: Expenses for advertising campaigns, SEO, and partnerships.
- Customer support: Costs for support staff and ticket management tools.
- Hosting and security fees: Costs for cloud servers and data security measures.

Revenue Streams

Monthly or yearly or semesterly subscriptions: Different pricing plans based on features and number of users.