



**Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique**

Université de Sousse

Ecole Supérieure des Sciences et de la Technologie de Hammam Sousse

Licence Fondamentale en Sciences de l'Informatique

Rapport de Stage de Fin d'Etudes

Conception et réalisation d'une application web pour
rechercher et contacter des avocats « MonAvocat.tn »

Réalisé par : Asma Methamem

&

Housseem Nasr Ben Hadj Amor

Soutenu le ../../2021, devant le jury composé de :

Président de jury

Encadrant universitaire : **Mme. Lobna Hlaoua**

Encadrant professionnel : **Mr. Taoufik Keskes**

Année Universitaire : 2020-2021

REMERCIEMENT :

Nous tenons compte à remercier dans un premier lieu, toute l'équipe pédagogique de l'ESSTHS et les intervenants professionnels responsables de la formation de licence fondamentale en sciences de l'informatique.

Nous tenons, bien évidemment, à remercier **Mme Lobna Hlaoua**, notre enseignante et notre encadrante universitaire durant ce stage, pour la qualité de son encadrement, pour sa patience et ces précieux conseils constructifs afin d'aboutir à ce rapport.

Nous remercions **Mr Taoufik Keskes**, notre encadrant au sein de la société **RIF** de nous avoir donné sa confiance, pour son suivi et son encouragement tout au long de ce travail. C'était un grand plaisir d'être parmi l'équipe **RIF**, en tant que stagiaires.

Bien sûr sans oublier nos familles, nous les remercions énormément pour tout ce qu'elles nous ont donné pour en arriver à ce stade de notre vie. Nous remercions également nos amis et nos collègues.

Finalement, nous tenons également à remercier nos honorables membres du jury d'avoir accepté de juger notre travail.

TABLE DES MATIERES

CHAPITRE 1 : PRESENTATION DU CADRE GENERAL DU STAGE **2**

1.1. INTRODUCTION :	3
1.2. PRESENTATION DE L'ORGANISME D'ACCUEIL :	3
1.3. PRESENTATION DU SUJET :	3
1.4. ETUDE DE L'EXISTANT :	4
1.4.1. DESCRIPTION DE L'EXISTANT :	4
1.4.2. CRITIQUE DE L'EXISTANT :	4
1.5. SOLUTION PROPOSEE :	5
1.6. METHODOLOGIE DU TRAVAIL :	6
1.7. PLANNING PROVISIONNEL DU PROJET :	7
Conclusion :	7

CHAPITRE 2 : ANALYSE ET SPECIFICATION DES BESOINS. **8**

2.1. INTRODUCTION :	9
2.2. SPECIFICATION DES BESOINS :	9
2.2.1. IDENTIFICATION DES ACTEURS :	9
2.2.2. SPECIFICATION DES BESOINS FONCTIONNELS :	9
2.3. SPECIFICATION DES BESOINS NON FONCTIONNELS :	11
2.4. LES DIAGRAMMES DE CAS D'UTILISATION :	12
2.4.1. SPECIFICATION ET RAFFINEMENT DE DIAGRAMME DE CAS D'UTILISATION :	13
2.4.1.1. Diagramme de cas d'utilisation raffiné de « S'inscrire » :	13
2.4.1.2. Diagramme de cas d'utilisation raffiné de « consulter questions fréquentes » :	14
2.4.1.3. Diagramme de cas d'utilisation raffiné de « gérer annuaire avocats » :	15
2.4.1.4. Diagramme de cas d'utilisation raffiné de « traiter les dossiers clients » :	16
2.4.1.5. Diagramme de cas d'utilisation raffiné de « mettre à jour planning » :	17
2.4.1.6. Diagramme de cas d'utilisation raffiné de « Demander Rendez-vous » :	18
2.4.1.7. Diagramme de cas d'utilisation raffiné de « consulter profil » :	19
CONCLUSION :	20

CHAPITRE 3 : ETUDE CONCEPTUELLE. **21**

3.1. INTRODUCTION :	22
3.2. ARCHITECTURE LOGICIELLE :	22
3.2.1. LES TYPES D'ARCHITECTURE PHYSIQUE :	22
3.2.2. CHOIX DE L'ARCHITECTURE PHYSIQUE :	24
3.3. CONCEPTION DETAILLEE :	26
3.3.1. DIAGRAMMES DE CLASSES GLOBALES :	26
3.3.2. DIAGRAMMES DE SEQUENCE :	28

CONCLUSION :	30
CHAPITRE 4 : REALISATION ET VALIDATION.	31
<hr/>	
4.1. INTRODUCTION :	32
4.2- ENVIRONNEMENTS DE TRAVAIL :	32
4.2.1- TECHNOLOGIES UTILISEES :	32
4.2.2- LANGAGES DE PROGRAMMATION :	34
4.2.3- OUTILS UTILISES :	34
4.3. TEST ET VALIDATION :	37
CONCLUSION :	53

TABLE DES FIGURES

Figure 1: logo RIF	3
Figure 2: cfbavocat.com.	4
Figure 3:avocat.org.tn.	4
Figure 4: processus de méthode agile "Scrum"	6
Figure 5: Diagramme de Jira	7
Figure 6: diagramme de cas d'utilisation général.	12
Figure 7: diagramme de cas d'utilisation « S'inscrire ».	13
Figure 8: diagramme de cas d'utilisation « Consulter questions fréquentes ».	14
Figure 9: diagramme de cas d'utilisation « gérer annuaire avocats ».	15
Figure 10: diagramme de cas d'utilisation « traiter les dossier clients ».	16
Figure 11: diagramme de cas d'utilisation « mettre à jour planning ».	17
Figure 12: diagramme de cas d'utilisation « demander rendez-vous ».	18
Figure 13: diagramme de cas d'utilisation « consulter profil ».	19
Figure 14: architecture client/serveur.	22
Figure 15: architecture en couches.	23
Figure 16: architecture orientée services.	24
Figure 17: model vue contrôleur « MVC ».	25
Figure 18: diagramme de classes global.	27
Figure 19: diagramme de séquence « Inscription ».	28
Figure 20: diagramme de séquence « Authentification ».	29
Figure 21: diagramme de séquence « Réservation rendez-vous ».	30
Figure 22: interface d'accueil	37
Figure 23: interface d'accueil	38
Figure 24 : interface question fréquentes	38
Figure 25: interface recherche	39
Figure 26: liste des avocats	39
Figure 27: profil avocat	40
Figure 28: interface avant inscription	40
Figure 29: formulaire d'inscription client	41
Figure 30: formulaire d'authentification	41
Figure 31: interface de prise de rendez-vous	42
Figure 32: sélection de la date et heure du rendez-vous	42
Figure 33: validation numéro de téléphone	42
Figure 34: interface de paiement	43
Figure 35: profil du client	43
Figure 36: liste de rendez-vous client	44
Figure 37 : envoyer une question à un avocat	44
Figure 38 : interface historique des consultations	45
Figure 39 : détails de consultation	45
Figure 40 : vérification cin avocat « pré-inscription »	46
Figure 41 : formulaire inscription avocat	46
Figure 42 : profil avocat	47
Figure 43 : modification mot de passe	47
Figure 44 : liste rendez-vous avocat	48
Figure 45 : modification rendez-vous	48
Figure 46 : mettre à jour planning avocat	49
Figure 47 : mettre à jour tarif avocat	49
Figure 48 : email de confirmation	50

<i>Figure 49: login administrateur</i>	51
<i>Figure 50: annuaire avocats</i>	52
<i>Figure 51: interface de liste des clients inscrits</i>	52
<i>Figure 52: interface de liste des avocats inscrits</i>	53

TABLE DES TABLEAUX

Tableau 1: critiques fonctionnels	5
Tableau 2: critiques ergonomiques	5
Tableau 3 : description de cas d'utilisation "s'inscrire"	13
Tableau 4: description de cas d'utilisation 'consulter questions fréquentes'	14
Tableau 5: description de cas d'utilisation 'gérer annuaire avocats'	15
Tableau 6: description cas d'utilisation 'traiter dossier clients'	16
Tableau 7 : description de cas d'utilisation 'Mettre à jour planning'	17
Tableau 8 : description de cas d'utilisation 'demander rendez-vous'	18
Tableau 9: description de cas d'utilisation 'consulter profil'	19

INTRODUCTION GENERALE :

La mission première d'un avocat est de vous défendre et de vous représenter en cas de procédure judiciaire, et agit en tant que plaider au cours d'un procès. Le métier d'avocat est un métier noble qui demande nécessairement la transparence et la justice qui est la première règle de la vie. Généralement le choix d'un avocat consiste sur sa localisation donc il vaut mieux qu'il soit à proximité de ses clients pour avoir une bonne interaction, or ici en Tunisie on a le problème de transport surtout d'une ville à une autre cela peut causer des difficultés, ainsi que le problème de gestion de temps quel que soit pour un avocat ou bien un client. Pour un avocat c'est difficile de gérer son temps entre ses rendez-vous et le déplacement vers le tribunal aussi pour un client qui est occupé par son travail et sa vie personnelle.

En vue d'obtention du Diplôme de licence en sciences de l'informatique, nous avons effectué un stage à l'entreprise « **RIF** » afin de réaliser une plateforme pour faciliter la recherche des avocats dans les divers domaines du droit ainsi que faire des consultations sans avoir se déplacer et même pour aider les avocats à gérer leurs rendez-vous. Donc cette application est destinée pour les avocats et leurs clients pour faciliter leurs vies.

Dans le cadre de notre projet de voie de développement d'une application web pour la recherche des avocats en ligne, nous articulons notre projet autour de quatre chapitres : le premier chapitre consiste à la présentation du cadre générale du stage. Le deuxième chapitre aura comme but d'analyser les besoins fonctionnels et non fonctionnels de notre application. La troisième partie portera sur l'étude conceptuelle de l'application à réaliser. Le dernier chapitre est réservé pour décrire l'implémentation et la réalisation de notre projet tout en mettons en place les outils de développement utilisés et les interfaces de notre application réalisée.

Finalement, nous clôturons ce rapport par une conclusion qui synthétise le travail élaboré et annonce des perspectives dans des futurs travaux.

Chapitre 1 : Présentation du cadre général du stage.

1.1. Introduction :

Dans le cadre de notre projet de fin d'étude nous présentons le cadre général à savoir définir le contexte général du projet et la problématique, avec une étude des applications existantes ainsi que leur critique qui nous mène à proposer la solution souhaitée en citant les objectifs du présent travail

1.2. Présentation de l'organisme d'accueil :

"RIF" est une start-up émergente de rassemblement des ingénieurs francophones. Créée en 2019, elle est spécialisée dans le développement et les services informatiques, big data et IA. C'est une jeune équipe dynamique capable de répondre aux besoins de ses clients. Elle nous a offert l'opportunité de développer et de mettre en pratique nos connaissances tirées de nos études pour notre projet de fin d'étude.



Figure 1: logo
RIF

1.3. Présentation du sujet :

De nos jours c'est indispensable d'avoir un avocat dans notre vie même pour des simples raisons. Donc il vaut mieux de trouver un des meilleurs et lui confier. La recherche d'un bon avocat peut prendre beaucoup de temps, aussi on a le problème de se déplacer pour prendre un rendez-vous ou bien faire une consultation et dans des situations comme l'actuelle là où on vit une pandémie et c'est interdit de se déplacer donc on est obligés de rester chez nous et cela va nous perdre du temps. Concernant un avocat c'est difficile de gérer son temps et ces rendez-vous à cause de son déplacement entre son bureau et le tribunal. Pour résoudre ces problèmes nous avons l'opportunité de développer une application web **Monavocat** qui répond aux besoins du public ciblé citant comme services :

- ✓ Elle permet aux clients de chercher facilement un avocat.
- ✓ Offre un domaine d'interaction entre les avocats et leurs clients.
- ✓ Chaque utilisateur inscrit peut gérer son profil.
- ✓ Aide les avocats à gérer leurs rendez-vous et les clients à les réserver.
- ✓ Simplicité et facilité d'utilisation pour tout type d'utilisateur.
- ✓ Elle donne au client l'opportunité de donner son avis.
- ✓ Donne aux jeunes avocats plus de chance à se présenter dans la vie professionnelle.

Critiques fonctionnels :

Cfbavocat.com	avocat.org.tn
<ul style="list-style-type: none">• pas d'espace d'interaction entre le client et l'avocat.• manque de prise de rendez-vous en ligne.• Pas de diversité dans les domaines du droit.	<ul style="list-style-type: none">• Pas d'interactions entre les clients et les avocats.• Pas d'espace pour les clients• les fonctionnalités du site ne sont pas claires.

Tableau 1: critiques fonctionnels

Critiques ergonomiques :

Cfbavocat.com	avocat.org.tn
<ul style="list-style-type: none">• La barre de menu est mal placée.• Site très basique.• Mauvaise structuration de disposition des boutons.	<ul style="list-style-type: none">• Le site trop chargé.• Le site présente un aspect visuelle médiocre.

Tableau 2: critiques ergonomiques

1.5. Solution proposée :

Suite aux constatations déduites par l'étude des projets existants et après avoir eu plusieurs idées et analyser des différents problèmes risqués, nous avons donc notamment pour mission le développement et la mise en place d'applications informatiques innovantes dans l'intérêt de la profession, notamment en vue de promouvoir l'utilisation des nouvelles technologies dans la relation entre avocats et clients tout en garantissant le strict respect des règles déontologiques et du secret professionnel. Cette plateforme insiste à avoir une bonne

interaction entre les avocats et leurs clients en les offrant la facilité de chercher l'avocat qui les satisfait, le contacter, lui demander des rendez-vous et organiser une consultation juridique à distance, en présentiel, par téléphone, ou par écrit, en toute confidentialité et transparence elle offre aussi l'opportunité aux jeunes avocats à se présenter dans la vie professionnelle.

1.6. Méthodologie du travail :

Afin d'optimiser la gestion de projet et d'assurer la coordination des acteurs et des tâches dans un souci d'efficacité et de rentabilité, il est essentiel d'utiliser une méthode de travail. C'est pour cela qu'on a choisi de travailler avec Scrum comme processus de développement qui est une parmi les méthodologies agiles.

Pour mettre en œuvre des méthodes agiles, il est nécessaire de diviser le projet en plusieurs petits projets. Une fois le petit projet terminé, il doit être vérifié par le client. Ce n'est qu'après vérification que l'équipe peut commencer à travailler sur le prochain petit projet. Avant vérification, le client peut bien entendu faire un retour d'expérience à l'équipe en charge du dossier et demander des ajustements. C'est là que les méthodes de gestion de projet agiles prennent en compte l'évolution des besoins des clients.

Scrum est considérée comme une méthode agile qui est basée essentiellement sur des « sprints ». Sprint : est une boîte de temps issu de découpage du projet. La durée du sprint puisse varier d'un projet à un autre (de quelques heures à un mois), son organisation reste la même. Chaque début de sprint commence par une estimation de l'équipe suivie d'une planification opérationnelle. Après avoir eu terminé un sprint le but est donc de délivrer un produit et passer à un prochain sprint.

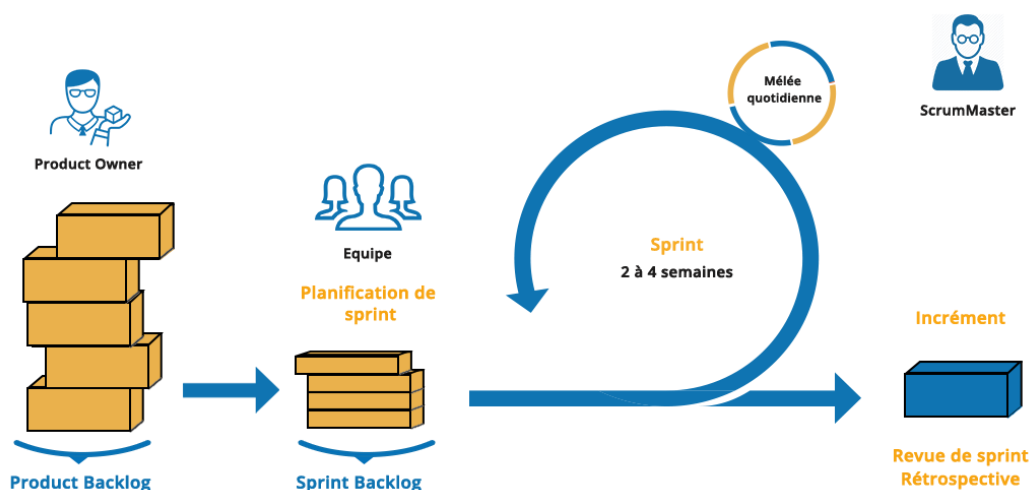


Figure 4: processus de méthode agile "Scrum"

1.7. Planning provisionnel du projet :

La clé principale de la réussite d'un projet est d'avoir un bon plan. En fait, le plan peut aider à bien répartir le travail et séparer les tâches à effectuer. Il peut fournir une meilleure évaluation et une meilleure gestion du temps pour chaque tâche. La figure suivante peut montrer visuellement l'avancement des différentes tâches qui constituent le projet.

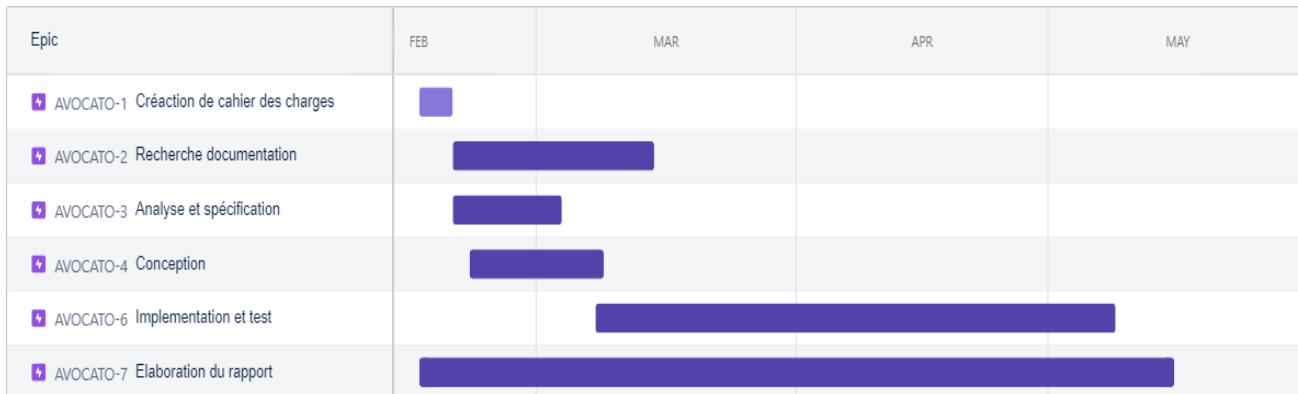


Figure 5: Diagramme de Jira

Conclusion :

Dans ce premier chapitre, nous avons présenté, dans un premier temps, l'entreprise accueillant. Ensuite, nous avons fait une étude de l'existant afin de préciser les objectifs à atteindre grâce aux solutions proposées. La dernière partie de ce chapitre a été consacrée pour l'identification de la méthodologie du travail adoptée pour notre projet et la planification estimée.

Chapitre 2 : Analyse et spécification des besoins.

2.1. Introduction :

Ce chapitre consiste à étudier les différentes expressions des besoins. En premier lieu, nous avons spécifié les besoins tout en identifiant nos acteurs. En second lieu nous détaillons les besoins fonctionnels et non fonctionnelles de notre projet. Et finalement nous décrivons les fonctionnalités de notre système par les diagrammes de cas d'utilisation.

2.2. Spécification des besoins :

Cette phase a pour but de comprendre le contexte du système. Il s'agit d'identifier les fonctions et les acteurs les plus pertinents, de spécifier les risques les plus critiques et de déterminer les cas d'utilisation initiaux.

2.2.1. Identification des acteurs :

Notre application fait intervenir quatre acteurs : L'administrateur, les visiteurs, les avocats et les clients.

- **Le client** : c'est un chercheur d'avocats dans les divers domaines du droit. Il peut gérer son profil, donner son avis et contacter des avocats, il peut aussi prendre des rendez-vous, ou faire une visite en ligne.
- **L'avocat** : chaque avocat inscrit dans la base de données du site peut gérer son profil, gérer ses rendez-vous, communiquer avec ses clients, et faire des visites en ligne.
- **L'administrateur** : C'est un utilisateur responsable pour l'administration de l'application. Il peut gérer les avocats et les clients selon différents critères.
- **Visiteur** : C'est un simple utilisateur de la plateforme, il peut visiter la plateforme et avoir une idée sur le site, il a un accès limité jusqu'à ce qu'il s'inscrit et se connecte.

2.2.2. Spécification des besoins fonctionnels :

Par rapport à l'administrateur, notre application doit répondre aux exigences suivantes :

Gère les utilisateurs :

- L'administrateur peut visualiser et supprimer un compte utilisateur.
- L'administrateur enregistre tous les avocats et les arrange dans un annuaire.

- L'administrateur peut consulter les questions fréquentes en les modifiant ou les supprimant.

Par rapport aux **clients**, notre application doit répondre aux exigences suivantes :

Gestion des rendez-vous :

- Le client peut demander un rendez-vous soit un rendez-vous en ligne ou un rendez-vous présentiel en passant par le paiement.

Gestion de compte :

- Le client a le droit de s'inscrire et créer son propre compte.
- Chaque client doit préciser ses coordonnées. L'accès à cette espace nécessite une authentification.
- Le client peut modifier son profil.

Autres accès :

- Le client peut visualiser les avocats, leurs informations et les contacter afin de finir sa recherche.
- Le client peut agir et donner son avis à propos des avocats.
- Peut avoir une réponse à travers le system chatbot.
- Recevoir des notifications.

Par rapport aux **avocats**, notre application doit répondre aux exigences suivantes :

Gestion de compte :

- Un avocat a le droit de s'inscrire et créer son propre compte par son identifiant unique.
- Chaque avocat doit préciser ses coordonnées. L'accès à cette espace doit être avec vérification.
- Chaque avocat peut modifier son profil.

Gestion des rendez-vous :

- Chaque avocat en consultant son planning, il confirme les rendez-vous ou il les met en attente.

Autres accès :

- Recevoir des notifications.

2.3. Spécification des besoins non fonctionnels :

Les exigences non fonctionnelles sont liées aux contraintes à considérer afin d'établir une solution suffisante pour répondre aux attentes des concepteurs d'architecture dynamique.

Notre application doit nécessairement assurer ces besoins :

- **L'extensibilité** : Dans le cadre de ce travail, l'application doit être extensible, c'est-à-dire qu'il peut y avoir la possibilité d'ajouter ou de modifier de nouvelles fonctionnalités.
- **La sécurité** : L'application doit être hautement sécurisée et les informations ne doivent pas être accessibles à tout le monde, c'est-à-dire que l'application peut être utilisée avec login et mot de passe.
- **L'interface** : L'application doit être conforme aux principes de l'interface homme-machine (IHM), tels que l'ergonomie et la fiabilité.
- **La performance** : L'application doit être performante c'est-à-dire que le système doit réagir dans un délai précis, quel que soit l'action d'utilisateur.
- **La convivialité** : L'application doit être simple et facile à manipuler, même par des non professionnels.
- **L'ergonomie** : Le thème utilisé dans l'application doit inspirer les couleurs et les logos typiques de l'entreprise d'accueil.

2.4. Les diagrammes de cas d'utilisation :

Nous présentons dans la figure ci-dessous le diagramme de cas d'utilisation général qui peut donner une vision globale sur le comportement fonctionnel de notre système et sur l'interaction des différents acteurs.

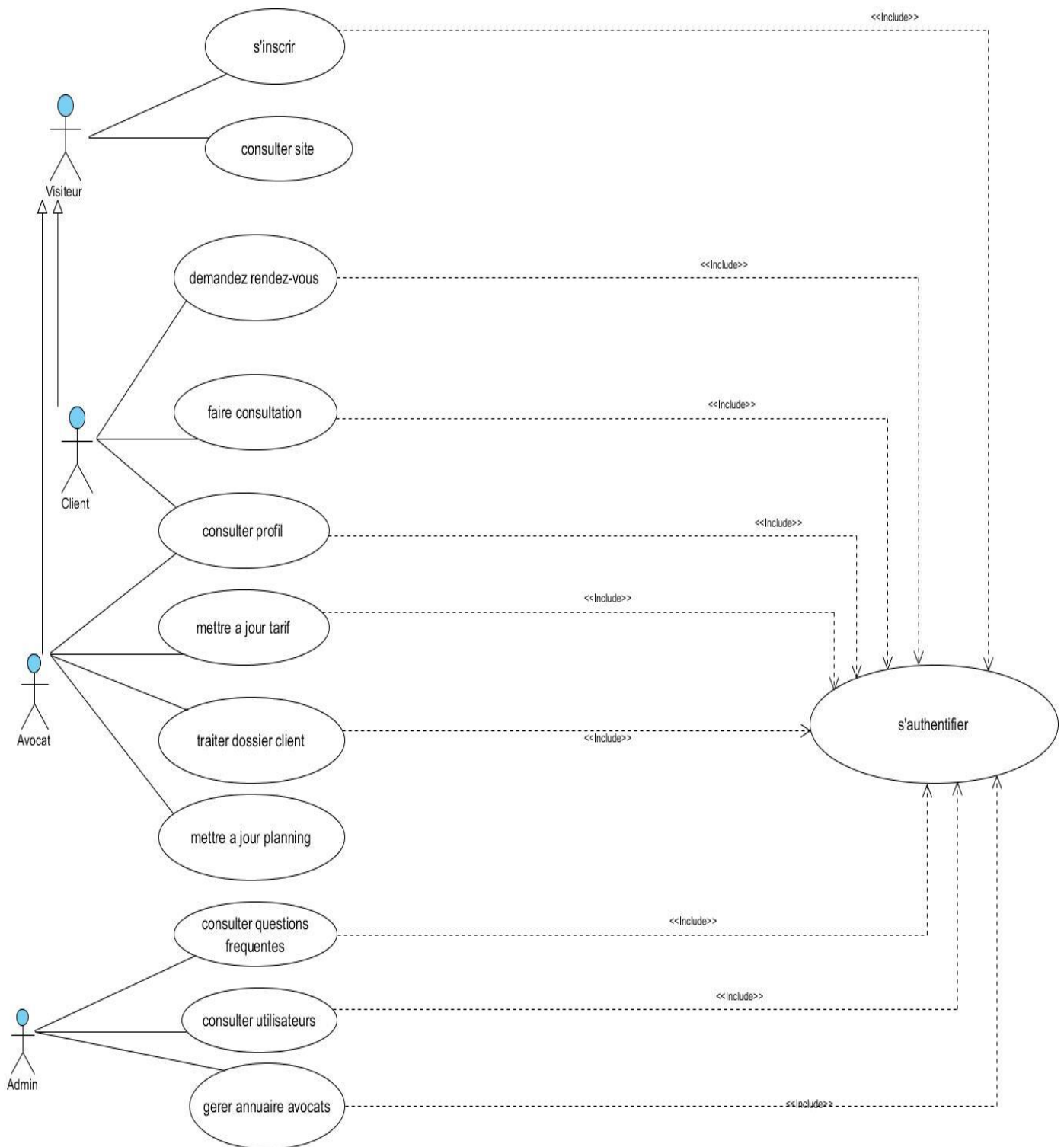


Figure 6: diagramme de cas d'utilisation général.

2.4.1. Spécification et raffinement de diagramme de cas d'utilisation :

2.4.1.1. Diagramme de cas d'utilisation raffiné de « S'inscrire » :

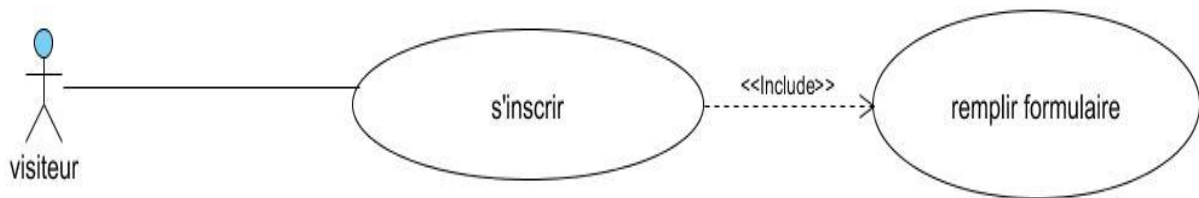


Figure 7: diagramme de cas d'utilisation « S'inscrire ».

Nom du cas :	-S'inscrire.
Acteur Principal	-Avocat, Client.
Résumé	-Ce diagramme permet à un visiteur de faire une inscription dans le site.
Près-condition	-Interface inscription ouverte.
Postcondition	-Le compte est créé et enregistré dans la base de données.
Scénario nominal	-Pour un avocat il doit remplir tout d'abord un champ « cin » pour confirmer qu'il est bien un avocat enregistré dans la base de données du site. Après la confirmation un avocat ou bien un client remplissent des champs nécessaires pour la création d'un compte. En cas de mal remplissage, des messages d'erreurs s'affichent sinon la création est faite et le système le dirige vers la page de Login.

Tableau 3 : description de cas d'utilisation "s'inscrire"

2.4.1.2. Diagramme de cas d'utilisation raffiné de « consulter questions fréquentes » :

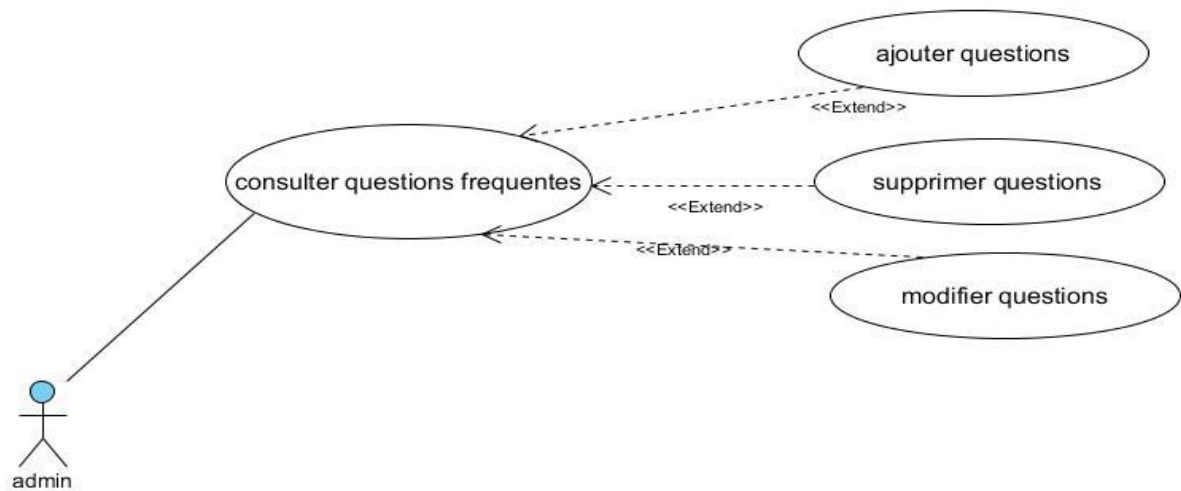


Figure 8: diagramme de cas d'utilisation « Consulter questions fréquentes ».

Nom du cas :	-Consulter questions fréquentes.
Acteur Principal	-Admin.
Résumé	-Ce diagramme permet à l'admin de visualiser les questions fréquentes.
Près-condition	-L'admin doit être authentifié.
Postcondition	-Les questions sont traitées.
Scénario nominal	<p>1 : L'application affiche une liste de questions.</p> <p>2 : L'administrateur choisit l'évènement ce qu'il veut faire (ajouter, modifier ou supprimer).</p> <p>3 : Le système réaffiche de nouveau la liste des questions.</p>

Tableau 4: description de cas s'utilisation 'consulter questions fréquentes"

2.4.1.3. Diagramme de cas d'utilisation raffiné de « gérer annuaire avocats » :

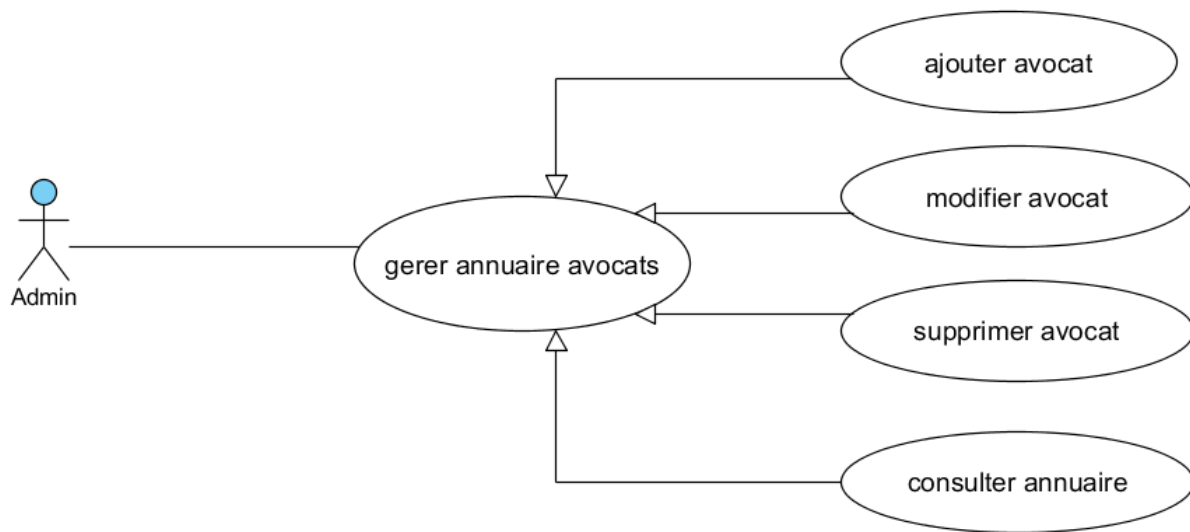


Figure 9: diagramme de cas d'utilisation « gérer annuaire avocats ».

Nom du cas :	-Gérer annuaire avocats.
Acteur Principal	-Admin.
Résumé	-Ce diagramme permet à l'admin de gérer l'annuaire des avocats.
Près-condition	-L'admin doit être authentifié.
Postcondition	-La liste des avocats est traitée.
Scénario nominal	<p>1 : L'application affiche la liste des avocats.</p> <p>2 : L'administrateur choisit l'évènement ce qu'il veut faire (ajouter, modifier ou supprimer).</p> <p>3 : Le système réaffiche de nouveau la liste des avocats.</p>

Tableau 5: description de cas d'utilisation 'gérer annuaire avocats'

2.4.1.4. Diagramme de cas d'utilisation raffiné de « traiter les dossiers clients » :

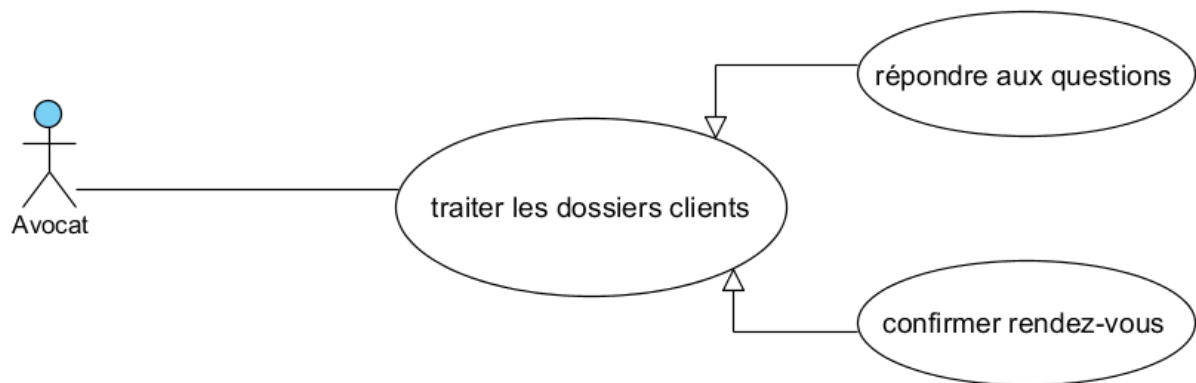


Figure 10: diagramme de cas d'utilisation « traiter les dossier clients ».

Nom du cas :	-Traiter les dossier clients.
Acteur Principal	-Avocat.
Résumé	-Ce diagramme permet à un avocat de traiter les dossiers de ses clients.
Près-condition	-L'Avocat doit être authentifié.
Postcondition	-le client est pris en charge par l'avocat.
Scénario nominal	-Si c'est la date de rendez-vous, l'avocat doit répondre à la question du client si le rendez-vous est de type 'Question écrite'. Sinon le système envoie un lien au client et l'avocat par courriel pour se rejoindre ensemble.

Tableau 6: description cas d'utilisation 'traiter dossier clients'

2.4.1.5. Diagramme de cas d'utilisation raffiné de « mettre à jour planning » :

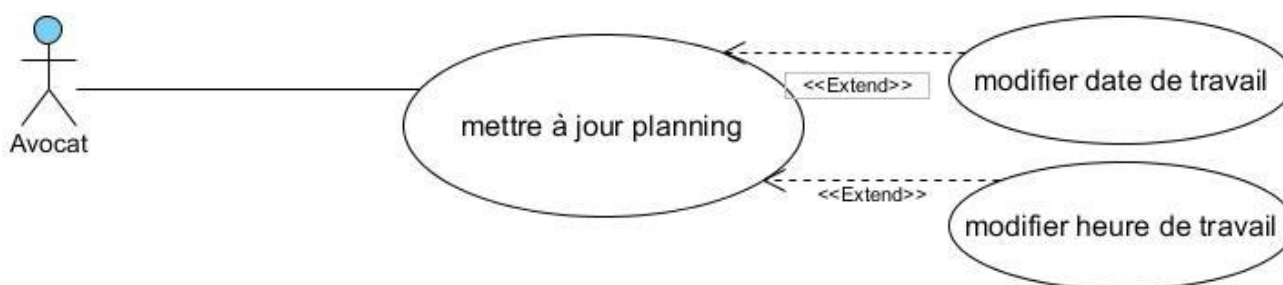


Figure 11: diagramme de cas d'utilisation « mettre à jour planning ».

Nom du cas :	-Mettre à jour planning.
Acteur Principal	-Avocat.
Résumé	-Ce diagramme permet à un avocat de mettre à jour son planning.
Près-condition	-L'Avocat doit être authentifié.
Postcondition	-Le planning de la semaine est mis à jour.
Scénario nominal	<p>-L'avocat doit mettre un planning de travail de la semaine qui va être affiché dans son profil pour vérifier sa disponibilité.</p> <p>Il peut modifier et choisir ses jours de travail en activant ou désactivant un jour précis.</p> <p>Il peut aussi modifier l'heure de début et fin de travail durant la journée tout en ajoutant sa pause entre les consultations et sa pause de midi.</p>

Tableau 7 : description de cas d'utilisation 'Mettre à jour planning'

2.4.1.6. Diagramme de cas d'utilisation raffiné de « Demander Rendez-vous » :

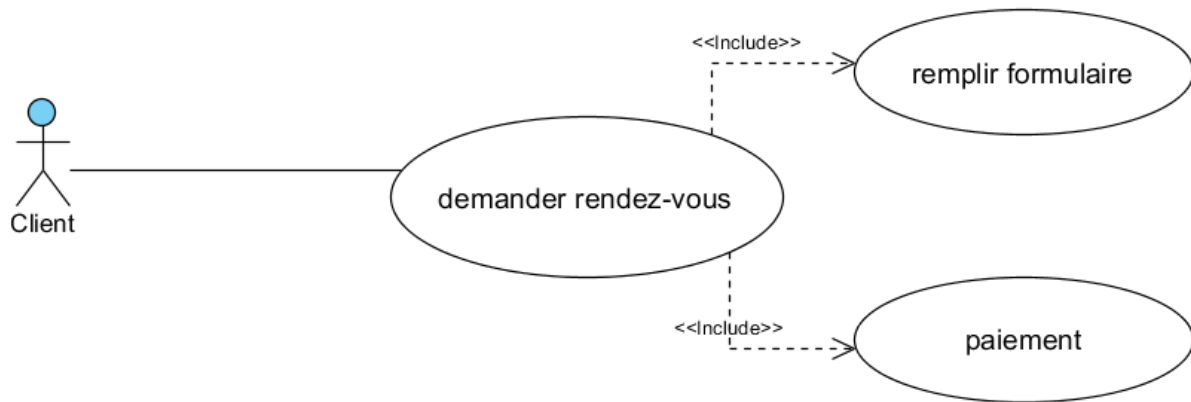


Figure 12: diagramme de cas d'utilisation « demander rendez-vous ».

Nom du cas :	-Demander rendez-vous.
Acteur Principal	-Client.
Résumé	-Ce diagramme permet à un client de demander un rendez-vous.
Près-condition	-Le client doit être authentifié.
Postcondition	-Rendez-vous pris.
Scénario nominal	<p>-Le client choisit le type de rendez-vous qu'il veut faire, puis il fait une recherche selon les trois types : Nom de l'avocat, son gouvernorat et sa compétence.</p> <p>Après avoir choisi le bon avocat, il peut visiter son profil ou directement demander un rendez-vous.</p> <p>Une interface s'ouvre en demandant au client de remplir un formulaire. Une fois terminé le client doit faire le paiement pour que son rendez-vous soit pris.</p>

Tableau 8 : description de cas d'utilisation 'demander rendez-vous'

2.4.1.7. Diagramme de cas d'utilisation raffiné de « consulter profil » :

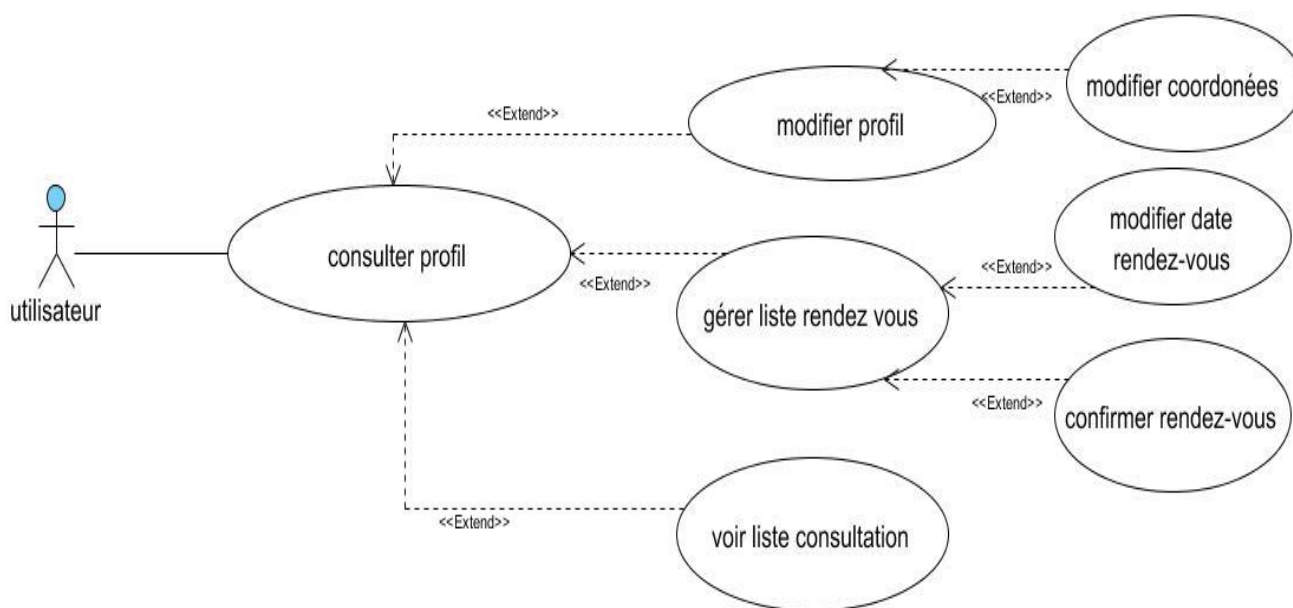


Figure 13: diagramme de cas d'utilisation « consulter profil ».

Nom du cas :	-Consulter profil.
Acteur Principal	-Client, avocat.
Résumé	-Ce diagramme permet à un utilisateur de consulter son profil.
Près-condition	-L'utilisateur doit être authentifié.
Postcondition	-Profil mis à jour.
Scénario nominal	<p>-Une interface de profil s'affiche, l'utilisateur peut modifier ses coordonnées personnelles.</p> <p>Il peut aussi gérer sa liste de rendez-vous en modifiant la date de rendez-vous ou bien accepter et confirmer un rendez-vous pour l'avocat.</p> <p>Il a la possibilité aussi de consulter et voir tout son historique de rendez-vous qu'il a fait.</p>

Tableau 9:description de cas d'utilisation 'consulter profil'

Conclusion :

Dans ce chapitre, nous avons tout d'abord spécifié les différents besoins que doit présenter l'application en essayant d'identifier les acteurs de notre projet puis en répartissant ces besoins en besoins fonctionnels et non fonctionnels. Enfin nous avons bien détaillé les fonctionnalités de notre système par les diagrammes de cas d'utilisation.

Nous terminerons l'étude conceptuelle dans le chapitre suivant pour répondre à ses besoins.

Chapitre 3 : Etude conceptuelle.

3.1. Introduction :

Dans ce chapitre nous essayons en premier lieu de définir qu'est-ce que l'architecture logicielle en citant quelques types d'architecture physique puis en expliquant notre choix d'architecture dans ce projet. Par la suite nous avons présenté la conception détaillée avec un diagramme de classe global et les diagrammes de séquence.

3.2. Architecture logicielle :

Il n'y a pas d'architecture logicielle parfaite pour tous les besoins. Avec l'évolution du temps et des projets, plusieurs architectures typiques sont apparues. Ils constituent un modèle architectural éprouvé qui peut être utilisé comme source d'inspiration pour de nouveaux projets.¹

3.2.1. Les types d'architecture physique :

Parmi les différentes architectures on peut citer quelques-unes les plus connues :

- Architecture client/serveur :

L'architecture client / serveur caractérise le système sur la base d'échanges réseau entre le client et un serveur central qui stocke des données applicatives.

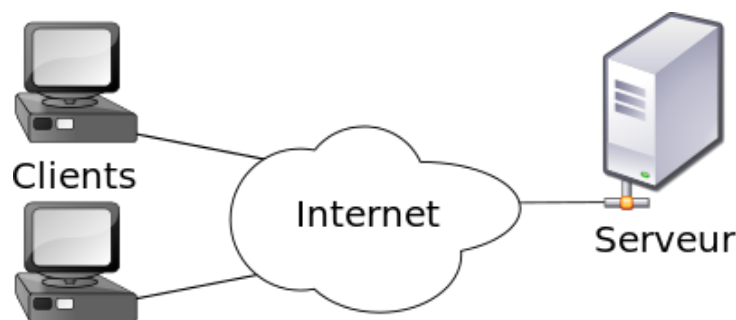


Figure 14: architecture client/serveur.

Le principal avantage de l'architecture client / serveur est la centralisation des données. Stockés au même endroit, ils sont plus faciles à conserver et à protéger. La taille des serveurs

¹ « Memoire Online - Conception et Developpement d'un logiciel de gestion commerciale - Mchangama Ismaila ».

qui les hébergent peut-être ajustée pour accueillir la quantité de données nécessaire et répondre aux besoins de nombreux clients. Ce genre de médaille a son contraire : le serveur constitue le nœud central du système et représente son maillon faible. En cas de panne (surcharge, indisponibilité, problème réseau), le client ne pourra plus fonctionner. Nous pouvons diviser les clients de l'architecture client / serveur en plusieurs types :

Client léger, utilisé uniquement pour l'affichage.

Client robuste, application native, spécialement conçue pour communiquer avec le serveur.

Le client riche combine les avantages d'un client léger et d'un client lourd.

- Architecture en couches :

L'architecture en couches (aussi appelée architecture multi-tiers) est une pratique d'architecture logicielle qui propose de concevoir le système comme une superposition de strates, chaque strate étant définie par une responsabilité spécifique.

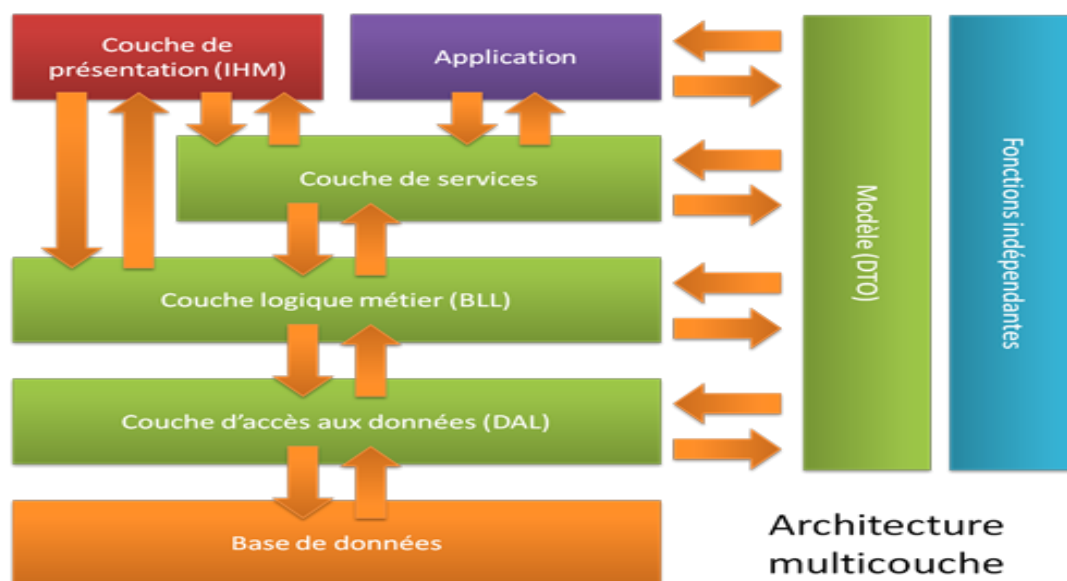


Figure 15: architecture en couches.

L'architecture respecte le principe de séparation des tâches et permet de comprendre les échanges au sein de l'application. Lorsque chaque couche correspond à un processus différent sur la machine, nous l'appelons une architecture à n couches, et n représente le nombre de

couches. Le navigateur Web accède à une page dynamique qui intègre les informations stockées dans la base de données, ce qui constitue un exemple classique d'architecture à trois niveaux.

- Architecture orientée services :

L'architecture orientée services (SOA) utilise un format d'échange commun (généralement XML ou JSON) pour décomposer le logiciel en un ensemble de services métier.

L'architecture de micro-service est la dernière variante, ce qui réduit la granularité du service pour assurer sa flexibilité et son évolutivité, mais au prix d'une plus grande distribution du système. La figure ci-dessous illustre la différence entre ces deux méthodes.

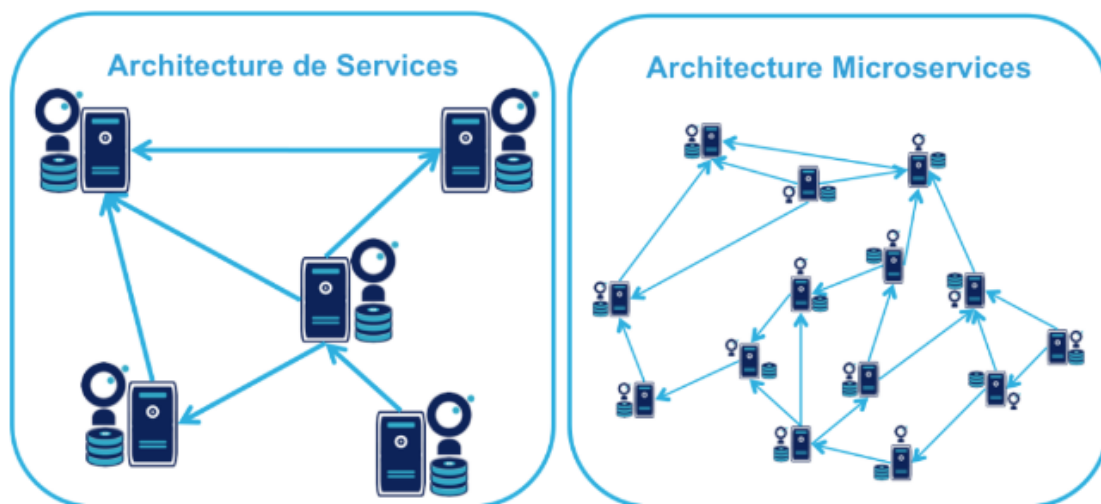


Figure 16: architecture orientée services.

Finalement l'architecture Modèle-Vue-Contrôleur sera détaillée dans la section suivante .

3.2.2. Choix de l'Architecture Physique :

On a cité tout à l'heure les différents types d'architecture physique les plus connus en les caractérisant.

Pour notre projet on a choisi l'architecture Modèle-Vue-Contrôleur qui est une architecture logicielle destinée aux interfaces graphiques lancée en 1978 et très populaire pour les applications web. Le motif est composé de trois types de modules ayant trois responsabilités différentes : les modèles, les vues et les contrôleurs [2].

La partie modèle combine le logique métier et l'accès aux données. Il peut s'agir d'un ensemble de fonctions ou d'une classe.

La partie vue traite de l'interaction avec l'utilisateur : représentation, saisie et vérification des données.

La partie contrôleur permet de gérer la dynamique de l'application, c'est le lien entre les deux autres parties.

Ce qui suit cette figure qui décrit le principe de fonctionnement du modèle de conception MVC.

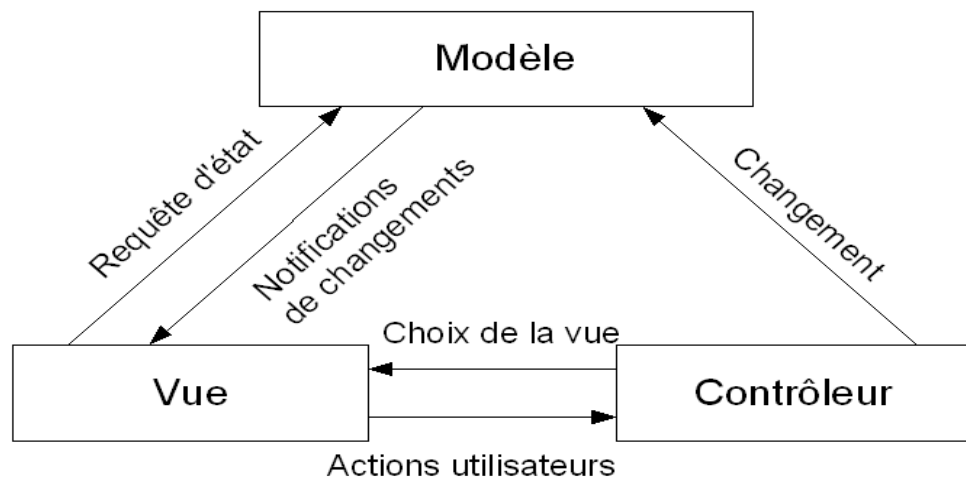


Figure 17: model vue contrôleur « MVC ».

L'approche MVC apporte de réels avantages :

- Conception claire et efficace en séparant la vue et les données du contrôleur.
- Un gain du temps de maintenance et de développement du site.
- Plus de flexibilité dans l'organisation du développement de sites Web entre différents développeurs.

L'architecture MVC est une solution très intéressante et puissante, qui est mieux implémentée dans les grands projets. L'indépendance des trois couches est très importante, car l'architecture obtient ses fonctions et sa logique grâce à cette différence. Cela facilite la mise à jour de la conception du site Web sans affecter les données ou l'organisation du site Web. Au contraire, il est facile de changer la structure ou la gestion du stockage des données sans interrompre l'affichage. ²

3.3. Conception détaillée :

Pour ce faire on a commencé par les diagrammes de cas d'utilisation (Use Case) qui permettent de donner une vue globale de l'application. Pas seulement pour un client non avisé qui aura l'idée de sa future application mais aussi le développeur s'en sert pour le développement des interfaces. En deuxième lieu on va présenter la chronologie des opérations par les diagrammes de séquences.³

3.3.1. Diagrammes de classes globales :

Un diagramme de classes fournit une vue globale d'un système en présentant ses classes dont chacune présente la description de l'ensemble d'objets possédants les mêmes critères.

Chaque classe possède des attributs et elle est reliée aux autres par des associations ou de généralisations

² « Patrons logiciels ».

³ « Patrons logiciels ».

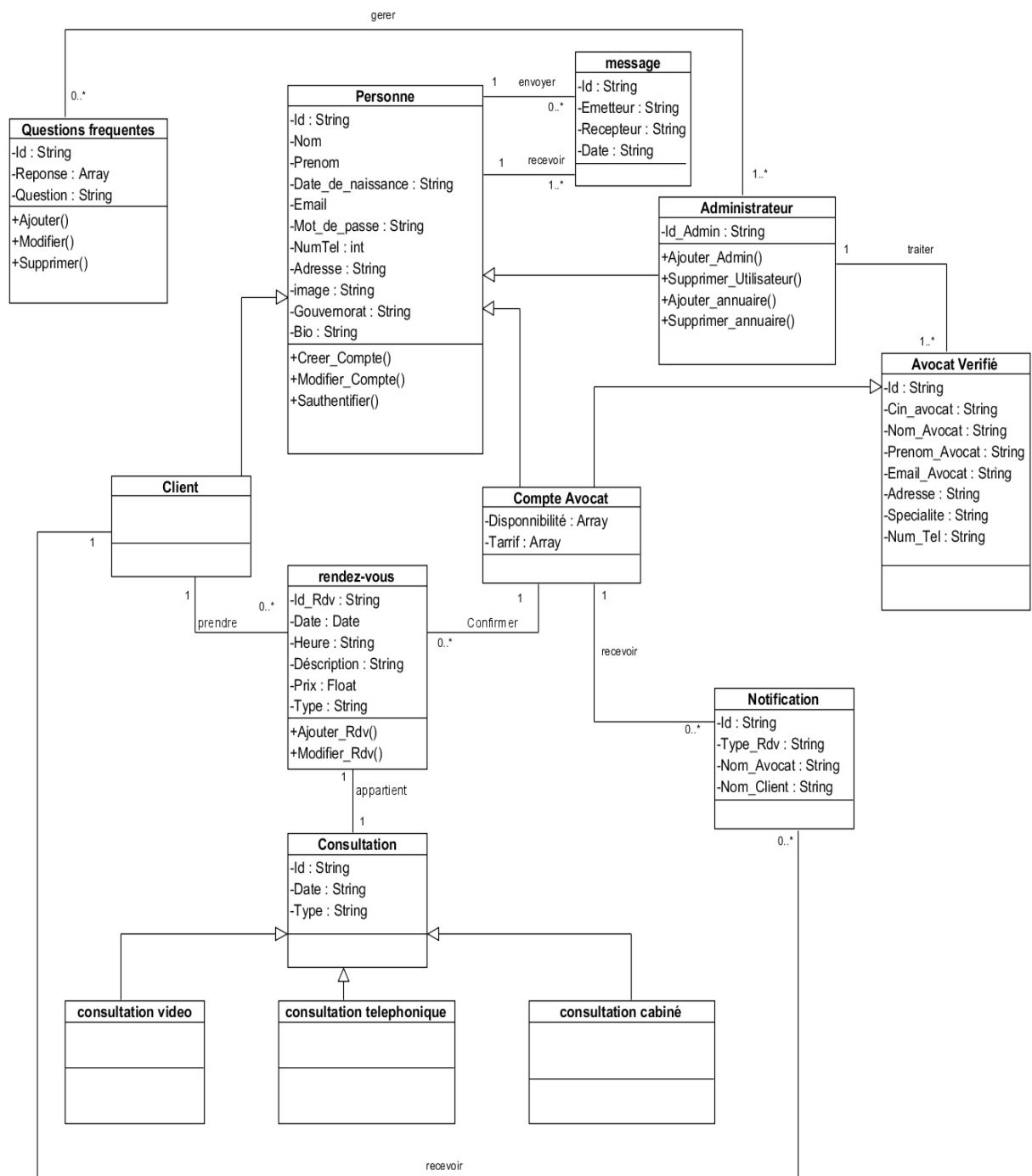


Figure 18: diagramme de classes global.

3.3.2. Diagrammes de séquence :

Le diagramme de séquence est utilisé pour montrer l'interaction des objets dans la scène du diagramme de cas d'utilisation.

Diagramme de séquence « inscription ».

Le diagramme illustré par la Figure 19 présente la démarche d'inscription d'un utilisateur.

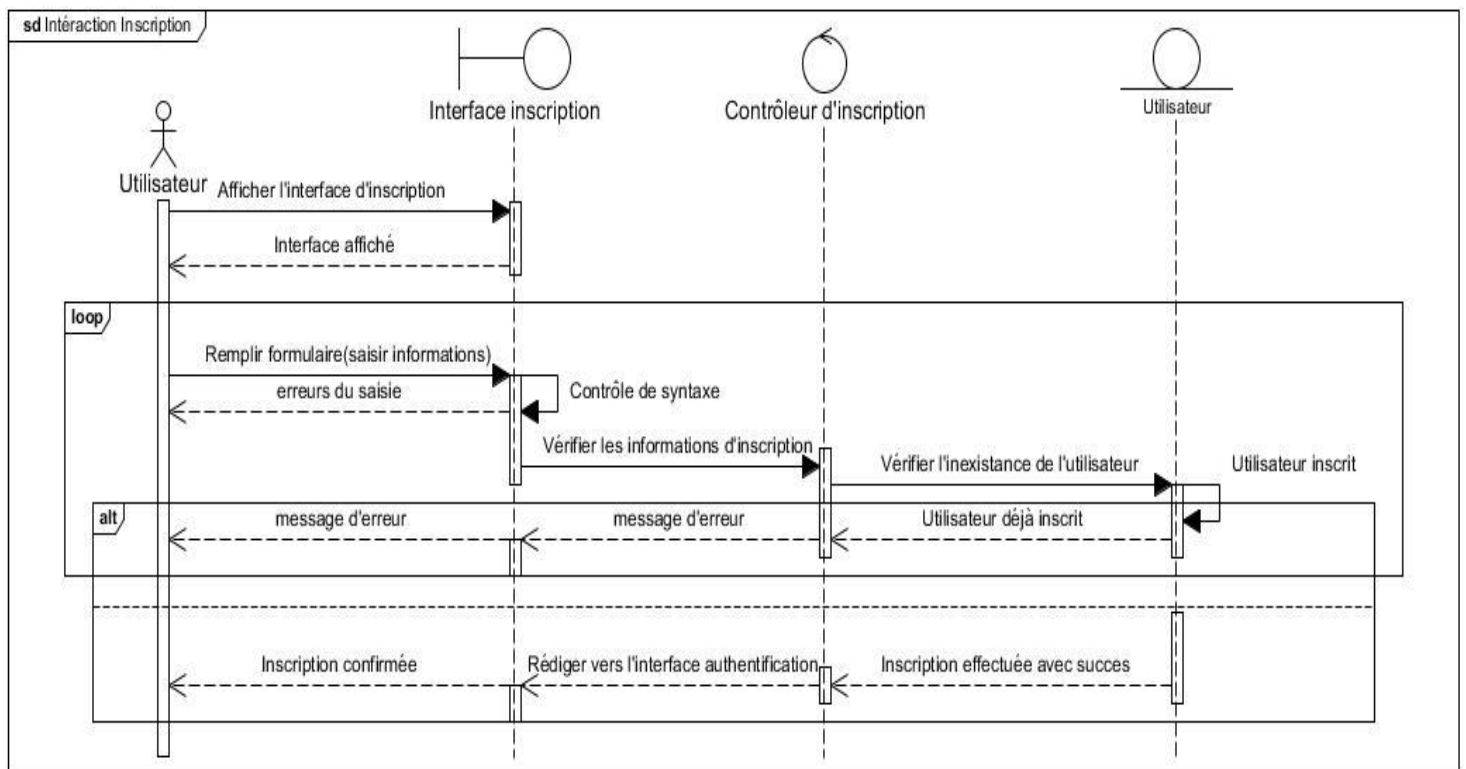


Figure 19: diagramme de séquence « Inscription ».

Diagramme de séquence « authentification ».

Le diagramme illustré par la Figure 20 présente la démarche d'authentification d'un utilisateur.

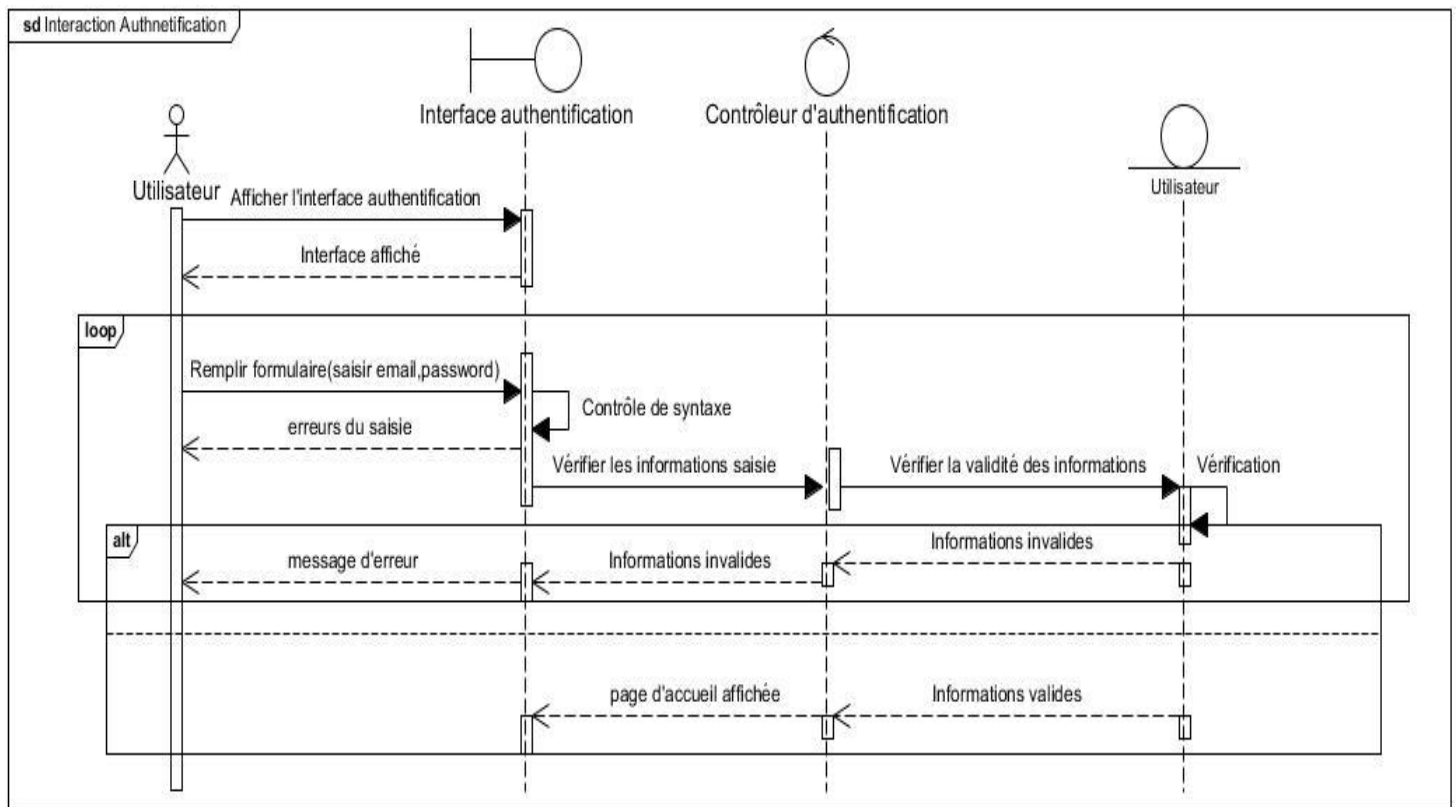


Figure 20: diagramme de séquence « Authentification ».

Diagramme de séquence « prise de rendez-vous ».

Le diagramme illustré par la Figure 21 présente la démarche de prise de rendez-vous par un client.

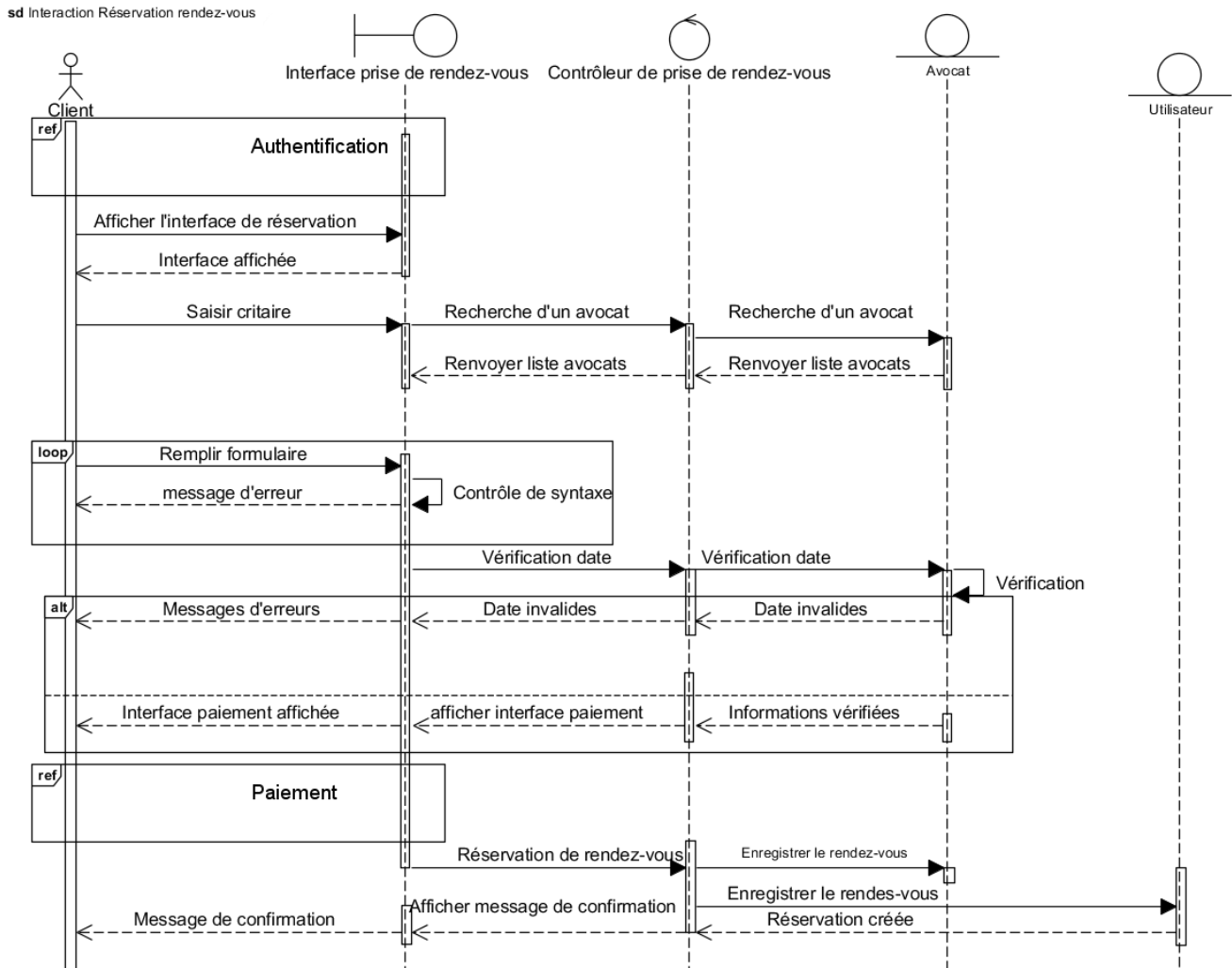


Figure 21: diagramme de séquence « Réservation rendez-vous ».

Conclusion :

Dans ce chapitre, nous avons expliqué notre choix d'architecture utilisée dans ce projet par définir qu'est-ce qu'une architecture logicielle en identifiant quelques types d'architecture et puis nous avons défini la conception détaillée de notre projet par une modélisation des différents composants du système à travers le diagramme de classe global et les diagrammes de séquence.

Chapitre 4 : Réalisation et validation.

4.1. Introduction :

Après avoir modélisé la conception du projet nous allons donc discuter sur la partie de réalisation et validation. Dans cette partie, en premier lieu on va préciser notre environnement de travail par le choix des technologies, les langages et les outils utilisées durant notre projet. Ensuite nous avons la partie de test et validation qui concerne l'étude des interfaces pour avoir une vision réelle sur le projet.

4.2- Environnements de Travail :

Pour bien réussir un projet, il faut bien prendre soin de la gestion de l'environnement de travail qui est un concept global qui comprend la gestion technique de tous les équipements, la gestion des services et des processus permettant le travail, la collaboration, l'innovation et l'apprentissage. En améliorant notre gestion, on vise d'augmenter l'efficacité, d'avoir une meilleure productivité et d'optimiser les coûts et le temps.

4.2.1- Technologies utilisées :



React Js : React est une bibliothèque JavaScript open-source qui est utilisée pour construire des interfaces utilisateur spécifiquement pour des applications d'une seule page. Elle est utilisée pour gérer la couche d'affichage des applications web et mobiles. React nous permet également de créer des composants d'interface utilisateur réutilisables. React a été créé par Jordan Walke, un ingénieur logiciel travaillant pour Facebook. React a été déployé pour la première fois sur le flux d'informations de Facebook en 2011 et sur Instagram.com en 2012[3].

Pourquoi utiliser ReactJs ?

React permet aux développeurs de créer de grandes applications Web pouvant modifier les données sans recharger la page. L'objectif principal de React est d'être rapide, évolutif et simple.

Examinons de plus près certaines caractéristiques importantes de React :

- Simplicité
- Facile à apprendre

- Approche native
- Performance
- Testabilité



Node Js : Node.js est une plateforme de développement Javascript , orientée vers les applications réseau événementielles hautement concurrentes qui doivent pouvoir monter en charge[4].

Pourquoi utiliser NodeJs ?

Node peut être comparé à Python, Ruby, Java, PHP. Cette bibliothèque présente de nombreux intérêts :

- Logiciel libre
- Performance du moteur v8
- Modèle non bloquant
- Communauté très active
- Système de paquet intégré (NPM) qui est la commande du gestionnaire de packages installée avec Node.js. Par conséquent, il permet de rechercher/installer/désinstaller des packages.



Axios : Axios est une bibliothèque JavaScript fonctionnant comme un client HTTP. Elle permet de communiquer avec des API en utilisant des requêtes. Comme avec les autres clients HTTP, il est possible de créer des requêtes avec la méthode POST[5].



Express : Express est une infrastructure d'applications Web Node.js minimaliste et flexible qui fournit un ensemble de fonctionnalités robuste pour les applications Web et mobiles[6].

Express apporte une fine couche de fonctionnalités d'application Web de base sans masquer les fonctionnalités de Node.js

4.2.2- Langages de programmation :



JavaScript : JavaScript est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web. Mais il est aussi utilisé dans de nombreux environnements extérieurs aux navigateurs web tels que Node.js, Apache CouchDB voire Adobe Acrobat. Le code JavaScript est interprété ou compilé à la volée (JIT). C'est un langage à objets utilisant le concept de prototype, disposant d'un typage faible et dynamique qui permet de programmer suivant plusieurs paradigmes de programmation : fonctionnelle, impérative et orientée objet[7].



HTML : HTML signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure. D'autres technologies sont utilisées avec HTML pour décrire la présentation d'une page (CSS) et/ou ses fonctionnalités interactives [8].



CSS : CSS (Cascading Style Sheets en anglais, ou « feuilles de style en cascade ») sont le code utilisé pour mettre en forme une page web, forment un langage informatique qui décrit la présentation des documents HTML et XML[9].

4.2.3- Outils utilisés :



Vs Code : Visual Studio Code est un éditeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'auto-complétions, la coloration syntaxique, le débogage, et les commandes git. Il est livré avec un support intégré pour JavaScript, TypeScript et Node.js et dispose d'un riche écosystème d'extensions pour d'autres langages (tels que C ++, C #, Java, Python, PHP, Go) et des environnements d'exécution (tels que .NET et Unity)[10].



Microsoft Word : Microsoft Word est un logiciel de traitement de texte publié par Microsoft. C'est l'un des logiciels les plus utilisés dans le monde et permet de rédiger des lettres, CV, rapports et tous types de documents texte. Word permet de mettre en page vos documents, de générer des sommaires, de numérotter automatiquement des pages, corriger la grammaire et l'orthographe, de créer des schémas, de faire du publipostage[11].



Jira : Jira est un système de suivi de bugs, un système de gestion des incidents, et un système de gestion de projets développé par Atlassian[12]. Jira nous permet de décomposer le projet en arborescence et d'allouer des ressources pour chaque tâche planifiée.



Mongo DB : Mongo DB est développé depuis 2007, c'est un système de gestion de base de données orienté documents, répartitionnable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++[13].



Robot 3T : Robot 3T est un outil conçu pour simplifier grandement l'utilisation de la base de données MongoDB, il permet d'accéder au serveur cible et de parcourir les documents stockés dans la base de données.

Robot 3T affiche tous les documents consultés dans un nouvel onglet, permettant une navigation rapide vers différentes zones de la base de données. Vous pouvez ouvrir des documents et insérer de nouveaux documents dans la base de données et supprimer ou modifier des documents existants.



Postman : Postman est un logiciel qui se focalise sur les tests des API. Il est devenu très populaire pour tester les Micro-services, notamment grâce à sa simplicité et ses fonctionnalités très spécialisées[14].

Il permet de :

- Tester des requêtes http
- Tester des requêtes http
- Utiliser des environnements
- Uploader des données
- Exporter en Json



Visual Paradigm : Visual Paradigm est un logiciel utilisé de création de diagrammes dans le cadre d'une programmation. Bref, il dispose de multiples options qui permettent un large éventail de possibilités de modélisation en ULM. Il fournit de nombreux outils pour créer différents types de diagrammes, tels que des diagrammes d'exigences et de cas d'utilisation. Il dispose de nombreux navigateurs, permettant de personnaliser chaque élément.

4.3. Test et validation :

Afin de tester cette application nous avons présenter des interfaces pour les 3 types d'utilisateurs.

Session Visiteur :

Pour un visiteur, comme présente la (figures 22) est la première page qui s'afficher, elle s'agit de la page d'accueil qui contient une navbar permettant l'accès aux principales fonctionnalités du site ainsi qu'une barre de recherche **Monavocat**.

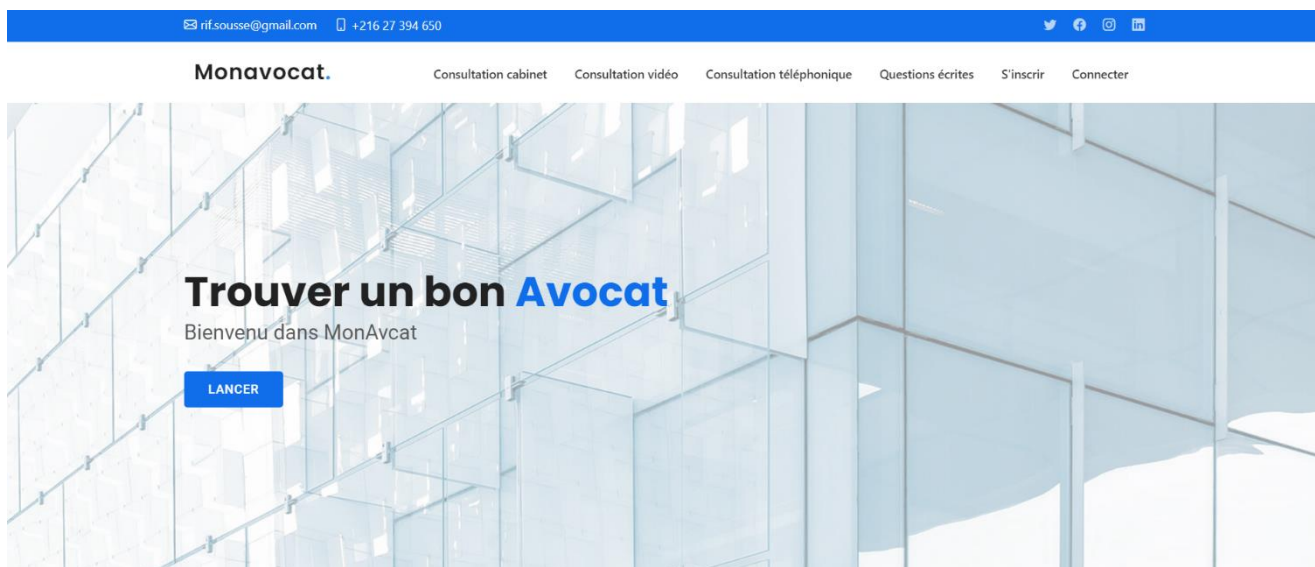


Figure 22: interface d'accueil

Cette interface (figures 23,24) permet aussi aux visiteurs d’avoir une idée concernant l’utilisation de ce site.

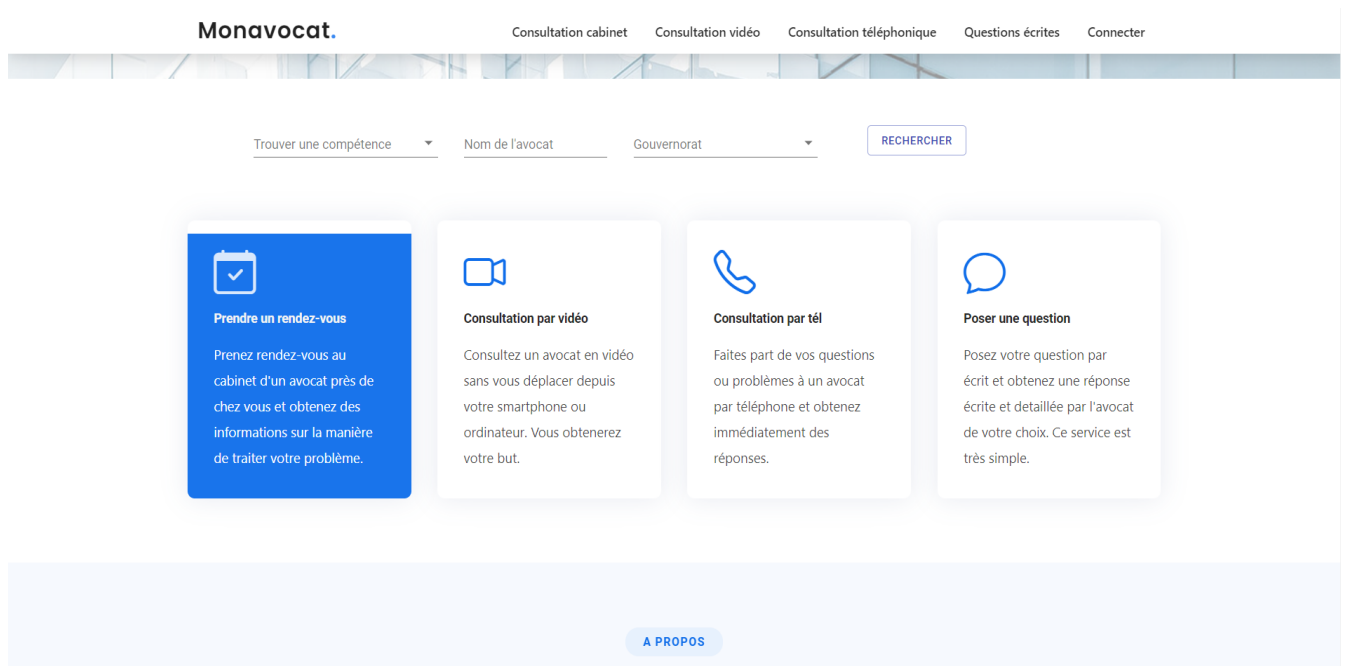


Figure 23: interface d'accueil

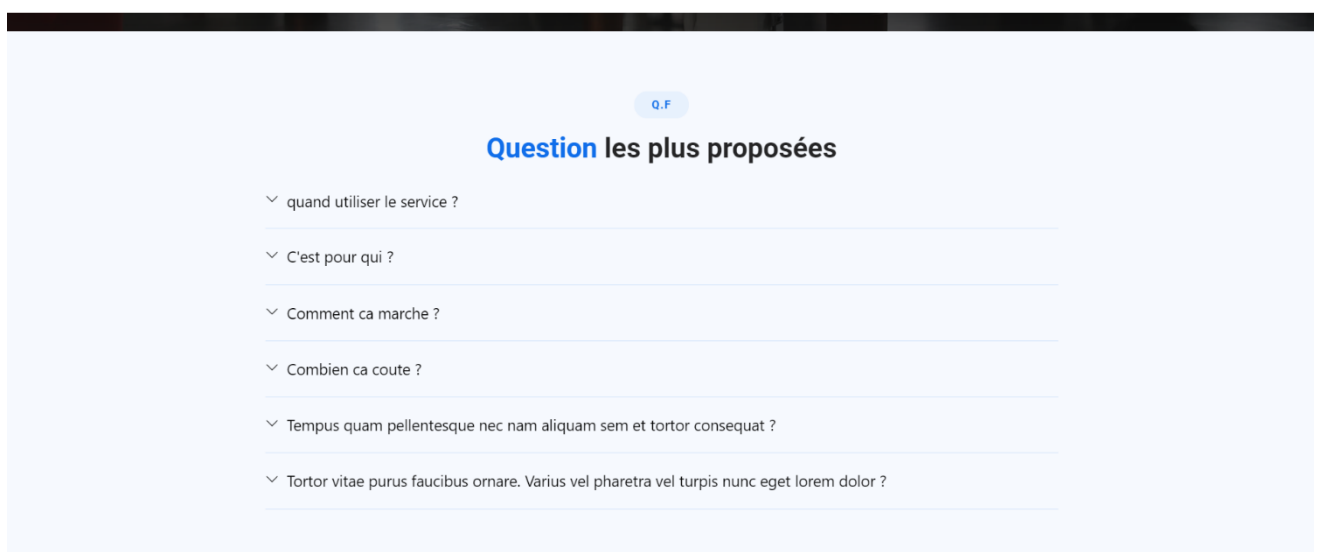


Figure 24 : interface question fréquentes

Cette page contient plusieurs fonctionnalités puisqu'elle permet aux visiteurs de faire une recherche sur les avocats présents dans ce site et visiter leurs profils pour avoir plus infos. Pour bien faciliter la recherche, elle peut être triée par les compétences, par nom ou bien par gouvernorat. (Figure 25,26).

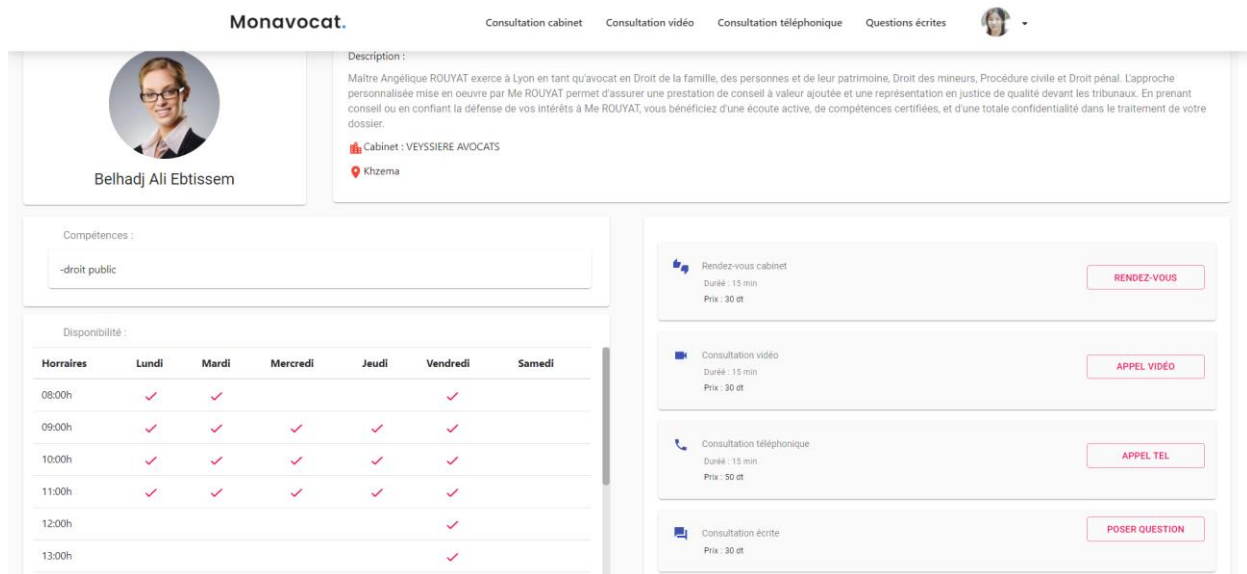
The screenshot shows the Monavocat website's search interface. At the top, there is a navigation bar with the Monavocat logo and links for 'Consultation cabinet', 'Consultation vidéo', 'Consultation téléphonique', 'Questions écrites', and 'Connecter'. Below the navigation bar is a large banner image showing hands writing on a document. The search bar is located below the banner, with a dropdown menu for 'Trouver une compétence' (currently set to 'Aucun') and input fields for 'Nom de l'avocat' and 'Gouvernorat'. A 'RECHERCHER' button is to the right of the input fields. Below the search bar, the text 'Comment ça marche ?' is displayed. Underneath, there are three steps: 1. 'Première étape' with a magnifying glass icon, 'Chosir votre avocat convenable à votre situation et le plus proche de vous.' 2. 'Deuxième étape' with a calendar icon, 'Prendre un rendez-vous à la date le plus tot possible.' 3. 'Troisième étape' with a speech bubble icon, 'Rencontrez votre avocat et disputer à propos votre problème.'

Figure 25: interface recherche

The screenshot shows the Monavocat website's search results page. At the top, there is a navigation bar with the Monavocat logo and links for 'Consultation cabinet', 'Consultation vidéo', 'Consultation téléphonique', 'Questions écrites', 'S'inscrire', and 'Connecter'. Below the navigation bar is a search bar with a dropdown menu for 'Trouver une compétence' (currently set to 'Droit Public') and input fields for 'Nom de l'avocat' and 'Gouvernorat'. A 'RECHERCHER' button is to the right of the input fields. Below the search bar, the text '2 avocats ont été trouvés' is displayed. Underneath, there are two lawyer profiles: 1. 'Khazema' with a photo of a woman, 'droit public', and a 'PRENDRE RENDEZ-VOUS' button. 2. 'kalaa' with a photo of a man, 'droit public', and a 'PRENDRE RENDEZ-VOUS' button. At the bottom, there are navigation buttons: 'Précédent', '1' (highlighted), and 'Suivant'.

Figure 26: liste des avocats

En finissant ça recherche un visiteur peut avoir plus d'idée sur un avocat en consultant son profil. (Figure 27).



Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Belhadj Ali Ebtissem

Description :
Maitre Angélique ROUYAT exerce à Lyon en tant qu'avocat en Droit de la famille, des personnes et de leur patrimoine, Droit des mineurs, Procédure civile et Droit pénal. L'approche personnalisée mise en oeuvre par Me ROUYAT permet d'assurer une prestation de conseil à valeur ajoutée et une représentation en justice de qualité devant les tribunaux. En prenant conseil ou en confiant la défense de vos intérêts à Me ROUYAT, vous bénéficiez d'une écoute active, de compétences certifiées, et d'une totale confidentialité dans le traitement de votre dossier.

Cabinet : VEYSSIERE AVOCATS
Khzema

Compétences :
-droit public

Disponibilité :

Horaires	Lundi	Mardi	Mercredi	Jeudi	Vendredi	Samedi
08.00h	✓	✓			✓	
09.00h	✓	✓	✓	✓	✓	
10.00h	✓	✓	✓	✓	✓	
11.00h	✓	✓	✓	✓	✓	
12.00h					✓	
13.00h					✓	

Rendez-vous cabinet
Durée : 15 min
Prix : 30 €t [RENDEZ-VOUS](#)

Consultation vidéo
Durée : 15 min
Prix : 30 €t [APPEL VIDÉO](#)

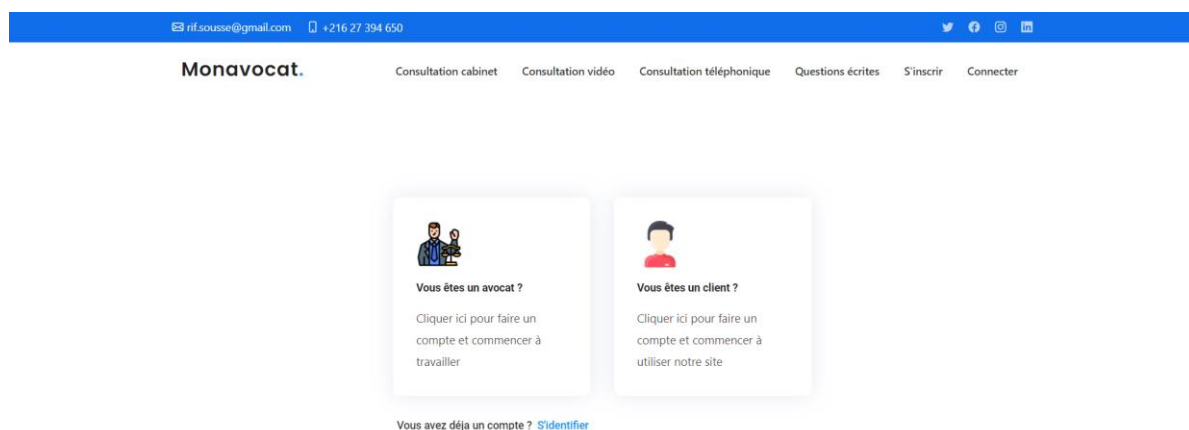
Consultation téléphonique
Durée : 15 min
Prix : 50 €t [APPEL TEL](#)

Consultation écrite
Prix : 30 €t [POSER QUESTION](#)

Figure 27: profil avocat

Puisqu'un visiteur n'a pas l'accès à toutes les fonctionnalités du site il est donc obligé de faire une inscription sur le site s'il n'a pas un compte, ou bien de se connecte.

L'inscription est permise au client ainsi qu'aux avocats. (Figure 28).



Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites S'inscrire Connecter

rif.sousse@gmail.com +216 27 394 650

Vous êtes un avocat ?
Cliquez ici pour faire un compte et commencer à travailler

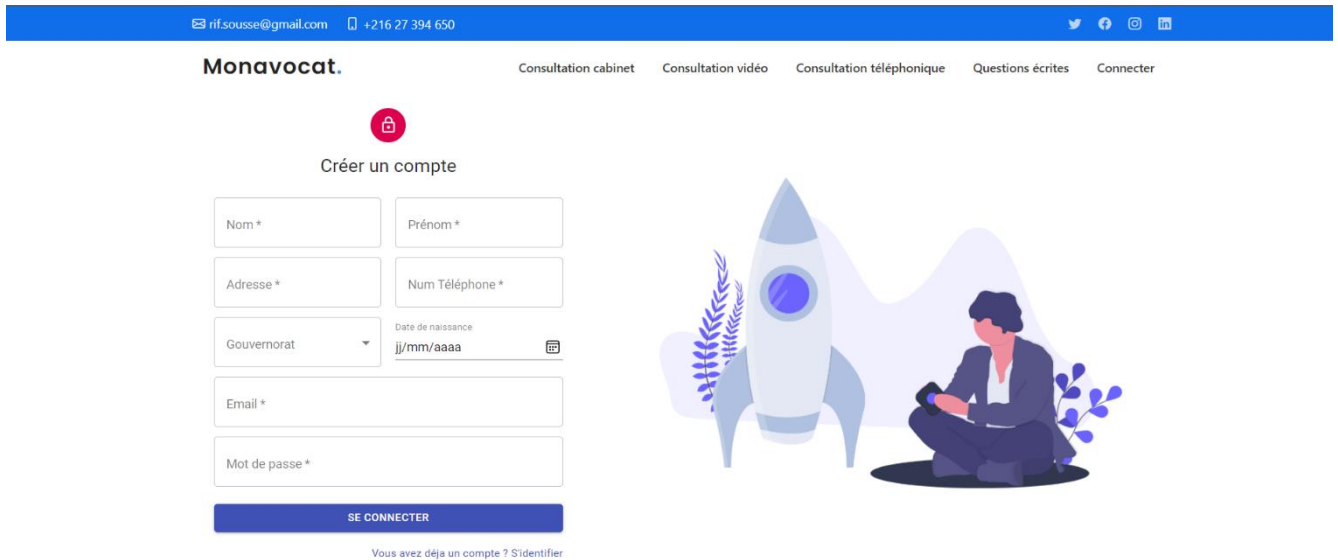
Vous êtes un client ?
Cliquez ici pour faire un compte et commencer à utiliser notre site

Vous avez déjà un compte ? [S'identifier](#)

Figure 28: interface avant inscription

Session Client :

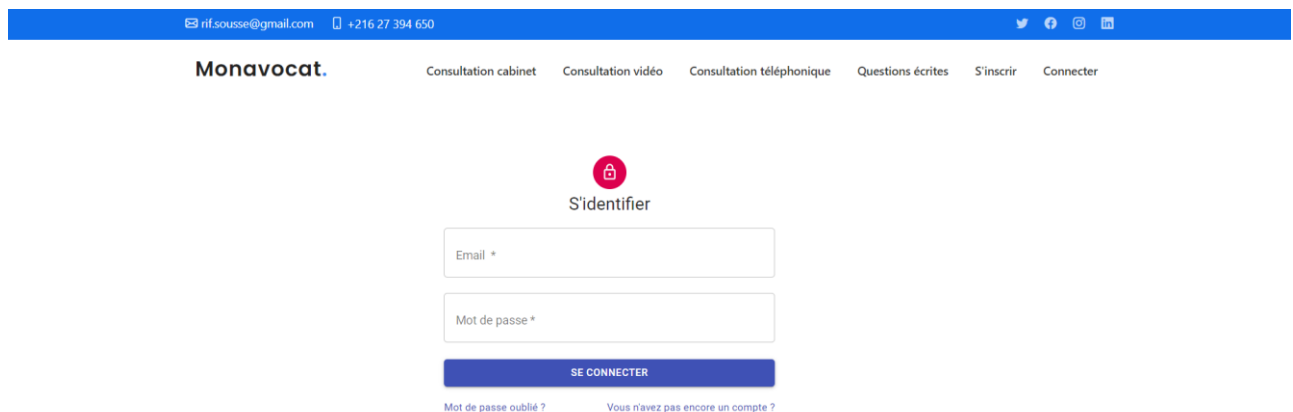
En choisissant de s'inscrire en tant qu'un client qui n'a pas encore un compte un formulaire d'inscription s'affiche. (Figure 29).



The screenshot shows the Monavocat website's client registration page. At the top, a blue header bar contains contact information (email: rlf.sousse@gmail.com, phone: +216 27 394 650) and social media icons. Below the header, the Monavocat logo is on the left, and navigation links (Consultation cabinet, Consultation vidéo, Consultation téléphonique, Questions écrites, Connecter) are on the right. The main content area features a red lock icon and the title "Créer un compte". The registration form includes fields for Nom *, Prénom *, Adresse *, Num Téléphone *, Gouvernorat (dropdown), Date de naissance (jj/mm/aaaa with a calendar icon), Email *, and Mot de passe *. A blue "SE CONNECTER" button is at the bottom of the form. To the right of the form is an illustration of a person sitting cross-legged next to a rocket ship. Below the form, a link "Vous avez déjà un compte ? S'identifier" is visible.

Figure 29: formulaire d'inscription client

Pour un visiteur quel que soit un client ou un avocat admettant déjà un compte il est obligé de remplir un petit formulaire d'authentification pour accéder à son compte (figure 30).



The screenshot shows the Monavocat website's login page. The header bar is identical to the registration page. The Monavocat logo is on the left, and navigation links (Consultation cabinet, Consultation vidéo, Consultation téléphonique, Questions écrites, S'inscrire, Connecter) are on the right. The main content area features a red lock icon and the title "S'identifier". The login form includes fields for Email * and Mot de passe *. A blue "SE CONNECTER" button is at the bottom of the form. Below the form, two links are visible: "Mot de passe oublié ?" and "Vous n'avez pas encore un compte ?".

Figure 30: formulaire d'authentification

Juste après avoir été inscrit puis authentifié en tant qu'un client ou un avocat une interface d'accueil est lancée, cette interface semblable à l'interface d'accueil d'un visiteur. Pour un utilisateur il a les mêmes accès qu'un visiteur mais avec d'autres options.

Un utilisateur a l'accès à la prise de rendez-vous en remplissant un formulaire, il doit : envoyer sa situation puis choisir la date et l'heure disponible et l'heure et confirmer ses informations par son numéro de téléphone (figure 31,32,33).

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Prendre rendez-vous

Belhadj Ali Ebtissem
Khazema
Rendez-Vous Cabinet : 30 Dt /15 min

Votre situation*

Utilisez svp cet espace pour indiquer l'objet de votre consultation , de sorte à ce que votre avocat puisse en avoir connaissance préalablement.

Vos disponibilités*

Préférence horaire : Date Temps

Disponibilité de l'avocat Ebtissem Belhadj Ali :

Horaires	Lundi	Mardi	Mercredi	Jeudi	Vendredi	Samedi
08:00h	✓	✓			✓	
09:00h	✓	✓	✓	✓	✓	
10:00h	✓	✓	✓	✓	✓	
11:00h	✓	✓	✓	✓	✓	
12:00h					✓	
13:00h					✓	
14:00h	✓	✓	✓	✓	✓	
15:00h	✓	✓	✓	✓		
16:00h	✓	✓	✓	✓		
17:00h	✓	✓	✓	✓		
18:00h						

Figure 31: interface de prise de rendez-vous

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Prendre rendez-vous

Belhadj Ali Ebtissem
Khazema
Rendez-Vous Cabinet : 30 Dt /15 min

Votre situation*

Bonjour

Vos disponibilités*

Préférence horaire : Date Temps

Figure 32: sélection de la date et heure du rendez-vous

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Prendre rendez-vous

Belhadj Ali Ebtissem
Khazema
Rendez-Vous Cabinet : 30 Dt /15 min

Votre numéro*

Entrez votre numéro de téléphone

VALIDER **RETOURNER**


Figure 33: validation numéro de téléphone

En validant ses informations l'interface de paiement est lancée et l'utilisateur remplit le formulaire de paiement pour valider la prise de rendez-vous. (Figure 34).

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

RÉCAPITULATIF DE VOTRE DEMANDE

Demande auprès de



Belhadj Ali Ebtissem
Khzema

Type de commande
Consultation Vidéo

Durée
15 min

Votre disponibilité
15/6/2021



Tarif **30 Dt**

Païement

Numéro de carte:

Date d'expiration: YY/MM

Code de vérification:


PAYER **RETOURNER**

Figure 34: interface de paiement

Pour un client il a le droit de consulter son profil et de le gérer. il peut modifier ces informations ainsi que changer son mot de passe et d'accéder à ses rendez-vous. (Figure 35,36).

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Paramètres de compte



asma methamem

Mon profile

Nom: Prénom:

Email: Numéro téléphone:

Gouvernorat: Adresse:

Date:

Enregistrer

Profile

Sécurité

Rendez-Vous

Consultations Faites

Figure 35: profil du client

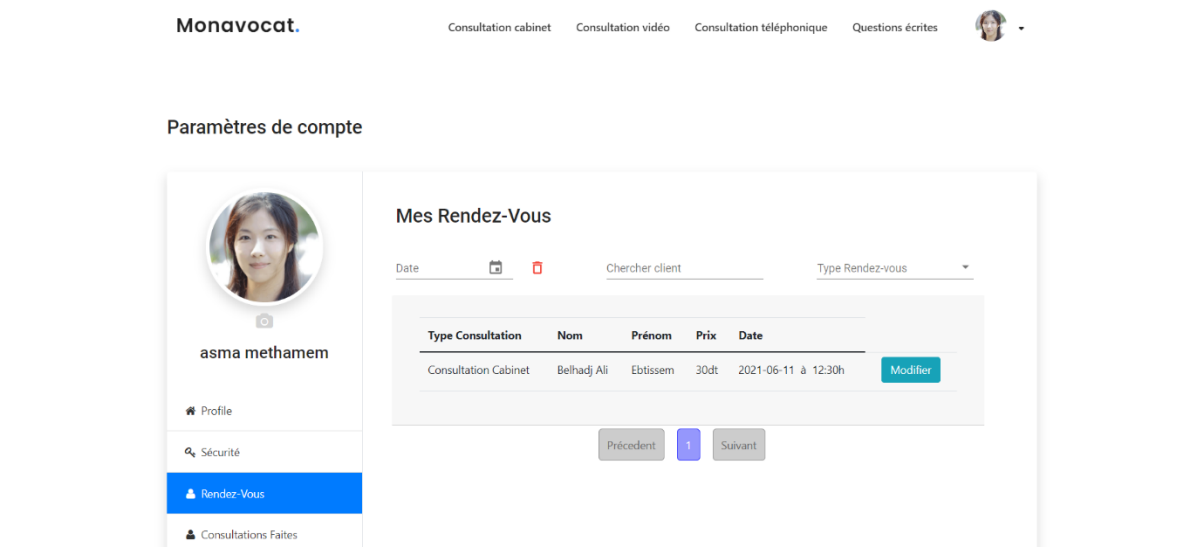


Figure 36: liste de rendez-vous client

Un client peut poser à un avocat et ce dernier peut le reprendre. (Figure 37)

Monavocat.

Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Question Ecrite

Belhadj Ali Ebtissem
Khzema
Question Ecrite : 30 Dt

Votre situation*

Utilisez svp cet espace pour indiquer l'objet de votre consultation , de sorte à ce que votre avocat puisse en avoir connaissance préalablement..

Vos disponibilités*

Temps estimé pour répondre : 24h

VALIDER

Figure 37 : envoyer une question à un avocat

Chaque client a le droit de consulter sa liste de consultation faite et voire son historique de consultations. (Figure 38).

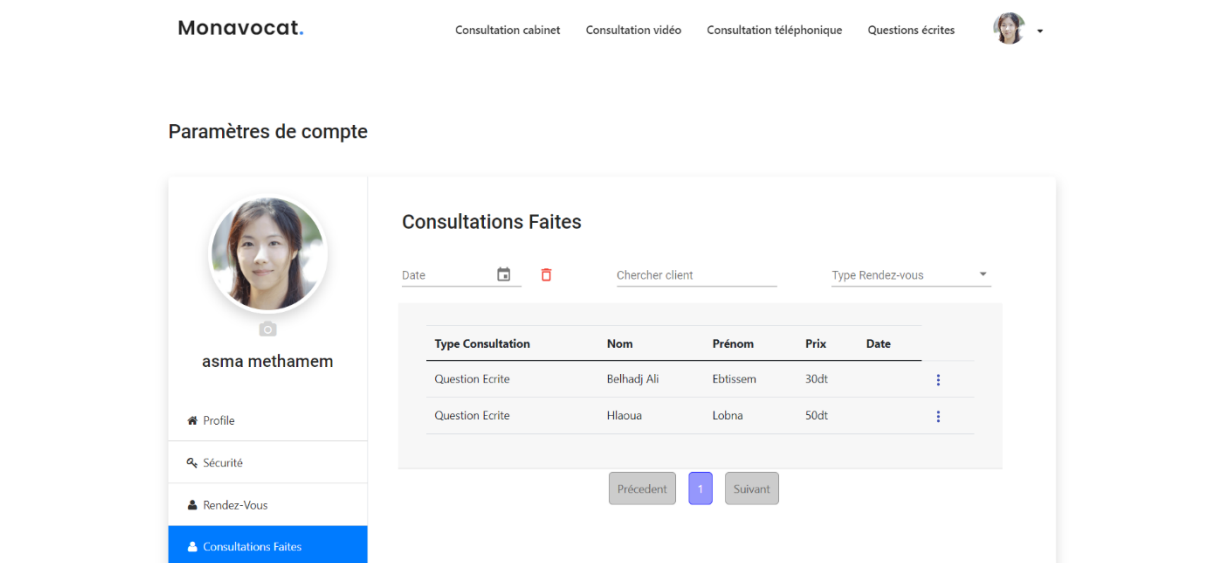


Figure 38 : interface historique des consultations

Le client peut aussi voir les détails de ses consultations comme la montre la figure suivante (Figure 39).

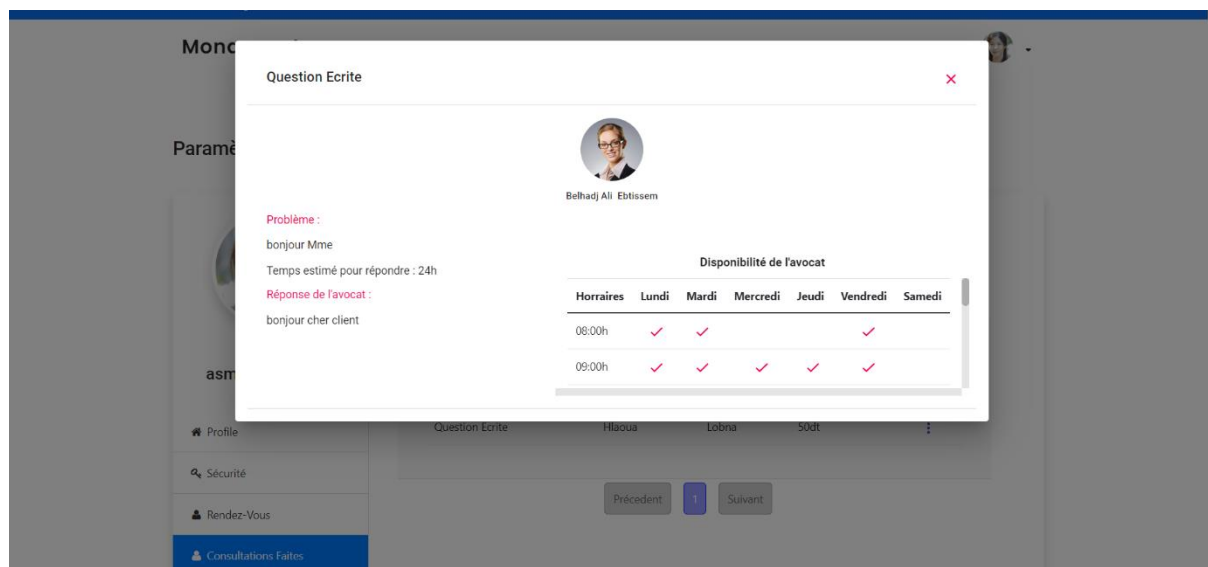


Figure 39 : détails de consultation

Session avocat :

Pour un avocat, un avocat non inscrit dans ce site doit tout d'abord saisir son cin pour vérifier ses informations. (Figure 40).

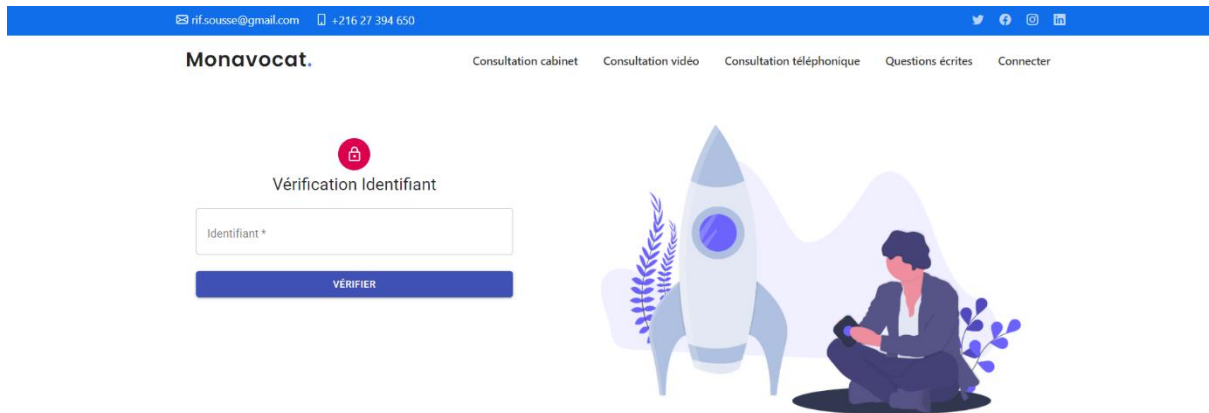


Figure 40 : vérification cin avocat « pré-inscription »

Ainsi un formulaire déjà rempli s'affiche. Afin d'avoir compléter le remplissage ce formulaire l'interface d'authentification s'affiche. (Figure 41).

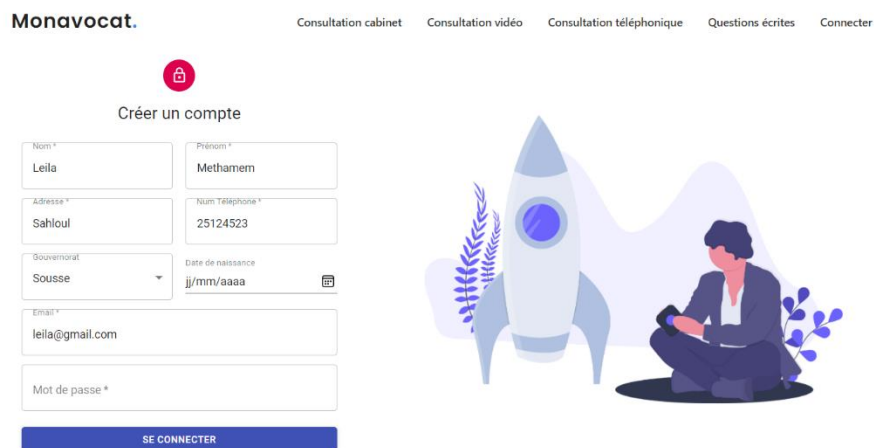


Figure 41 : formulaire inscription avocat

Un avocat aussi peut consulter son profil. (Figure 42).

The screenshot shows the 'Mon profil' (My profile) page on the Monavocat website. The user is Ben hadj Ali Ibtissem. The page is divided into a left sidebar with navigation links (Profil, Sécurité, Tarifs, Rendez-Vous, Mes Plannings) and a main content area. The main area contains a form with the following fields: Nom (Ben hadj Ali), Prénom (Ibtissem), Email (sarah@gmail.com), Numéro téléphone (23444262), Gouvernorat (Sousse), Adresse (Mourouj), Date (05/05/2021), Spécialité (Droit Public), Cin (21372703), and a Bio field. An 'Enregistrer' (Save) button is at the bottom.

Figure 42 : profil avocat


Il peut aussi changer son mot de passe (cette interface est la même pour un profil client). Ainsi qu'accéder à ses rendez-vous et pouvoir les modifier.(Figure 43,44,45).

The screenshot shows the 'Paramètres de compte' (Account settings) page on the Monavocat website. The user is Ben hadj Ali Ibtissem. The page is divided into a left sidebar with navigation links (Profil, Sécurité, Tarifs, Rendez-Vous, Mes Plannings) and a main content area. The main area contains a form titled 'Mot de passe' (Password) with the following fields: Mot de passe courant (Current password), Nouveau mot de passe (New password), and Confirmer nouveau mot de passe (Confirm new password). An 'Enregistrer' (Save) button is at the bottom.

Figure 43 : modification mot de passe




Paramètres de compte



Belhadj Ali Ebtissem

- Profile
- Sécurité
- Tarifs
- Mes Plannings
- Rendez-Vous**
- Consultations Faites

Mes Rendez-Vous


Date  Chercher client Type Rendez-vous ▼

Type Consultation	Nom	Prénom	Prix	Date	
Consultation Cabinet	Hlaoua	Lobna	30dt	2021-06-14 à 09:30h	Modifier Confirmer
Consultation Cabinet	asma	methamem	30dt	2021-06-11 à 12:30h	Modifier

Précédent 1 Suivant

Figure 44 : liste rendez-vous avocat

Consultation Cabinet



Methamem Asma
Numéro : 21372703

Problème :
Bonjour

Date 01/06/2021 Temps 08:15

[ENREGISTRER](#) [CONFIRMER](#)

Methamem Asma

Numéro Téléphone : 21372703

Date : 2021-05-28 à 07:15

[Confirmer](#) [Modifier](#)

Figure 45 : modification rendez-vous

Un avocat peut gérer son planning et même mettre son tarif à travers son profil. (Figure 46,47).

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Mes Plannings

Jour	Début de travail	Fin de travail	Pause entre consultation	Pause midi	Statut
Lundi	8h	18h	30 min	12h-13h	Actif
Mardi	8h	18h	30 min	12h-13h	Actif
Mercredi	8h	18h	30 min	12h-13h	Actif
Jeudi	8h	18h	30 min	12h-13h	Actif
Vendredi	8h	18h	30 min	12h-13h	Actif
Samedi	8h	18h	30 min	12h-13h	Inactif

Enregistrer

Figure 46 : mettre à jour planning avocat

Monavocat. Consultation cabinet Consultation vidéo Consultation téléphonique Questions écrites

Paramètres de compte

Mes Tarifs

Type de consultation	Prix	Durée	Statut
Prix Consultation Cabiné en DT	80	30 min	Actif
Prix Consultation Vidéo en DT	100	30 min	Actif
Prix Consultation Téléphonique en DT	50	15 min	Actif
Prix Question Ecrite en DT	30		Actif

Enregistrer

Figure 47 : mettre à jour tarif avocat

Après la prise, modification ou bien la confirmation d'un rendez-vous un mail sera envoyé au client et de même pour l'avocat. (Figure 48).

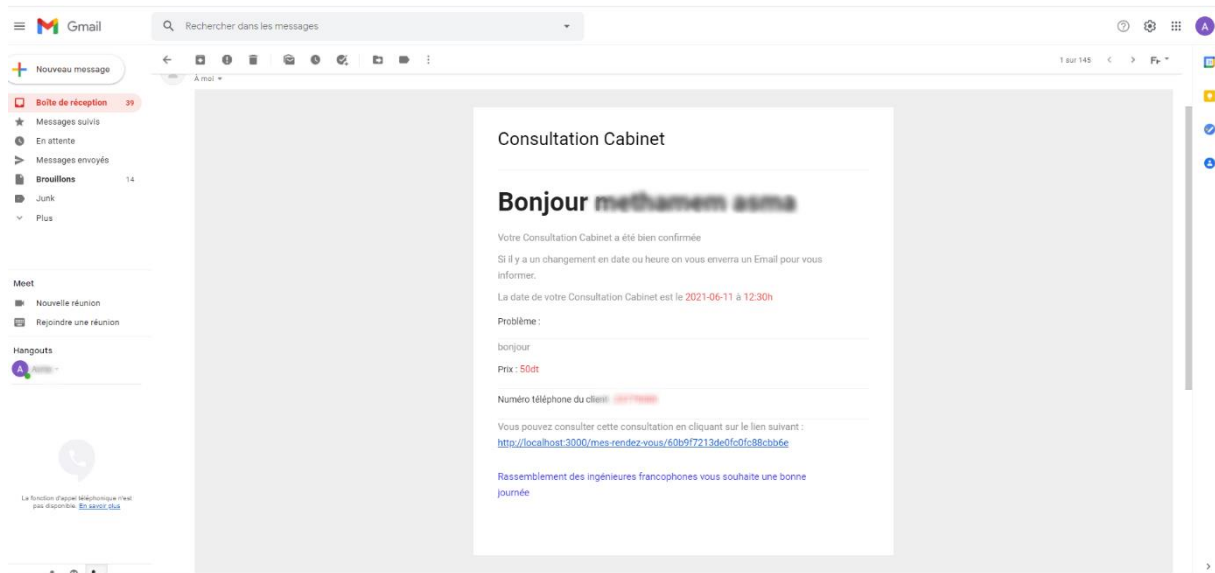


Figure 48 : email de confirmation

Session Administrateur :

Pour rendre la gestion du site plus efficace il faut avoir un administrateur, cet administrateur a plusieurs taches à faire pour avoir des bonnes conditions de navigation.

En effet il est nécessaire à l'administrateur d'avoir été authentifié pour faire ces missions.

Cette figure (49) présente l'interface login pour l'administrateur.

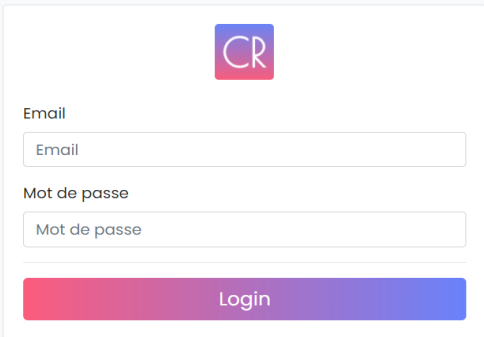


Figure 49: login administrateur

Il suffit l'authentification est validée, les interfaces de gestion s'affichent. A propos de cette interface il présente un annuaire d'avocat là ou l'administrateur peut ajouter, modifier ou supprimer un avocat selon les informations disponibles à « l'ordre national des avocats » pour vérifier la confidentialité totale. (Figure 50).

CIN	NOM	PRENOM	EMAIL	SPECIALITE	GOUVERNORAT	
						Edit Delete
						Edit Delete
						Edit Delete
						Edit Delete
						Edit Delete
						Edit Delete
						Edit Delete
						Edit Delete

Figure 50: annuaire avocats

Cette interface permet à un administrateur d’accéder à la liste des clients inscrit dans ce site, dont il peut juste supprimer un compte. (figure 51).

NOM	PRENOM	EMAIL	
			Delete
			Delete
			Delete
			Delete
			Delete
			Delete
			Delete
			Delete

Figure 51: interface de liste des clients inscrits

Cette interface presente la liste des avocats inscrit dans ce site là où il peut supprimer un compte avocat.(figure 52).

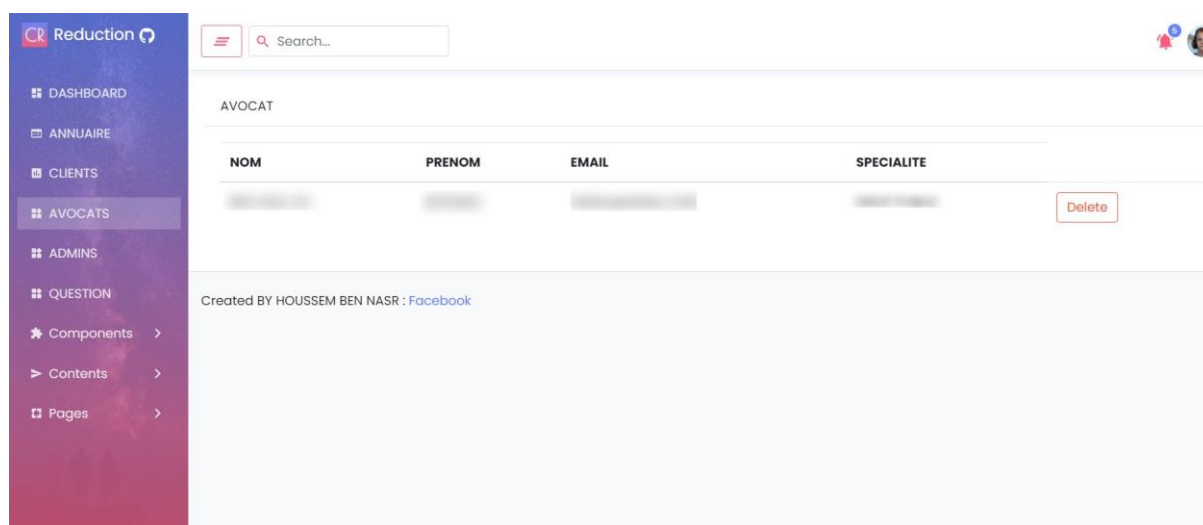


Figure 52: interface de liste des avocats inscrits

Conclusion :

Dans ce chapitre, nous avons parlé de notre environnement de travail tout en citant les technologies utilisées, les langages de programmations adaptés à ce projet et les outils utilisés. Ensuite, nous avons la partie de test et validation qui présente les interfaces et leurs utilités.

CONCLUSION GENERALE :

Durant la période de notre projet de fin d'étude passé au sein de la boîte de développement « **RIF** » nous avons pratiqué les connaissances acquises tout au long de notre formation académique à l'école supérieure des sciences et de la technologie de Hammem-Sousse. Ce projet nous a permis de constater un lien entre connaissances acquises durant notre formation et leur pratique au sein d'une entreprise professionnelle.

Nous avons essayé d'adopter les meilleures solutions techniques et méthode de développement. Dans un premier lieu, nous avons commencé par introduire le cadre de l'élaboration du projet ainsi qu'une présentation de sujet de ce projet qui est basé sur les difficultés qui peuvent intervenir chez un client lors de sa recherche d'un avocat, suivit d'une étude de l'existant.

Ensuite en essayant de prendre en considération les solutions déjà existantes, nous avons essayé de répondre à ces problèmes par des solutions que nous avons proposées. Nous avons trouvé ces solutions par des études théoriques et pratiques grâce aux méthodes de travail, la conception et le choix des outils techniques.

Bien que les principaux objectifs de notre projet soient atteints, mais comme tout travail, des difficultés ont été rencontrées tout au long de ce stage, notamment la contrainte du temps et le manque de documentations. Cependant, malgré toutes ces difficultés, les objectifs ont été atteints. Cette application pourrait prochainement être un sujet de quelques améliorations des futurs travaux.

Enfin ce stage nous a permis d'améliorer et approfondir nos connaissances techniques en assistant à des travaux afin de découvrir en près le domaine de ce métier, ses acteurs, ses contraintes, participer à ses enjeux à travers des missions confiées.

WEBOGRAPHIE

[1] **GM-Savoir-metier** Architecture en couches. [En ligne] [Consulté le 18 Mai 2021] :
<https://sites.google.com/site/gldmrmtl/genie-logiciel/pratiques/architecture-en-couches>

[2] **Wikipédia** Model-vue-Contrôleur. [En ligne] [Consulté le 18 mai 2021] :
<https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur>

[3] : **ibracilinks** ReactJS. [En ligne] [Consulté le 31 mai 2021] :
<https://ibracilinks.com/blog/quest-ce-que-reactjs-et-pourquoi-devrions-nous-utiliser-reactjs>

[4] : **makinacorp** NodeJS. [En ligne] [Consulté le 31 mai 2021] :
<https://makina-corp.com/blog/metier/2014/introduction-a-nodejs>

[5] : **journaldunet** Axios. [En ligne] [Consulté le 31 mai 2021] :
<https://www.journaldunet.fr/web-tech/developpement/1441159-axios-concevoir-une-requete-post-pour-renvoyer-les-donnees-d-un-formulaire/#:~:text=Axios%20est%20une%20biblioth%C3%A8que%20JavaScript,requ%C3%AAtes%20avec%20la%20m%C3%A9thode%20POST.>

[6] : **expressjs** Express. [En ligne] [Consulté le 31 mai 2021] :
<https://expressjs.com/fr/>

[7]: **developer.mozilla** JavaScript. [En ligne] [Consulté le 31 mai 2021] :
<https://developer.mozilla.org/fr/docs/Web/JavaScript>

[8] : **developer.mozilla** HTML. [En ligne] [Consulté le 31 mai 2021] :
<https://developer.mozilla.org/fr/docs/Web/HTML>

[9] : **developer.mozilla** CSS. [En ligne] [Consulté le 31 mai 2021] :
https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/CSS_basics

[10] : **framlibre** VsCode. [En ligne] [Consulté le 31 mai 2021] :
<https://framalibre.org/content/visual-studio-code>

[11] : **cours-informatique** Microsoft-office-Word. [En ligne] [Consulté le 31 mai 2021] :
<https://cours-informatique-gratuit.fr/dictionnaire/microsoft-office-word/>

[12] : **Wikipédia** Jira. [En ligne] [Consulté le 31 mai 2021] :
<https://fr.wikipedia.org/wiki/Jira>

[13] : **mongodb** Mongodb. [En ligne] [Consulté le 31 mai 2021] :

<https://www.mongodb.com/what-is-mongodb>

[14] : **openclassrooms** Postant. [En ligne] [Consulté le 31 mai 2021] :

<https://openclassrooms.com/en/courses/4668056-construisez-des-microservices/5123020-testez-votre-api-grace-a-postman>

FORMULAIRE DE « DÉCLARATION »

DECLARATION SUR L'HONNEUR

Par la présente, nous soussignés **Asma Methamem et Housseem Nasr Ben Hadj Amor** certifions avoir rédigé notre mémoire de stage intitulé : **Conception et réalisation d'une application web pour rechercher et contacter des avocats « MonAvocat.tn »** seuls(e) et sans aucune aide interdite. Nous ne nous sommes basés(e) sur aucune autre source que celles mentionnées et citées dans le manuscrit. Nous confirmons avoir apporté les corrections qui nous ont été demandées par les membres du jury de la soutenance.

Ce mémoire n'a ni dans sa forme actuelle, ni dans une forme proche, été soumis à une autre institution.

Hamam Sousse, le/...../.....

Signature

Par la présente, je soussigné **Lobna Hlaoua** Superviseur de **Asma Methamem et Housseem Nasr Ben Hadj Amor**, certifie avoir lu le manuscrit intitulé : **Conception et réalisation d'une application web pour rechercher et contacter des avocats « MonAvocat.tn »** et être en accord avec son contenu et il est en accord avec les objectifs pédagogiques visés par le stage de fin d'études.

Hamam Sousse, le/...../.....

Signature

RESUME

Ce travail a été élaboré dans le cadre du projet de fin d'étude pour l'obtention du diplôme de licence fondamentale en sciences de l'informatique qui a été réalisé au sein de la société **RIF**.

Notre projet consiste à concevoir et développer une application web de prise de rendez-vous et consultation en ligne pour la recherche et la gestion des avocats.

Ce projet est structuré par quatre chapitres : tout d'abord nous présentons le cadre général du stage. Ensuite nous analysons les besoins pour ce projet par l'étude des besoins fonctionnels et non fonctionnels, ainsi que l'étude conceptuelle au quelle on spécifie l'architecture logiciel choisie et finalement après ces études et ces analyses nous mentionnons le test et validation de notre application par des interfaces expliquant les principales fonctionnalités.

Abstract

This work was developed as part of the end of study project for obtaining the basic license degree in computer science which was carried out within the company **RIF**.

Our project consists in designing and developing a web application for making appointments and online consultation for the search and management of lawyers.

This project is structured by four chapters: first we present the general framework of the internship. Then we analyze the needs for this project by the study of the functional and non-functional needs, as well as the conceptual study in which we specify the chosen software architecture and finally after these studies and these analyzes we mention the test and validation of our application by interfaces explaining the main functionalities.

