

الجمهورية الجزائرية الديمقراطية الشعبية

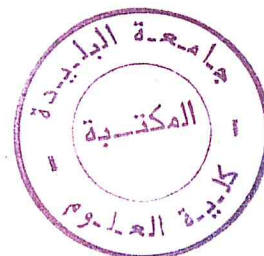
République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahleb de Blida

Faculté des sciences exactes

Département d'informatique



Mémoire de fin d'études :

En vue de l'obtention du diplôme
d'ingénieur d'état en informatique

Option : système d'information

THEME

*Conception et réalisation d'un
système de téléconsultation médicale
en utilisant la technologie WAP.*

Proposé et encadré par : Mlle Semar kahina

Présenté par :

Adis Rabah.

Nebbaa Rachid.

Organisme d'accueil : Centre de développement des technologies avancées (CDTA)
Baba Hassen

PROMOTION : 2003/2004

MIG-004-35-1



Dédicaces

Je remercie « Allah », le tout puissant de m'avoir donné la santé et la force nécessaire pour pouvoir terminer ce modeste travail que je dédie :

A mes très chers parents pour leur intérêt constant, que dieu tout puissant vous protège, et vous accord long vie.

A mes cher frères et sœurs qui n'ont pas cessé de m'encourager tout au long de mon projet.

A toute ma famille.

A mon très cher amis et mon partenaire Rachid Nebbaa.

*A mes très cher amis: B.M'hamed, T.AEK, H.AEK,, Ali et Rachid,
D.Redouane, Hamid, Brahim, Farouk, Arbi, Ali, Hamza, F.Mourad,*

H.M'hamed.

Et en fin à tous ceux que j'ai connu.

Rabah

Dédicaces

Je remercie « Allah », le tout puissant de m'avoir donné la santé et la force nécessaire pour pouvoir terminer ce modeste travail que je dédie :

A mon très cher père Taher et à ma très cher mère Kheira pour leur intérêt constant, que dieu tout puissant vous protège, et vous accord long vie.

A la mémoire de mes grands pères

A mes grands mères pour leur attentions et leurs encouragement.

A mes cher frères et sœurs qui n'ont pas cessé de m'encourager tout au long de mon projet.

A mes chers oncles pour leur soutient.

A toute ma famille

A mon très cher amis et mon partenaire Rabah Adis.

A mes très cher amis.

A la mémoire de mon ami : Abache Said.

Et en fin à tous ceux que j'ai connu.

Rachid

Remerciements

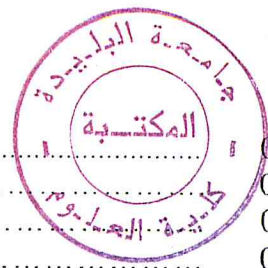
Nous adressons tous nos remerciements les plus chaleureux à Mlle : S. Kahina, qui à assumé lourde tache de l'encadrement et de suivi du projet.

Nous tenons à remercier l'équipe de centre de développement des technologies avancées (CDTA) surtout l'équipe du laboratoire d'architectures des systèmes (AS).

A tout les enseignants de département d'informatique.

Il nous est, enfin, agréable d'exprimer notre gratitude à toutes les personnes qui nous ont apporté un appui moral ou matériel durant l'élaboration de notre projet.

Sommaire



Résumé.....	01
Introduction générale.....	02
CHAPITRE I : La téléconsultation médicale	04
Introduction	04
I- La télémédecine	04
I-1- Définitions	04
I-2- Objectifs de la télémédecine	05
I-3- Typologie de la télémédecine	06
I-4- Les services de la télémédecine	07
I-5- Technologies de la télémédecine	07
I-5-1- Technologies de télécommunications	07
I-5-2- Technologie de l'information	07
II- La téléconsultation médicale	08
II-1- La messagerie électronique	08
II-2- Les forums	09
II-3- La conversation textuelle synchrone (Chat)	09
III- La téléconsultation médicale via le WAP	09
III-1- Accès au réseau	11
III-2- Communications	11
III-3- Interface utilisateur	11
Conclusion	11
CHAPITRE II : La technologie WAP	12
Introduction	12
I- Le protocole WAP (Wireless Application Protocol)	13
I-1- Le modèle World Wide Web et le modèle WAP	15
I-1-1- Le modèle World Wide Web (WWW)	15
I-1-2- Le modèle WAP	16
I-2- Les différents composants de la technologie WAP	18
I-2-1- Le serveur Web	18
I-2-2- La passerelle WAP (gateway)/ serveur WAP	18
I-2-3- Le réseau de l'opérateur mobile	19
I-2-4- Le navigateur (sur le téléphone portable ou PDA)	19
I-3- Les différentes implémentations d'une plate forme WAP	20
I-3-1- Passerelle WAP hébergée chez l'opérateur de téléphonie mobile	20
I-3-2- Passerelle WAP hébergée chez un fournisseur d'accès	21
I-3-2-1- Avec hébergement des serveurs Web	21
I-3-2-2- Sans hébergement des serveurs Web	21
I-3-3- Passerelle WAP en interne	22
I-4- Modes de Communication dans un réseau WAP (PULL et PUSH).....	22
I-4-1 Le pull	23
I-4-2 Le push	25
I-5- Les services accessible par la technologie WAP	25
I-5-1 Services de communication	25
I-5-2- Services d'information et media	26
I-5-3- Services de m-commerce.....	26
II- Le WAP et la sécurité.....	26
III- les avantages et les inconvénients de la technologie WAP	27
Conclusion.....	28

CHAPITRE III : le langage WML	29
Introduction.....	29
I- Le langage WML (Wireless Markup Language).....	29
I-1- Caractéristique du langage WML	30
I-2- Structure d'un deck	32
II- Le langage WML script	33
III- Les outils de développement	34
III-1- Editeurs WML.....	34
III-1-1- Easypad WAPtor	34
III-1-2- DotWAP	34
III-2- les convertisseurs d'images	35
III-3- Les émulateurs WAP	35
III-4- Les kits de développement	36
III-4-1- Nokia Internet mobile toolkit	36
III-4-2- OpenWave SDK	37
Conclusion	38
CHAPITRE IV : interfaces WAP/Base de données	39
Introduction.....	39
I- Les différentes interfaces Web/base de données	39
I-1- Common Gateway Interfaces (CGI).....	39
I-2- Internet Server Application Programming Interfaces (ISAPI)	39
I-3- Active Server Pages [ASP]	39
I-3-1- La structure générale du modèle ASP	40
I-4- Personal Home Page [PHP].....	42
I-4-1- Syntaxe de base	42
I-4-1-1- Intégration de script PHP dans les pages WML.....	42
I-4-1-2- Les commentaires	42
I-4-1-3- Types de donnée	43
I-4-2- Les variables /constantes	43
I-4-3- Les tableaux	43
I-4-4- Les fonctions	43
I-4-5- Structures de contrôle	44
I-4-5-1- L'instruction <i>if</i>	45
I-4-5-2- L'instruction <i>switch</i>	45
I-4-5-3- Boucle <i>for</i>	45
I-4-5-4- Boucle <i>while</i>	46
I-4-5-5- Boucle <i>do... while</i>	46
I-4-5-6- Boucle <i>foreach</i>	46
I-4-6- Entrée/Sortie (I/O) et accès au disque	46
I-4-6-1- Envoyer des données vers le navigateur	46
I-4-6-2- Récupérer des données à partir d'un formulaire	47
I-4-6-3- Les cookies	47
I-4-6-4- Lecture et écriture de fichier	47
I-4-6-5- Accès aux bases de données	47
II- Les composants des interfaces Web/Base de données	47
II-1- Active Database Object (ADO).....	48

II-1-1- Objet Connexion	49
II-1-2- Objet Command	49
II-1-3- Objet Recordset	50
II-2- Le module ODBC du PHP	51
II-2-1- Connexion à une base de donnée	52
II-2-2- Exécution d'une commande	52
II-2-3- Récupération d'un résultat	52
II-2-4- Manipulation des champs	52
II-2-5- Préparer et exécuter une requête	53
II-2-6- Fermeture d'une connexion	53
III- Choix de la technique	53
Conclusion.....	54
CHAPITRE V : Conception	55
Introduction.....	55
I- Infrastructures et équipement	55
II- Description des modules :.....	58
II-1- Droit d'accès :.....	58
II-2- Module d'inscription :.....	58
II-3- Générateur des informations médicales.....	60
II-4- questionnaire des médecins et malades.....	60
II-5- Gestionnaire des pathologies, médicaments et examens.....	61
II-6- Module de communication	61
III- description du système d'information	61
III-1- base de donnée de téléconsultation	61
III-2- Forum	65
III-3- Messagerie	67
III-4- Chat.....	68
IV- description de l'interface utilisateur.....	69
IV-1- Le module base de données	70
IV-2- Le module forum de discussion	71
IV-3- Le module de messagerie	72
IV-4- Le module de chat	73
Conclusion.....	73
CHAPITRE VI : Implémentation et test de l'application	74
Introduction	74
I- Présentation de l'environnement technique	74
I-1- Les outils WEB	74
I-1-1- Serveur Web IIS (Internet Information Serveur) 5.0	74
I-1-2- Le langage de script : PHP (Personal Home Page)	75
I-2- Les outils WAP	75
I-2-1- Le module Nokia WAP Toolkit 3.1	75
I-2-2- La passerelle WAP (WAPLite).....	76
I-2-3- L'éditeur de page WML.....	77
I-2-4- Outils graphique WAP.....	77
I-3- L'implémentation de la base de données.....	78
II- Etude technique détaillée.....	78
II-1- Configuration du post client.....	78

III- Configuration du serveur WEB/WAP	79
III-1- répertoire virtuel	79
III-2- Les type mimes	79
III-3- Le script PHP	80
III-4- Création de la source de données système (DSN).....	81
IV- L'application	82
IV-1- Analyse fonctionnelle	82
IV-2- Test de l'application	83
IV-2-1- Accès a la base de données	84
IV-2-2- Le module de 'Forum de discussion'	88
IV-2-3- Accès au 'chat'	92
IV-2-4- Le module de messagerie	93
Conclusion.....	94
Conclusion générale.....	95
Annexe.....	97
Glossaire.....	102
Bibliographie.....	106

Résumé

Le but de ce projet est de mettre au point un système de téléconsultation médicale, afin d'interconnecter les médecins du territoire national entre eux.

L'application réalisée se base sur la technologie WAP, elle se compose de plusieurs module : accès aux bases de données, forum de discussion, chat et messagerie, que nous allons présenter en détail tout long de ce mémoire.

Notre but principal est d'offrir aux médecins lorsqu'ils sont en déplacement un moyen de communication virtuel qui permet l'échange de leurs opinions et leur savoir-faire.

Mot clé : WAP, WML, Mobile, Télémedecine, PHP, IIS, Internet, BDD.....

Introduction générale

INTRODUCTION GENERALE

Depuis une dizaine d'années, l'Internet a connu un succès incroyable. Aujourd'hui, on recense plus d'un milliard de pages Web, mais jusqu'à présent, la majorité des internautes se connectaient depuis un poste fixe via une ligne téléphonique. Cette technique bien que performante, a pour contrainte de ne pas être mobile car on ne peut se connecter que d'un endroit fixe en utilisant un micro-ordinateur qui reste un peu coûteux. L'Internet est utilisée dans plusieurs domaines tels que la santé, on parle donc de la télémédecine.

La télémédecine est l'exercice de la médecine à distance, permet à des médecins d'échanger des informations entre eux, d'effectuer des consultations ou d'analyser et interpréter des examens complémentaires, en particulier de l'imagerie médicale.

C'est face à ce besoin de mobilité et grâce au développement parallèle des téléphones portables, que se sont développés des ordinateurs de poches comme les assistant personnel numérique.

C'est alors qu'est née l'idée d'intégrer l'accès et la navigation sur Internet depuis un simple téléphone portable. Le WAP est donc le protocole permettant de faire converser le monde de la téléphonie mobile et le monde de l'Internet.

L'objectif de notre travail est de mettre au point un système de Téléconsultation pour interconnecter les hôpitaux du territoire national (Algérien) entre eux, permettant un échange de données entre médecins distants via la technologie WAP en vue de diagnostic ou de thérapeutique. L'application réalisée est fondée sur une architecture WAP client-serveur composée d'un module d'accès et de traitements des données du patient, un module de chat, un Forum, un module de messagerie.

Notre application sera donc principalement basée sur l'utilisation des différentes technologies dédiées à la mobilité tel que le protocole WAP, et son langage de marquage le WML, des outils WEB tel que le serveur Web, ainsi que les SGBD et les interfaces d'accès distant aux bases de données.

Ce mémoire est organisé en six chapitres et se présente comme suit :

Le premier chapitre est une présentation des concepts de la télémédecine et de la téléconsultation médicale.

Le deuxième chapitre est une présentation du protocole WAP (Wireless Application Protocol), standard international de communication dans les grands réseaux mobiles.

Le troisième chapitre est une description des langages de développement des applications WAP à savoir le WML et le WMLScript.

Le quatrième chapitre est une description des interfaces WAP/base de données existantes et plus précisément le php (Personal Home Page).

Les deux derniers chapitres sont consacrés à la plate forme que nous avons développée.

La première partie du chapitre V penche plus en détail sur la conception des bases de données de la plate forme, Tandis que la deuxième partie décrit les différentes parties qui composent la plate forme et les interactions entre elles.

Le chapitre VI fait état de l'implémentation de la plate forme. D'abord par la présentation de l'environnement incluant les différents outils choisis pour la réalisation et la mise en œuvre de l'application. Ensuite par la description des interfaces de la plate forme, c'est à dire le système tel qu'il sera vu par les différents utilisateurs mobiles.

Et nous terminerons par une conclusion sur le travail réalisé.

Chapitre I

La téléconsultation médicale

Introduction :

La télémédecine a été pratiquée depuis longtemps : ce n'est pas une nouvelle technique, certains pensent même qu'elle est née avec l'introduction du téléphone. Les années 60 ont connu un immense progrès où la télévision devait devenir le nouveau média incontournable de la télémédecine.

Aujourd'hui la nouvelle technologie de l'information et de la communication génère un développement de télémédecine via l'Internet dans un temps où la Santé est devenue une des plus fortes valeurs sociales.

I- La télémédecine :**I-1- Définitions :**

Le mot télémédecine, composé de la racine grecque « télé » signifiant « à distance », et de terme « médecine », se définit comme une pratique de la médecine à distance. De nombreux auteurs définissent comme étant l'union des télécommunications et de la médecine. C'est la définition littéraire de la télémédecine, mais il existe d'autres définitions parmi lesquelles :

La télémédecine est « l'accès rapide au partage de l'expertise médicale distantes par les moyens des télécommunications et les technologies de l'information ». Celles-ci sont définies comme étant les technologies qui permettent le stockage, la manipulation et la communication de l'information sous forme de texte, image, audio et vidéo (multimédia). [IND 99]

La télémédecine se définit aussi comme étant la pratique médicale en utilisant l'audiovisuel et les communications des données. Elle comprend : les soins médicaux, le diagnostic, la consultation et le traitement ainsi que l'enseignement et le transfert des données médicaux.

La télémédecine consiste à « établir un diagnostic d'un patient se trouvant n'importe où dans le monde en combinant les télécommunications et l'expertise médicale »

La télémédecine est définie aussi comme : « l'emploi des réseaux de télécommunication pour relier les spécialistes des soins de santé ou hôpitaux, clinique, médecins de soins primaires et patients afin de fournir à distance des diagnostics, traitements consultation et formation continue, par exemple : Télécardiologie, Télédématologie, Télépsychiatrie et Téléradiologie ». [HOR 99]

I-2- Objectifs de la télémédecine :

Les médecins spécialistes ne sont pas systématiquement présents dans tous les sites hospitaliers. Une urgence dans un hôpital périphérique, par exemple, devra trouver la compétence pour l'établissement de diagnostic. De plus en plus les spécialistes font appel à la télémédecine qui permet une meilleure prise de décision et donc un diagnostic plus sur et rapide.

Pour un cas grave, le service d'urgences de l'hôpital périphérique dispose de trois possibilités :

[MEU 96]

- Faire venir le spécialiste, ce qui est peu probable.
- Transférer le patient avec les risques que cela comporte, par des transports routiers ou aériens très coûteux. A son arrivée dans le service concerné, le malade ou l'accidenté fera à nouveau l'objet d'un examen plus précis.
- Transmettre le dossier médical du patient le plus rapidement possible pour établir le diagnostic par téléphone avec le spécialiste. Cette possibilité semble présenter plus d'avantages d'où la nécessité de télémédecine.

On peut résumer les objectifs de la télémédecine dans les points suivants :

- Offrir un cadre interactif d'échanges en Information médicale, télédiagnostic, téléthérapeutique et en recherche médicale.
- Permettre l'accès aux réseaux Internet et de messagerie électronique.
- Encadrer à distance les professionnels de la santé tout en assurant leur formation continue.
- Faciliter la collaboration entre les médecins en permettant l'accès à un conseil plus qualifié lors de la découverte d'une problématique qui dépasse le cadre de savoir du médecin et en créant des passerelles communes de communication et de transmission de savoir entre les médecins.
- Eviter les déplacements inutiles des patients dont l'état ne nécessite peut-être pas un transfert.
- Limiter la redondance des actes, des examens en permettant à chaque membre de la chaîne de suivi médical du patient d'accéder à tout moment à son dossier.
- Meilleur contrôle des dépenses de santé car chaque intervention est informatisée donc comptablement analysable.
- Améliorer la transmission du savoir intra ou inter services, en utilisant une banque de données disponible en temps réel faisant état de l'historique des interventions des acteurs de la télémédecine, de conseils théoriques et d'avis empiriques sur tout problème rencontré.

- Contribuer à la formation continue des médecins en multipliant les possibilités de contact professionnel via le réseau, en utilisant les outils informatiques pour développer le téléenseignement.
- Satisfaire des problèmes d'isolement géographique de population où il est parfois indispensable de parcourir un certain nombre de kilomètres pour obtenir un simple diagnostic.
- Améliorer l'aide aux pays sous développés ou émergents qui n'ont pas une structure hospitalière suffisamment importante pour couvrir tous leurs besoins.

I-3- Typologie de la télémédecine :

La télémédecine englobe l'ensemble de pratiques médicales et peut être caractérisé par le type de l'information envoyé et par les moyens utilisés pour les transmettre. La disponibilité de l'information est un point clé pour un bon management médical. Pour cela l'information peut être représenté sous trois formes : données, audio et images.

Modalité	Exemples
VOIX	Consultation téléphonique (comprise comme une véritable composante de la prestation de soins médicaux)
DONNÉES	Utilisation du courrier électronique et des services télématiques pour des problèmes isolés ou dans le cadre de groupes d'intérêts, d'associations de recherche, etc.
VOIX-DONNÉES	Intégration de l'informatique et de la téléphonie
VOIX-DONNÉES-IMAGES EN TEMPS DIFFÉRÉ	Dossier médical électronique, exploitation collective d'images cliniques
VOIX-DONNÉES EN TEMPS RÉEL	Télesurveillance médicale. Par exemple, surveillance à distance de signes vitaux (battements de cœur...), contrôle de la conformité aux indications posologies, surveillance des besoins des personnes âgées (systèmes d'alarme)
VOIX-DONNÉES-IMAGES EN TEMPS RÉEL	Télémédecine "classique" - la "consultation virtuelle"

Tab.I.1 : Modalités technologiques de la télémédecine

I-4- Les services de la télémédecine :

La Télémédecine est l'exercice de la médecine à distance par la communication audiovisuelle et le transfert de données interactives par les moyens de réseaux de télécommunication, téléphone, lignes spécialisées, satellites. La télémédecine permet à des médecins d'échanger des informations entre eux, d'effectuer des consultations ou d'analyser et interpréter des examens complémentaires, en particulier de l'imagerie médicale.

Elle couvre les domaines suivants :

- Téléconsultation.
- Télédiagnostic.
- Téléassistance.
- Téléchirurgie/Télémanipulation.
- Téléexpertise Télé imagerie.
- Téléopération.
- Télésurveillance/Téléobservance.
- La téléformation.

I-5- Technologies de la télémédecine :

La mise en place d'un système de télémédecine doit relever de deux aspects : un aspect de transmission des données par les technologies des télécommunications et un autre qui est celui de l'acquisition et stockage des données médicales par les technologies de l'information.

I-5-1- Technologies de télécommunications :

Les applications de télémédecine peuvent être caractérisées comme exigeant une transmission à largeur de bande étroite, moyenne ou large. Le choix de réseau inclut la téléphonie de base, une ligne numérique terrestre, le téléphone cellulaire, les satellites et les réseaux à largeur de bande tels que les ATM (Asynchrone Transfer Mode).

I-5-2- Technologie de l'information :

Les technologies de l'information sont utilisées en télémédecine pour acquérir, stocker, manipuler et communiquer l'information sous forme de vidéo, audio ou donnée

Un système de télémédecine doit comprendre :

Un hardware comprenant un ordinateur, une caméra ou un scanner... Dans le cas d'un site WAP, il faut un téléphone portable qui supporte la technologie WAP.

Un software incluant des logiciels de gestion des transmissions, de traitement d'image, de compression et de stockage tel que les bases de données.

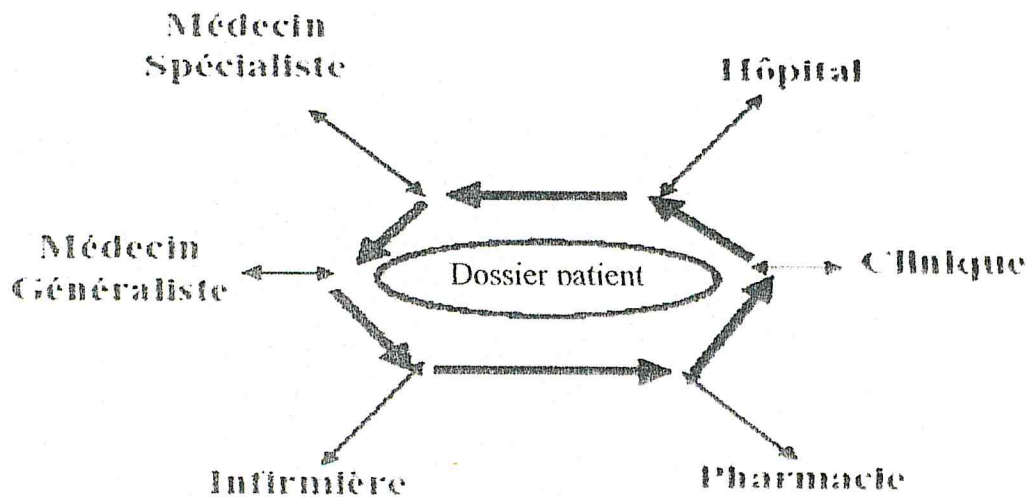
Un système de télémédecine possède deux modes d'opérations :

Temps réel où le patient est présent avec le médecin dans un site distant du spécialiste.

Temps différé où les informations sont transmises électroniquement au spécialiste et la réponse sera donnée plus tard.

II- La téléconsultation médicale :

C'est la consultation médicale à distance. Les réseaux de télémédecine offre la possibilité de communication entre les médecins et les autres prestataires de soins de santé. La consultation dans ce cas peut se faire en temps réel par téléphone ou visioconférence ou en déferé en utilisant par exemple la messagerie électronique, les forums, chat... L'exemple le plus simple de cette application est l'utilisation de téléphone où le médecin peut demander l'avis d'un confrère. Un grand nombre de site WEB fournissent actuellement des services de téléconsultation.



II-1- La messagerie électronique :

La messagerie électronique est un outil désormais bien connu de tous et qui est devenu quasiment indispensable. Elle permet la transmission asynchrone de messages, vers un ou plusieurs correspondants, avec la possibilité de joindre un document au texte du message.

La messagerie électronique est très nécessaire dans un système de téléconsultation médicale, elle permet aux médecins de communiquer entre eux ou avec ces patients à n'importe quel moment.

II-2- Les forums :

Les forums permettent de gérer les échanges asynchrones de messages relatifs à un même sujet, au sein d'un groupe de plusieurs apprenants.

Les forums de discussion éliminent les contraintes horaires. Ils permettent aux médecins et patients de poser des questions et de faire part de ses difficultés. En formation, un forum peut, par exemple, servir d'outil de discussion et d'échanges pour une étude de cas, un jeu de rôles, une résolution problèmes.

II-3-La conversation textuelle synchrone (Chat) :

La conversation textuelle synchrone ou IRC (Internet Relay Chat) permet la transmission synchrone de textes courts entre les participants d'une même conférence. Il est généralement possible d'envoyer soit des messages publics, vers tous les participants, soit des messages privés, visibles seulement par leur(s) destinataire(s).

III- La téléconsultation médicale via le WAP :

C'est la consultation médicale à distance. Les réseaux de télémédecine offre la possibilité de consultation entre les médecins et les autres acteurs de soins de santé.

Dans le contexte actuel, beaucoup de patients se retrouvent hospitalisés et doivent ensuite être suivis de manière régulière en milieu médicalisé. Leur état de santé pourrait leur permettre de reprendre une activité normale, mais il est encore trop instable pour les dispenser de suivi. Beaucoup d'autres sont sujets à des malaises, pensons par exemple aux personnes âgées. La première catégorie de patients est hospitalisée, nécessite un grand déploiement de moyens humains et financiers. Les autres ne sont tout simplement pas suivis, l'hospitalisation étant trop contraignante et le suivi à domicile trop cher. Cette situation entraîne aujourd'hui des pertes d'argent considérables. La technologie WAP a résoudre ce problème et offre une solution peu coûteuse, et très confortable pour les patients. En effet, ils pourraient bénéficier de la sécurité d'un suivi médical, sans l'inconvénient de la prise en charge hospitalière, il suffit d'un téléphone portable compatible WAP et abonnement.

Pour qu'une application de téléconsultation médicale fonctionne parfaitement il faut vérifier les conditions suivantes :

- Le téléphone du patient doit être compatible WAP.
- Le médecin devra s'équiper d'un téléphone portable pour être joignable et qui sera capable de se connecter à Internet. En effet, en cas d'alarme, la décision prise devra souvent être rapide pour être efficace. L'accès immédiat à Internet est donc indispensable.

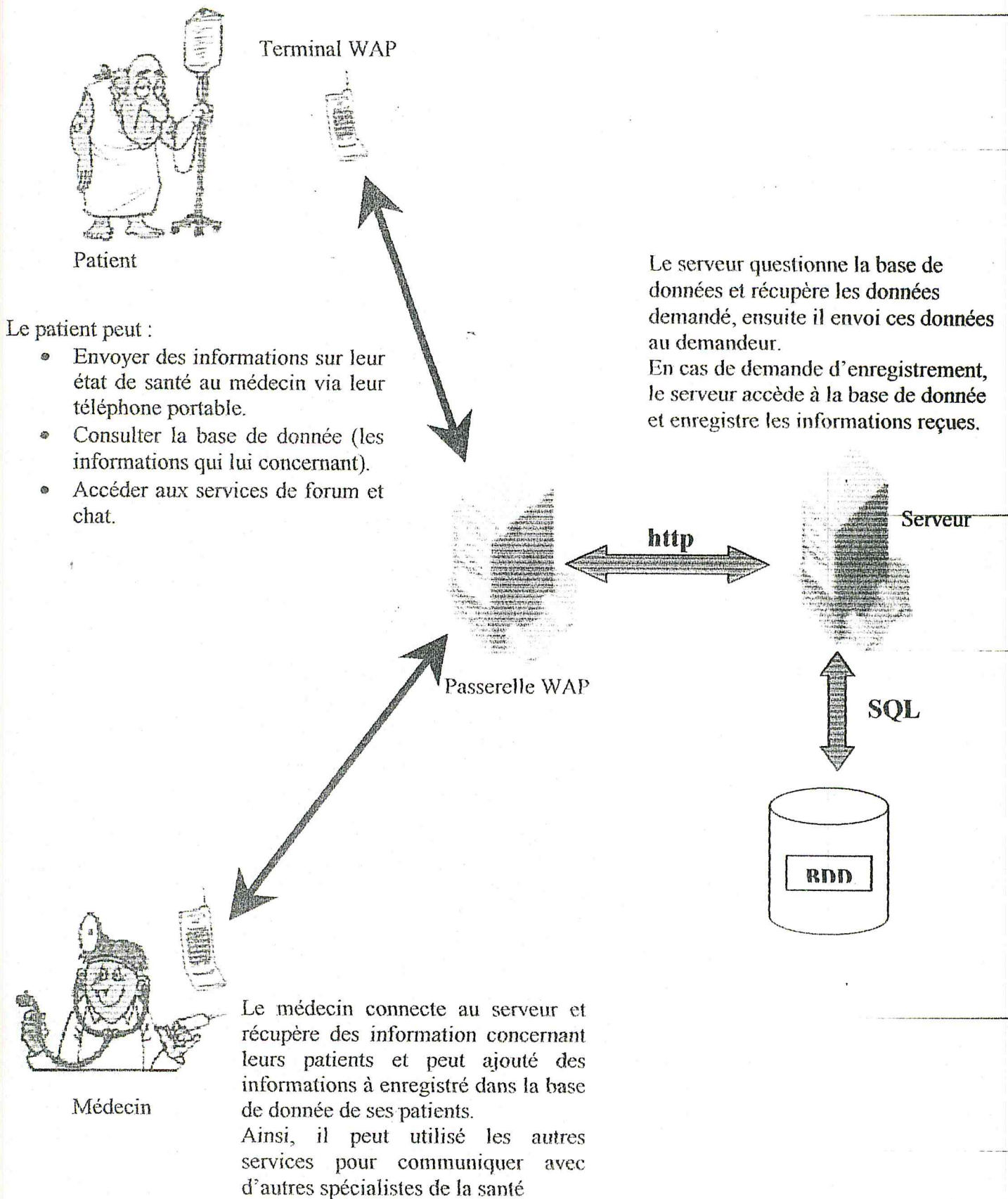


Figure.I.I : schéma générale d'un système de téléconsultation médicale via le WAP

III-1- Accès au réseau :

Dans le cas des réseaux sans fil, l'accès au réseau n'est pas forcément garanti. Un tunnel, une position géographique défavorable, des conditions météorologiques particulières, et les transmissions peuvent être réduites, ou tout simplement impossibles. L'application doit donc être conçue pour pallier ces inconvénients. Elle doit être suffisamment intelligente pour permettre la re-transmission des données si le réseau est indisponible, pour permettre le stockage des informations entre temps, ou l'annulation de leur transmission si les informations non traitées en temps utiles sont obsolètes.

III-2- Communications :

Dans le cadre des communications sans fil, tout le monde partage la même largeur de bande, et même si celle-ci est importante, elle reste à prendre en considération. Il faut prendre en compte les débits. Ils changent en effet en fonction du protocole utilisé.

III-3- Interface utilisateur:

Les applications portables s'exécutent sur des appareils aux dimensions réduites, aussi bien pour les unités d'entrée que de sortie. Le programmeur doit donc prendre en compte ces aspects. L'affichage est généralement limité à quelques centimètres d'écran LCD TFT, avec une résolution jusqu'à 640*480 (VGA). Les conditions de lecture sont souvent défavorables, notamment à cause de la luminosité. La saisie des champs sur les appareils portables est laborieuse, en raison de l'absence de clavier. Il faut réduire au minimum ces entrées et proposer des alternatives, comme des boîtes à cocher, des listes cliquables.

Conclusion :

Nous concluons que la santé est devenue une valeur importante dans la politique de la plus part des pays en développement, mais l'infrastructure médicale dans ces pays reste insuffisante à cause de leur pauvreté.

Le développement des moyens de communication a aidé ces pays à résoudre ce problème avec le moindre possible de coût et de temps.

Parmi ces technologies nous citons le WAP (Wireless Application Protocol), qui offre la possibilité de faire des diagnostics à n'importe quel moment et à n'importe quel endroit via l'Internet mobile sans besoin d'un grand matériel, juste un téléphone portable.

Dans le chapitre qui suit, nous allons définir la technologie WAP et ces différentes architectures.

Chapitre II

La technologie WAP (Wireless Application Protocol)

Introduction :

Depuis quelques années, on a pu assister à une véritable révolution du monde des télécommunications. En effet, de nouvelles normes de communication ont fait leur apparition, on peut dorénavant envisager des applications que l'on réservait au monde de la science fiction il n'y a pas si longtemps.

Les experts du marché mondial croyaient qu'Internet aurait le taux de pénétration le plus rapide qui n'ait jamais existé, de fait qu'ils ont vu le nombre de connectés doubler tous les 6 mois. Cette explosion du marché était en fait négligeable comparée au développement du marché des télécommunications numériques sans fil (GSM). En l'espace de 5 ans, le parc de mobiles est passé de quelques milliers à plusieurs dizaines de millions. L'accroissement n'est pas pour autant fini puisqu'on s'attend à voir le nombre de portables passer à plus d'un par personne, chacun ayant un téléphone personnel et professionnel. Dans ce contexte, il n'est pas difficile de comprendre que les entreprises veulent profiter de cette manne d'utilisateurs en leur apportant un contenu directement sur leur mobile. C'est l'objectif du WAP... [PIC 00]

Le WAP est né de l'alliance en 1997 de plusieurs grands groupes regroupant les constructeurs de mobiles (Nokia, Ericsson, Motorola,...), des opérateurs en téléphonie mobile et des multinationales (phone.com, Microsoft...) au sein du WAP forum. Celui-ci est chargé de valider les spécifications techniques proposées par les sociétés participantes. La version 1.0 du protocole WAP a été publiée en Mai 1998.

L'objectif principal du standard WAP est de fournir un environnement complet (protocoles + langages de développement d'application + spécification d'un navigateur et de passerelles) pour l'utilisation des services de données sur des terminaux dits légers « petits écrans, réseau sans fil à faible débit et faible qualité de service, taille mémoire limitée,... ».

I- Le protocole WAP (Wireless Application Protocol) : [PIC 00]

Le WAP ou le protocole d'application mobile a pour but de fournir un accès Internet à des terminaux comme les téléphones portables et les assistants personnels numériques comme les pockets pc.

Au départ ce protocole a été créé à l'initiative d'entreprise comme Nokia et Ericsson il est basé sur des technologies Internet existantes comme le XML et IP.

Techniquement le lien entre le réseau téléphone mobile et le réseau Internet est effectué par une passerelle ou « Gateway » qui a comme fonction d'assurer la communication entre les terminaux WAP et les serveurs web. Le mode d'échange des données du protocole WAP est basé sur le modèle client-serveur utilisé par le web.

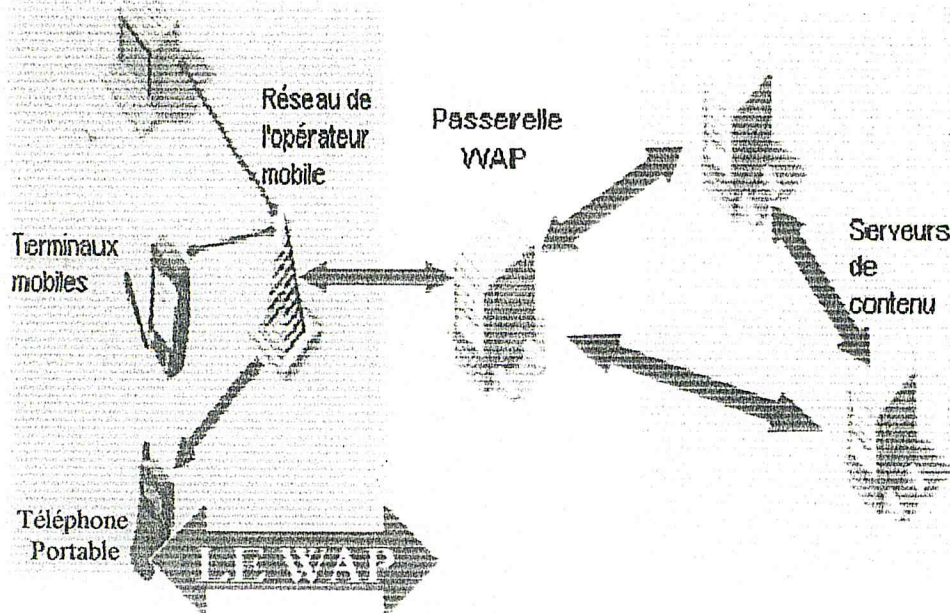


Figure.II. 1 : Architecture matérielle générale du WAP

Le WAP forum a défini l'architecture du protocole WAP comme une hiérarchie des services et de protocoles qui fonctionnent en couches superposées. [www2]

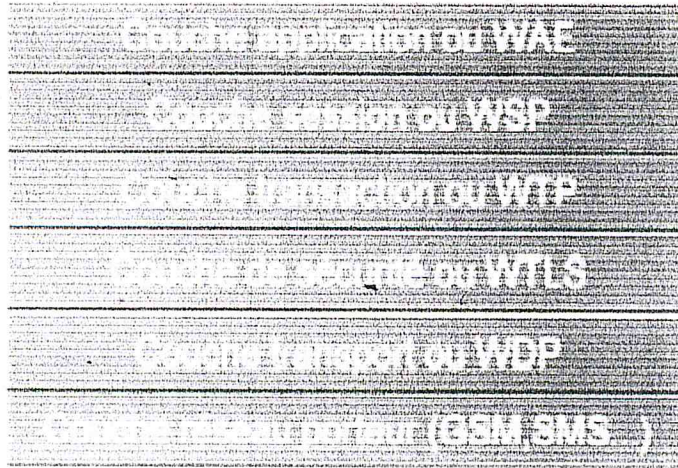


Figure.II.2 : Les différentes couches du WAP.

1. La couche réseau porteur :

Le WAP utilise le réseau mobile comme réseau porteur, il y a bien entendu la norme GSM (Global System for Mobile Communication) mais les autres réseaux de transmission de données comme le SMS (Short Message Service) et CDMA (Code Division Multiple Access) sont compatibles.

2. La couche transport WDP (Wireless Datagram Protocol):

Assure la communication (le transport) entre le Gateway et le terminal WAP. Le protocole utilisé est Wireless Datagram Protocol (WDP).

3. La couche sécurité WTLS (Wireless Transport Layer Sécurité):

Cette couche a été développée pour l'employer avec la couche transport, elle assure la sécurité des échanges de données grâce au protocole Wireless Transport Layer Sécurité (WTLS).

4. La couche transaction WTP (Wireless Transaction Protocol):

Gère le déroulement de la transaction elle définit donc la fiabilité du service, elle contrôle trois types de transactions :

- Invocation non fiable sans message de résultat.
- Invocation fiable sans message de résultat.
- Invocation fiable avec message de résultat.

Elle utilise le protocole Wireless Transaction Protocol (WTP) qui garantit la bonne conclusion des transactions sous forme de requêtes - réponses.

5. La couche session WSP (Wireless Session Protocol) :

WSP fournit à la couche application de WAP (WAE) une interface logique pour deux services orientés. Le premier est un service orienté connexion qui opère au dessus de la couche transaction (WTP). Le second est un service non connecté qui opère sur un service datagramme (WDP) sécurisé ou non.

WSP consiste en des services spécialisés pour des applications de navigation comportant les fonctionnalités suivantes :

- ✓ Support du protocole http 1.1.
- ✓ Gestion des sessions utilisateur.
- ✓ Suspension et destruction des sessions avec possibilité de migration de la session.
- ✓ Possibilité de fournir des données de type push.
- ✓ Négociation des caractéristiques du protocole.

Les protocoles dans la famille WSP sont optimisés pour les réseaux de faible bande passante avec des temps de latences relativement longs. WSP a été conçu pour permettre à Proxy WAP d'interconnecter un client WSP à un serveur http.

6. La couche application WAE (Wireless Application Environment) :

Qui utilise le protocole Wireless Application Environment (WAE), elle définit l'environnement WAP dans le format des fichiers en WML (Wireless Markup Language), format des images, et des exécutable comme le WML-Script.

I-1- Le modèle World Wide Web et le modèle WAP:**I-1-1- Le modèle World Wide Web (WWW):**

L'architecture du World Wide Web fournit un modèle de programmation simple et puissant. Les applications et le contenu sont présentés dans des formats de données standard et sont lues par des applications appelées « navigateurs ». Un navigateur est une application réseau de type client/serveur, c'est-à-dire qu'il envoie des requêtes pour obtenir des objets auprès d'un serveur sur le réseau, et que celui-ci lui répond en envoyant les objets demandés en utilisant des formats de données standard.

Le World Wide Web fournit un grand nombre de mécanismes standard :

- **Modèle de nommage standard** : Tous les serveurs et tous les documents sont accessibles à travers une adresse appelée URL (Uniform Resource Locator).
- **Définition du contenu** : Tout objet sur le World Wide Web se voit attribuer un type permettant aux navigateurs d'interpréter correctement leur contenu.

- **Formats de contenu standard :** Tous les navigateurs supportent un certain nombre de formats de contenu standard, comme HTML (HyperText Markup Langage), Java script, ASP (Active Server Page) ...
- **Protocole standard :** Ces protocoles permettent à n'importe quels navigateurs de communiquer avec n'importe quels serveurs. Le protocole le plus utilisé est http (HyperText Transport Protocol).

Cette architecture permet aux utilisateurs d'atteindre facilement un très grand nombre d'applications et de services fournissant du contenu. Par ailleurs, elle permet également aux développeurs d'application de créer très facilement des applications et des services fournissant du contenu à un très grand nombre d'utilisateurs.

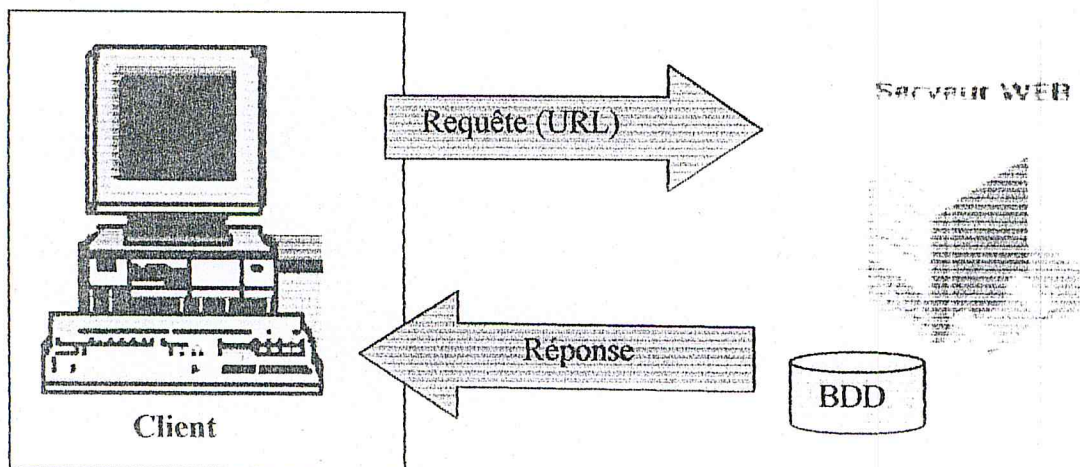


Figure.II.3: Le modèle World Wide Web.

I-1-2- Le modèle WAP :

Le modèle WAP est similaire à celui du World Wide Web. Ceci apporte un grand nombre d'avantages aux développeurs d'applications, en leurs permettant de réutiliser leurs outils et leurs modèles de programmation sur une architecture connue et stable. Des extensions, comme les langages WML et WMLScript, permettent de prendre en charge les caractéristiques spécifiques aux environnements des réseaux mobiles. Les standards existants ont été autant que possibles utilisés, ou utilisés comme point de départ pour la technologie WAP.

Les applications et les contenus WAP sont écrits dans des formats similaires aux formats en usage sur le World Wide Web. Ces contenus sont transportés en utilisant des protocoles de communication standard du World Wide Web. Un micro navigateur dans le terminal mobile gère l'interface utilisateur vis-à-vis d'un navigateur Web.

L'architecture WAP définit un certain nombre de composants standard qui permettent la communication entre les terminaux mobiles et les serveurs Web :

- **Un modèle de nommage standard** : utilisant les URLs du World Wide Web pour localiser les contenus et les applications WAP.
- **Un type spécifique a été attribué à tous les types de contenus WAP** : Ceci permet aux navigateurs WAP de gérer correctement les contenus reçus en se basant sur leurs types.
- **Un jeu de protocole de communication standard** : permet aux périphériques WAP de communiquer avec un serveur Web du réseau.

Ainsi le WAP met en œuvre une passerelle permettant d'interconnecter le monde du réseau mobile avec celui du World Wide Web. Cette passerelle comprend généralement les fonctionnalités suivantes:

- Une passerelle de protocoles traduit les requêtes en provenance de la pile de Protocole WAP (WSP, WTP, WTLS et WDP) vers la pile de protocoles World Wide Web (HTTP et TCP/IP).
- Un encodeur/décodeur de contenus traduit le contenu WAP en un format codé compact afin de réduire la taille des datas au travers du réseau mobile.

Cette infrastructure assure aux utilisateurs des terminaux mobiles de pouvoir utiliser une grande variété d'applications et de contenus.

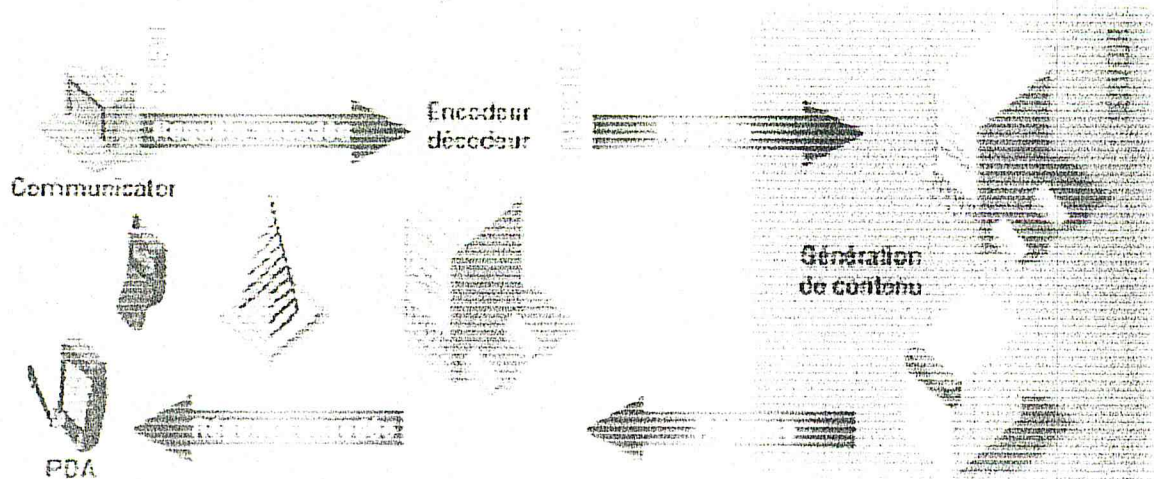


Figure.II.4 : Le modèle WAP

I-2- Les différents composants de la technologie WAP : [PIC 00]

Pour assurer le transport des services WAP aux terminaux mobiles il faut intégrer quatre briques technologies qui sont :

- Serveur Web et applicatif.
- Passerelle WAP.
- Réseau de l'opérateur.
- Navigateur WAP.



I-2-1- Le serveur Web :

Celui-ci a pour but de fournir des documents WML. Les technologies mises en oeuvre sont identiques aux techniques Internet. C'est donc un serveur HTTP qui diffuse un contenu statique ou dynamique. La seule différence avec Internet est le format de diffusion des pages en WML et non plus en HTML.

I-2-2- La passerelle WAP (gateway)/ serveur WAP:

La passerelle WAP est l'élément qui différencie le WAP du Web, elle assure deux fonctionnalités principales :

- Elle réalise la connexion entre le réseau informatique et le réseau téléphonique sans file.
- Elle code et décode les requêtes et les réponses entre le terminal WAP et le serveur web.

Cette dernière fonctionnalité est spécifique au WAP. Afin de réduire la taille des données transmises, les requête et les réponses sont encodées dans un format binaire compact, les fichiers au format WAP sont compiler par la passerelle avant d'être envoyé vers le terminal qui se charge de les décoder puis de les interpréter.

Certain constructeurs intègrent un serveur applicatif dans leurs passerelle afin d'offrir des services complémentaires, dans ce cas on parle de serveur WAP, qui produit lui même les documents qui seront transmis au téléphone mobile.

Notons que Nokia a mis sur le marché un produit appelé "serveur WAP" qui est en réalité une passerelle WAP associée à un serveur http. La partie passerelle s'occupe de la connexion, le serveur s'occupe de fournir le contenu.

Il existe plusieurs passerelle WAP dans le marché on cite parmi elles :

- Audicode WAP Server.
- DSR Catapult WAPGateway.
- Ericsson WAPGateway/Proxy.
- EXa Link eXa.Flow.

- Infinite WAPLite.
- Kannel.
- Nokia WAP Server.
- Phone.com UP.Link WAP server.
- RealWow REALgew

I-2-3- Le réseau de l'opérateur mobile :

L'opérateur fournit les services de transmission radio nécessaires au fonctionnement des mobiles. Toutes les transmissions de données WAP utilisent donc le réseau d'un opérateur pour atteindre le mobile.

Les normes de transmissions des données dans les réseaux de téléphonie mobile ont connu déjà plusieurs générations qui sont :

1ère génération : systèmes analogiques

- AMPS (Advanced Mobile Phone System) 1983
- TACS (Total Access Communication System) 1985
- NMT (Nordic Mobile Téléphone) 1986

2ème génération :

- GSM (Global System for Mobile Communication) 1991 : 9.5kbit/s
- CDMA (Code Division Multiple Access)

2ème vers 3ème génération :

- GPRS (General Packet Radio Service) 2000 : 171.2kbit/s
- EDGE (Enhanced Data for GSM Evolution) 2001 : 473.6kbit/s

3ème génération:

- UMTS (Universal Mobile Telecommunication System) 2002 : jusqu'à 2Mb/s

I-2-4- Le navigateur (sur le téléphone portable ou PDA) :

Le navigateur se charge de décoder les informations transmises par la passerelle WAP afin d'afficher les pages pour l'utilisateur, il joue le même rôle qu'un navigateur Web tel qu'Internet Explorer ou Netscape navigateur.

Notons qu'il faut prendre en considération lors du développement d'un site WAP les contraintes suivantes sur les terminaux WAP :

- Les périphériques offrent des possibilités de saisies utilisateurs très limités, un téléphone ne permet que de saisir aisément des chiffres, plus difficilement des lettres et éventuellement quelques autres touches de fonctions spécialisées.

- Les périphériques possèdent de très faibles capacités de traitement CPU et possèdent généralement une très faible quantité de mémoire RAM.

Il existe plusieurs navigateurs de téléphones mobiles disponibles sur le marché on cite parmi eux :

- Up.Browser de phone.com.
- Navigateur 7110 du Nokia.
- Microsoft mobile explorer.
- WAP man.
- Navigateur Ericsson pour Palm pilot.

I-3- Les différentes implémentations d'une plate forme WAP : [PIC 00]

Les architectures suivantes présentent les différentes possibilités d'intégration des briques technologiques qui composent une plate forme WAP :

I-3-1 Passerelle WAP hébergée chez l'opérateur de téléphonie mobile:

Cette architecture est la solution par défaut proposé par les opérateurs de téléphonie mobile pour accéder au service WAP, l'opérateur héberge dans ce cas le service d'accès distant (accès réseau IP) et la passerelle.

Les services WAP référencés sur les portails des opérateurs utilisent ce type de solution dont la mise en œuvre est simple.

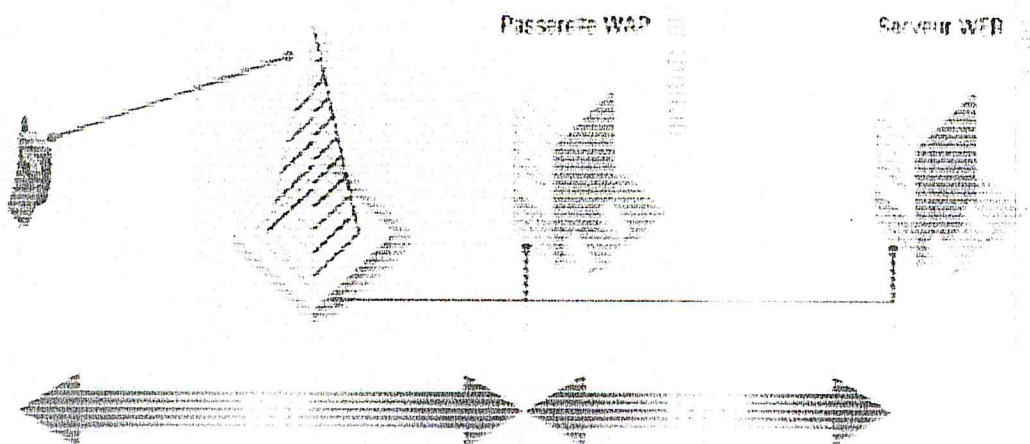


Figure.II. 5 : Passerelle WAP hébergée chez l'opérateur

I-3-2- Passerelle WAP hébergée chez un fournisseur d'accès :

I-3-2-1- Avec hébergement des serveurs Web :

Cette solution permet d'être indépendante des opérateurs de téléphonie. L'utilisateur compose ainsi directement le numéro du fournisseur d'accès qui dispose en interne de modems et d'une passerelle WAP.

Le fournisseur d'accès WAP joue le rôle d'un opérateur télécom spécialisé dans la connexion à Internet via le WAP.

Cette solution permet de s'affranchir de certaines contraintes rencontrées chez les opérateurs de téléphonie mobile.

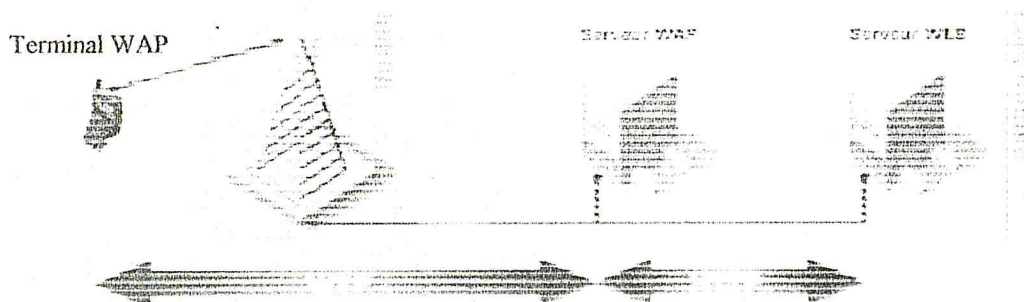


Figure.II. 6 : Passerelle WAP hébergée chez un fournisseur d'accès avec hébergement de serveur Web

I-3-2-2- Sans hébergement des serveurs Web :

Dans cette architecture, les fournisseurs d'accès WAP jouent le même rôle que les fournisseurs d'accès Web. Leur mission est ici de faciliter l'accès à des services WAP hébergés sur Internet via leur passerelle. Cette architecture permet aux clients de garder l'entière maîtrise des services hébergés sur leurs serveurs Web. Dans tous les cas, le client doit porter une grande attention à la politique de refacturation des communications.

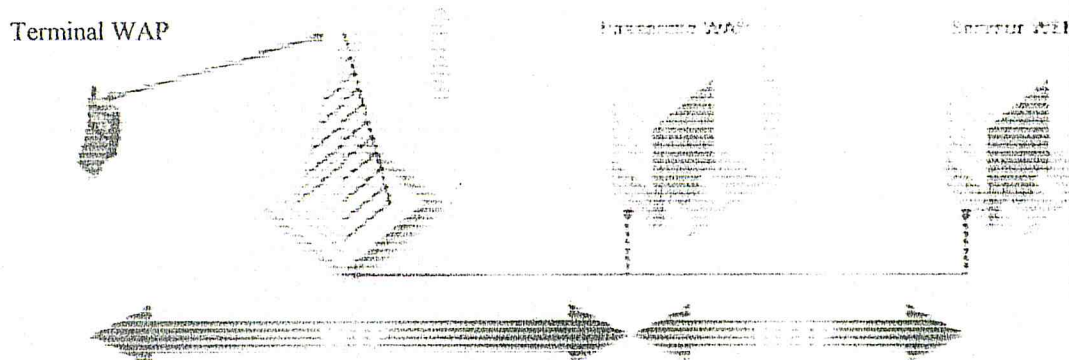


Figure.II. 7 : Passerelle WAP hébergée chez un fournisseur d'accès Sans hébergement de serveur Web

I-3-3- Passerelle WAP en interne :

L'intérêt de cette solution est de garder la maîtrise des connexions WAP. Les applications WAP peuvent ainsi accéder à certaines informations spécifiques qui ne sont pas visibles depuis le serveur Web. Les accès aux contenus et donc aux données de l'Intranet peuvent ainsi être contrôlés plus efficacement

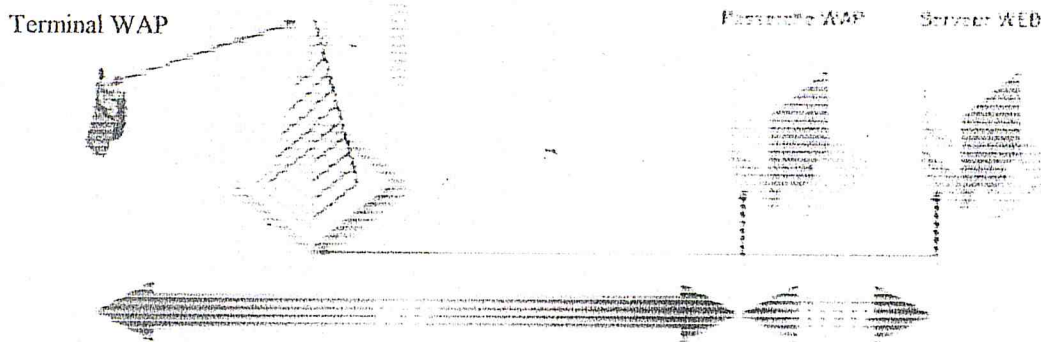


Figure.II. 8 : Passerelle WAP en interne

I-4- Modes de Communication dans un réseau WAP (le PULL et PUSH) :

Pour accéder à l'information l'utilisateur peut aller la chercher directement : on parle de « pull ». Il peut aussi (grâce à un abonnement), la recevoir sans demande particulière : on parle de « push ».

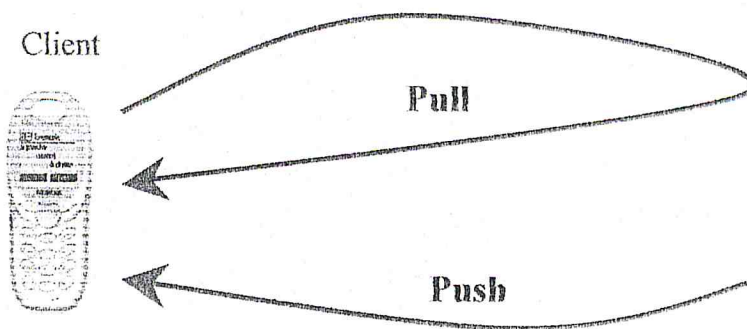


Figure.II.9 : Le pull et le push

I-4-1- Le pull :

Correspond à celui mis en œuvre sur le Web. L'utilisateur saisit dans son navigateur l'adresse du site qu'il souhaite consulter. La consultation de pages WML à l'aide d'un navigateur compatible WAP utilise aussi ce modèle.

Le schéma suivant illustre cet échange :

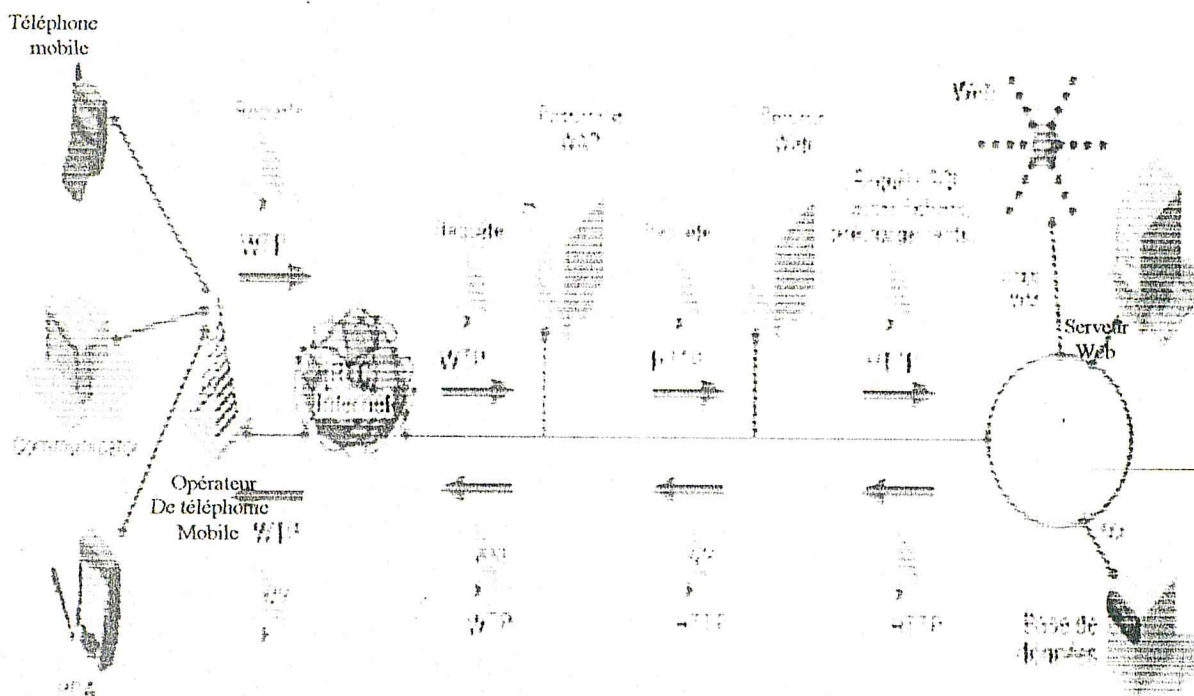


Figure.II.10 : Mécanisme d'échange de données selon la technologie pull.

Tout d'abord les adresses de la passerelle et de serveurs Web employé sont préalablement configurés dans les paramètres de connexion du terminal WAP.

Si un utilisateur veut consulter un site WAP il doit saisir son adresse dans le navigateur de terminal WAP. La requête générée par l'utilisateur va être transportée vers le site recherché au format WTP jusqu'à la passerelle qui assure la conversion de cette requête du format WTP vers le format http qui est le protocole utilisé dans le Web.

Le serveur Web reçoit la requête http génère en retour l'information en WML issue des contextes applicatifs (autres serveurs Web, serveurs applicatifs, base de données). Puis il envoie les données en format WML véhiculé au format http vers la passerelle, ce dernier assure la conversion des données http au format WTP. Puis il envoie ces données au terminal WAP.

Si le serveur Web fournit du contenu WAP (des fichiers WML par exemple), la passerelle WAP le récupère directement. Par contre si le serveur Web fournit un contenu World Wide Web (documents HTML par exemple), un filtre est utilisé pour traduire le contenu World Wide Web en contenu WAP. Par exemple le filtre transformerait un document HTML en document WML.

Le serveur sans fil de l'application de téléphonie (WTA) est un serveur d'origine ou de Gateway qui répond aux demandes du client WAP directement. Le serveur de WTA est utilisé pour permettre d'accéder WAP aux dispositifs de l'infrastructure de télécommunications sans fil du fournisseur de réseau.

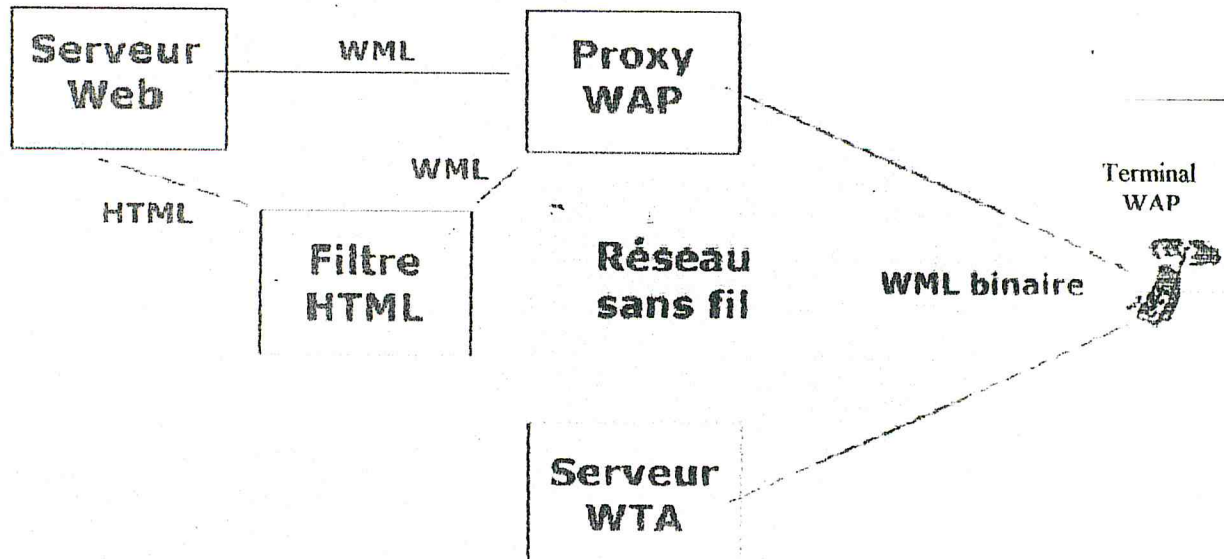


Figure.II. 11 : Filtre HTML

Le serveur WTA (Wireless Telephony Application)

Le serveur WTA peut être vu comme un serveur Web délivrant un contenu demandé par un client. A l'instar d'un navigateur Web, un agent utilisateur WTA utilise des URLs pour référencer des contenus sur le serveur WTA.

Une URL peut également référencer une application sur un serveur Web qui sera exécutée. Ce genre d'applications peut servir à gérer un très grand nombre de tâches comme par exemple générer du contenu dynamique, ou interagir avec des périphériques externes.

Un serveur WTA peut également utiliser ce concept. En référençant des applications sur un serveur WTA, il est possible de créer des services qui utilisent des URLs pour interagir avec le réseau mobile et d'autres entités (un système de messagerie vocale par exemple). Le concept de référencement d'application sur un serveur WTA fournit un modèle simple mais puissant pour intégrer des services à valeur ajoutée qui s'exécutera localement sur le client WAP.

I-4-2- Le push :

Le principe de fonctionnement du push consiste à utiliser une nouvelle brique technologique : la passerelle push Proxy qui peut être incorporée directement dans un serveur WAP.

Le schéma suivant montre cette technologie :

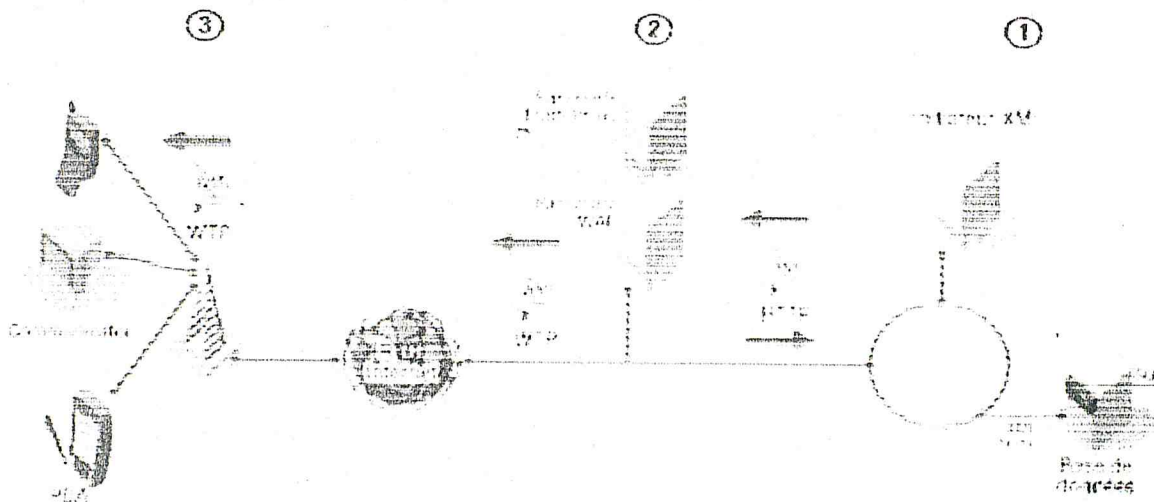


Figure.II.12 : Mécanisme d'échange de données selon la technologie push.

1. Le serveur initiateur envoie l'information à cette passerelle au format XML.
2. La passerelle push Proxy assure l'envoi de l'information au terminal mobile.
3. La passerelle informe le serveur initiateur sur l'acheminement de l'information.

I-5- Les services accessible par la technologie WAP : [PIC 00]

L'Internet mobile ne consiste pas à naviguer sur Internet avec un téléphone mobile mais à proposer des services dans un contexte de mobilité. Ces services sont regroupés dans trois grandes catégories comme suit :

I-5-1- Services de communication :

Regroupe tous les services de base comme :

- Email, annuaire.
- Moteurs de recherche.
- Chat et forum...

I-5-2- Services d'information et media :

Les services d'information et media, en mode « pull » à l'initiative de l'utilisateur ou en mode « push » sous la responsabilité de fournisseur de contenu, constituent des services informationnels plutôt basés sur la notion de consultation :

- News.
- Information de trafic.
- Météo.
- Guide touristique.
- Horaire de vol...

I-5-3- Services de m-commerce :

Cette catégorie basée sur des notions d'interactivité plus importantes s'appuiera beaucoup sur le protocole de sécurité WTLS, on cite parmi elles :

- Commande de livres.
- Services bancaires (gestion de compte, virement...).
- Réservation de train, d'avion...

II- Le WAP et la sécurité :

Il existe certains services accessibles par le protocole WAP qui nécessitent un haut niveau de sécurité, parmi ces services on trouve le m-commerce, pour cela le WAP forum a spécifié une couche qui assure la sécurité des échanges entre le terminal WAP et la passerelle WAP, il s'agit de Wireless Transport Layer security (WTLS), qui a été développée en s'inspirant du modèle SSL (Secure Socket Layer).

Le protocole WTLS assure les fonctions suivantes :

- La confidentialité : elle assure aux deux parties de la transaction qu'elles sont les seules à avoir l'accès aux données échangées.
- L'intégrité : cette propriété assure que le contenu des données n'a pas été modifié entre l'émission et la réception.
- L'authentification : similaire à l'authentification dans le modèle client/ serveur classique.
- Le non-répudiation : elle permet de la non-répudiation de la transaction par une ou les deux parties de la transaction.

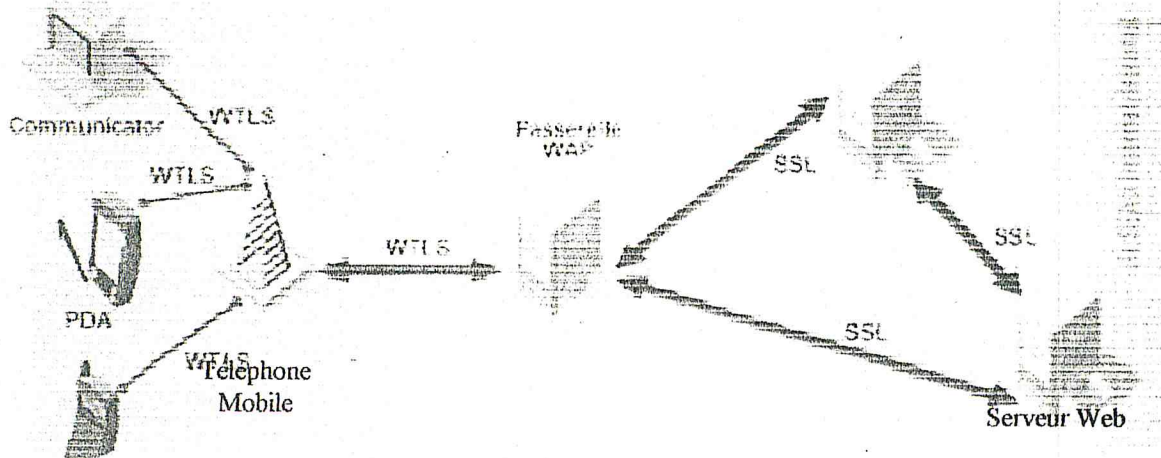


Figure.II.13 : Mode d'échange sécurisé des données

III- les avantages et les inconvénients de la technologie WAP :

a) les avantages :

Les téléphones mobiles intégrant la technologie WAP permettent tout d'abord d'accéder à des services interactifs tels que la consultation de la météo, de l'évolution des embouteillages dans les grandes villes, de consulter son compte en banque, comme ils permettent également de consulter des pages WML. On peut donc, n'importe où, profiter de l'immense regroupement de connaissance se trouvant sur le Web. De plus, avec l'évolution du GSM vers le GPRS, et donc l'évolution des taux d'accès sur le réseaux (le GPRS étant annoncé comme offrant un débit de communication dix fois supérieures à celui du GSM actuel), le protocole WAP permettra d'offrir à ces clients des services d'une grande interactivité (en utilisant par exemple le système de localisation du porteur du téléphone en temps réel).

b) Les inconvénients :

Malgré tous les efforts développés pour établir une compatibilité entre les sites Web et l'affichage sur téléphone mobile, la surface de l'écran de ces téléphones ne dépassera pas quelques lignes. De plus, avec les taux de transfert actuel (9,6 Kbps), l'affichage d'image reste possible mais guère performante.

Conclusion :

Internet est devenue donc l'outil le plus utilisé pour atteindre l'information, partout dans le monde, en toute sécurité avec une très grande vitesse, et accéder à différents services qui sont devenus une nécessité pour l'homme actuel tels que l'email, les forums, etc. et cela à partir de son PC de bureau, sans aucun déplacement. Aussi il est devenu possible d'atteindre l'information en temps réel sans aucune limite en se déplaçant en toute liberté, en utilisant simplement un téléphone portable.

Les téléphones mobiles intégrant la technologie WAP permettent donc d'accéder à des services interactifs tels que la consultation de la météo, de l'évolution des embouteillages dans les grandes villes, consultation des comptes bancaires, etc. On peut donc, de n'importe où, profiter de l'immense regroupement de connaissance se trouvant sur le Web. En utilisant les voix de communication mobile tels que GSM, GPRS,...

Dans le chapitre qui suit nous allons voir les différents aspects du langage compatible WAP : le langage WML (Wireless Markup language).

Chapitre III

Le langage WML (Wireless Markup Language)

Introduction :

Le W3C (World Wide Web Consortium) a abouti en Février 1998 à la publication des spécifications d'un nouveau langage XML qui est un métalangage qui offre la possibilité de structurer, syntaxer et ordonner les données en fonction des applications désirées.

Malgré que le XML représente une solution optimale en comparaison au HTML, il reste trop lourd pour être supporté actuellement par les terminaux sans fil, d'où un nouveau langage WML.

I- Le langage WML (Wireless Markup Language): [BOU 02]

Le WML «Wireless Markup Language» est un langage à balises dérivé du XML, il permet de spécifier le contenu des applications destinées aux équipements à faible bande passante tel que les téléphones portables, ou les pokets pc. Ces contenus apparaissent sous forme de textes, d'images, de listes de sélection, etc.

Les pages WML ou document WML sont appelés des « decks » ou « paquet » chaque paquet consiste en une ou plusieurs «cards», chaque paquet commence par la balise <WML>, et se termine par la balise </WML>, et chaque carte commence par la balise <card>, et se termine par la balise </card>.

Quand un micro navigateur accède à un document WML (paquet), il charge le paquet entièrement et l'utilisateur peut ensuite naviguer entre les différentes cartes sans avoir besoin de charger d'autres données.

Le WML comprend 4 domaines :

☛ Présentation et mise en forme du texte : le WML intègre la prise en charge du texte et des images, ainsi qu'une grande diversité de commandes de présentation et de mise en page, par exemple la possibilité d'afficher du texte en gras.

☛ L'organisation reprenant la métaphore du « jeu de carte » (deck), avec la structure carteWML/deckWML. Toute information WML est structurée sous forme de cartes et de decks. Les cartes spécifiant un ou plusieurs éléments d'interface utilisateur (menu de sélection, écran de texte, champ d'entrée de texte, etc.).

En générale l'utilisateur explore une série de cartes WML, passe en revue le contenu de chacune d'elles, saisit l'information requise, effectue des choix puis se déplace vers une autre carte.

Les cartes sont regroupées en decks, unités WML similaires à des pages HTML, elles sont identifiables par une URL. Les decks sont l'unité élémentaire de transmission d'un contenu WML.

☛ Exploration et lien de carte à carte : le WML prend en charge la gestion explicite de l'exploration entre les cartes et les yeux. Il offre également des indications sur la gestion des événements survenant sur le mobile.

La gestion des événements WML peut être utilisée pour la navigation ou pour l'exécution de script.

☛ Paramétrage et gestion des états : tous les decks WML peuvent être paramétrés à l'aide d'un modèle d'état. Des variables à valeur numérique peuvent être utilisées à la place de chaînes, et chargées par leur valeur au moment de l'exécution. Ce paramétrage optimise l'utilisation des ressources du réseau.

I-1- Caractéristique du langage WML : [PIC 00]

Comme nous l'avons dit précédemment le langage WML est très proche du XML, comparé au HTML le WML est néanmoins beaucoup plus stricte lorsqu'un document est interprété par un micro navigateur il ne doit y avoir aucune erreur. Ce langage utilise des balises entourées par des < et des >, la plupart de ces balises vont par paires, la première pour commencer <balise> et l'autre pour fini </balise>.

Les paragraphes suivants donnent beaucoup plus de détails sur les caractéristiques du langage WML.

Les URL :

WML utilise essentiellement les URLs absolues pour indiquer les liens de navigation vers une autre page. Les URLs sont aussi utilisées pour spécifier des ressources externes, comme les scripts WML Script et les images au format WBMP.

Il est possible d'employer des URL relatives au lieu de répéter toute l'URL, dans ce cas l'URL de base est celle du deck.

Les éléments :

Les éléments spécifiant l'ensemble d'information de balisage et de structuration concernant un deck WML, ils peuvent contenir une balise d'ouverture, un contenu et une balise de fermeture.

Les attributs :

Les attributs WML introduisent une information supplémentaire sur un élément plus précisément, les attributs spécifiant une information sur un élément qui ne fait pas partie du contenu de l'élément. Les attributs sont toujours spécifiés dans la balise d'ouverture d'un élément.

Les commentaires :

Les lignes commentaires sont encadrées par un signe inférieur et un signe supérieur et commencent toujours par un point d'exclamation et deux tirets, pour se terminer par deux tirets avant le signe supérieur.

```
<!--ceci est un commentaire-->
```

Les types MIME (content-type) :

Il s'agit de "Multipurpose Internet Mail Extensions", qui assure l'échange de données entre deux ordinateurs lorsque des fichiers qui ne sont pas du simple texte sont transmis à travers Internet. Lorsqu'un utilisateur Internet appelle une page Web, le serveur indique au navigateur de quel type MIME il s'agit, pour que le fichier puisse être affiché correctement. Cette information doit être indiquée dans l'entête http.

Il est nécessaire de déclarer le type MIME sur le serveur Web pour fournir le contenu WML et le format d'image associé WBMP.

Le tableau ci-dessous donne la liste de tous les types que le serveur doit connaître :

Type de fichier	Type MIME associé	Description
Wml	text/vnd.wap.wml	Fichier wml
Wmls	text/vnd.wap.wmlscript	Fichier wmlscript
Wbmp	image/vnd.wap.wbmp	Fichier graphique wbmp
Wmlc	application/vnd.wap.wmlc	Fichier wml compilé
Wmlsc	application/vnd.wap.wmlscriptc	Fichier wmlscript compilé

Tab.III.1 : Les types mimes.

I-2- Structure d'un deck :

Un deck est la plus petite unité WML transmise à un terminal WAP. Il est composé d'une série de cartes comme illustré ci-dessous : [www4]

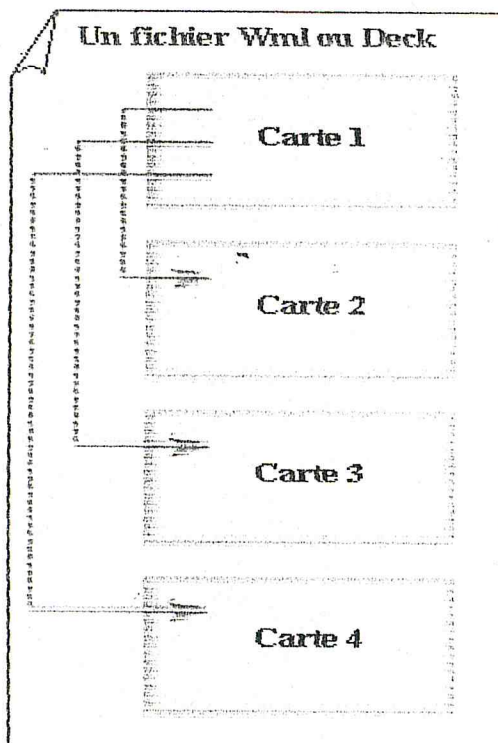


Figure.III. 1 : structure d'un deck.

Chaque document WML doit spécifier la version de XML, utilisée ainsi que la référence à la DTD (*Document Type Definition*) définissant l'ensemble des balises utilisées.

Le début d'un document WML se présente toujours de la façon suivante :

```
<?xml version="1.0" ? >
<!DOCTYPE WML PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
" http://www.wapforum.org/DTD/wml_1.1.xml">
```

Exemple :

```
?xml version="1.0"?>
```

```
<!DOCTYPE wml PUBLIC "-//OPENWAVE.COM//DTD WML 1.1/EN"
```

```
"http://www.openwave.com/dtd/wml11.dtd">
```

```
<wml>
```

```
<card id="ct1" title="Example">
```

```
<p align="left">
```

```
à gauche
```

```
</p>
```

```
<p align="center">
```

```
centré
```

```
</p>
```

```
<p align="right">
```

```
à droite
```

```
</p>
```

```
</card>
```

```
</wml>
```

II- Le langage WML script :

Le WML possède son propre langage de script coté client : WMLScript, qui même s'il est moins puissant que Java Script reste suffisant pour le WML.

Le WMLScript est un langage de script cote client, qui ajoute de nombreuses capacités à un navigateur WAP. Les concepteurs du WMLScript ont retenu les caractéristiques les plus utiles des environnements de données sans fil. Parmi les fonctionnalités WMLScript on trouve :

- ☛ Contrôle de la validité des informations saisie par un utilisateur.
- ☛ Affichage des messages d'alertes, d'information ou de confirmation visualisées rapidement par l'utilisateur sans emprunter le réseau. [LETOO]

Exemple :

Extern function alert ()

{

Dialogs.alert ("Pour participer il faut vous inscrire merci ");

WMLBrowser.go ("intermed.wml#card1");

}

III- Les outils de développement :

Pour développer un site WAP il faut utiliser les outils suivants :

- Un éditeur WML.
- Un convertisseur d'image.
- Un émulateur WAP.

Ou utiliser simplement un kit de développement.

III-1- Editeurs WML :

Il existe un nombre important des éditeurs WML dans le marché, et voilà les éditeurs les plus utilisés :

III-1-1- Easypad WAPtor :

Simple d'usage et permettant de visualiser directement le résultat du code Wml, cet éditeur peut être franchement conseillé pour une première approche du WAP et du Wml. Un petit reproche cependant : sa reproduction du code source Wml n'est pas tout à fait plus complète.

III-1-2- DotWAP :

Sa principale qualité est de pouvoir visualiser directement le code Wml sur une simulation d'un mobile Ericsson. Rudimentaire sur le plan des balises Wml disponibles, il propose également une possibilité de création de pages WAP aisément accessible.

Il existe d'autres éditeurs Wml tout aussi valables parmi eux on cite : WapSite, WapPage, CardOne et CoffeeCup Wireless Web builder

III-2- les convertisseurs d'images :

La plupart des téléphones portables ne supportent que les images de format Wbmp ou Png. Pour cela il faut utiliser un convertisseur des images de tout formats vers le format Wbmp compatible avec les terminaux WAP. Mais il existe actuellement dans le marché des téléphones portables qui supportent autre format d'images couleurs comme bmp, jpeg...

Il existe plusieurs logiciels convertisseur vers Wbmp comme le Pic 2 wbmp, wbmp converter..., et comme paint shop pour les images de format Png.

III-3- Les émulateurs WAP :

Les émulateurs WAP sont des outils permettant au développeurs de tester leur applications WAP sans d'être besoin d'un véritable téléphone portable, tel que il peuvent consulter un site WAP depuis leurs disques durs et non a partir d'une adresse http sur le Web.

Il existe plusieurs émulateurs sur le marché comme :

- Emulateur Nokia 7110.
- Ericsson.
- Siemens S45
- Openwave.
- M3 gate.
- Win WAP...

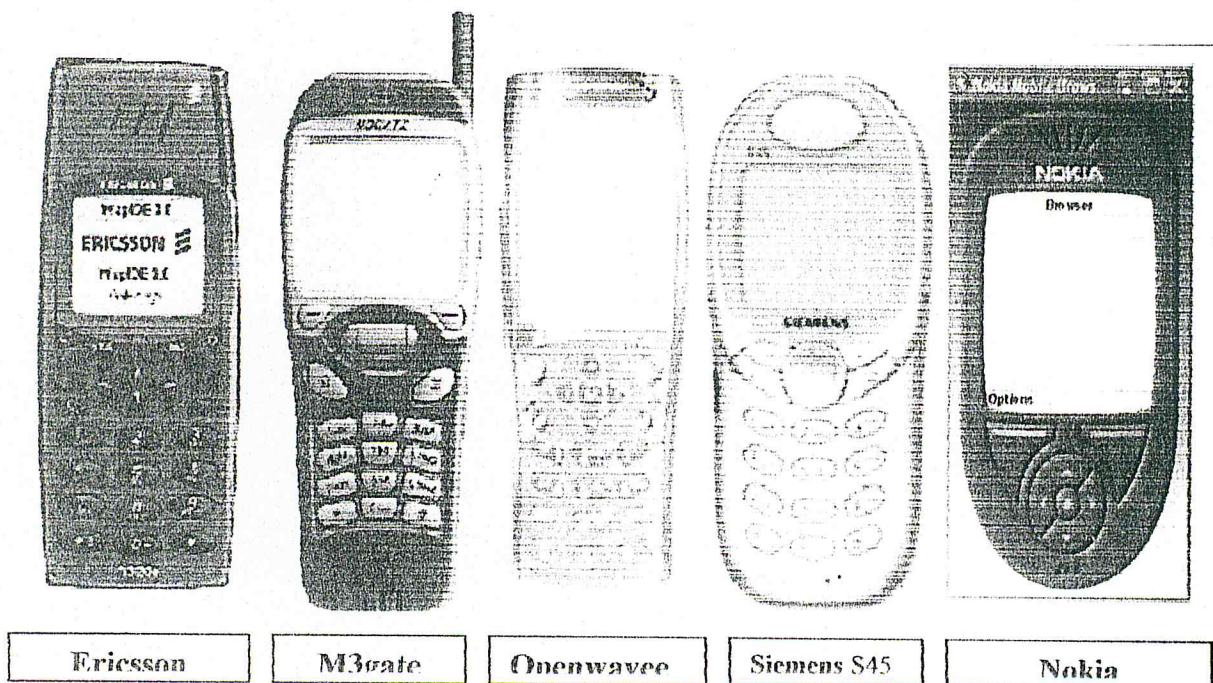


Figure.III. 2 : les différents émulateurs WAP.

III-4- Les kits de développement :

Les kits de développement proposés par les constructeurs des téléphones portables sont la meilleure solution pour les WAP - développeurs. Vu les problèmes de compatibilité du WAP entre les marques et les types de mobile, c'est une solution idéale pour tester un site. Bien entendu leur prise en main est moins intuitive que les éditeurs précédents. Les trois acteurs majeurs de marché de téléphonie mobile que sont NOKIA, Motorola et Ericsson proposent chacun une plate forme de développement et de test adapté a leur produits.

III-4-1- Nokia Internet mobile toolkit :

Nokia dispose un kit de développement qui s'appelle « Nokia mobile Internet toolkit », ce kit propose une série de fonctions qui sont :

- Simulation des mobiles 7110, 6110 et 6150.
- Visualisation de requête des erreurs.
- Editeur des images WBmp.
- Visualisation des variables locales.
- Connexion à une passerelle WAP.
- Connexion directe a un serveur http.
- Edition, analyse syntaxique.
- Documentation complète sur WAP, WML, et WML script (en anglais)

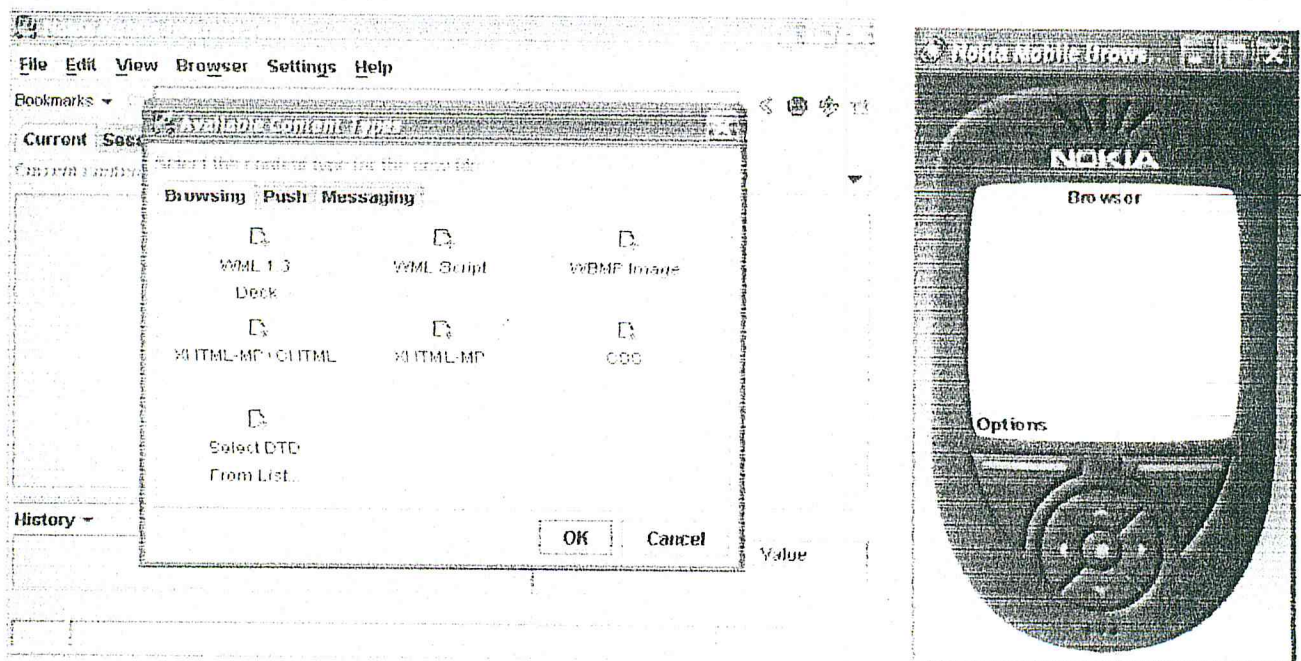


Figure.III. 3 : Nokia Mobile Internet Toolkit.

III-4-2- OpenWave SDK :

Tout comme le Nokia mobile Internet toolkit, le kit de développement Openwave SDK assure une série de fonction tels que :

- Simulation des mobiles Siemens S45 et Openwave.
- Visualisation de requête des erreurs.
- Visualisation des variables locales.
- Connexion à une passerelle WAP.
- Connexion directe a un serveur http.
- Edition, analyse syntaxique des documents WML, XHTML, HTML...
- Exemples du code WML.

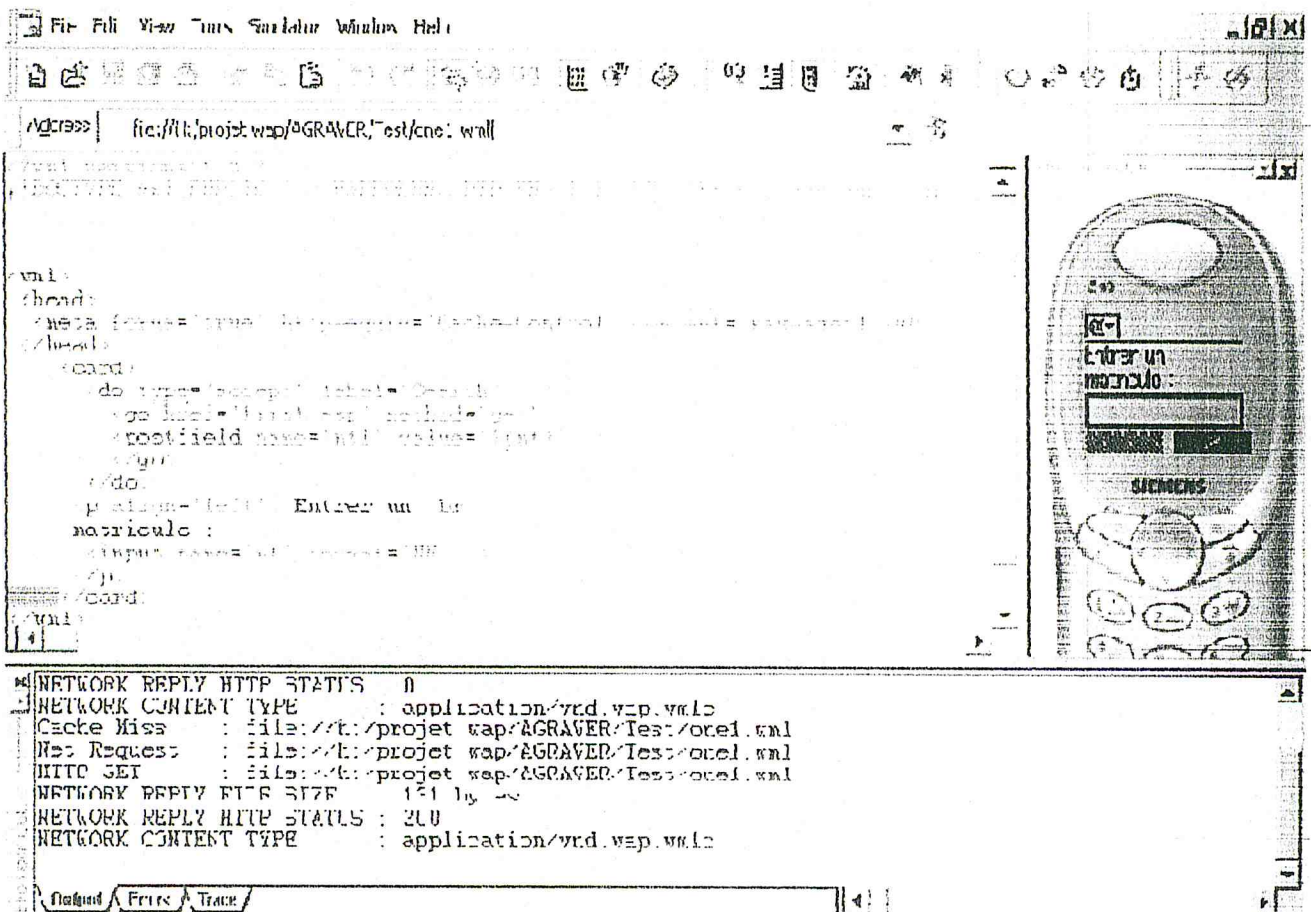


Figure.III. 4 : OpenWave SDK toolkit.

Conclusion :

Le langage WML Script ajoute au langage WML des capacités procédurales et des bibliothèques de fonctions. Ensemble ils fournissent un environnement de développement d'applications riche.

Le WML Script n'est pas suffisant dans le cadre de notre application, car nous avons besoin de créer des pages WML dynamiques pour accéder à la base de données qui stock les informations des membre. De ce fait nous avons besoin d'étudier quelques langages du script coté serveur.

Chapitre IV

Interfaces WAP/Base de données

Introduction :

Les services WAP à l'instar des services Web nécessitent l'échange d'information entre le client et le serveur.

La quantité colossale des informations échangées doit être structurée et stockée dans des bases de données, ces dernières nécessitent la définition des méthodes d'accès à distance pour l'extraction des informations demandés par le client.

Nous allons présenter dans ce chapitre les différentes interfaces de connexion Web/base de données.

I- Les différentes interfaces Web/base de données :**I-1- Common Gateway Interfaces (CGI): [MAG96]**

C'est un standard de programmations permettant d'accéder aux données de formulaires HTML, il définit une méthode pour que le serveur http et un programme extérieur puissent partager des informations. lorsque le serveur reçoit d'un client une requête lui demandant d'exécuter un programme (souvent appelé script CGI), il résume les informations présentes dans la requête en un ensemble de variables d'environnement. le script lit ces variables et y trouve les information lui permettant de répondre à la requête.

I-2- Internet Server Application Programming Interfaces (ISAPI) :

Est une interface propre à Microsoft, travaille avec les DLL (dynamique Link Library) ce qui revient à la création de bibliothèque de liens dynamique et non pas des application exécutables comme dans CGI. A l'intérieure d'une DLL, on dispose d'un ensemble de fonction qui permet de récupérer les informations en provenance du navigateur et de lui transmettre en retour des pages générées dynamiquement.

I-3- Active Server Pages [ASP] :

Active Server Page est une technologie Microsoft qui permet de développer des applications Web dynamiques et interactives. Elle repose sur l'utilisation des serveurs Web (Internet Information Server (IIS) pour Windows NT).

Les scripts ASP mélangent du code présent sous forme de script (VB Script, JAVA Script) et des éléments de présentation et des structurations de document, via les langages HTML ou WML.

Le serveur doit pouvoir distinguer un script à exécuter pour créer dynamiquement la page, d'un script à envoyer au navigateur. Active Server Page utilise les balises<%.....%>Pour baliser un script devant être exécuté sur le serveur et <script></script>pour baliser un script devant être exécuter par le navigateur lors du changement de la page ASP.

I-3-1- La structure générale du modèle ASP : [HOM98]

Bien que l'utilisation des scripts dans une page ASP soit limitée. Les scripts permettent d'agir sur des objets appartenant à l'environnement ASP via leurs propriétés et méthodes. Il y a dans l'ASP cinq objets prédéfinis, le schéma suivant illustre cette organisation :

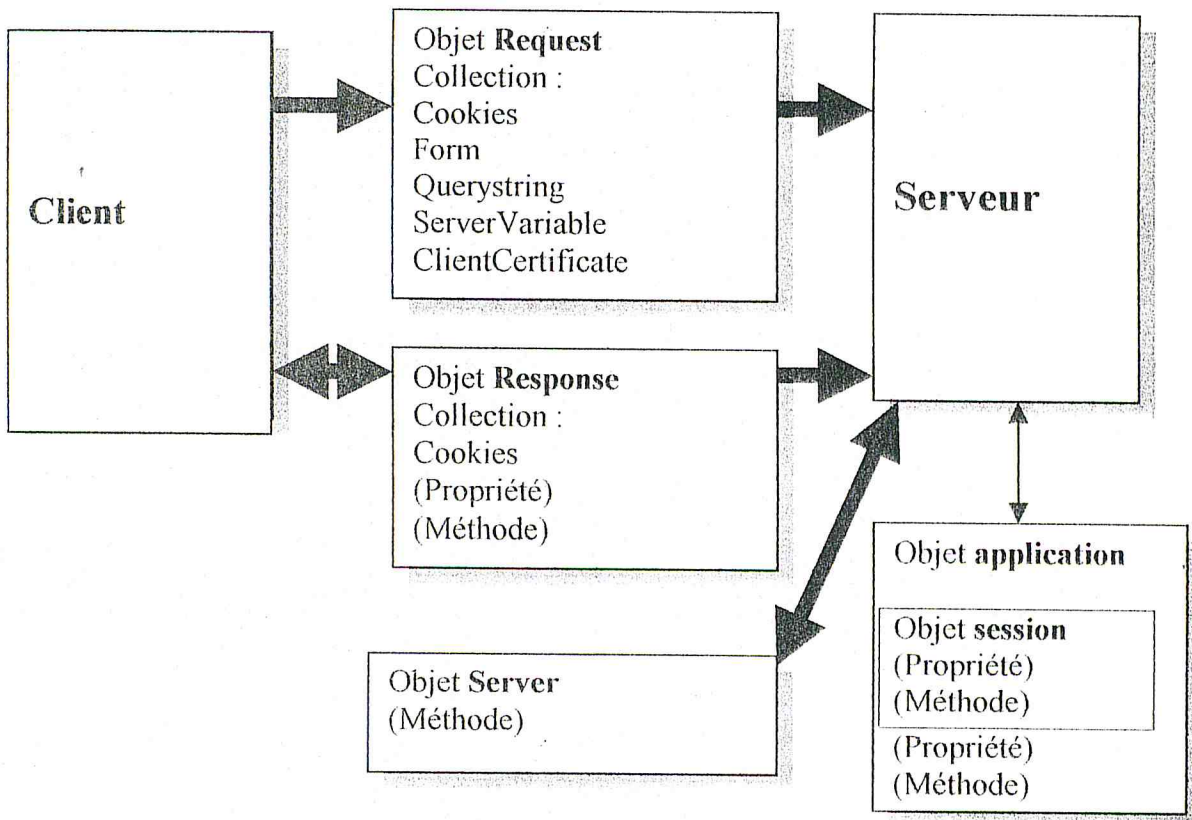


Figure.IV.1 : Le modèle ASP

- **Objet Request :**

L'objet Request fournit tout un ensemble d'informations sur les utilisateurs lors de leurs interactions avec l'application. Il permet donc de manipuler des informations allant du client vers le serveur.

- **Objet Response :**

On utilise l'Objet Response pour renvoyer des informations au navigateur. Il dispose des méthodes et propriétés pour créer et modifier ce qui est envoyé au navigateur ainsi que pour rediriger la requête vers une autre page «méthode Redirect»

- **Objet Server :**

Cet objet représente l'environnement dans lequel s'exécutent les pages. Il permet de gérer et d'administrer le serveur Web. La méthode la plus fréquemment utilisée est celle qui permet de créer l'instance d'un composant ActiveX (Server. CreateObjet).

Les autres méthodes permettent d'appliquer des codages spécifiques (codage URL ou HTML) à des chaînes, de définir des chemins virtuels vers des chemins physiques et de préciser les délais d'exécution d'un script avant son expiration.

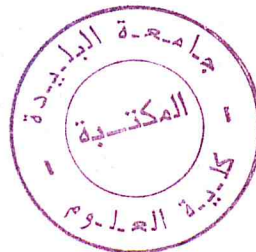
- **Objet Application :**

Cet objet permet de partager des informations (variable, constantes) entre tous les utilisateurs d'une même application. Il permet également de verrouiller et déverrouiller une variable de niveau application, traiter un script lors du démarrage ou l'arrêt de l'application.

- **Objet Session :**

Chaque client qui accède à une page est doté d'un objet session. La session est créée lors de la première demande du client et détruite par défaut 20mn après sa dernière demande.

L'objet session permet de gérer des informations d'un utilisateur particulier de l'application Web. Il permet également de verrouiller et déverrouiller une variable de niveau session et de traiter un script lors du démarrages ou l'arrêt d'une session.



I-4- Personal Home Page [PHP]:

PHP (PHP *Hyper Texte Processor* ou *Personal Home Page*) est un outil permettant de développer des application Web dynamique. C'est un langage de script côté serveur mélangeant au sein d'un même fichier du code HTML ou même WML et des instructions propres au langage PHP.

PHP est multi plateforme, il fonctionne sur UNIX, Windows 95/98 et NT, macintosh. Il peut être intégré dans n'importe quel serveur Web y compris Internet Information Server de Microsoft (IIS).

Les mêmes scripts PHP peuvent s'exécuter à partir de différentes plates formes sans subir aucune modification. En outre, PHP supporte la norme ISAPI qui offre de meilleures performances sur les serveurs de Microsoft.

I-4-1- Syntaxe de base :**I-4-1-1- Intégration de script PHP dans les pages WML :**

Les instructions PHP s'intègrent directement dans les pages Web. Il y à différents moyen pour passer du code WML au code PHP :

1. `<?..... ?>` : méthode classique pour baliser du code PHP.
2. `<?..... ?>` : méthode ajoutée dans PHP pour le rendre compatible avec XML.
3. `<SCRIPT LANGUAGE= □PHP□ ></SCRIPT>` : certains éditeurs de texte, comme Microsoft FrontPage, ne comprennent pas la balise ouvrante `< ?d'` ou l'implémentation de cette méthode.

I-4-1-2- Les commentaires :

Le PHP supporte les commentaires :

1. `//ceci est un commentaire sur une ligne`

2. `/* ceci est un commentaires`

Sur plusieurs lignes

`*/`

3. `# ceci est un commentaires sur une ligne`

I-4-1-3- Types de donnée :

PHP ne nécessite pas de déclaration explicite du type d'une variable est déterminé par le contexte d'utilisation. Il dispose des types de données suivants :

Les tableaux, les nombres à virgule flottante, entiers, les objets, les chaînes de caractères, booléen, les ressources qui sont des valeurs entières utilisées pour représenter des ressources telles que des fichiers ouverts, des connexions à des bases de données ,.....

I-4-2- Les variables /constantes :

En PHP, les variables sont représentées par un signe dollar suivi du nom de la variable.

```
$variable = valeur ;
```

La portée d'une variable dépend du contexte dans lequel la variable est définie pour la plupart des variables, la portée concerne la totalité d'un script PHP.

Mais, lorsqu 'on définit une fonction, la portée d'une variable définie dans cette fonction est locale à la fonction.

Pour définir des constantes, utilise la fonction «define ()» :

```
define (□CONSTANTE□, □valeur□) ;
```

I-4-3- Les tableaux :

Les tableaux enregistrent plusieurs valeurs dans des listes. On peut accéder a chaque élément en utilisant un index unique, qui peut être une valeur numérique (tableau scalaires) ou textuelle (tableau associatifs). Dans un même tableau, on peut trouver des index de ces deux types. Les valeurs peuvent correspondre à n'importe quel type notamment un autre tableau (on parle alors de tableau multidimensionnel).

On peut créer un tableau en utilisant les fonction *liste ()* ou *array ()*, ou bien en affectant explicitement chacune des valeurs.

I-4-4- Les fonctions :

La définition de la fonction peut se trouver n'importe où dans le script, notamment dans un fichier inclus.

```
Function nom_ fonction ($arg_1, $arg_2, .....$arg_n)
```

```
{
  [Bloc de code]
}
```

- **Appel d'une fonction :**

L'appel d'une fonction se fait de la manière suivante :

nom_fonction (\$arg_1, \$arg_2, \$arg_n) ;

Une fonction peut appeler d'autres fonctions ou s'appeler elle-même.

- **Les arguments :**

1. Une fonction peut ne pas avoir d'argument
2. Les arguments doivent être précédés d'un signe \$ pour devenir des variables de la fonction.
3. Un argument est transmis, par défaut, par valeur, ce signifie que seule la valeur est transmise. Toute modification au sein de la fonction n'est pas perçue à l'extérieur.
4. Un argument peut être transmis par référence, ce que signifie que l'adresse en mémoire est transmise. Toute modification au sein de la fonction sera perçue à l'extérieur.
5. Un argument (non précédé du signe □\$□) peut avoir une valeur par défaut.
6. Un argument pour lequel une valeur par défaut à été prévue peut être appelé par références.
7. Une fonction ne peut pas être déclarée avec un nombre d'argument variable, mais il existe une manière de contourner cette limite : en utilisant des fonctions *func_num_args* et *func_get_arg*, qui servent respectivement à fournir le nombre d'argument reçus et à lire un argument spécifique.

- **Valeur de retour :**

Les valeurs sont renvoyées en utilisant une instruction de retour optionnelle *return*.

I-4-5- Structures de contrôle :

Les instructions de contrôles permettent d'exécuter des blocs de code sous certaines conditions.

I-4-5-1- L'instruction if :

```
If (expression 1) {
```

Ce bloc instruction est exécuté si l'expression 1 est vraie

```
}
```

```
Elseif (expression2)
```

```
{
```

Ce bloc instruction est exécuté si l'expression 1 est fausse et si l'expression 2 est vraie

```
}
```

```
Else
```

```
{
```

Ce bloc instructions est exécuté si l'expression 1 et l'expression 2 sont fausses.

```
}
```

I-4-5-2- L'instruction switch :

```
Switch (variable){
```

```
    Case valeur1 :
```

```
        Bloc d'instruction
```

```
        Break;
```

```
    Case valeur2 :
```

```
        Bloc d'instructions
```

```
        Break;
```

```
    .....
```

```
    Défaut:
```

```
        bloc d'instructions
```

```
}
```

Il ne faut pas omettre le break en fin de bloc, sans quoi le reste du switch est exécuté.

I-4-5-3- Boucle for :

```
For ( initialisation ; test sortie ; incmentation )
```

```
{
```

```
    bloc instruction
```

```
}
```

I-4-5-4- Boucle while:

```
While (expression))
{
    Bloc d'instructions ;
}
```

I-4-5-5- Boucle do... while:

La condition de sortie est située en fin de boucle. Ainsi la boucle est parcourue une fois au minimum.

```
Do {
    Bloc instructions;
}
While (expression)
```

I-4-5-6- Boucle foreach :

Cette instruction est relative aux tableaux. Elle récupère les paires index/valeur.

```
Foreach (array as $index=>$valeur)
{
    bloc instructions
}
```

□as□ est un mot réservé.

I-4-6- Entrée/Sortie (I/O) et accès au disque :

Dans cette partie, nous allons étudier toutes les techniques dont dispose PHP pour échanger des données : à partir d'un fichier ou d'une base de données, ou via Internet ou encor à partir de données inscrites dans un formulaire par un visiteur du site.

I-4-6-1- Envoyer des données vers le navigateur :

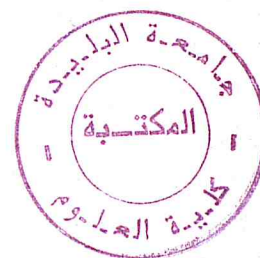
PHP propose trois fonctions qui permettent d'envoyer du texte au navigateur :
Echo, print, printf.

Bibliographie :

- [DES 99] : WAP : Wireless Application Protocol /Alain Deseine.-1999.
- [HOR 99] :A.Horch et T.Balbach : Telemedical information system -1999
- [MEU 96] : K.meunier : Le réseau de télémédecine en radiologie au CHU de Nancy- 1996
- [BOU 02] : Net Economic/ Jérôme Bouteriller.-23 octobre2002.
- [HOM 98] : Interface entre WEB-et base de données/Alex Homer, Darren Gill, Stephen Jakab.- Edition Eyrolles : 1998.
- [LET 00] : Construction d'une application WAP/Laurent Letourmy, Thomas Papiernik, Alain Hélaïli, Xavier Martzel.- Edition Eyrolles : 2000.
- [MAG 96] : Serveur Web/Jonathan Magid, R.douglas Matthews, Paul Jones.- Edition International Thomson Publishing.-1996.
- [PIC 00] : Le WAP : livre blanc/Olivier Picard, Jean-Michel Allemand.- Alladin technologies : 2000.
- [WAP 01] : Le Guide officiel WAP 1.2/ WAPForum ; trad. Valéry Frémaux, Joëlle Cornavin, .../Eyrolles ; fev.2001

Les sites WEB:

- [www1]: comment ça marche: <http://www.commentcamarche.fr>
- [www2] : Guill.net- Les protocoles de communication.
[http:// www.guill.net](http://www.guill.net)
- [www3] : PHP Every Where
<http://PHP.weblogs.com>
- [www4] : Nokia mobile phone, Nokia WAP Toolkit 3.0 user's guide, juin 2000.
<http://www.forumnokia.com>
- [www4] : [WAP@WML](http://www.wml.com) / Van Lancker Luck .- 2001
<http://www.ccim.be/ccim328/wap>
- [www5]:Débuter en PHP: <http://www.php.net>
- [www6]: tous sur le WAP: <http://www.wapforum.com>



I-4-6-2- Récupérer des données à partir d'un formulaire :

Les données en provenance des utilisateurs sont introduites via des formulaires HTML ou WML. Chaque champ d'un formulaire est reçu par le script au sein d'une variable globale qui porte le nom du champ.

I-4-6-3- Les cookies :

PHP supporte les cookies. Le cookie est un mécanisme d'enregistrement d'informations sur le disque dur du client, et de lecture de ces informations. Ce système permet d'authentifier et de suivre les visiteurs.

On peut envoyer un cookie avec la commande *setcookie()*. Les cookies font partie de l'entête HTTP, ce qui impose que *setcookie()* soit appelé avant tout affichage sur le client.

I-4-6-4- Lecture et écriture de fichier :

La communication avec les fichiers respecte le processus suivant : ouverture d'un accès au fichier à l'aide de la fonction *fopen()*, lecture ou écriture du fichier à l'aide des fonctions *fgets()* et *fputs()*, fermeture de l'accès à l'aide de la fonction *fclose()*.

Il est possible d'indiquer une adresse Internet en remplaçant le nom du fichier par son URL.

I-4-6-5- Accès aux bases de données :

La plus grande qualité et le plus important avantage du langage PHP est sa capacité d'interfaçage avec un grand nombre de base de données. Ces interfaces sont fournies sous forme de modules appelées « extensions » qui sont compilées avec PHP ou chargées dynamiquement.

II- Les composants des interfaces Web/Base de données :

L'interface ISAPI offre plusieurs techniques d'accès aux bases de données et la génération de contenu Web dynamique. Un de ses composants livrés avec Active Server Page (ASP) est conçue pour répondre à ce besoin : il s'agit du composant ADO (Active Data Object). Cependant, PHP (Personnel Home Page) qui intègre ce composant, offre aussi des fonctions d'accès à la base de données en utilisant le standard ODBC avec le pilote adéquat.

Ce schéma illustre les différentes combinaisons d'accès à la base de données via ASP et PHP :

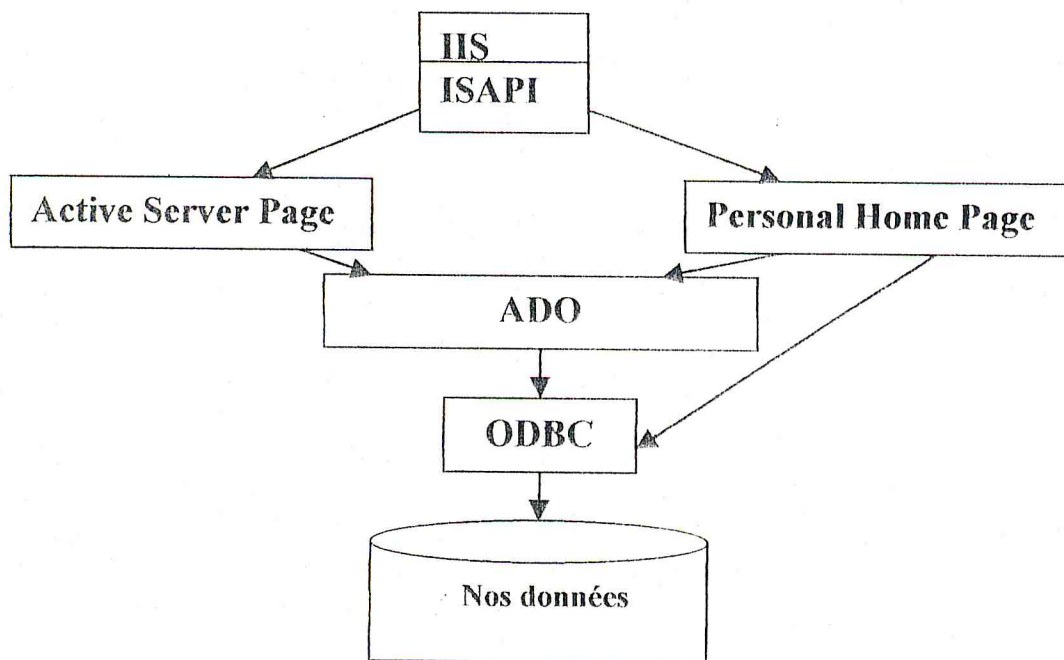


Figure.IV.2 : Les composants de l'interface ISAPI/base de données

II-1- Active Database Object (ADO):

Active Database Object (ADO) est une technologie permettant la connexion aux sources de données, telles que les bases de données, et les feuilles de calcul.

Les objets ADO peuvent être utilisés à partir de Microsoft visual basic, java script, Active Server Page (ASP), et Personnel Home Page (PHP).

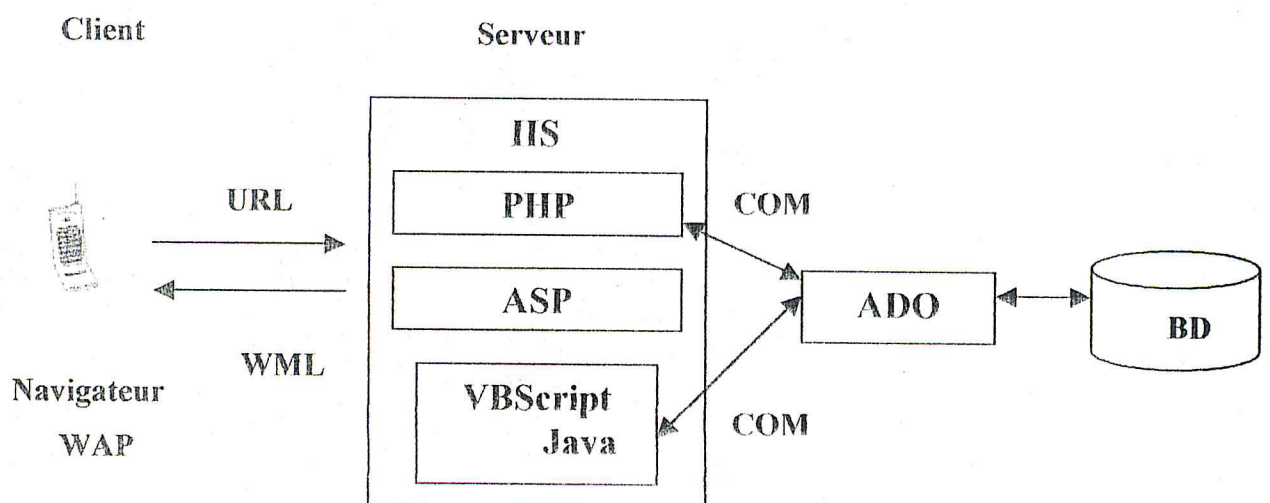


Figure.IV.3 : Accès à la base de données via ADO

Le composant ADO comprend trois objets principaux :

II-1-1- Objet Connexion :

L'objet connexion propose des propriétés et des méthodes qu'on peut utiliser pour ouvrir ou fermer des connexions aux bases de données et envoyer des requêtes pour mettre à jour des informations.

- **Ouverture d'une connexion :** La méthode *Open* de l'objet *Connexion* permet d'ouvrir la source de données après avoir créé la connexion.
- **Exécution des commandes sur la connexion :** Grâce à la méthode *execute* de l'objet *Connexion*, on peut envoyer des commandes. Une commande peut être soit une requête SQL, soit le nom d'une procédure stockée.
- **Fermeture d'une connexion :** En fin de traitement sur la base de données, on peut fermer la connexion et pour libérer toutes les ressources utilisées.
- **Utiliser les transactions :** Pour faire plusieurs mises à jours simultanés, on peut utiliser le mécanisme de transaction en demandant au système d'effectuer des modifications, puis les validant en une opération, plutôt que de les écrire les unes après les autres.

L'objet *Connexion* dispose des méthodes suivantes :

- ✓ *BeginTrans* : lance une nouvelle transaction.
- ✓ *CommitTrans* : enregistre les modifications et termine la transaction.
- ✓ *RollbackTrans* : annule les modifications et termine la transaction.

II-1-2- Objet Command :

Grâce à l'objet *command*, on peut exécuter les requêtes de la même façon qu'avec l'objets *connexion*, sauf qu'avec l'objet *command*, on peut préparer et compiler une requête sur la source de la base de données, puis la soumettre à plusieurs reprises sur différents ensembles de valeurs. Ce type de compilation des requêtes a l'avantage de réduire considérablement le temps nécessaire à l'application pour effectuer les modifications nécessaires sur une requête existante. En outre, les requêtes SQL peuvent rester partiellement indéfinies, car on a la possibilité de modifier certaines parties des requêtes, juste avant leur exécution.

- **Créer une connexion :** Pour créer une connexion avec l'objet *Command*, on utilise la propriété *ActiveConnexion* après avoir créé une instance de l'objet *Command*.
- **Créer et ajouter des paramètres :** Si la requête nécessite des paramètres, on peut les ajouter à la collection *Parameters* de l'objet *Command*.

- **Exécuter une commande** : La méthode *Exécute* de l'objet *Command* permet d'exécuter la requête SQL ou la procédure stockée indiquée dans la propriété *CommandText* de l'objet *Command*. Si la méthode *Exécute* exécute une requête partiellement indéfinie, on doit mettre la propriété *prepared* de l'objet *Command* à *true* avant l'exécution.

II-1-3- Objet Recordset :

Les deux méthodes que nous avons étudié pour exécuter une requête créent des ensembles d'enregistrement *Recordset* contenant le résultat de la requête.

L'objet *Recordset* dispose d'un ensemble de méthodes et propriétés qui permettent, selon les contraintes de la requête, de récupérer et d'afficher un ensemble d'enregistrements, provenant d'une base de données. L'objet *Recordset* conserve la position de chaque enregistrement renvoyé par une requête, permettant ainsi d'examiner les résultats, élément par élément.

- **Déplacement dans un Recordset :**

Il existe cinq méthodes qui permettent de se déplacer dans un *Recordset* :

1. La méthode *Move* : déplace le pointeur d'enregistrement de n positions, vers l'avant (nombre positif) ou vers l'arrière (nombre négatif), relativement à l'enregistrement courant.
2. La méthode *MoveFirst* : déplace le pointeur sur le premier enregistrement.
3. La méthode *MoveLast* : déplace le pointeur sur le dernier enregistrement.
4. La méthode *MoveNext* : déplace le pointeur sur enregistrement suivant.
5. La méthode *MovePrevious* : déplace le pointeur sur enregistrement précédant.



Le tableau suivant résume les différents codes d'appel aux objets ADO dans l'ASP et le PHP :

		ASP	PHP
Objet Connexion	Création	Set idc = Server.CreateObject ("ADODB.Connection")	\$ idc = new com ("ADODB.Connection")
	Ouverture	idc.Open chaine_connexion	\$idc->Open chaine_connexion
	Exécution	idc.Execute "commande"	\$ idc->Execute ("commande")
	Fermeture	idc.Close	\$ idc-> Close
Objet Command	Création	Set cmd=Server.CreateObject ("ADODB.Command")	\$cmd = new com ("ADODB.Command")
	Ouverture	cmd.ActiveConnection "chaine_connexion"	\$ cmd->ActiveConnection "chaine_connexion"
	Exécution	cmd.CommandText = "requête SQL ou procédure" cmd.execute	\$cmd->CommandText= "requête SQL ou procédure" ; \$cmd->exécute
Objet Recordset	Création	Set \$rs = server.CreateObject (ADODB.Recordset)	\$rs = new com (ADODB.Recordset)

Tab.IV.1: différents codes d'appel aux objets ADO dans l'ASP et le PHP

II-2- Le module ODBC du PHP :

Le PHP offre un ensemble de fonctions permettant la communication avec plusieurs bases de données, ou même d'accéder à des bases de données compatibles ODBC.

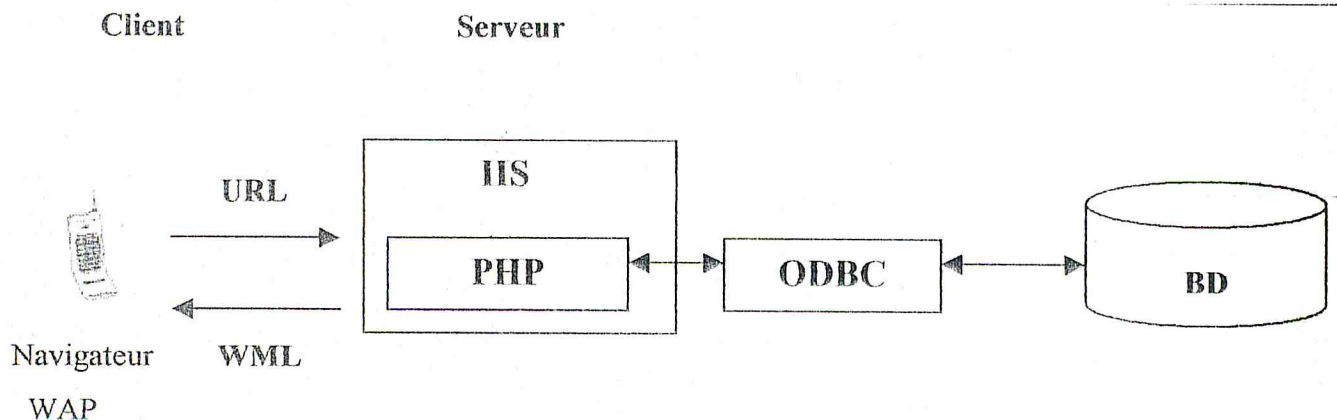


Figure.IV.4 : Accès à la base de données via le module ODBC du PHP

Dans cette partie nous allons présenter les fonctions du module ODBC permettant la communication avec des bases de données via des sources de données ODBC.

II-2-1- Connexion à une base de donnée :

Int odbc_connect (string dsn, string password, int cursor_type):

permet de se connecter à une source de données ODBC. L'identifiant de connexion retournée par cette fonction est nécessaire pour toutes les autres fonctions ODBC. On peut avoir de multiples connexions en même temps. Le quatrième paramètre (facultatif) fixe l'utilisation d'un type de curseur de résultat particulier utilise pour cette connexion, il prend la valeur « SQL_CUR_USE_ODBC » pour contourner certains problèmes ODBC.

II-2-2- Exécution d'une commande :

String odbc_do(int conn_id, string query): exécute la requête *query* avec la connexion *conn_id*, elle renvoi un identifiant de résultats *result_id*.

II-2-3- Récupération d'un résultat :

Int odbc_fetch_into(int result_id, int rangée, array champs): Reçoit l'indice de rangée au sein du *result_id* spécifié et place le contenu des colonnes dans le tableau *champs*, argument qui doit être passe en référence.

La fonction renvoie le nombre de colonnes ou 'false'. Si l'argument *range* est omis, la prochaine rangée est envoyée, ce qui permet de parcourir un résultat.

Int odbc_fetch_row (int result_id, int rangée): Prend une rangée de données à partir de *result_id*. les données de la ligne sont stockées en mémoire interne, prêtes à être reçues par la fonction *odbc_result*. L'argument *rangé* reste optionnel, s'il n'est pas spécifié la prochaine rangée disponible sera envoyée.

String odbc_crusor (int result_id): lit le pointeur de fiche courante (*cursorname*) pour le résultat *result_id*.

II-2-4- Manipulation des champs :

String odbc_field_name (int result_id, int champ) : fournit le nom d'un champ au sein d'un *result_id*.

String odbc_field_type (int result_id, int champ): fournit le type d'un champ au sein d'un *result_id*.

Int odbc_field_len (int result_id, int champ): fournit la longueur d'un champ au sein d'un result_id.

Int odbc_num_fields (int result_id) : fournit le nombre de champ au sein d'un result_id.

String odbc_result (int result_id, string champ): fournit la valeur d'un champ pour la rangée courante au sein d'un result_id. les champs peuvent être référencés par des noms ou par des nombres. La numérotation doit commencer par 1.

II-2-5- Préparer et exécuter une requête :

Int odbc_prepare (int Connection_id, string query): cette fonction analyse une requête et prépare son exécution. Un identifiant du résultat, qui peut être exploité par odbc_execute, est envoyé.

Int odbc_execute (int result_id, tableau parametres) : exécute une instruction préparée. L'argument result_id correspond à l'identifiant du résultat renvoyé par odbc_prepare.

II-2-6- Fermeture d'une connexion :

Void odbc_close (int Connection_id): ferme la connexion avec la base de données. Cette fonction échouera si il y a des transactions en cours sur cette connexion, dans ce cas, la connexion restera ouverte.

Void odbc_close all : ferme toutes les connexions ODBC à base de données.

III- Choix de la technique :

L'ASP et le PHP sont deux techniques de génération de pages WAP dynamiques. ASP est une technologie Microsoft supportée par un nombre restreint de serveur Web ; cependant, PHP est multi-plateforme, il fonctionne sur Unix, windows95/98 et NT, Macintosh. PHP peut être intégré dans n'importe quel serveur Web y compris Internet Information Server de Microsoft (IIS).

Ainsi les mêmes scripts PHP peuvent s'exécuter à partir de différentes plateformes sans subir aucune modification.

ASP est bâti sur une architecture COM : Quand un programmeur ASP utilise VBScript, il utilise un objet COM. Quand il répond au client, il fait appel à un objet COM : objet réponse, quand il accède à une base de données, il utilise un autre objet COM : ADO. Tous ces appels aux objets COM augmentent le temps de réponse. Dans PHP, tout s'exécute dans le même espace de la mémoire. Cela veut dire que le code PHP s'exécute plus rapidement s'il n'y a aucune communication avec les objets COM. [www3]

Pour ces raisons, nous avons choisi le PHP comme langage de script côté serveur et technique d'accès à la base de données via son module ODBC.

Conclusion :

Dans ce chapitre nous avons fait un tour complet des technologies qui s'exécutent du côté serveur, permettant de publier, de stocker, et de récupérer des informations sur un site. Ces dernières peuvent avec leurs moyens propres, offrir toutes les fonctionnalités nécessaires pour concevoir et réaliser un simple système de base de données.

Le chapitre qui suit présente le processus de conception et de définition d'une base de donnée qui est essentielle pour le développement des applications dynamiques.

Chapitre V

Conception de notre système

Introduction :

L'Internet mobile est une révolution technologique qui a affecté plusieurs domaines de la vie, ainsi elle a fait ses preuves comme étant un moyen de communication en se déplaçant sans aucune limite. Cette technologie offre un environnement efficace de communication grâce aux applications WAP.

Notre projet consiste à concevoir et à réaliser une application WAP qui offre un moyen de gestion et de communication permettant aux médecins de suivre leur malades a distance et d'échanger leurs opinions et leur savoir faire en se basant sur la messagerie.

Notre plate forme doit prendre en charge les changements de façon automatique, c'est pour ça la plus part des pages devront être générées dynamiquement a partir des bases de données.

Pour ce faire nous avons besoin d'une interface utilisateur WAP qui permet aux médecins, de consulter la base de données, ajouter, modifier et supprimer des informations, de consulter le forum, envoyer des messages et utiliser le chat pour discuter avec d'autre médecins...etc, et des bases de données pour le stockage des informations.

I- Infrastructures et équipement :

Le système de téléconsultation a été développé pour l'environnement Windows 2000 serveur doté du serveur Internet IIS (Internet Information Server) avec le langage de script PHP et comme logiciel de base de données SQL server.

Donc à l'aide de ces outils on a entamé la conception de l'application avec comme première étape le recensement des principaux modules composant l'application et on a abouti au schéma suivant :

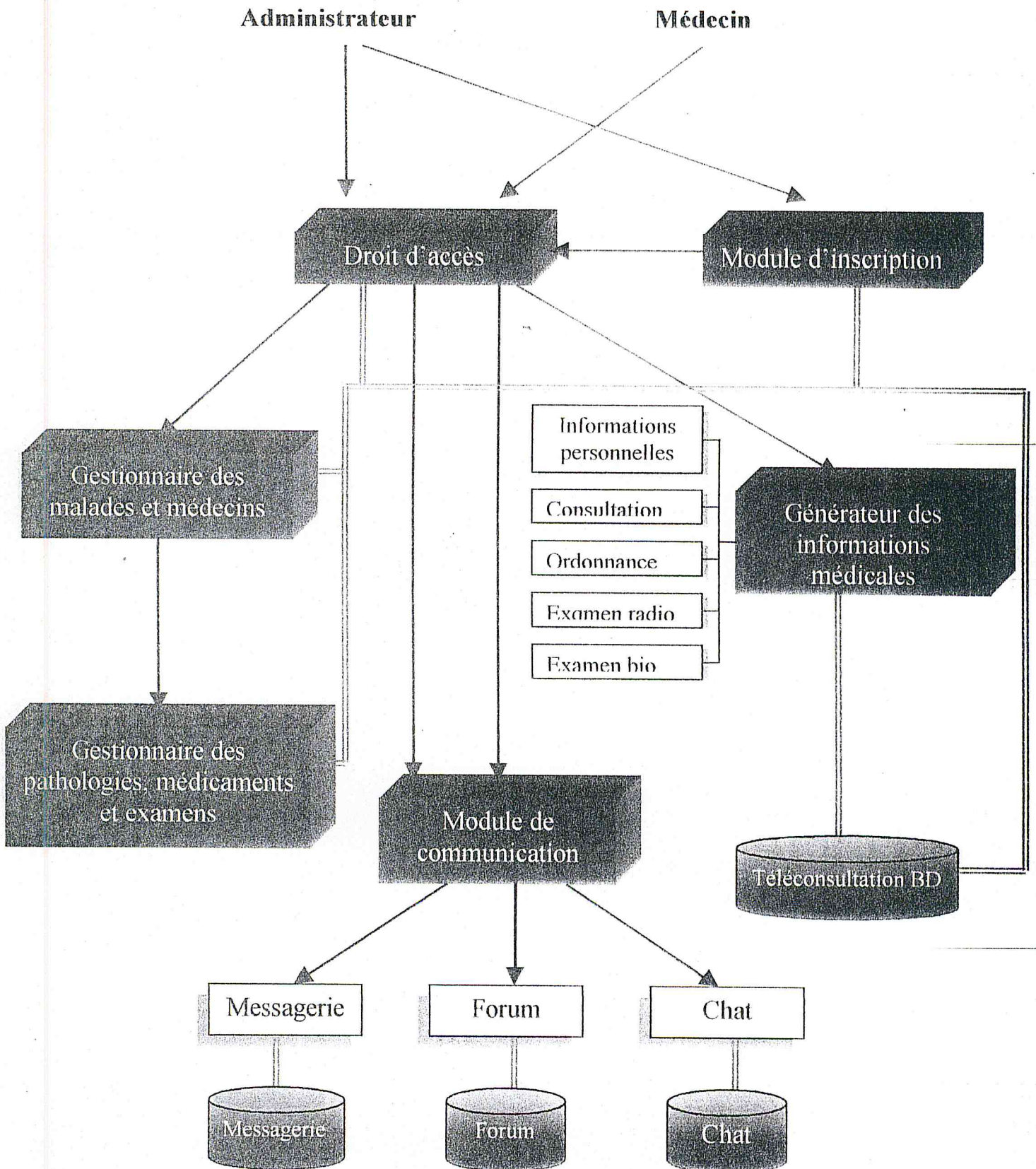


Figure.V.1 : Le plan de l'application

D'après la **figure.V.1**, tous les modules de système sont alimentés par une source de données (base de données).

L'accès à cette source de données est réalisé via le PHP par un mode d'accès avec DSN (Source de Données Système) à l'aide des instructions suivantes :

La chaîne de connexion :

```
<?
$connectionstring = odbc_connect ("source", "sa", "123456");
?>
```

Exemple de requête SQL :

```
<?
$query1 = "SELECT * FROM malade";
$queryexe1 = odbc_do ($connectionstring, $query1);
?>
```

Récupération de données :

```
<?
while (odbc_fetch_row ($queryexe1))
{
$mat_mal= odbc_result ($queryexe6, 1);
$nom= odbc_result ($queryexe6, 2);
$prenom= odbc_result ($queryexe6, 3);
$dne= odbc_result ($queryexe6, 4);
$idn= odbc_result ($queryexe6, 5);
$adresse= odbc_result ($queryexe6, 6);
$profession= odbc_result ($queryexe6, 7);
$sex= odbc_result ($queryexe6, 8);
$song= odbc_result ($queryexe6, 9);
$telephone= odbc_result ($queryexe6, 10);
?>
```

II- Description des modules :**II-1- Droit d'accès :**

Cette fonction sert à protéger la plate forme contre les accès non autorisés, grâce a cette fonction le système réclame pour chaque utilisateur les informations suivantes :

Un nom utilisateur et un mot passe qui permettent d'identifier l'utilisateur (médecin ou administrateur), et de récupérer les informations le concernant.

Si la phase d'authentification est concluante, une page d'accueil personnalisée s'affiche alors dans le navigateur WAP.

Ceci se traduit par l'algorithme suivant :

Saisie du nom d'utilisateur et du mot de passe.

Recherche de l'utilisateur dans la base de données et vérification du mot de passe

Si l'utilisateur n'existe pas ou si le mot de passe est faux

Alors Un accès non autorisé.

Sinon sélectionner le droit d'utilisateur à partir de la base de données

Si droit égale à Admin.

Alors afficher la page d'accueil propre à l'administrateur.

Sinon (droit=autre) afficher la page d'accueil spécifique au médecin.

Fin si ;

Fin si ;

II-2- Module d'inscription :

Ce module permet aux médecins de s'inscrire à la base de données ou au forum. L'opération d'inscription est une procédure simple, il suffit juste de remplir un formulaire. Le script suivant montre comment se déroule cette opération :

Récupération des informations sur médecin :

```
<postfield name="mat_med" value="$mat_med"/>
```

```
<postfield name="nom_med" value="$nom_med"/>
```

```
<postfield name="pre_med" value= "$pre_med"/>
```

```
<postfield name="adr_med" value= "$adr_med"/>
```

```
<postfield name="tel_med" value= "$tel_med"/>
```

```
<postfield name="eml_med" value= "$eml_med"/>
```

```
<postfield name="grd_med" value= "$grd_med"/>
```

```
<postfield name="cod_pat" value= "$cod_pat"/>
```

```
<postfield name="ser_med" value= "$ser_med"/>
```

Connexion à la base de donnée:

```
<?
$connectionstring = odbc_connect ("source", "sa", "123456");
?>
```

Vérification de l'existence des informations saisies :

```
<?
$query2 = "SELECT pre_med from medecin WHERE mat_med='$mat_med'";
$queryexe2 = odbc_do ($connectionstring, $query2);
?>
```

Si ces informations n'existe pas alors valider l'inscription sinon afficher un message d'avertissement:

```
<?
If (! (odbc_fetch_row ($queryexe2)))
{
$query = "INSERT INTO medecin (mat_med, nom_med, pre_med, adr_med, tel_med, eml_med,
grd_med, cod_pat, ser_med)
VALUES ('$mat_med','$nom_med', '$pre_med', '$adr_med', '$tel_med', '$eml_med',
'$grd_med', '$cod_pat', '$ser_med')";
$queryexe = odbc_do($connectionstring, $query);
}
else
{
print "Il y a un autre medecin qui a le même matricule, veuillez le changer";
}
?>
```

II-3- Générateur des informations médicales :

Ce module est spécifique au médecin, grâce auquel il peut insérer des différentes informations sur leurs patients :

Information personnelles : il s'agit des informations personnelles sur chacun de ses patients tel que : nom, prénom, date de naissance, lieu de naissance, adresse,...

Consultations : c'est l'ensemble des consultations fait par le médecin sur un malade, chaque consultation a une date et un résultat.

Ordonnance : il s'agit des ordonnances données par le médecin à un malade, chaque ordonnance est caractérisé par un numéro, et contient un ou plusieurs médicaments.

Antécédents personnels : il s'agit de l'historique du patient.

Antécédents familiaux : Le Médecin peut lister la liste de toutes les pathologies existantes dans l'entourage familial du patient, pour d'éventuelles pathologies héréditaires.

Examen biologique : il s'agit l'ensemble des examens radiologiques du patient.

Examen radiologique : il s'agit l'ensemble des examens biologiques du patient.

II-4- Gestionnaire des médecins et malades :

Ce module est réservé à l'administrateur du système, il lui permet de gérer l'ensemble des médecins et malades participant au système, les outils de gestion proposés sont :

- Validation de l'inscription d'un médecin.
- Suppression ou modification d'un médecin.
- Suppression ou modification d'un malade.

II-5- Gestionnaire des pathologies, médicaments et examens:

Ce module est réservé à l'administrateur, il lui permet de gérer l'ensemble des médicaments, pathologies, examen radiologique et examen biologique avec la possibilité de :

- Ajouter, modifier ou supprimer un médicament.
- Ajouter, modifier ou supprimer une pathologie.
- Ajouter, modifier ou supprimer un examen radiologique.
- Ajouter, modifier ou supprimer un examen biologique.

II-6- Module de communication :

Ce module est partagé par tous les utilisateurs du système (administrateur, médecin ou malade), il leur permet de communiquer les uns avec les autres selon différents modes d'échange (messagerie, chat, forum).

III- Description du système d'information :

Une bonne conception de l'application doit permettre à celle-ci de correspondre parfaitement aux besoins du client. La conception est donc une étape fondamentale qui permet notamment de se préoccuper de la structuration de l'information. D'où la nécessité de concevoir une base de données.

Les bases de données correspondante à notre système comprend toutes les informations nécessaires pour une bonne structuration du système de téléconsultation médicale et les services de forum, chat et messagerie tel que les médecins, les malades, les consultations... etc.

III-1- Base de donnée de téléconsultation :

La base de données de téléconsultation offre aux médecins et aux administrateurs un espace de stockage et de gestion des informations médicales. Elle est constituée de sept tables pour assurer le bon fonctionnement du système.

Nous en déduisant le modèle conceptuel de données (MCD). Son modèle entité association est le suivant :

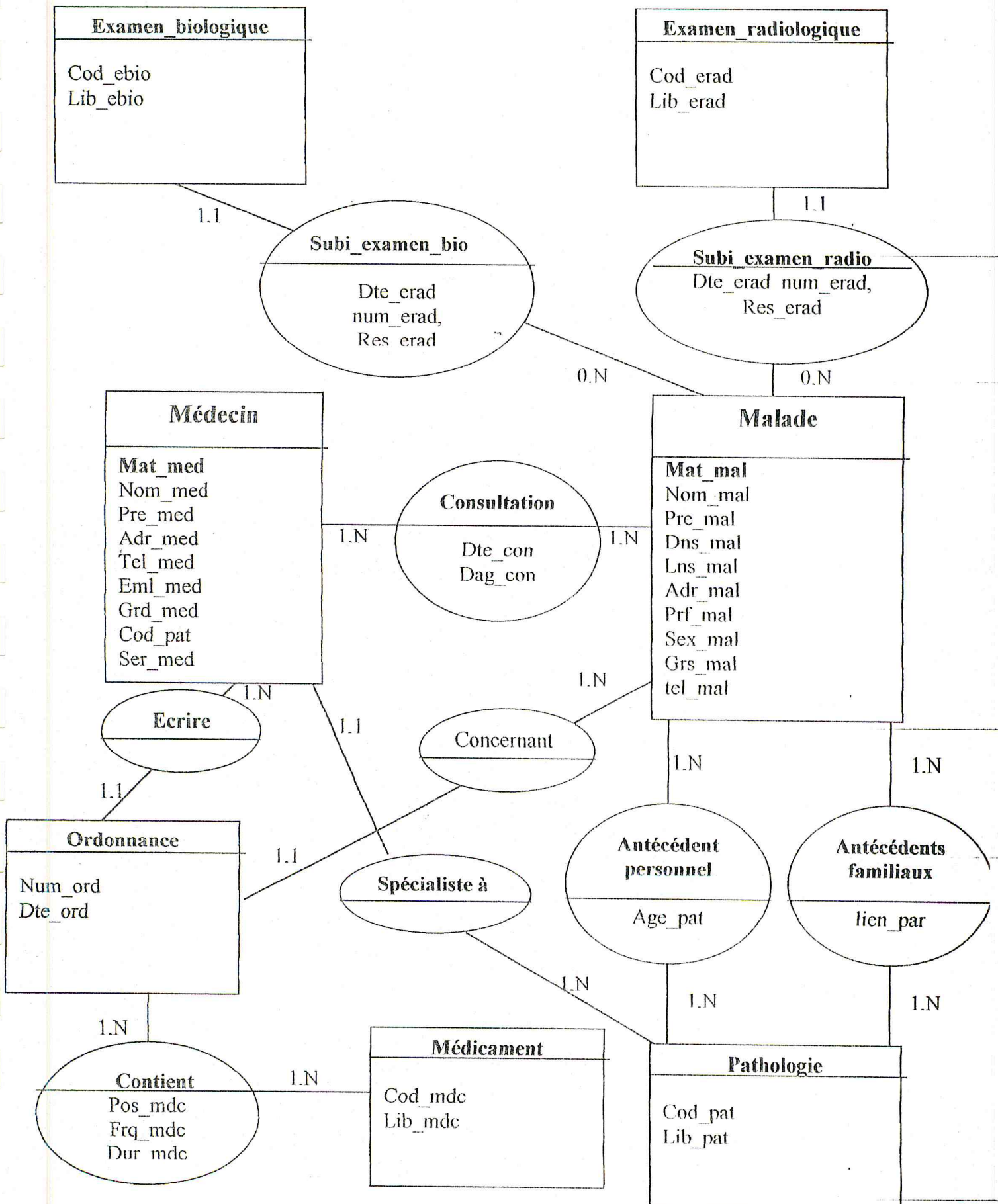


Figure.V.2 : MCD de la base de données de téléconsultation médicale

Table médecin :

Regroupe tous les médecins participant à notre système, chaque médecin est distingué par leur propre matricule. Cette table est générée par les attributs suivants :

Mat_med: le matricule de malade, c'est l'attribut clé de cette table.

Nom_med : le nom de médecin.

Pre_med : le prénom de médecin.

Adr_med : de médecin.

Tel_med : le téléphone de médecin.

Eml_med : l'adresse E-mail du médecin.

Grd_med : le grade du récepteur du message.

Cod_pat : le code de pathologie où le médecin est spécialisé.

Ser_med : le service du médecin.

Table malade :

Regroupe l'ensemble des malades participant à notre système, chaque malade est identifiée par un matricule. Cette entité est engendré par les attributs suivants :

Mat_mal: il s'agit du matricule de malade qui est le clé de cette table, il permet de distinguer les malade entre eux.

Nom_mal: c'est le nom du malade.

Pre_mal: cet attribut permet l'identification du prénom du malade.

Dns_mal: il s'agit de la date de naissance du malade.

Lns_mal: le lieu de naissance du malade.

Adr_mal: l'adresse du malade.

Prf_mal: la profession du malade.

Sex_mal: le sexe du malade du malade.

Grs_mal: le groupe de sang du malade.

Tel_mal: le numéro de téléphone du malade.

La table ordonnance :

Regroupe l'ensemble des ordonnances données par les médecins à des malades, chaque malade est identifiée par un numéro. Cette entité est caractérisée par les attributs suivants :

Num_ord: le numéro de cette ordonnance est la clé de cette table

Dte_ord: la date de l'ordonnance.

La table médicament :

Cette table contient la liste des médicaments qui peuvent exister pour les utiliser directement dans notre système. Elle contient les attributs suivants :

Cod_mdc : le code de médicament pour distinguer les médicament entre eux.

Lib_mdc : le nom de médicament.

La table pathologie :

Il s'agit de l'ensemble des pathologie ou maladie, elle est engendré par les attributs suivants

Cod_pat : le code de pathologie qui est la clé de cette table.

Lib_pat : il s'agit du nom de la pathologie.

La table examen radiologique :

Cette table regroupe les différents examens radiologiques qui peuvent se faire au sein de l'hôpital, chaque examen est identifié par un code. Elle contient les attributs suivants :

Cod_erad : c'est la clé de cette table, il s'agit du code de l'examen radiologique.

Lib_erad : il s'agit du nom de l'examen radiologique.

La table examen biologique :

Cette table regroupe les différents examens biologiques qui peuvent se faire au sein de l'hôpital, chaque examen est identifié par un code. Elle contient les attributs suivants :

Cod_ebio : c'est la clé de cette table, il s'agit du code de l'examen biologique.

Lib_ebio : il s'agit du nom de l'examen radiologique.

III-2- Forum :

Cette base de données permet de gérer le forum de discussion du système, elle est constituée de quatre tables reliées entre eux, comme le montre le MCD suivant :

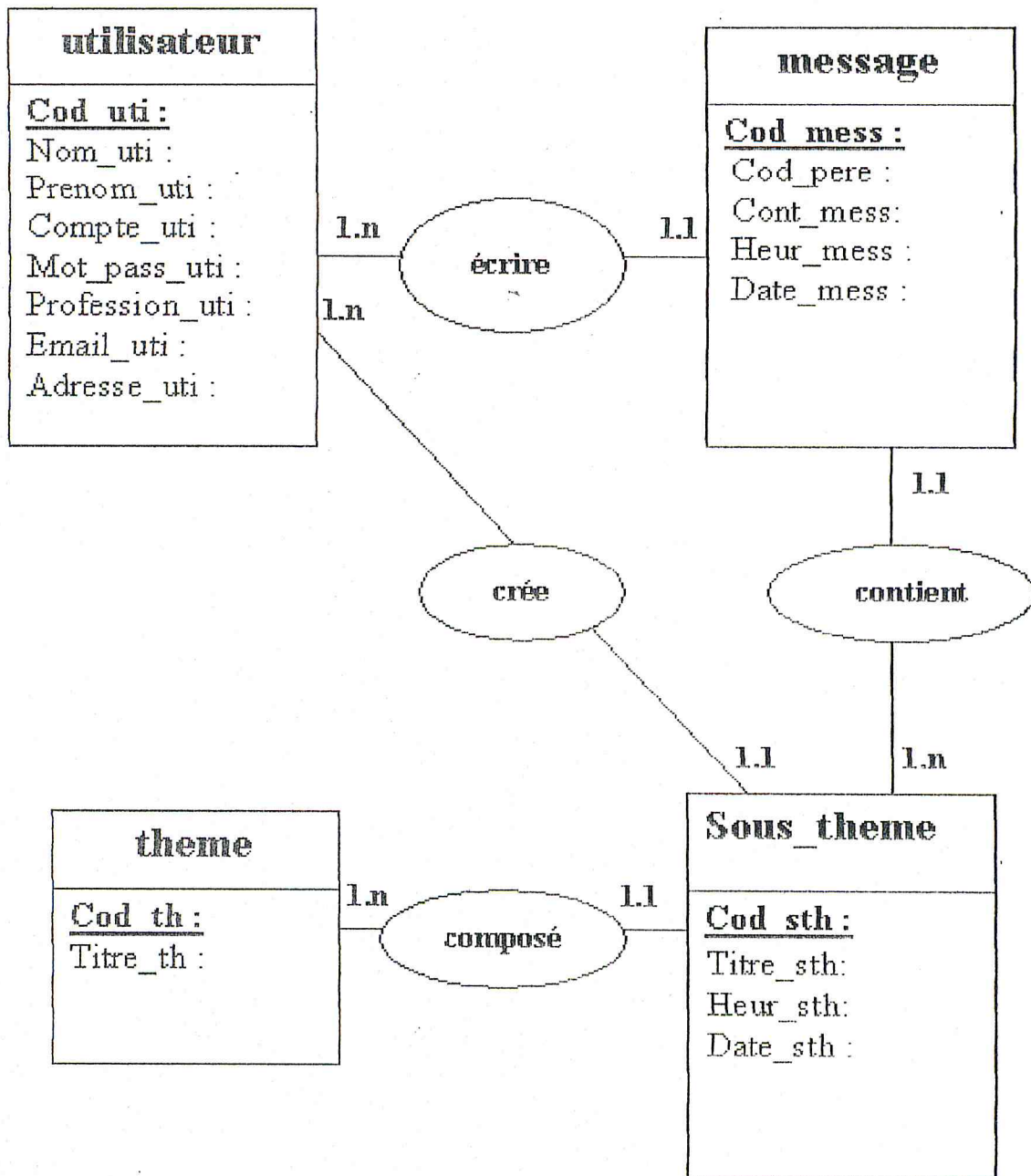


Figure.V.3 : Modèle conceptuel de données (MCD) du forum.

Entité utilisateur :

Regroupe l'ensemble des utilisateurs qui vont accéder au forum Pour participer aux différents sujets de discussions proposés.

Cette entité permet de sauvegarder toutes les informations concernant les utilisateurs, elle se caractérise par les champs suivants :

- **Cod_uti** : de type entier, il sert à identifier chaque utilisateur, et le différencier des autres, il représente la clé de cette entité.
- **Nom_uti** : nom de l'utilisateur qui accède au forum, il est de type texte.
- **Prenom_uti** : prénom de l'utilisateur de type texte.
- **E-Mail** : représente l'adresse électronique de l'utilisateur.
- **Compte** : l'abonné utilise un compte pour s'authentifier afin d'accéder au forum, il est de type caractère.
- **Mot de passe** : aussi par un mot de passe qui est de type caractère.
- **Profession** : il s'agit de la profession de l'abonné, elle est de type texte.
- **Adresse** : adresse classique de l'abonné.

Entité message :

Considérée comme le cœur de notre base de données, cette entité regroupe l'ensemble des messages et les sujets qui ont été créés par les utilisateurs, ainsi que, les réponses à ces messages.

Chaque message est identifié par un code. L'entité message est engendrée par les attributs suivants :

- **Cod_mess** : pour identifier tous les messages stockés dans la base de données, il est de type numérique.
- **Cod_père** : indique code père d'un message, la procédure est la suivante :

Si c'est un nouveau message, le champ code père aura la valeur 0, sinon, si c'est une réponse à un autre message, le code père dans ce cas aura comme valeur le code de ce dernier message.

Exemple : si un message de code 30 est une réponse à un message de code 25 alors le code père du message de code30 a la valeur 25.

- **cont_mess** : il s'agit du corps qui peut être une réponse ou un nouveau message (cod_pere=0), il est de type caractère.
- **Date_mess** : correspond à la date de création des messages, de type texte.
- **Heure_mess** : indique l'heure de création des messages, de type texte.

Entité thème :

Correspond aux thèmes principaux proposés dans le forum, aux quelles les abonnés doivent créer leurs messages et poser leurs questions. Elle est caractérisée comme suit :

- **Cod_th** : utilisé pour identifier les thèmes, il est de type numérique.
- **Titre_th** : chaque thème à sons propre titre.

Entité sous-thème :

Chaque thème contient un ensemble de sous thèmes proposés par les abonnés, l'entité sous thème regroupe l'ensembles des titres des messages écrits par les abonnés.

Chaque sous_ thème est identifié par un code.

- **Cod_Sth** : numéro donné pour identifier chaque sous thème d'une manière bien définit.
- **Titre_Sth** : chaque message, possède un titre, ce dernier est donner par l'abonnée lors de la création d'un nouveau sujet.
- **Date_Sth** : la date de création d'un sous_ thème.
- **Heure_sth** : est l'heure de création de sous_ thème.

III-3- Messagerie :

Pour le service de messagerie on a utilisé une base de données à une seule table nommée « message » comme suit :

Messages
id_mess
Nom_exp
Email
Objet
Corps
Date

- **Id_mess** : c'est l'attribut clé de cette table il permet de distinguer les messages entre eux.
- **Nom_exp** : le nom de l'expéditeur du message.
- **Email** : l'adresse Email du récepteur du message.
- **Objet** : l'objet du message.
- **Corps** : le corps du message.
- **Date** : la date du message.

III-4- Chat :

Pour le service de chat on a utilisé une base de données à deux tables nommées « utilisateur » et « message » comme suit :

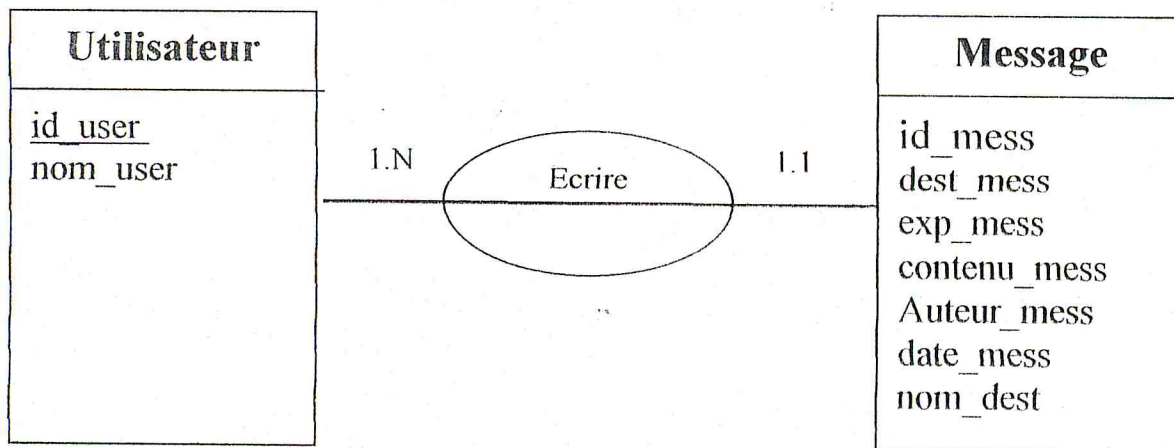


Figure.V.4 : Modèle conceptuel de données (MCD) du Chat

Table utilisateur :

Regroupe tous les noms d'utilisateurs de chat, elle est engendrée par les attributs suivants :

- **id_user** : c'est l'attribut clé de cette table il permet de distinguer les utilisateurs entre eux.
- **nom_user** : le nom de l'utilisateur.

Table message :

Regroupe tous les messages d'utilisateurs de chat, elle contient :

- **id_mess** : c'est l'attribut clé de cette table il permet de distinguer les messages entre eux.
- **dest_mess** : le code de destinataire de message.
- **exp_mess** : le code de l'expéditeur de message.
- **contenu_mess** : le corps de message.
- **Auteur_mess** : le nom de l'expéditeur de message.
- **date_mess** : la date d'édition de message.
- **nom_dest** : le nom de destinataire de message.

IV- Description de l'interface utilisateur :

Il s'agit de concevoir un site WAP dynamique, cette conception permet aux utilisateurs du système, spécialistes dans le domaine médical, et qui ont besoin d'accéder à la plate forme lorsqu'ils sont en déplacement, ou dans les cas d'urgences, pour demander l'aide d'un autre spécialiste dans le domaine. Ce type d'application permet la diffusion d'informations fiables et en temps réel aidant à la prise de décisions rapides et réfléchies.

L'interaction entre le client et le système se fait à travers une *interface* qui offre aux utilisateurs le privilège de formuler aisément leurs requêtes, afin de manipuler les informations médicales de la Base de Données à distance.

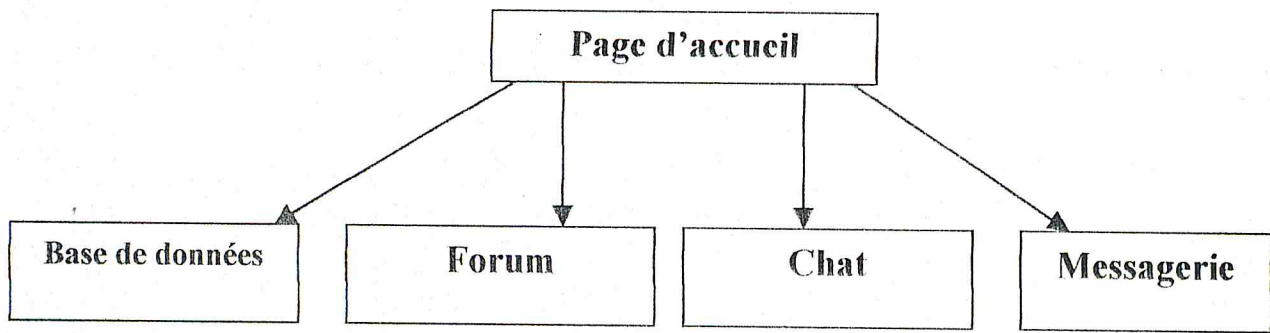


Figure.V.5 : Présentation générale des modules de notre système

IV-1- Le module base de données :

Ce module propose l'accès à une base de données à caractère médical entre les professionnels de la médecine.

Le bon fonctionnement de ce système est assuré par une interface utilisateurs conçue pour être facile à utiliser comme le montre la figure ci-dessus :

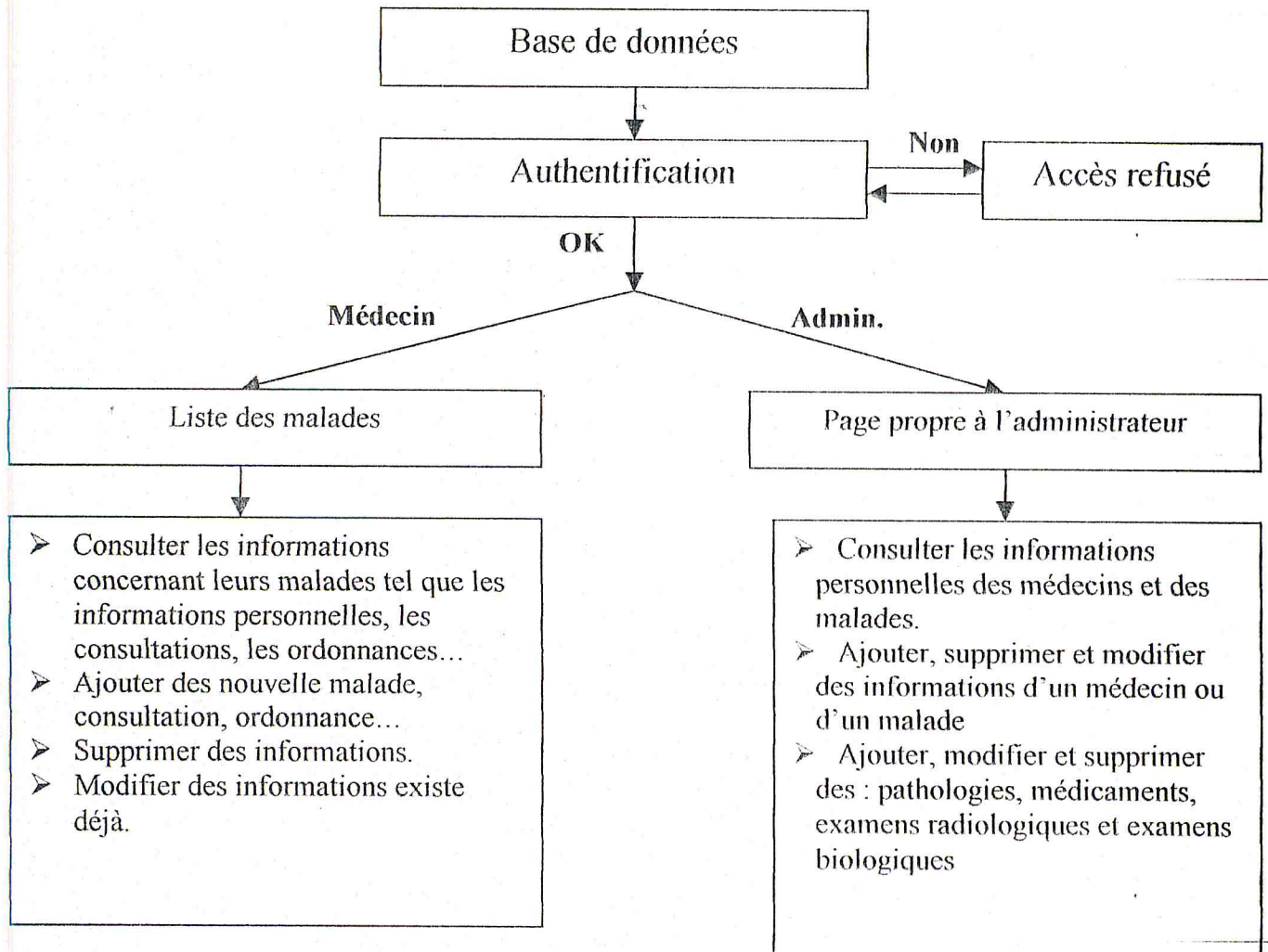


Figure.V.6 : présentation générale des modules de la base de donnée

IV-2- Le module forum de discussion :

Notre forum est formé par des abonnés qui trouvent un moyen très efficace pour échanger leurs informations qui peuvent être des questions ou des réponses en exprimant leurs occupations, leurs demandes, et leur idées.

Il est composé de plusieurs modules qui sont représentés d'une façon générale dans le schéma ci-dessus :

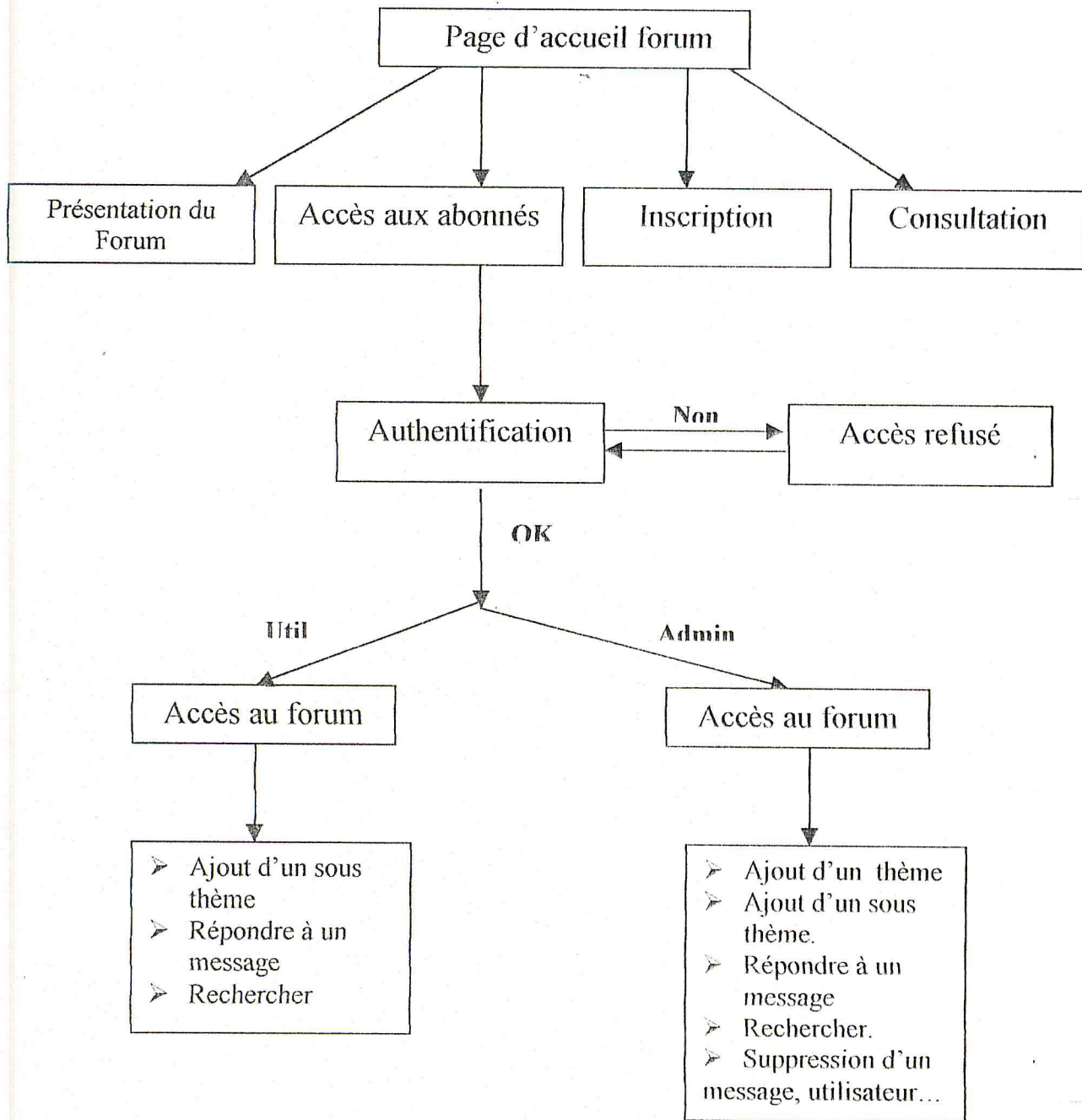


Figure.V.7 : représentation générale des module de forum

IV-3- Le module de messagerie :

Ce module offre aux participant du système la possibilité de communiquer entre eux d'une façon très simple et efficace dans n'importe quelle moment et n'importe où.

Cette fonction est assuré par une interface utilisateur conçue pour être facile à utilisé et soit compatible avec le téléphone portable.

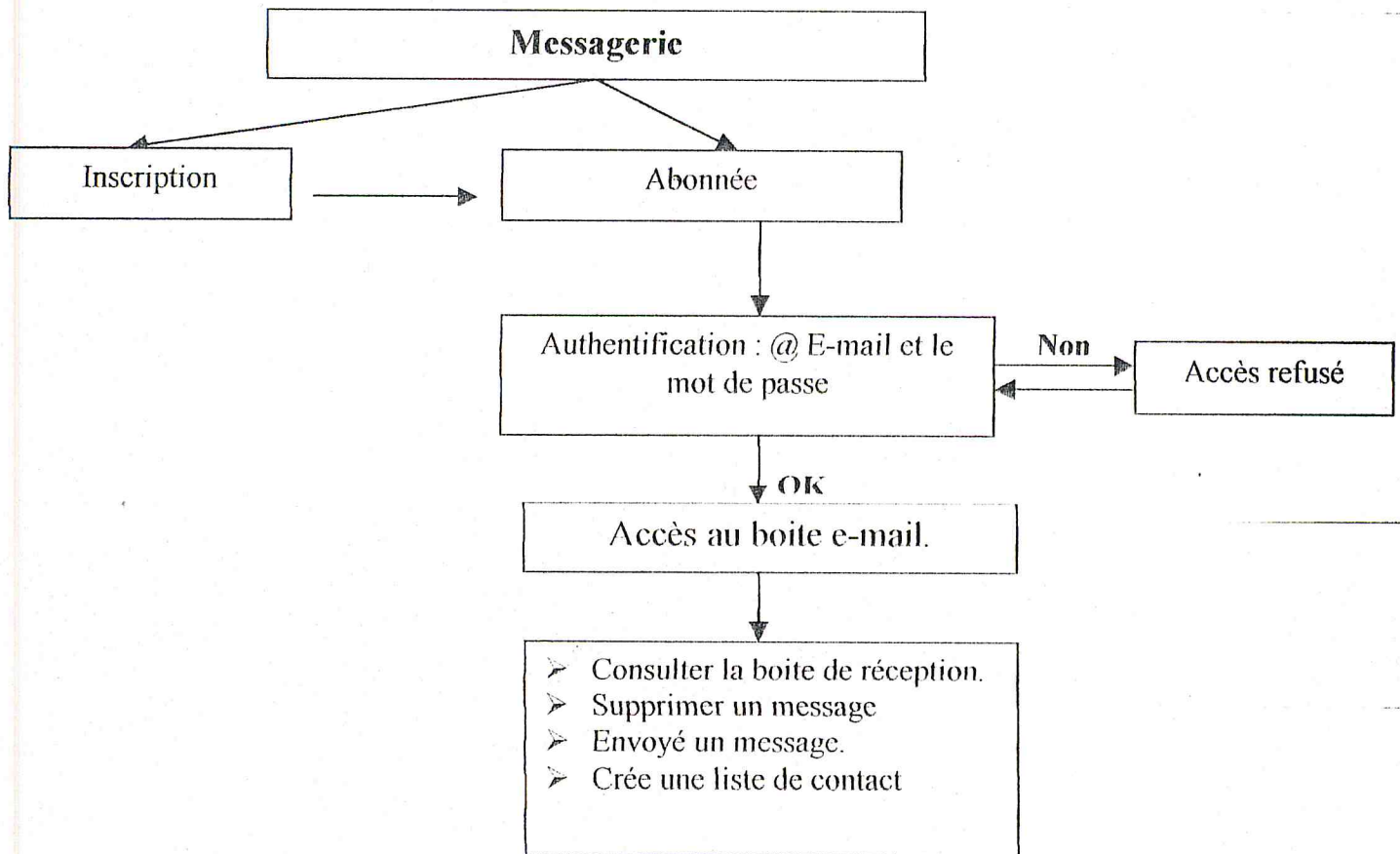


Figure. V.8 : présentation générale du module de messagerie



Chapitre VI

Implémentation et test de l'application

Introduction :

Les caractéristiques du système à développer, observées dans l'étape de conception, imposent un certain nombre de contraintes quant à son implémentation, les principales contraintes à prendre en compte sont :

Les différents internautes doivent pouvoir accéder à la plate forme à n'importe quel moment à partir de leurs terminaux mobiles équipés de micro navigateurs. De ce fait , l'interface utilisateur consiste en un site WAP également constitué de pages statiques et pages dynamiques stockées sur un serveur capable de générer du contenu WAP.

Afin de stocker les données envoyées par les utilisateurs de la plate forme et de pouvoir les consulter par la suite, le système doit être capable d'interagir avec les bases de données. Pour cela, le serveur Web hébergeant le système est appelé à communiquer avec un serveur applicatif contenant un SGBD qui supporte le SQL.

I- Présentation de l'environnement technique :

Le choix de la plate forme Windows s'impose de lui-même. D'une part, il s'agit de la plate forme la plus utilisée sur les PCs d'Internet, d'autre part les outils WAP utilisés ne sont disponibles, pour la plus part d'entre eux, que sur cette plate forme. Le système d'exploitation utilisé est Microsoft Windows 2000 server.

L'environnement technique est décomposé en deux familles : les outils Web et les outils WAP.

I-1- Les outils WEB :

Du point de vue technique, une application WAP est semblable à une application Web. Les seules différences sont le langage WML utilisé au lieu de HTML, et la taille de l'écran du terminal de consultation.

Il est donc nécessaire de passer par un serveur Web capable de délivrer des pages WML, par analogie avec les pages HTML destinées aux navigateurs Web standard.

I-1-1- Serveur Web IIS (Internet Information Serveur) 5.0 :

On a configuré sur un PC pour créer notre application le serveur Web IIS 5.0, ce dernier constitue le service Web de Microsoft Windows 2000 qui facilite la publication d'informations sur un réseau Internet ou Intranet. IIS permet d'intégrer le moteur de script coté serveur PHP.

I-1-2- Le langage de script : PHP (Personal Home Page) :

On peut créer un contenu dynamique indépendant du navigateur en utilisant des script et des composants coté serveur.

On a choisi pour cela le PHP, car il permet d'accéder a l'intégralité des flux de demandes et de réponses http, il fournit une interface de connexion aux bases de données et offre la possibilité de personnaliser le contenu en fonction du navigateur utilisé, la simplicité d'écriture des scripts, et la possibilité d'inclure le script PHP au sein d'une page WML.

I-2- Les outils WAP :

La réalisation d'un site WAP est impossible sur un réseau GSM de faible débit (9,6 kb/s), ce réseau peut transporter des données a l'aide de GSM Data ; cela nécessite que l'opérateur téléphonique mobile modifie son réseau pour faire transiter les contenus WAP.

Etant donné que nos réseaux téléphoniques mobiles n'utilisent pas encore le GSM Data, la connexion terminal mobile, serveur reste impossible. Nous utiliserons alors un simulateur d'interface de terminal mobile WAP pour tester notre application.

I-2-1- Le module Nokia WAP Toolkit 3.1:

L'outil de développement Nokia Toolkit 3.1 est un produit de la société Nokia. Il offre aux utilisateurs la possibilité de créer et tester leurs applications WAP. Ce Toolkit inclut les composants suivants [www4]:

- . Edition WML, WMLScript, et WBMP.
- . Navigateur WML, muni d'un interpréteur et de librairies WMLScript.
- . Simulateur d'interface de téléphone WAP, le Nokia 7210 et le blueprint.
- . Un débogueur.
- . Piles de protocole WAP et http et le module d'accès aux fichier.
- . Un simulateur de serveur WAP.
- . Une passerelle WAP configurée par défaut.

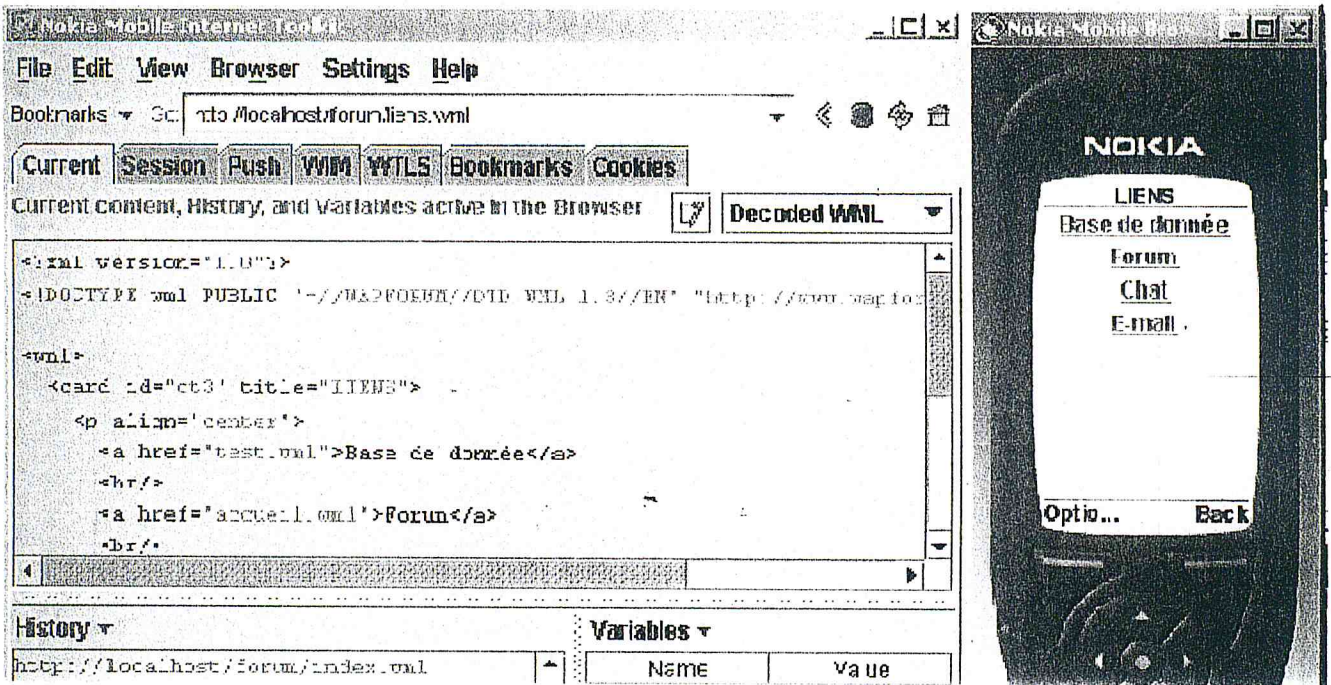


Figure.VI. 1 : le module Nokia WAP Toolkit 3.1.

I-2-2- La passerelle WAP (WAPLite):

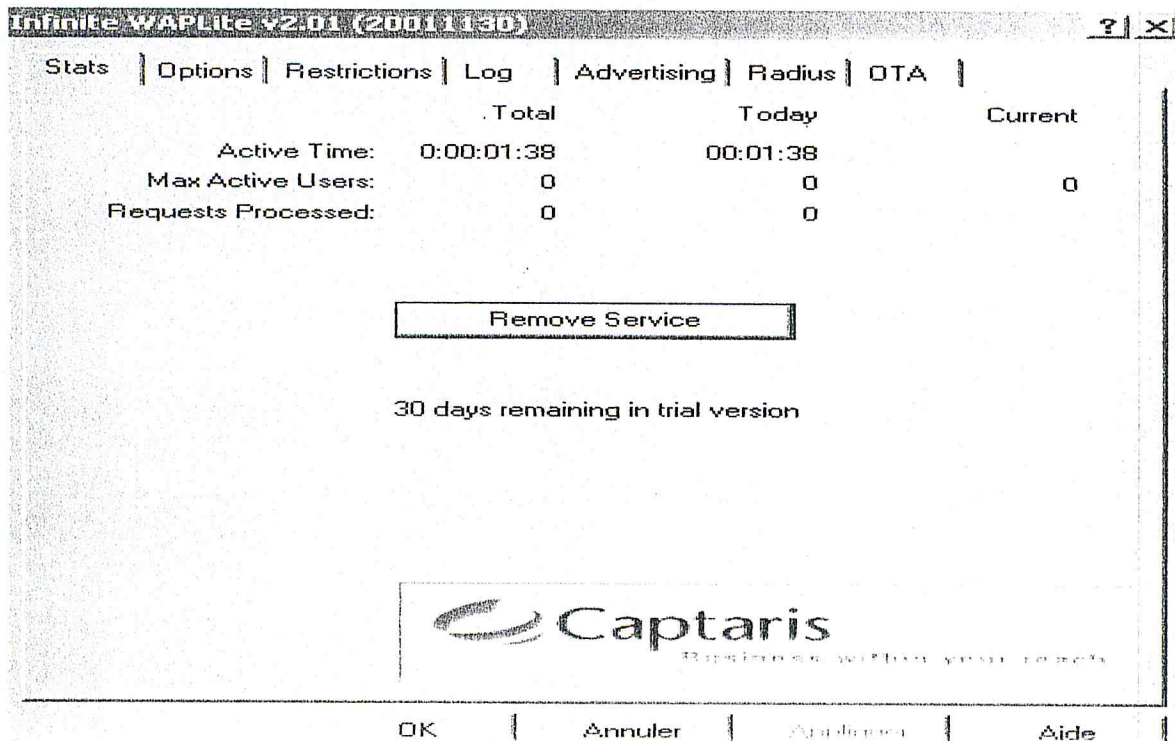


Figure.VI. 2 : La passerelle WAPLite d'infinité technologie.

La passerelle WAPLite d'Infiniété technologie, elle assure la liaison entre le monde de l'Internet mobile, elle s'adresse aux entreprises désirant offrir une solution WAP a ses employés. WAPLite supporte la connexion aux outils de développement WAP compatible WML

I-2-3- L'éditeur de page WML:

Il existe plusieurs éditeurs de page WML, comme par exemple «WAPsite», qui permet l'édition des pages WML avec l'intégration des script PHP, les scripts de WMLScript, ainsi que les images WBMP, et pour la rédaction de nos pages wml, nous avons choisi l'éditeur proposé par Nokia Toolkit (Toolkit Editors).

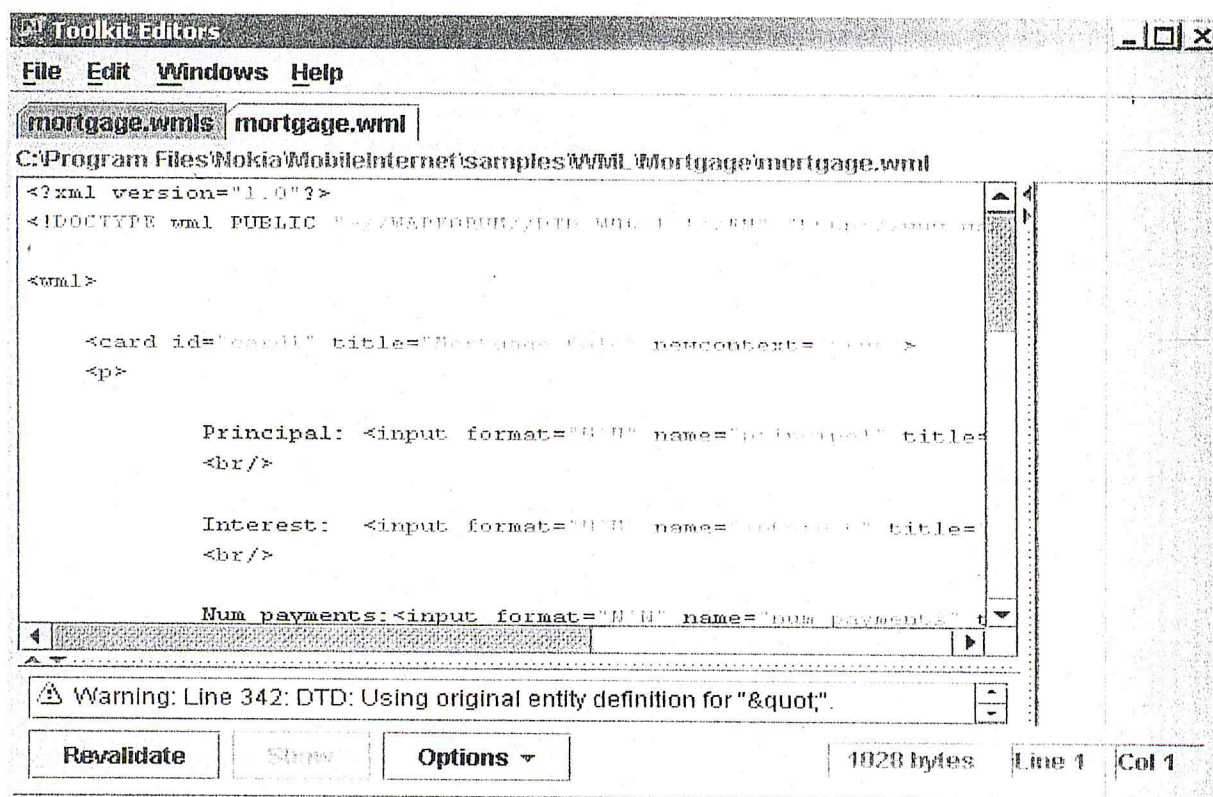


Figure.VI. 3 : L'éditeur de texte de Toolkit 3.1.

I-2-4- Outils graphique WAP:

ccGraphic est un convertisseur d'image au format WBMP. Il permet de modifier la taille et la luminosité de l'image pour qu'elle puisse être affichée sur l'écran de téléphone mobile.

I-3- L'implémentation de la base de données:

Les SGBD constituent un maillon essentiel dans l'architecture des sites WEB et WAP dynamique. Leur rôle est le stockage des données d'un système d'information.

SQL serveur 7.0 est le SGBD utilisé pour l'implémentation de notre base de données.

On a opté à l'utilisé grâce a sa parfaite intégration avec le système d'exploitation Windows NT server, qui est installé sur notre machine, il détient des records de performances en termes d'adaptabilité et de fiabilité et aussi de vitesse de développement d'application ou de traitement des transactions.

SQL Server est un système de stockage de données sous forme de tableaux relationnels, il admet des commandes en langage SQL et peut recevoir des connexions de clients qui veulent manipuler des données sur n'importe quel ordinateur connecté a Internet, et aussi il permet la programmation client/serveur.

II- Etude technique détaillée:

II-1- Configuration du post client:

Après l'installation du Nokia Toolkit, la configuration est nécessaire pour assurer l'échange de données entre le client et le serveur.

Pour paramétrer le logiciel, il faut choisir l'icône *Toolkit* et cliquer sur l'option *Device Setting*, la boite de dialogue suivante apparaît :

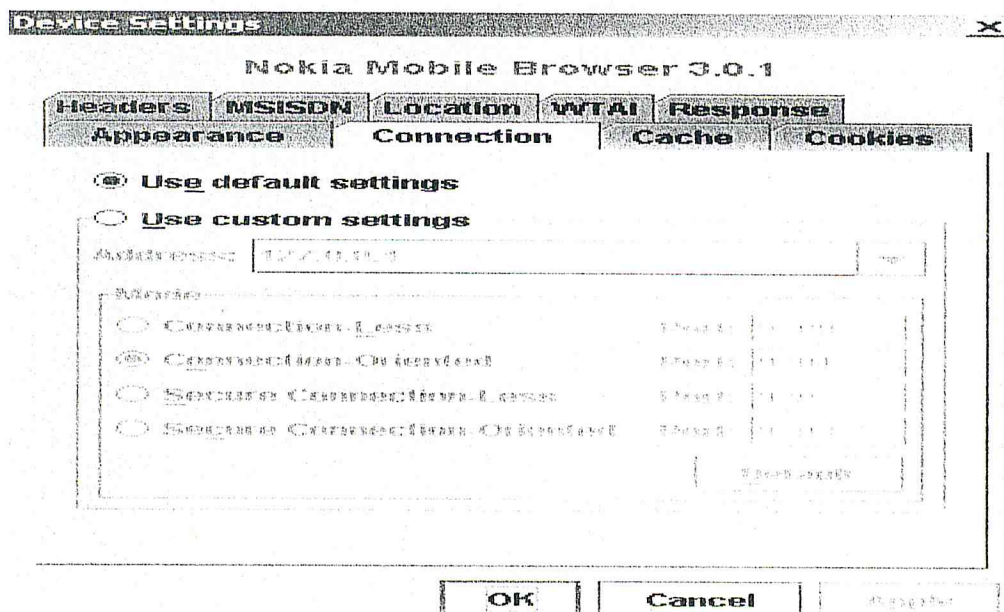


Figure.VI. 4 : Configuration de Toolkit.

Cette boîte de dialogue propose deux mode de connexion :

- Use default settings : le Toolkit est configuré pour afficher un contenu WML issue d'un serveur Web en mode http en utilisant la passerelle WAP proposée par défaut par le Toolkit.
- Use custom settings : le Toolkit est configuré pour afficher un contenu WML issue d'un serveur Web, à travers une passerelle WAP configurée sur la machine (comme la passerelle WAPLite d'infinité technologie par exemple).

III- Configuration du serveur WEB/WAP :

III-1- répertoire virtuel :

Pour créer un site Web, il faut d'abord créer un répertoire virtuel sur le serveur Web de telle sorte que les utilisateurs puissent établir une connexion http et afficher les pages web.

Pour cela, nous avons assemblé tous nos fichier (WML, WMLScript, WBMP, PHP) dans un répertoire sur le réseau et nous avons créé un répertoire virtuel qui pointe vers ce répertoire pour pouvoir afficher nos pages sur le navigateur WAP.

III-2- Les type mimes :

Lorsque le serveur Web renvoie un fichier à un navigateur, il indique au navigateur le type du contenu du fichier.

Cela permet au navigateur de déterminer s'il peut afficher le fichier lui-même ou s'il doit faire appel à une autre application. Le serveur Web reconnaît les types de fichiers en mappant leurs extensions vers une liste de types MIME (Multipurpose Internet Mail Extensions) existant dans la propriété « Type Fichiers » du répertoire virtuel.

Pour ce faire, nous devons ajouter les différents type MIME associées à une application WAP, dans l'onglet entête http de IIS comme l'illustre la figure suivante :

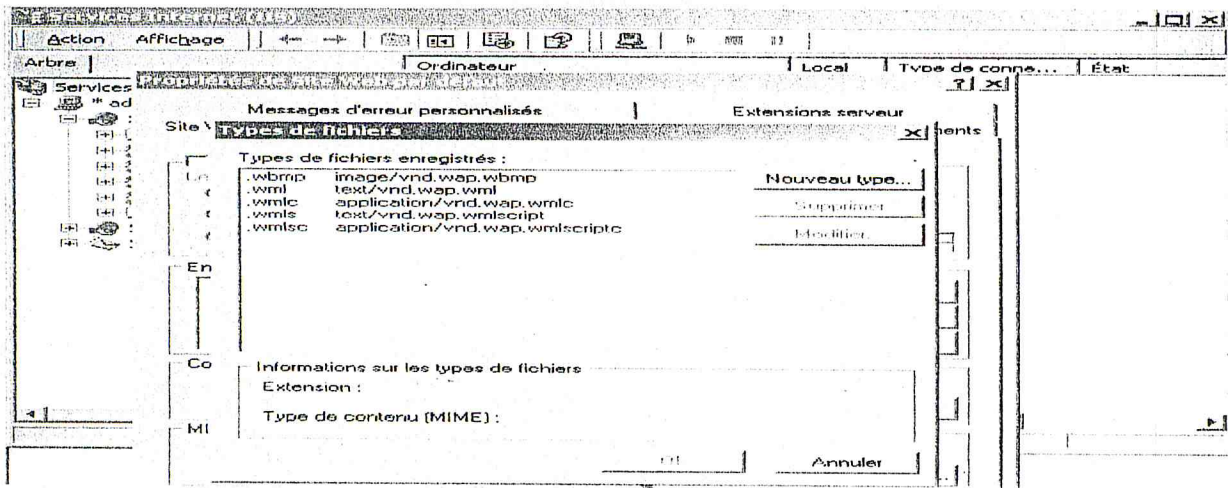


Figure.VI.5 : configuration des types mime.

III-3- Le script PHP :

L'installation du PHP dans un serveur Web IIS, se fait manuellement sur le serveur, mais après l'installation il faut :

1. indiquer à IIS que les fichiers dont l'extension est « .php » devront être traités par PHP.
2. arrêter puis redémarrer le serveur web.

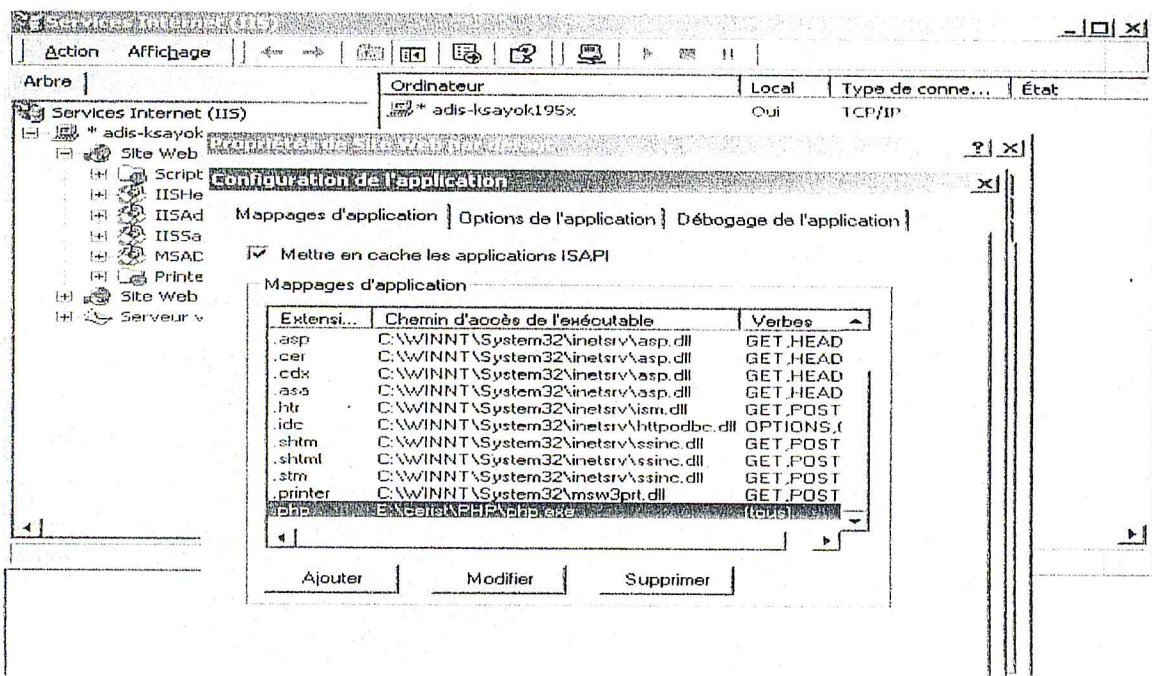


figure.VI.6 : filtre ISAPI.

IV-4- Le module de chat :

Ce module permet aux utilisateurs de communiquer entre eux directement en échangeant des messages, pour cet effet nous avons conçu un interface utilisateur simple pour assurer la facilité d'utilisation et l'efficacité de chat au même temps.

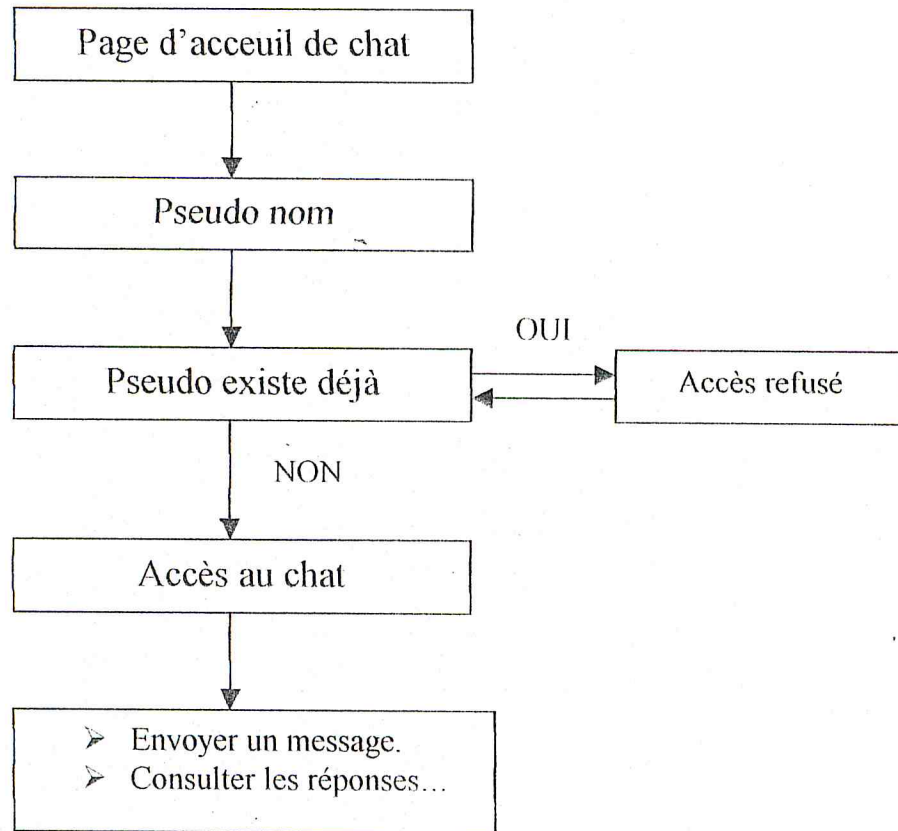


Figure.V.9 : présentation générale de chat

Conclusion :

En générale, pour le développement d'une application (WEB ou WAP) dynamique, le développeur doit prendre en compte l'efficacité de l'application pour répondre au maximum aux besoins des utilisateurs, pour leur offrir une interface simple, conviviale et facile à manipuler. Dans ce sens, nous avons essayé de respecter ces deux cotés : l'efficacité et l'ergonomie lors du développement de notre application, que nous allons présenter en détails dans le chapitre suivant.

Pour qu'un navigateur WAP puisse interpréter une page PHP résultant du script, il faut indiquer au serveur le type de contenu qui doit être transmis au navigateur WAP, grâce à l'entête spécifiée au début de page PHP :

```
<? header ("content-type: text/vnd.wap.wml"); ?>
```

III-4- Création de la source de données système (DSN):

Une source de données système désigne la base de données dans laquelle on peut récupérer des données. Dans la mesure où la connexion se fait via des méthodes ODBC, on peut utiliser tout système de gestion de base de données disposant d'un pilote ODBC.

Quand on configure un serveur pour accéder à la base de données via ODBC, il faut créer au préalable un nom de source de données qui est relié à la base de données. De cette façon, tout script créé s'exécutera correctement

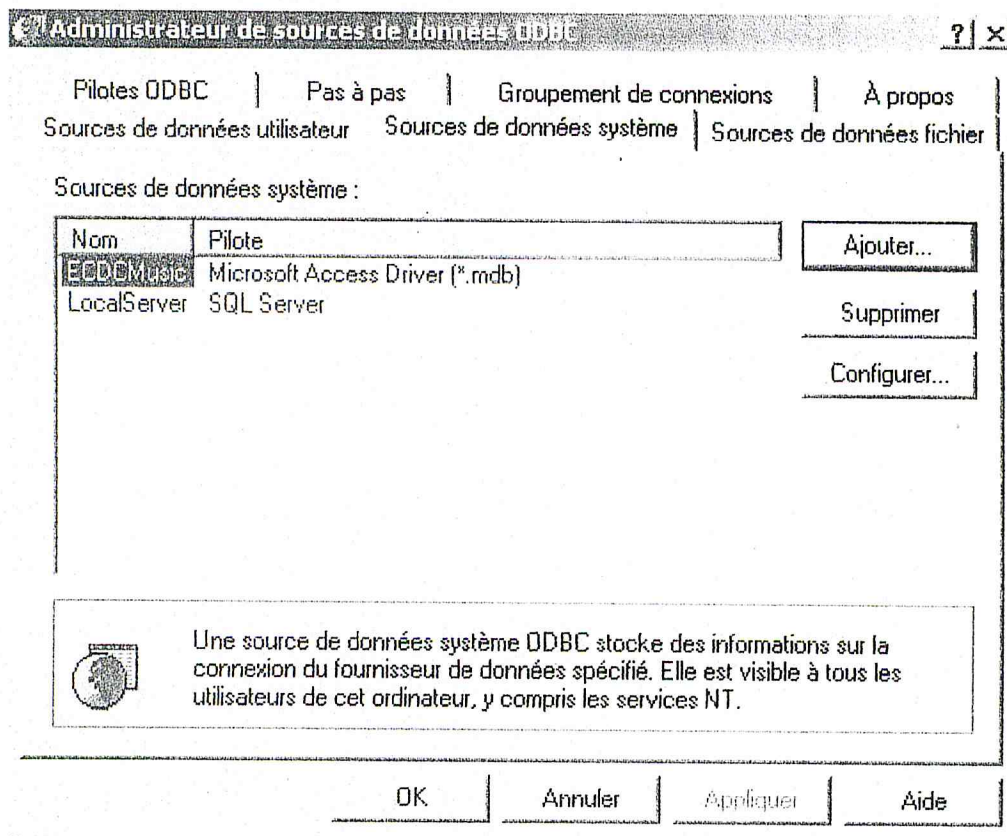


Figure.VI.7 : Source de données système

IV- L'application :**IV-1- Analyse fonctionnelle :**

Comme la quasi-totalité des applications Internet, notre application WAP est constituée d'une suite d'écrans statiques et dynamiques.

Elle débute par une page d'accueil, puis l'accès directement aux différents pages de notre application ; également des menus, des boîtes de saisie et des pages d'affichage ; et aussi des liens sur chaque cartes pour faciliter aux utilisateurs la navigation entre les différentes cartes et les decks qui constitue l'application.

Processus d'exécution d'une requête :

- pour la consultation du notre site, l'utilisateur doit saisir l'URL du site.
- Une requête est envoyée à la passerelle WAP (WAPLite ou à celle proposé par le Toolkit) en utilisant le protocole WAP. Cette dernière, assure la conversion de cette requête en format http et de la transmet au serveur web.
- Si la requête concerne une page statique (.WML), le serveur recherche simplement le document, indique au navigateur WAP qu'il va recevoir une réponse au format http.
- Si la requête concerne une page dynamique (.PHP), le moteur de script PHP se charge d'établir la connexion à la base de données via une source de données ODBC, d'exécuter la requête SQL et récupérer le résultat, ainsi le serveur envoie le code résultant à la passerelle au format http.
- La passerelle vérifie l'entête http et le contenu WML, et code le tout dans un format binaire.
- La passerelle crée alors une réponse au format WAP qui est transmise au navigateur WAP. Le navigateur reçoit une réponse. Il interprète le contenu WML et affiche la première carte du deck.

IV-2- Test de l'application :

Le projet 'téléconsultation médicale' propose l'accès à une base de données à caractère médical entre les professionnels de la médecine.

L'accès en question favorise des échanges de données (Informations sur des patients), en vue d'un meilleur diagnostic.

Les professionnels de la médecine trouveront, avec ce projet, un moyen de Communiquer à travers un service de messagerie et de Chatting (Chat).

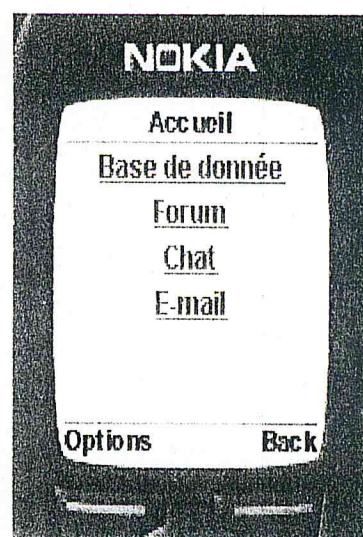
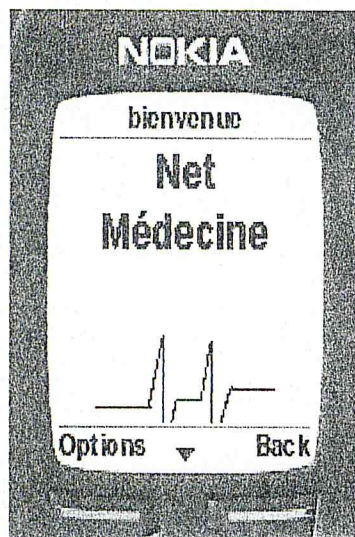
En premier lieu une fois que l'utilisateur a saisi l'URL du site dans son navigateur, ce dernier affiche la page d'accueil envoyée par le serveur. Celle-ci contient plusieurs liens vers d'autres pages. Certaines sont statiques (.WML) et d'autres dynamiques avec l'extension .PHP.

La page d'accueil est une page WML statique, elle donne l'accès à différentes cartes selon les utilisateurs.

La page d'accueil du système 'Télé Consultation Médicale' comporte les points ci après :

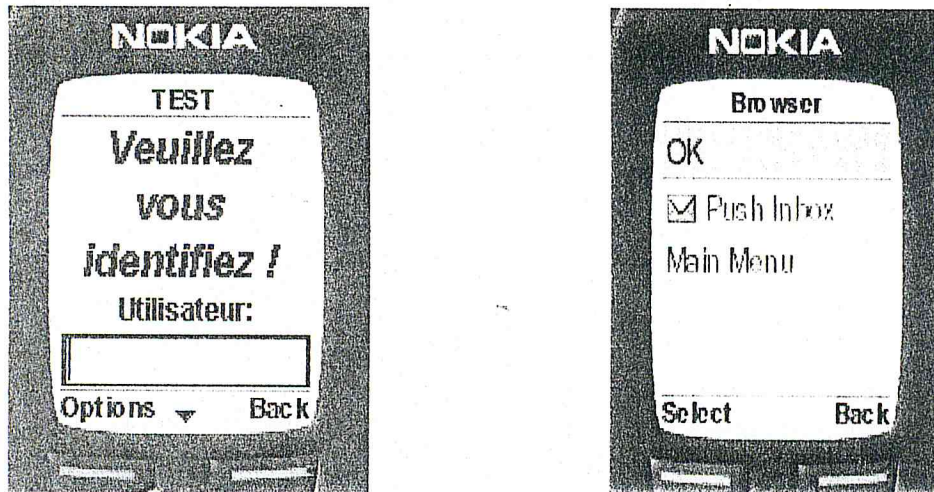
- Une Base de données médicale.
- Un Forum entre les professionnels de la médecine.
- Un Chat.
- Une Messagerie.

Notre système démarre par une image suivie par un autre page qui reste 3 seconds, suivi par un page d'accueil :



IV-2-1- Accès a la base de données :

L'interface ci-dessous permet aux utilisateurs de s'identifier, à travers un nom d'utilisateur et un mot de passe, afin d'accéder à la Base de Données.



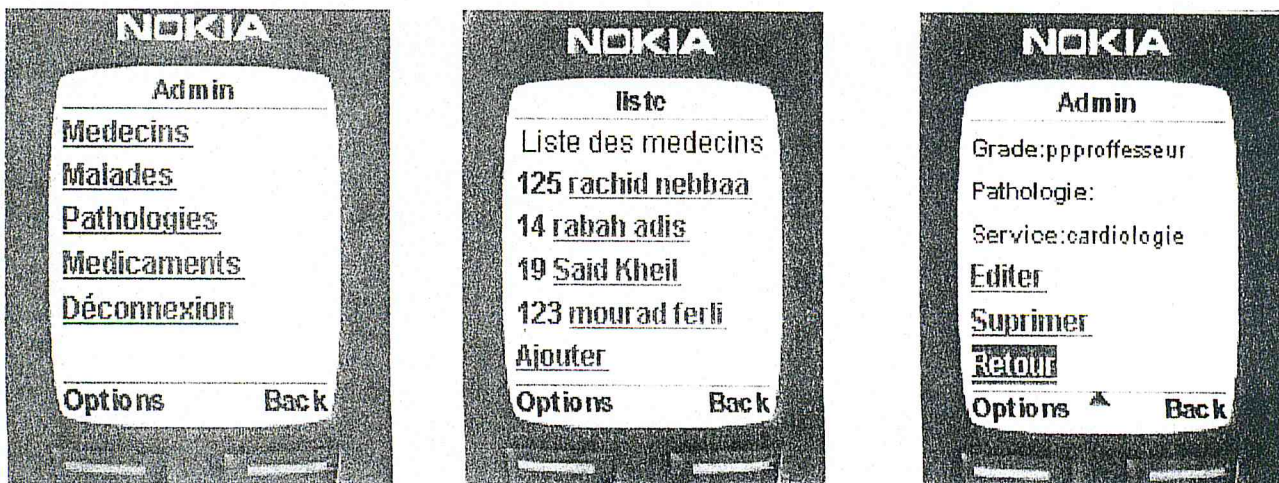
Dans ce système d'accès à la Base de Données, deux types d'utilisateurs peuvent y accéder :

- Administrateur : Utilisateur ayant le privilège d'accéder à toute la base de données médicales pour consultation et mise à jour.
- Médecin : Utilisateur ayant le privilège d'accéder, pour consultation et mise à jour, uniquement aux informations se rapportant à ses patients.

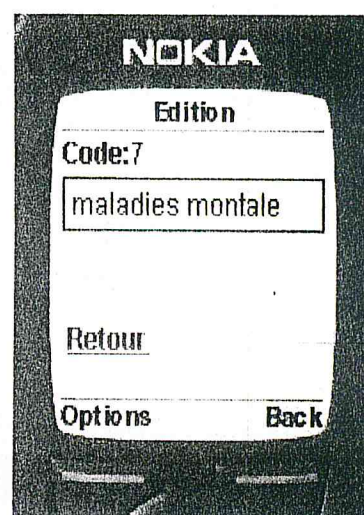
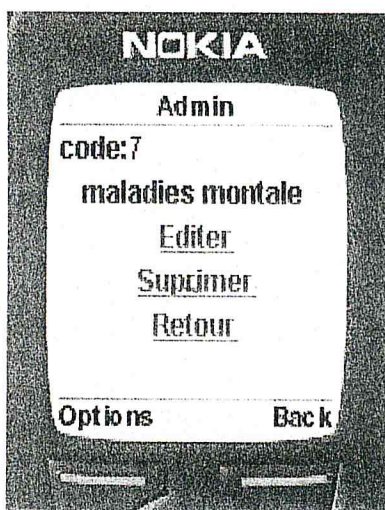
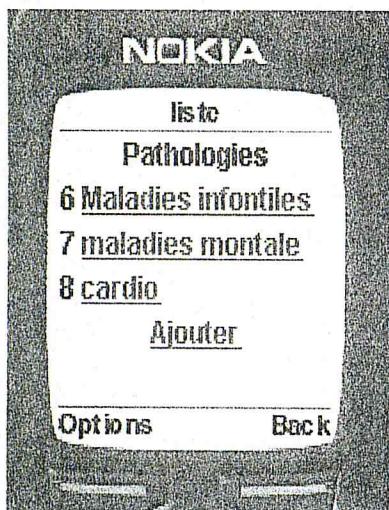
Après authentification de l'utilisateur, à travers son nom d'utilisateur et son mot de passe, deux cas de figure sont présentés :

L'utilisateur est un Administrateur :

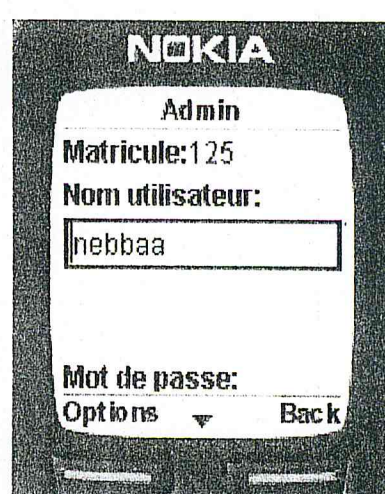
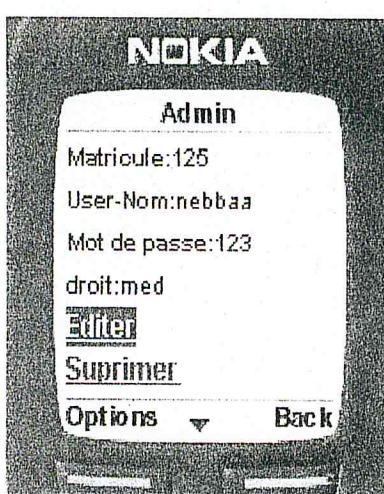
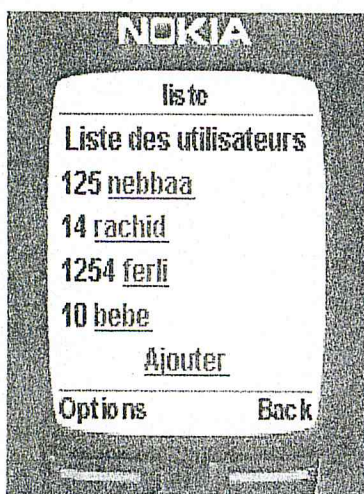
L'Administrateur a tous les Droits de consultation et de mise à jour de la base de Données médicales. Il a le privilège de lister tous les médecins répertoriés dans la base de données. Il peut consulter, éditer et mettre à jour les informations relatives à tous les médecins, sans aucune restriction.



L'Administrateur peut également avoir accès à toutes les tables en rapport avec l'administration de la Base de Données (Pathologies, médicaments, Examens biologiques et radiologiques, etc....).



Ultime privilège, l'administrateur a la faculté d'accéder à la table Sécurité afin d'administrer les comptes utilisateurs et leur mot de passe, en apportant des mises à jour.

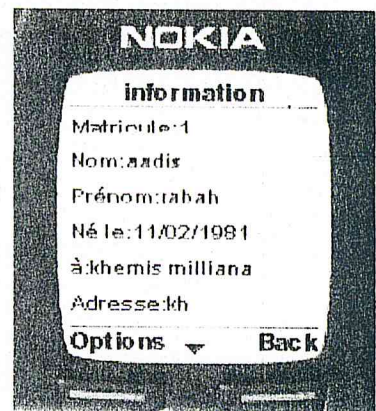
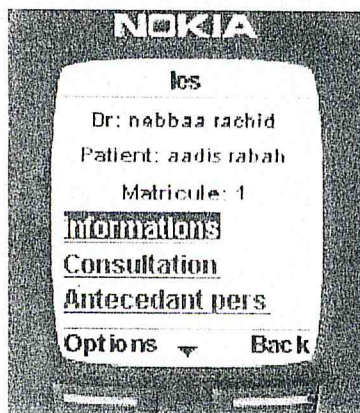
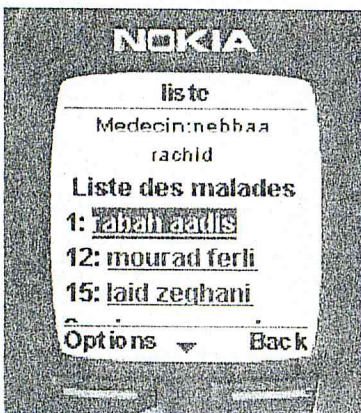
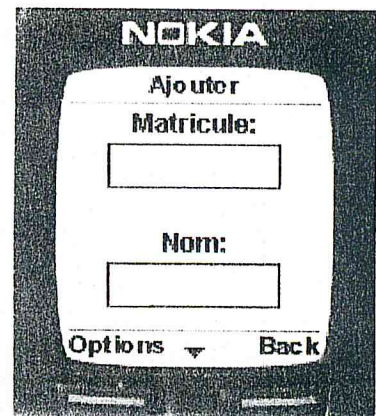
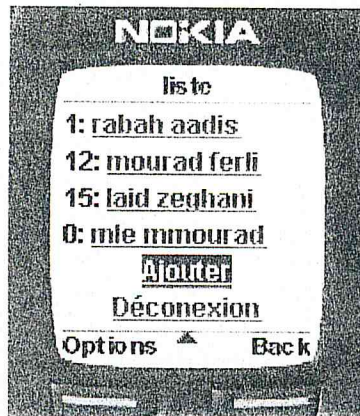


L'utilisateur est un Médecin :

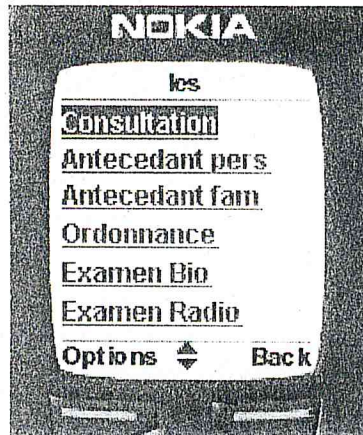
Le médecin ne peut accéder, pour consultation et mise à jour, quant aux Informations se rapportant à ses propres patients.

Il peut, lors d'une nouvelle consultation, ajouter un nouveau patient avec toutes les données qui le concernent.

Pour un patient donné, le Médecin peut voir ses antécédents personnels et familiaux, ses consultations, ses ordonnances et leur contenu, les examens biologiques et radiologiques subis.



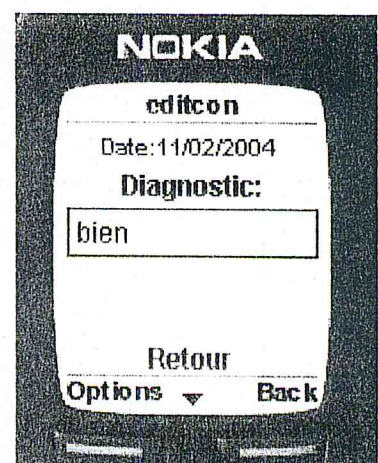
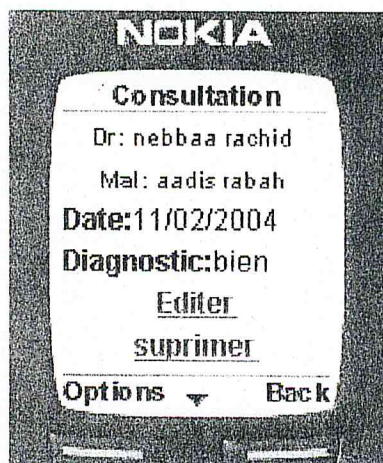
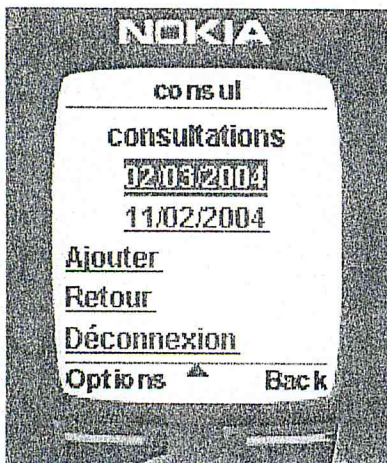
A titre d'exemple, prenant la Table 'consultation' :



Le médecin peut lister la liste de toutes les consultations fait par un malade, ainsi, il a une possibilité d'ajouté une nouvelle consultation.

Le contenu de cette table peut être mis à jour en cliquant sur le lien *Editer*.

En cas de mauvaise manipulation des données, on peut les supprimer en cliquant sur le lien *Supprimer*.



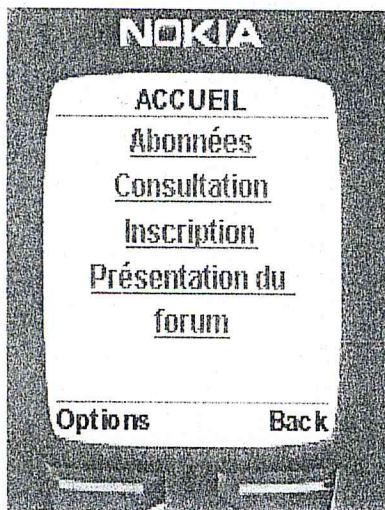
IV-2-2- Le module de 'Forum de discussion' :

Ce module offre la possibilité aux utilisateurs de communiquer et d'échanger des idées sur des sujets multiples.

Le forum est composé de plusieurs sujets, autour desquels des questions sont posées par les participants. Les visiteurs du site fournissent des réponses à ces questions et proposent d'autres sous thèmes. Les thèmes sont créés par l'*administrateur*.

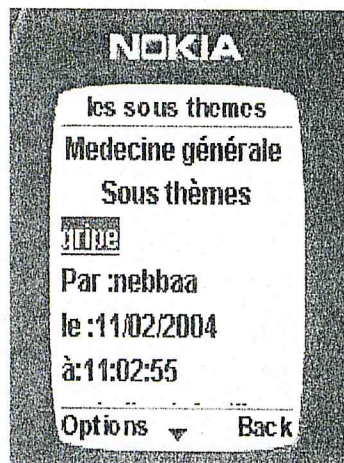
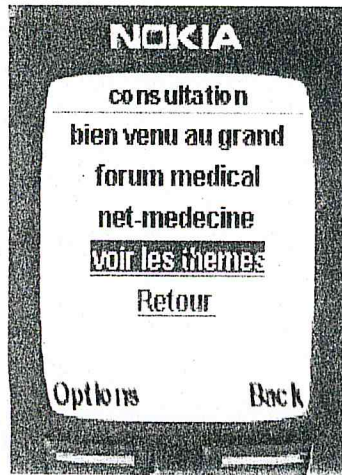
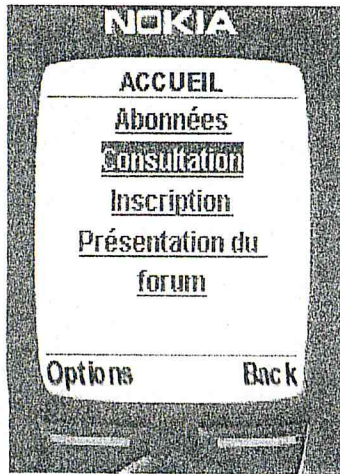
La page d'accueil du forum est une page WML statique, elle donne l'accès à différents modules du forum :

- Abonné : Comprend le lien vers la page authentification.
- Consultation : Contient le lien direct vers la page d'accueil du forum.
- Inscription : Dont le lien est vers le formulaire d'inscription.



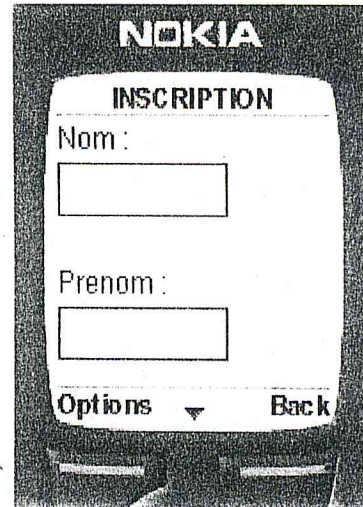
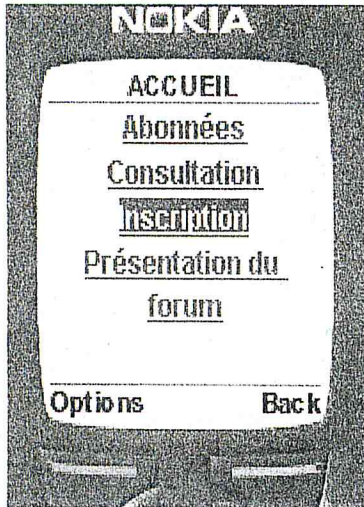
L'utilisateur non abonné et non inscrit peut seulement consulter le forum en cliquant sur le lien consultation.

La page d'accueil du forum apparaîtra, en sélectionnant voir les thèmes qui est sous forme de lien, la liste des thèmes en lien sera visualisée, puis à travers chaque thème l'utilisateur pourra consulter les sous thèmes existant pour chaque thème ainsi que le corps du message avec l'auteur, le titre, la date, l'heure de l'envoi, et les réponses appropriées selon les fenêtres suivantes :

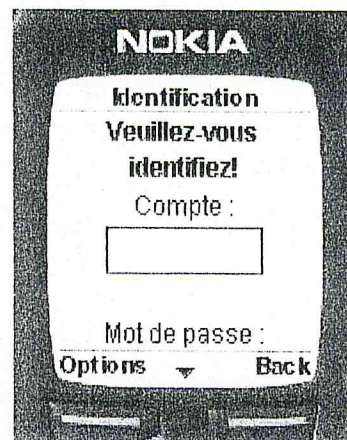
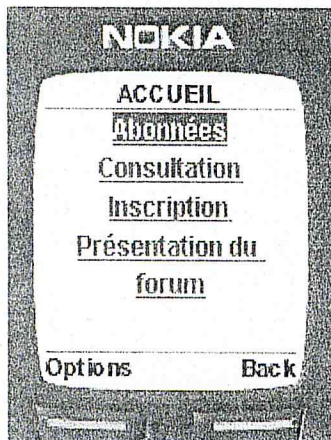


Pour devenir membre, l'utilisateur clique sur le lien inscription de la page d'accueil. La page contenant le formulaire d'inscription qui est une page (.PHP) s'affichera, il saisira les renseignements le concernant.

Un script est lancé pour se connecter à la base de données et insérer le nouvel enregistrement, ensuite afficher au client un message « cliquer ici » pour accéder au forum, c'est-à-dire qu'il a été enregistré.

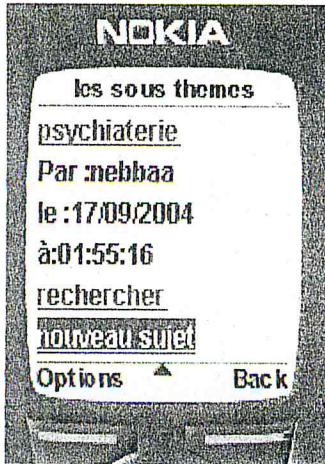


Une fois que l'utilisateur s'est abonné, il accède au forum à travers le lien Abonné, un formulaire lui permet s'identifier avec son compte et son mot passe. Après que l'abonné s'est authentifié, une page statique s'affiche dans son navigateur, un lien apparaîtra qui est « voir les thèmes » envoie vers une autre page dynamique et ainsi de suite :

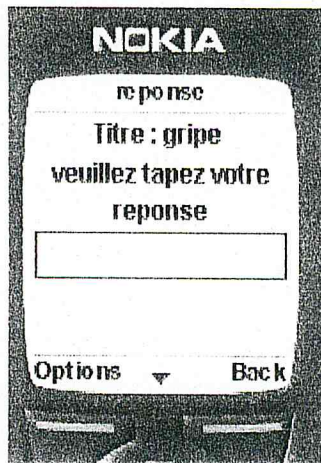


Pour le lien « voir les thèmes » quand l'abonné clique dessus, un script PHP est lancé pour extraire les données de la base de données, et construire la page dynamique d'extension (.PHP). Cette page est envoyée à l'abonné, elle comporte la liste des thèmes qui existe sur notre forum sous forme de liens (vue précédemment). Cette page comporte aussi un lien recherche, pour lancer une recherche dans le forum le résultat de cette recherche est visualisé comme le montre les figures suivantes :

L'abonné sélectionne un thème il visualisera dans son navigateur une page dynamique, cette page comporte une liste des sous thèmes affichés sous forme de liens, ainsi q'un bouton « nouveau sujet » pour un nouveau sous thème, une fois que l'utilisateur ait saisi un nouveau message et ait cliqué sur envoyer, le message est inséré automatiquement dans la base de données grâce à un script PHP, et un message apparaîtra sur le navigateur (votre message est inséré).

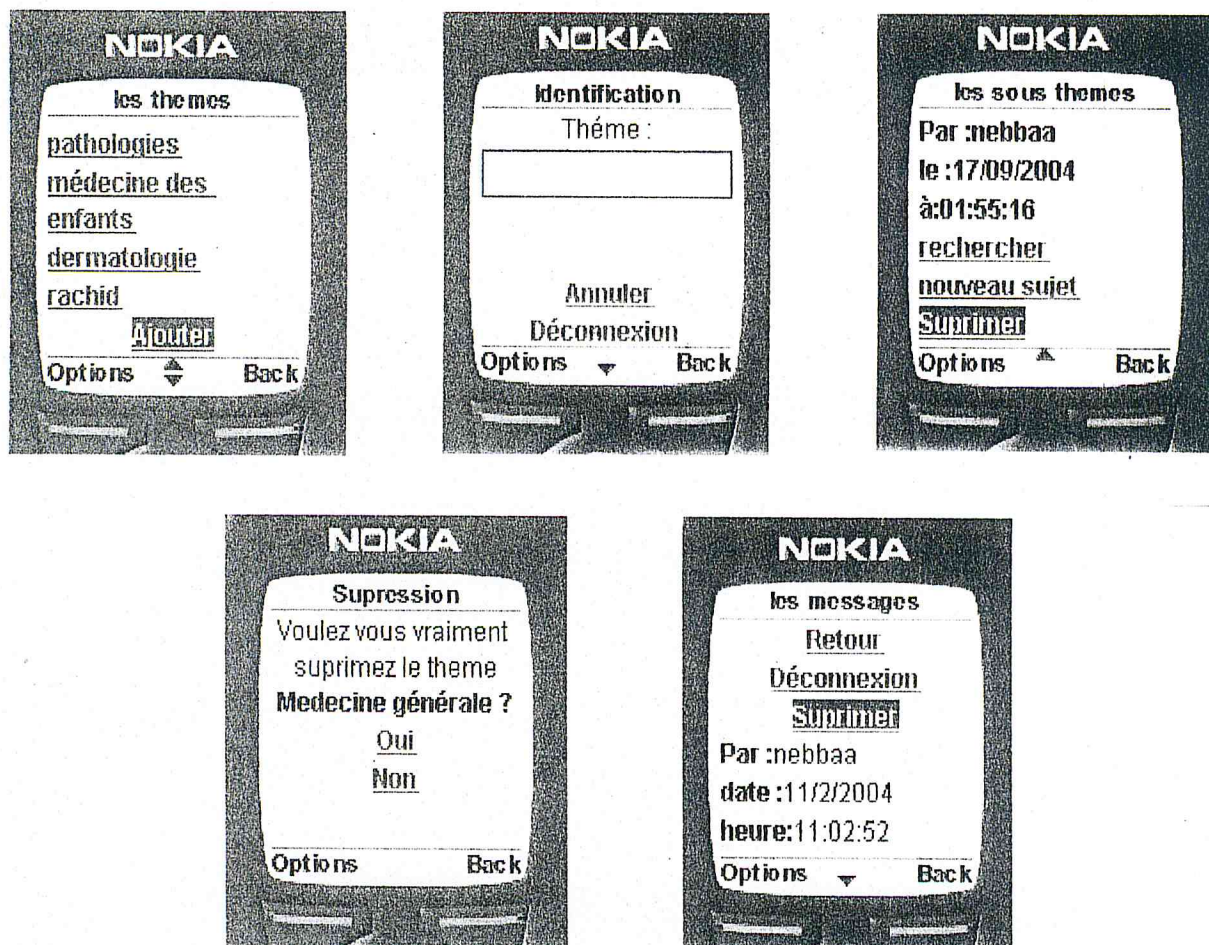


L'abonné peut répondre à un sujet, il sélectionne un sous thème voulu qui est sous forme de lien, une page dynamique apparaîtra pour saisir la réponse et l'envoyer avec le bouton envoyer comme suit :



Et en fin le lien présentation du forum permet aux utilisateur de découvrir le but et objectif de notre forum.

Pour un administrateur toutes les fonctions sont les même avec la possibilité d'ajouter ou supprimer un thème, sous thème, ou une message.



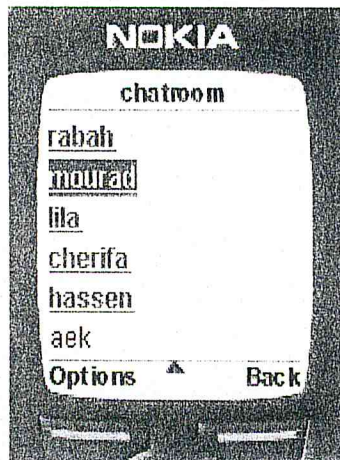
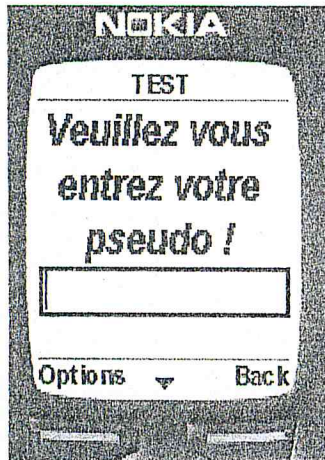
IV-2-3- Accès au 'chat' :

Le module de Chat permet aux différents utilisateurs d'échanger des messages, en direct, pour mener des discussions sur des sujets en rapport avec l'activité médicale. On peut citer comme sujet, les consultations, les conseils et recommandations, les clarifications, etc

Le mot chat (prononcé en anglais *chatter*) n'a pas de traduction française satisfaisante. Il désigne une discussion en ligne en temps réel.

L'utilisateur entre dans le salon de chat (chatroom) après avoir saisi un pseudo qui permet l'identifier dans le salon dans ce cas une page constituée, affiche la liste des utilisateurs participant aux chats sous forme de liens. L'utilisateur choisi un nom, une nouvelle page affichera comprend un espace de saisie de message, le nom de récepteur de ce message, et un bouton d'envoi du message.

Le rafraîchissement de la liste des messages et de la liste des utilisateurs connectés est automatique, il n'est pas nécessaire de la demander explicitement par un bouton.



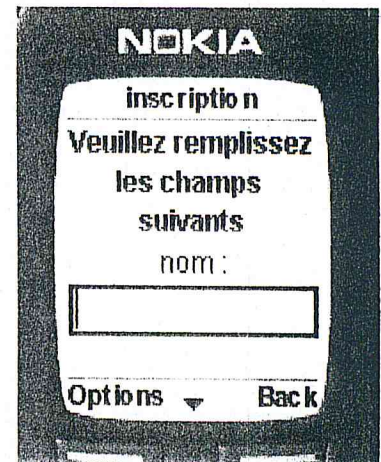
IV-2-4- Le module de messagerie :

Ce service offre au médecin toute les techniques liées à la messagerie (ouvrir une boîte aux lettres électroniques, écrire un message, lire un message, supprimer un message...).

Dans ce module le médecin retrouve la totalité des messages qui lui ont été envoyés par des autres médecins ou par l'administration, et pour chaque message il trouvera les informations suivantes :

- Le nom et le prénom de l'expéditeur.
- Le sujet du message sous forme de liens vers le contenu du message.
- La date d'envoi du message.





Conclusion :

L'implémentation de notre application nous a permis de collecter divers notions sur Internet, la téléphonie mobile et surtout la communication entre ces deux réseaux via le protocole WAP, pour cela en utilisant le langage WML et la technique PHP afin de générer des pages Web dynamiques dédiées à un terminal mobile issues de données stockées sur une base de données et pour faciliter aux utilisateurs la navigation dans notre application et lui fournir une meilleur manipulation d'un site WAP dynamique.

Conclusion générale



CONCLUSION GENERALE

Le travail qui nous a été confié par le Centre de Développement des Technologies Avancées le (CDTA), porte sur la conception et la réalisation d'un système de téléconsultation médicale en utilisant la technologie WAP.

Ce système a été conçu dans le but de répondre à un besoin précis, permettre aux médecins accédant à Internet dans un contexte de mobilité, d'échanger des informations entre eux, d'effectuer des consultations ou d'analyser et interpréter des examens complémentaires, de partager leurs idées et leurs savoir faire sous forme de messages électroniques d'une façon simple et rapide.

Ce projet nous a été bénéfique car il nous permis d'enrichir nous connaissance dans le domaine des technologies Internet et la téléphonie mobile, et d'étudier en détail le standard d'Internet mobile à savoir le protocole WAP.

Le fruit de cette étude est la conception et la réalisation d'une application de téléconsultation médicale via le WAP, ce qui nous a permis de tester pratiquement les connaissances acquises dans le domaine et de se familiariser avec les langages WML, WMLScript et PHP.

Après avoir fait une étude comparative sur les déférentes technologies d'accès aux bases de données (PHP, ASP, ADO,...) et après avoir étudier la technologie WAP, nous avons proposé une solution permettant :

- Accès aux données via le mobile à n'importe quelle moment.
- Traitement facile est efficace des données médicales.
- Application standard supportable par la plupart des navigateurs WAP.
- La communication entre les médecins à tout moment ce qui permet l'échange des connaissances et de savoir faire entre eux.
- Faire des diagnostics sans besoins de la présence physique du malade.

En fin, notre projet reste ouvert pour des améliorations, pour ça nous proposons les perspectives suivantes :

- La nouvelle génération des téléphones portables supporte divers formats d'images avec une haute résolution qui offre la possibilité de traitement des images médicales.
- Les nouveaux navigateurs WAP supporte le langage XHTML qui offre plus d'options pour l'interface utilisateur.

L'application réalisée représente une modeste étude dans le domaine proposé, espérant qu'elle peut être considérée comme un outil assez complet pouvant répondre aux exigences des utilisateurs.

Annexe

Les balises WML :

Les balises WML sont organisées en groupes selon leurs fonctions dans le deck :

Les balises d'entête :

<wml> : décrit une pile WML c'est-à-dire une collection de cartes.

<head> : Spécifie des informations valables pour tout le deck. Les balises incluses dans ce bloc ne sont pas affichées par le navigateur.

<access> : En l'absence de cet élément, tous les decks sont déclarés "public", c'est-à-dire que les autres cartes appartenant à d'autres decks peuvent accéder aux cartes du deck courant.

<meta/> [WAP 01] : La balise méta indique des informations sur la plate-forme et l'affichage au browser des pages WML.

<card> : Les cartes sont au cœur d'un programme WAP. Une carte est un ensemble d'informations de formatage, de contenu affichable et d'instructions de traitement. Chaque carte d'un deck doit contenir un ou plusieurs éléments.

Exemple :

```
<?xml version="1.0" ?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
" http://www.wapforum.org/DTD/wml13.dtd ">
<wml>
<card id="card1" title="ACCEUIL">
<onevent type="ontimer">
<go href="service.php"/>
</onevent>
<timer value="50"/>
<p align="center">
<br/><i><big>net-medecine</big></i>
</p>
</card>
</wml>
```

Les balises de mise en forme:

L'ensemble de balises de mise en forme offert par WML est le même que celui offert par le HTML. Le tableau suivant résume quelques balises de mise en forme d'un texte :

La balise	Fonction
<i>.... </i>	Texte en italique
.... 	Texte en gras
<u>.... </u>	Texte souligné
<big>.... </big>	Texte plus grand
<Small>.... </Small>	Texte plus petit
 	Pour sauter la ligne
<p>.... </p>	Pour définir un paragraphe

Tab: Balises de mise en forme.

Exemple :

```
< ?xml version="1.0" ?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
" http://www.wapforum.org/DTD/wml13.dtd ">
<wml>
<card id="card1" tittle="baliseP">
<p align="left">
<i>à gauche</i>
</p>
<p align="right">
<big><u>à droite</u><big>
</p>
</card>
</wml>
```

La balise d'insertion d'images ;

Le WML utilise le format graphique WBMP (Wireless Bitmap) pour afficher les images sur les terminaux mobiles, ce format est pour l'instant limité à deux couleurs : le noir et le blanc.

Exemple :

```
<?xml version="1.0" ?>
<!DOCTYPE wml PUBLIC "
-//WAPFORUM//DTD WML 1.3//EN"
" http://www.wapforum.org/DTD/wml13.dtd ">
<wml>
<card id="card1" tittle="Image">
<p align="center">

</p>
<p align="center">
<big><i>terminal mobile</i></big>
</p>
</card>
</wml>
```

Les balises de taches:

Elles définissent les actions à exécuter en réponse à des évènements. Lorsqu'une tache est exécutée, elle effectue une sorte de branchement et modifie le contenu de la pile historique d'un périphérique.

Il existe quatre types de taches WML :

<noop> : cet élément est utilisé pour supprimer ou désactiver les définitions d'événement dans un deck.

<prev> : cet élément est utilisé pour naviguer à la carte précédente dans la pile historique du navigateur WAP.

<refresh> : Définit une action qui permet d'actualiser les variables du mobile. Cette balise doit être utilisées dans les balises <do>, <onevent>, et <anchor>.

<go> : Initie une navigation vers une URL. Celle-ci peut pointer vers un nouveau deck WML résidant sur le serveur ou, à l'aide d'un fragment, vers une autre carte du deck actuel.

Les balises de variables :

Le WML permet de définir des variables dans des decks WML, de leurs attribuer des valeurs, de les afficher à l'écran et de les utiliser dans des expressions et de les envoyer dans l'entête http vers le serveur.

<setvar> : Permet de déclarer des variables et d'en fixer le contenu dans le contexte du navigateur. On peut utiliser l'élément **<setvar>** dans les éléments **<go>**, **<prev>**, **<refresh>** pour initialiser les variables avant l'exécution d'une tâche.

<postfield> : Permet de définir des paires de champs *nom/valeur* transmises à la passerelle WAP puis au serveur http qui reçoit la requête **<go>**.

On peut utiliser les éléments **<input>**, **<select>** pour déclarer et attribuer des valeurs aux variables, et pour définir leurs valeurs par défaut.

Les balises d'évènements :

Les tâches ne s'exécutent pas dans n'importe quel contexte : elles doivent s'exécuter en association avec un événement.

■ Les ancres :

<anchor> : Les programmes WML peuvent comporter des liens vers d'autres éléments du programme. Un lien ou ancre est signalé de manière spéciale sur l'écran, entre crochet dans le navigateur de phone.com par exemple.

Le texte de l'ancre est obligatoirement présent.

■ Evènements intrinsèques :

<onevent> : Cette balise permet d'associer un événement avec une action. Lorsque l'événement est déclenché, la tâche est exécutée. La valeur de l'attribut *type* de cette balise indique le nom de l'événement à considérer ; il est obligatoire.

■ Horloges (timer) :

<timer> : Initialise un événement *timer*. A l'expiration d'un certain délai, une tâche est exécutée. L'exécution de la tâche est prise en charge par un gestionnaire d'événements comme *onevent*. L'élément *timer* initialise le temporisateur en dixième de seconde.

■ Événement déclenché par l'utilisateur :

<do> : Définir une zone qui peut être constituée d'une ou de plusieurs tâches différentes ainsi qu'une interface utilisateur. Les tâches sont exécutées dès que l'utilisateur déclenche l'interface.

■ Événement au niveau du deck :

<template> : Permet de définir pour toutes les cartes du deck des gestionnaires d'événement. La définition d'un événement dans une template est équivalente à sa définition dans chaque carte.

Les balises de saisies de données :

▪ **Saisie par l'utilisateur :**

<input> : Spécifie un champ de saisie. Le résultat est affecté ensuite à une variable et peut être transmis à d'autres cartes, par exemple par <go>.

▪ **Listes de choix :**

<select> : Permet de définir une liste d'option à choisir par l'utilisateur. Chaque élément option contient une valeur affectée à la variable définit dans <select>.

<option> : Définit une possibilité de sélection à l'intérieur d'un élément select.

▪ **Choix à plusieurs niveaux :**

<optgroup> : Définit des listes hiérarchiques de choix. Cela complète la balise *option* en regroupant des choix dans un groupe. Le navigateur peut l'afficher sous forme de vue hiérarchique.

Glossaire

PHP

Personal Home Page est un outil permettant de développer des applications Web dynamique. C'est un langage de script côté serveur mélangeant au sein d'un même fichier du code HTML ou même WML et des instructions propres au langage PHP.

ASP

Active Server Page est un langage développé par Microsoft et dédié au développement des applications dynamiques pour internet.

ODBC

Open Data Base Connectivity. Standard élaboré par Microsoft est un ensemble de compagnies pour l'interfaçage à des bases de données.

BDD

Abréviation pour base de données.

Email

Courrier électronique échangé grâce aux réseaux

Hypertexte

Document contenant des liens qui relient certaines parties de phrases à d'autres ressources (texte, son ou image) et qui permettent, lorsqu'on les active de visualiser ces informations supplémentaires.

Internet

Réseau reliant des milliers de réseaux dans le monde, il donne à chaque ordinateur une adresse unique et lui permet d'envoyer et de recevoir des paquets d'informations sans savoir comment ils sont acheminés.

Téléconsultation médicale

C'est la consultation médicale à distance via des réseaux de télécommunication.

ADO

Active Data Object, mécanisme de connexion à tous types de données dont le plus connu est une base de données relationnelle.

IIS

Internet Information Server. Serveur Web sur NT.

Mime

Multipurpose Internet Mail Extensions, qui assure l'échange de données entre deux ordinateurs lorsque des fichiers qui ne sont pas du simple texte sont transmis à travers Internet.

Cache

Espace mémoire de taille limitée permettant de stocker des données. Dans les navigateurs Internet (Web ou WAP), le cache permet de stocker des pages localement pour éviter d'effectuer des accès sur Internet coûteux en temps de réponse.

Cookies

Informations stockées sur le navigateur et pouvant être accédées depuis un serveur Web.

CPU

Unité de calcul d'un ordinateur (micro-processeur)

Format binaire

Le format binaire désigne communément le format numérique dans le quel l'information est encodée.

Get/post :

Méthode de transfert des informations entre un client et un serveur sur le protocole http.

GSM

Global System for Mobile communication, principale norme de téléphonie mobile.

GPRS

Une extension de la norme GSM qui permettra de multiplier par un facteur dix le débit de transfert des informations et surtout qui permettra une tarification des services au volume d'information et non au temps de connexion.

Intranet

Réseau d'entreprise, utilisant les technologies d'internet.

Java

Le langage de programmation développé par Sun.

Passerelle

Brique technologique permettant l'interconnexion de couches réseaux de normes différentes.

Pull

Procède par le quel l'internaute va chercher l'information qu'il désire consulter.

Push

Procède permettant d'amener l'information à l'utilisateur sans que celui n'en fasse une demande explicite.

SMS

Short Messaging System, petit message que l'on peut envoyer sur un téléphone portable à partir d'un autre mobile ou d'un site web.

WAP

Wireless Application Protocol, défini par Motorola, Nokia, Ericsson et phone.com, le protocole WAP est une adaptation des protocoles Internet aux caractéristiques techniques de téléphone mobile.

Gateway

Une passerelle qui assure la liaison entre le réseau d'internet et le réseau de téléphonie mobile.

WML

Wireless Markup Language, langage de description de document dédié à la téléphonie mobile.

Wbmp

Il s'agit d'une extension d'image spécifique aux téléphones portables.

XML

Extensible Markup Language, Langage de description de document orienté données.

Bibliographie

Bibliographie :

- [DES 99] : WAP : Wireless Application Protocol /Alain Deseine.-1999.
- [HOR 99] :A.Horch et T.Balbach : Telemedical information system -1999
- [MEU 96] : K.meunier : Le réseau de télémédecine en radiologie au CHU de Nancy- 1996
- [BOU 02] : Net Economic/ Jérôme Bouteriller.-23 octobre2002.
- [HOM 98] : Interface entre WEB-et base de données/Alex Homer, Darren Gill, Stephen Jakab.- Edition Eyrolles : 1998.
- [LET 00] : Construction d'une application WAP/Laurent Letourmy, Thomas Papiernik, Alain Hélaïli, Xavier Martzel.- Edition Eyrolles : 2000.
- [MAG 96] : Serveur Web/Jonathan Magid, R.douglas Matthews, Paul Jones.- Edition International Thomson Publishing.-1996.
- [PIC 00] : Le WAP : livre blanc/Olivier Picard, Jean-Michel Allemand.- Alladin technologies : 2000.
- [WAP 01] : Le Guide officiel WAP 1.2/ WAPForum ; trad. Valéry Frémaux, Joëlle Cornavin, .../Eyrolles ; fev.2001

Les sites WEB:

- [www1]: comment ça marche: <http://www.commentcamarche.fr>
- [www2] : Guill.net- Les protocoles de communication.
[http:// www.guill.net](http://www.guill.net)
- [www3] : PHP Every Where
<http://PHP.weblogs.com>
- [www4] : Nokia mobile phone, Nokia WAP Toolkit 3.0 user's guide, juin 2000.
<http://www.forumnokia.com>
- [www4] : [WAP@WML](http://www.wml.com) / Van Lancker Luck .- 2001
<http://www.ccim.be/ccim328/wap>
- [www5]:Débuter en PHP: <http://www.php.net>
- [www6]: tous sur le WAP: <http://www.wapforum.com>

