

MA-004-208-1

UNIVERSITE SAAD DAHLAB DE BLIDA
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE



Mémoire en vue d'obtention du diplôme de Master
Spécialité : ingénierie du logiciel

**Sujet : Un système de recommandation sensible au contexte
pour un accès adapté aux SIW à travers des dispositifs mobiles**

Présentée par :
BELKHIR Fatiha
SAHKI Laitmas

Promotrice :
M^{me} REZOUG Nachida.

President de jury : M^{me} Zalina
Examinateur 1 : M^{me} Chesfa
Examinateur 2 : M^{me} Mezi

2013/2014

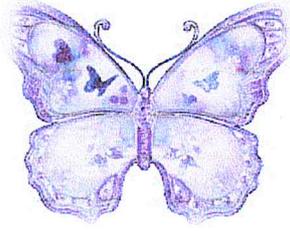
MA-004-208-1

Remerciement

Nous remercions Allah le tout puissant de nous avoir donné la force, le courage, la volonté, la patience grâce à ses bienfaits on a pu réaliser ce projet.

Nous tenons à exprimer nos profonde gratitude et nos respectueux remerciements à M^{eme} Rezoug Nachida notre promotrice, qui nous a patiemment guidées dans l'élaboration de ce travail et pour les informations très utiles qu'elle nous a mises à notre disposition, nous la remercions de nous avoir guidées, orientées et aidées durant notre recherche.

Nos vifs remerciements à toutes les personnes qui ont bien voulu, nous aider de près ou de loin dans notre projet de fin d'études.



Dédicaces

*Je dédie ce modeste travail à la femme qui m'a mise au monde.
Grâce à elle et à ses encouragements je suis arrivée à ce stade.*

À ma chère mère, la femme la plus adorable de ma vie.

À mes chères sœurs : Sabrina, Amel et Marwa grâce à leur soutien, je les souhaite tout le bonheur et la prospérité dans toute la vie.

À ce qui m'a aidé et soutenu dans tous mes études l'ami qui a durement travaillé pour que je puisse réaliser l'un de mes rêves mon meilleur ami Hamza Mazari.

À mon binôme Sahki Laitmas et à toute sa famille .

À toutes les bibliothécaires surtout Djouher et ses sœurs.

À tous mes amis filles et garçons.

*À tous ceux qui me connaissent et à tous les étudiants de la section
2^{ème} année master informatique.*

Dédicaces

C'est avec une immense fierté que je dédie ce modeste travail à ceux qui ont fait de moi ce que je suis aujourd'hui, grâce à leurs sacrifices, leur tendresse et leur amour : mes très chères parents que Dieu les garde.

A mon très cher mari Zouhir Ait Kaci qui m'a soutenu durant mon parcours .

A ma petite famille : Amina , Wassila , Omar

pour leur soutien morale et leurs encouragements, un grand merci à eux

A mes cousines : Dyhia, Assia , Wissem.

A mon binôme belkhir Nassima et à toute sa famille .

A toutes les personnes qui nous ont Prodiguées des encouragements et se sont données la peine de nous soutenir durant cette formation.

A nos chers enseignants sans aucune exception SAHRI Laitmas

Resumé

De nos jours, l'interaction avec les applications informatiques profitent de plus en plus de la technologie mobiles, Les systèmes d'informations traitent de grande quantité de données, par conséquent ils délivrent un nombre important de résultat, de ce fait l'utilisateur se trouve incapable de distinguer l'information la plus pertinente, un système d'information basé sur le web sont des systèmes (SI) traditionnels, ces systèmes permettent de collecter et stocker les informations via le web.

Les systèmes de recommandation sont des outils qui permettent de surmonter le problème de surcharge d'informations en offrant aux utilisateurs des contenus pertinents, on prend en charge leurs profils contextuels.

Dans ce papier, nous proposons un système de recommandation sensible au contexte basé à la fois sur le profil d'utilisateur et le contexte d'utilisation, ce système permettra à l'utilisateur nomade d'accéder à un système d'information web en fournissant une information pertinente selon ses informations contextuelles.

Mots-clés : sensibilité au contexte, Système d'information basé sur le web, adaptation, système de recommandation sensible au contexte, ontologie.

Abstract

Nowaday, interaction with computer applications increasingly benefit of mobile technology, the information systems handle a large amount of data , therefore they deliver a large number of results, thus the user are unable to distinguish the most relevant information

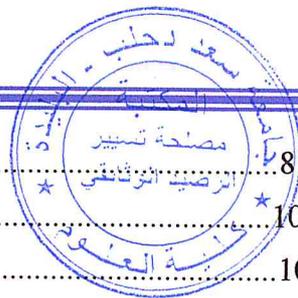
Web Information Systems are traditional (SI) systems, these systems allow us to collect and store information via the web.

Recommender systems are tools that overcome the problem of information overload, by providing users with relevant content, take care of their contextual profiles.

In this paper, we propose a system named a context-aware recommender system based on both user profile and context, this system will allow mobile users to access to a web information system by providing relevant information based on these contextual information.

Keywords: context awarness , Web Information Systems, adaptation , context-aware recommender system, ontology.

TABLE DE MATIERE



1. Introduction Générale	8
2. Problématique	10
3. Objectif	10
4. Organisation du mémoire.....	11

Chapitre 1 : Définition et modélisation du contexte

1. Introduction.....	13
2. Mobilité et informatique mobile.....	14
2.1. Définition de la mobilité.....	14
2.2. L'informatique mobile.....	14
2.2.1. Nomadisme.....	14
2.2.2. L'informatique ubiquitaire	14
3. Définition de la notion du contexte.....	15
3.1. Définition du contexte	15
3.2. Contexte et pertinence.....	15
3.3. Sensibilité au contexte.....	15
3.4. L'importance du contexte dans le domaine informatique.....	16
3.5. les informations de contexte.....	16
3.5.1. Caractéristiques des informations du contexte.....	17
3.5.2. Acquisition du contexte	17
3.5.3. Modélisation des informations du contexte	19
3.5.4. Exemple qui modélise le contexte en utilisant l'anthologie.....	20
4. Conclusion	21

Chapitre 2 : Système d'information basé sur le web

1. Introduction.....	23
2. Généralité sur le système d'information basé sur le web.....	24
2.1. Définition d'un système d'information basé sur le web.....	24
2.2. Système d'information web sensible au contexte	24
3. L'adaptation à un environnement mobile et nomade	25
3.1. Définition de l'adaptation	25
3.2. Classification de l'adaptation.....	25
3.3. Exemple de système qui adapte l'information.....	26
3.3.1. Système hypermédia adaptatif	26
3.3.2. Système multi-agents (SMA)	26
3.4. Représentation de l'utilisateur	27
3.4.1. Représentation du profil de l'utilisateur.....	27
3.4.2. Représentation du contexte de l'utilisateur	27
3.4.3. Représentation des préférences de l'utilisateur	28
3.5. Gestion des préférences de l'utilisateur	29

TABLE DE MATIERE

3.5.1. Les préférences d'activité.....	29
3.5.2. Les préférences de résultat.....	29
3.5.3. Les préférences d'affichage	30
4. Discussions des travaux SIW ubiquitaire (sensible au contexte).....	31
5. Conclusion.....	33

Chapitre 3 : Système de recommandation sensible au contexte

1. Introduction.....	35
2. Définition du système de recommandation sensible au contexte	36
3. Les étapes de la recommandation	36
3.1. Collecte d'information.....	36
3.1.1. Collecte des données explicite (Filtrage actif).....	36
3.1.2. Collecte des données implicite (Filtrage passif).....	36
3.2. Modèle d'utilisateur (Bâtirai une matrice)	37
3.3. Liste de recommandation.....	37
4. Méthode de système de recommandation	37
4.1. Filtrage collaboratif	37
4.2. Filtrage basé sur le contenu.....	37
4.3. Filtrage basé sur la connaissance	37
4.4. Filtrage Hybride.....	37
5. Contexte dans le système de recommandation.....	39
6. Système de recommandation sensible au contexte (SRSC)	39
6.1. Définition des systèmes de recommandation sensible au contexte.....	39
6.2 Paradigme pour l'utilisateur de l'information contextuelle.....	40
7. Conclusion.....	42

Chapitre 4 : Approche proposé pour un accès SIW mobile

1. Introduction.....	44
2. Architecture d'accès au SIW mobile.....	45
2.1. Description générale de l'architecture	46
2.2. les étapes de recommandation (étude de cas pour un utilisateur de profession « enseignement » et « architecture »).....	46
2.3. Les différentes phases d'architecture pour un accès adapté d'un SIW mobile.....	47
1. Collecte de l'information.	47
2. Modélisation de profil contextuel.....	48
2.1 Modélisation de profil contextuel en utilisant l'ontologie.....	48
2.2 Modélisation de profil contextuel en utilisant UML.....	49

TABLE DE MATIERE

2.3 Langage de représentation OWL.....	54
3. Traitement des informations	54
4. La recommandation.....	56
5. Conclusion	60

Chapitre 5 : Implémentation et Réalisation

1. Introduction	62
2. Présentation des outils du développement	63
2.1. Les langages de programmation.....	63
2.2. MySql.....	63
3. Les interfaces de l'application.....	64
4. Conclusion.....	71
 Conclusion générale	 72
 Bibliographic	 73
Annexe.....	80

Liste des figures

Figure 1.1. Architecture d'accès au SIW mobile	45
Figure 1.2. Ontologie de profil contextuel.....	48
Figure 1.3. Diagramme de classes générales du profil contextuel.....	49
Figure 1.4. Diagramme de classes spécifique du profil contextuel.....	50
Figure 2.1. Interface principale	64
Figure 2.2. Interface d'authentification de l'utilisateur	65
Figure 2.3. Interface de création de compte	66
Figure 2.5 L'interface de localisation	67
Figure 2.4: L'interface de recommandation fonctionnalité	68
Figure 2.6 Interface de recherche de fonctionnalité.....	69
Figure 2.7 L'interface de recommandation de plan.....	70

Liste des tableaux

Tab 1.1. Caractéristiques des approches de modélisation du contexte.....	20
Tab 2.1. Comparatif des systèmes d'informations web restituant des informations.....	32
Tab 3.1. Avantages et inconvénient des approches de système de recommandation.....	38
Tab 3.2. Facteurs ajoutés aux définitions de la sensibilité au contexte.....	39
Tab 4.1. Description de l'architecture d'accès au SIW mobile	46

1. Introduction générale

Actuellement, l'informatique est devenue un élément fondamental, un moyen rapide pour exécuter une application quelle qu'elle soit. Après l'installation durable des ordinateurs à domicile, l'informatique devient de plus en plus présente au quotidien. L'explosion de la téléphonie mobile et l'arrivée sur le marché des ordinateurs portables, d'assistants numériques et de toutes sortes de plates-formes clientes portables constituent des indicateurs forts des utilisations futures des systèmes informatiques impliquant plus de mobilité et de variété de situations d'usage.

L'informatique mobile issue de l'évolution des systèmes distribués se présente comme un cadre de définition de ces nouveaux usages [1]. L'informatique mobile se distingue donc de l'informatique classique fixe par les deux aspects tels que la mobilité des usages ainsi que leurs ordinateurs et les contraintes de ressource mobile tels que la largeur de bande sans fil limitée et l'autonomie limitée des batteries [2], la deuxième façon de considérer l'informatique mobile est de l'inscrire dans une approche plus globale que constitue l'informatique ubiquitaire[3].

L'informatique ubiquitaire est une vision récente, dans laquelle un nombre croissant d'appareils (capteurs, processeurs, actionneurs) inclus dans divers objets physiques participent à un réseau d'information global. La mobilité et la reconfiguration dynamique seront des traits dominants de ces systèmes, imposant une adaptation permanente des applications. Les principes d'architecture applicables aux systèmes d'informatique ubiquitaire restent encore largement à élaborer. L'informatique ubiquitaire a donc pour objectif d'intégrer les technologies informatiques au quotidien pour faciliter à l'homme l'utilisation de l'outil informatique dans de nombreux domaines tels que l'information, la formation, le transport [4].

La différence essentielle entre l'informatique actuelle et l'informatique ubiquitaire est le besoin de mobilité, par conséquent la prise en compte de l'environnement. En effet, une particularité de l'informatique ubiquitaire est le besoin de communication d'un objet avec son environnement, ceci dans le but de pouvoir coopérer avec les objets qui l'entourent et ainsi accéder aisément à l'information. L'utilisateur sera alors pris en compte avec son contexte physique afin qu'il puisse avoir un accès mobile à des données et traitements ce qui permettra de lui offrir les meilleures conditions de service. La mise en œuvre de ces services s'appuiera sur des réseaux sans-fil.

A travers les réseaux sans fil, les utilisateurs peuvent accéder au système d'information basé sur web (SIW) via à leurs dispositifs mobiles (DM) tels que les téléphones portables, les PDA pour obtenir l'information souhaitée par rapport à leurs caractéristiques et à celles de leur DM. Les concepteurs de SIW doivent compter sur des mécanismes capables d'offrir aux utilisateurs nomades une information adaptée au contexte d'utilisation qui se repose sur une représentation de divers éléments tels que les activités menées par l'utilisateur, ses préférences, les caractéristiques du dispositif utilisé, la localisation ou encore le temps [5].

Plusieurs systèmes permettent d'adapter l'information pour que l'utilisateur puisse avoir une information correspondante à son contexte d'utilisation, parmi ces systèmes on retrouve ; les systèmes hypermédias adaptatif qui prend en considération non seulement les caractéristiques de l'utilisateur mais aussi ses préférences, et aussi les systèmes multi-agents.

Les systèmes de recommandation permettent de nous fournir des informations susceptibles d'intéresser l'utilisateur, cependant, une nouvelle famille de ce système appelée Systèmes de Recommandation Sensibles au Contexte (SRSC) se basant sur les caractéristiques de l'utilisateur "profil" et "contexte d'utilisation"[6].

2. Problématique:

Avec les avancées des technologies de l'environnement mobile et l'apparition de divers types de dispositifs, les utilisateurs d'internet demandent à accéder et échanger de l'information là où ils sont, à tout moment et à travers n'importe quel type de dispositifs. Ces nouvelles exigences font que l'information demandée et recherchée devrait non seulement répondre au besoin de l'utilisateur mais aussi devrait être supportée sur n'importe quel type de dispositif. En effet, l'utilisateur mobile (qui change tout le temps de localisation ou de dispositif) peut obtenir une masse d'informations qui sont pour la plupart inutiles et si elles le sont, elles pourraient être non supportées sur son dispositif.

La problématique ainsi traitée par la suivante thématique est l'accès à un SIW à travers les dispositifs mobiles.

3. Objectifs

L'objectif de ce présent sujet est de permettre à utilisateur mobile d'accéder à un SIW en prenant en charge

- la mobilité de l'utilisateur nomade.
- l'émergence des DM dans l'accès web.
- l'adaptation de l'information recherchée aux ressources réduites des DM (taille mémoire, résolution de l'écran, et taille du disque dur) et aussi aux caractéristiques de l'utilisateur mobile.

Sur ce, un système recommandé sensible au contexte est proposé pour un accès adapté au SIW en fournissant une information pertinente basée sur une recherche intelligente de l'information et un affichage approprié (en se basant sur les caractéristiques de l'utilisateur "profil" et ceux de son dispositif mobile "contexte")

Le chapitre 1 est composé de deux parties; dans la première partie nous présentons la définition de la mobilité , l'informatique mobile et ses aspects tels que le nomadisme et l'informatique ubiquitaire, dans la deuxième partie, nous commençons par la définition concernant la notion de contexte, la pertinence de contexte, la sensibilité au contexte et l'importance de son utilisation dans le domaine informatique ,puis nous décrivons, les informations de contexte ,ensuite nous exposons les méthodes utilisées pour acquérir les informations contextuelles et nous décrivons les différentes approches de modélisation du contexte Enfin, nous abordons un exemple de modélisation de contexte.

Le chapitre 2 est consacré aux systèmes d'information basé sur le web , nous présentons une généralité sur ces systèmes ensuite, nous nous intéressons à l'adaptation au environnement mobile et nomade, nous présentons d'abord la définitions de l'adaptation , classification de l'adaptation puis nous mentionons deux systèmes qui permettent d'adaptation de l'information ,enfin nous présentons un travail lié à la représentation de l'utilisateur et sur la gestion de ces préférences

Le chapitre 3 est consacré aux systèmes de recommandation sensible au contexte , nous présentons la définition de ces systèmes , les différents étapes de recommandation , nous exposons les méthodes de recommandation , ensuite nous abordons une synthèse sur ses méthodes en montrant ses avantages et ses inconvénients, nous décrivons le contexte dans ses systèmes ,puis nous définissons le système de recommandation sensible au contexte, enfin nous concentrons sur les paradigme de SRSC pour l'utilisation de l'information contextuelles.

Dans le chapitre 4 nous proposons une architecture d'accès aux SIW mobile, nous commençons à faire une description détaillée sur l'architecture proposée puis nous définissons ses étapes ,enfin nous détaillons les différentes phases de cette architecture.

Chapitre 5 chapitre est dédié à une implémentation et réalisation d'une application d'un système de recommandation sensible au contexte.

Chapitre 1

Définition et modélisation de contexte

1. Introduction

La mobilité est profondément inscrite dans la nature humaine, le concept de l'informatique mobile issue de l'évolution des systèmes distribués se présente comme un cadre de définition de ces nouveaux usages des systèmes d'information [1], parmi ces aspects on distingue le nomadisme et l'ubiquité.

Le nomadisme des usagers traduit le fait qu'il peut se connecter à des points d'accès différents en ayant recours éventuellement à des liaisons sans fil tandis que l'ubiquité des usagers traduit le fait qu'ils peuvent vouloir rester connectés lorsqu'ils sont en mouvement [2]. Cette évolution apparaît par le comportement d'utilisateurs. Accompagnés dans leurs déplacements par un ensemble de dispositifs mobiles, téléphones, ordinateurs, assistants personnels, etc. Ces utilisateurs veulent accéder à leurs informations personnelles et à différents services d'information quels que soient l'endroit, le moment et les conditions d'utilisations.

Pour comprendre les causes de comportement des utilisateurs mobiles ; la notion de contexte nous à donner un sens opérationnel pour le développement d'applications ubiquitaires et mobiles. Pour prendre en compte, la sensibilité au contexte est essentielle d'avoir un modèle de contexte pour permettre la reconnaissance des informations provenant des sources variées et hétérogènes.

2. Mobilité et informatique mobile

La disponibilité du service internet devient de plus en plus importante, ce qui permet aux utilisateurs d'accéder aux applications web via leurs périphériques portables dans un environnement mobile.

2.1 Définition de la mobilité

La mobilité caractérise ce qui peut se mouvoir ou être mû, ce qui peut changer de place ou de position, elle peut être définie comme la capacité d'accéder à partir de n'importe quel endroit [3].

2.2 L'informatique mobile

Le concept de l'informatique mobile réfère à la possibilité pour des usagers munis de périphériques portables ou l'ordinateur mobile d'accéder à des services et des applications évolués à travers une infrastructure partagée de réseau indépendamment de leur localisation physique ou de leur comportement de mouvement. L'informatique mobile se distingue donc de l'informatique classique fixée par deux aspects [2]:

- La mobilité des usages et de leurs ordinateurs.
- Les contraintes de ressource mobile telles que la largeur de bande sans fil limitée et l'autonomie limitée des batteries.

Cette notion intuitive de l'activité humaine se traduit cependant par deux aspects différents :

2.2.1 Nomadisme

La première façon de considérer l'informatique mobile consiste à fournir un dispositif portable aux utilisateurs. L'utilisation de ces dispositifs reste néanmoins immersive, c'est-à-dire que l'usage de ces dispositifs requière toute l'attention de l'utilisateur et s'applique à tout déplacement, indépendamment de celui-ci. Cette approche est souvent appelée « nomadisme », bien que ce terme puisse prendre d'autres significations dans d'autres domaines. L'un des exemples les plus connus de nomadisme sont les ordinateurs portables [3].

2.2.2 Informatique Ubiquitaire

La deuxième façon de considérer l'informatique mobile est de l'inscrire dans une approche plus globale que constitue l'informatique ubiquitaire. Cette dernière, appelée également informatique pervasive, a pour but de rendre accessible toutes sortes de services, n'importe quand, n'importe où, à travers des DM [3].

3. Définition de la notion de contexte

Le terme de contexte s'exprime en plusieurs définitions, son importance dans le domaine informatique évolue avec l'évolution des technologies mobiles.

3.1. Définition de Contexte

La notion de contexte est définie dans plusieurs travaux, parmi ces travaux nous citons la définition la plus explicative donnée par Dey [7] présente le contexte comme « Toute information qui peut être employée pour caractériser la situation d'une entité. Une entité est une personne, un endroit, ou un objet qui est considéré pertinent à l'interaction entre un utilisateur et une application y compris l'utilisateur et les applications eux-mêmes », est souvent reprise car elle permet de faire un équilibre entre les deux tendances. Elle est l'une des définitions du contexte la plus expressive et suivie en littérature. Cette définition présente un intérêt particulier pour nos travaux car elle mentionne clairement l'utilisation de l'information sur une entité pour décrire sa situation [8].

3.2. Contexte et Pertinence

La notion de pertinence du contexte permet de déterminer les informations du contexte qui sont utiles et qui sont impliquées dans la conception d'un système sensible au contexte [8]. Dey [7] mentionne ce besoin dans sa définition concerne particulièrement la conception des systèmes sensibles au contexte, puisqu'elle tient compte de la pertinence des éléments pour l'interaction entre l'utilisateur et l'application. Cependant, l'impossibilité d'énumérer tous les éléments d'information pertinents pour toutes les situations est évidente car ces éléments changent inévitablement d'une situation à l'autre.

3.3. Sensibilité au contexte

La sensibilité au contexte est une approche prometteuse permettant de supporter les approches ubiquitaires et orientées utilisateurs. Pour Schmidt et al [9], la sensibilité au contexte est la sensibilité à la situation dans laquelle se trouve un utilisateur. Selon Dey [7], un système sensible au contexte est « un système qui utilise le contexte pour fournir des informations et/ou des services pertinents à un utilisateur. La pertinence dépend de la tâche de l'utilisateur », alors que Henricksen [10], considère qu'une application est sensible au contexte si elle s'adapte aux changements de l'environnement et aux besoins de l'utilisateur.

3.4. Importance du contexte dans le domaine informatique

Le contexte a été introduit dans différentes spécialités du domaine informatique comme le traitement du langage où le contexte a un rôle très important dans l'interprétation correcte du langage humain qu'il s'agisse du texte ou de la parole. Les systèmes de prise de décisions et l'informatique sensible au contexte. Par analogie au mode de fonctionnement du

raisonnement humain, l'utilisation du contexte dans ces domaines consiste à ajouter une adaptabilité et une aide à la prise de décision à différents types d'applications. Parmi l'un des travaux de recherche qui représente la notion de contexte dans l'informatique sensible au contexte est citée comme suite [11]:

Contexte dans l'informatique sensible au contexte

L'informatique sensible au contexte regroupe les applications ou les systèmes qui utilisent le contexte pour adapter leurs comportements. En se basant sur les travaux de Dey et Abowd [12], ceux de Schilit et al. [13], et ceux de Brown et al. [14], Chalmers a identifié six utilisations possibles du contexte dans un environnement sensible au contexte [15] :

- Présentation et affichage des informations de contexte comme par exemple la localisation.
- Association d'informations de contexte à une donnée de l'application comme par exemple l'association des personnes présentes à une conférence et leur localisation au programme de la conférence.
- La configuration sensible au contexte, comme par exemple le fait d'effectuer une impression sur l'imprimante la plus proche, sans intervention de l'utilisateur.
- Lancement d'actions de réaction selon la valeur du contexte, comme le chargement d'une carte géographique d'une zone quand l'utilisateur se déplace vers cette zone.
- Médiation contextuelle, qui consiste par exemple à utiliser le contexte pour modifier un service fourni.
- Présentation sensible au contexte qui consiste à modifier l'interface d'une application selon le contexte.

3.5 Les informations de contexte

Les informations de contexte sont des informations collectées à partir de plusieurs sources. Les informations collectées peuvent être incohérentes, obsolètes, incorrectes ou incomplètes. La complexité des informations de contexte nécessite des moyens de modélisation pour simplifier leur utilisation [11]. Ces informations sont de trois types (contexte d'interaction, contexte physique, contexte d'utilisation) [16] :

- a. Les contextes d'interaction avec le média décrivant en premier lieu le dispositif utilisé pour accéder au système (laptop, téléphone mobile, tablette) et par conséquent définir la technologie réseau utilisée (câblé, WIFI), les ressources du dispositif utilisé ainsi que le type de support de données (textes, musique, images, vidéos ou simplement des requêtes) .

- b. Les contextes physiques définissent les caractéristiques physiques de l'environnement de l'utilisateur comme par exemple niveau du bruit et la température.
- c. Le contexte d'utilisation nous informe sur les circonstances de l'utilisation courante comme par exemple le profil de l'utilisateur (porte essentiellement sur son activité courante) et la localisation.

3.5.1. Caractéristiques des informations de contexte

On définit les caractéristiques techniques de l'information de contexte comme suite [17]:

a. **Le contexte est changeable avec le temps** : l'information de contexte change continuellement de valeur avec le temps. Par exemple, le déplacement de l'utilisateur implique que le contexte localisation de l'utilisateur change de valeur.

b. **L'information de contexte est hétérogène** : l'hétérogénéité provient du fait que le contexte est capturé à partir d'une variété de sources [18]. Particulièrement, ce contexte peut être capturé (*Sensed*) directement de capteurs physiques ou récupérés à partir de composants logiciels, donné par l'utilisateur (*Profiled*) ou bien dérivé (*Derived*) en synthétisant plusieurs sources ou interprété d'une seule source pour obtenir un niveau levé de l'abstraction de cette information.

c. **L'information de contexte est imparfaite** : cette caractéristique de contexte est déterminée selon la source de cette information, le contexte est ambigu si des ressources séparées fournissent une même information avec des niveaux de différents granularités, le contexte peut être imprécis ou bien le contexte peut être erroné ou même inconnu.

d. **L'information de contexte est interdépendante** : l'information de contexte peut être dépendante d'une autre information de contexte. Le changement de sa valeur peut l'influencer sur une autre valeur de contexte [19].

3.5.2. Acquisition du contexte

Selon la synthèse faite par Soukkarieh Bouchra, « l'acquisition du contexte est un processus principal lors de la conception d'un système sensible au contexte. Ce processus vise à permettre au système de capturer les informations contextuelles nécessaires à son fonctionnement ». Pour ce faire, deux manières peuvent être utilisées [20] :

a. **Acquisition explicite** : les informations contextuelles sont données par l'utilisateur avec son intervention. Cette acquisition lui permet d'exprimer en lui-même les informations contextuelles dont le système a besoin pour lui restituer la réponse la plus pertinente. Par exemple, quand l'utilisateur lance une requête, il peut exprimer ses caractéristiques statiques et ses préférences mais aussi sa localisation, etc. de manière explicite via une interface dédiée.

b. Acquisition implicite : les informations contextuelles sont déterminées sans l'intervention de l'utilisateur par des matériels et/ou des logiciels spécifiques donc les paramètres du contexte sont récupérés implicitement par des capteurs. Par exemple le type de dispositif utilisé via un programme java, la localisation de l'utilisateur via un capteur physique de géo-localisation GPS (Global Positioning System).

3.5.3. Modélisation des informations du contexte

Selon la synthèse faite par **Belhanafi benhalouli nabiha** [11], pour faciliter l'utilisation du contexte, il est indisponible de définir un modèle pour le décrire aux environnements mobile. En effet, il existe quatre approches qui sont définies comme suite [11]:

a. Approches paires/triplets

Les approches paires/triplets sont caractérisées par une pauvreté d'expressivité et la simplicité des données qu'elles représentent. Elles sont basées sur une description par un tuple (clé/valeur) ou par un triplet (contexte, valeur, degré de certitude). De ce fait, elles ne permettent pas de décrire les observables, ni les relations entre les informations de contexte, ni les règles d'interprétation.

b. Approches orientées modèle

Les approches orientées modèle sont des approches prometteuses car elles utilisent non seulement un modèle formel pour décrire le contexte mais aussi elles offrent un méta-modèle de description qui peut être réutilisé par plusieurs applications. Les approches orientées modèle existantes ne prennent pas en compte la description des règles d'interprétation de contextes de haut niveau, ni la modélisation des capteurs. Cependant, ces approches offrent une possibilité d'extension pour permettre la description de nouveaux types de relations entre les informations de contexte ainsi la possibilité d'intégrer les notions qui manquent.

c. Approches basées sur la logique

Les approches basées sur la logique sont des approches formelles dont l'utilisation a été longtemps restreinte au domaine de l'intelligence artificielle. Ces approches permettent de raisonner sur les informations de contexte pour déduire de nouvelles valeurs du contexte ou pour lancer des réactions au niveau de l'application ou du système. Les approches appartenant à cette catégorie permettent de décrire des relations entre les informations de contexte, mais leurs applications aux systèmes existants restent limitées .

d. Approches orientées ontologie

Une ontologie est une description sémantique, structurée et formelle des concepts d'un domaine et de leurs inter-relations [16]. Plusieurs modèles d'ontologies ont été proposés pour décrire le contexte. Ces approches permettent non seulement de modéliser le contexte, mais aussi de raisonner sur les données décrites. Les approches orientées ontologies sont caractérisées par une possibilité d'extension et le partage des données

En effet, les caractéristiques de partage et de distribution des données ont été exploitées afin de définir des méta-modèles de description du contexte.

De plus, les moteurs d'inférence fournis par les ontologies ont été utilisés pour déduire des contextes de haut niveau à partir des données collectées. De nombreux langages informatiques sont apparus pour construire et manipuler des ontologies, nous distinguant trois type d'ontologie de description du contexte: COBRA-ONT (*Context Broker Architecture ontology*), CONON (*CONtextONtology*), et CoOL (*ContextOntology Language*).

Dans le but de mettre au point un langage standardisé, le W3C a créé en novembre 2001 un groupe de travail qui a abouti à la recommandation OWL (*Ontology Web Language*) [23]. OWL est un langage basé sur RDF [11], OWL définit une syntaxe pour décrire et construire des vocabulaires afin de créer des ontologies.

Le résumé de ces approche de modélisation tel qu'il a été fait par **Belhanafi benhalouli nabiha** [11] est decris dans le tableau suivant :

Approche de modélisation	Caractéristique	Description de relations entre informations de contexte	Raisonnement sur le contexte	Réutilisation et extension du modèle	Publication de données décrites
Paire/ Triplet	Simplicité d'utilisation, pauvreté d'expression	NON	NON	NON	NON
Basée sur la logique	Très formelle	NON	OUI	NON	NON
Orientée modèle	Utilisation d'un metamodelle	OUI	NON	OUI	NON
Orientée ontologie	Utilisation d'un moteur d'inférence	OUI	OUI	OUI	OUI

Tab 1.1. Caractéristique des approches de modélisation du contexte [11].

3.5.4 Exemple qui modélise le contexte en utilisant l'ontologie

Pour définir un modèle de contexte basé sur l'ontologie, nous citons ci-après un exemple inspiré des rapports de thèse d'après **Bouneffouf Djallel** [6], Le contexte est modélisé par un triplet $C = (O_{\text{Localisation}}, O_{\text{Temps}}, O_{\text{Social}})$ où chaque $O_i, i \in \{\text{Localisation}, \text{Temps}, \text{Social}\}$ est une ontologie qui gère le type de données d'une dimension du contexte, il se concentre sur trois dimensions citées comme suite [6]:

a. Localisation

La localisation est une position géographique à base de coordonnées qui peut être défini par une adresse ou par une information plus sémantique permettant de caractériser le type de lieu. Il existe différentes manières de caractériser la localisation retournée par le GPS.

b. Temps

Abstraction de temps par des périodes utiles dans divers domaine d'application (jour de travail, vacances, pause déjeuner) permet de définir l'aspect temporel caractérisant la situation de l'utilisateur.

c. Relations sociales

Les relations sociales d'un utilisateur se réfèrent à la typologie des interlocuteurs potentiels de l'utilisateur, comme par exemple ses collègues, un client ou son manager.

4. Conclusion

Dans ce chapitre, nous avons mis l'accent sur les principales définitions, une première section est consacrée sur la notion de la mobilité suivie par la notion de l'informatique mobile portée par ces deux aspects tels que le nomadisme et l'informatique ubiquitaire.

Dans la deuxième section, nous avons exposé la notion du contexte et la définition de la sensibilité au contexte suivi par l'importance du contexte dans le domaine informatique. Ensuite, nous avons décrit les informations de contexte ainsi ses caractéristiques et les différentes méthodes d'acquisition de ces informations. Par la suite, nous avons présenté les différentes modélisations des informations du contexte (paires/triplets ,orientées modèles, des modèles basées sur la logique , des modèles basées sur l'ontologie).

Chapitre 2

Systeme d'Information basé sur le Web

1. Introduction

Les systèmes d'information basée sur le web (SIW) sont des systèmes d'information traditionnelle qui font parties des développements technologiques, ces systèmes permettent de délivrer à l'utilisateur des informations et des services en utilisant des technologies du web.

L'hors d'un accès au SIW à travers un DM (dispositif mobile), les concepteurs de SIW doivent tenir en compte le type de DM (par exemple, la taille de l'écran, la mémoire, le disque dur) pour afficher l'information la plus pertinente de l'utilisateur selon son DM approprié

Dans un tel système, l'adaptation de l'information devient un facteur majeur permettant de fournir à l'utilisateur des informations et des services adaptés, l'adaptation de l'information à l'utilisateur nomade prendre en compte les préférences de l'utilisateur ainsi son contexte d'utilisation (la localisation de l'utilisateur, les caractéristiques du *DM*).

3. L'adaptation à un environnement mobile et nomade

Les utilisateurs nomades souhaitent obtenir l'information la plus pertinente par rapport leurs caractéristiques et à celles de leur(s) dispositif(s) d'accès via le web, un nouveau besoin en adaptation est apparu. Ce besoin est la capacité pour l'adaptation de prendre en compte le profil de l'utilisateur ainsi son contexte d'utilisation

3.1 Définition de l'adaptation

L'adaptation de l'information vise donc à délivrer à un utilisateur une information en adéquation avec son profil contextuel. Cette adaptation change la fonctionnalité, l'interface, la teneur en information ou l'aspect d'un système pour augmenter sa pertinence personnelle. La sélection des préférences qui peut être appliquée lors de la session en cours d'un utilisateur repose sur le contexte d'utilisation [26].

3.2 Classification de l'adaptation

Selon Soukkarieh Bouchra [20], pour Jameson [31] et Simonin et al.[32] ont classifié l'adaptation en fonction du but visé en deux types : l'adaptation à l'utilisateur et l'adaptation au contexte.

a. L'adaptation à l'utilisateur

Un système puisse réaliser un processus d'adaptation à l'utilisateur, il se base sur un profil de cet utilisateur (les informations sur les buts, les caractéristiques, les préférences et les connaissances). A partir de ce profil, le système peut préciser les modifications nécessaires pour réaliser l'adaptation .Qui peut s'effectuer en deux manières différentes Beggas [33]. Si l'utilisateur peut modifier certaines caractéristiques du système, afin de l'ajuster à ses préférences personnelles et à son comportement Opperman et al.[34] ; le système est dit adaptable. Si le système est capable d'adapter son comportement aux besoins, capacités et préférences de l'utilisateur courant ; le système est dit adaptatif Brusilovsky[35].

b. L'adaptation au contexte :

L'adaptation au contexte, appelée l'adaptabilité : est un changement dans le système pour prendre en compte un changement dans l'environnement du système qui concernent le contexte de l'exécution (besoins des utilisateurs, contraintes matérielles, contraintes du réseau, etc.) [36].

Cette adaptation englobe toutes les stratégies et les techniques mises en œuvre par le système et les applications afin de répondre aux variations du contexte. Ces variations peuvent se

2. Généralité sur le système d'information basé sur le web

L'émergence du Web rend les techniques et les méthodes utilisées dans les Systèmes d'Information "classiques" pour modéliser les données inefficaces. Les technologies du Web, qui peut être considéré comme un système hypermédia ¹ public fonctionnant sur Internet, permet de consulter, avec un navigateur, des pages mises en ligne dans des sites. L'intégration entre, d'une part, ses technologies visant initialement à servir de l'information et d'autre part, les services des Systèmes d'Information traditionnels a conduit à l'apparition de la notion de Système d'Information Web [20].

2.1 Définition d'un système d'information basé sur le web

Un Système d'Information basé sur le Web est un système d'information (SI) traditionnel qui utilise des paradigmes et des technologies du Web pour offrir des informations et des services aux utilisateurs ou aux autres Systèmes d'Information / applications [25].

SIW permet de collecter, structurer, stocker, gérer et diffuser de l'information de la même manière que les Systèmes d'Information (SI) traditionnels le font mais via le Web. Un SIW offre aux utilisateurs des fonctionnalités complexes qui sont activées à travers un navigateur Web en utilisant une interface hypermédia [26], ce système a un accès universel ou contrôlé à un espace d'informations sur lequel divers traitements sont activables, le plus souvent par le biais de requêtes. Dans de nombreux domaines, les SIW sont exploités par différents types d'application dont le nom est préfixé par « e » comme le commerce et la vente e-business, l'enseignement à distance e-Learning, etc. Par exemple pour le e-commerce, sa modélisation et ses services à mettre en œuvre dans le cas d'un SIW sont bien différents que ceux d'un e-Learning. Ces systèmes permettent de s'adapter à l'utilisateur, pour leur confort et leur durabilité [27].

2.2 Système d'information web sensible au contexte

Systèmes d'Information Web sensibles au contexte sont des systèmes guidés principalement par le contexte afin de fournir à l'utilisateur des informations et des services pertinents [20].

1. L'hypermédia correspond à un réseau de nœuds véhiculant des informations de différents types, comme du texte, des images, des données audio ou vidéo, éventuellement combiné.

produire aux différents niveaux tels que la bande passante réseau, les réseaux disponibles, la localisation géographique ou les ressources disponibles sur le terminal [37].

3.3 Exemple de système qui adapte l'information

Parmi les systèmes qui ont permis l'adaptation de l'information sont les systèmes hypermédias adaptatifs et les systèmes multi agents.

3.3.1. Système Hypermédia Adaptatif

Brusilovsky [38] définit un Système Hypermédia Adaptatif comme « tout système hypertexte et hypermédia reflétant des caractéristiques de l'utilisateur dans son propre modèle d'utilisateur, modèle utile pour adapter plusieurs aspects du système à l'utilisateur ». Selon **Brusilovsky** [39], les systèmes hypermédia adaptatifs construisent un modèle de buts, de préférences et de connaissances de chaque utilisateur individuel et utilisent ce modèle à travers l'interaction avec l'utilisateur afin d'adapter l'information à ses besoins.

3.3.2 Système Multi-Agents (SMA)

Carabelea et al. [40] définissent un SMA comme « une fédération d'agents logiciels qui interagissent dans un environnement partagé coopérant et coordonnant leurs actions compte tenu de leurs buts et de leurs plans».

Un système multi agents est un système distribué composé d'un ensemble d'agents. les SMA sont conçus et implantés idéalement comme un ensemble d'agents interagissant, le plus souvent, selon des modes de coopération, de concurrence ou de coexistence , Un agent est une entité caractérisée par le fait qu'elle est, au moins partiellement, autonome, ça peut être un processus, un robot, un être humain ... [41].

3.4 Représentation de l'utilisateur

Afin d'adaptation de l'information, nous définissons le profil de l'utilisateur et le contexte d'utilisation ainsi les préférences de l'utilisateur.

3.4.1 Représentation du profil de l'utilisateur

Selon **Bouzeghoub et al.** [42], le profil de l'utilisateur peut être vu comme un modèle personnalisé d'accès à l'information qui régit la manière de présenter les résultats du système. Le profil de l'utilisateur est composé [43] :

- **Des caractéristiques statiques** qui ajoutent à la définition du contexte des informations spécifiques à l'utilisateur telles que son nom, son prénom, son adresse électronique, sa fonction, etc.
- **Des caractéristiques évolutives** qui ajoutent au contexte de l'utilisateur des informations concernant : son environnement : localisation de l'utilisateur, date de la requête, Ses préférences (déclarées par l'utilisateur), temps, etc.

Tamine et al. [44] et **Zemirli et al. [45]** présentent trois approches différentes de représentation du profil de l'utilisateur :

(i) Approche ensembliste : le profil est généralement formalisé sous forme de vecteurs de termes pondérés ou de classes de vecteurs. Le contenu est constitué d'un ou de plusieurs vecteurs définis dans un espace de termes. Ces termes sont obtenus à partir de plusieurs sources d'information concernant l'utilisateur. Les coordonnées des vecteurs correspondent aux poids associés aux termes retenus dans le profil. L'utilisation de plusieurs vecteurs correspond à deux préoccupations : prendre en compte des centres d'intérêt multiples et gérer leur évolution dans le temps.

(ii) Approche sémantique : la représentation du profil met en évidence, dans ce cas, les relations entre les contenus de l'information. Cette représentation se base sur l'utilisation d'ontologies ou de réseaux sémantiques probabilistes. Chaque catégorie de la hiérarchie représente la connaissance d'un domaine d'intérêt de l'utilisateur.

(iii) Approche multidimensionnelle : le profil est conçu comme un ensemble de dimensions représenté selon divers formalismes, cherchant à modéliser/représenter l'utilisateur. Parmi les dimensions, on trouve les données personnelles, les centres d'intérêt, la qualité attendue des résultats délivrés, les préférences, etc.

3.4.2 Représentation du contexte d'utilisation

Le contexte d'utilisation se repose sur une représentation de divers éléments tels que les activités menées par l'utilisateur, ses préférences en termes de contenus et de présentation, les caractéristiques du dispositif utilisé, la localisation ou encore le temps.

3.4.3 Représentation des préférences de l'utilisateur

D'après **Carrillo Ramos Angela Cristina [26]**, **Hafenrichter et al. [46]** et **Freuder et al. [45]**, une préférence personnelle exprime les souhaits de l'utilisateur pendant ses interactions avec le système, **Freuder et al. [47]** modélisent les préférences comme des contraintes légères où chaque combinaison de valeurs pour les variables est fournie par l'utilisateur et il indique l'importance de cette contrainte dans le système. **Hafenrichter et al. [48]** modélisent

les préférences comme des ordres partiels stricts de la manière suivante (en se basant sur le modèle de préférences de Kießling [49]) :

Soit $A = \{A_1, A_2, \dots, A_k\}$, l'ensemble d'attributs A_j avec des domaines $\text{dom}(A_j)$ où :

$\text{dom}(A) = \text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_k)$

Une préférence P est un ordre partiel strict $P = (A, <_P)$, où $<_P \subseteq \text{dom}(A) \times \text{dom}(A)$

« $x <_P y$ » est interprété comme « je préfère y à x ».

3.5 Gestion des préférences de l'utilisateur

Plusieurs travaux ont adopté une approche multidimensionnelle pour représenter le profil de l'utilisateur. Nous citons principalement l'approche de Carrillo Ramos Angela Cristina [26]. Nous distinguons trois types de préférences [26] :

3.5.1. Les Préférences d'Activité

Une activité est un ensemble de fonctionnalités c'est tout simplement l'appel d'un service qui doivent être exécutés. Une fonctionnalité peut être liée à d'autres fonctionnalités qui s'exécutent de manière séquentielle, concurrente ou conditionnelle par rapport à celle-ci.

Les Préférences d'Activité décrivent la manière dont l'utilisateur envisage d'accomplir ses activités dans le système. Nous définissons ce type de préférence de la manière suivante :

Préférence_Activité(type, {critère}, A)

« *type* » prend pour valeur : *générale* ou *spécifique*.

« {critère} » est une liste de critères d'adaptation (par exemple, la localisation et le type de DM) dont le système tient compte pour l'exécution des fonctionnalités dans une session.

« *A* » est l'**Activité** que l'utilisateur souhaite accomplir dans le système (par exemple, une consultation).

Cette *activité* est exprimée à travers une **chaîne de fonctionnalités** exécutées de manière séquentielle, concurrente ou conditionnelle. Dans le **premier cas**, nous considérons qu'un utilisateur souhaite exécuter une *activité* *A* composée d'une ou plusieurs fonctionnalités.

Lord d'un utilisateur est au niveau de département :

Par exemple dans la gestion soutenance, afin d'évaluer le travail des étudiants, un enseignant peut faire la tâche « rendez-vous ».

Utilisateur-enseignant peut exécuter plusieurs fonctionnalités par exemple : « faire RDV » cette fonctionnalité permet de fixer des rendez-vous, et « consulter RDV » pour avoir des informations sur les rendez-vous fixés (nom étudiant, cycle, temps, date).

F1=FaireRDV(<nom_étudiant>, <cycle>, <spécialité>, <date>, <temps>)

F2=ConsulterRDV(<nom_étudiant >)

Un enseignant peut aussi faire la tache « emplois de temps ».

L'utilisateur-enseignant peut exécuter plusieurs fonctionnalités tel que « fixe cours », cette fonctionnalité permet de fixe la date des cours, et « consulte cours » pour avoir des informations sur les cours fixé.

F1=FixeCours (<nom_module>,<date>,<salle>,<temps>)

F2=ConsulterCours(<nom_module>)

Préférence_Activité (Générale, {}, F1|F2;F3)

Le *deuxième cas* consiste à exécuter une ou plusieurs fonctionnalités en tenant compte des critères (tels que sa localisation, son DM, etc.) définis par l'utilisateur. Ces résultats peuvent être différents selon ces critères.

Lors d'un utilisateur au niveau de l'université,

L'utilisateur –chercheur consulte les conférences il obtient comme résultat la liste des conférences. Aussi il peut consulter les réunions et il obtient comme résultat la liste de toutes les réunions. Lors d'un utilisateur au niveau de département L'utilisateur –enseignant consulte les cours il obtient comme résultat la liste des cours a enseigné La préférence et les fonctionnalités sont définies de la manière suivante :

F1= Consulter_conférence (<nom_chercheur, date>, <conférence>)

F2= Consulter_reunion (<nom_chercheur, date>, <réunion>)

F3= Consulter_cours (<nom_enseignant, date>, <cours>)

Préférence_Activité (Générale, {localisation}, {if (localisation == "université") then F1p else (if (localisation == "département") then F2p)})

3.5.2 Les Préférences de Résultat

Les Préférences de Résultat définissent les résultats attendus par un utilisateur à l'exécution d'une ou plusieurs fonctionnalités. Un utilisateur peut établir ses formats préférés (par exemple, « texte », « vidéo » ou « image ») pour afficher chaque résultat. Ce type de préférence est représenté à travers un tuple comme suit :

Préférence_Résultat (type, F, <(s1,fp1),(s2,fp2)...(sk,fpk)>)

« F » est le nom de la fonctionnalité.

Le dernier terme du tuple est une liste de couples :

(si, fpi) où si représente un résultat de F et fpi le format souhaité associé.

Un utilisateur peut choisir les résultats de la fonctionnalité (parmi les résultats propres de la fonctionnalité), c'est-à-dire, l'utilisateur peut souhaiter obtenir seulement certains résultats (par exemple s3, s5, s8). La propriété fp peut prendre la valeur de nil, ce qui signifie que l'utilisateur n'a pas défini le format préféré pour ce résultat.

Afin d'illustrer la notion de préférence de Résultat, considérons que lorsqu'un enseignant exécute la fonctionnalité « Visualisé conférence », il souhaite que la conférence sera affiché (sous forme de vidéo) et le rapport de conférence (définies par T3 format texte) :

**Préférence_Résultat(Spécifique, «visualisé _conférence»
<(conférence,vidéo),(T3,texte)>**

3.5.3 Préférences d’Affichage

Les Préférences d’Affichage décrivent la manière dont l'utilisateur souhaite que son DM affiche l'information (par exemple, l'utilisateur désire seulement l'information au format image). Nous les classons par rapport aux formats préférés exprimés par l'utilisateur lors des préférences de Résultats et à l'ordre d'affichage de l'information. Ces préférences sont définies par des tuples comme suit :

Préférence_Affichage(type, caractéristiques)

« Type » peut prendre pour valeur : « ordre », « forma », Par exemple, les préférences qui concernent le format sont représentées par un tuple comme suite :

PA_Format(format, {caractéristiques}, préférence_ associée).

Dans ces tuples, le format peut prendre pour valeur : « vidéo », « texte », « image », « son».

Py est une préférence associée à Pz. Cela signifie que, dans le cas où Py ne peut pas être satisfaite, alors Pz est considérée. S'il n'y a pas de préférence associée, cette propriété prend la valeur de nil dans le tuple. Chaque format repose sur un ensemble de caractéristiques.

4. Discussion des travaux des SIW ubiquitaire (sensible au contexte)

Dans la littérature beaucoup de SIW sensibles au contexte ont été proposés, selon **Soukkarieh Bouchra** [20] une synthèse des travaux est détaillé dans les paragraphes suivants :

Le travail présenté par **Virgilio** [28] a permis de modéliser SIW sensible au contexte à travers une architecture répendant à un ensemble d'exigence d'adaptation , En effet l'adaptation est assurée en prenant compte le profil et le contexte de l'utilisation , Le profil ainsi définit permet de représenter d'une manière abstraite la configuration des pages web qui composent la réponse de l'utilisateur , alors que le contexte est sollicité pour représenter ces pages web en prenant en considération les caractéristique du dispositif , les préférences utilisateurs , et de sa localisation , etc.

En se qui concerne **Kirsch-Pinheiro** [29] elle se focalise sur les utilisateurs nomades et formalise le contexte avec le modèle objet, elle prend aussi en considération les aspects physiques (la localisation ou le dispositif) , également les aspects liés au processus coopératif dans lequel l'utilisateur est impliqué (groupe, rôle, activités, etc.), sans oublier la présentation de l'information dans le but d'adapter un processus de filtrage qui utilise ce modèle objet basé sur le contenu . En effet, ce modèle peut aussi être utilisé pour l'adaptation des informations fournies par le système, spécialement à travers un contexte précis.

La troisième analyse traite un travail de **Hinz et al.** [30] qui présentent une nouvelle architecture nommée **AMACONT**. Cette dernière vise à la génération dynamique des présentations du Web selon les préférences de l'utilisateur et les capacités de son dispositif. Cette technique aide à fournir à l'utilisateur des interfaces intelligentes adaptatives, en prenant en considération les capacités hétérogènes des dispositifs. Ce travail se base spécifiquement sur le contexte et spécialement sur les caractéristiques du dispositif car ils prennent en charge l'adaptation afin d'optimiser la performance de l'architecture et de réduire la charge du serveur lorsque nombreux utilisateurs sont connectés.

Le résumé des travaux tel qu'il a été fait par Soukkarieh Bouchra [20] est d'écrit dans le tableau suivant, une partie de ces travaux ont été détaillé dans les paragraphes ci-dessus

	Elément du contexte prise en compte				Type d'adaptation	Mise en œuvre de l'adaptation
	Par rapport à l'utilisateur	Par rapport au dispositif	Par rapport aux réseaux	Autre		
Selon vigilio [28]	Préférences de l'utilisateur et localisation	Caractéristiques du dispositif			Contenu	Approche d'adaptation (chaque profil associé à une configuration)
Selon Kirsch - Pinheiro [29]	Localisation	Caractéristiques du dispositif		Processus coopératif (groupe, rôle, etc.)	Contenu	Processus de filtrage
Selon Hinz et al [30]	Préférences de l'utilisateur	Caractéristiques du dispositif			Contenu	Règles d'adaptation

Tab 2.1. Comparatif des Systèmes d'Information Web restituant des informations [20].

Un travail plus récent qui adapte réellement l'accès de l'information web par rapport au profile et au contexte de l'utilisateur est proposé par Carillo Ramos Angela Cristina [26] :

Ce travail de recherche a été réalisé dans le but d'offrir aux utilisateurs nomades un accès a l'information en fonction de leur dispositif et d'adapter l'information au contexte d'utilisation ainsi au profile de l'utilisateur. Pour atteindre ce but une solution à été réalisé « un Framework appelé PUMAS », L'approche qui été choisie est celle des agents. En effet l'architecture de PUMAS est composée de quatre Systèmes Multi-Agents (SMA), par la suite dans ce travail un Système de Gestion de Profil Contextuel (SGPC) a été élaboré qui contribue à l'adaptation de l'information délivrée à un utilisateur nomade sur trois aspects : i) une formalisation de la notion de préférence de l'utilisateur, ii) un algorithme de correspondance contextuelle, iii) un mécanisme qui gère les conflits pouvant survenir entre les préférences de l'utilisateur. A la fin de ce travail le SGPC a été intégré à PUMAS au sein du SMA dédié à l'adaptation de l'information.

5. Conclusion

Ce chapitre a été consacré aux systèmes d'informations basé sur le web .Tout d'abord, nous avons présenté une définition sur les SIW, on a parlé de ces fonctionnalités telles que l'accès, la recherche, l'affichage et le stockage d'information.

Ensuite, nous nous sommes intéressées à l'adaptation de l'information, cette adaptation a été classifié en deux types ; l'une consiste à l'adaptation à l'utilisateur et l'autre l'adaptation au contexte, parmi les systèmes qui adaptent l'information sont les systèmes hypermédias adaptatifs et les systèmes multi-agents.

En ce qui concerne la représentation des préférences de l'utilisateur, plusieurs travaux ont été proposés, nous nous sommes intéressées aux travaux de Carrillo pour présenter les différents types de préférences (activité, Résultat et affichage).

En terme de représentation du contexte d'utilisation et de la sensibilité du contexte de ce système ,on a fait une légère discussions des travaux de **Soukkarieh Bouchra [20]** , **Virgilio [28]**, **Kirsch-Pinheiro [29]**, **Hinz et al [30]** qui intègrent des caractéristiques des environnements nomades ,pour finir avec une comparatif des systèmes d'information sensible au contexte restituant des informations présentées par **Carillo Ramos Angela Cristina [26]**.

Afin d'adapter l'information à l'utilisateur, les systèmes hypermédias adaptatifs exploitent les préférences et les caractéristiques de l'utilisateur, ces systèmes ne considèrent pas la localisation de l'utilisateur, ni les caractéristiques de son dispositif d'accès afin d'adapter l'information à son contexte. Pour cette raison, nous nous sommes orientées vers les systèmes de recommandation sensible au contexte pour leurs capacités d'adapter l'information aux deux types d'adaptation.

Chapitre 3

Systeme de recommandation sensible au contexte

1. Introduction

Pour le confort et la facilité d'obtention de contexte adapté à un individu, Un système de recommandation (SR) aide les utilisateurs qui n'ont pas suffisamment d'expérience ou la compétence nécessaire pour évaluer le nombre potentiellement important, d'alternatives offertes par un site (web) [50].Ce système est utilisé dans les domaines comme la navigation web pour suggérer des ressources pertinentes aux utilisateurs au moment adéquat. La plupart des approches existantes ne tient pas en compte les informations contextuelles telles que le temps, la localisation, la date etc.

2. Définition de système de recommandation

C'est un système capable de fournir des recommandations personnalisées, ou permettant de guider l'utilisateur vers des ressources (items²) intéressantes ou utiles au sein d'un espace de données important[51], ce système aide les utilisateurs à faire leurs choix dans un domaine où ils disposent peu d'informations pour trier et évaluer les alternatives possibles **Resnick et Varian, (1997) [52]**.

3. Les Etapes de la recommandation :

Pour recommander des items à un utilisateur, le système doit disposer d'un profil représentatif de ses préférences. Pour le construire il doit suivre ces trois étapes [53] :

3.1. Collecte d'Information

Un système de recommandation doit pouvoir faire des prédictions sur les intérêts des utilisateurs. Il faut donc pouvoir collecter un certain nombre de données sur ceux-ci afin d'être capable de construire un profil pour chaque utilisateur. Une distinction peut être faite entre deux formes de collecte de données:

3.1.1 Collecte de données explicite (Filtrage actif): repose sur le fait que l'utilisateur indique explicitement au système ses intérêts, cette opération a une capacité à reconstruire l'historique d'un individu et la capacité d'éviter d'agréger une information qui ne correspond pas à cet unique utilisateur (plusieurs personnes sur un même poste) par exemple demander à un utilisateur de commenter, taguer/étiqueter, noter, aimé ou encore ajouter comme favoris des contenus qui l'intéressent(objets, articles...).

3.1.2 Collecte de données implicite (Filtrage passif): repose sur une observation et une analyse des comportements de l'utilisateur effectué de façon implicite dans l'application qui embarque le système de recommandation, le tout se fait en "arrière-plan" (en gros sans rien demander à l'utilisateur) par exemple Obtenir la liste des éléments que l'utilisateur a écoutés, regardés ou achetés en ligne.

2. « Item » est le terme général utilisé pour dénoter ce que le système recommande aux utilisateurs. Par exemple il peut être une page web, un livre, un film, de la musique, etc.

3.2 Modèle Utilisateur (bâtir une matrice)

Le modèle utilisateur se présente généralement sous forme de matrice. On peut le représenter comme un tableau qui contient des données recueillies sur l'utilisateur associées aux produits disponibles sur le site web.

Un autre point important est comment le temps influence le profil de l'utilisateur. Les intérêts des utilisateurs, généralement, évoluent au cours du temps. Les données du modèle utilisateurs devraient donc constamment être réajustées pour rester conformes aux nouveaux centres d'intérêts de l'utilisateur.

3.3. Liste de recommandations

Pour extraire une liste de suggestions à partir d'un modèle utilisateur, les algorithmes utilisent la notion de mesure de similarité entre objets ou personnes décrits par le modèle utilisateur. La similarité a pour but de donner une valeur ou un nombre (au sens mathématique du terme) à la ressemblance entre deux choses. Plus la ressemblance est forte, plus la valeur de la similarité sera grande. A l'inverse, plus la ressemblance est faible, plus la valeur de la similarité sera petite.

4. Méthodes de système de recommandation

Pour déterminer les items à recommander, plusieurs méthodes sont possibles

4.1. Filtrage collaboratif effectue des recommandations par analyse à la fois des opinions de l'utilisateur sur les ressources qu'il a consultées ainsi que des autres utilisateurs sur les ressources qu'ils ont consultées [54].

4.2. Filtrage basé sur le contenu effectue des recommandations en comparant le contenu sémantique des ressources avec les goûts exprimés par l'utilisateur [55].

4.3. Filtrage basé de connaissances effectue des recommandations en exploitant les connaissances sur l'utilisateur et des heuristiques préétablies [56].

4.4. Filtrage Hybride : Un système hybride est une combinaison entre le filtrage collaboratif et basé sur le contenu. D'après Adomavicius et Tuzhilin [57] on distingue quatre façons de combiner les deux méthodes précédentes à savoir :

- Implémenter la méthode collaboratif et la méthode basée sur le contenu séparément puis combiner leurs prédictions.
- Incorporer quelques caractéristiques de la méthode basée sur le contenu dans l'approche collaboratif.
- Incorporer quelques caractéristiques de la méthode collaboratif dans l'approche à base de contenu.
- Construction d'un modèle général unifié qui incorpore les caractéristiques des deux modèles.

Pour mieux cerner les avantages et les inconvénients de chacune des approches on a proposé de présenter notre synthèse d'approches sous forme d'un tableau décrit comme suit :

	Avantage	Inconvénient
Filtrage collaboratif	<ul style="list-style-type: none"> -N'a pas besoin d'une large communauté d'utilisateurs pour pouvoir effectuer des recommandations. Une liste de recommandations peut être générée même s'il n'y a qu'un seul utilisateur [58]. -Avoir la possibilité de manipuler des ressources de toute langue, de tout media (texte, audio, vidéo), puisque ne requérant aucune indexation de la ressource [58]. -Découlent de la possibilité de tirer profit des évaluations effectuées par les autres utilisateurs du système [59]. 	<ul style="list-style-type: none"> -Difficulté de trouver des utilisateurs ou groupes d'utilisateurs similaires [58]. -Ne propose une recommandation intéressante que si le chevauchement entre les historiques est important et que l'ensemble des contenus est relativement statique [60].
Filtrage Basé sur le contenu	<ul style="list-style-type: none"> -Pas besoin de données sur les autres utilisateurs [50]. -Possibilité de faire des recommandations à des utilisateurs avec des goûts « uniques » [50]. -Possibilité de recommander de nouveaux items ou même des items qui ne sont pas populaires [50]. -Un utilisateur pourra recevoir des recommandations même s'il est le seul inscrit par conséquent il est absolument indépendant des autres [61]. 	<ul style="list-style-type: none"> - Pas d'historique pour des nouveaux utilisateurs [50]. -Impossible d'exploiter les jugements des autres utilisateurs [50]. -L'utilisateur risque de ne pas découvrir de nouveaux domaines potentiellement intéressants pour lui car l'utilisateur ne reçoit que les recommandations relatives aux thèmes déjà présents dans son profil [61].
Filtrage hybride	<ul style="list-style-type: none"> -Aucune augmentation requise [62]. -Un Feedback de préférence qualificatif détaillé [62]. -Sensible au changement de préférence [62]. -Recommandations personnalisées [62]. 	<ul style="list-style-type: none"> -Engineering des connaissances [62].

Tab 3.1. Avantage et inconvénient de chaque approche.

5. Contexte dans les systèmes de recommandation

Le contexte est un concept à multiples facettes qui a été étudié dans différentes disciplines de recherche, on se concentre sur les domaines qui sont directement liés à des systèmes de recommandation tels que : e-commerce personnalisées, bases de données, la recherche de l'information, le marketing les systèmes d'information, ainsi les systèmes mobile et ubiquitaire sensibles au contexte [63].

Systèmes mobile et ubiquitaire sensibles au contexte

Dans la littérature relative aux systèmes sensibles au contexte, le contexte a été initialement défini comme la localisation de l'utilisateur identité des personnes à proximité de l'utilisateur, les objets autour et l'évolution de ces éléments [64].

La liste des facteurs du contexte définit dans [64] n'est pas exhaustive, un ensemble de facteur a été rajouté par les auteurs de [14] [65] [66] dans le tableau suivant :

Recherche faites par :	Facteurs qui ont été ajoutés à cette définition
Brown & al. [14]	inclure la date, la saison, et la température
Ryan et al. [65]	ajouter les états physiques et conceptuels de l'intérêt pour un utilisateur.
Dey et al. [66]	information sur état émotionnel (interaction entre utilisateur et application)

Tab 3.2. Facteur ajouté à la définition de la sensibilité au contexte

6. Système de recommandation sensible au contexte (SRSC)

La majorité des approches existantes de systèmes de recommandation se focalise sur la recommandation des éléments les plus pertinents pour les utilisateurs individuels et ne prennent pas en considération l'information contextuelle telles que le temps, le lieu et la compagnie d'autres personnes (par exemple, pour regarder des films ou dîner à l'extérieur) .En d'autres termes, les systèmes de recommandation qui se traitent avec des applications ne comportant que deux types d'entités ; les utilisateurs et les item [63].

6.1. Définition des systèmes de recommandation sensible au contexte

Les systèmes de recommandation sensible au contexte doivent être en mesure d'obtenir des informations contextuelles qui correspondent à l'activité de l'utilisateur (par exemple, faire un achat ou d'une qualification article). D'une part, cette information contextuelle est potentiellement nécessaire dans le cadre du processus d'apprentissage et de modélisation

(telles que la découverte de règles, segmentation des utilisateurs ou la construction de modèles de régression) .D'autre part, pour une donnée d'un utilisateur et d'un élément cible. Le système doit être en mesure d'identifier les valeurs des variables contextuelles spécifiques dans le cadre d'une interaction continue de l'utilisateur avec le système [63].

6.2 Paradigmes pour l'utilisation de l'information contextuelle

Selon Adomavicius et Tuzhilin [63] Les systèmes de recommandation traditionnelle sont conçus à partir des connaissances partielles de préférence de l'utilisateur (ensemble d'items qui représente les préférences pour certains utilisateurs) qui sont souvent limités. Les données d'entrée des systèmes de recommandation traditionnels sont généralement basés sur les enregistrements de la forme <utilisateur, item, taux de préférences >.

Alors que les données d'entrée des systèmes de recommandation sensibles au contexte basées sur les préférences de l'utilisateur, sont généralement basé sur les enregistrements de la forme <utilisateur, item, contexte, taux de préférences >, où chaque enregistrement spécifique comporte l'intérêt porté par un utilisateur a un article mais aussi l'information contextuelle dans laquelle l'élément a été consommé par cet utilisateur (par exemple, le contexte = Samedi).

En outre, un système de recommandation sensible au contexte permet d'introduire le contexte dans le processus de recommandation. En particulier, du point de vue algorithmique, les recommandations des approches sensibles au contexte fonctionnent avec des données de la forme $U \times I \times C \times R$, où « C » est une dimension contextuelle supplémentaire, produire une liste de recommandations contextuelles i_1, i_2, i_3, \dots pour chaque utilisateur « u ». Le processus de recommandation dans les systèmes de recommandations sensible au contexte peut prendre l'une des trois formes suivantes:

- (a) **Pré-filtrage Contextuel:** dans ce paradigme, les informations qui concernent le contexte actuel « c » sont utilisées pour la sélection des données pertinentes et les préférences peuvent être prédites en utilisant n'importe quel système classique de recommandation 2D sur les données sélectionnées.
- (b) **Post-filtrage Contextuel:** dans ce paradigme, l'information contextuelle est initialement ignorée et les préférences peuvent être prédites en utilisant n'importe quel système classique de recommandation 2D sur toutes les données. Alors l'ensemble de recommandations résultants est ajusté (contextualité) pour chaque utilisateur en utilisant les informations contextuelles.
- (c) **Modélisation Contextuel:** dans ce paradigme l'information contextuelle est utilisée directement dans les techniques de modélisation comme une partie d'estimation d'un utilisateur pour des items. Pendant que les approches contextuelles « pré-filtrage » et « post-filtrage » peuvent utiliser des fonctions traditionnelles de recommandation 2D, l'approche de modélisation contextuelle donne

lieu à des fonctions de recommandation véritablement multidimensionnels, qui représentent des modèles prédictifs (arbre de décision, régression, modèle probabiliste, ou autre technique) ou les calculs heuristiques qui incorporent des informations contextuelles en plus des données d'utilisateur (user) et d'élément (item).

$$\text{Taux de préférences} = R(\text{Utilisateur}, \text{Items}, \text{Contexte}).$$

Un nombre significatif d'algorithmes de recommandation basé sur une variété de procédés heuristiques ainsi des techniques de modélisation prédictives ont été développées au cours des 10-15 dernières années et certaines de ces techniques peuvent être prolongées de la 2D à la configuration de recommandations multidimensionnelles. Pour cela, nous allons présenter deux exemples d'approches multidimensionnelles pour la modélisation contextuelle [67].

(i) Approche heuristique :

La dimension traditionnelle (2D) de l'approche heuristique de plus proche voisin [68] [69] peut être étendue dans les cas multidimensionnel qui inclure les informations contextuelles d'une manière directe, en utilisant la distance métriques à n-dimensions à la place de « utilisateur -utilisateur » ou « item-item » métriques de similarité traditionnellement.

(ii) Approche Base de modèle

Il ya eu plusieurs techniques de recommandations à base de modèles de systèmes de recommandation en deux dimensions , Certaines de ces méthodes peuvent être directement étendue dans le cas multidimensionnel, telle qu'elle est proposée dans [70] qui montrent que leur technique 2D surpasse certaines méthodes de filtrage collaboratif déjà connus. La méthode proposée par Ansari et al. [70] combinent les informations sur les utilisateurs et les items dans un modèle hiérarchique unique fondé sur la régression des préférences bayésienne qu'utilise Markov Chain Monte Carlo (MCMC), une technique pour estimer les paramètres.

7. Conclusion

Dans ce chapitre, nous avons mis l'accent sur les systèmes de recommandation, ces systèmes permettent de faciliter la recherche et l'accès à l'information en proposant des items à un utilisateur, selon son profil et ses préférences. Pour déterminer les items à recommander nous avons défini les méthodes qui permettent d'effectuer cette recommandation. Ces systèmes disposent de deux types d'entités (utilisateur, items), tandis que le système de recommandation sensible au contexte comporte trois types d'entités (utilisateur, items, contexte).

Chapitre 4

Approche proposée pour un accès SIW mobile

1. Introduction

Pour faciliter le développement des applications sensibles au contexte, nous proposons une architecture d'accès aux systèmes d'information web mobile, l'objectif de cette architecture est l'adaptation de l'information dans les environnements mobiles dans le but est d'établir l'accès aux SIW mobile, pour cela nous nous sommes intéressées vers les systèmes de recommandation sensibles au contexte afin d'aboutir à une donnée pertinente selon le contexte de l'utilisation et le profil de l'utilisateur. L'architecture d'accès aux SIW mobile proposée se base sur trois couches : couche utilisateur, couche sémantique, couche système de recommandation sensible au contexte.

Au niveau de la couche utilisateur, nous récupérons les informations de profil d'utilisateur et de son contexte, ces informations sont collectées afin de les utiliser dans la prochaine étape.

Au niveau de la couche sémantique, nous utilisons une ontologie pour modéliser le profil contextuel aussi un langage OWL DL afin d'écrire l'ontologie de profil contextuel, nous proposons un algorithme de recommandation de profil contextuel qui permet de fournir à l'utilisateur de profession « enseignement » ou « architecture » l'information la plus pertinente soit par son localisation actuelle ou par son activité préférée ou seulement par rapport à son emplacement.

Au niveau de la couche de système de recommandation sensible au contexte, nous choisissons la méthode de modélisation contextuelle, l'une des techniques de cette méthode est l'approche heuristique, cette dernière nous a permis de représenter un modèle prédictif. Afin de recommander à l'utilisateur un SIW approprié par rapport à son activité et des items déjà consultés par lui dans le passé, nous avons utilisé deux méthodes de filtrage « basé sur le contenu » et « collaboratif ».

2. Architecture d'accès au SIW mobile

L'architecture d'accès au SIW mobile est composée de trois principaux couches, la couche utilisateur, la couche sémantique et la couche système de recommandation sensible au contexte voir figure1. 1.

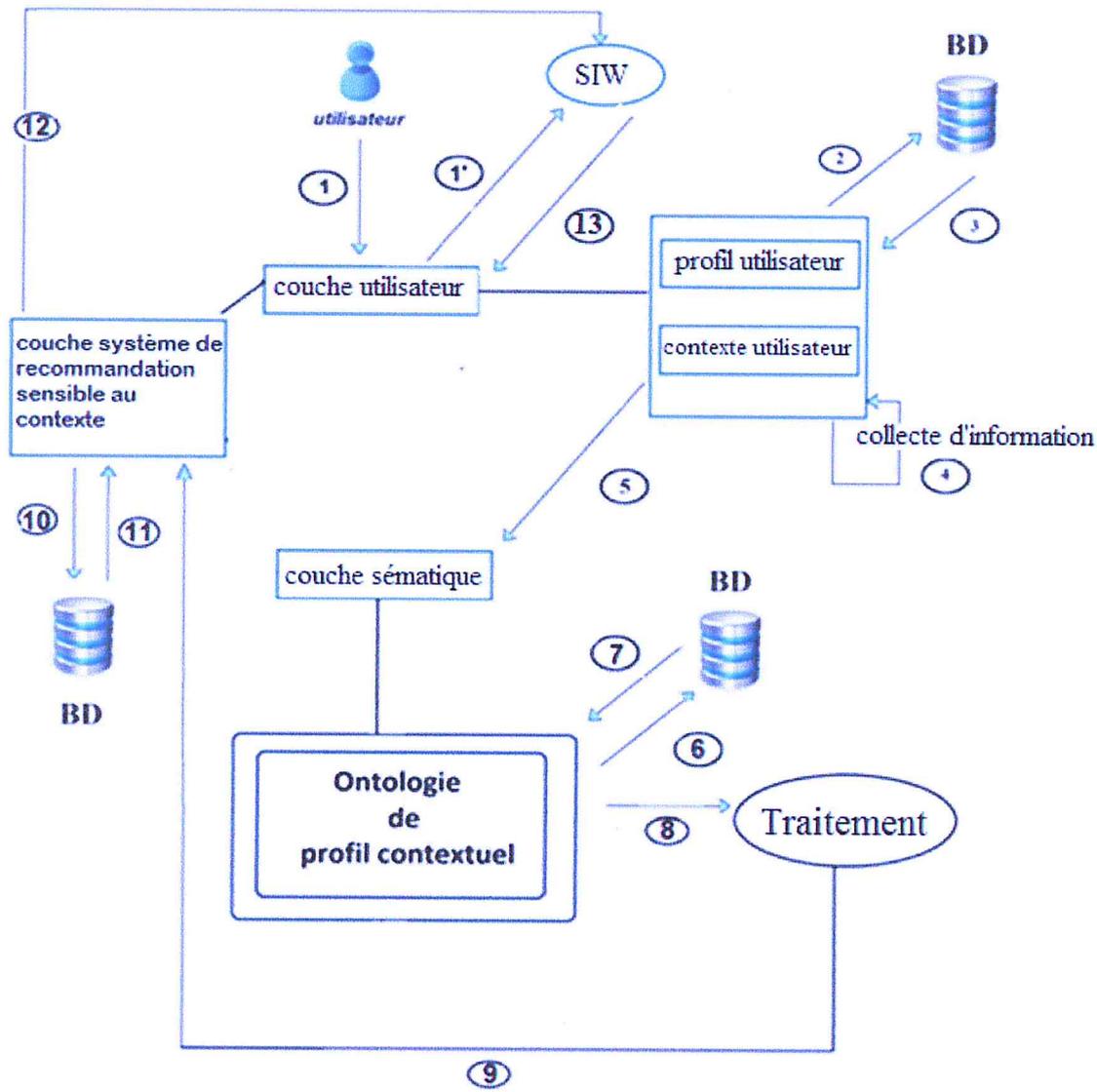


Figure 1.1. Architecture d'accès aux SIW mobile

2.1 Description générale de l'architecture proposée

Numéro	Description
1+1'	Authentification de l'utilisateur afin d'accéder au SIW via son profil.
2	Envoi des données saisies par l'utilisateur à la base de données
3	Etablir l'accès au SIW
4	Collecte des informations
5	Envoi des données collectées
6	Envoi des données apparié à la base de données
7	Envoi des résultats
8	Envoi des données afin de les traités
9	Envoi des résultats traités
10	Envoi les données à recommandé
11	Résultat de la recommandation
12	Envoi des résultats de la recommandation
13	Affichage de résultat final de la donnée pertinente au SIW approprié à un utilisateur

Tab 4.1. Description de l'architecture

2.2. Les étapes de recommandation (étude de cas d'un utilisateur de profession « enseignement » et « architecture »)

Pour mieux comprendre les différentes phases d'architecture, nous présentons les étapes de recommandation destinées à des utilisateurs d'activités enseignants, chercheurs et architectes reflétant étape par étape le déroulement de notre proposition pour un accès adapté aux SIW mobile.

L'utilisateur-enseignant a pour objectif de gérer et consulter son emploi du temps et de fixer les dates de rendez-vous pour les étudiants de fin d'étude afin de consulter leur évaluation de travail, d'autre part l'utilisateur-chercheur peut visualiser les conférences réalisées par l'université et avoir la liste des réunions prévues à travers leur DM, cependant l'utilisateur-architecte peut avoir la liste des plans de travail.

Etape 1 : une interface est utilisée pour récupérer les informations de l'utilisateur afin de lui permettre d'accéder au SIW via son profil aussi pour le mettre à jour, ces informations sont transmises à la base de donnée pour lui permettre l'accès via le SIW approprié.

Lorsque l'utilisateur s'authentifie, il choisi une seule fois l'activité souhaitée, le système va extraire son activité et sa localisation afin d'aboutir les préférences d'activité de l'utilisateur et de les utilisées comme des variables d'entrée pour la prochaine étape.

Etape 2 : nous avons modélisé une ontologie dans laquelle on trouve tous les concepts concernant le profil d'utilisateur et son contexte.

A fin d'établir l'accès via SIW approprié, nous avons utilisé les préférences d'activité pour obtenu l'activité préférée de l'utilisateur pour lui fournir l'information la plus pertinente.

Etape 3 : le système de recommandation sensible au contexte recueille les informations de l'utilisateur pour effectuer des recommandations selon son contexte (sa localisation) et ses préférences d'activités, le résultat obtenu est envoyé à la base de donnée, une fois la donnée est récupérée le résultat est remis au SRSC à ce stade, le résultat final sera affiché au SIW approprié à son activité où l'utilisateur récupère l'information désirée.

2.3. Les Différentes phases d'architecture pour un accès adapté d'un SIW mobile

1. Collecte des informations :

La phase collecte des informations est la phase de la couche utilisateur, cette dernière est constituée d'un ensemble de profils utilisateurs et d'un ensemble de contexte d'utilisation, Le but du profil utilisateur est de stocker toutes données potentiellement utile sur un utilisateur [43]. Le profil de l'utilisateur est composé des caractéristiques statiques et des caractéristiques évolutives (cf. section 2. 3.4), tandis que le seul facteur de contexte d'utilisateur prise en charge dans le cadre de notre thématique est la localisation. Cette phase permet de collecter les informations contextuelles d'un utilisateur. La collecte des informations contextuelles ce fait d'une manière explicite et implicite (cf. section 3.3).

Exemple

Implicite : fournir à un nouveau utilisateur un formulaire web en lui demandant de choisir son activité actuelle.

Explicite : détecté l'emplacement de l'utilisateur à travers une API commerciale (Google maps), nous aide à récupérer l'activité actuelle de l'utilisateur.

Dans notre proposition, la collecte d'information se repose sur le profil d'utilisateur et son contexte, pour détecter les préférences d'activités de l'utilisateur on extrait son activité d'après son profil ainsi sa localisation d'après le contexte de l'utilisateur afin de définir le lieu actuel de l'utilisateur.

2. Modélisation de profil contextuel

La phase de modélisation de profil contextuel est la phase de la couche sémantique, cette dernière à pour objectif de modéliser le contexte d'utilisation ainsi le profil de l'utilisateur. Afin de modéliser le profil contextuel, nous avons choisi une modélisation par ontologie et une modélisation par le langage UML.

2.1 .Modélisation de profil contextuel en utilisant l'ontologie

2.1.1 Définition d'ontologie

D'après Gruber [71], une ontologie est « une spécification explicite d'une conceptualisation ». Dans la définition d'ontologie de Gruber, « une spécification explicite » signifie que les concepts et les relations d'un modèle abstrait reçoivent des noms et des définitions explicites. Une « conceptualisation » concerne un modèle abstrait qui représente la manière dont les personnes conçoivent les choses réelles dans le monde [72].

2.1.2 Ontologie de profil contextuel :

Nous avons choisi l'ontologie comme un modèle de représentation de l'ensemble des concepts de profil contextuel, elle permet de modéliser le contexte et le profil, Nous citons principalement le schéma de représentation de l'ontologie pour la modélisation de profil et de contexte d'utilisation inspiré de travail d'Angela Carrillo-Ramos [24].

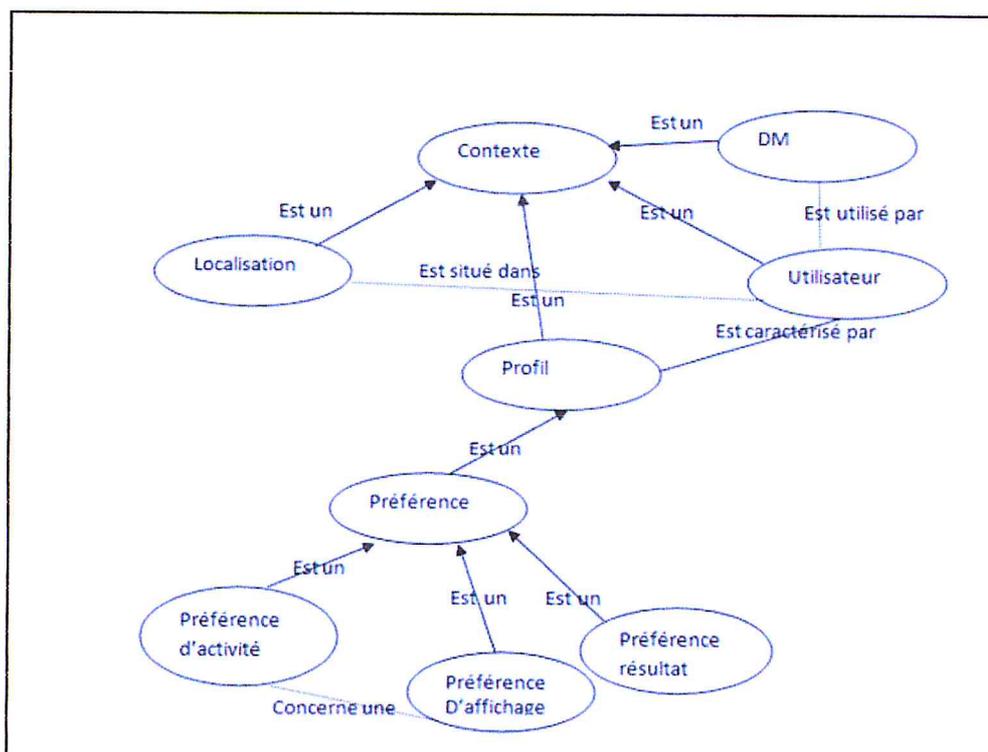


Figure 1.2. Ontologie de profil contextuel.

L'explication des concepts de cette ontologie est inspirée de travail d'Angela Carrillo-Ramos [24], ces classes sont décrites comme suite :

L'ontologie présentée dans la figure 1.2 représente tous les concepts qui sont liées à l'ontologie de « contexte » tels que le type de dispositif d'accès, l'utilisateur, la localisation qui définit l'emplacement de l'utilisateur ainsi le concept de profil qui est lié au concept préférences, ces derniers sont composés d'un ensemble de préférences classées en trois types activités, résultat et affichage (cf. section 2. 3.5). Ces préférences représentent respectivement les activités qu'un utilisateur souhaite accomplir dans le système, les résultats attendus de ces activités, et la manière dont ces résultats sont affichés sur le dispositif d'accès.

2.2. Modélisation de Profil Contextuel en utilisant le langage UML

Dans cette section, nous montrons les diagrammes de classes qui représentent les trois types de préférences (préférences d'activité, de résultat et d'affichage) et le profil contextuel de l'utilisateur représenté comme suite :

(i) Diagramme de classe générale

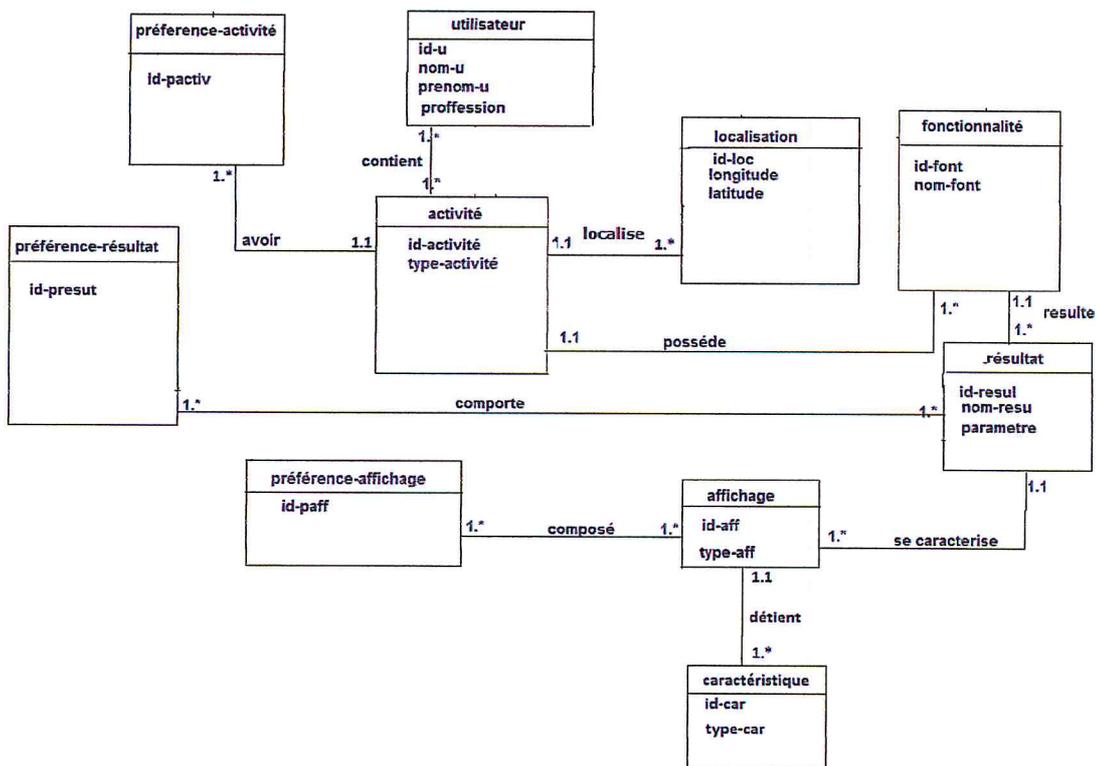


Figure 1.3. Diagramme de classes générales de profil contextuel.

(i) Diagramme de classe spécifique

Ce diagramme de classe représente le profil contextuel destiné aux utilisateurs de profession « enseignement » ou « architecteur » représenté comme suite :

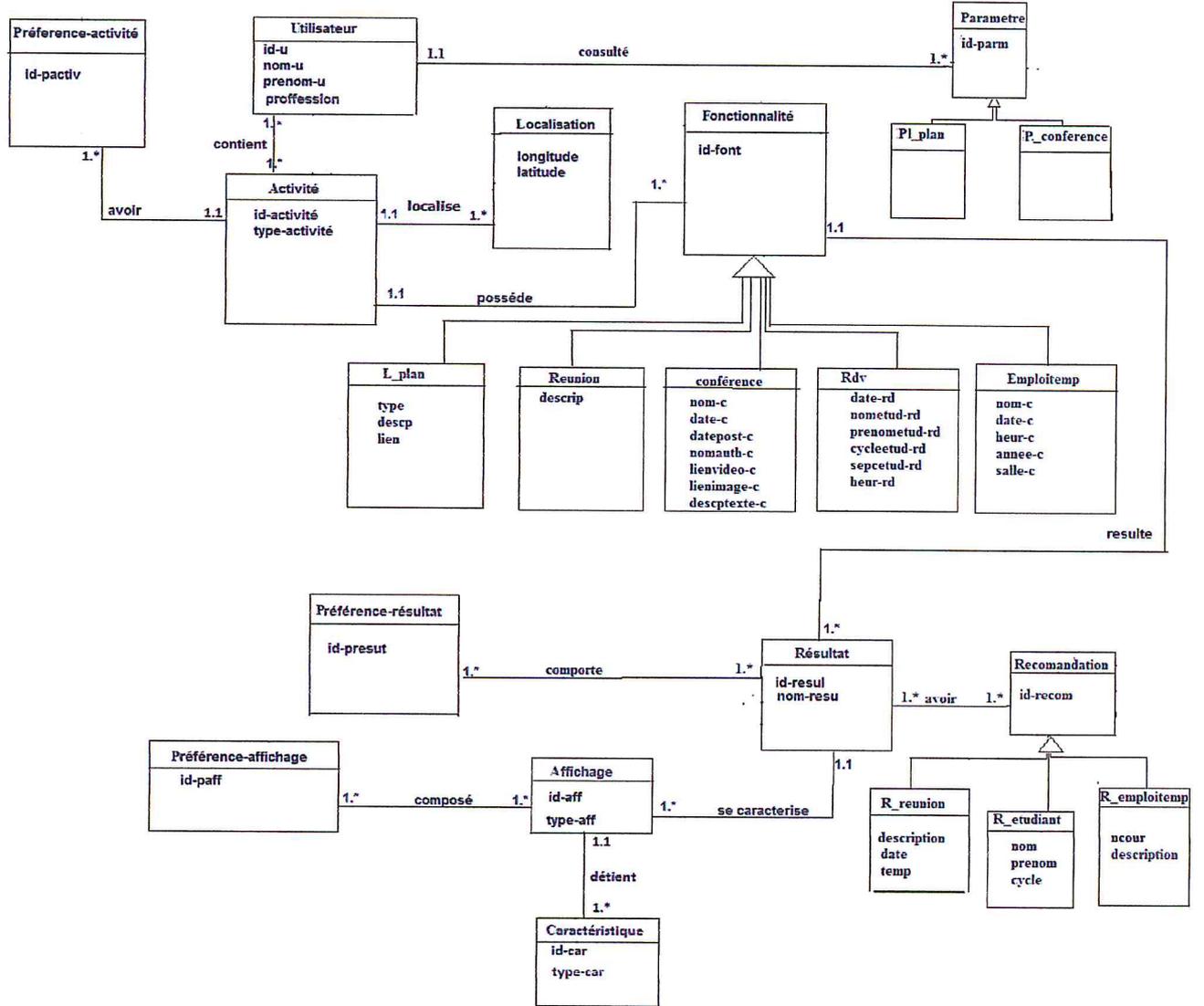


Figure 1.4. Diagramme de classes spécifique de profil contextuel.



Description des classes

Classe : <i>utilisateur</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-u	Code utilisateur	Integer	10	Identifiant
nom-u	Nom utilisateur	String	100	
prenom-u	Prénom utilisateur	String	100	
proffesion-u	Profession utilisateur	String	100	
Classe : <i>activité</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-activité	Code activité	Integer	10	Identifiant
type-activité	Type activité	String	100	
Classe : <i>localisation</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
longitude	Déterminé la localisation géodésique	String	100	Identifiant
latitude	Déterminé la localisation géodésique	String	100	Identifiant
Classe : <i>préférence activité</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-pactiv	Code préférence activité	Integer	10	Identifiant
Classe: <i>fonctionnalité</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id -font	Code fonctionnalité	Integer	10	Identifiant
Classe : <i>l plan</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
type	Type de plan	Integer	10	
descp	Description de plan	String	100	
lien	Lien d'image de plan	String	100	
Classe : <i>reunion</i>				
Attribut	Désignation de l'attribut	Type	Taille	Observation
descrip	Description de la réunion	String	100	

Classe : conférence				
Attribut	Désignation de l'attribut	Type	Taille	Observation
nom-c	Nom de la conférence	Integer	10	
date-c	Date de la conférence	String	100	
datepost-c	Date de poste de la conférence	String	100	
nomauth-c	Nom d'auteur de la conférence	String	100	
lienvidéo-c	Lien de la vidéo de la conférence	String	100	
lienimage-c	Lien de l'image de la conférence	String	100	
desciptexte-c	Description de la conférence	String	100	
Classe : Rdv				
Attribut	Désignation de l'attribut	Type	Taille	Observation
date-rd	Date de rendez vous	String	10	
nometud-rd	Nom de l'étudiant	String	100	
prenometud-rd	Prénom de l'étudiant	String	100	
cycleetud-rd	Cycle de l'étudiant	String	100	
sepecetud-rd	Spécialité de l'étudiant	String	100	
heur-rd	Heur de rendez vous	String	06	
Classe : Emploitemp				
Attribut	Désignation de l'attribut	Type	Taille	Observation
nom-c	Nom de cours	String	100	
date-c	Date de cours	String	100	
heur-c	Heur de cours	String	06	
année-c	Année universitaire	String	100	
salle -c	Salle de cours	Integer	10	
Classe : parametre				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-parm	Code préférence activité	Integer	10	Identifiant
Classe: fonctionnalité				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id -font	Code fonctionnalité	Integer	10	Identifiant
Classe : résultat				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-resu	Code résultat	Integer	10	Identifiant
nom-resu	Nom de résultat	String	100	
Classe : recommandation				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-recom	Code de recommandation	Integer	10	Identifiant

Classe : R_reunion				
Attribut	Désignation de l'attribut	Type	Taille	Observation
description	Description de la réunion à recommandé	String	100	
date	Date de la réunion à recommandé	String	10	
temp	Heur de la réunion à recommandé	String	06	
Classe : R_etudiant				
Attribut	Désignation de l'attribut	Type	Taille	Observation
nom	Nom de l'étudiant à recommandé	String	100	
prenom	Prénom de l'étudiant à recommandé	String	100	
cycle	Cycle de l'étudiant à recommandé	String	100	
Classe : R_emploitemp				
Attribut	Désignation de l'attribut	Type	Taille	Observation
ncour	Nom de cours à recommandé	String	100	
description	Description de cours à recommandé	String	100	
Classe : affichage				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id-aff	Code préférence activité	Integer	10	Identifiant
type -aff	Type de préférence d'activité	String	100	
Classe: caractéristique				
Attribut	Désignation de l'attribut	Type	Taille	Observation
id -cart	Code de caractéristique de DM	Integer	10	Identifiant
type-car	Type de caractéristique de DM	String	100	
Classe : préférence-affichage				
Attribut	Désignation de l'attribut	Type	Taille	Observation
Id-paff	Code préférence d'affichage	Integer	10	
Classe :préférence-resultat				
Attribut	Désignation de l'attribut	Type	Taille	Observation
Id-presult	Code préférence résultat	Integer	10	

2.3. Langage de représentation OWL

Le W3C a proposé OWL (acronyme de « *Ontology Web Language* »), OWL définit une syntaxe pour décrire et construire des vocabulaires afin de créer des ontologies. OWL est un langage basé sur RDF [24], il est défini en trois sous-langages de plus en plus expressifs, chacun étant une extension du précédent. OWL-Lite est le sous-langage le plus simple. Il est destiné aux utilisateurs qui ont besoin d'une hiérarchie de concepts simples. OWL-DL permet une expressivité bien plus importante. Il est fondé sur la logique descriptive qui est un domaine de recherche étudiant la logique, ce qui confère à OWL-DL son adaptation au raisonnement automatisé. OWL-Full est la version la plus complexe de OWL, mais également celle qui permet le plus haut niveau d'expressivité [73].

3. Traitement des informations

La phase de traitement est une phase qui permet le traitement des concepts obtenus à partir de la couche sémantique pour ce fait, nous avons proposé un pseudo algorithme de profil contextuel de l'utilisateur, cet algorithme permet de déduire l'activité préférée d'un utilisateur, on se focalise sur son activité ainsi son contexte d'utilisation.

Pseudo algorithme de profil contextuel

Etape 1 : Authentification () ;

Etape 2 : récupérer la localisation de l'utilisation

Var localisation = get_Localisation() ; //prendre la localisation

Etape 3 : recommander information (par localisation ou bien par activité préférée et localisation)

Si (l'utilisateur est nouveau)

Alors var activite_préférée = get_Activite(localisation) ; //prendre les activités qui correspond à sa localisation détecté

Sinon

Var Activite [] = get_Activite(id_utilisateur) ;

Var Préférence-A [] ;

Pour i < -1 à Activite.taille() //parcourir le tab d'activité jusqu'à sa dernière taille

Faire

Préférence-A [i] = getRepetitionActivité(Activite[i]) ; //prendre la valeur d'activité et recherche par fonction le nombre d'activité qui a été répéter

Fait

Var IndiceMaxi = Max (Préférence-A) ; //prendre l'indice de préférence d'activité la plus utilisé

```

Var activite_ préférer =Activité [IndiceMaxi] ; //retiré l'activité qui correspond a l'indice
trouvé
Si(activite_ préférer . getLocalisation()=localisation)
Alors //recommander cette selon cette activité préférer
Sinon //faire la recommandions par localisation
Finsi ;
    
```

Exemple :

(a) Premier cas : recommandation par activité préférer et localisation

Latitude =36,6645(université par exemple).

Longitude =36.0000 (université par exemple).

Tableaux Activité qui correspond la localisation détecté

Indice	Valeur	Localisation (latitude)
1	Chercheur	36,6645
2	Enseignant	36,6900

Activité = getActivity(36,6645) ;

Activité = {chercheur, Enseignant} ;

Tableaux qui correspondent les Préférence d'activité

Indice Préférence d'activité	Valeur
1	Chercheur
2	Enseignant
3	Chercheur
4	Chercheur

Préférence-A [1]=getRepititionActivité(Activité[1])

Préférence-A [1]=getRepititionActivité(chercheur)

Préférence-A [1]=3

Préférence-A [2]=getRepititionActivité(Activité[2])

Préférence-A [2]=getRepititionActivité(enseignant)

Préférence-A [2]=1

Tableaux de préférence-A

Indice	Valeur
1	3
2	1

IndiceMaxi =Max (Préférence-A)

IndiceMaxi =1

Activité-Pref=Activité [1]

Activité-Pref=chercheur

(b) Deuxième cas : recommandation par localisation

Si Latitude =36,6645(université par exemple)

Alors activité_ préférer =chercheur.

4. La recommandation

La recommandation est une phase de la couche système de recommandation sensible au contexte, La recommandation se réalise une fois que d'une part les informations sont collectées (contexte d'utilisation, profil) , le contexte d'utilisation est constitué d'information sur la localisation actuelle de l'utilisateur, La localisation de l'utilisateur peut être obtenue à l'aide d'un dispositif GPS ou par d'autre méthode ,nous utilisons une API commerciale (Google maps) qui permet de passer d'une localisation à base de coordonnées à une adresse .Cependant, le profil est constitué d'information sur les activités de l'utilisateur .

D'un autre part, l'appariement de l'ensemble des informations collectées par rapport aux classes d'ontologie sont faites au niveau de la phase de la modélisation contextuel .Afin de déduire l'activité préférée d'un utilisateur, la phase de traitement se base sur l'activité de l'utilisateur et sur son emplacement.

Le système de recommandation sensibilité aux contextes permet d'adapter l'information à un utilisateur selon ses informations contextuelles, ses informations se focalisent sur le contexte d'utilisation est l'activité de l'utilisateur.

L'accès aux SIW nous permettra de découvrir les préférences d'activité des différents utilisateurs, cela nous permettra de recommander le SIW approprié à l'utilisateur.

Exemple de recommandation de SIW :

Utilisateur 1 à comme préférence activité « architecte », utilisateur 2 à comme préférence activité « enseignant », ses utilisateurs ont un accès aux SIW, pour l'utilisateur 1 en lui recommande le SIW adéquat à l'activité « enseignant » et pour l'utilisateur 2 en lui recommande le SIW adéquat à l'activité« architecte ».

(a). Approche heuristique

Pour l'utilisation de l'information contextuelle, nous avons utilisé l'approche de modélisation contextuel qui représente des modèles prédictifs pour évaluer la classe d'un objet ou pour évaluer la

valeur ou l'intervalle d'un attribut, pour cela , nous avons présenté un exemple d'approche heuristique pour cette modélisation.

Nous montrons dans cet exemple comment notre approche fonctionne dans une application de recommandation de donnée pour un utilisateur d'activité enseignant , chercheur ou architecte, Nous définissons les différentes activités (fixer cours , faire les rendez- vous, assister à la conférence, gérer les soutenances ...).

Au niveau des modèles, Chacun est défini comme une variable booléenne .Par exemple, un enseignant soit «il assiste à une conférence » soit «il n'assistera pas à une conférence » au moment de sa localisation au niveau de l'université.

Par exemple, notre premier modèle représente un utilisateur-chercheur qui va à l'université pour assister à une conférence, Un ensemble de règles de cause peut être utilisé pour modéliser la façon dont un tel utilisateur sera localisé à l'université pour lui recommander le planning de toutes les conférences présentées dans une date précise.

Exemple de recommandation des informations contextuelle d'utilisateur (enseignant, chercheur ou architecte) :

Model: Traitement d'utilisateur qui a comme une activité chercheur, enseignant ou architecte.

UL : utilisateur lieu.

US : utilisateur situation.

UP : utilisateur préférence.

if UL. Lieu == "université" then

US. Activité = "chercheur"; UP. Préférence = "consulter planning conférence";

UP. Préférence = "consulter réunion prévu";

else if UL.Lieu == "département" then

US.Activité = "enseignant "; UP. Préférence = "faire RDV"; UP. Préférence = "fixer cours";

if UL. Lieu == "bureau d'étude " then

US. Activité = "architecte"; UP. Préférence = "consulter plan ";

Heuristique M1

M1 : « consulter planning conférence »

UA : université.

UC : utilisateur chercheur

if UA. Programme == "présentation conférence"

then if date = 19/07/2014

then if UC.coopère == "assister conférence"

then valueof(M1) = "yes";

else if UC.coopère == "n' assiste pas conférence "

then valueof(M1) = "no";

else if UA.Programme == " presentation conference" then

```
if date >19/07/2014
then if US.coopère ==“visualiser conference” then valueof(M1)=“yes”;
```

Heuristique M2

M1 : « consulter réunion prévu »
UA : université.
UC : utilisateur chercheur

```
if UA.Programme ==“présentation réunion”
then if date =07/11/2014
then if UC.coopère ==“assister réunion”
then valueof(M1)=“yes”;
else if UC.coopère ==“n’assiste pas réunion ”
then valueof(M1)=“no”;
else if UA.Programme ==“ presentation réunion” then
if date >07/11/2014 then if US.coopère ==“visualiser réunion” then valueof(M1)=“yes”.
```

Heuristique M3

M2 : faire RDV

LE : liste des étudiants de fin d'étude
UE : utilisateur enseignant

```
If UE.Activité = “enseignant”; then if LE.cycle== “Master2”; then if LE.cycle == “licence ”; then if
S.spécialité = “informatique ” then if T.time=mm/ss/hh then valueof(M1)=“yes”.
```

Heuristique M4

M2 : fixer cours.
LE : liste des étudiants.
UE : utilisateur enseignant.
SL : salle.
LC: liste cours.

```
If UE.Activité = “enseignant”; then if LE.cycle== “Master2”; then if LE.cycle == “licence ”; then if
SL.value= integer ; then if T.time=mm/ss/hh ; then D.date = jour /moi /année ; then if C.cours ∈LC
then valueof(M1)=“yes”.
```

Heuristique M5

M1 : « consulter plan»
UA : bureau d'étude.
UR : utilisateur architecte

```
if UA.Programme ==“présentation plan”
then if US.coopère ==“visualiser plan” then valueof(M1)=“yes”.
```

(b). Filtrage des informations

La recommandation à pour objective de recommander aux utilisateurs des items et un SIW selon son profil contextuel. Les méthodes de filtrage de système de recommandation «filtrage basée sur le contenu » et « filtrage collaboratif » (cf. chapitre système de recommandation sensible au contexte) sont ceux qui sont utilisées dans notre proposition

Filtrage collaboratif

Pour la méthode « filtrage collaboratif », nous essayons de recommander à l'utilisateur un SIW en tenant compte la similarité entre les préférences d'activités d'un utilisateur et celles d'autres utilisateurs.

Par exemple : on a utilisateur1 et utilisateur 2 qui ont comme préférence activité «enseignant », donc il semble que ces deux utilisateurs ont une activité similaire .Cette méthode consiste à trouver des utilisateurs ayant des préférences d'activité similaire à d'autre utilisateurs, cela permet à l'utilisateur courant de lui recommander un SIW qui correspondent à ses préférences activités.

Filtrage basé sur le contenu

Pour la méthode « basée sur le contenu » nous essayons de recommander à l'utilisateur des items utilisés par lui dans le passé.

Par exemple : utilisateur 1 a comme activité « chercheur », cet utilisateur à l'habitude de consulter les conférences, le système permet de lui recommander des conférences a déjà vue ou des conférences qui se porte sur le même sujet.

5. Conclusion :

Ce chapitre nous a permis de tracer plusieurs perspectives, en premier lieu nous avons collecté les informations contextuelles, en second lieu nous avons réalisé une modélisation de profil contextuel en utilisant une ontologie, ensuite en utilisant UML, nous avons choisi le langage OWL- DL pour décrire les classes de profil contextuel, finalement nous avons mis l'accent sur la recommandation afin d'aboutir à une donnée pertinente.

Chapitre 5

Implémentation et Réalisation

1. Introduction

Ce chapitre, donne une description détaillée du système conçu. Elle permet de détailler la démarche de réalisation de l'application mobile et les technologies associées. Afin de valider notre proposition présentée dans la (section 4), nous avons construit une application de recommandation sensible au contexte qui adapte l'information aux utilisateurs mobile et qui lui permet l'accès via le SIW adéquat à ses préférences d'activités de l'utilisateur en se basant sur ses activités et son emplacement actuel.

2. Présentation des outils du développement

Maintenant, nous abordons la phase de la réalisation on va présenter les outils du développement et les langages de programmation utilisés. Enfin nous présenterons quelques interfaces du travail réalisé.

2.1. Les langages de programmation

Eclipse JEE est un IDE (environnement de développement intégré) puissant pour Java. Il permet au développeur de gérer ses projets avec une interface riche, il est plus rapide, plus efficace et plus fiable que d'autre Java IDE.

Nous avons travaillé avec deux langages : java pour le développement des composants et Java Server Pages (JSP) pour les interfaces (IHM). Les JSP et en générale des pages HTML contenant du code Java.

2.2 MySQL :

MySQL (Structured Query Language, traduisez langage de requêtes structuré) est un système de gestion de base de données (SGBD) qui fonctionne dans une architecture client-serveur, il utilise transact-SQL (LDD, LMD et LCD) comme langage d'interrogation et de transactions sur les bases de données.

3. Les interfaces de l'application

3.1 L'interface principale

La figure 1.2 présente une copie d'écran de l'interface principale pour la plate forme de système de recommandation sensible au contexte pour un accès au SIW mobile. En cliquant sur cette interface, l'utilisateur peut se connecter afin d'accéder au SIW.

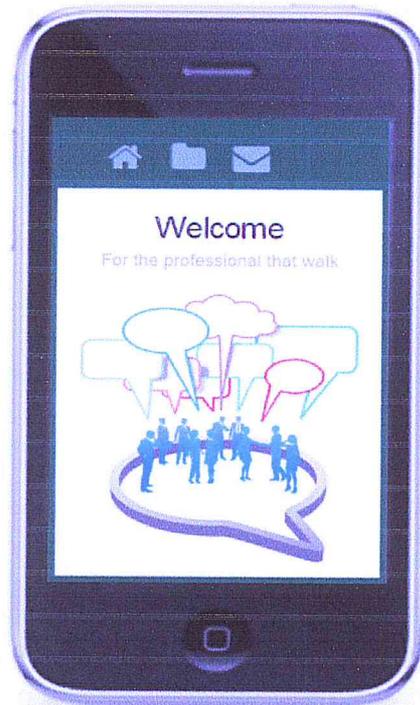


Figure 2.1. Interface principale

3.2 Interface d'authentification de l'utilisateur :

La figure 2.2 présente l'interface d'authentification pour la plate forme de système de recommandation sensible au contexte pour un accès au SIW mobile. Chaque utilisateur se connecte à la plate forme il doit s'authentifier.

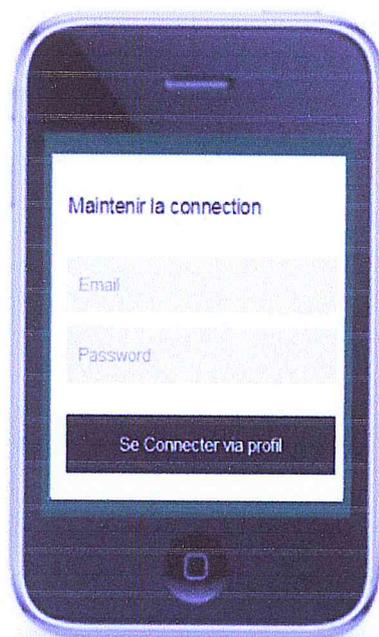


Figure 2.2. Interface d'authentification de l'utilisateur.

3.3 L'interface de création du compte

La figure 2.3 présente l'interface de création du compte pour la plate forme de système de recommandation sensible au contexte pour un accès au SIW mobile. Chaque nouvel utilisateur veut accéder à la plate forme de SIW, il doit d'abord créer un compte.

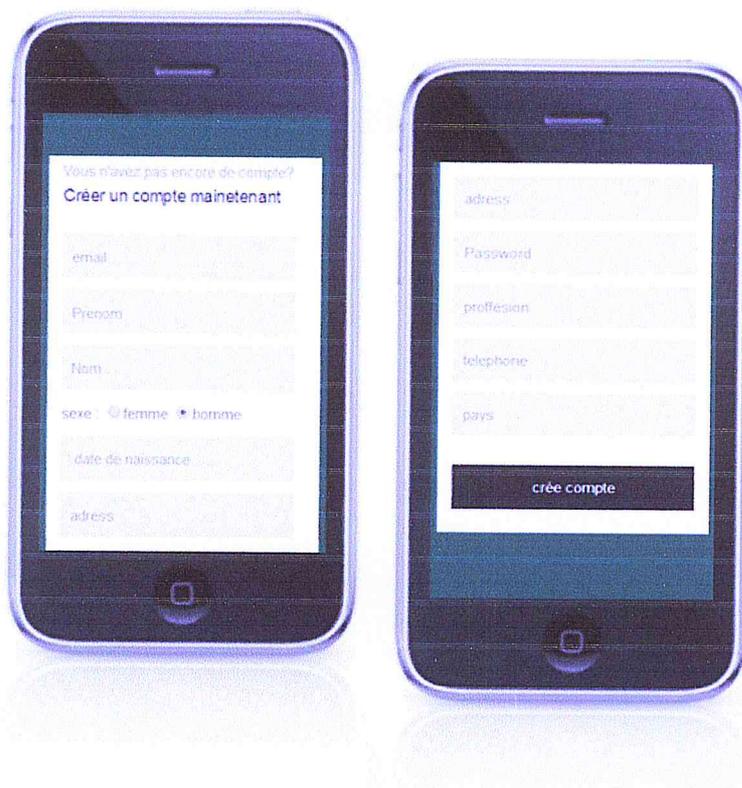


Figure 2.3: L'interface de création du compte

3.4 Interface de localisation d'un utilisateur de profession « architecte »

La figure 2.5 présente l'interface de localisation, cette interface à pour but de localiser un nouveaux utilisateur d'activité « architecte » ou « enseignant » ou « chercheur ». Une fois que l'utilisateur nouveau s'authentifie d'une part, d'une manière implicite il indique son activité, d'un autre part, d'une manière explicite le système détecte sa localisation actuel.

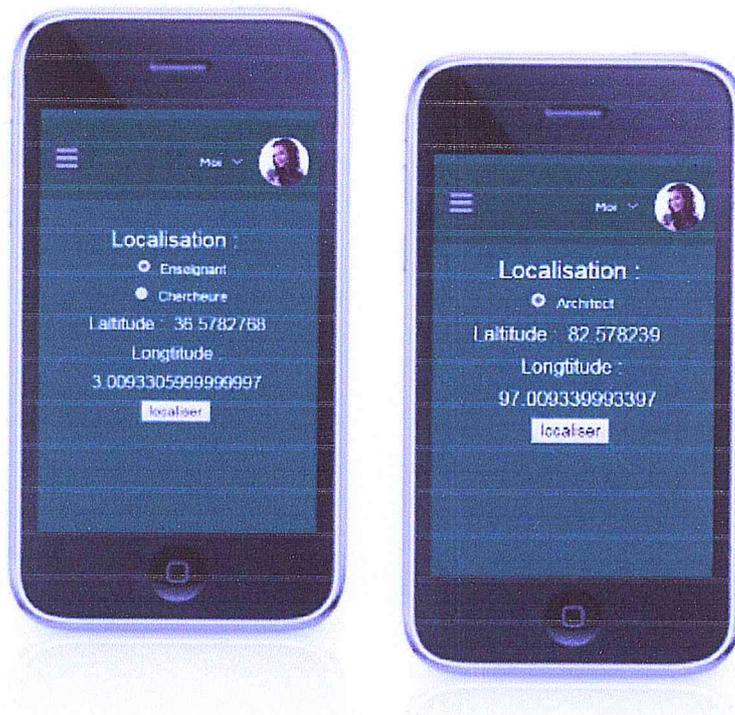


Figure 2.4 L'interface de localisation

3.5 Interface de recommandation de SIW approprié à l'activité d'un utilisateur

La figure 2.4 montre la recommandation des fonctionnalités de l'utilisateur d'activité « enseignant » « chercheur », « architecte », une fois que l'utilisateur s'est authentifié en lui recommande le SIW approprié selon ses préférences d'activité.



Figure 2.5 L'interface de recommandation fonctionnalité

3.6 Interface de recherche de fonctionnalité

La figure 2.6 présente l'interface de recherche de fonctionnalité, cette interface a pour but d'afficher les résultats recherchés indiqués par l'utilisateur. Par exemple, si l'utilisateur recherche la liste des rendez-vous, le système lui affiche toutes les rendez-vous prévus.



Figure 2.6 Interface de recherche de fonctionnalité

3.7 Interface de recommandation de plan

La figure 2.7 présente l'interface de recommandation de plan d'un utilisateur d'activité « architecte », l'utilisateur a l'habitude de consulter la liste des plan, le but de recommandation est d'essayer de lui recommander des plans qui portent sur le même sujet qu'il a déjà consulté, cette interface nous donne les résultats de la recommandation de plan.

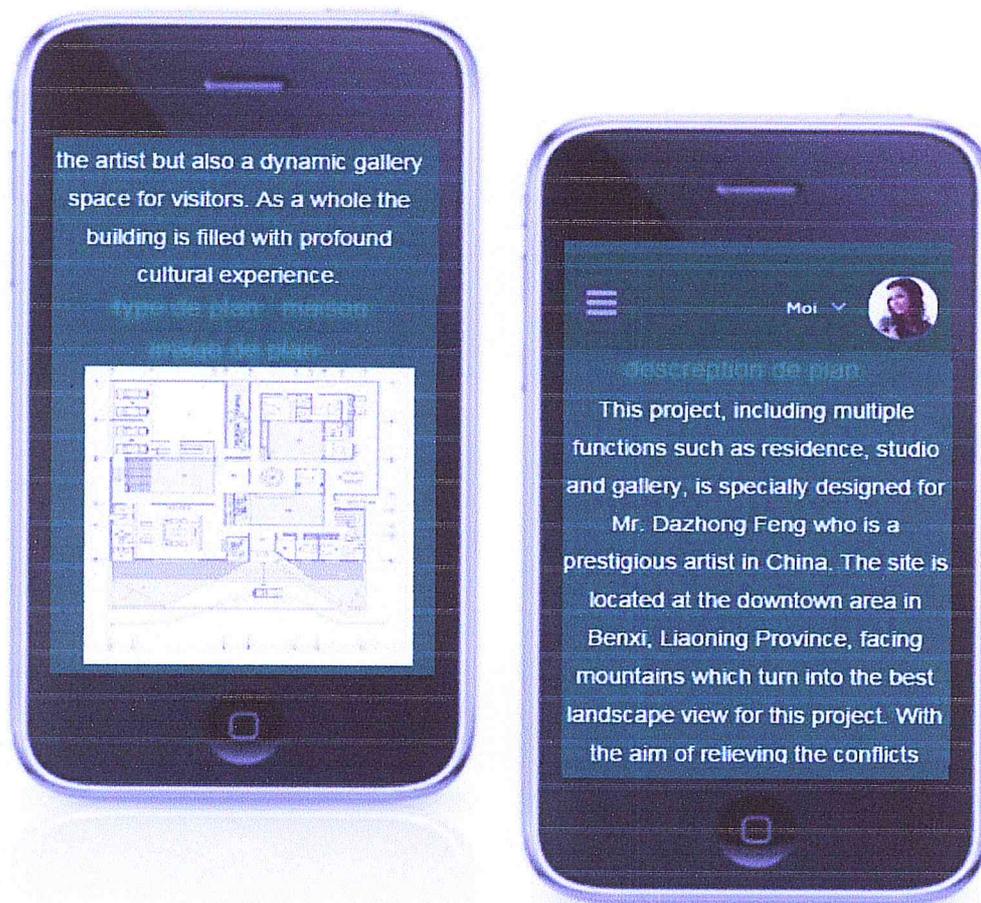


Figure 2.7 L'interface de recommandation de plan

4. Conclusion

La phase de réalisation est une étape importante dans la conception de notre application, cette phase nous a permise d'avoir le résultat concret de toutes les étapes précédentes .Pour réaliser notre application, nous avons recours à plusieurs outils du développement .Nous avons choisi java/EE parce que c'est un langage de haut niveau et portable sur d'autres plateformes et pour le système de gestion de base de données nous avons choisi MySql.

Conclusion générale :

De nos jours, de nouveaux besoins en systèmes d'information sont apparus, pour cela avec l'émergence de la technologie, des systèmes d'information dits ubiquitaire sont mise en œuvre, afin que l'utilisateur nomade puisse avoir l'information quelque soit le moment ou le lieu où il se trouve.

Dans notre mémoire, nous avons présenté l'importance d'utiliser un système de recommandation afin d'adapter les informations les plus pertinentes a l'utilisateur, notre problématique alors concerne l'accès aux SIW à travers les différents dispositifs mobiles.

L'élaboration de notre travail permet de concevoir un système qui adapte l'information et qui permet aux utilisateurs nomades d'accéder a un SIW en tenant compte leurs mobilités, l'utilisation des différents dispositifs mobiles et de leurs capacités, comme solutions nous avons mis le point sur un système de recommandation sensible au contexte pour un accès adapté au SIW en fournissant une information pertinente aux utilisateurs nomade en fonction de leurs dispositifs mobiles.

Dans notre proposition, nous avons identifié différentes approches de modélisation du contexte, Ceci nous a permis de choisir celle qui nous semble être la mieux adaptée à la description du contexte dans un environnement mobile.

Pour un accès à un SIW mobile, une architecture a été proposée, nous avons identifié trois contributions principales :

- ✦ La première contribution réside dans la phase de collecte d'informations contextuelles, cette phase repose sur le profil d'utilisateur et son contexte, son rôle est d'extraire son activité d'après son profil et sa localisation pour définir son emplacement actuel.
- ✦ La deuxième contribution concerne la représentation et la réalisation d'une ontologie qui nous a permis de modéliser le profil contextuel.
- ✦ La troisième contribution se focalise sur le système de recommandation sensible au contexte, ce système permet d'adapté l'accès a un SIW mobile.

Bibliographie

- [01] Mathieu Petit, « Approche spatiale pour la caractérisation du contexte d'exécution d'un système d'information ubiquitaire », Thèse de doctorat, l'École Nationale Supérieure d'Arts et Métiers Spécialité “ Informatique ”, 2010
- [2] Samuel pierre, Réseaux et système informatique mobile, Edition revue et augmentée, (consulté le avril 2003).
- [3] [http://www.deptinfo.unicc.fr/introduction\(index.html\)/partic2 les systèmes collaboratifs \(collaboraticiel\).html](http://www.deptinfo.unicc.fr/introduction(index.html)/partic2%20les%20systèmes%20collaboratifs(collaboraticiel).html), (Consulté en 18 avril 2014).
- [4] <http://wapiti.telecom-lille1.eu/introduction/introduction.html>, (Consulté en 27 avril 2014).
- [5] Angela Carrillo-Ramos, Marlène Villanova-Oliver, Jérôme Gensel et Hervé Martin, Gestion des préférences utilisateurs pour les Systèmes d'Information ubiquitaires, France,2006.
- [6] Djallel Bouneffouf , « Recommandation mobile, sensible au contexte de contenus évolutifs: Contextuel-q-Greedy », Department of Computer Science, Télécom SudParis, UMR CNRS Samovar, 91011 Evry Cedex, France , 2014.
- [7] Anind K. Dey. Understanding and using context. *Personal Ubiquitous Computing*, 5(1):4–7, 2001.
- [8] Benazzouz Yazid, « Découverte de contexte pour une adaptation automatique de services en intelligence ambiante», Thèse de doctorat, École Nationale Supérieure des Mines de Saint-Étienne, page 29-31,2011.
- [9] Schmidt, A., Beigl, M. et Gellersen, H. (1999). There is more to context than location. *Comput Graphics*, 23(6):893_901.
- [10] Henricksen, K. (2003). *A Framework for context-aware pervasive computing applications*. Thèse de doctorat, University of Queensland, Queensland, Australia.
- [11] Behlouli Belhanafi Nabiha , « Ajout de mécanismes de réactivité au contexte dans les intergiciels pour composants dans le cadre d'utilisateurs nomades», Thèse de doctorat, l'Université d'Évry Val d'Essonne , 2006.
- [12] P. J. Brown, N. Davies, M. Smith, and P. Steggles. Towards a better understanding of context and context-awareness. In H.-W. Gellerson, editor, *Handheld and ubiquitous*

Bibliographie

computing, number 1707 in Lecture Notes in Computer Science, pages 304–307. Springer, September 1999.

[13] B. Schilit, N. Adams, and R. Want. Context-Aware Computing Applications. In IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, US, 1994.

[14] P. J. Brown, J. D. Bovey, and X. Chen. Context-aware Applications : from the Laboratory to the Marketplace. IEEE Personal Communications, 4(5) :58–64, October 1997.

[15] D. Chalmers, N. Dulay, and M. Sloman. Towards Reasoning About Context in the Presence of Uncertainty. In Proceedings of Workshop on Advanced Context Modelling, Reasoning And Management at UbiComp 2004, 2004.

[16] Nachida Rezoug, Fahima Nader, Omar Boussaid, "Implémentation d'OLAP dans les environnements mobiles : aperçu de l'état de l'art". International Conference on Information Systems and Technologies Tebessa, Algeria, ICIST, 24-26 April 2011.

[17] El Ghayam Yassine, « La Sensibilité au Contexte dans un Environnement Mobile », thèse de doctorat, Université Mohammed V Souissi – RABAT, 2011.

[18] Henricksen K. and Indulska J., "Developing context-aware pervasive computing applications: Models and approach", Journal of Pervasive and Mobile Computing, Vol 2, N 1, pp 37-64, 2006.

[19] Bettini C., Brdiczka O., Henricksen K., Indulska J., Nicklas D., Ranganathan A., and Riboni D., "A survey of context modelling and reasoning techniques," Journal of Pervasive and Mobile Computing, Special Issue on Context Modelling, Reasoning and Management, Vol. 6, N 2, pp 161–180, 2010

[20] Soukkarieh Bouchra, « Technique de l'internet et ses langages : vers un système d'information Web restituant des services Web sensibles au contexte», thèse de doctorat , l'Université Toulouse III - Paul Sabatier,2010

[21] Mostéfaoui K., Pasquier-Rocha J. et Brézillon P. (2004). Contextaware computing: a guide for the pervasive computing community. In Proceedings of the IEEE/ACS International Conference on Pervasive Services (IPCS'04), p. 39-48, USA.

[22] M. Uschold and M. Grüninger. Ontologies : principles, methods, and applications. Knowledge Engineering Review, 11(2) :93–155, 1996.

Bibliographie

- [23] W3C. OWL Web Ontology Language Use Cases and Requirements. <http://www.w3.org/TR/webont-req/>, W3C Recommendation ,(Consulté en 10 juillet 2014).
- [24] W3C. Resource Description Framework (RDF) : Concepts and Abstract Syntax. <http://www.w3.org/TR/rdf-concepts/>, W3C Recommendation (Consulté en juillet 2014).
- [25] Villanova-Oliver M. (2002). Adaptabilité dans les systèmes d'information sur le web : Modélisation et mise en œuvre de l'accès progressif. Thèse de doctorat, Institut Nationale Polytechnique de Grenoble, France.
- [26] Angela Cristina CARRILLO RAMOS, « Agents ubiquitaires pour un accès adapté aux systèmes d'information : Le Framework PUMAS» thèse de doctorat, Université Joseph Fourier, 2007
- [27] Jérôme Gensel, Marlène Villanova-Oliver, Manuele Kirsch-Pinheiro , « Modèles de contexte pour l'adaptation à l'utilisateur dans des Systèmes d'Information Web collaboratifs», Université Leuven.
- [28]Virgilio R. D. et Torlone R. (2005). A General Methodology for Context-Aware Data Access. In Proceedings of the 4th ACM International Workshop on Data engineering for wireless and mobile access (MobiDE'05), p. 9-15, Baltimore, Maryland, USA.
- [29] Kirsch-Pinheiro M. (2006). Adaptation Contextuelle et Personnalisée de l'Information de Conscience de Groupe au sein des Systèmes d'Information Coopératifs, Thèse de doctorat, Université Joseph Fourier, Grenoble 1, France.
- [30] Hinz M. et Fiala Z. (2004). AMACONT: A System Architecture for Adaptive Multimedia Web Applications. In Proceedings of the 11th international conference on 3D web technology, p. 65-74, Germany.
- [31] Jameson A. (2003). Adaptive Interfaces and Agents. In J. Jacko, A. Sears, éditeurs: the Human Computer Interaction Handbook, chapter 15, p. 305-330, USA.
- [32] Simonin J. et Carbonell N. (2007). Interfaces adaptatives : adaptation dynamique à l'utilisateur courant. In Imad S. et Djef R, éditeurs : Interfaces Numériques (Collection information, hypermédias et communication), p. 37-54, Paris, Hermès et Lavoisier.
- [33] Beggas M. (2005). Modélisation par un système multi-agents d'un hypermédia éducatif adaptatif dynamique. Rapport de stage de Master, Université de Cheick Larbi Tébessi, Algérie.

Bibliographie

- [34] Opperman R., Rashev R. (1997). Adaptability and Adaptivity in Learning Systems. In A. Behrooz, éditeur : Proceedings of Knowledge Transfer, volume 2, p. 173-179, London.
- [35] Brusilovsky P. et Pesin L. (1994). ISIS-tutor: An adaptive hypertext learning environment. In H. Ueno et V. L. Stefanuk, éditeurs: JCKBSE'94: Proceedings of the Japan-CIS Symposium on Knowledge Based Software Engineering, p. 83-87, Tokyo.
- [36] Subramanian N. et Chung L. (2001). Software Architecture Adaptability: An NFR Approach. In Proceedings of the International Workshop on Principles of Software Evolution (IWPSE'01), p. 52-61, Vienna, Austria, ACM Press.
- [37] Kaddour M. (2005). Conception et Réalisation d'une Plate-forme de Réalisation dédiée aux Environnements Nomades. Thèse de doctorat, Ecole Nationale Supérieure des Télécommunications de Paris, France.
- [38] Brusilovsky, P. Methods and Techniques of Adaptive Hypermedia. In: User Modeling and User Adapted Interaction, vol 6, no 2-3 (1996), pp. 87-129.
- [39] Brusilovsky, P. Adaptive Hypermedia: From Intelligent Tutoring Systems to Web Based Education. In: Gauthier, G., Frasson, C., VanLehn, K. (eds.): Proceedings of the International Conference on Intelligent Tutoring Systems (ITS 2000) (Montreal, Canada, June 19-23, 2000), Lecture Notes in Computer Science, vol. 1839, Springer-Verlag, Berlin Heidelberg (2000), pp. 1-7.
- [40] Carabelea, C., Boissier, O., Ramparany, F. Benefits and Requirements of Using Multi-agent Systems on Smart Devices. In: Kosch, H., Böszörményi, L., Hellwagner, H. (eds.): Proceedings of the 9th International Euro-Par Conference on Parallel Processing (Euro-Par 2003) (Klagenfurt, Austria, August 26-29, 2003), Lecture Notes in Computer Science, vol. 2790, Springer-Verlag, Berlin Heidelberg (2003), pp. 1091-1098.
- [41] Mili Seif Eddine, « conception d'une architecture basée agents pour la création d'un monde virtuel », Université Mentouri de Constantine, Magister en informatique, 2008.
- [42] Bouzeghoub, M., Kostadinov, D. Personnalisation de l'information : aperçu de l'état de l'art et définition d'un modèle flexible de profils. Actes de CORIA 2005 (Grenoble, France, 9-11 mars, 2005), pp. 201-218.
- [43] Romain Picot-Clément, « Une architecture générique de Systèmes de recommandation de combinaison d'items. Application au domaine du tourisme », l'Université de Bourgogne, thèse de doctorat, 2011[50].

Bibliographie

[44] Tamine, L., Bahsoun, W. Définition d'un profil multidimensionnel de l'utilisateur. Actes de CORIA 2006 (Lyon, France, 15-17 mars, 2006), pp. 225-236.

[45] Zemirli, N., Lechani Tamine, L., Boughanem, M. Accès personnalisé à l'information : Proposition d'un profil utilisateur multidimensionnel. In: Proceedings of the 7th International Symposium on Programming and Systems (ISPS'2005) .

[46] Hafenrichter, B., Kießling, W. Optimization of Relational Preference Queries. In: Williams, H. E., Dobbie, G. (eds.): Proceedings of the 16th Australasian Database Conference (ADC 2005) (Newcastle, Australia, January 31st - February 3rd 2005), Conferences in Research and Practice in Information Technology, vol. 39 (2005), pp. 175-184

[47] Freuder, E.C., Likitvivatanavong, C., Moretti, M. Rossi, F., Wallace, R.J. Computing Explanations and Implications in Preference-Based Configurators. In: O'Sullivan, B. (ed.): Proceedings of the Recent Advances in Constraints, Joint ERCIM/CologNet International Workshop on Constraint Solving and Constraint Logic Programming (Cork, Ireland, June 19-21, 2002), Selected Papers, Lecture Notes in Computer Science, vol. 2627, Springer-Verlag, Berlin Heidelberg (2003), pp. 76-92.

[48] Hafenrichter, B., Kießling, W. Optimization of Relational Preference Queries. In: Williams, H. E., Dobbie, G. (eds.): Proceedings of the 16th Australasian Database Conference (ADC 2005) (Newcastle, Australia, January 31st - February 3rd 2005), Conferences in Research and Practice in Information Technology, vol. 39 (2005), pp.175-184.

[49] Kießling, W. Preference Queries with SV-Semantics. In: Haritsa, J.R., Vijayaraman, T. M. (eds.): Proceedings of the 11th International Conference on Management of Data Advances in Data Management (COMAD 2005) (Goa, India, January 6-8, 2005) [En ligne] Computer Society of India (2005), pp. 15-26.

[50] NEGRE Elsa, «Les systèmes de recommandation », Université Paris-Dauphine, (Consulté en juillet 2014).

[51] geoffary bonnin , « vers des systemes de recommandation robustes pour la navigation Web : inspiration de la modélisation statistique du langage»,thèse de doctorat ,université Nancy2,2010.

[52] Resnick, P. and Varian, H. R. 1997. Recommender Systems, Communications of the ACM, 40(8):56-58.

Bibliographie

- [53] [http://www.podcastscience.fm/dossier/2012/04/25/les algorithmes de recommandation](http://www.podcastscience.fm/dossier/2012/04/25/les_algorithmes_de_recommandation) , (Consulté en juillet 2014).
- [54] D. Goldberg, D. Nichols, B.M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12) :61– 70, 1992
- [55] M. Pazzani and D. Billsus. *The Adaptive Web*, chapter Content-Based Recommendation Systems, pages 325–341. Springer Berlin / Heidelberg, 2007.
- [56] R. Burke, K. Hammond, and E. Cooper. Knowledgebased navigation of complex information spaces. In *Proc. of the 13th National Conference on Artificial Intelligence (AAAI'96)*, pages 462–468, Menlo Park, Canada, 1996.
- [57] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems : A survey of the state-of-the-art. *IEEE Transactions on Knowledge and Data Engineering*, 17(6) :734–749, 2005.
- [58] Armelle Brun, Ahmad Hamad, Olivier Buffet, Anne Boyer, « Vers l'utilisation de relations de préférence pour le filtrage collaboratif », Nancy-Université, 2009.
- [59] Amokrane BELLOUI, « L'usage des concepts du web sémantique dans le filtrage d'information collaboratif », Mémoire de magistère, Institut National d'Informatique-Alger, 2008.
- [60] Schafer J. B., Konstan J., Riedi J., « Recommender systems in e-commerce », *Proceedings of the 1st ACM conference on Electronic commerce, EC '99*, ACM, New York, NY, USA, p. 158-166, 1999.
- [61] Damien Poirier , Françoise Fessant, Isabelle Tellier, « De la classification d'opinions à la recommandation : l'apport des textes communautaires », 2011.
- [62] Robin Burke, « Knowledge-based recommender systems », University of California, Irvine, 2000.
- [63] Gediminas Adomavicius and Alexander Tuzhilin, « Context-Aware Recommender Systems », 2010.
- [64] Schilit, B.N., and Theimer, M.M., Disseminating active map information to mobile hosts. *IEEE network*, 8(5):22–32, 1994.

Bibliographie

- [65] Ryan, N., Pascoe, J., and Morse, D., Enhanced Reality Fieldwork: the Context-Aware Archaeological Assistant. Gaffney, V., van Leusen, M., Exxon, S.(eds.) Computer Applications in Archaeology. British Archaeological Reports, Oxford, 1997.
- [66] Dey, A.K., Abowd, G.D., and Salber, D., A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16(2):97–166, 2001.
- [67] Adomavicius, G., and Tuzhilin, A., Incorporating context into recommender systems using multidimensional rating estimation methods. In *Proceedings of the 1st International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces (WPRSIUI 2005)*, 2005.
- [68] Breese, J.S., Heckerman, D., and Kadie, C., Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, volume 461, pages 43–52. San Francisco, CA, 1998.
- [69] Sarwar, B., Karypis, G., Konstan, J., and Reidl, J., Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.
- [70] Ansari, A., Essegai, S., and Kohli, R., Internet recommendation systems. *Journal of Marketing Research*, 37(3):363–375, 2000.
- [71] Gruber, T.R. A translation approach to portable ontologies. In: *Knowledge Acquisition*, vol. 5, no. 2, June 1993. Academic Press. (1993), pp. 199-220.
- [72] Buccella, A., Cechich, A., Brisaboa, N.R. Ontology-Based Data Integration Methods: A Framework for Comparison. In: Arenas, a., Perez, J., Carrillo, E. (eds.): *Colombian Journal of Computation*. vol. 6, no 2. Décembre 2005 [En ligne]. Disponible sur: http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r61_art3_c.pdf (Consulté en aout 2014).
- [73] Introduction à OWL, un langage XML d'ontologies Web , Xavier Lacot , juin 2005.

Exemple d'ontologie OWL DL de profil contextuel

Le diagramme de classes en UML de la figure 1.4 « Diagramme de classes spécifique de profil contextuel » indique les classes de profil contextuel d'un utilisateur dont une description détaillée est faite comme suite :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE rdf:RDF [
<!ENTITY owl "http://www.w3.org/2002/07/owl#" >
<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >]>

<rdf:RDF
xmlns = "&Siw;"
xmlns:Siw = "&Siw;"
xml:base = "&Siw;"
xmlns:owl = "http://www.w3.org/2002/07/owl#"
xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
xmlns:xsd = "http://www.w3.org/2001/XMLSchema#">
<owl:Ontology rdf:about="">
<rdfs:comment>Ontologie décrivant le profil contextuel</rdfs:comment>
<rdfs:label>Ontologie de profil contextuel</rdfs:label>
</owl:Ontology>

<!-- Définition des classes -->

<owl:Class rdf:ID="Utilisateur" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#contient" />
<owl:minCardinality rdf:datatype="&xsd;int">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>

<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#consulté" />
<owl:minCardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

Annexe

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-u" />
<owl:cardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nom-u" />
<owl:Cardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#prenom-u" />
<owl:Cardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#proffesion" />
<owl:minCardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="Parametre" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-parm " />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="Pl_plan">
<rdfs:subClassOf rdf:resource="#parametre" />
```

Annexe

```
</owl:Class>
```

```
<owl:Class rdf:ID="P-conference ">  
<rdfs:subClassOf rdf:resource="#parametre " />  
</owl:Class>
```

```
<owl:Class rdf:ID="Activité" >  
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#localise" />  
<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>  
</owl:Restriction>  
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#avoir"/>  
<owl:minCardinality  
rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>  
</owl:Restriction>  
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#possède"/>  
<owl:minCardinality  
rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>  
</owl:Restriction>  
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#id-activité" />  
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>  
</owl:Restriction>  
</rdfs:subClassOf>  
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#type-activité" />  
<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>  
</owl:Restriction>  
</rdfs:subClassOf>  
</owl:Class>
```

Annexe

```
<owl:Class rdf:ID="Localisation" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#longitude" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#latitude" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="préférence-activité" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id_pactiv" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="Fonctionnalité" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#resulte" />
<owl:minCardinality rdf:datatype="&xsd;int">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-fonct" />
<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nom-fonct" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

Annexe

```
</owl:Class>
```

```
<owl:Class rdf:ID="L_plan ">  
<rdfs:subClassOf rdf:resource="#Fonctionnalité " />  
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#type" />  
<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>  
</owl:Restriction>  
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#desc" />  
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>  
</owl:Restriction>  
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#lien" />  
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>  
</owl:Restriction>  
</rdfs:subClassOf>  
</owl:Class>
```

```
<owl:Class rdf:ID="réunion ">  
<rdfs:subClassOf rdf:resource="#Fonctionnalité " />  
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#descrip" />  
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>  
</owl:Restriction>  
</rdfs:subClassOf>  
</owl:Class>
```

```
<owl:Class rdf:ID="conference ">  
<rdfs:subClassOf rdf:resource="#Fonctionnalité " />  
<rdfs:subClassOf>  
<owl:Restriction>  
<owl:onProperty rdf:resource="#nom-c" />  
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>  
</owl:Restriction>  
</rdfs:subClassOf>
```

Annexe

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#date-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#dateposte-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nomauth-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#lienvidéo-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#lienimage-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#descptexte-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="Rdv ">
<rdfs:subClassOf rdf:resource="#Fonctionnalité " />
<rdfs:subClassOf>
```

Annexe

```
<owl:Restriction>
<owl:onProperty rdf:resource="#date-rd" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nometud-rd" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#prenometud-rd" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#cycleetud-rd" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#specetud-rd" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#heur-rd" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

Annexe

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#descptexte-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="Emploitemp">
<rdfs:subClassOf rdf:resource="#Fonctionnalité " />
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nom-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#date-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#heur-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#année-c" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#salle-c" />
```

Annexe

```
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="resultat" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#comporte"/>
<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#avoir "/>
<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#se caracterise"/>
<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-resu" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nom-resu" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="Recommandation" >
<rdfs:subClassOf>
<owl:Restriction>
```

Annexe

```
<owl:onProperty rdf:resource="#id-recom " />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="R_reunion">
<rdfs:subClassOf rdf:resource="#Recommandation " />
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#description" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#date" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#temp" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="R_etudiant">
<rdfs:subClassOf rdf:resource="#Recommandation " />
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#nom" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#prenom" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
```

Annexe

```
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
```

```
<owl:Restriction>
```

```
<owl:onProperty rdf:resource="#cycle" />
```

```
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
```

```
</owl:Restriction>
```

```
</rdfs:subClassOf>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="R_emploitemp">
```

```
<rdfs:subClassOf rdf:resource="#Recommandation " />
```

```
<rdfs:subClassOf>
```

```
<owl:Restriction>
```

```
<owl:onProperty rdf:resource="#cour" />
```

```
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
```

```
</owl:Restriction>
```

```
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
```

```
<owl:Restriction>
```

```
<owl:onProperty rdf:resource="#description" />
```

```
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
```

```
</owl:Restriction>
```

```
</rdfs:subClassOf>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="préférence-resultat" >
```

```
<rdfs:subClassOf>
```

```
<owl:Restriction>
```

```
<owl:onProperty rdf:resource="#id-presult" />
```

```
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
```

```
</owl:Restriction>
```

```
</rdfs:subClassOf>
```

```
</owl:Class>
```

```
<owl:Class rdf: ID="Affichage" >
```

```
<rdfs:subClassOf>
```

```
<owl:Restriction>
```

```
<owl:onProperty rdf:resource="#composé" />
```

```
<owl:minCardinality rdf:datatype="&xsd;int">1</owl:minCardinality>
```

```
</owl:Restriction>
```

```
</rdfs:subClassOf>
```

Annexe

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#détient" />
<owl:mincardinality rdf:datatype="&xsd:int">1</owl:mincardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-aff" />
<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#type-aff" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="caractéristique" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-car" />
<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf>
```

```
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#type-car" />
<owl:Cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

```
<owl:Class rdf:ID="préférence-affichage" >
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#id-paff" />
<owl:Cardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:Cardinality>
</owl:Restriction>
```

Annexe

```
</rdfs:subClassOf>
</owl:Class>
```

<!-- Propriétés d'objet -->

```
<owl:ObjectProperty rdf:ID="Contient">
<rdfs:domain rdf:resource="#Utilisateur " />
<rdfs:range rdf:resource="#Activité" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="Consulté">
<rdfs:domain rdf:resource="#Utilisateur " />
<rdfs:range rdf:resource="#Parametre" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="localise ">
<rdfs:domain rdf:resource="#Activité" />
<rdfs:range rdf:resource="#Localistion" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="possède">
<rdfs:domain rdf:resource="#Activité " />
<rdfs:range rdf:resource="#Fonctionnalité " />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="resulte">
<rdfs:domain rdf:resource="#Fonctionnalité" />
<rdfs:range rdf:resource="#Resultat" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="se caractérise">
<rdfs:domain rdf:resource="#Resultat" />
<rdfs:range rdf:resource="#Affichage" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="Comporte">
<rdfs:domain rdf:resource="#Resultat" />
<rdfs:range rdf:resource="#Préférence_resultat" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="composé ">
<rdfs:domain rdf:resource="#Affichage" />
<rdfs:range rdf:resource="#Préférence_affichage" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="avoir">
```

Annexe

```
<rdfs:domain rdf:resource="#Préférence_activité" />
<rdfs:range rdf:resource="#Activité" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="détient">
<rdfs:domain rdf:resource="#Caractéristique" />
<rdfs:range rdf:resource="#Affichage" />
</owl:ObjectProperty>
```

```
<owl:ObjectProperty rdf:ID="avoir">
<rdfs:domain rdf:resource="#Résultat " />
<rdfs:range rdf:resource="#Recommandation" />
</owl:ObjectProperty>
```

<!-- Propriétés de type de donnée -->

```
<owl:DatatypeProperty rdf:ID="id-u">
<rdfs:domain rdf:resource="#Utilisateur" />
<rdfs:range rdf:resource="&xsd;integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="nom-u">
<rdfs:domain rdf:resource="#Utilisateur" />
<rdfs:range rdf:resource="&xsd;string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="prenom-u">
<rdfs:domain rdf:resource="#Utilisateur" />
<rdfs:range rdf:resource="&xsd;string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="profession-u">
<rdfs:domain rdf:resource="#Utilisateur" />
<rdfs:range rdf:resource="&xsd;string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-activité">
<rdfs:domain rdf:resource="#Activité" />
<rdfs:range rdf:resource="&xsd;integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="type-activité ">
<rdfs:domain rdf:resource="#Activité />
```

Annexe

```
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-pactiv">
<rdfs:domain rdf:resource="#Préférence-activité " />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="longitude">
<rdfs:domain rdf:resource="#Localisation " />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="latitude">
<rdfs:domain rdf:resource="#Localisation " />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-font">
<rdfs:domain rdf:resource="#Fonctionnalité" />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-parm">
<rdfs:domain rdf:resource="#Parametre " />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="type">
<rdfs:domain rdf:resource="#L_plan " />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="descp">
<rdfs:domain rdf:resource="#L_plan " />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="lien">
<rdfs:domain rdf:resource="#L_plan " />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="descrip ">
<rdfs:domain rdf:resource="#Réunion " />
<rdfs:range rdf:resource="&xsd:integer" />
```

Annexe

```
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="nom-c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="date-c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="datepost-c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="nomauth -c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="lienvidéo -c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="lienimage-c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="descptexte-c ">  
<rdfs:domain rdf:resource="#Confrence " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="date-rd ">  
<rdfs:domain rdf:resource="#Rdv " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="nometud-rd ">
```

Annexe

```
<rdfs:domain rdf:resource="#Rdv" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="prenometud-rd">
<rdfs:domain rdf:resource="#Rdv" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="cycleetud-rd">
<rdfs:domain rdf:resource="#Rdv" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="specetud-rd">
<rdfs:domain rdf:resource="#Rdv" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="heur-rd">
<rdfs:domain rdf:resource="#Rdv" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-resu">
<rdfs:domain rdf:resource="#Résultat" />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="nom-resu">
<rdfs:domain rdf:resource="#Résultat" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-presut">
<rdfs:domain rdf:resource="#préférence-résultat" />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-aff">
<rdfs:domain rdf:resource="#Affichage" />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

Annexe

```
<owl:DatatypeProperty rdf:ID="type-aff">
<rdfs:domain rdf:resource="#Affichage" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-car">
<rdfs:domain rdf:resource="#caracteristique" />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="type-car">
<rdfs:domain rdf:resource="#caracteristique" />
<rdfs:range rdf:resource="&xsd:string" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-paff">
<rdfs:domain rdf:resource="#préférence affichage" />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="id-recom ">
<rdfs:domain rdf:resource="#Recommandation " />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="description ">
<rdfs:domain rdf:resource="#R_reunion " />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="date ">
<rdfs:domain rdf:resource="# R_reunion " />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="temp ">
<rdfs:domain rdf:resource="# R_reunion " />
<rdfs:range rdf:resource="&xsd:integer" />
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="nom ">
<rdfs:domain rdf:resource="#R_etudiant " />
```

Annexe

```
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="prenom ">  
<rdfs:domain rdf:resource="# R_etudiant " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="cycle ">  
<rdfs:domain rdf:resource="# R_etudiant " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="ncour ">  
<rdfs:domain rdf:resource="#R_emploitemp " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

```
<owl:DatatypeProperty rdf:ID="description ">  
<rdfs:domain rdf:resource="# R_emploitemp " />  
<rdfs:range rdf:resource="&xsd:string" />  
</owl:DatatypeProperty>
```

