



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

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : [Génie Informatique]

Par :

AOUCI Mhamed

CHACHOUA Abdelkader

Sur le thème

Elaboration d'un système de suscription Multi-Tenant SaaS avec une Base de données unique

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

Mr Zioual Tahar

MAA Université tiaret

Président

Mme Laaredj Zohra

MAA Université tiaret

Examinateur

Mr Mokhtari Ahmed

MAA Université tiaret

Encadreur

2023-2024

Remerciements

Avant tout, nous remercions Dieu le très haut qui nous a donné le courage et la volonté de réaliser ce modeste travail.

« (CELUI QUI NE REMERCIE PAS LES GENS, NE REMERCIE PAS ALLAH.) »

[Authentique Hadith]

Nous tenons également à exprimer nos sincères remerciements à notre chef de projet, Monsieur [Mokhtari Ahmed], pour son soutien continu et ses précieux conseils. Ses idées et ses recommandations ont été très utiles au cours de cette projet.

Nous tenons également à remercier tout particulièrement les membres du jury qui ont généreusement donné de leur temps pour évaluer ce modeste travail.

Nos remerciements vont également à toute l'équipe pédagogique de [Génie Civil] pour la qualité de leur enseignement et leur accompagnement tout au long de nos études.

Enfin, nous tenons à remercier chaleureusement notre famille et nos amis pour leur soutien et leurs encouragements tout au long de ce projet. Leur présence et leurs encouragements ont été pour nous une précieuse source d'inspiration et de motivation .

Dédicace

À nos très chers parents,

Aucun mot, aucune dédicace ne peut exprimer notre respect, notre considération et l'amour éternel pour les sacrifices que vous avez consentis pour notre instruction et notre bien-être.

Votre générosité et votre bonneté ont toujours été un exemple pour nous tous.

Trouvez en ce travail le fruit de votre dévouement de votre patience et l'expression de notre gratitude et profond amour

Je t'aime papa je t'aime maman et sachez que je suis très reconnaissant

À toute notre chère famille,

À nos professeurs,

À nos chers amis,

À nos chers collègues,

À tous ceux qui nous ont aidé de près ou de loin,

< Mhamed & AEK >

Résumé

Le modèle Software as a Service (SaaS) consiste à offrir des services logiciels en ligne et à la demande via Internet comme support de diffusion principal. En tant qu'outil d'exploitation d'économie d'échelle, les modèles SaaS favorisent la Multi-Tenancy, notion de partage d'instances entre un grand groupe de clients, appelés locataires.

les architectures multi locataires se sont avérées essentiels en tant qu'avantages supplémentaires pour les firmes qui utilisent des systèmes virtualisés tels que le Cloud, le Fog, l'Edge, et autres. Elles sont considérées aujourd'hui comme une compétence vitale pour les ingénieurs et les développeurs .

Le principal objectif de ce mémoire est de dissiper l'ambiguïté conceptuelle et de relever les défis d'une l'implémentation liés à cette tendance, en utilisant un système d'abonnement simple pour la prise de rendez-vous en ligne comme application cible. Ce mémoire se concentre sur l'utilisation du Framework Laravel pour développer cette application d'abonnement distincte.

Mots-clés :

Systeme virtualisé, Cloud, SaaS, Multi-tenancy.

Abstract

The Software as a Service (SaaS) model involves offering software services online and on-demand via the Internet as the primary delivery medium. As an economy of scale operating tool, SaaS models promote Multi-Tenancy, the notion of sharing instances between a large group of customers, called tenants.

Multi-tenant architectures have proven to be essential as additional benefits for firms using virtualized systems such as Cloud, Fog, Edge, and others. They are considered today a vital skill for engineers and developers .

The main objective of this dissertation is to dispel the conceptual ambiguity and address the implementation challenges associated with this trend, using a simple Book Appointment Online subscription system as the target application. This dissertation focuses on using the Laravel Framework to develop this distinct subscription application.

Keywords :

Virtualized system, Cloud, SaaS, Multi-tenancy.

ملخص

يتضمن نموذج البرنامج كخدمة (SaaS) تقديم خدمات برمجية عبر الإنترنت وعند الطلب عبر الإنترنت كوسيلة التسليم الأساسية. باعتبارها أداة تشغيل اقتصادية الحجم، تعمل نماذج SaaS على تعزيز Multi-Tenancy ، وهي فكرة مشاركة المثيلات بين مجموعة كبيرة من العملاء، تسمى المستأجرين.

لقد أثبتت البنى التحتية متعددة المستأجرين أهميتها باعتبارها فوائد إضافية للشركات التي تستخدم أنظمة افتراضية مثل Cloud و Fog و Edge وغيرها. وهي تعتبر اليوم مهارة حيوية للمهندسين والمطورين.

الهدف الرئيسي من هذه الأطروحة هو تبديد الغموض المفاهيمي ومعالجة تحديات التنفيذ المرتبطة بهذا الاتجاه، وذلك باستخدام نظام اشتراك بسيط لحجز المواعيد عبر الإنترنت باعتباره التطبيق المستهدف. تركز هذه الأطروحة على استخدام Laravel Framework لتطوير تطبيق الاشتراك المتميز هذا.

الكلمات الاساية :

نظام افتراضي ,سحابة , Saas ,متعدد الايجار .

Table des matières

Remercîments	1
Résumé	3
Table des matières	5
Table des figures	6
Liste des tableaux	8
Liste d'abréviation	8
Introduction générale	9
Contexte & Sujet	9
Défis, objectif et énoncé de mémoire	9
Structure Générale	10
 <i>Chapitre I : Généralités sur le Cloud Computing :</i>	
1.1 Introduction	12
1.2 Historique	12
1.3 Définition	13
1.4 Eléments constitutifs du Cloud Computing	14
1.5 Caractéristiques du Cloud Computing	16
1.6 Modèles de déploiement du Cloud Computing.....	17
1.7 Types de services dans les environnements de Cloud Computing	20
1.8 Acteurs du Cloud Computing	22
1.9 Relations entre les acteurs du Cloud Computing	23
1.10 Avantages et inconvénients du Cloud Computing.....	24
Conclusion	25
 <i>Chapitre 2 :La Multi Tenant dans les modèles de service "SaaS":</i>	
2.1 Introduction	27
2.2 Définition	27
2.2.1 Multi-tenant	27
2.2.2 Single -tenant	27
2.3 Comparaison entre Single-tenant et Multi-tenant	28

2.4 L'approche orienté base de données.....	29
2.5 Avantages et inconvénients du Cloud Computing.....	32
2.6 Les Framework qui supporte la multi-tenant.....	32
2.7 Travaux connexes Migration vers le multi-tenant SaaS.....	33
Conclusion.....	33
Chapitre 3 : Conception et réalisation :	
3.1 Introduction.....	35
3.2 Description du système.....	35
3.3 La Modélisation du Système.....	36
3.4 Les outils d'implantation et de réalisation :.....	39
3.5 Les Interface du système.....	41
Conclusion.....	50
Conclusion Général & Perspectives	51
Bibliographie	53

Table des figures

Chapitre 1 : Généralités sur le Cloud Computing

Figure 1.1 : Cloud Computing.....	13
Figure 1.2 : Exemple de virtualisation.....	14
Figure 1.3 : Datacenter.....	14
Figure 1.4 : Plateforme collaborative.....	15
Figure 1.5 : Les Modèles de déploiement du Cloud Computing.....	17
Figure 1.6 : Cloud public.....	18
Figure 1.7 : Cloud Privé.....	18
Figure 1.8 : Cloud hybride.....	19
Figure 1.9 : Cloud communautaire.....	19
Figure 1.10 : Types de services Clouds.....	20

Figure 1.11 : Relations entre les acteurs du Cloud Computing.....	23
Figure 1.12 : Obstacles à l'adoption du Cloud Computing	25

Chapiter 2 : La Multi Tenant dans les modèles de service "SaaS":

Figure 2.1 : Illustration de l'approche multi-locataire	27
Figure 2,2 : Illustration de l'approche Single-tenant.....	28
Figure 2.3 : Stratégie de stockage des données.....	29
Figure 2.4 : L'approche orienté base de données unique.....	29
Figure 2.5 : Base de données unique, schéma multiple.....	30
Figure 2,6 : Base de données unique, schéma unique.....	31

Chapiter 3 : Conception et réalisation :

Figure 3.1 : Diagramme de cas d'utilisation	36
Figure 3.2 : Diagramme de classe	37
Figure 3.3 : Diagramme de sequence.....	38
Figure 3.4 : Interface principale de Health Track.....	41
Figure 3.5 : Interface de Connexion.....	42
Figure 3.6 : Interface de Compte Admin.....	42
Figure 3,7 : Interface d'afficher les plan.....	43
Figure 3.8 : Interface d'ajouter plan.....	43
Figure 3.9 : Interface d'afficher la liste des médecins.....	44
Figure 3.10 : Interface d'ajouter un médecins.....	44
Figure 3,11 : Interface de déconnexion.....	45
Figure 3.12 : Interface d'inscription.....	45
Figure 3.13 : Interface de Connexion.....	46
Figure 3.14 : Interface s'affiche Avec son lien qu'il fournit aux clients.....	46
Figure 3,15 : L'interface de l'Abonnement.....	47
Figure 3.16 : Interface du pays d'abonnement.....	47
Figure 3.17 : Interface de votre plan d'abonnement.....	48

Figure 3.18 : Interface d'ajouter et Consulter la liste des rendez vous.....	48
Figure 3.19 : Interface des rendez-vous	49
Figure 3.20 : Interface d'inscription.....	49
Figure 3.21 : Interface des prendre et consulté les rendez-vous.....	50

Liste des tableaux

Tableau 1.1 : Les grands acteurs mondiaux du Cloud.....	21
Tableau 1.2 : Avantages et inconvénients des services	24
Tableau 2.1 : Comparaison entre Single-tenant et Multi-tenant.....	28
Tableau 3.1 : Les caractéristiques de l'ordinateur	38

Liste d'abréviation

IBM	: International Business Machines Corporation
ISO	: The International Organization for Standardization
NIST	: National Institute of Standards and Technology
VM	: Machines virtuelles
PaaS	: Platform as a Service
SaaS	: Software as a Service
IaaS	: Infrastructure as a Service
CISCO	: Commercial & Industrial Security Corporation
MVC	: Model-View-Controller
UML	: Unified Modeling Language
NIST	: National Institute of Standards and Technology

Introduction générale

Contexte & Sujet ;

Le Cloud Computing et le modèle SaaS (Software as a Service) ont révolutionné la façon dont les entreprises gèrent leurs infrastructures informatiques et déploient des applications. Le Cloud Computing permet l'accès à des ressources informatiques à la demande via Internet, offrant une flexibilité et une évolutivité inégalées. Le modèle SaaS va plus loin en fournissant des logiciels hébergés sur le cloud, accessibles via un navigateur web, ce qui élimine le besoin d'installation et de maintenance locale des applications. Cette approche a considérablement réduit les coûts et amélioré l'efficacité opérationnelle pour de nombreuses entreprises, en leur permettant de se concentrer sur leur cœur de métier plutôt que sur la gestion de l'infrastructure informatique .

Avec l'essor du SaaS et la popularité des paradigmes orientés services, le concept de multi-location a récemment suscité un grand intérêt dans la communauté des applications cloud. La multilocation dans le Cloud représente une voie de recherche prometteuse, qui suscite toujours beaucoup d'intérêt. Cet intérêt vient principalement de sa capacité à partager (selon des niveaux de maturité) des ressources matérielles/logicielles avec plusieurs locataires «consommateurs » du Cloud, c'est-à-dire à augmenter les profits des applications Cloud actuelles.

Défis, objectif et énoncé de mémoire:

Les consommateurs et les entreprises préfèrent payer le moins possible pour l'hébergement ou l'utilisation de logiciels. Avec des applications conçues et modifiées pour un locataire spécifique, puis hébergées en tant qu'instance individuelle, les coûts d'hébergement augmentent linéairement par locataire.

L'écart entre les paradigmes traditionnels hébergés et cloud a nécessité la création de solutions multi-locataires. Afin de servir de nombreux locataires d'applications, ces solutions multi-locataires construisent une instance d'application unique, permettant à chaque client d'améliorer, de personnaliser ou de modifier l'application comme s'il était le seul locataire.

Les solutions multi-locataires constituent une alternative puissante pour les entreprises cherchant à obtenir des avantages différentiels. C'est la raison pour laquelle l'étude présente exactement une telle entreprise comme étude de cas. Notre thèse s'inscrit dans ce contexte de système logiciel en tant que service multi-tenant, un contexte qui contient principalement deux voix conceptuelles, d'une part multi-bases de données et d'autre part mono-base de données. Cette étude se concentre sur la compréhension de ce qu'est la multilocation et pourquoi elle est pertinente pour la conception de solutions basées sur le cloud. Une étude de cas est utilisée pour fournir un contexte pertinent à la recherche de quelles conceptions sont dessinées et enfin une application simple implémentant la multi-location pour les exigences du cas spécifique est dérivée.

L'objectif final de ce projet est de concevoir une application de Réservation des rendez-vous en ligne sous forme de SaaS (Software as a Service), qui repose sur un modèle d'abonnement pour accéder à toutes ses fonctionnalités, La direction base de donnée unique a été retenue pour réaliser ce travail via le framework Laravel.

Structure Générale :

Le présent mémoire est organisé en trois chapitres. Il commence par une introduction générale décrivant le contexte, la problématique et les objectifs de notre travail.

- **Chapitre I :** Présente un aperçu des concepts de base du Cloud Computing, définitions ; modèles de déploiement ; modèles de service ; caractéristiques essentielles ; acteurs principaux ; obstacles et avantages du Cloud.
- **Chapitre II :** Dans ce chapitre, nous allons présenter une vue d'ensemble de l'architecture multi-tenant, en décrivant sa définition, ses avantages et ses inconvénients. De plus, nous allons examiner les différents niveaux de maturité SaaS en relation avec la multi location, mettant en évidence l'importance cruciale de la multi-location pour créer des applications SaaS.
- **Chapitre III :** Dans ce chapitre, nous allons fournir une description détaillée des fonctionnalités et du comportement de l'application proposée en utilisant le langage UML. De plus, nous allons présenter le processus de mise en œuvre qui comprend l'expression du langage de programmation utilisé, l'environnement de développement choisi, et enfin, nous allons illustrer les résultats obtenus grâce à notre application. Et pour conclure, nous allons fournir une visualisation des résultats obtenus grâce à notre application basée sur le modèle d'abonnement .

Le mémoire s'achève par une conclusion générale qui rappelle le travail réalisé et propose des perspectives futures .

Chapitre I : Généralités sur le Cloud Computing

1.1 Introduction

L'avancement rapide des technologies de l'information et de la communication a permis le développement de nouveaux paradigmes informatiques, où les techniques de traitement, de stockage, de communication, de partage et de diffusion de l'information ont radicalement changés. Les individus et les organisations sont de plus en plus recours à des serveurs externes pour le stockage et la diffusion efficace et fiable d'informations..

Le Cloud Computing, ou « informatique dans les nuages », est une nouvelle technologie informatique qui permet le déplacement des traitements et fichiers informatiques de l'ordinateur local vers des serveurs distants. Elle consiste à proposer les services informatiques sous forme de services à la demande, accessibles de n'importe où, n'importe quand et par n'importe qui grâce à une connexion internet.

Le but de ce chapitre est de se familiariser avec les notions de base du Cloud Computing (caractéristiques, types de déploiement, types de services, etc.) .

1.2 Historique

Techniquement, le concept de Cloud Computing est loin d'être nouveau, il est même présent depuis des décennies. On en trouve les premières traces dans les années 1960, quand John McCarthy¹ affirmait que cette puissance de traitement informatique serait accessible au public dans le futur. Le terme en lui-même est apparu plus couramment aux alentours de la fin du XXe siècle et il semblerait qu'Amazon.com soit l'un des premiers à avoir assemblé des Datacenter et fournit des accès à des clients.

Les entreprises comme IBM et Google ainsi que plusieurs universités ont seulement commencé à s'y intéresser sérieusement aux alentours de 2008, quand le Cloud Computing est devenu un concept à la mode. Réalisant ce qu'ils pourraient faire de toute cette puissance, de nombreuses compagnies ont ensuite commencé à montrer un certain intérêt, puis à échanger leurs anciennes infrastructures et applications internes contre ce que l'on appelle : les pay per-use service (services payés à l'utilisation) [1]. C'est la cinquième génération de l'informatique après les Mainframes, les PCs, les Clients/Serveurs et le Web [2].

¹ *John McCarthy* (né le 4 septembre 1927, à *Boston, Massachusetts*) est le principal pionnier de l'intelligence artificielle. Il est également l'inventeur en 1958 du langage Lisp. A la fin des années 1950, il a créé avec *Fernando Cobarto* la technique du temps partagé, qui permet à plusieurs utilisateurs d'em- ployer simultanément un même ordinateur.

1.3 Définition du Cloud Computing

Le Cloud Computing traduit de l'anglais « informatique dans les nuages » est devenu un mot à la mode populaire, littéralement l'informatique dans les nuages est un concept qui consiste à déporter sur des serveurs distants des stockages et des traitements informatiques traditionnellement localisés sur des serveurs locaux ou sur le poste de l'utilisateur. Il consiste à proposer des services informatiques sous forme de service à la demande, accessibles de n'importe où, n'importe quand et par n'importe qui, grâce à un système d'identification, via un PC et une connexion à Internet. Cette définition est loin d'être simple à comprendre, toute fois l'idée principale à retenir est que le Cloud n'est pas un ensemble de technologies, mais un modèle de fourniture, de gestion et de consommation de services et de ressources informatiques [3].

Pour NIST² [4], le Cloud Computing est un modèle permettant un accès réseau pratique et à la demande à un pool partagé de ressources informatiques fiables (par exemple : des réseaux, serveurs, stockage, applications, services) pouvant être rapidement provisionnés et libérés avec un minimum d'effort de gestion du consommateur ou d'interaction avec le fournisseur de services.

Selon Cisco³ : « Le Cloud Computing est une plateforme de mutualisation informatique fournissant aux entreprises des services à la demande avec l'illusion d'une infinité de ressources ».

Les différentes définitions affirment que le nuage est un ensemble de matériels, de raccordements réseau et de logiciels qui fournissent des services sophistiqués que les utilisateurs peuvent exploiter à volonté depuis n'importe où dans le monde.

De ce fait, le Cloud Computing est un basculement de tendance : au lieu d'obtenir de la puissance de calcul par acquisition de matériel et de logiciel, le consommateur se sert de puissance mise à sa disposition par un fournisseur via internet. Ainsi, le Cloud Computing est une nouvelle façon de délivrer les ressources informatiques, et non une nouvelle technologie, il peut également être vu comme un réseau informatique distribué, dont les ressources peuvent être approvisionnées de manière dynamique et reposant sur un contrat de service entre un fournisseur et un client.



Figure 1.1: Cloud Computing[5]

4.1 Eléments constitutifs du Cloud Computing

Un environnement Cloud Computing ne peut être défini sans un ensemble d'éléments qui sont :

1.4.1 La virtualisation

La virtualisation se définit comme l'ensemble des techniques matérielles et/ou logicielles qui permettent de faire fonctionner sur une seule machine, plusieurs systèmes d'exploitation, appelées machines virtuelles (VM), ou encore OS invitée [7]

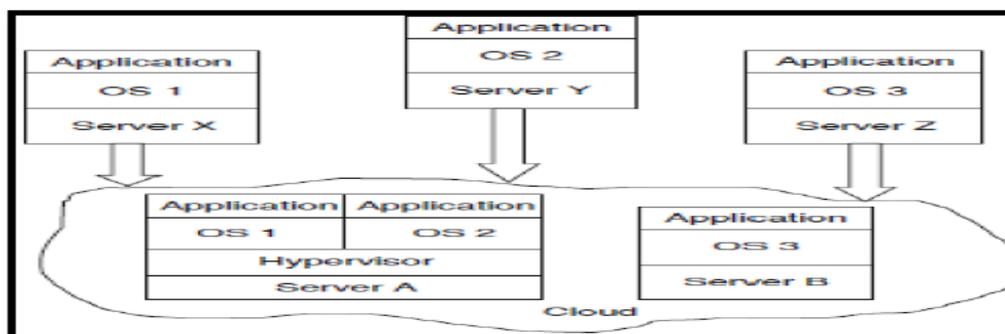


Figure 1.2: Exemple de virtualisation [6]

La virtualisation des serveurs (Figure 1.2) permet d'avoir une plus grande modularité dans la répartition des charges et la reconfiguration des serveurs en cas d'évolution ou de défaillance momentanée.

1.4.2 Centre de Calcul (Datacenter)



Figure 1.3: Datacenter [11].

Un centre de traitement de données (*data center en anglais*) est un site physique (Figure 1.3), sur lequel se trouvent regroupés des équipements constituant le système d'information de l'entreprise (*mainframes, serveurs, baies de stockage, équipements réseaux et de télécommunications, etc.*). Il peut être interne et/ou externe à l'entreprise, exploité ou non avec le soutien de prestataires, il

Chapitre I : Généralités sur le Cloud Computing

comprend en général un contrôle sur l'environnement (*climatisation, système de prévention contre l'incendie, etc.*), une alimentation d'urgence et redondante, ainsi qu'une sécurité physique élevée. Cette infrastructure peut être propre à une entreprise et utilisée par elle seule ou à des fins commerciales. Ainsi, des particuliers ou des entreprises peuvent venir y stocker leurs données suivant des modalités bien définies [7].

1.4.3 Plateforme collaborative

Une plateforme de travail collaborative est un espace de travail virtuel. C'est un site qui centralise tous les outils liés à la conduite d'un projet et les met à disposition des acteurs (Figure 1.4). L'objectif du travail collaboratif est de faciliter et d'optimiser la communication entre les individus dans le cadre du travail ou d'une tâche.

Les plateformes collaboratives intègrent généralement les éléments suivants : des outils informatiques ; des guides ou méthodes de travail en groupe pour améliorer la communication, la production et la coordination ; un service de messagerie ; un système de partage des ressources et des fichiers ; des outils de type forum ou pages de discussions ; un trombinoscope ou annuaire des profils des utilisateurs, des groupes, par projet ou par thématique et un calendrier [7].



Figure 1.4: Plateforme collaborative [7].

1.5 Caractéristiques du Cloud Computing

Généralement, un service, une solution ou un environnement d'exécution de vrait satisfaire une liste de caractéristiques pour qu'il soit considéré comme étant du Cloud Computing. Parmi ces caractéristiques, il y a celles qui sont reconnues comme fondamentales. Par exemple, le NIST [4] définit cinq caractéristiques essentielles qui sont :

1- Un service à la demande : Un utilisateur peut allouer des ressources informatiques (serveurs, réseau, stockage, environnement d'exécution, application) au besoin, de façon automatique et sans nécessité d'interaction humaine avec chaque fournisseur de service.

2- Un accès aux ressources par le réseau : Les fonctionnalités sont disponibles sur le réseau et accessibles via des mécanismes standards qui favorisent leurs utilisations à partir des appareils clients hétérogènes (*par exemple : téléphones mobiles, tablettes, ordinateurs portables et stations de travail*).

3- Mise en commun des ressources : Les ressources informatiques du fournisseur sont regroupées pour servir plusieurs consommateurs à l'aide d'un modèle multi locataire. Différentes ressources physiques et virtuelles sont allouées et libérées dynamiquement selon la demande des consommateurs. Généralement, l'utilisateur n'a ni le contrôle ni la connaissance de l'emplacement exact des ressources allouées. Dans certains cas, il peut choisir l'emplacement géographique à un niveau haut (par exemple : par pays, continent ou datacenter). Les exemples de ressources incluent le stockage, le traitement, la mémoire et la bande passante du réseau.

4- Flexibilité des ressources : Les ressources sont allouées et libérées d'une façon élastique, dans certains cas automatiquement pour s'adapter rapidement à la demande qu'elle soit croissante ou décroissante. Pour le consommateur, les fonctionnalités disponibles pour l'allocation apparaissent comme illimitées et peuvent être adaptées à n'importe quelle quantité et à tout moment.

5- Un service mesuré : Les systèmes en nuage contrôlent et optimisent automatiquement l'utilisation des ressources afin de mesurer leurs consommations avec un niveau d'abstraction approprié selon le type du service (par exemple : stockage, temps de calcul et bande passante). L'utilisation des ressources peut être surveillée offrant ainsi une transparence à la fois au fournisseur et au consommateur du service utilisé.

1.6 Modèles de déploiement du Cloud Computing

Quel que soit le modèle de service utilisé (SaaS, PaaS ou IaaS), le NIST [4] a défini quatre modèles de déploiement pour les services de Cloud, avec des variantes qui répondent à des besoins spécifiques.

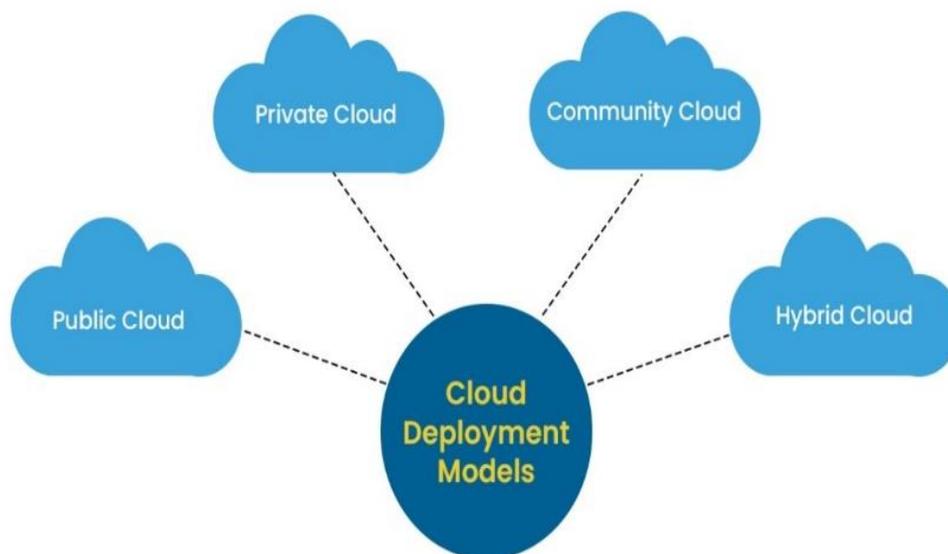


Figure 1.5: Les Modèles de déploiement du Cloud Computing[8]

Cloud public

Un Cloud public est basé sur un modèle standard de Cloud Computing, dans lequel un prestataire de service met les ressources, telles que les applications ou le stockage, à la disposition du grand public via internet (Figure 6).

Le Cloud public peut être gratuit ou fonctionner selon un paiement à la consommation. L'avantage de ce genre d'architecture est d'être facile à mettre en place, pour des coûts relativement raisonnables. La charge du matériel, des applicatifs, de la bande passante étant couverte par le fournisseur, ce modèle permet de proposer une souplesse et une évolutivité accrue afin de répondre rapidement au besoin. Il n'y a pas de gaspillage de ressources car le client ne paye que ce qu'il consomme [9].

Bien que le Cloud public offre un maximum de flexibilité, il demande de lourds investissements pour le fournisseur de services. De plus, ce type de Cloud n'est pas sécurisé.

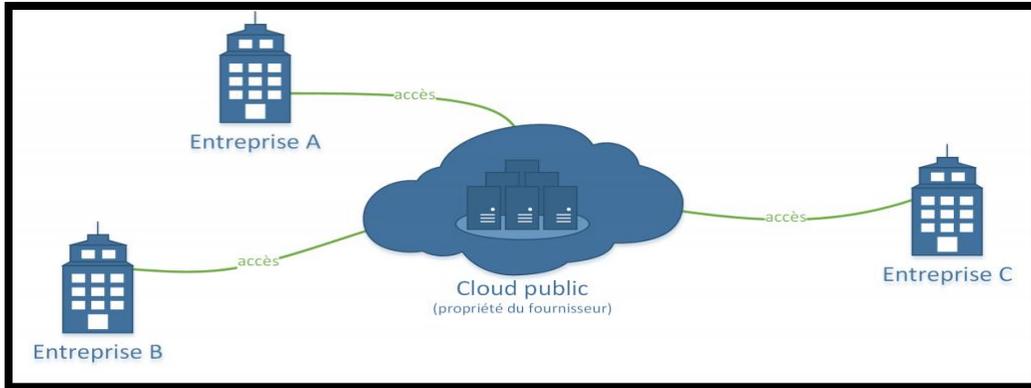


Figure 1.6 : Cloud public [10].

Cloud privé

L'infrastructure en nuage est configurée pour une utilisation exclusive par une seule organisation comprenant plusieurs consommateurs (*par exemple, des unités commerciales*) (Figure 7). Il peut être possédé, géré et exploité par l'organisation, un tiers, ou une combinaison de ceux-ci, et il peut exister sur site ou hors site [4].

L'avantage majeur du Cloud privée est qu'il est dédié et sécurisé. Cependant, il est moins flexible comparé au Cloud public et plus cher.

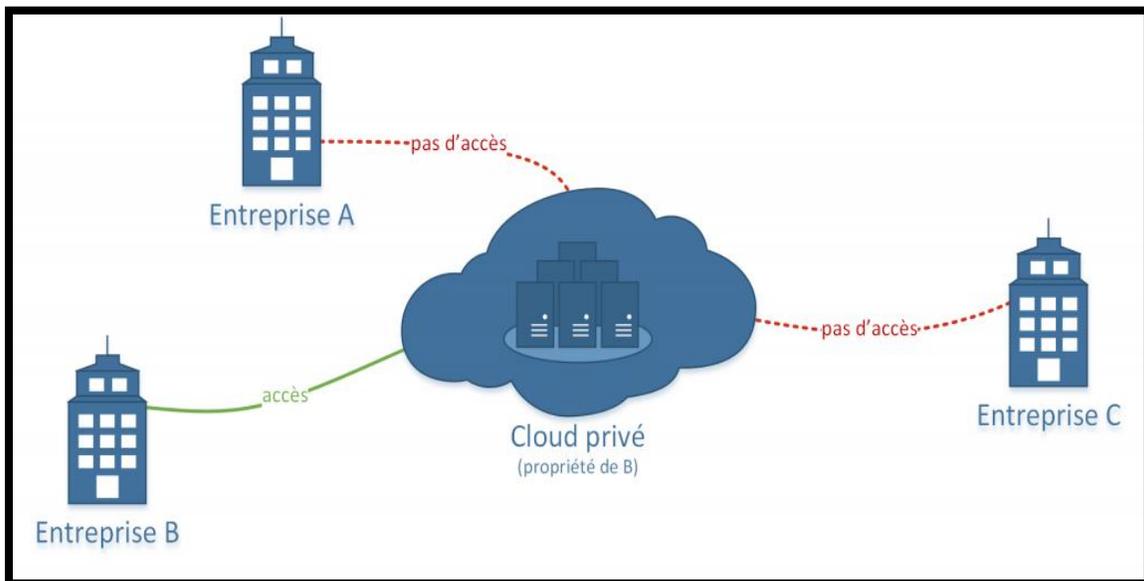


Figure 1.7: Cloud Privé [10].

Cloud hybride

L'infrastructure de Cloud dans un Cloud hybride est une composition d'au moins de deux Clouds (*privé, communautaire ou public*) (Figure 8). Chacune d'entre elles reste une entité unique mais est reliée par une technologie standard ou exclusive. Cette technologie permet la portabilité des données et des applications. Les avantages des nuages hybrides sont l'évolutivité, la flexibilité et la rentabilité. Les inconvénients sont les problèmes de réseau et les règles de sécurité [11].

Avec un Cloud hybride, nous pouvons déporter les applications vers un Cloud public qui consommera des données stockées et exposées dans un Cloud privé, ou bien faire communiquer deux applications hébergées dans deux Clouds privés distincts, ou encore consommer plusieurs services hébergés dans des Clouds publics différents.

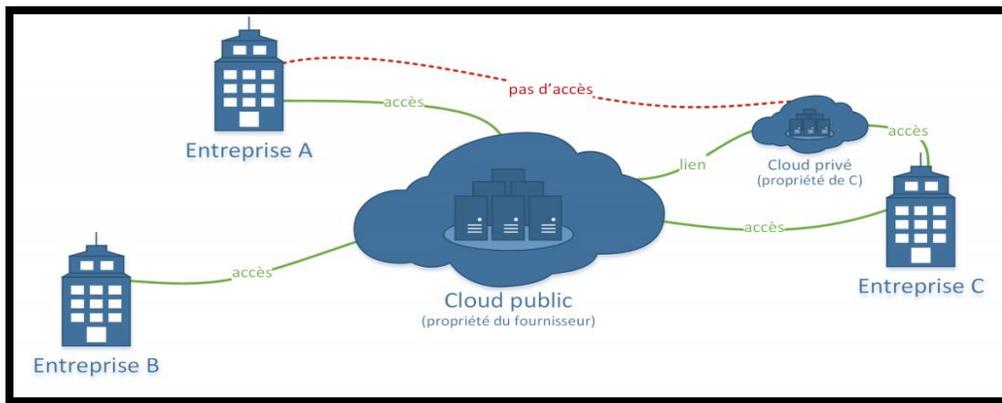


Figure 1.8: Exemple de Cloud hybride [10].

Cloud communautaire

L'infrastructure de Cloud Computing est configurée pour une utilisation exclusive par une communauté de consommateurs ou des organisations qui ont des préoccupations communes (par exemple : la mission et les exigences en matière de sécurité). Il peut être administré, géré et exploité par un ou plusieurs des organismes de la communauté, un tiers, ou une combinaison des deux, et il peut exister sur ou à l'extérieur des locaux [12].

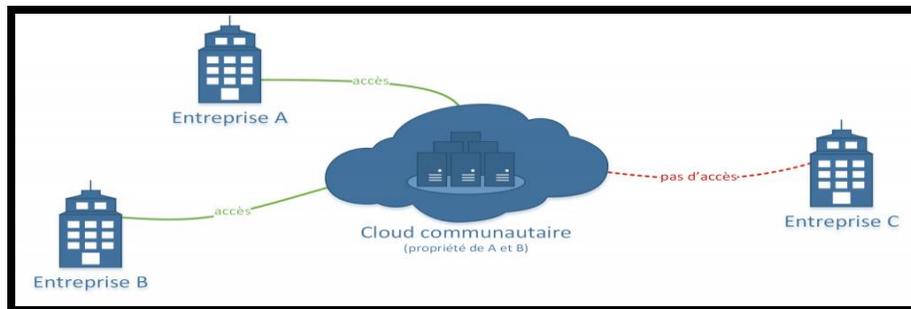


Figure 1.9: Exemple de Cloud hybride [10].

1.7 Types de services dans les environnements de Cloud Computing

La partie visible du Cloud Computing se décompose en 3 parties essentielles qui sont (Figure 4) :

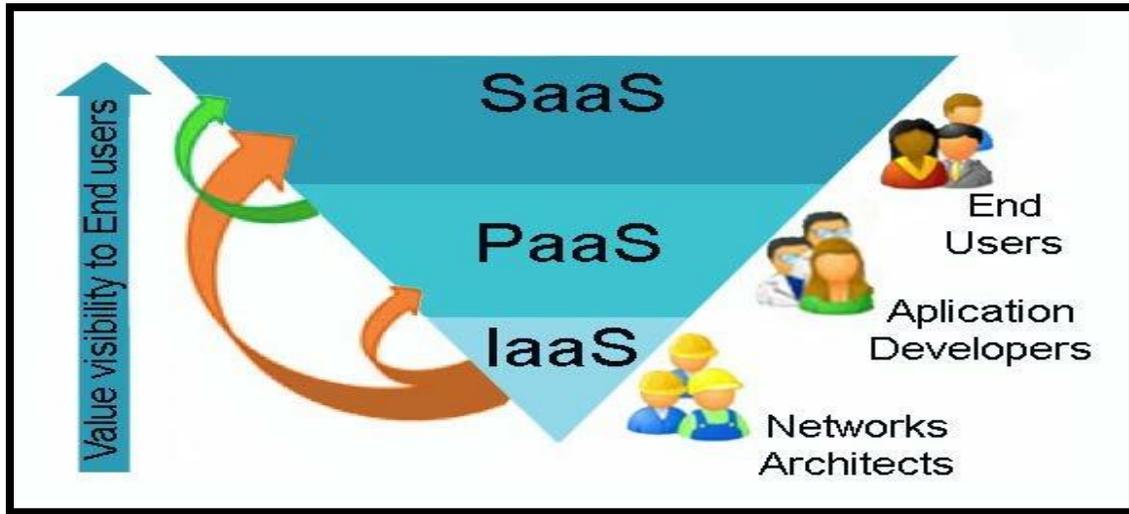


Figure 1.10: Types de services Clouds [13].

SaaS (Software as a Service)

Concept consistant à proposer un abonnement à un logiciel plutôt que l'achat d'une licence. On oublie donc le modèle client serveur et aucune application n'est installée sur l'ordinateur, elles sont directement utilisables via le navigateur Web. L'utilisation reste transparente pour les utilisateurs, qui ne se soucient ni de la plateforme, ni du matériel, qui sont mutualisés avec d'autres entreprises. Le SaaS remplace l'ASP (Application Service Provider), aussi appelé fournisseur d'applications hébergées (FAH), qui est une entreprise qui fournit des logiciels ou des services informatiques à ses clients à travers un réseau.

Il s'agit de la mise à disposition d'applications sous forme de service (*CRM*⁴, *outils collaboratifs*, *messagerie*, *ERP*⁵, ...). Ce concept consiste à proposer un abonnement à un logiciel plutôt que l'achat d'une licence.

De ce fait, pas d'installation, pas de mises à jour à gérer (elles sont gérées par le fournisseur) et le paiement est à l'usage [14].

⁴ La gestion de la relation client (GRC), ou gestion des relations avec les clients, en anglais Customer Relationship Management (CRM), est l'ensemble des outils et techniques destinés à capter, traiter, analyser les informations relatives aux clients, dans le but de les fidéliser en leur offrant le meilleur service. ⁵ Enterprise Resource Planning, signifiant littéralement en anglais, « planification des ressources de l'entreprise », et traduit en français par « progiciel de gestion intégré » (PGI).

Chapitre I : Généralités sur le Cloud Computing

PaaS (Platform as a Service)

Il s'agit des plateformes du nuage, regroupant principalement les serveurs mutualisés et leurs systèmes d'exploitation. En plus de pouvoir délivrer des logiciels en mode SaaS, le PaaS dispose d'environnements spécialisés au développement comprenant les langages, les outils et les modules nécessaires[14].

IaaS (Infrastructure as a Service)

Il s'agit de la mise à disposition, à la demande, de ressources d'infrastructures (*serveurs, moyens de stockage, réseau ...*) dont la plus grande partie est localisée à distance dans des Datacenter. L'IaaS permet l'accès aux serveurs et à leurs configurations pour les administrateurs de l'entreprise. Le client a la possibilité de louer des clusters, de la mémoire ou du stockage de données. Le coût est directement lié au taux d'occupation. Une analogie peut être faite avec le mode d'utilisation des industries des commodités (*électricité, eau, gaz*) ou des télécommunications [14].

Le tableau 1 suivant donne des exemples d'offres de services Clouds existants sur le marché des leaders mondiaux.

IaaS	PaaS	SaaS
-Amazon offres EC2 et AWS	-Microsoft offre Azur	- Google offre Google Apps (messagerie et bureautique)
-Microsoft offre Azur	- Google offre Google App Engine	- Sales Force CRM (Customer Relationship Management)

Tableau 1.1: Les grands acteurs mondiaux du Cloud Computing [15].

1.8 Acteurs du Cloud Computing

Des acteurs importants jouent un rôle dans l'architecture de référence du Cloud Computing de NIST, notamment en tant que consommateur, courtier (broker) et fournisseur de Cloud. Chaque acteur est considéré comme une entité (une personne ou une organisation) pouvant participer à un processus ou à une transaction et / ou effectuer des tâches dans le Cloud Computing. Dans ce qui suit, nous allons donner plus de détails sur ces acteurs en reprenant les explications de [16].

Consommateur Cloud ‘Cloud Consumer’

Le consommateur de services Cloud est le principal acteur dans le Cloud Computing. Un consommateur représente une personne ou une organisation qui entretient une relation commerciale avec un fournisseur de nuage en utilisant ses services. Il parcourt le catalogue de services à partir d'un fournisseur de nuage, demande le service approprié, établit des contrats de service avec le fournisseur de nuage et utilise le service. Le client en nuage peut être facturé pour le service fourni et doit assurer les paiements en conséquence. [17].

Fournisseur Cloud ‘Cloud Provider’

Un fournisseur de Cloud est une personne ou une organisation, c'est l'entité chargée de mettre un service à la disposition des parties intéressées. Un fournisseur de Cloud acquiert et gère l'infrastructure informatique requise pour fournir les services, exécute le logiciel Cloud qui fournit les services et effectue des arrangements pour fournir les services Cloud aux consommateurs Cloud via l'accès réseau. [17].

Il existe actuellement une multitude de fournisseurs Clouds qui délivrent des services de différents types (IaaS, PaaS et SaaS) et pour des domaines distincts. Parmi les fournisseurs les plus actifs sur le marché nous citons :

- Amazon (www.amazon.com)
- Google
- Microsoft
- IBM
- Zoho (www.zoho.com)
- Rackspace (www.rackspace.com)
- Citrix (www.citrix.com)
- Intel
- Sales force (www.salesforce.com)

Courtier en nuage ‘Cloud Broker’

Un courtier en nuage est utilisé comme technique de médiation pour permettre une interaction entre différents systèmes en nuage qui ne sont pas équipés pour interagir les uns avec les autres. Cette technique peut réduire la nécessité de modifier les systèmes en nuage actuels [18].

L’institut national de normalisation et de technologie (NIST) définit un courtier en nuage comme « une entité qui gère l’utilisation, les performances et la fourniture de services en nuage et négocie les relations entre les fournisseurs de nuage et les consommateurs de nuage » [19].

1.9 Relations entre les acteurs du Cloud Computing

Les relations qui existent entre les acteurs du Cloud Computing sont illustrées dans la figure 11. Un consommateur de Cloud peut demander des services à un fournisseur de Cloud de deux manières. Soit une demande directe auprès d'un fournisseur de Cloud ou bien la négociation avec un courtier en Cloud. Un nouveau service pourrait être offert lorsqu'un courtier en Cloud est demandé. Un courtier peut offrir de nouveaux services en intégrant différents services ou en améliorant les services disponibles. Comme illustré à la figure 11, un consommateur de Cloud ne connaît pas le fournisseur de Cloud en question et doit contacter directement le courtier en Cloud [16].

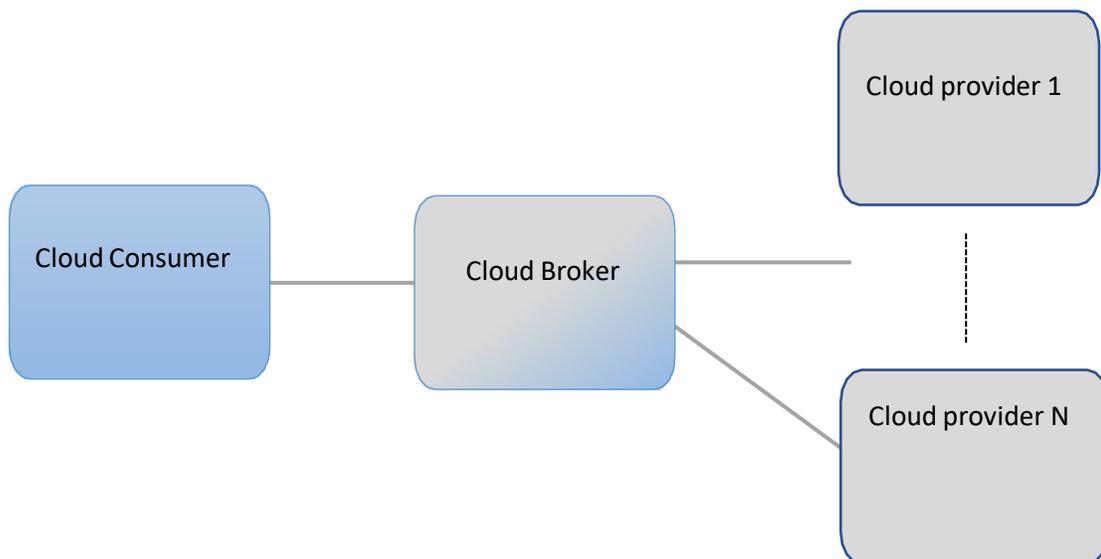


Figure 1.11 : Relations entre les acteurs du Cloud Computing [16].

1.10 Avantages et inconvénients du Cloud Computing

Le marché des services Clouds connaît une croissance fulgurante ces dernières années, prouvant ainsi qu'il ne s'agit pas juste d'un boom médiatique mais une vraie technologie dans le domaine des applications distribuées et aussi pour la délivrance de services. Cela est dû principalement à sa politique d'utilisation qui offre de nombreux avantages. Toutefois, l'adoption du Cloud Computing se heurte à plusieurs obstacles. Les avantages et les inconvénients du Cloud Computing diffèrent selon le type du service délivré. Le tableau 3 donne les principaux avantages et inconvénients selon le type du service Cloud.

	Avantage	inconvénient
SaaS	-pas d'installation -plus de licence -migration	-logiciel limité -sécurité -dépendance des prestataire
Paas	-pas d'infrastructure nécessaire -pas d'installation -environnement hétérogène	-limitation des langages -pas de personnalisation dans la configuration des machines virtuelles
IaaS	-administration -personnalisation -flexibilité d'utilisation	-sécurité -besoin d'un administrateur système

Tableau 1.2 : Avantages et inconvénients des services Cloud [3]

Les obstacles qui s'opposent à l'adoption du Cloud Computing (Figure 12) constituent actuellement de vraies pistes de recherche qui suscitent l'intérêt des chercheurs et des industriels. Parmi les problèmes majeurs du Cloud Computing, nous citons :

- **Interopérabilité** : L'un des principaux obstacles à l'adoption du Cloud Computing est l'interopérabilité. Il s'agit de la capacité des ressources d'un fournisseur de Cloud à communiquer avec des ressources d'un autre fournisseur de Cloud.

Chapitre I : Généralités sur le Cloud Computing

- **Latence** : Un réseau (ou internet si des Clouds publics doivent être fournis) rend les ressources Cloud accessibles. Cependant, il crée une latence (temps de retard) dans n'importe quel type de connexion entre les utilisateurs et l'environnement.
- **Sécurité** : La confidentialité des données inquiète les utilisateurs du Cloud. La plupart du temps, les responsables de l'organisation qui utilise les services ne sont pas parfaitement au courant des emplacements des fournisseurs de Cloud Computing qui sauvegardent leurs données [16].

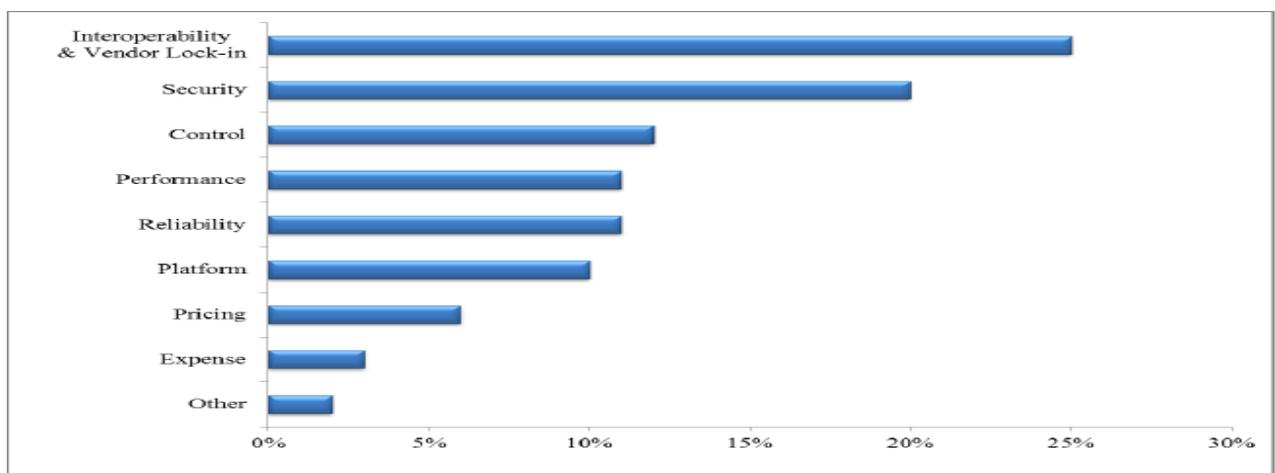


Figure 1.12 : Obstacles à l'adoption du Cloud Computing [16].

Conclusion

Le Cloud Computing est considéré comme la nouvelle révolution de l'informatique à travers l'internet. Il offre des avantages pour les organisations et les individus. Dans ce chapitre, nous avons passé en revue les notions de base relatives à ce concept.

Dans le prochain chapitre, nous concentrons sur le modèle de service SaaS, en mettant particulièrement l'accent sur l'une de ses caractéristiques : la multi location. Nous aborderons également les frameworks qui supportent cette caractéristique, ainsi que quelques articles traitant de ce sujet .

Chapitre II :La Multi-Tenant dans les modèles de service "SaaS":

2.1 Introduction :

L'architecture multi-tenant dans le domaine du cloud computing représente une approche fondamentale pour la fourniture efficace de services et de ressources informatiques partagés à travers le cloud. Cette approche permet à plusieurs utilisateurs ou locataires de partager une seule instance d'application ou de ressource .

Ce chapitre se concentre sur la présentation des concepts, des idées et de l'approche de l'architecture multi-tenant, ainsi que sur l'explication des principales solutions qui abordent cette problématique dynamique qui a récemment suscité beaucoup d'engouement .

2.2 Définition Multi-tenant :

la multilocation est définie comme « un mode de fonctionnement dans lequel plusieurs locataires partagent la même instance d'une application logicielle ou d'un service basé sur le cloud, chaque locataire étant isolé des autres et capable de personnaliser une partie de l'application ou du service[20]

la multilocation comme caractéristique clé du cloud computing, déclarant que « la multilocation permet de partager les mêmes services ou ressources avec plusieurs locataires, tout en séparant et en isolant les données, le traitement et la configuration de chaque locataire »[21] .

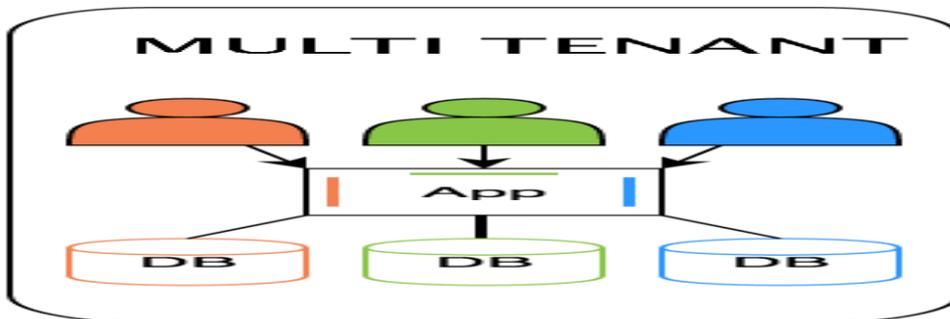


Figure 2.1: Illustration de l'approche multi-locataire[22]

Définition Single-tenant :

un mode de fonctionnement dans lequel une seule instance d'une application logicielle ou d'un service basé sur le cloud est attribuée à un seul locataire, et ce locataire a la possibilité de personnaliser l'application ou le service dans une certaine mesure[20] .

la location unique comme « un modèle architectural dans lequel le fournisseur de services fournit des ressources dédiées (serveurs, stockage, réseau, etc.) pour chaque client individuel (locataire)[23].

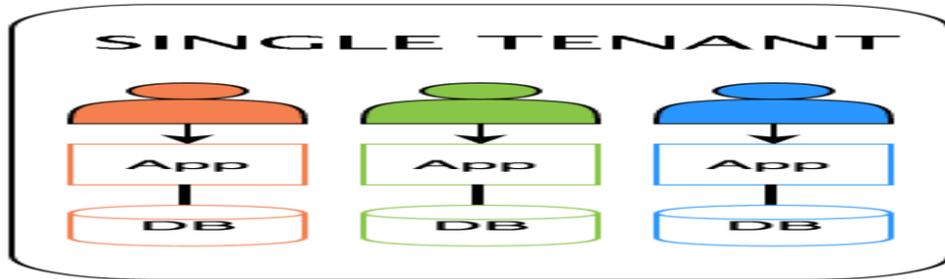


Figure 2.2: Illustration de l'approche Single-tenant[22].

2.3 - Comparaison entre Single-tenant et Multi-tenant :

	Single-tenant	Multi-tenant
Disponibilité	Les ressources informatiques sont dédiées	Les ressources informatiques sont partagées
Coût	Un seul utilisateur responsable de tous les coûts d'exploitation	Plusieurs locataires se partagent les coûts d'exploitation
Efficacité	Les ressources sont fixes, de sorte que la capacité inutilisée est gaspillée	Les ressources sont disponibles de manière dynamique, de sorte que la capacité est disponible en fonction des besoins
Type de cloud	Cloud privé	Cloud public
Mises à jour	Chaque instance est mise à jour séparément	Effectuer des mises à jour d'une instance unique et partagée
Responsabilités	Les équipes internes sont responsables de la maintenance, des mises à jour logicielles, des correctifs,	Le fournisseur gère toutes les mises à jour logicielles de maintenance, les correctifs, la sauvegarde, la restauration et la reprise après sinistre

Tableau 2.1: Comparaison entre Single-tenant et Multi-tenant[24]

2.4- L'approche orienté base de données;

Dans cette tendance, on distingue deux grandes approches ,Ceux-ci sont illustrés à la figure 2.3

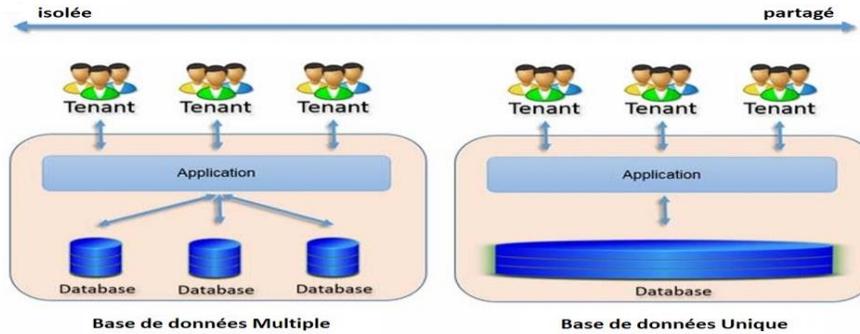


Figure 2.3: Stratégie de stockage des données[25]

Base de données Multiple;

Cette méthode implique que chaque locataire possède sa propre base de données pour le stockage de ses données, mais partage les ressources de calcul et le code d'application avec d'autre locataires.[26]

Base de données Unique :

locataires Utilisez un logiciel commun. De plus, ce modèle est divisé en deux modèle. Il s'agit de Shared Database-Separate Schema et de Shared Schema Schéma partagé par la base de données[26]. Ceux-ci sont illustrés à la figure 2.4

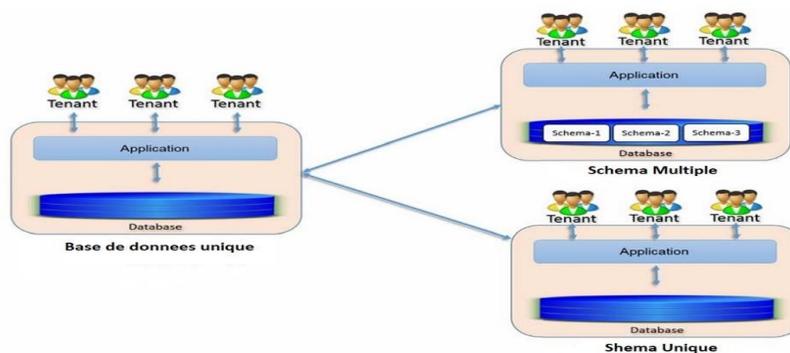


Figure 2.4:L'approche orienté base de données unique[25]

Base de données unique, schéma multiple :

dans cette approche, chaque locataire dispose de ses propres tables de base de données distinctes sur le réseau Ma base de données. L'exemple est illustré à la figure 3. Au fur et à mesure que des schémas distincts sont créés pour chaque locataire, la mise en œuvre est en partie plus facile et peut être facilement façonné pour répondre aux besoins d'un utilisateur. L'inconvénient est qu'un grand nombre de tables doivent être conservées [26].

L'approche suivante consiste à stocker les données de plusieurs locataires dans la même base de données, mais séparément plan. Une fois qu'un locataire s'inscrit pour la première fois pour utiliser une application spécifique, un Ensemble de tables de base de données associées à ce client. Il existe également des options pour restreindre l'accès ici Seuls les utilisateurs appartenant à des clients autorisés peuvent accéder à ces tables. Cette méthode fournit Fournit un niveau modéré d'isolation logique des données pour les locataires et les supports soucieux de la sécurité Un plus grand nombre de clients par serveur de base de données.

Le plus gros inconvénient de cette approche Les données du locataire sont plus difficiles à sauvegarder et à restaurer, car les sauvegardes ne peuvent pas être restaurées l'ensemble de la base de données. Cela écrasera les données des autres locataires. Dans l'ensemble, cette approche Cela se traduit par des coûts d'exploitation inférieurs dans la première méthode[27].

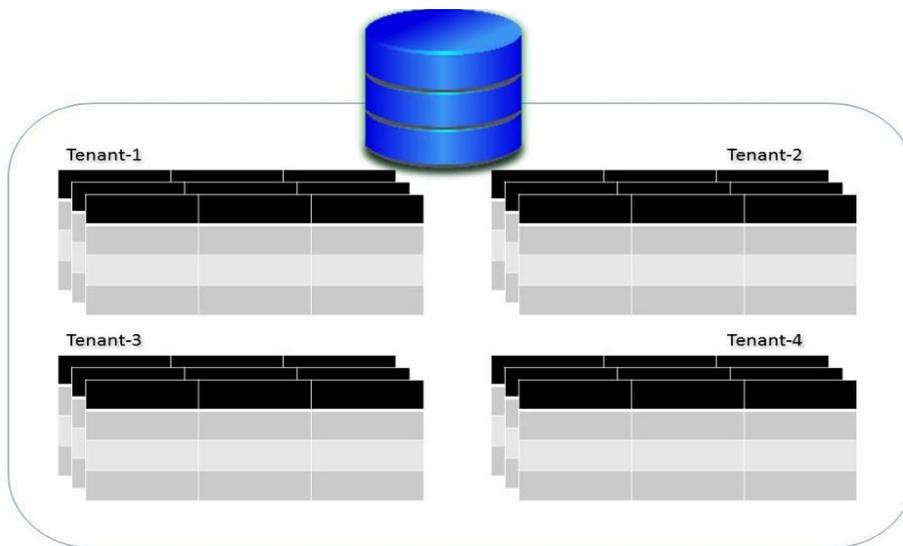
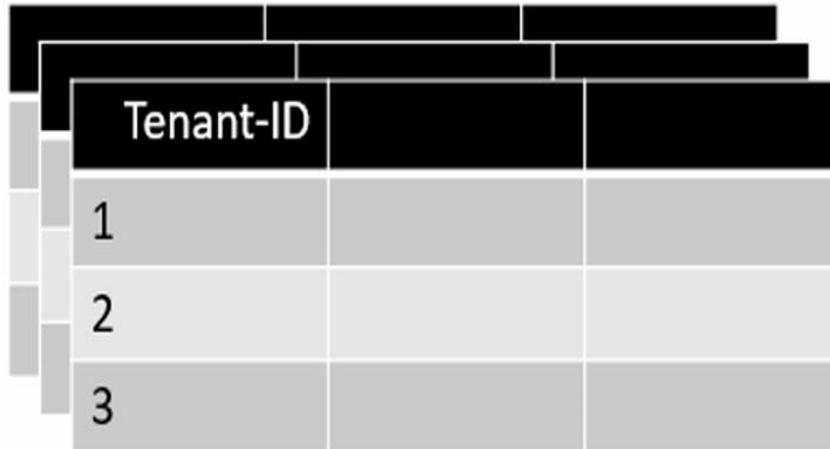


Figure 2.5:Base de données unique, schéma multiple[25]

Base de données unique, schéma unique :

dans cette approche, il y a une base de données commune et un schéma commun pour tous les locataires. Ce modèle offre le moins d'utilisation et d'entretien en termes d'utilisation la plus efficace du matériel. Le L'inconvénient est que tous les locataires sont dans le même schéma.

Le troisième modèle illustre une véritable mutualisation, où les utilisateurs partagent entièrement l'application et Base de données (efficacité maximale). Ce modèle nécessite un processus plus complexe, en particulier pour les Modifiez le schéma de la base de données en ajoutant des identificateurs de locataire à chaque table et vue Réécrivez tous les accès SQL pour ajouter des filtres pour les locataires. C'est très populaire l'approche multi-locataire pure. Avec cette approche, les données de chaque locataire sont stockées Au sein de la même base de données et du même schéma, la séparation des locataires s'effectue via l'allocation **Tenant_id** représente une colonne d'une table appartenant au locataire[27][28]



Tenant-ID		
1		
2		
3		

Figure 2.6: Base de données unique, schéma unique[25]

Chapitre II :La Multi-Tenant dans les modèles de service "SaaS"

2.5 les avantages des l'architecture multi-tenant :

la multi-location présente plusieurs avantages pour les fournisseurs de logiciels et les clients[29] :

1-Pour les fournisseurs : la multi-location permet une utilisation efficace des ressources, une maintenance de logiciels simplifiée et des mises à niveau et des mises à jour rationalisées .

2-Pour les clients : la multi-location offre des solutions rentables, une évolutivité facile et une réduction des frais généraux

les inconvénient des l'architecture multi-tenant :

-les termes de sécurité des données et de personnalisation .

-Étant donné que plusieurs locataires partagent la même instance de l'application .

- La personnalisation, comme l'ajout de nouvelles fonctionnalités ou la modification de celles existantes, peut également être difficile[29]

2.6 Les Framework qui supporte la multi-tenant :

Dans cette partie, nous allons citer quelques cadres utilisés pour implémenter la location multiple.

Laravel :

Laravel est un Framework PHP open source conçu pour le développement d'applications web. Il offre la possibilité de supporter l'abonnement grâce aux packages Paddle, Stripe et PayPal[30].

Express :

Express est un Framework léger pour les applications web basées sur Node.js . Il offre la possibilité de supporter l'abonnement grâce aux packages Stripe , Braintree et PayPal[31].

Spring :

Spring est un cadre de développement Java conçu pour les applications d'entreprise .Il offre la possibilité de supporter l'abonnement grâce aux packages SpringPayflow Pro et Spring Commerce Tools[32] .

Django:

Django est un Framework web open-source en Python qui permet aux développeurs de créer rapidement des applications web complètes Il offre la possibilité de supporter l'abonnement grâce aux packages Django payments , Djstripe et Django-oscar[33].

2.7 Migration vers le multi-tenant SaaS (travaux connexes) :

Des travaux antérieurs ont examiné un modèle pour orienter la transition des applications Web classiques vers le modèle SaaS multi-tenant. Dans cette thèse, nous nous appuyons sur deux publications influentes dans le domaine du Cloud SaaS, largement citées.

Saleh et.al. [34] a proposé un cadre de migration en incorporant l'élément de Personnalisation de l'interface utilisateur et du flux de travail, de la logique métier et de la configuration de la base de données.

Bezemer et.al. [35] Effectuer des modèles de migration d'applications qui prennent en compte le matériel partagé, haut degré de configurabilité et instance de base de données partagée. Il y a trois composants impliqués dans ces cadres :

1. Module d'authentification pour mapper les informations d'identification des utilisateurs finaux aux locataires,
2. Module de configuration pour gérer les paramètres du locataire.
3. Module de base de données pour adapter, insérer, modifier et interroger les données orientées client.

Conclusion

Ce chapitre analyse en profondeur l'architecture multi-tenant, mettant en lumière les diverses solutions et approches de cette vision architecturale prometteuse, ainsi que les différents niveaux de maturité qui y sont associés. Il souligne également l'importance capitale de la multi-tenant dans la création de systèmes de qualité. Le chapitre suivant vise à maximiser les efforts pour concevoir et mettre en œuvre un système multi-tenant avec un base de données unique basé sur l'abonnement, en utilisant une approche orientée vers une seule base de données pour gérer efficacement les données des utilisateurs et leur fournir une expérience utilisateur homogène.

Chapitre III : Conception et réalisation:

3.1 Introduction :

De nos jours, l'intégration du modèle de service SaaS et de l'architecture multi-tenant dans les applications modernes pose des défis en termes de conception et de programmation, ce qui rend leur réalisation et leur mise en œuvre complexes .

Pour remédier à ce problème, de nombreux frameworks et leurs bibliothèques associées proposent une approche plus rationalisée de la mise en œuvre du modèle de service SaaS, avec des fonctionnalités d'abonnement et de multi-location.

Dans ce chapitre, nous examinerons en détail l'approche adoptée pour rendre notre système 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 notre système final appelé Helthy Track .

3.2 Description du système :

La prise de rendez-vous en ligne avec les médecins est devenue incroyablement pratique et efficace, surtout dans le monde trépidant d'aujourd'hui. Grâce aux plateformes et applications en ligne, vous pouvez prendre rendez-vous avec des professionnels de la santé sans avoir à passer d'appels téléphoniques ou à attendre en attente, L'objectif de ce mémoire est de créer une application de reservation des rendez-vous en ligne Multi-tenant de type service web avec un système d'abonnement appelé "Health Track" , Ce système offrira aux locataires, à savoir les médecins, la possibilité de souscrire à un abonnement payant. Chaque locataire disposera ainsi d'un compte personnel destiné à ses clients, les patients, leur permettant de prendre des rendez-vous en ligne, et ajouter ou modifie les rendez-vous, ainsi que de consulter la liste des patients. De leur côté, les patients auront la possibilité de créer un compte pour identifier les médecins disponibles et consulter leurs propres rendez-vous.

En somme le système de Health track offre plusieurs avantages, parmi eux: Les patients peuvent prendre rendez-vous depuis n'importe où, à tout moment, sans se déplacer physiquement jusqu'au cabinet médical ,Évitez les longues attentes au téléphone en réservant facilement votre rendez-vous en ligne, à toute heure du jour ou de la nuit. Gardez une trace de tous vos rendez-vous passés et futurs, ce qui facilite la gestion de votre santé et de vos suivis médicaux.

Dans les sections suivantes, nous décrirons la méthodologie adoptée, ainsi que la conception du modèle finalisé et la mise en œuvre du système cible appelé Health Track, ainsi que ses différentes interfaces .

3.3 La Modélisation du Système :

La modélisation de notre système repose principalement sur l'approche UML en particulier sur le diagramme de classe et diagramme de cas d'utilisation et le diagramme de sequence . Ce diagramme exprime de manière précise les différents besoins fonctionnels de notre système d'abonnement multi-locataire. L'UML (Unified Modeling Language) est une méthode de

Chapitre III : Conception et réalisation

modélisation visuelle utilisée pour concevoir et développer des programmes orientés objet. Grâce à l'UML, il est possible de modéliser une grande variété d'applications logicielles qui peuvent fonctionner sur différentes plateformes, systèmes d'exploitation et réseaux, et qui peuvent être écrites dans divers langages de programmation.

Diagramme de cas d'utilisation :

Les acteurs de Health-Track sont :

1-Administrateur : gère les utilisateurs, vérifie les profils des médecins, gère les rendez-vous et gère les plans de paiement. **2-Médecin** : interagit avec le système de création et de gestion de profils, de vérification des profils, de gestion des rendez-vous, de conduite de consultations et de participation aux plans de paiement. **3-Patient** : interagit avec le système pour l'inscription, la gestion des profils, la recherche de médecins, la prise de rendez-vous.

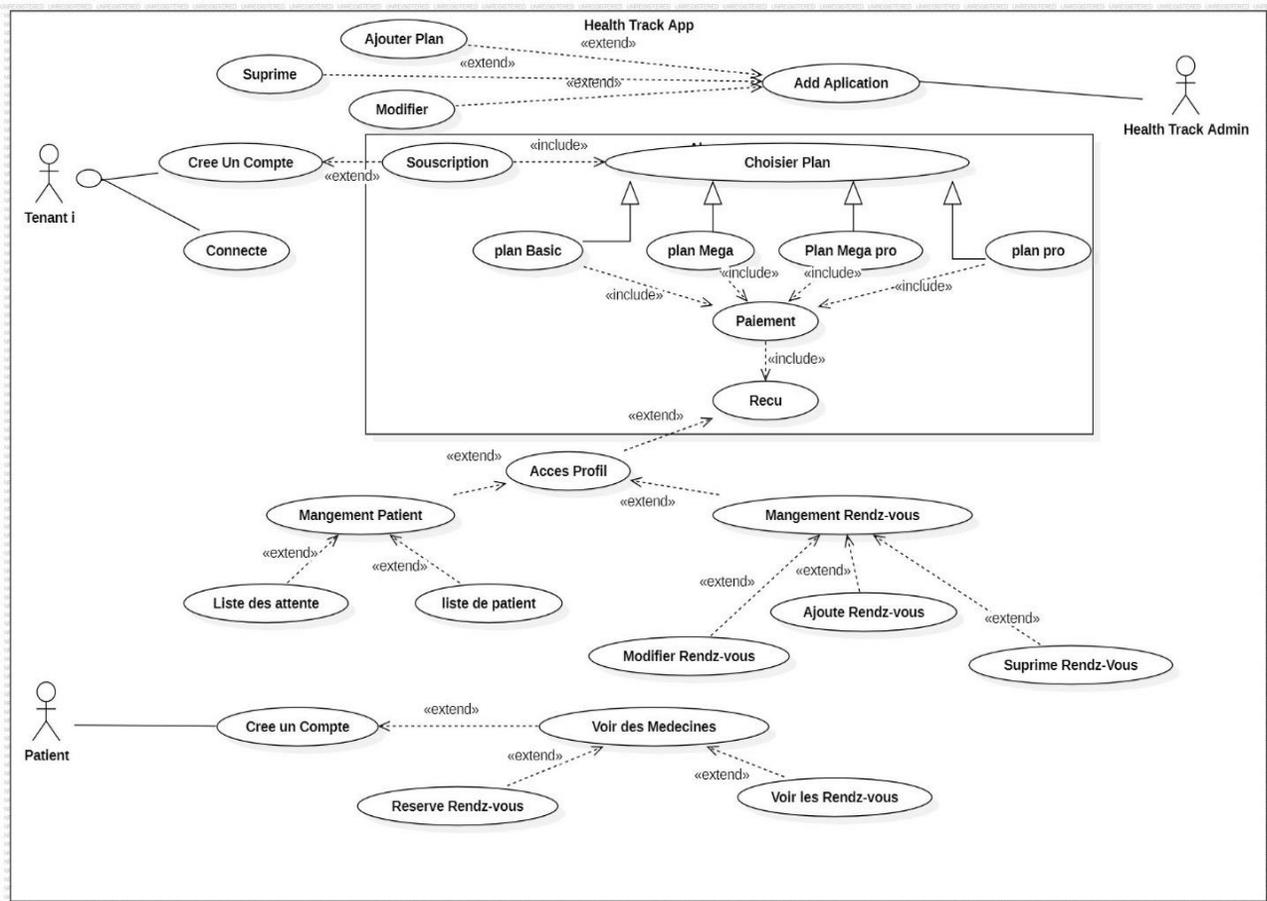


Figure 3.1: Diagramme de cas d'utilisation

Diagramme de Classe:

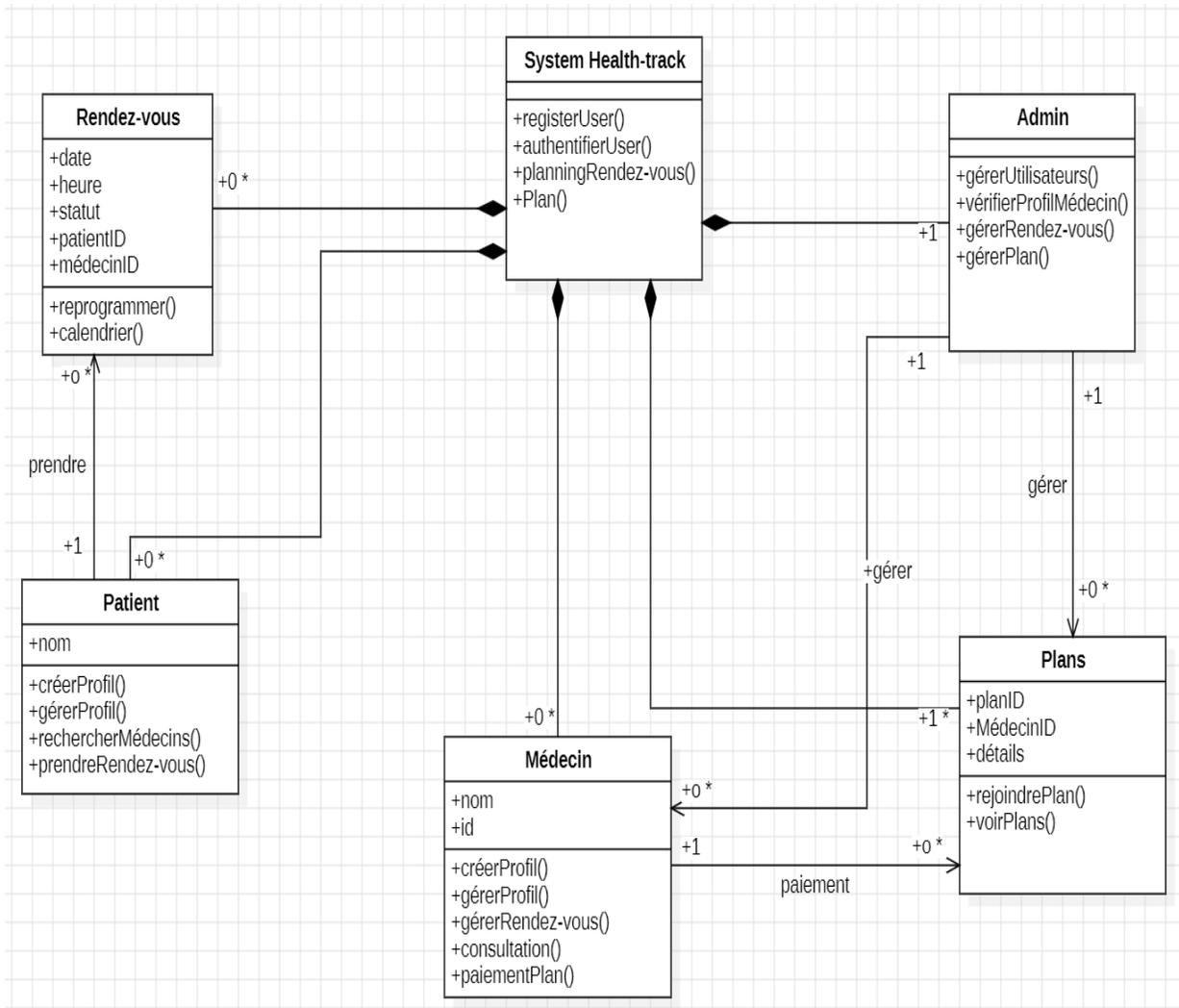


Figure 3.2: Diagramme de Classe

Diagramme de sequence:

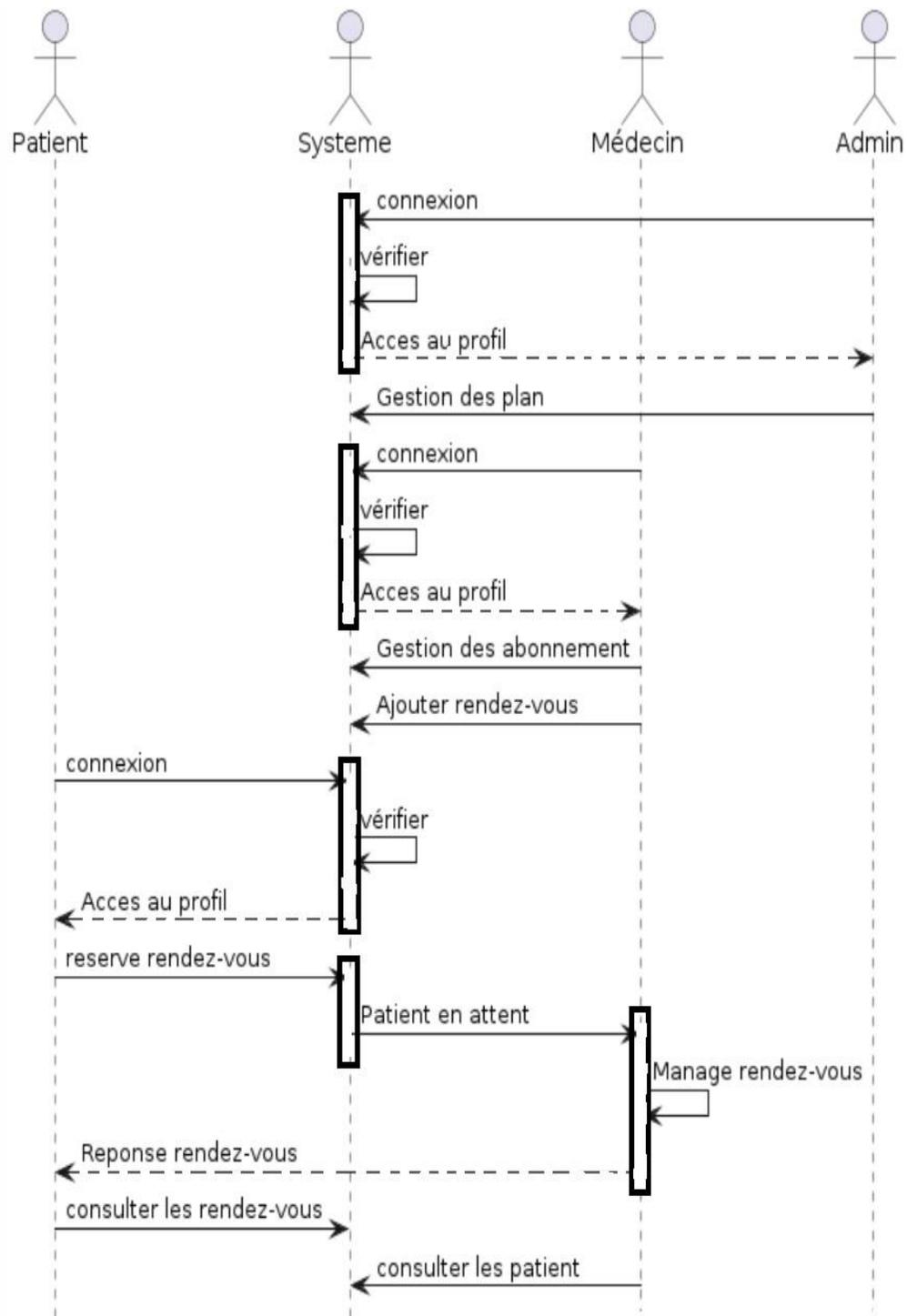


Figure 3.3 : Diagramme de sequence

3.4 Les outils d'implantation et de réalisation du système :

Environnement matériel :

Nous avons mis en œuvre le système de ce mémoire en utilisant un ordinateur "Dell" avec une configuration détaillée dans le tableau suivant

Processeur	i5-8350U CPU @ 1.70GHz 1.90 GHz
Disque dur	256 GB SSD
Mémoire vive	8 Go
Système opératoire	Windows 10 Pro
Carte graphique	Intel(R) UHD Graphics 620

Tableau 3.1: Les caractéristiques de l'ordinateur

Environnement logiciel :

Le développement d'un système peut être complexe en raison de la multitude de langages de programmation et de Framework disponibles. Le choix de la technologie à utiliser peut être coûteux en termes de temps et d'argent. Nous avons également été confrontés à ce défi, car l'accès aux plateformes payantes telles qu'Azure, Google Cloud ou SAP nous était limité en raison de l'absence d'une carte de paiement électronique. "Carte Visa". Cependant, après des recherches approfondies, nous avons découvert le Framework Laravel, qui est gratuit, et il nous a permis de développer notre système.

Laravel :

Créé en 2011 par Taylor Otwell, Laravel est devenu le logiciel gratuit et open source le plus populaire PHP dans le monde. Il est utilisé dans le développement d'applications web tout en suivant les MVC et basé sur Symfony[36].

Pourquoi Laravel?

Selon son fondateur Taylor Otwell, « Laravel est le plus puissant de l'écosystème PHP tout simplement parce qu'il inclut les fonctionnalités nécessaires pour créer des applications web modernes ». C'est vrai donc un cadre élégant et propre, avec une syntaxe élégante pour créer de grandes Applications. Parmi les avantages de ce cadre, on peut citer :

Utilise les dernières fonctionnalités de PHP,

Intégration avec les services de messagerie

Chapitre III : Conception et réalisation

Dispose d'un outil de ligne de commande intégré appelé Artisan : L'outil appelé Artisan les aide à Créer du code squelette et gérez le système de base de données avec succès. L'aide de l'artisan générer les fichiers MVC de base et gérer les ressources, y compris leurs Configurations.

Excellente documentation : Vous trouverez de bonnes explications détaillées sur le style de codage, Méthodes et classes[37]

PHP :

Abréviation de "Hypertexte Pre-processor", est un langage de script largement utilisé par les développeurs web pour créer des sites web dynamiques. Pour notre projet, nous avons spécifiquement utilisé la version 8.1.27 de PHP[38]

MySQL :

Installation du serveur web apache avec mysql : nous avons choisi le PHP le plus populaire environnement de développement XAMPP est un Apache entièrement gratuit et facile à installer distribution contenant MySQL, PHP et Perl [nous avons utilisé la version 8.1.27][39]

Visual studio code :

Visual Studio Code est un logiciel open source, gratuit et multiplateforme (Windows, Mac et Linux) éditeur de texte, développé par Microsoft[40].

Hostinger :

Hostinger International Ltd : est une société d'hébergement Web mondialement reconnue qui fournit des solutions d'hébergement.

Les services d'hébergement Web de Hostinger sont divisés en quatre catégories : l'hébergement partagé, l'hébergement cloud, l'hébergement VPS, l'hébergement WordPress géré et l'hébergement de messagerie. Les serveurs LiteSpeed et le système d'exploitation Linux alimentent tous les services de l'entreprise. Le forfait Cloud Startup, qui s'appuie sur Google Cloud Platform, n'est pas inclus[41].

Les interface du système :

Lorsque les utilisateur (admin ou locatiare ou patient) se connectent au système pour la première fois, ils sont dirigés vers la page d'accueil qui intègre la gestion de l'authentification (connexion et inscription) Cette fonctionnalité est illustrée dans la Figure 3.4.

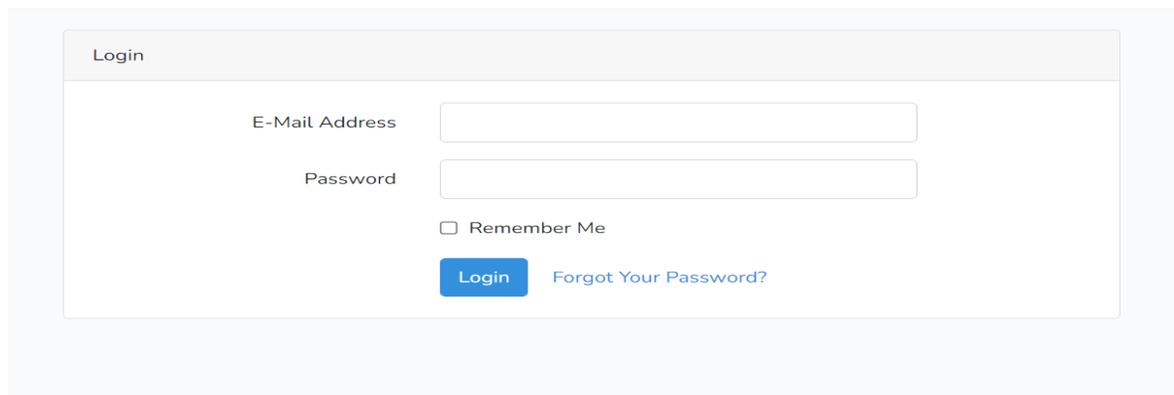


Figure 3.4: Interface principale de Health Track

L'interface de Compte admin :

Une fois que l'interface principale est affichée et le Admin appuie sur le bouton de connexion, Il doit saisir l'adresse e-mail et le mot de passe comme indiqué

dans la figure 3.5 .



The screenshot shows a 'Login' form with the following elements:

- Header: Login
- Input field: E-Mail Address
- Input field: Password
- Checkbox: Remember Me
- Buttons: Login (blue), Forgot Your Password? (text link)

Figure 3.5: Interface de Connexion

Une fois que l'admin a accédé à son compte, il met à jour ses informations comme illustré à la figure 3.6.

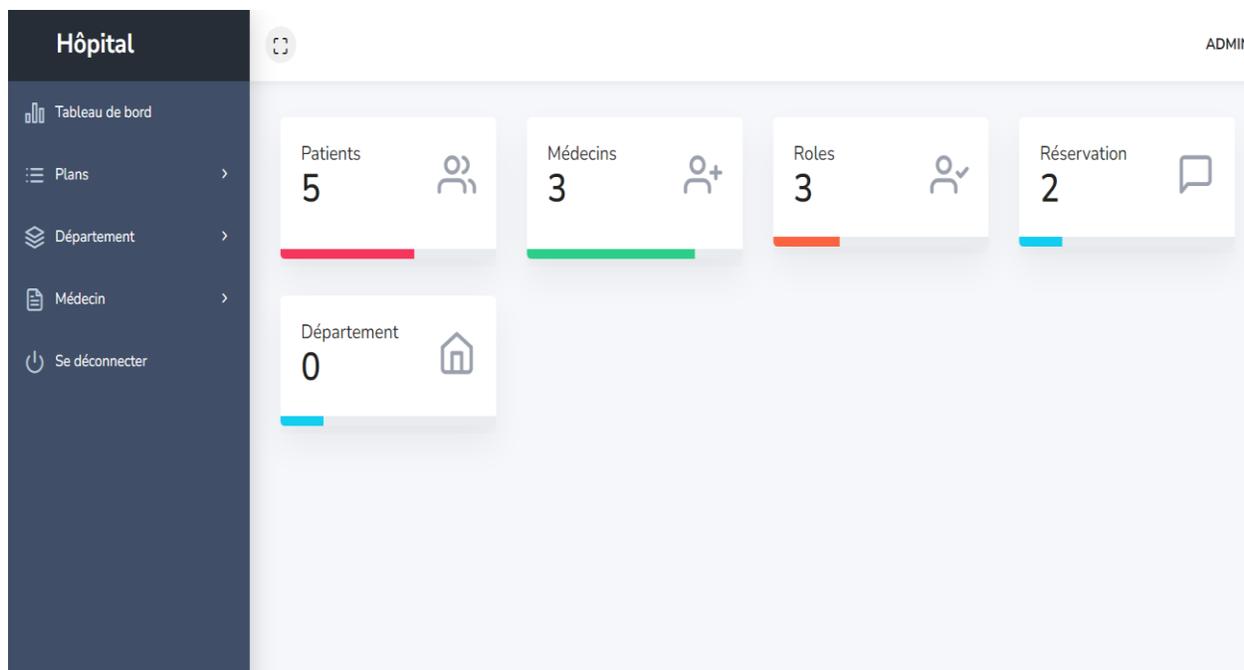
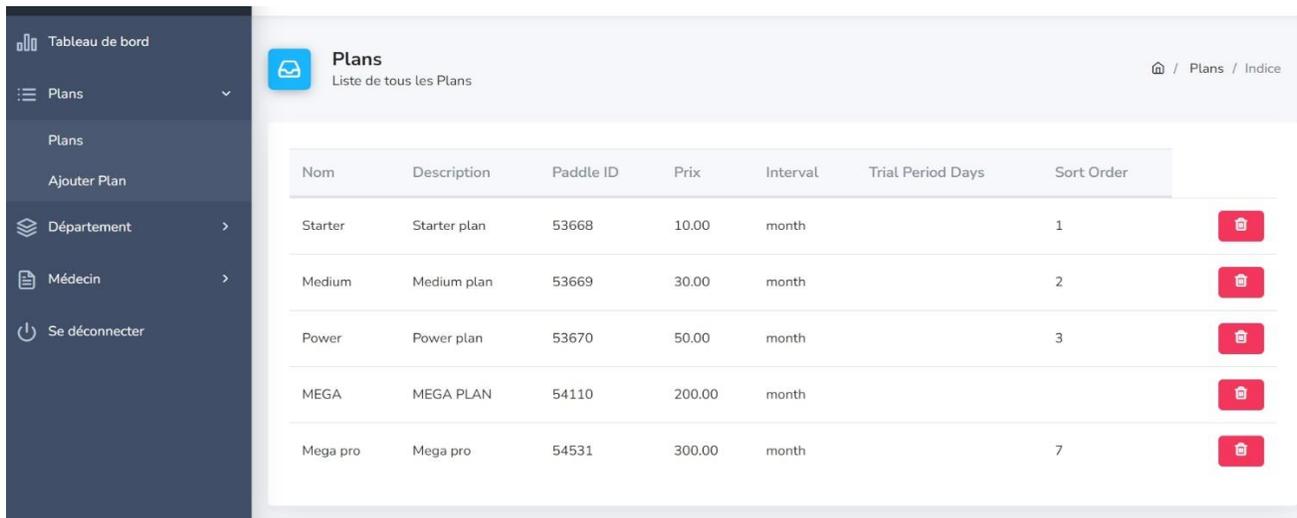


Figure 3.6: Interface de Compte Admin

Chapitre III : Conception et réalisation

Comme cette application n'est pas gratuite il existe des abonnements mensuels et annuels qui varient en fonction du nombre de réservations, Pour que l'admin puisse mettre à jour ces abonnements en appuyant sur le bouton "Plan" affichant deux options "Plans " et "Ajouter plan "

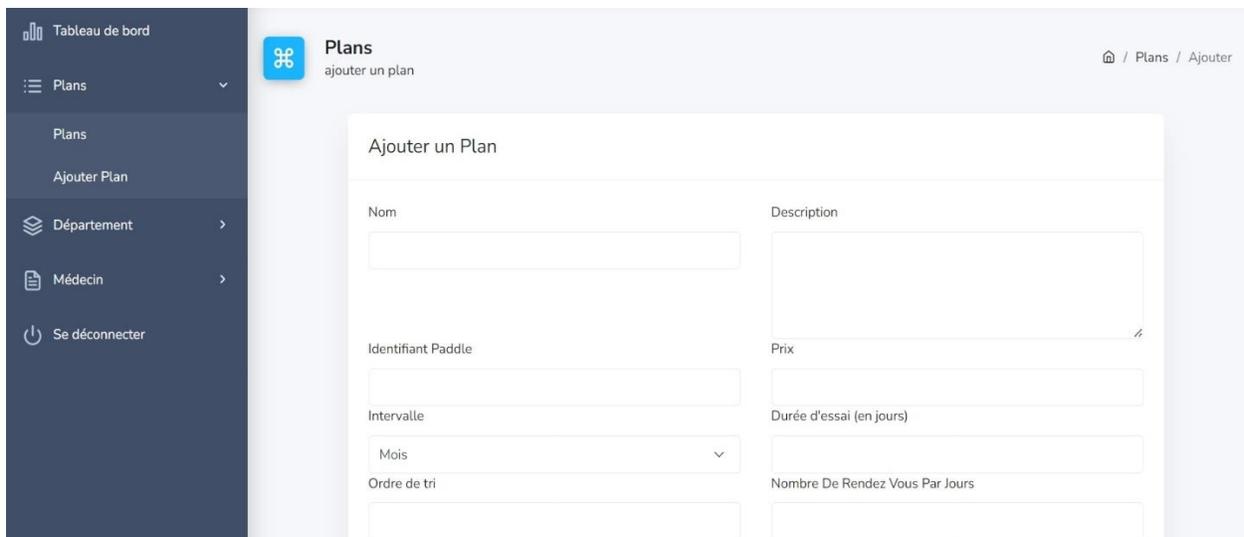
1-Plans : Pour afficher et mettre à jour des abonnements, tels que la modification ou la suppression . comme indiqué dans Figure 3.7.



Nom	Description	Paddle ID	Prix	Interval	Trial Period Days	Sort Order	
Starter	Starter plan	53668	10.00	month		1	
Medium	Medium plan	53669	30.00	month		2	
Power	Power plan	53670	50.00	month		3	
MEGA	MEGA PLAN	54110	200.00	month			
Mega pro	Mega pro	54531	300.00	month		7	

Figure 3.7: Interface d'afficher les plan

2-Ajouter Plan : Ajouter un nouvel abonnement , En renseignant les informations d'abonnement ,tel qu' illustré dans la Figure 3.8.



Ajouter un Plan

Nom	Description
<input type="text"/>	<input type="text"/>
Identifiant Paddle	Prix
<input type="text"/>	<input type="text"/>
Intervalle	Durée d'essai (en jours)
Mois <input type="button" value="v"/>	<input type="text"/>
Ordre de tri	Nombre De Rendez Vous Par Jour
<input type="text"/>	<input type="text"/>

Figure 3.8: Interface d'ajouter plan

Chapitre III : Conception et réalisation

Dans cette section, l'admin a la possibilité d'afficher la liste des médecins en appuyant sur le bouton " Vérifier ", Comme le montre la figure3.9.

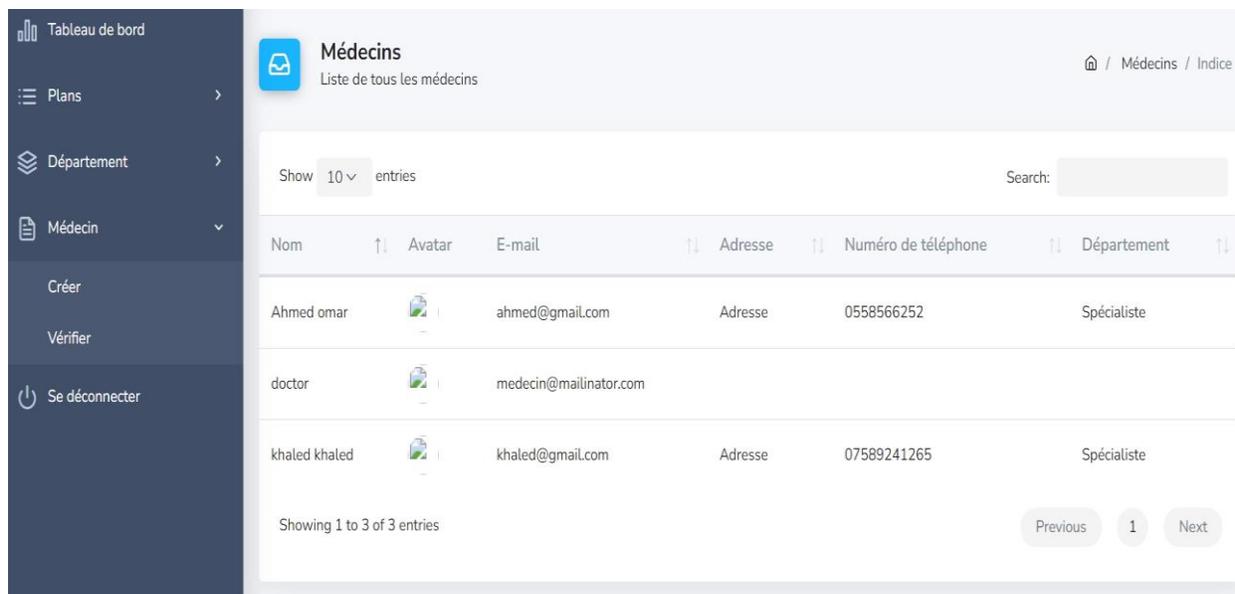


Figure 3.9: Interface d'afficher la liste des médecins

Et aussi ajouter en appuyant sur le bouton "Créer ", comme illustré à la figure 3.10

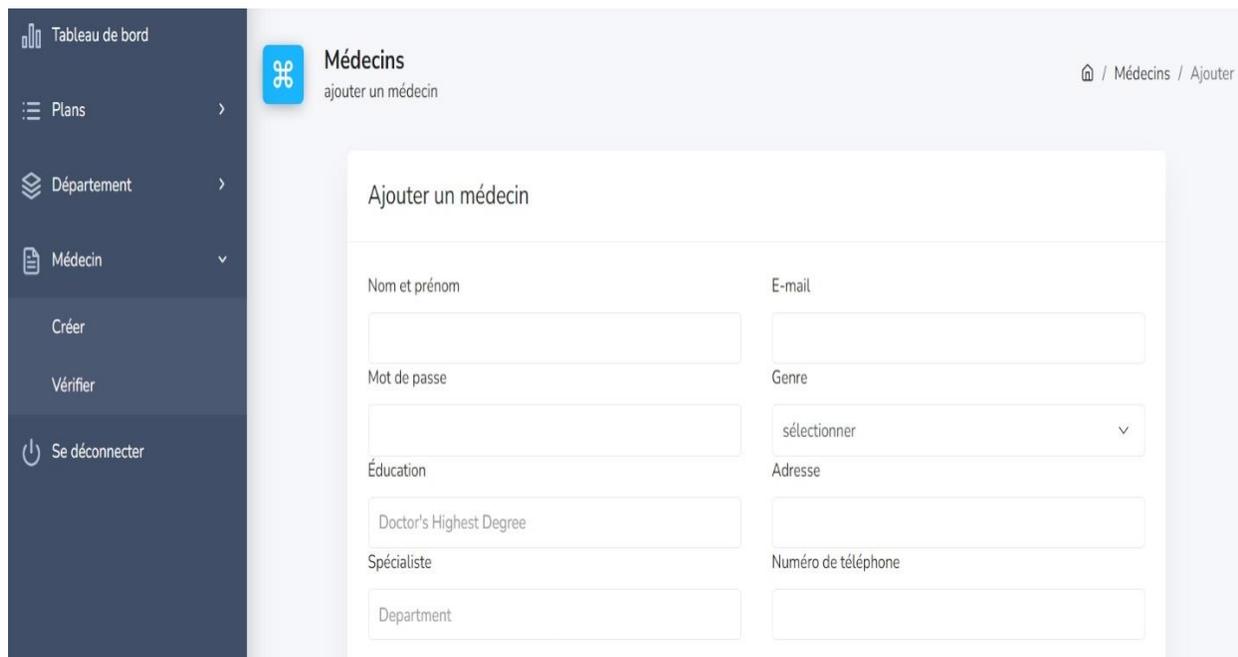


Figure 3.10: Interface d'ajouter un médecins

Chapitre III : Conception et réalisation

Si l'admin veut se déconnecter, il appuie sur le bouton " Se déconnecter " Cela est illustré dans la Figure 3.11.

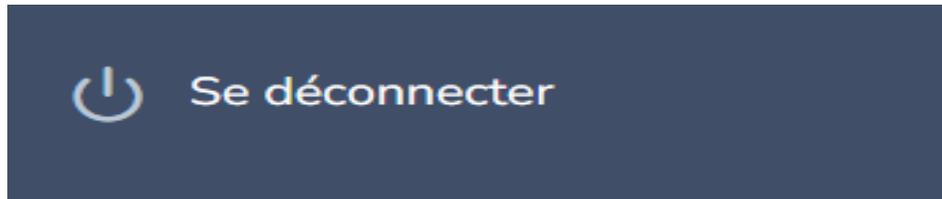


Figure 3.11: Interface de déconnexion

L'interface de Compte Locataire :

Une fois que le locataire accède à l'interface principale de site, le locataire appuie sur le bouton de connexion (ou inscription), il doit d'abord s'inscrire en fournissant son nom, son adresse e-mail et son mot de passe, et en les confirmant comme illustré dans la Figure 3.12.

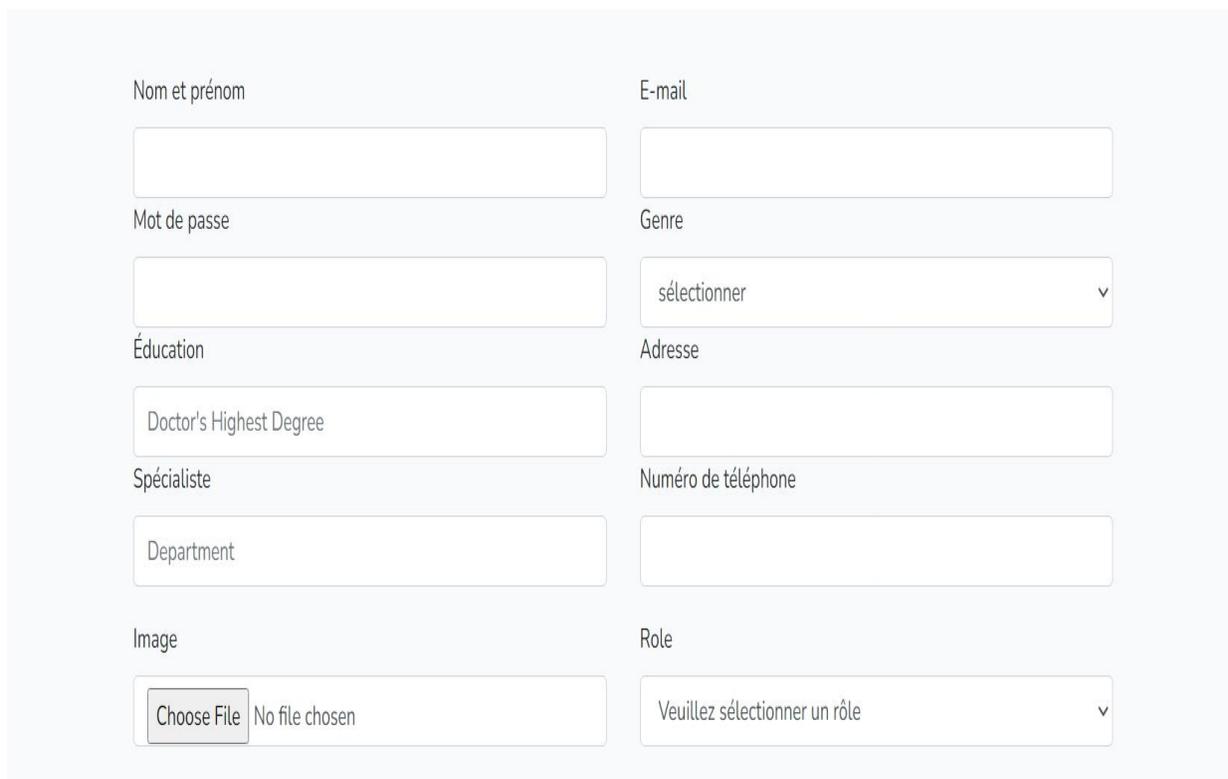
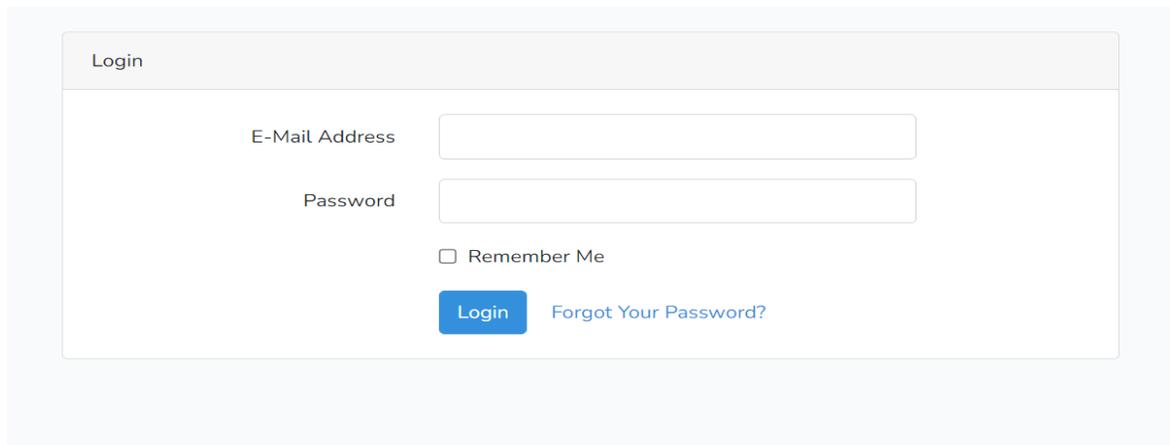


Figure 3.12: Interface d'inscription

Chapitre III : Conception et réalisation

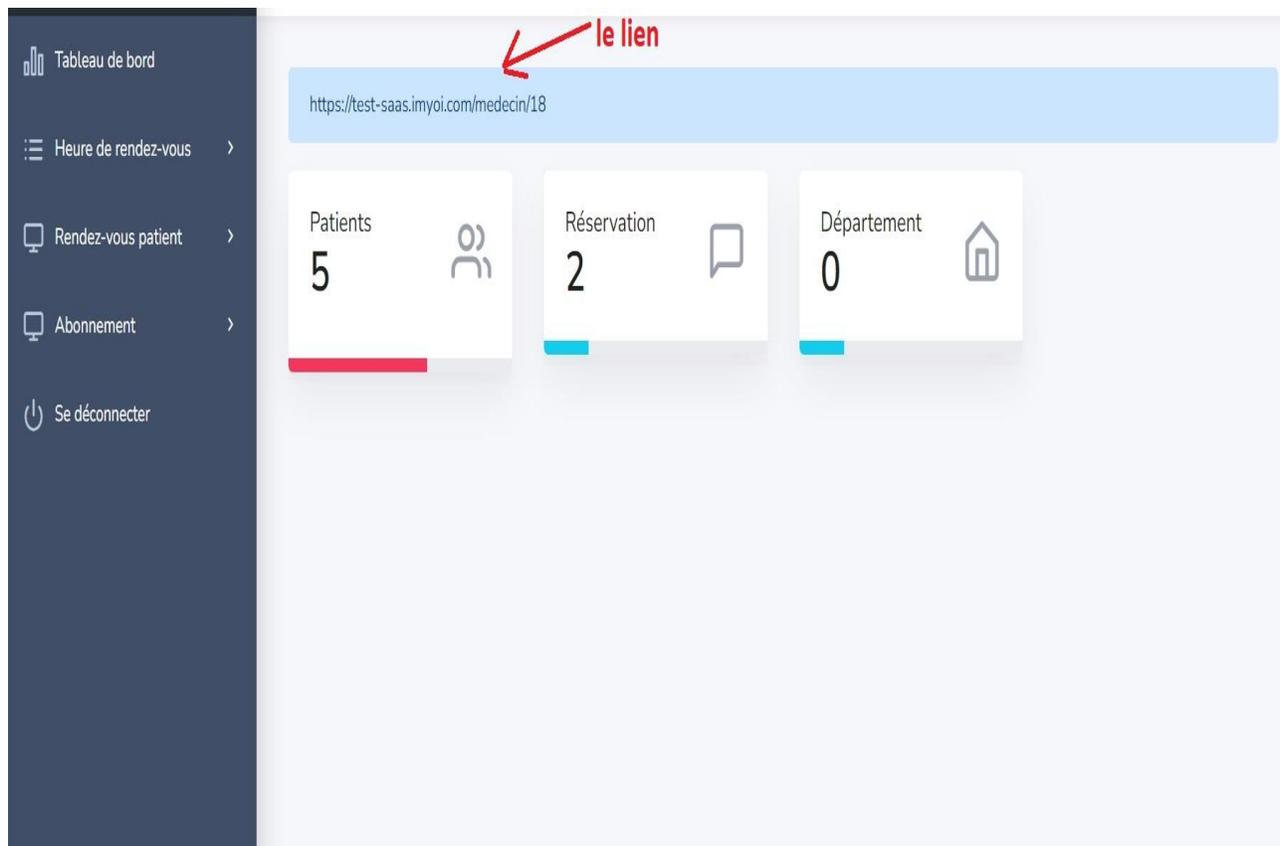
Alternativement, il peut simplement saisir son adresse e-mail et son mot de passe comme indiqué dans la Figure 3.13



The image shows a login form titled "Login". It contains two input fields: "E-Mail Address" and "Password". Below the password field is a checkbox labeled "Remember Me". At the bottom, there is a blue "Login" button and a link "Forgot Your Password?".

Figure 3.13: Interface de Connexion

Une fois que le locataire s'est connecté à son compte,, Cette interface s'affiche Avec son lien qu'il fournit aux clients pour prendre rendez-vous , comme indiqué dans la Figure 3.14



The image shows a dashboard interface. On the left is a dark sidebar with menu items: "Tableau de bord", "Heure de rendez-vous", "Rendez-vous patient", "Abonnement", and "Se déconnecter". The main content area has a light blue header with a URL: <https://test-saas.imyoi.com/medecin/18>. A red arrow points to this URL with the text "le lien". Below the header are three cards: "Patients" with a value of 5, "Réservation" with a value of 2, and "Département" with a value of 0. Each card has a corresponding icon and a colored bar at the bottom.

Figure 3.14: interface s'affiche Avec son lien qu'il fournit aux clients

Chapitre III : Conception et réalisation

Étant donné que ce système n'est pas gratuite, il existe des abonnements, Si le locataire souhaite s'abonner, il doit choisir l'offre qui lui convient



Figure 3.15: L'interface de l'Abonnement

Une fois que le locataire a choisi le type d'abonnement, il enregistre le pays de résidence et choisit le mode de paiement par carte ou PayPal, comme indiqué à la figure 3.16

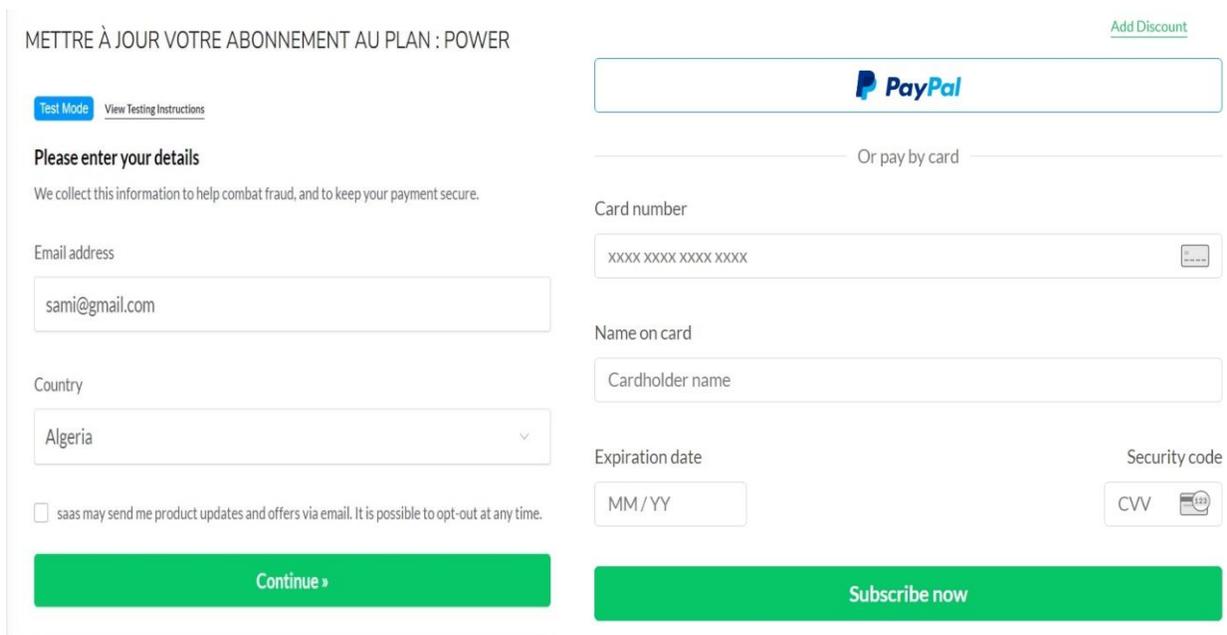


Figure 3.16: Interface du pays d'abonnement.

Chapitre III : Conception et réalisation

Après le paiement, le statut de l'abonnement est mis à jour et La date du prochain paiement s'affiche , comme illustré à la figure 3.17

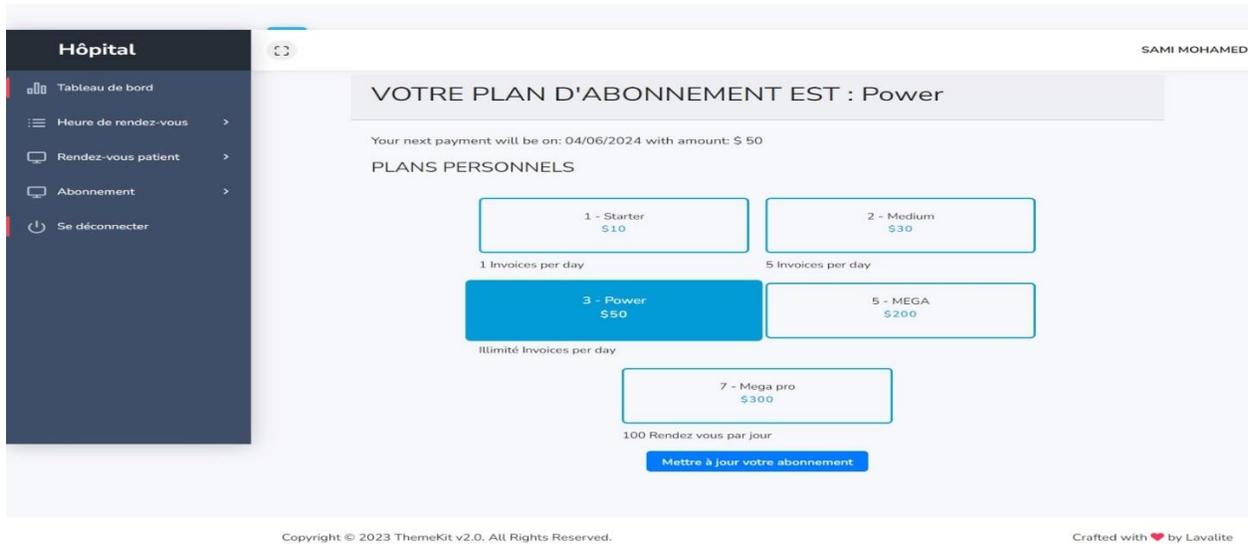


Figure 3.17: Interface de votre plan d'abonnement.

Après s'être abonné, le système permet au locataire d'effectuer plusieurs actions.

Ajouter et Consulter la liste des rendez vous , comme indiqué à la figure 3.18

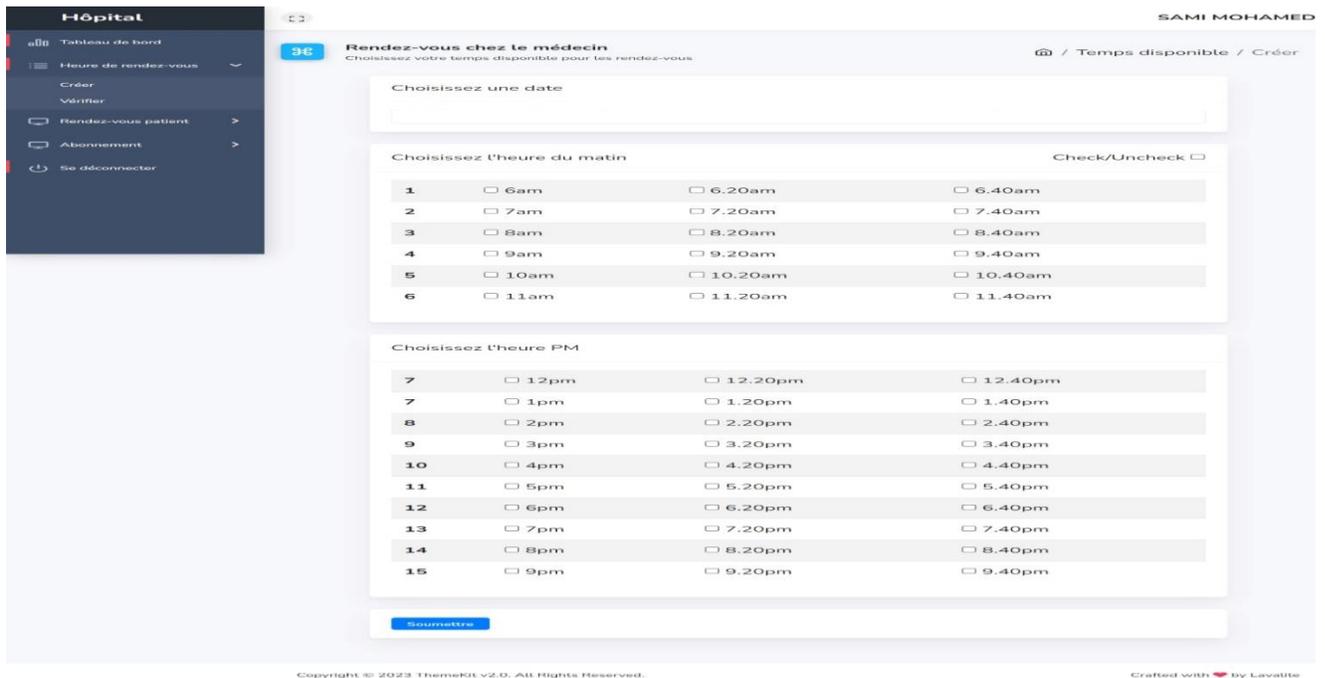


Figure 3.18: Interface d'ajouter et Consulter la liste des rendez vous .

Chapitre III : Conception et réalisation

Consulter la liste des patients qui prennent rendez-vous et approuver les rendez-vous en attente , Comme le montre la figure3.19

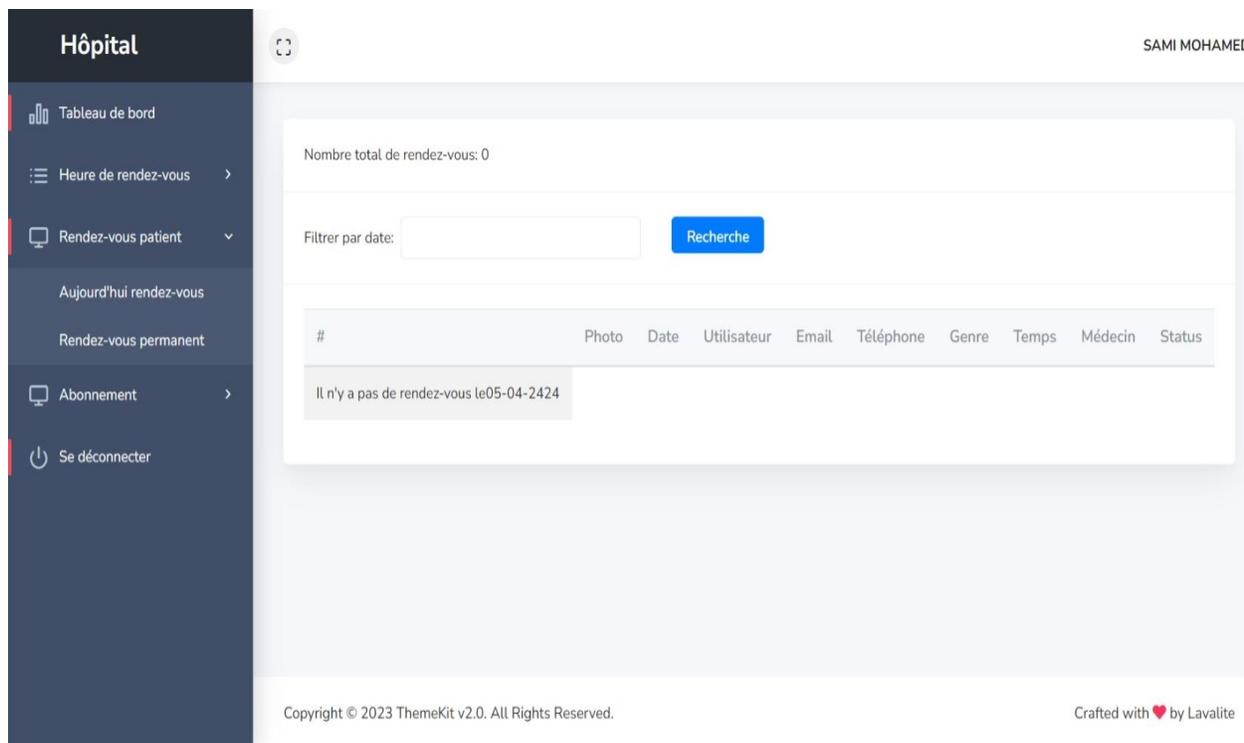


Figure 3.19: Interface des rendez-vous

L'interface de Compte patient :

Après que le locataire publié son lien vers les clients , Le patient crée un compte en fournissant son nom, son adresse e- mail et son mot de passe, et en les confirmant comme illustré dans la Figure 3.20.

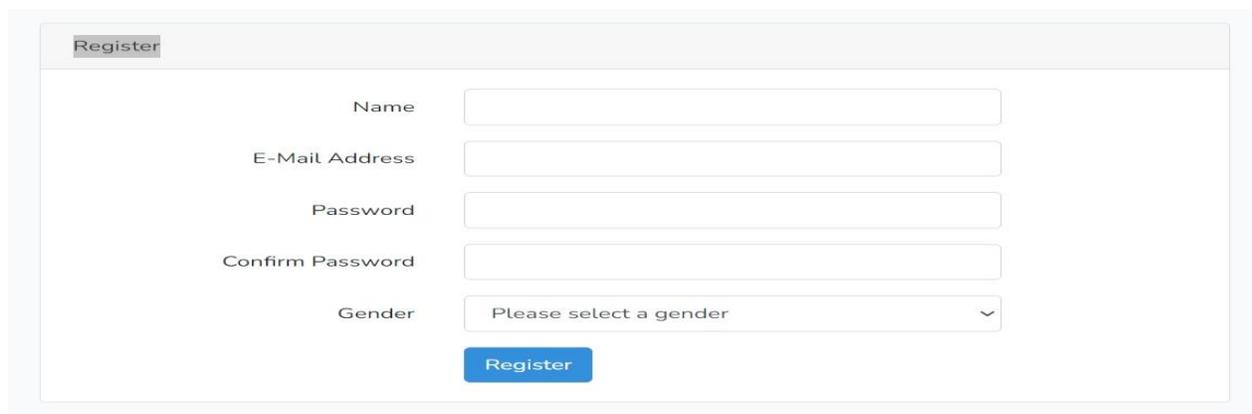


Figure 3.20: Interface d'inscription

Chapitre III : Conception et réalisation

Une fois qu le patient créé un compte , il peut prendre et consulté des rendez-vous ,

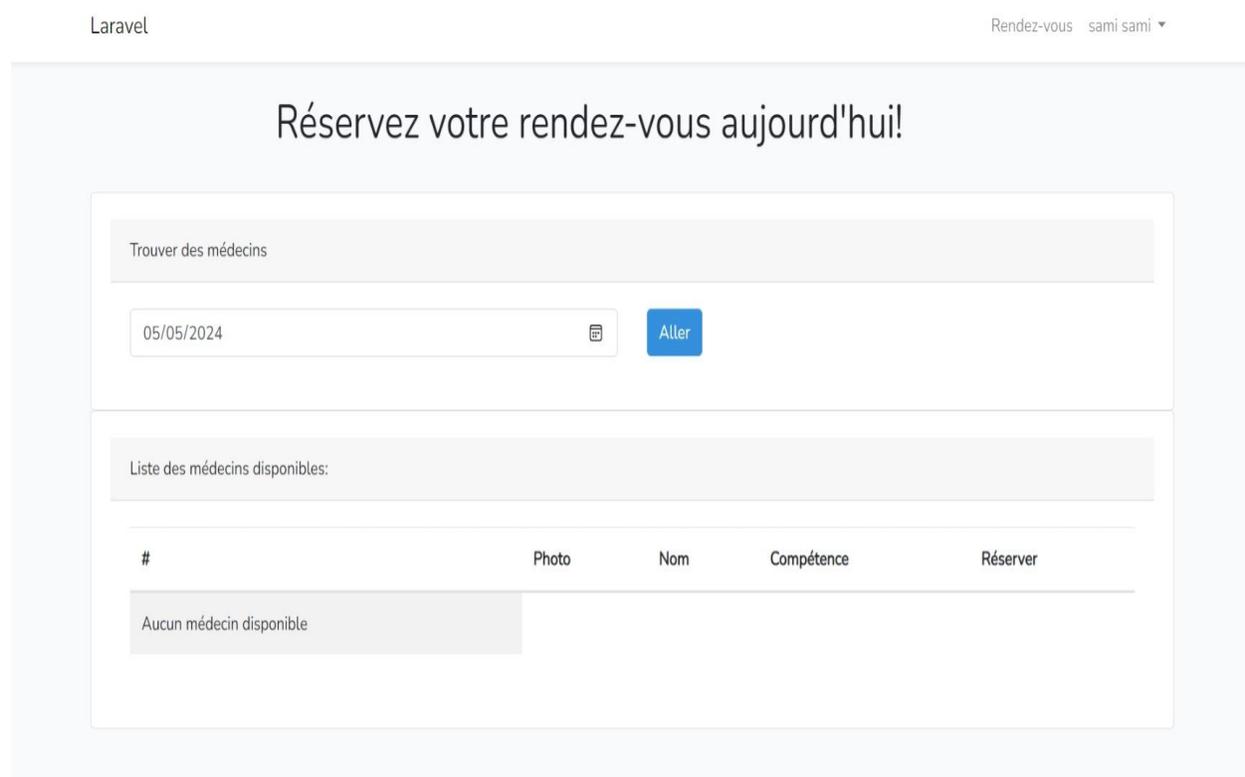


Figure 3.21: Interface des prendre et consulté les rendez-vous

Conclusion :

Dans ce chapitre final, nous avons fourni une description détaillée de l'implémentation et de la réalisation de notre système. Nous avons présenté l'environnement matériel et logiciel utilisé pendant l'implémentation, ainsi que la description du système d'abonnement ciblé par notre approche. De plus, nous avons exposé le processus de développement suivi et la méthodologie adoptée, en nous inspirant des meilleures pratiques. Enfin, nous avons expliqué les différentes interfaces de notre système multi-tenant basé sur l'abonnement.

Conclusion Générale & Perspectives

Conclusion Générale & Perspectives

L'intégration du modèle de service SaaS avec une architecture multi-tenant sur des applications hébergées dans le cloud offre une approche puissante pour répondre aux besoins des entreprises modernes. En combinant la flexibilité et la scalabilité du cloud avec la capacité à servir simultanément plusieurs clients dans un environnement sécurisé, cette approche optimise les coûts tout en offrant une expérience utilisateur de qualité. En conclusion, cette synergie entre SaaS et architecture multi-tenant dans le cloud représente une stratégie gagnante pour les entreprises cherchant à rester compétitives dans un environnement numérique en constante évolution.

Dans le cadre de ce mémoire, nous avons documenté les efforts entrepris pour créer et mettre en place une application d'abonnement multi-locataire en utilisant le Framework Laravel, avec une approche de base de données unique. L'objectif principal de ce travail était de décrire les étapes suivies pour la conception et la réalisation du système d'abonnement destiné .

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 (à base de données unique ou multiple) dans le cloud computing.
- La présentation et l'apprentissage du Framework Laravel, qui offre d'excellentes bibliothèques pour le modèle SaaS et l'architecture multi-locataire.
- La conception et mise en place d'un système d'abonnement SaaS typique nommé "Health Track" avec le Framework Laravel.

À long terme, l'architecture multi-tenant présentée dans ce mémoire vise à s'intégrer dans un environnement virtualisé basé sur le cloud computing en tant que paradigme de recherche. Cependant, dans une perspective future, nous envisageons de proposer des implémentations similaires dans d'autres environnements virtualisés. Cela signifie que nous pourrions explorer l'implémentation du multitenancy dans des domaines tels que le Fog computing, Edge computing, et d'autres encore.

Références Bibliographique

- [1] K. Rubenstein, *Cloud Computing in Life Sciences R&D*. Insight Pharma Reports. Cambridge Healthtech Institute, 2010.
- [2] S. Rajan and A. Jairath, "Cloud computing: The fifth generation of computing," in 2011 International Conference on Communication Systems and Network Technologies, 2011, pp. 665–667.
- [3] É. Michon, J. Gossa, S. Genaud, L. Unbekandt, and V. Kherbache, "Schlounder: A broker for IaaS clouds," *Futur. Gener. Comput. Syst.*, vol. 69, pp. 11–23, 2017.
- [4] E. Simmon, "Evaluation of Cloud Computing Services Based on NIST SP 800-145," *NIST Spec. Publ.*, vol. 500, p. 322, 2018.
- [6] B. Furht and A. Escalante, *Handbook of cloud computing*, vol. 3. Springer, 2010.
- [7] R. Buyya, C. Vecchiola, and S. T. Selvi, *Mastering cloud computing: foundations and applications programming*. Newnes, 2013.
- [10] Silia BENKHALLEF and Amel ABER. *Design and Realization of Cloud SaaS Multi-Tenant Application*. PhD thesis, Université Ibn Khaldoun-Tiaret-, 2022.
- [11] R. Bhoyar and N. Chopde, "Cloud computing: Service models, types, database and issues.," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 3, 2013.
- [12] B. Sosinsky, *Cloud computing bible*, vol. 762. John Wiley & Sons, 2010.
- [13] O. Avila. *Computación en la nube*. *Contacto S*, 80:45–52, 2011.
- [14] R. Bhoyar and N. Chopde, "Cloud computing: Service models, types, database and issues.," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 3, 2013.
- [16] R. Rezaei, T. K. Chiew, S. P. Lee, and Z. S. Aliee, "A semantic interoperability framework for software as a service systems in cloud computing environments," Elsevier, 2014.
- [17] F. Liu et al., "NIST cloud computing reference architecture," *NIST Spec. Publ.*, vol. 500, no. 2011, pp. 1–28, 2011.
- [18] H. Ali, R. Moawad, and A. A. F. Hosni, "A Cloud Interoperability Broker (CIB) for data migration in SaaS," *Futur. Comput. Informatics J.*, vol. 1, no. 1–2, pp. 27–34, 2016.
- [19] S. S. Chauhan, E. S. Pilli, R. C. Joshi, G. Singh, and M. C. Govil, "Brokering in interconnected cloud computing environments: A survey," *J. Parallel Distrib. Comput.*, 2018.
- [23] Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud Computing: Concepts, Technology & Architecture*. Prentice Hall.
- [25] Gozde Karatas Baydogmus, Akhan Akbulut, Ferit Can. " Multi-tenant architectures in the cloud: A systematic mapping study "Conference Paper · September 2017

Références Bibliographique

- [26] Doudou Fall, Gregory Blanc, Takeshi Okuda, Youki Kadobayashi, and Suguru Yamaguchi. Toward quantified risk-adaptive access control for multi-tenant cloud computing. In The 6th Joint Workshop on Information Security, pages 1–14, 2011.
- [27] V.V. bandgar, G.A. Fattepurkar, C.M.Jadhav, S.M. Kawale, "Development of Multitenant Application as SAAS", In International Journal of Engineering Research & Technology, pp. 2, (2015).
- [28] M. Yang and H. Zhou, "New Solution for Isolation of Multi-tenant in cloud computing", 3rd International Conference on Mechatronics, Robotics and Automation, Beijing, China, pp. 334-335, (2015).
- [30] Matt Stauffer. Laravel : Up & running : A framework for building modern php apps. O'Reilly Media, 2019.
- [32] Thomas Risberg Mark Pollack Michael Hunger Jonathan L. Brisbin, Oliver Gierke. Spring data,. June 29, Year 2023.
- [33] William S Vincent. Django for Beginners : Build websites with Python and Django. WelcomeToCode, 2022.
- [34] Eyad Saleh, Nuhad Shaabani, and Christoph Meinel. A framework for migrating traditional web applications into multi-tenant saas. In The second International Conference on Advanced Communications and Computation (INFOCOMP),pages 100–104, 2012.
- [35] Cor-Paul Bezemer, Andy Zaidman, Bart Platzbeecker, Toine Hurkmans, et al. Enabling multi-tenancy : An industrial experience report. In 2010 IEEE International Conference on Software Maintenance, pages 1–8. IEEE, 2010.
- [37] Andri Sunardi et al. Mvc architecture : A comparative study between laravel framework and slim framework in freelancer project monitoring system web based. Procedia Computer Science, 157 :134–141, 2019.
- [38] Danial Kafi Ahmad, Mariam Farida Ahmad, Mohd Nazmi Ahmad, and Abdullah Syafiq Ahmad. An experiment of animation development in hypertext preprocessor (php) and hypertext markup language (html). Int. J. Sci. Res. in Computer Science and Engineering Vol, 8(2), 2020.

Webographies :

- [5] "Cloud computing and Hosting " [en ligne]Disponible(www.vicoast.com/cloud-computing-and-hosting/)[visité le 1/4/2024]
- [8] "what is cloud"[en ligne]Disponible (techniblogic.com/what-is-cloud-computing-a-beginners-guide/)[visité le 3/4/2024]
- [9] « cloud computing » [En ligne] Disponible(www.azure.microsoft.com) [visité le 12/04/2024]
- [15] "Le cloud computing" [Enligne]. Disponible : www-igm.univ-mlv.fr/~dr/XPOSE2009/cloudcomputing/types.html. [Visité le 30/04/2024].

Références Bibliographique

- [20] " Single-tenant "[En ligne] Disponible (<https://www.nist.gov/publications/nist-cloud-computing-standards-roadmap>)(visité le 30/04/2024)
- [21] "Multi-tenant " [En ligne]Disponible(<https://standards.ieee.org/ieee/2302/7056/>)(visité le 30/04/2024)
- [22] "Single Tenant versus Multi Tenant "[En ligne] Disponible (blog.leotechnosoft.net/single-versus-multi-tenant-saas-which-is-best/)(visité le 30/04/2024)
- [24] "Single Tenant vs Multi Tenant "[En ligne] Disponible (www.sailpoint.com/identity-library/single-tenant-multi-tenant/)(Visite le 30/04/2024)
- [29] "Multi location" [en ligne]Disponible (fastercapital.com/fr/contenu/Multi-location--demystifier-multi-location-en-SaaS.html) (Visite le 30/04/2024)
- [31] "Experss" [en ligne] disponible(<https://expressjs.com/>) (visité le 10/05/2024).
- [36] "Laravel" [en ligne] disponible(<https://en.wikipedia.org/wiki/Laravel>) (visité le 10/01/2024)
- [39] "Apache" [en ligne] disponible(www.apachefriends.org) (visité le 1/04/2024)
- [40] "Visualstudio code"[en ligne] disponible (www.visualstudio.com) (visité le 1/03/2024).
- [41] "Hostinger" [en ligne] disponible (en.wikipedia.org/wiki/Hostinger) (visité le 1/05/2024)