

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique

Université Mohamed Khider – BISKRA

Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie

Département d'informatique

Laboratoire de L'Informatique Intelligente (Linfi)

N° Série :.....

THESE

Présentée pour obtenir le grade de

DOCTORAT 3ème CYCLE EN INFORMATIQUE

Par

Melle HOADJLI Abir

THEME

**Une approche Cloud Computing pour
le développement de 'Smart City' à
base de recommandations**

Soutenue le :

Devant le jury composé de :

LASKRI Mohamed Tayeb	Professeur à l'Université d'Annaba	Président
REZEG Khaled	Maitre de conférences à l'Université de Biskra	Rapporteur
KAZAR Okba	Professeur à l'Université de Biskra	Examineur
AMIRAT Abdelkrim	Professeur à l'Université de Souk Ahras	Examineur
TERRISSA Sadek Labib	Professeur à l'Université de Biskra	Examineur

2019/2020

Remerciements

Je tiens en tout premier lieu à exprimer ma profonde gratitude et remerciement

A Dieu qui m'a donné le pouvoir et la patience, pour mettre tous mes efforts

dans ce travail

A mon directeur de thèse Dr. REZEG Khaled pour la qualité de son

encadrement, sa patience, sa rigueur et sa disponibilité

Mes remerciements à Monsieur LASKRI Mohamed Tayeb, Professeur des universités à l'Université de ANNABA pour son acceptation de présider le jury.

Egalement à Monsieur AMIRAT Abelkrim Professeur des universités à

l'Université de SOUK AHRAS, et à Monsieur TERRISSA Sadek Labib

Professeur des universités à l'Université de BISKRA ; d'avoir accepté de juger

ce travail et de faire partie du jury.

Aussi mes sincères remerciements à Monsieur KAZAR Okba Professeur des

universités à l'Université de BISKRA ; pour son accompagnement et son

accueil au sein du laboratoire LINFI ; et son acceptation de juger ce travail.

A l'enseignante HOADJLI HADIA pour ses grands efforts malgré ses

nombreuses préoccupations

Au professeur KAHLOUL Laid pour ses conseils et son encouragement

Et à tous ceux qui ont participé de près ou de loin

A la réussite de ce travail

Merci.

Dédicaces

*C'est avec profonde gratitude et sincères mots que je dédie ce travail
premièrement et avant tout à mes parents ma mère et mon père qui m'ont
supporté et soutenu pendant toutes mes années d'études,*

*À ma sœur LAMIA, ma sœur HADIA et son mari YUCEF pour tout ce qu'ils
ont fait pour m'aider au long de cette thèse.*

A ma précieuse famille.

Résumé

Avec l'innovation des nouvelles technologies de l'information et de la communication (TIC) et les besoins d'échange d'informations et de partage des connaissances au sein de la ville, un système de ville intelligente vise à améliorer la qualité de vie de ses citoyens en offrant un ensemble de services publics. Dans ce contexte, l'objectif de ce travail de recherche est de proposer une approche de ville intelligente pour l'utilisation de ces technologies dans la prestation de services publics afin de permettre une plus grande interaction en temps réel avec les citoyens; nous proposons un framework de prestation de services basé sur une approche de recommandation contextuelle afin d'améliorer les services numériques de la ville intelligente en fonction du contexte du citoyen et de l'approche BAAS (Back-end As A Service) sur le cloud afin d'améliorer la scalabilité du système dans des charges de travail importantes. Dans ce cadre, nous avons évalué la scalabilité et l'interactivité de l'approche proposée en mesurant les métriques de temps de réponse et données échangées. Pour estimer la scalabilité, nous avons effectué une série de tests de charge où les valeurs de temps de réponse augmentaient avec l'ajout de demandes dans les différents cas mais restaient acceptables. En termes d'interactivité, la quantité de données échangées entre l'application citoyenne et le back-end de la ville a été mesurée par un maximum de 117 Ko/s. Les expérimentations menées ont révélé que l'approche de recommandation contextuelle optimisait les interactions en réduisant la quantité de données échangées, et l'approche BAAS améliorait très bien la scalabilité du système et permettait le traitement des demandes simultanées. Dans les travaux futurs, nous espérons renforcer notre système de recommandation en incluant l'aspect social des citoyens et bénéficier des technologies IOT comme sources de données supplémentaires.

Mots clés: Backend cloud, citoyens, recommandations contextuelles, smart city, services publics, scalabilité, TIC.

Abstract

With the innovation of the new Information and Communication Technologies (ICT) and the needs of information exchange and knowledge sharing among the city, a smart city system aims to improve the life quality of its citizens by offering a set of public services. Within this context the aim of this research work, is to propose a smart city approach for using these technologies in the delivery of public services to allow greater real-time interactions with citizens. In our work, we propose a service providing framework based on context-aware recommendation approach in order to improve the smart city digital services according to the citizen's context and the Backend As A Service (BAAS) approach on cloud to enhance the scalability of the system in large workloads. In this study, we evaluated the scalability and interactivity of the proposed approach by measuring the response time and exchanged data metrics. To estimate the scalability, we performed series of load tests where the response time values increased with the addition of requests in the different cases but remained acceptable. In terms of interactivity, the amount of exchanged data between the citizen application and the city's backend was measured by a maximum of 117Kb/s. The conducted experiments revealed that the context-aware recommendation approach optimized the interactions by reducing the amount of exchanged data, and the BAAS approach improved very well the scalability of the system and allowed the handling of concurrent requests. In the future work, we hope to strengthen our recommender system by including the social aspect of citizens and benefit from IOT technologies as additional data sources.

Keywords: citizens, cloud backend, context-aware recommendations, ICT, public services, smart city, scalability.

ملخص

مع ابتكار تقنيات المعلومات والاتصالات الجديدة واحتياجات تبادل المعلومات و المعرفة في المدينة ، يهدف نظام المدينة الذكية إلى تحسين نوعية حياة مواطنيها من خلال تقديم مجموعة من الخدمات العامة. في هذا السياق ، الهدف من هذا العمل البحثي ، هو اقتراح منهجية للمدينة الذكية لاستخدام هذه التقنيات في تقديم الخدمات العامة والسماح بتفاعلات أكبر في الوقت الحقيقي مع المواطنين ؛ من أجل هذا نقترح إطار عمل لتقديم الخدمات يعتمد على منهجية التوصية المدركة للحالة من أجل تحسين الخدمات الرقمية للمدن الذكية وفقاً لحالة المواطن ومنهجية النظام الخلفي كخدمة (BAAS) على الأنظمة السحابية لتعزيز قابلية النظام للتوسع في أعباء العمل الكبيرة. في هذه الدراسة ، قمنا بتقييم قابلية تفاعل وتوسع المنهجية المقترحة من خلال قياس زمن الاستجابة و مقياس البيانات المتبادلة. لتقدير قابلية التوسع ، أجرينا سلسلة من الاختبارات حيث تزايدت قيم وقت الاستجابة مع إضافة عدد الطلبات في الحالات المختلفة ولكنها ضلت مقبولة. من حيث التفاعل ، تم قياس كمية البيانات المتبادلة بين تطبيق المواطن والنظام الخلفي للمدينة و التي حددت بحد أقصى 117 كيلو بايت / ثانية. كشفت التجارب التي أجريت أن منهجية التوصية المدركة للحالة أدت إلى تحسين التفاعلات عن طريق تقليل كمية البيانات المتبادلة ، كما أن منهجية الBAAS قد حسنت بشكل كبير قابلية النظام للتوسع وسمحت بمعالجة الطلبات المتزامنة. في العمل المستقبلي ، نأمل في تعزيز نظام التوصية الخاص بنا من خلال تضمين الجانب الاجتماعي للمواطنين والاستفادة من تقنيات IOT كمصادر إضافية للبيانات.

الكلمات المفتاحية: تكنولوجيا المعلومات والاتصالات ، المدينة الذكية ، المواطنون ، الخدمات العامة ، التوصيات المدركة للحالة ، النظام الخلفي السحابي ، قابلية التوسع.

Sommaire

Introduction générale	1
-----------------------------	---

CHAPITRE 01 : Smart city : contexte et applications

I.	Introduction	6
II.	Concepts et notions de base d'une smart city	6
1.	Définition de smart city	6
2.	Rapport entre la ville digitale, la ville ubiquiteuse et la ville intelligente	7
3.	Rôle des nouvelles technologies TICs	7
4.	Dimensions d'une smart city	8
5.	Caractéristiques des smart city	9
6.	Besoins des smart cities	10
6.1.	Besoins techniques	11
7.	Défis des villes intelligentes	13
7.1.	Défis sociaux	14
8.	Cloud computing et Smart cities	15
9.	Intelligence territoriale et smart cities	16
10.	Synthèse	16
III.	Les Domaines d'Applications des Smart Cities	17
1.	Mobilité et gestion de transport	17
2.	Gestion intelligente de l'eau	18
3.	Tourisme intelligent	20
4.	Développement du e-gouvernement	20
4.1.	Modèles de gouvernance intelligente	20
4.2.	Synthèse des modèles de gouvernance	21
4.3.	Concept de "smart administration" dans le contexte de « smart city »	24
4.4.	Types de gouvernance	24
IV.	Conclusion	26

CHAPITRE 02 : Cloud Computing et systèmes de recommandations

1.	Introduction	27
2.	Définition du cloud computing	27
3.	Architecture du cloud computing	27
4.	Services du cloud computing	29
4.1.	IAAS	29
4.2.	PAAS	30
4.3.	SAAS	30
5.	Types de cloud computing	30
6.	Les technologies du cloud computing	30
6.1.	La virtualisation	31

6.2.	Le Datacenter	31
6.3.	La Plateforme collaborative	31
7.	Quelques Acteurs du cloud	32
7.1.	Amazon	32
7.2.	Microsoft	33
7.3.	Google	33
8.	Des solutions cloud opensources	33
8.1.	Eucalyptus	33
8.2.	OpenStack	34
8.3.	Synthèse des solutions de cloud opensource	35
9.	Cloud computing et sensibilité au contexte (context-awareness)	35
10.	Systèmes de recommandation	36
10.1.	Définition formelle d'un système de recommandation	37
10.2.	Recommandation basée sur le contenu	38
10.3.	Recommandation basée sur la collaboration	39
10.4.	Système de recommandation sensible au contexte	39
10.5.	Le problème de scalabilité dans les systèmes de recommandation	41
11.	Synthèse	41
12.	La scalabilité et ses solutions	43
12.1.	Définition d'un système scalable	43
12.2.	Stratégies de scalabilité	46
12.2.1.	Scalabilité verticale	46
12.2.2.	Scalabilité horizontale	47
12.3.	L'approche backend as a service	48
12.3.1.	Comparaison de l'approche BAAS par rapport aux services du cloud computing	50
13.	Conclusion	51

CHAPITRE 03 : Travaux relatifs

1.	Introduction	52
2.	Projets de smart city existants	52
2.1.	IBM smarter city	52
2.1.1.	IBM intelligent water	53
2.1.2.	IBM intelligent transportation v1.0	56
2.1.3.	Solutions d'IBM smarter city sur cloud (SaaS)	60
2.2.	Seoul Smart City	61
2.2.1.	U-Seoul Net	61
2.2.2.	Open Governance 2.0	62
2.2.3.	U-Seoul Safety Service	62
2.3.	Synthèse des projets de smart city	62
3.	Les Travaux de Recherche	63
3.1.	Recommandations proactives et réactives	63
3.1.1.	Moteur de recommandation réactif	64
3.1.2.	Moteur de recommandation proactif	65

3.1.3.	Synthèse	65
3.2.	ASSO the Asistant to Citizens for Service Selection	66
3.2.1.	Profil du citoyen	66
3.2.2.	Composition d'un paquet de services	67
3.2.3.	Evaluation et classement des fournisseurs de services	68
3.2.4.	Synthèse	68
4.	Conclusion	69

CHAPITRE 04 : Modélisation de l'approche proposée

1.	Introduction	70
2.	Fonctionnement et objectif général	70
3.	Formalisation du problème	71
4.	La modélisation du citoyen	71
4.1.	L'Acquisition des données	72
4.2.	Construction du profil citoyen	73
4.2.1.	Dimension démographique	74
4.2.2.	Dimension contextuelle	75
4.2.3.	Dimension de préférences	76
4.3.	Evolution du profil citoyen	77
5.	Modélisation des services et requête du citoyen	78
5.1.	Requête du citoyen	78
5.2.	Modélisation des services	79
6.	L'approche de recommandation	80
6.1.	Les sources de données	81
6.2.	Modélisation et évaluation des préférences	81
6.3.	La fonction de rating	82
6.4.	Utilisation des informations contextuelles par l'algorithme de post-filtrage	83
6.4.1.	Algorithme de filtrage	84
6.4.2.	Algorithme de recommandation	86
7.	Le modèle conceptuel	87
7.1.	Le système backend	88
7.2.	Le système frontend	89
8.	Fonctionnement de BAAS	89
9.	Conception détaillée	90
9.1.	La partie back-end	90
9.1.1.	La couche de gestion de services	90
9.1.2.	La couche Recommandation	93
9.1.3.	La couche de gestion des données	95
9.2.	La partie front-end	96
10.	Fonctionnement du système	96
10.1.	Diagramme de classes	96
10.2.	Diagramme de cas d'utilisation	97
10.3.	Diagrammes de séquences	98
10.4.	Conclusion	100

CHAPITRE 05 : Validation du modèle proposé

1.	Introduction	101
2.	Etude de cas	101
2.1.	L'Infrastructure technologique des TICs en Algérie	101
2.2.	Synthèse	105
2.3.	Démarches administratives	106
2.4.	Les institutions administratives de Biskra	107
3.	Outils de développement	108
3.1.	Android Studio	108
3.2.	Le langage JAVA	109
3.3.	Les APIs publiques	109
3.4.	Google Cloud Endpoints	110
4.	Implémentation	112
4.1.	Le système back-end	112
4.2.	Le système front-end	116
5.	Évaluation des résultats et des performances	123
5.1.	Estimation de la scalabilité	123
5.2.	Évaluation de l'interactivité	126
6.	Conclusion	128
	Conclusion générale	129
	Références bibliographiques	132
	Liste des publications	138

Liste des figures

Figure 3.1. Modèle conceptuel de cloud computing	28
Figure 3.2. Services de cloud computing	29
Figure 3.3. Principe de fonctionnement d'eucalyptus	34
Figure 4.1. Un système scalable idéalisé	43
Figure 4.2. Réponse versus demande	44
Figure 4.3. Echec de scalabilité	45
Figure 4.4. Métrique de réponse par rapport à une demande pour une ressource	45
Figure 4.5. Exemple de scalabilité verticale	46
Figure 4.6. Exemple de scalabilité horizontale	47
Figure 5.1. Solutions d'IBM smarter city	53
Figure 5.2. Les composants d'IBM intelligent water	54
Figure 5.3. Système de gestion de trafic avec IBM intelligent transportation	57
Figure 5.4. Fonctionnement de l'approche proactive et réactive	64
Figure 5.5. Composition d'un paquet de services	68
Figure 6.1. Schéma décrivant les dimensions d'un profile citoyen	74
Figure 6.2. Raffinement de la dimension démographique	75
Figure 6.3. Raffinement de la dimension contextuelle	75
Figure 6.4. Raffinement de la dimension des préférences	77
Figure 6.5. Modèle général de l'entité service	79
Figure 6.6. Modèle d'un exemple d'évènement social	79
Figure 6.7. Les étapes de l'approche de recommandation	80
Figure 6.8. Paradigme de post-filtrage contextuel	83
Figure 6.9. Le processus de post-filtrage contextuel	84
Figure 6.10. Cadre conceptuel global du système proposé de ville intelligente	88
Figure 6.11. Le fonctionnement de l'approche BAAS	90
Figure 6.12. Modèle conceptuel du sous système de filtrage	94
Figure 6.13. Modèle conceptuel du sous système d'adaptation	95
Figure 6.14. Modèle conceptuel du système front-end	96
Figure 6.15. Diagramme de classes	97
Figure 6.16. Diagramme de cas d'utilisations	98
Figure 6.17. Diagramme de séquence du filtrage	98
Figure 6.18. Diagramme de séquence d'adapatation	99

Figure 6.19. Diagramme de séquence des gestionnaires d'informations	100
Figure 7.1. Schéma représentant l'évolution de la pénétration d'internet en Algérie	102
Figure 7.2. Schéma représentant l'évolution de la pénétration de l'utilisation mobile en Algérie	103
Figure 7.3. Capture de la page des démarches administrative sur le site web du gouvernement d'intérieur de l'Algérie	107
Figure 7.4. Schéma décrivant la solution Google Cloud Endpoints	111
Figure 7.5. Le data-store du système back-end	112
Figure 7.6. Les services REST API disponibles du système back-end	113
Figure 7.7. Capture d'écran de l'exécution de la méthode	115
Figure 7.8. Capture d'écran du résultat de l'exécution	116
Figure 7.9. Capture d'écran de l'activité d'accueil	117
Figure 7.10. Capture d'écran de l'activité des événements sociaux	118
Figure 7.11. Capture d'écran de l'activité de la carte et de recherche pendant un événement de changement de position	119
Figure 7.12. Capture d'écran d'un changement de localisation	120
Figure 7.13. Capture d'écran des informations sur l'institution administrative et ses services en ligne	122
Figure 7.14. Capture d'écran d'un service en ligne	123
Figure 7.15. Scalabilité du système lors du premier test de charge	124
Figure 7.16. Scalabilité du système lors du deuxième test de charge	124
Figure 7.17. Scalabilité du système lors du troisième test de charge	125
Figure 7.18. Répartition des réponses dans le test initial (en haut à gauche), le deuxième test (en haut à droite) et le test final (en bas)	125
Figure 7.19. Historique des requêtes d'appels des services API	127
Figure 7.20. Captures d'écran des statistiques du réseau du système dans les activités de visualisation de la carte (à droite) et de recherche (à gauche)	128

Liste des tableaux

Tableau 1.1. Tableau récapitulatif des besoins techniques d'une smart city	13
Tableau 6.1. Exemple d'évaluation d'une institution administrative par trois citoyens	83

INTRODUCTION GENERALE

1. Contexte de recherche

Le concept de smart city est apparu comme une approche stratégique reliant les différents aspects d'une ville en un système unique, à travers des initiatives dans de nombreux domaines urbains (transports, énergie, gestion de l'eau, gouvernance...) afin de gérer et administrer plus efficacement la ville et réaliser son développement et d'accroître sa compétitivité et améliorer la qualité de vie, ces initiatives sont souvent des idées simples et toujours ingénieuses basées sur les opportunités offertes par les nouvelles technologies de l'information et de la communication (TIC).

Le cadre de ce travail concerne la proposition d'une approche pour le développement d'une smart city, plus particulièrement le domaine d'administration intelligente qui a comme noyau le partage d'informations et services publics au sein de la ville.

Le domaine de smart city est particulièrement touché par le problème de la surcharge d'information car il renferme un volume d'information assez conséquent. Le secteur d'administration est aussi confronté à cette problématique. De ce fait, les citoyens sont confrontés à plusieurs problèmes : ils sont submergés par le nombre très important de choix possibles dans l'espace qu'ils explorent. L'exploitation de cette longue liste de services est très complexe pour les citoyens, qui doivent passer beaucoup de temps pour sélectionner les options qui correspondent le plus à leurs intérêts. De plus, les citoyens ne savent pas forcément ce qu'ils devraient voir ou ce qu'ils pourraient apprécier, le parcours qu'ils font n'est alors en général pas réfléchi, ou bien ils se limitent à voir les services les plus connus.

En conséquence, ils peuvent perdre du temps en utilisant des services qui ne les intéressent pas. Inversement, ils peuvent manquer d'autres qui auraient pu être plus utiles pour eux.

Un des domaines de recherche principaux relatifs à la problématique de la surcharge d'information est le domaine de la recherche d'information. Le principe général est d'élaborer des méthodes et des algorithmes afin de rechercher des ressources (par exemple, des pages web, des films et dans notre cadre d'application des services digitales) en fonction de requêtes formulées par des utilisateurs. Il n'est cependant pas toujours évident pour un utilisateur de savoir comment exprimer sa demande. De plus, sa requête correspond généralement à une quantité importante de ressources et il est difficile de savoir quels résultats lui présenter en premier, d'autant plus que d'un utilisateur à un autre, l'ordre de priorité peut changer.

Un autre domaine de recherche relatif à cette problématique est le domaine des systèmes de recommandation. Ces systèmes sont capables de fournir des recommandations adaptées aux préférences et aux besoins des utilisateurs. Ils se sont avérés être très satisfaisants pour aider les utilisateurs à accéder aux ressources désirées dans un temps limité. Initialement conçus pour la recommandation de ressources web, films, etc. les systèmes de recommandation sont devenus de plus en plus populaires et sont aujourd'hui un composant principal de beaucoup d'applications dans différents domaines. Un avantage très conséquent des systèmes de recommandation est que l'utilisateur n'a pas besoin de formuler de requêtes. Sa seule requête est implicite, elle peut se traduire par : "Quelles sont les ressources qui correspondent à mes préférences, mes besoins et mes contraintes ? ". Les systèmes de recommandation peuvent être classés en plusieurs types d'approches : les approches basées sur le contenu et les approches basées sur le filtrage collaboratif. Les recommandations basées sur le contenu sont effectuées en identifiant les ressources similaires à celles appréciées par un utilisateur en fonction de leur contenu [1].

Les approches basées sur le filtrage collaboratif, quant à elles, permettent de fournir des recommandations à un utilisateur sans forcément considérer le contenu des ressources, mais en se basant sur l'analyse du comportement et/ou des appréciations de l'utilisateur afin de recommander les ressources qui ont été appréciées par d'autres utilisateurs ayant des goûts similaires [1].

Notre travail de recherche concerne les systèmes de recommandation comme approche de gestion de smart city dans le but d'offrir aux citoyens des services administratifs personnalisés.

2. Problématique

Pour proposer un système de recommandation dans le cadre de l'administration intelligente d'une smart city, il faut faire face à de nombreux problèmes. Certains concernent les systèmes de recommandations tels que la scalabilité. D'autres problématiques sont plus spécifiques à notre domaine d'application. En effet, pour l'administration dans une smart city d'autres problématiques s'ajoutent. D'autres sont relative au citoyen, la problématique se résume à mettre le citoyen dans une situation acceptable durant la réalisation de ces tâches administratives en évitant les différents problèmes qui peuvent lui rencontrés. Parmi ces problématiques nous citons :

- C'est dans ce contexte que les systèmes de recommandation sont confrontés à de nouvelles opportunités et défis. La charge accablante d'informations et de services dans les applications de gouvernance électronique permet le développement et l'utilisation de solutions de recommandation particulières pour différentes tâches et parties prenantes.
- La Smart City rencontre des problèmes fondamentaux d'échange d'informations et de partage de connaissances entre ses entités formelles réparties dans le pays et les citoyens de la ville. Au même temps, l'objectif principal d'une ville intelligente reste avant tout d'améliorer la qualité de vie des citoyens en proposant un ensemble de services publics. À cette fin, la gouvernance de la ville intelligente a été définie comme l'application des TIC dans les interactions avec les citoyens par la prestation de services, l'échange d'informations, la communication et les transactions. L'une des nombreuses solutions de gouvernance consiste à fournir une large gamme de services électroniques aux citoyens du gouvernement. En réaction à cette solution, le besoin de services de gouvernance électronique davantage centré sur les citoyens devient plus important. Par conséquent, des progrès doivent être accomplis dans le développement d'approches personnalisées qui permettent non seulement de fournir des services électroniques plus efficaces, mais aussi d'accroître la satisfaction et l'engagement des citoyens.
- Nous considérons la situation où un citoyen recherche les démarches administratives qu'il doit effectuer afin de réaliser une tâche administrative précise exprimées en fonction d'un ensemble de services. Un citoyen ne sait pas forcément quels services vont être utiles pour lui afin qu'il puisse réaliser la tâche administrative qu'il veut effectuer. Son processus passe alors par trois étapes : Premièrement, il commence par décider et choisir les démarches administratives qu'il va effectuer. Ensuite, il va découvrir les services qu'il préfère utiliser. Finalement, il commence à utiliser les services et affiner ses préférences, par conséquent son profil sera plus riche et ses préférences seront plus précises. Une problématique est alors de concevoir un système de recommandation pour la gestion de smart city qui prend en compte ces trois étapes et de définir quelle approche est adaptée à quelle étape.
- Aussi, il est important de pouvoir guider le citoyen, c'est-à-dire lui proposer une liste de démarches administratives à contempler successivement dans un ordre défini. Aussi, le citoyen dispose généralement d'un temps limité pour effectuer ces tâches, il est alors nécessaire de pouvoir recommander un parcours, qui aide le citoyen à se déplacer de manière efficace, ce qui lui ferait perdre du temps et réduirait sa satisfaction. C'est pour cela que la prise en compte des informations contextuelles est alors nécessaire dans notre cadre de travail.

3. Objectif

Notre objectif consiste à concevoir des systèmes, sur dispositif mobile, afin d'aider à améliorer les services administratives d'une smart city par rapport aux besoins des citoyens. Dans ce cas, il s'agit d'adapter et de personnaliser la liste des services administratifs proposés au citoyen, en fonction de sa requête, ses préférences, son contexte et ses contraintes. Notre travail s'articule principalement sur les systèmes de recommandation qui se sont avérés être très efficaces pour aider les citoyens à accéder aux ressources par lesquels il serait potentiellement intéressés.

4. Contribution

Dans ce travail de recherche, nous nous référons à l'approche de recommandation contextuelle afin d'améliorer la qualité des services offerts aux citoyens en les adaptant en fonction de leur contexte. Ce type de systèmes souffre d'un problème de scalabilité, en particulier dans le contexte de la ville intelligente où la scalabilité est très nécessaire, afin de surmonter ce problème, nous nous référons à l'approche back-end as a service sur le cloud qui permet une mise à l'échelle automatique du système.

Notre contribution consiste à proposer une solution scalable capable de traiter un grand taux de données en combinant l'approche de recommandation contextuelle et l'approche de back-end as a service afin d'améliorer le processus de partage d'informations publiques et de prestation de services dans le contexte de l'administration de la ville intelligente [2] [3].

5. Organisation de la thèse

Ce travail de recherche est réparti sur cinq chapitres différents dont l'organisation est comme suit :

Une introduction générale situant le contexte et explicitant la problématique et objectifs à atteindre ainsi que les résultats obtenus et l'organisation du mémoire.

Chapitre 1 : fait l'objet de la présentation des concepts et Notions de Base d'une ville intelligente en plus des applications des smart cities qui regroupent les logiciels (tels que des applications) qui permettent à un groupe d'acteurs (ville, communauté, citoyens) de faire face conjointement aux défis de la ville de manière intelligente et plus efficaces.

Chapitre 2 : Dans ce chapitre nous avons fait un survol sur d'une part le Cloud Computing qui est défini comme un paradigme de calcul distribué émergent dans lequel les données et les services sont disponibles dans des data centers extensibles et peuvent être accédés de

manière transparente depuis des appareils. D'autre part les systèmes de recommandations dont leurs objectif est de fournir une recommandation, prédiction, opinion, ou une liste configurée d'articles qui aide l'utilisateur à les évaluer. Nous avons par la suite présenté la scalabilité qui représente la capacité de changer, à la demande et de manière transparente pour l'utilisateur, l'ordre de grandeur (accroître ou réduire les capacités) d'un équipement, d'un logiciel ou d'un processus informatique, ainsi que quelques solutions de la scalabilité.

Chapitre 3 : Fait l'objet de l'étude de non seulement des travaux de recherche dans ce domaine mais aussi des exemples du monde réel. Plus précisément dans les systèmes de recommandations afin de bénéficier et exploiter les avantages de leurs approches.

Chapitre 4 : représente notre contribution qui se résume à la modélisation d'une approche cloud computing pour le développement de 'smart city' à base de recommandations.

Chapitre 5 : fait l'objet de la validation du modèle proposé. Afin de bien illustrer notre approche de système d'administration de smart city nous avons choisi comme cas d'étude l'administration de la ville de Biskra.

Nous avons conclu notre thèse par une **conclusion générale** durant laquelle nous avons présenté le bilan du travail réalisé ainsi que les perspectives envisagé.

CHAPITRE 01 :

Smart city : contexte et applications

I. Introduction

A la convergence des projets commerciaux et des recherches académiques, le concept de « smart cities » s'est imposé dans les années 2000. Parce que Les villes sont des systèmes complexes couvrant une multitude de domaines différents, le concept initial d'une ville intelligente a commencé avec une simple utilisation des technologies de communication et a évolué jusqu'à l'utilisation de technologies de l'information pour être plus intelligent et plus efficace.

La smart city est apparue comme une nouvelle démarche permettant l'utilisation prévue et sage des ressources humaines et naturelles au sein d'une ville dans le but d'offrir une meilleure qualité de vie aux citoyens.

II. Concepts et notions de base d'une smart city

Afin de bien Introduire le domaine des villes intelligentes, dans ce chapitre nous allons citer les différents concept et notions de base relatives au smart city ainsi que ses besoins, caractéristiques et défis.

1. Définition de smart city

« Une ville devient intelligente lorsque les investissements humaines et sociales et les infrastructures traditionnelles (transport) et modernes (TIC) favorisent une croissance économique durable et une qualité de vie élevée, grâce à une gestion efficace des ressources naturelles et d'une gouvernance participative ». [4]

Le concept de smart city ou bien ville intelligente, est apparu à la fin du XXe siècle, Il n'y a pas encore une interprétation unique de la ville intelligente mais il y a un certain nombre d'interprétations [5].

Selon le point de vue conceptuel, une ville intelligente est l'approche stratégique reliant les différents aspects du développement de la ville à un seul système, avec l'utilisation de l'intelligence artificielle, les TIC et le potentiel écologique et sociale de développer la ville et d'accroître sa compétitivité [5].

Selon l'approche de l'infrastructure, une ville intelligente est la combinaison de réseaux intelligents; c'est la nouvelle qualité de vie avec l'utilisation des technologies innovantes, exigeants l'utilisation efficace économiquement et écologiquement des systèmes urbains.

Cependant, dans le XXI^{ème} siècle l'environnement urbain est déterminé non seulement par la disponibilité de l'infrastructure traditionnelle (ressources matérielles), mais aussi des décisions intelligentes fournissant le développement de percée. [5]

2. Rapport entre la ville digitale, la ville ubiquiteuse et la ville intelligente

Cette notion de ville intelligente peut être considérée comme une extension et une évolution d'autres concepts de la ville, tel que la «ville digitale» et la «ville ubiquiteuse».

La ville numérique, la plus ancienne de ces concepts, a été créée pour faire référence à toute initiative numérique entreprise par une ville, à commencer par la fourniture d'un accès à Internet (dans ce cas, nous faisons également référence à la «ville câblée») jusqu'à la représentation 3D. de la ville (la soi-disant "cyber-ville"). Au sens le plus général, la ville numérique est identifiée comme un système d'information qui collecte des informations numériques sur la ville réelle et les met à disposition dans un espace public virtuel, où les citoyens peuvent la consulter, mais aussi interagir avec le système et avec d'autres utilisateurs (d'où le terme souvent utilisé de «ville intelligente») [6].

La ville ubiquiteuse (également appelée «U-city») développe plus avant l'idée de la ville numérique, en créant une nouvelle génération d'espace urbain, qui résulte de la convergence entre le monde physique et la réalité virtuelle. La U-city est définie comme un modèle innovant visant à améliorer la gestion de la ville, la qualité de la vie et le développement économique en identifiant le facteur de succès déterminant de l'attention portée à l'utilisateur final. Néanmoins, il existe des projets qui ne concernent que certaines catégories de citoyens (par exemple les jeunes), ce qui atténue la nature centrée sur l'utilisateur de la ville ubiquiteuse [6].

3. Rôle des nouvelles technologies TICs

En plus du courant de pensée qui se concentre sur les TICs, il convient également de mentionner que les villes intelligentes sont des villes qui développent leurs propres gouvernance et conditions de développement socio-économique. Ce sens, même s'il ne renonce pas le soutien dérivant de l'utilisation large et innovante des TIC, il est centré sur la satisfaction des besoins des citoyens, des entreprises et des autres organisations. De ce point de vue, une ville intelligente, en surveillant et en intégrant ses infrastructures critiques, qu'il s'agisse de l'infrastructure physique (routes, ponts, etc.), de l'infrastructure technologique (matériel, logiciel, réseau) ou de son potentiel intellectuel et social (résultant les relations

entre les membres de la communauté), planifie les activités de prévention, de maintenance et de gestion, utilise efficacement ses ressources et optimise l'efficacité de ses services [6].

Dans ces conditions, une ville intelligente constitue un contexte urbain à la fois innovant, compétitif, efficace, efficient, sûr, habitable, équitable et durable.

La principale différence entre cette signification de ville intelligente et la précédente réside dans le rôle attribué aux TIC. Dans le premier cas, les TIC sont un élément indispensable autour duquel tout les éléments de la ville tourne; dans le second cas, il ne s'agit que de l'un des piliers du modèle, dont il représente un facteur habitant important, mais pas nécessairement le seul et, parfois, même pas le plus important.

Le sens non centré sur les TIC se caractérise par une vision plus large, plus flexible et plus ouverte. Une vision qui semble plus cohérente et convaincante, certainement plus cohérente avec l'objectif de création de valeur publique locale. Comme pour l'approche centrée sur les TIC, même cette signification peut être reliée à d'autres concepts de la ville récemment développés, et en particulier à celui de «ville du savoir» [6].

4. Dimensions d'une smart city

A l'intention de clarifier ce qui constitue une ville intelligente, nombreux chercheurs ont séparé ce concept dans de nombreuses fonctionnalités et dimensions, justifiant cette décision à la complexité de la gestion d'une ville intelligente d'une manière globale. Pour cela six composantes ont été identifiées (Giffinger and Gudrun, 2010) [7] :

- **Smart Economy** : l'économie intelligente qui comprend tous les facteurs de la compétitivité économique de l'innovation, l'esprit d'entreprise, marques de commerce, la productivité et la flexibilité du marché du travail ainsi que l'intégration dans le marché national.
- **Smart People**: est non seulement décrite par le niveau de qualification ou de l'éducation des citoyens, mais aussi par la qualité des interactions sociales en matière d'intégration et de la vie publique et l'ouverture vers le monde «extérieur».
- **Smart Governance**: comprend les aspects de la participation politique, les services pour les citoyens ainsi que le fonctionnement de l'administration.
- **Smart Mobility** : la disponibilité des technologies de l'information et de la communication et des systèmes de transport modernes et durables.

- Smart Environment : est décrite par des conditions naturelles attractives (climat, espaces verts, etc.), la pollution, la gestion des ressources et aussi par les efforts visant à la protection de l'environnement.
- Smart Living : comprend divers aspects de la qualité de vie que la culture, la santé, la sécurité, le logement, le tourisme, etc. [7]

5. Caractéristiques des smart city:

Afin de décrire une ville intelligente, un certain nombre de caractéristiques clés sont nécessaires [8]:

- La ville doit être instrumentée pour permettre la collecte des quantités croissantes de données sur la vie de ville.
- Les données provenant des différentes sources et des systèmes de la ville seront disponibles pour être facilement agrégées ensemble pour gagner une meilleure perception sur ce qui se passe dans la ville.
- Les données doivent être présentées dans une variété de formats, en fonction du contexte et de la personne ou système technique dont il a besoin, permettant une meilleure visualisation et accès facile, pour être plus utile.
- Une connaissance en temps réel, détaillée et mesurable sur la ville va donc être disponibles à tous les niveaux de sorte qu'elle soit facilement accessible, pour que n'importe quel personne ou système technique serait capable de l'utiliser pour aider à remplir leur rôle ou de réaliser leurs objectifs, dans le contexte du fonctionnement global de la ville.
- En outre, des analyses et des systèmes de prise de décision seront utilisés, de sorte que cette connaissance peut être utilisée de manière efficace, à la fois par les gestionnaires, les planificateurs, et les citoyens de la ville pour soutenir en temps réel la prise de décision.
- La ville sera également automatisée, pour permettre aux fonctions de la ville d'être livrés de façon fiable, et efficacement, sans avoir besoin d'intervention humaine directe.
- La ville aura un réseau d'espaces collaboratifs, pour permettre aux communautés dynamiques de stimuler l'innovation, la croissance et d'améliorer le bien-être des citoyens.
- La présence d'une interaction continue entre les mondes physiques et numériques permettant au processus de prise de décision d'être beaucoup plus ouvert et inclusive, pour que les

citoyens, les décideurs politiques et les entreprises peuvent travailler ensemble efficacement pour gérer la vie de la ville.

Nous constatant dans tous ces caractéristiques que les TIC ont un rôle clé dans chacun d'eux.

6. Besoins des smart cities

Dans un domaine aussi complexe et multi-disciplinaire que les "Smart Cities", il y aura de nombreux défis pour la normalisation.

Nous constatons que le rôle central de la technologie est l'une des caractéristiques déterminantes d'une Smart City.

En effet, le concept de « l'intelligence » est abordé en termes de performance pertinente pour des solutions technologiquement réalisables, en conformité avec le développement durable et la résilience des communautés [9].

Accomplir les besoins d'une ville normale peut ne pas exiger de la «technologie» du tout: si "des rues plus sûres", est compris dans un contexte particulier en le sens de "réduction des crimes", il se peut que l'amélioration générale des conditions socio-économiques réduise la probabilité d'un tel crime sans autre solutions.

Il y aura de nombreuses exigences de normalisation pour les villes intelligentes qui vont au-delà des TIC ou n'utilisent pas une technologie du tout.

Cependant, la prévalence accrue des TIC dans de nombreux domaines de la vie quotidienne signifie que même dans des domaines qui ne sont pas spécifiquement centrée sur les TIC, il ya des aspects qui sont néanmoins TIC influencé.

Cette prévalence des TIC est ce qui peut aider à transformer une ville dans une ville intelligente.

Une solution socio-économique des crimes, par exemple, semblent ne pas impliquer des TIC, mais pourrait bénéficier d'analyses détaillées des données provenant de multiples sources qui, ensemble, peuvent fournir aux décideurs des informations pertinentes suffisantes pour prendre des décisions qui ont l'impact socio-économique souhaité. Ces analyses sont fortement dépendantes des TICs [9].

6.1. Besoins techniques

Les villes intelligentes sont caractérisées par la disponibilité d'une large gamme de technologies capables de travailler ensemble pour fournir des systèmes et des solutions complexes. Elles ont besoin de technologies robustes et résistantes pour aider à [10]:

- Fournir une compréhension commune des concepts de base qui sous-tendent les villes intelligentes et qui peuvent être utilisés pour développer des modèles cohérents et donc d'améliorer l'interopérabilité des systèmes.
- Faciliter l'instrumentation grâce à l'utilisation de plusieurs types de dispositif pour la détection, la capture et le stockage et exploiter l'utilisation de données provenant de sources multiples, fixes ainsi que mobiles.
- L'échange de données fluide et rapide entre les différents types de topologie de réseau utilisant différents types de communication et de transmission.
- Faciliter l'utilisation et l'agrégation des données par des systèmes et des services qui peuvent ne pas avoir les générer.
- Permettre aux systèmes des TIC hétérogènes de travailler ensemble.
- Assurer l'échange et l'utilisation des données en toute sécurité.
- Permettre une plus grande automatisation qui permet les fonctions de la ville d'être livrés de façon fiable, et efficacement, en réduisant le besoin d'une intervention humaine directe où et quand cela est approprié.

Les TIC dans la santé est l'un des principaux domaines de changement dans le secteur de la santé et des services sociaux. Les technologies mobiles sont parmi ceux qui permettent, notamment, de nouveaux services qui pourraient conduire à un changement radical dans les organismes de santé et les pratiques de prestation de soins. Celles-ci pourraient être définies comme les technologies de communication et de réseaux mobiles émergents pour les systèmes de soins de santé, y compris les capteurs, les réseaux locaux sans fil, par satellite, et des systèmes cellulaires mobiles actuels et futurs.

Les biocapteurs et les autres nouvelles technologies médicales permettent de réduire les coûts de façon spectaculaire, et conduisent à faire soi-même des soins à domicile. Les progrès

récents dans l'image et la transmission vidéo sans fil vont permettre le diagnostic à distance aussi dans les scénarios sans fil et mobiles (par exemple, ambulances) [10].

En outre, les téléphones intelligents, les tablettes PC, postes de télévision, Web et vidéo et des techniques d'analyse audio sont actuellement en cours dans les principales révolutions, en changeant la façon dont les gens accèdent à l'information et la communication. Parmi les exigences techniques qui sont associés à l'adoption à grande échelle des TIC liés à la santé, on peut identifier la sécurité des données, des dispositifs de connectivité et l'interactivité, les exigences de puissance pour les appareils, les problèmes d'interface de l'utilisateur final, entre autres [10].

Les exigences techniques clés à aborder dans ce domaine sont: la sécurité (cryptage, authentification et autorisation), la découverte de service, l'évolutivité et la survie, la persistance, l'interfonctionnement, la propagation de la messagerie de l'application de communauté à communauté, l'audit et l'exploitation forestière, le partage des informations de localisation, et la migration de services d'application. [10]

Les besoins techniques d'une ville intelligente sont présentés dans le tableau suivant :

Domaine	applications de Smart city	Technologies principales
Smart energy	Gestion de la demande, le support EV, programme d'efficacité énergétique, l'intégration de l'énergie renouvelable.	compteurs intelligents, gestion de l'énergie à la maison, automatisation de la distribution, l'analyse de réseau, les systèmes de gestion de la demande (DR), stockage d'énergie
Smart water	Mises à jour du système d'eau, surveillance de la consommation, traitement des eaux usées, les systèmes de sécurité de l'environnement, la gestion des inondations	compteurs d'eau Intelligents, réseaux de capteurs et de communications, les systèmes de surveillance et de gestion de l'eau, l'analyse de réseaux d'eau, les prévisions météorologiques
Smart transportation	Surveillance et gestion du trafic, gestion de la congestion, la charge de l'utilisateur de la route, l'auto-partage, intervention d'urgence, systèmes d'information publiques, parking intelligent, la gestion des feux de circulation intégrée	Stations de voyage par CIVIQ Smartscapes, systèmes de transport intelligents, systèmes de charge EV, systèmes de tarification de l'usage de la route, réseaux de capteurs, surveillance et gestion de parking, la surveillance du trafic, l'analyse prédictive, télématique des véhicules, portails publics et applications intelligentes, plates-formes ouvertes de données
Smart buildings	Programmes de gestion de l'énergie du secteur public, intégration de réseau pour l'électricité renouvelable, stations de recharge EV, gestion de l'éclairage / déchets / eau	La construction de masse, la construction de systèmes de gestion de l'automatisation, la gestion de la performance énergétique, l'intégration au réseau, systèmes d'éclairage intelligents
Smart government	sécurité publique, protection sociale, la télésanté, e-éducation, l'éclairage public intelligent, portails des citoyens, la collecte des déchets	Les réseaux de capteurs, les services de cloud computing, l'analyse de données, plates-formes de données ouvertes, des réseaux d'éclairage, systèmes d'intervention d'urgence
Smart people	La mobilité sociale, la sécurité du public, des informations spécifiques de quartier	Les réseaux de capteurs, surveillance des vidéos, analyse de données, surveillance de l'utilisation d'Internet, la technologie de localisation, l'accès de Smartphone

Tableau 1.1. Tableau récapitulatif des besoins techniques d'une smart city [9]

7. Défis des villes intelligentes

Les défis que la smart city rencontre peuvent être résumés en trois catégories différentes: social, marché et affaires, et techniques [8] :

- Les grands défis sociaux comprennent: la communication sociale, l'accès aux services publics et privés, l'aide de soins de santé, la politique et l'éthique, et la sécurité des personnes vivant de façon autonome; utilisation des TIC comme la base pour augmenter les possibilités de socialisation des personnes âgées; échange de l'aide informelle ou un système de commerce d'échange local.

- Le Marché et les opportunités commerciales abordent: une chaîne de valeur révolutionnaire pour montrer les relations au sein de l'écosystème, de nouvelles opportunités d'affaires également pour les entreprises privées et les fournisseurs de services, un plan d'aller au marché qui doit inclure la chaîne/canal de distribution de produits, les aspects économiques et financiers, tels que la stratégie de prix, le cycle de vie du produit, démonstration publique avec un lieu de lancement.
- Enfin, les grands défis techniques englobent: la localisation géographique et le positionnement; l'interopérabilité et l'entretien de leur contexte de connectivité, tout en résidant sur un appareil mobile et traversant plusieurs réseaux (par exemple, cellulaire et WLAN, entre autres); plates-formes middleware à marge omniprésente; configurables, adaptables, des frameworks sécurisés, et les systèmes d'aide à la décision [8].

7.1. Défis sociaux

Toutes les villes cherchent à améliorer la qualité de vie de leurs citoyens et résidents. La complexité croissante des villes intelligentes présentent de nouveaux défis et opportunités à travers l'utilisation des TIC.

Les villes ont toujours été assimilées à l'anonymat de la personne, la prévalence de téléphones mobiles et d'autres appareils ainsi que les services basés sur la localisation en temps réel signifie maintenant que les villes utilisent de plus en plus l'anonymat.

Les villes représentent également une échelle de l'organisation sociale qui rend la prise de décision «en masse» peu pratique avec le résultat que les niveaux complexes d'un gouvernement représentatif et la gouvernance ont émergé au fil du temps. Les TIC éliminent la plupart des limites physiques de la prise de décision de masse tout en éliminant la plupart des aspects sociaux du discours et de l'interaction en face-à-face.

Parmi les défis sociaux dont la ville intelligente fait face, on peut citer [8]:

- L'élaboration de scénarios et prise de décision participative : est une autre situation où un Smart City framework peut fournir un ensemble précieux d'outils.
- Confidentialité et Partage d'informations : Les TIC offre des moyens plus sophistiqués par lesquels les individus peuvent gérer les aspects de la vie privée au sein de certaines normes sociales et juridiques. Développement des EFVP, identification des flux de données en utilisant ou ayant un impact sur les renseignements personnels.

- Développement des évaluations de l'impact de la confidentialité, identification des flux de données ayant un effet sur les informations personnelles.
- Environnement: échange de données entre les systèmes de Planning des Ressources d'Entreprise (ERP) et les systèmes d'information environnementaux, tels que la qualité de l'air, gestion des déchets et de traitement. la gestion intelligente de l'eau (SWM) dans les villes cherche à atténuer les défis de la gestion de l'eau en milieu urbain et le secteur de l'eau à travers l'incorporation des informations et produits des technologies de la communication (TIC), et des solutions et les systèmes dans les domaines de la gestion de l'eau et l'assainissement.
- Améliorer la capacité des villes intelligentes à répondre aux défis posés par le changement climatique et les communications entre l'équipement de sécurité / systèmes.
- L'interopérabilité entre les données du bâtiment et les systèmes de communication.

8. Cloud computing et smart cities

Le « Cloud computing » est la livraison de l'informatique comme un service plutôt qu'un produit, de sorte que les ressources partagées, des logiciels et des informations sont fournies à des ordinateurs et d'autres appareils comme un utilitaire (comme le réseau d'électricité) sur un réseau (généralement Internet). Les nuages peuvent être classés comme public, privé ou hybride. [11]

Le cloud computing offre une plate-forme technologique pour la collecte et l'exploitation de données ainsi que produire des applications de la ville sur Internet public ou un réseau virtuel privé.

D'une part, la plupart des réseaux de villes ne sont pas équipés pour fonctionner avec les inputs en temps réel à partir de capteurs, des smartphones et des compteurs électriques, ou d'autres sources d'entrée sur les fonctions des villes. Cela nécessite un matériel et un logiciel spécialisé pour recueillir des données, ainsi que des applications pour lui donner un sens, Cela demande beaucoup d'expertise.

D'autre part, le Cloud permet de réaliser une alliance nécessaire entre les capteurs et d'autres sources de données en temps réel avec des applications de smart city. [11]

Un exemple est le projet Smarter durable Dubuque de recherche à Dubuque, Iowa, qui utilise les services de cloud fournis par IBM pour tirer parti des applications qui aident les résidents et les entreprises de surveiller leur utilisation de l'eau et de l'électricité. Utiliser le cloud computing permet de passer à des applications plus efficaces et plus coordonnées [8].

9. Intelligence territoriale et smart cities

L'intelligence territoriale est un moyen pour les chercheurs, pour les acteurs et pour la communauté territoriale d'acquérir une meilleure connaissance du territoire, mais également de mieux maîtriser son développement. L'appropriation des technologies de l'information et de la communication, et de l'information elle-même, est une étape indispensable pour que les acteurs entrent dans un processus d'apprentissage qui leur permettra d'agir de façon pertinente et efficace. L'intelligence territoriale est notamment utile pour aider les acteurs territoriaux à projeter, définir, animer et évaluer les politiques et les actions de développement territorial durable. [12]

10. Synthèse

La mise en place et la gestion d'une smart city et l'interprétation de grandes quantités de données collectées requièrent l'existence et le développement d'une infrastructure de technologies de l'information et de la communication (TIC) vaste, robuste et évolutive. Ces nouvelles technologies sont au cœur du développement des Smart Cities, Elles permettent aux acteurs de la ville de rester connectés et de croiser les données nécessaires à l'amélioration de leurs besoins concrets et sont utilisées pour améliorer la qualité, la performance et l'interactivité des services, pour améliorer les contacts entre les citoyens et les parties prenante de la ville. Les applications smart city fonctionnent au-dessus de l'infrastructure TIC et sont développées pour gérer les flux urbains et permettre des réponses en temps réel qui améliorent la qualité de vie.

III. Les domaines d'applications des smart cities

Les applications des smart cities regroupent tous les logiciels qui permettent à un groupe d'acteurs (ville, communauté, citoyens) de faire face conjointement aux défis de la ville de manière intelligente et plus efficace. Elles composent un niveau fondamental des villes intelligentes: la couche numérique et comptent sur une série de technologies d'information, de communication et de programmation, Dans la partie suivante nous allons détailler quelque domaines d'application des villes intelligente [13] :

1. Mobilité et gestion de Transport

Dans le monde entier, les villes sont en face à une demande croissante de voyage ainsi qu'une incapacité d'acquérir une infrastructure suffisante pour atteindre tous les besoins.

Les systèmes de transport « intelligents » peuvent fournir aux citoyens un déplacement facile, un transport public innovant et de marchandise, accès régulier au centre de la ville, une réduction des externalités du transport public, réduction de la congestion, réduction des accidents et de la pollution de l'aire et du son.

Des solutions de gestion de mobilité font recours aux gens, en temps réel, des données utiles sur le trafic de routes qui mènent à la destination souhaitée, et le commerce avec les régions voisines, et permettre de gérer et profiter de la plupart des infrastructures (rues, parking, etc ...) et des équipements et moyens (véhicules publiques, bicyclettes, point de chargement électrique, etc ...), inclus par des applications user friendly sur tablettes et smartphones. [13]

Afin de garantir une durable « smart city », Les applications dans ce domaine doivent couvrir les exigences suivantes: réduire les besoins de mobilité pour les personnes et les biens; l'optimisation de la planification et la gestion des déplacements, la sélection du mode de transport et permettant la multimodalité transparente, améliorer le processus de fabrication de véhicules pour inclure directement les considérations de Smart Cities, augmenter la capacité des véhicules de passager et des marchandises; et la création des réseaux de transport plus efficaces. [9]

Selon leurs besoins et de la situation de circulation, chaque ville peut bien profiter des réseaux de capteurs, l'internet des objets et d'autres moyens techniques pour modifier le système de transport traditionnel, et mettre en place le système de gestion intelligente du

trafic, y compris le signal de trafic adaptatif (contrôle automatique du trafic lumières selon l'écoulement du temps) système de contrôle, système de contrôle du trafic urbain et ainsi de suite. À ce stade, le système de gestion du trafic intelligent peut réaliser l'intégration de la planification urbaine, la construction, la gestion et les opérations, et de fournir un support complet pour les autres sous-systèmes du système urbain intelligent. [14]

2. Gestion intelligente de l'eau

L'eau potable est généralement obtenue à partir de la collecte de pluie, la fonte des neiges et en prenant de l'eau de rivière pour le traitement. L'eau est utilisée pour des besoins divers, après l'usage il sera recueillis à nouveau en tant qu'eaux usées pour le traitement avant d'être retournées à la rivière pour la décharge finale dans l'océan. Le cycle des ressources d'eau se termine lorsque l'évaporation de l'eau produit des nuages d'où l'eau tombe à terre à nouveau sous forme de pluie.

Cependant, cette circulation d'eau ne fonctionne pas effectivement pour les pays et les régions souffrant d'insuffisance d'eau. En outre, il est prévu que de nouveaux moyens et technologies pour la production d'eau seront nécessaires dans les endroits où ce cycle ne fonctionne pas actuellement, mais à peine. En effet, le cycle naturel seul devient insuffisant en raison de manque d'eau causé par l'urbanisation et la concentration de la population qui sont prédit pour l'avenir de plus en plus [15].

Pour cela deux solutions sont existantes : une approche potentielle est l'utilisation de la technologie de dessalement d'eau de mer pour produire de l'eau propre à partir de l'océan, une autre consiste à utiliser la technologie de recyclage de l'eau pour une utilisation efficace des eaux usées.

Le système d'eau intelligent se compose de systèmes et technologies qui rendent l'utilisation de ressources d'eau efficace en augmentant le nombre de voies de circulation de l'eau. C'est un élément essentiel dans la réalisation de villes intelligentes.

Les technologies pour l'implémentation et la mise en œuvre d'un système d'eau intelligent inclus : des technologies de traitement conventionnelles pour l'approvisionnement et l'assainissement de l'eau, technologies de traitement des eaux usées industrielles, technologies de surveillance et contrôle des systèmes, dessalement d'eau de mer avec l'utilisation des membranes OR (Osmose Inverse), technologies pour l'offre fiable d'eau potable en utilisant les systèmes de contrôle de distribution d'eau pour réduire le taux de

fuites, technologies de gestion des informations de clients en utilisant les systèmes de compteurs intelligents, technologie avancée de traitement des eaux usées en utilisant des méthodes comme celle des systèmes bioréacteurs à membrane.[15]

La disponibilité croissante des moyens et TIC plus intelligentes pour gérer et protéger les ressources d'eau de la planète a conduit au développement de la gestion de l'eau intelligente SWM (Smart Water Management). L'approche de SWM favorise la consommation durable des ressources d'eau grâce à la gestion coordonnée de l'eau, en intégrant les produits des TIC, des solutions et des systèmes, visant à maximiser le bien-être socio-économique d'une société sans compromettre l'environnement. SWM peut être appliqué à de multiples secteurs (par exemple les industries, l'agriculture) et les environnements urbains.

Dans les villes, SWM cherche à atteindre trois objectifs principaux à travers l'utilisation des TIC, à savoir: (a) la gestion et la distribution coordonnée des ressources d'eau, (b) la protection environnementale renforcée, et (c) la fourniture durable des services publics et des efforts économiques.

Dans les milieux urbains, la mise en œuvre de SWM peut apporter des améliorations significatives dans la distribution de l'eau, en aidant à réduire les pertes dues à l'eau non-revenue, et en aidant à améliorer les eaux usées et la gestion des eaux pluviales. Le SWM assure la bonne qualité de l'eau et la fiabilité, en assurant une bonne gestion des systèmes écologiques, ce qui diminue la perte d'eau due à une fuite, ce qui réduit les coûts d'exploitation, et l'amélioration du contrôle de la clientèle et de choix. Ces améliorations augmentent l'efficacité du secteur de l'eau, tout en contribuant à sa viabilité économique depuis les municipalités et les services d'eau sont mieux en mesure de recouvrer les coûts de l'eau non-revenue, y compris la détection de connexions illégales. [16]

C'est l'utilisation des systèmes d'information en temps réel pour le pilotage des réseaux d'eau potable et d'assainissement. Ce supplément d'intelligence qui s'appuie sur le diagnostic et la prédiction permet d'économiser l'eau et de limiter le débordement et la pollution. En rendant les réseaux communicants, on accède à un niveau de performance sans précédent. Les réseaux d'eau potable bénéficient donc de la multiplication des équipements intelligents. L'exploitation et la maintenance sont concernées, avec des enjeux de pilotage de la qualité et des performances énergétiques ou hydrauliques. Grâce aux capteurs acoustiques installés sur le réseau, il est possible de détecter les fuites à tout moment et de programmer les

interventions au plus vite. Il permet aux exploitants d'intervenir sur les réseaux avec réactivité, fluidité et discrétion, sans déranger la population [16].

3. Tourisme intelligent

Le tourisme intelligent est le seul moyen de faire voyager l'information. Il doit être fondée sur l'information et l'infrastructure touristique existante, en profitant bien de l'information numérique et l'internet des objets pour obtenir la mise en place d'un ensemble de solutions, qui peuvent examiner et remplir les tâches de gestion liées au tourisme, comme les services touristiques en ligne, la gestion de la relation avec le client, la gestion de la zone opérationnelle, le développement du marché du tourisme à l'intérieur et à l'étranger, le système de gestion intelligente de l'écran, la collecte d'informations et la prévision de développement du tourisme [17]. En outre, sur la base de l'intégration de plate-forme matérielle et logicielle d'information et de services de la ville intelligente, le tourisme intelligent peut profiter du marché totalement intégré du tourisme, des attractions touristiques, des ministères et de l'information et des services des entreprises pertinentes pour promouvoir le développement de tourisme. [18]

4. Développement du E-gouvernement

Le développement de l'administration électronique efficace et effective est une condition préalable pour le développement des villes intelligentes. Les applications et technologies E-gouvernementales doivent être en mesure de répondre aux questions fondamentales de la façon dont les villes fonctionnent, comment ils sont organisés, et comment ils peuvent être amenés à travailler de façon plus intelligente pour les citoyens et les entreprises.

Une Smart City sera en mesure de réunir la technologie, l'information et la vision politique dans un programme cohérent d'amélioration urbaine et de service. Le développement des villes intelligentes affectera des milliers de zones urbaines à travers le monde, qui sont à des stades très différents de développement technologique, politique et administratif ; ces différences de maturité administrative et technologique vont façonner et limiter la capacité des différentes villes à devenir plus «intelligentes». [9]

La Gouvernance électronique ou E-gouvernance est un mouvement vers le «gouvernement en ligne» pour offrir des services et programmes, fournir des informations du gouvernement et d'interagir avec le citoyen. Il en résulte une relation conviviale avec les

citoyens, les entreprises et l'État (gouvernement). L'E-gouvernance comprend la vision, les stratégies, la planification, la direction et les ressources pour organiser les pouvoirs politiques et sociaux dans le cadre de la constitution nationale.

L'Etat ou gouvernement utilise la technologie de l'information et Internet pour soutenir les opérations gouvernementale, permettre la participation des citoyens et fournir des services. L'interaction peut être sous la forme d'obtenir des informations, des documents ou effectuer des paiements et d'autres activités via le World Wide Web.

L'UNESCO définit l'e-gouvernance comme –la Gouvernance qui se réfère à la pratique de l'autorité politique, économique et administrative dans la gestion des affaires d'un pays, y compris les citoyens et l'articulation de leurs intérêts et de la pratique de leurs droits et obligations juridiques. La gouvernance électronique peut être comprise comme la performance de cette gouvernance via le support électronique en vue de faciliter un processus efficace, rapide et transparent de diffusion de l'information au public et à d'autres organismes tout en effectuant des activités de l'administration du gouvernement.

"Un pays intelligent signifie que les gens et les entreprises sont habilitées par un accès accru aux données, plus participative grâce à la contribution des idées et des solutions innovantes, et un gouvernement plus anticipative qui utilise la technologie pour mieux répondre aux besoins des citoyens." (Infocomm Autorité de Développement de Singapour) [19].

4.1. Modèles de gouvernance intelligente

Les Smart Cities utilisent ce qu'ont appel la notion de gouvernance intelligente ; une forme avancée de Gouvernement. Premièrement et avant tous il est très important d'identifier le modèle de gouvernance approprié a chaque ville, quatre modèles de gouvernance électronique peuvent être constatés [19]:

a) Modèle-1 Aucune transformation de la structure et les processus du gouvernement

Dans ce modèle, la gouvernance intelligente est juste la gouvernance électronique d'une «ville intelligente» comme tout autre type de ville où aucune transformation n'est envisagée dans la pratique existante de la gouvernance. Elle consiste à faire les bons choix politiques et leur mise en œuvre d'une manière efficace et efficiente.

Ici la gouvernance intelligente comprend les aspects de la participation politique, des services efficaces pour les citoyens et une administration qui fonctionne bien. De plus, le

gouvernement doit approuver le développement de la ville intelligente et il peut prioriser certains domaines d'action approprié pour la Smart City [19].

b) Modèle-2 Innovation dans le processus et la mise en œuvre de prise de décision

Ce modèle définit la gouvernance intelligente par le processus de collecte de toutes sortes de données et d'informations relatives à la gestion publique à partir des capteurs. Les nouvelles technologies sont utilisées pour renforcer la rationalité du gouvernement en utilisant une information plus complète et facilement disponible et accessible pour les processus de prise de décisions gouvernementales et la mise en œuvre de ces décisions.

Le système d'aide à la décision (Decision Support System DSS) est utilisé par les entreprises en utilisant les entrées de l'information qui est disponible en temps réel. Les décisions stratégiques sont prises en utilisant le DSS pour prospérer dans les affaires basées sur la réalité actuelle. La Gouvernance municipale peut également utiliser le DSS pour la prise de décision participative basée sur les plus récentes informations.

Puisque la ville est une entité spatiale, la décision basée sur l'emplacement revêt une grande importance et les systèmes d'information géographique (SIG) permettent le développement de système de soutien décisionnel territorial qui peut être utilisé pour diverses tâches en matière de gouvernance [19].

c) Modèle-3 Création d'administration intelligente

Ce modèle estime que la gouvernance intelligente consiste à créer une administration intelligente. L'administrateur doit être en mesure de travailler avec des gens, obtenir la participation des dirigeants politiques élus et les entreprises locales, en utilisant ses compétences en ingénierie sociale. Il devrait avoir la capacité de prise de décision et la mise en œuvre rapide en utilisant des pratiques du E-gouvernance modernes et des outils de gestion de projet et des systèmes sensibles en temps réel, D'où ce modèle intègre le modèle 2.

Il devrait avoir un personnel de soutien dans la création de base de données qui peut être spatiale ou non spatiale, et permettra d'automatiser toutes les décisions par E-gouvernance ne laissant que quelques décisions importantes administratives basées sur les règles à faire par lui-même [19].

d) Modèle-4 Réorganiser la gouvernance comme Dicté par les Smart Cities

Le quatrième modèle perçoit que la gouvernance intelligente tourne autour de la réorganisation de la position du gouvernement dans le système urbain comme dicté par le système urbain lui-même. Ceci établit un système urbain très sensible de la ville intelligente avec l'interaction des capacités avec les villes externes, même si elle ne fait pas partie de la même nation.

Dans ce modèle la gouvernance intelligente signifie la collaboration entre les ministères et avec les communautés, aidant à promouvoir la croissance économique et au niveau le plus important rendant les opérations et services véritablement centrés sur le citoyen. Nous pouvons noter que c'est une adoption généralisée d'un modèle plus communautaire de la gouvernance avec une plus grande connectivité étant facilitée par les nouvelles technologies, les liens de la communauté peuvent être en dehors de la nation et de l'Etat. [19]

4.2. Synthèse des modèles de gouvernance

Comme conclusion nous constatons que chacun des modèles de gouvernance cités représente une extension de son précédent avec plus d'efficacité et notamment plus de complexité et difficulté d'implémentation tels que:

- Le 1er modèle représente une gouvernance électronique et partage d'informations avec les citoyens.
- Le 2eme modèle est basé sur les technologies de collecte de données et d'informations géographique afin de soutenir le processus de prise de décision.
- Le 3eme modèle consiste à créer une administration intelligente en plus d'une gouvernance électronique automatisée à l'aide de la participation des citoyens.
- Le 4eme modèle représente une gouvernance avec une grande connectivité avec les villes externes.

Étant donné que chaque Smart City est unique, il est impossible de recommander un modèle pour toutes les villes intelligentes potentielles. Sur la base de nombreuses considérations, le gouvernement adopte des modèles les plus approprié afin de ressortir le meilleur rendement de la ville intelligente. Les modèles 1 et 2 sont plus faciles à mettre en œuvre que les modèles 3 et 4.

Il est susceptible que les technologies de la ville intelligentes peuvent se propager dans de nombreuses villes qui est beaucoup plus facile que la mise en œuvre de tous les autres composants difficiles qui rendent le système de ville intelligente; discuté dans ce chapitre. Dans ce cas, le modèle 2 devient plus pertinent aujourd'hui. Une longue expérience et une évaluation critique des villes intelligentes sont nécessaires pour passer à des modèles 3 et 4 [19].

4.3. Concept de “smart administration” dans le contexte de « smart city »

Elle est définie comme étant un rapport intelligent entre les citoyens et l'administration qui offre la possibilité de la stabilité du développement et la qualité de l'infrastructure de transmission et de gestion de données, qui influe sur l'efficacité des services offerts aux citoyens [20].

Le concept d'une administration intelligente coïncide avec le concept philosophique et politique de la « gouvernance ouverte » (open governance). La gouvernance ouverte est en fait la troisième phase d'une nouvelle approche que le gouvernement adopté en conséquence de son adaptation avec les nouvelles technologies. La nécessité de trouver et d'adapter les meilleures formes de possibilités offertes par la technologie pour parvenir à la stabilité et au développement de l'administration publique.

Les réseaux internet ont transformé leurs utilisateurs qui étaient de simples consommateurs vers de véritables producteurs d'informations.

La nécessité d'une administration intelligente signifie le besoin d'une 'administration différente'. Qu'est-ce qui changera dans le concept d'une «administration intelligente»? Le noyau du concept comme étant au service du citoyen exige la reconstitution de la forme et du contenu de cette administration. C'est pour cela une Smart Administration signifie : Le développement durable des activités d'informations et de services envers le citoyen, En plus du partage de données (open data) qui permet de reformer les relations entre les citoyens et la gouvernance et la transparence qui est l'expression la plus importante de la gouvernance ouverte [20].

4.4. Types de gouvernance

Le secteur public, et par conséquent le domaine de la gouvernance électronique, est complexe et implique diverses parties prenantes. Certaines théories ont considéré les

«principaux» groupes de parties prenantes, tels que le gouvernement, les citoyens et les entreprises, tandis que d'autres ont segmenté les parties prenantes en termes de "rôles". Par exemple, dans [21], Rowley identifie douze rôles de parties prenantes dans l'administration électronique: les gens en tant qu'utilisateurs de services, les gens en tant que citoyens, entreprises, petites et moyennes entreprises, administrateurs publics (employés), organismes gouvernementaux, associations à but non lucratif, des politiciens, gestionnaires de l'administration en ligne, concepteurs et développeurs informatiques, fournisseurs et partenaires, et chercheurs et évaluateurs. Dans ce contexte, une personne peut appartenir à plusieurs groupes et jouer plusieurs rôles, par exemple, elle peut être un utilisateur de services, un citoyen et un employé d'une entreprise. Pour simplifier le cadre pour le gouvernement, les citoyens et les entreprises, trois composantes principales de la gouvernance électronique reflétant les relations existant entre ces parties prenantes sont communément acceptées [22]:

- La gouvernance électronique de gouvernement à citoyen (G2C), qui vise à fournir aux citoyens une variété d'informations en ligne et de services électroniques de manière efficace et rentable, et à renforcer la relation entre le gouvernement et les citoyens utilisant les TIC. Les services G2C permettent aux citoyens d'accéder aux documents gouvernementaux (lois et règlements, par exemple), d'effectuer des transactions (paiement de taxes et de services publics par exemple) et d'effectuer des tâches bureaucratiques (par exemple, changer d'adresse et demander des installations et des subventions). Dans une communication à double sens, les services G2C permettent également aux citoyens d'envoyer directement un message aux administrateurs publics, d'envoyer des votes électroniques à distance, de proposer, de discuter et de voter des initiatives publiques [22].

- La gouvernance électronique de gouvernement à entreprise (G2B), qui vise à faciliter l'interaction entre le gouvernement, les personnes morales et les organisations du secteur privé, dans le but de fournir aux entreprises des informations et des conseils sur les meilleures pratiques en matière de commerce électronique. Les services G2B permettent aux entrepreneurs d'accéder en ligne aux informations sur les législations et les réglementations, ainsi qu'aux formulaires nécessaires pour se conformer aux exigences gouvernementales en matière d'entreprise (par exemple, la déclaration de revenus des sociétés et les marchés publics).

- La gouvernance électronique de gouvernement à gouvernement (G2G), qui vise à faciliter l'interaction non commerciale en ligne entre les organisations gouvernementales, les

départements et les autorités dans le but de réduire les coûts, par exemple. dérive du fouillis de papier, des communications excessives et du personnel inutile [22].

Les différents intérêts, objectifs et avantages des parties prenantes cibles impliquent des caractéristiques dominantes des services de gouvernance électronique [21]. Des fonctionnalités distinctes peuvent donc contribuer à la qualité d'un service électronique particulier: variété de fonctionnalités, facilité d'utilisation perçue, esthétique, support client, information, communication, livraison, sécurité, fiabilité, fiabilité, fiabilité, accessibilité, compatibilité et personnalisation [22].

IV. Conclusion

Une smart city peut être définie comme un système complexe qui recouvre une multitude de domaines divers, Parmi ces domaines on peut constater l'administration intelligente qui est au cœur du développement de la ville intelligente. Elle correspond aux services et aux interactions qui lient et intègrent les différentes entités publiques, privées, et civiles afin que la ville fonctionne plus efficacement. Pour cela, le domaine d'administration intelligente sera le domaine d'application choisi dans ce travail de recherche afin de bien illustrer l'approche de smart city proposée.

Pour bien présenter les éléments principaux exploités dans notre travail, nous avons réservé le chapitre suivant au cloud computing et les systèmes de recommandation.

CHAPITRE 02 :

Cloud Computing et systèmes de recommandations

1. Introduction

Les villes intelligentes sont un réseau de technologies interconnectées qui communiquent, transfèrent et analysent des données importantes pour maintenir et améliorer les opérations urbaines. Parmi Les technologies qui ont influencés le plus le domaine de smart city nous allons faire l'étude du cloud computing et des systèmes de recommandations.

2. Définition du cloud computing

Le cloud computing est défini comme un paradigme de calcul distribué émergent dans lequel les données et les services sont disponibles dans des data centers extensibles et peuvent être accédés de manière transparente depuis des appareils (ordinateurs, téléphones, grappes, ...) connectés par Internet. [23]

Les solutions cloud reposent principalement sur des technologies de virtualisation et d'automatisation. Trois caractéristiques clés du Cloud le différencient de l'informatique traditionnelle :

- Mutualisation et allocation dynamique de capacité (adaptation élastique aux variations de charge).
- Services à la place de produits technologiques avec mise à jour en continu et automatique.
- Self-service et paiement à l'usage (en fonction de ce que l'on consomme). [24]

Le cloud computing a comme objectifs de :

- Fournir des services sur des machines virtuelles allouées sur des plates-formes de grandes tailles.
- Une méthode pour résoudre les problèmes d'extensibilité et de disponibilité pour les grandes applications.
- Le calcul distribué (enfin) démocratisé. [23]

3. Architecture du cloud computing

L'architecture de référence du Cloud Computing définit cinq grands acteurs: le cloud consommateurs, cloud fournisseurs, cloud transporteur, cloud auditeur et cloud courtier. Chaque acteur est une entité (une personne ou une organisation) qui participe à une

transaction ou un processus et / ou effectue des tâches dans le cloud computing. Les activités générales des acteurs sont comme suit [25] :

- Cloud consommateurs : Une personne ou une organisation qui entretient une relation d'affaires et utilise le service des fournisseurs de cloud.
- Cloud fournisseurs : Une personne, une organisation ou entité responsable de mettre les services à la disposition des consommateurs intéressés.
- Cloud transporteur : Un intermédiaire qui fournit la connectivité et le transport des services de cloud computing à partir des fournisseurs jusqu'aux consommateurs.
- Cloud auditeur : Un groupe qui peut procéder à une évaluation indépendante des services de cloud computing, le fonctionnement du système d'information, la performance et la sécurité de la mise en œuvre du cloud.
- Cloud intermédiaire : Une entité qui gère l'utilisation, la performance et la livraison des services de cloud computing, et négocie les relations entre les fournisseurs et les consommateurs.
- Les interactions entre les acteurs : Un consommateur de cloud peut demander des services cloud à partir d'un fournisseur directement ou par l'intermédiaire d'un courtier de cloud. Un auditeur effectue des vérifications indépendantes et peut communiquer avec les autres pour recueillir les informations nécessaires. [25]

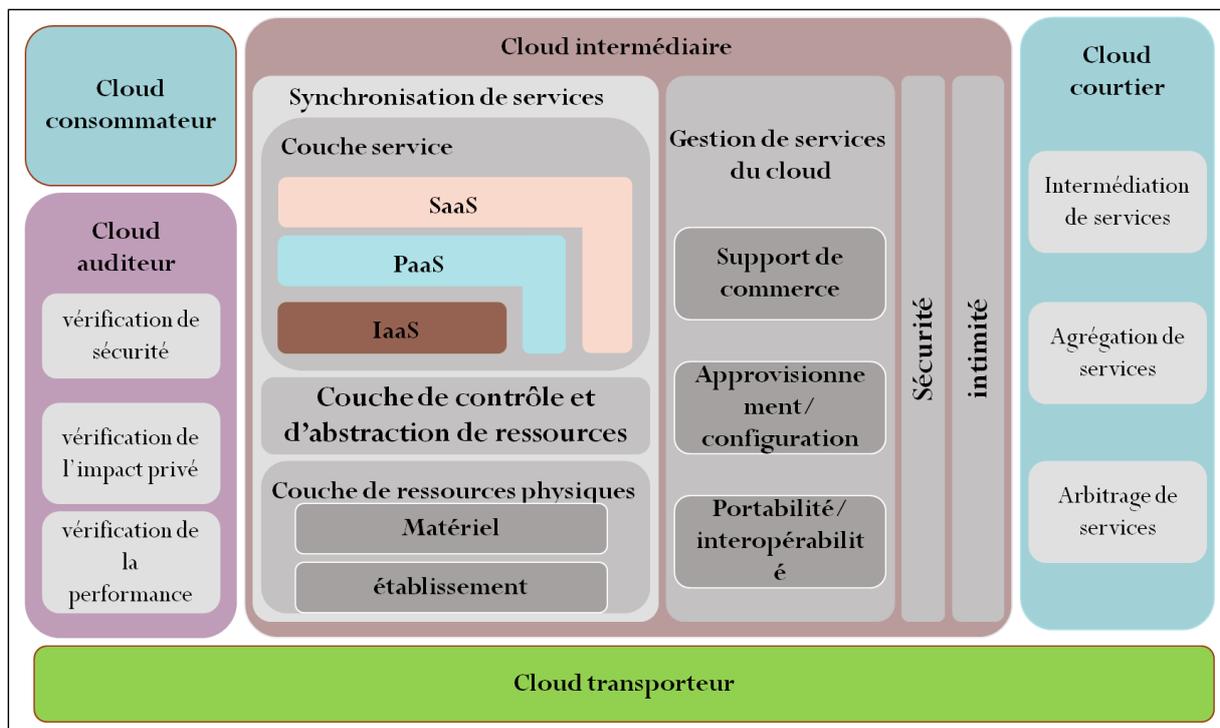


Figure 3.1. Modèle conceptuel de cloud computing [25]

4. Services du cloud computing

Bien que le Cloud soit un environnement accessible à distance, pas toutes les ressources informatiques résidant sur le cloud peuvent être mise à disposition pour l'accès à distance. Par exemple, une base de données ou un serveur physique déployé dans un nuage ne peuvent être accessibles par d'autres ressources informatiques qui se trouvent dans le même nuage. Un programme de logiciel avec une API publiée peut être déployé spécifiquement pour permettre l'accès des clients distants.

Un service de cloud est une ressource informatique qui est faite accessible à distance via un cloud. Contrairement à d'autres domaines de l'informatique qui sont orientée-services, le terme «service» dans le contexte du cloud computing est particulièrement large. Un service de cloud peut exister en tant que simple programme de logiciel basé sur le Web avec une interface technique appelée via l'utilisation d'un protocole de messagerie, ou comme point d'accès à distance pour les outils d'administration ou des environnements plus grands et d'autres ressources informatiques. [26]

Dans ce qui suit nous citons trois des types les plus communs de services de cloud tels que :

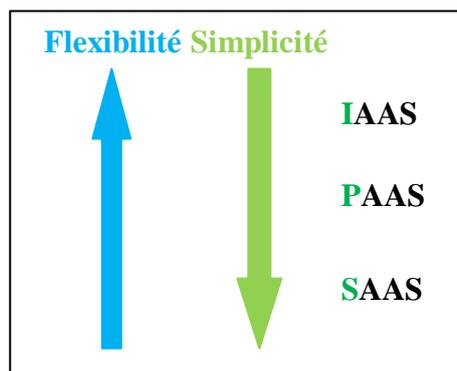


Figure 3.2. services de cloud computing [26]

4.1. IAAS

Infrastructure As A Service (IAAS) qui désigne la mise à disposition de ressources matérielles (puissance de calcul, espace de stockage). Il s'agit ici de la mise à disposition de serveurs "virtualisés". Là où dans le passé et encore aujourd'hui on louait un serveur physique aux caractéristiques déterminées, il s'agit ici de louer un serveur dont les ressources peuvent évoluer à la demande. [27]

4.2. PAAS

Platform as a Service (PAAS). Il s'agit d'offrir des ressources machines et de l'espace de stockage, mais aussi une plateforme d'exécution pour un logiciel. Ces plateformes sont spécifiques à un langage et à une base de données. L'exemple le plus connu et le plus répandu est celui des plateformes LAMP : Linux, Apache, MySQL, PHP. Les offres plus récentes de PAAS incluent la notion de temps CPU, d'espace de stockage, de bande passante consommée. On paie donc en fonction de ce que l'on consomme. [27]

4.3. SAAS

Software as a Service (SAAS). C'est l'application qui est mise à la disposition de l'utilisateur final. Ce dernier n'a besoin d'avoir aucune connaissance technique préalable. Il ouvre un compte, paie éventuellement un abonnement et utilise le logiciel (souvent des applications web) sans se soucier d'espace de stockage, de puissance machine ou même de mise à jour de logiciel. [27]

5. Types de cloud computing

Les principaux types du cloud computing sont [28]:

- le cloud privé (ou interne) : réseau informatique propriétaire ou un centre de données qui fournit des services hébergés pour un nombre limité d'utilisateurs.
- le cloud public (ou externe) : prestataire de services qui propose des services de stockage et d'applications Web pour le grand public. Ces services peuvent être gratuits ou payants.

Exemples de clouds publics: Amazon Elastic Compute Cloud (EC2), Sun Cloud, IBM's Blue Cloud, Google AppEngine And Windows Azure Services Platform.

- le cloud hybride (interne et externe) : un environnement composé de multiples prestataires internes et externes. Un exemple, IBM avait conclu un partenariat avec Juniper Networks. Cette association a permis à Big Blue de déployer son offre de cloud hybride. Ainsi les entreprises qui utilisent ce service peuvent faire basculer, par un simple glisser-déposer, des applications hébergées dans un nuage privé interne vers un nuage public sécurisé. [28]

6. Les technologies du cloud computing

Le cloud computing repose sur les trois technologies suivantes qui représentent des éléments essentiels pour sa mise en œuvre.

6.1. La virtualisation

La virtualisation représente une plate-forme de technologie utilisée pour la création d'instances virtuelles de ressources informatiques. Une couche de logiciel de virtualisation permet aux ressources informatiques physiques de fournir plusieurs images virtuelles d'eux-mêmes afin que leurs capacités de traitement puissent être partagées par plusieurs utilisateurs.

Avant l'avènement des technologies de virtualisation, le logiciel a été limité à résider sur des environnements matériels statiques. Le processus de virtualisation sépare cette dépendance entre le logiciel et le matériel, des exigences matérielles peuvent être simulées par un logiciel d'émulation en cours d'exécution dans les environnements virtualisés.

Les technologies de virtualisation établies peuvent être attribuées à plusieurs caractéristiques des clouds et des mécanismes de cloud computing, ayant inspiré un bon nombre de leurs caractéristiques essentielles. Comme le cloud computing a évolué, une génération de technologies de virtualisation modernes a émergé pour surmonter les performances, la fiabilité, et les limites d'évolutivité des plates-formes de virtualisation traditionnelles.

En tant que fondation de la technologie de cloud contemporaine, la virtualisation moderne offre une variété de types de virtualisation et des couches technologiques. [26]

6.2. Le Datacenter

Un centre de traitement de données (data center) est un site physique sur lequel se trouvent regroupés des équipements constituant le système d'information de l'entreprise (mainframes, serveurs, baies de stockage, équipements réseaux et de télécommunications, etc.). Il peut être interne et/ou externe à l'entreprise, exploité ou non avec le soutien des prestataires. Il comprend en général un contrôle sur l'environnement (climatisation, système de prévention contre l'incendie, etc.), une alimentation d'urgence et redondante, ainsi qu'une sécurité physique élevée. Cette infrastructure peut être propre à une entreprise et utilisée par elle seule ou à des fins commerciales. Ainsi, des particuliers ou des entreprises peuvent venir y stocker leurs données suivant des modalités bien définies. [29]

6.3. Plateforme collaborative

Une plate-forme de travail collaboratif est un espace de travail virtuel. C'est un site qui centralise tous les outils liés à la conduite d'un projet et les met à disposition des acteurs.

L'objectif du travail collaboratif est de faciliter et d'optimiser la communication entre les individus dans le cadre du travail ou d'une tâche. Les plates-formes collaboratives intègrent généralement les éléments suivants [29]:

- Des outils informatiques.
- Des guides ou méthodes de travail en groupe, pour améliorer la communication, la production, la coordination.
- Un service de messagerie.
- Un système de partage des ressources et des fichiers.
- Des outils de type forum, pages de discussions
- Un trombinoscope, ou annuaire des profils des utilisateurs.
- Des groupes, par projet ou par thématique.
- Un calendrier.

7. Quelques acteurs du cloud

Le cloud computing connaît un grand nombre d'acteurs dont il dépend fortement. Parmi les majeurs nous constatons : Amazon, Microsoft, Google, Salesforce, Oracle, IBM, OpenStack.

7.1. Amazon

En 2002, Amazon lançait « Amazon Web Services ». Le 'fait divers' relatant la location des ressources non-utilisées par le site-marchant à des entreprises, a donné l'idée à la société de Seattle de mettre en place un ensemble de services Web destinés à plusieurs types de clients. Le catalogue de services s'est enrichi avec le temps, et on ne compte pas moins d'une quinzaine d'offres de services à l'heure actuelle.

L'offre la plus connue est certainement Amazon Elastic Compute Cloud, ou EC2, qui permet un déploiement de machines virtuelles directement par le client, de manière automatisée, et en fonction du besoin, d'où le terme élastique. Les possibilités sont multiples et la technologie permet de définir chaque machine virtuelle comme un serveur virtuel privé[30].

7.2. Microsoft

Windows Azure l'offre commerciale proposée par Microsoft, peut être vue comme la base d'une solution SAAS. Avec Windows Azure, Microsoft met en place un « OS in the cloud », et assure donc une couche qui va avoir comme rôle de structurer l'ensemble des technologies associées, que ce soit SQL, .NET, etc. [30]

7.3. Google

Lancé en 2008, Google AppEngine est disponible uniquement en Cloud Public, sous la forme d'une offre gratuite. Sa grande force réside dans le fait qu'il soit justement gratuit, et propose des applications de qualité, malgré le très grand nombre de requête effectuées sur les serveurs. [30]

8. Des solutions clouds opensources

Les solutions open source du cloud computing sont destinées au déploiement de l'architecture en privé pour un usage en interne. Les terminologies utilisées dans les différentes solutions existantes telles que Eucalyptus, OpenNebula, Nimbus et Openstack peuvent varier mais le principe de fonctionnement de ces différents nœuds reste approximativement le même [31].

8.1. Eucalyptus

Eucalyptus est un outil open source issu d'un projet de recherche de l'université de Californie. Il est développé en C, Java, Python. Il permet de construire aussi bien les solutions privées du cloud computing que les solutions publiques. Son grand avantage est qu'il est intégré dans les distributions Ubuntu et Debian. Eucalyptus offre des interfaces compatibles avec les services EC2 d'Amazon. Ce qui lui confère la possibilité d'être employé pour les solutions hybrides de cloud computing. L'architecture d'Eucalyptus est constituée de quatre composants principaux [31] :

- Le contrôleur de nœud (Node controller NC) : contrôle l'exécution, et l'arrêt des machines virtuelles présentes sur le nœud où il est exécuté.
- Le contrôleur de cluster (cluster controller CC) : collecte les informations sur les différents nœuds d'un cluster et planifie l'exécution des machines virtuelles sur chaque nœud.

- Le contrôleur de stockage (Warlus) : c'est le composant qui gère l'accès au service de stockage. Il est souvent intégré au contrôleur du cloud (CLC).
- Le contrôleur de cloud (CLC): C'est le point d'entrée (Front end) des utilisateurs et administrateurs du système. Il collecte des informations sur les nœuds et planifie leur exécution au travers des contrôleurs de clusters (CCs). Il expose les services du cloud à travers une application Web mais également à travers des interfaces compatibles EC2. [31]

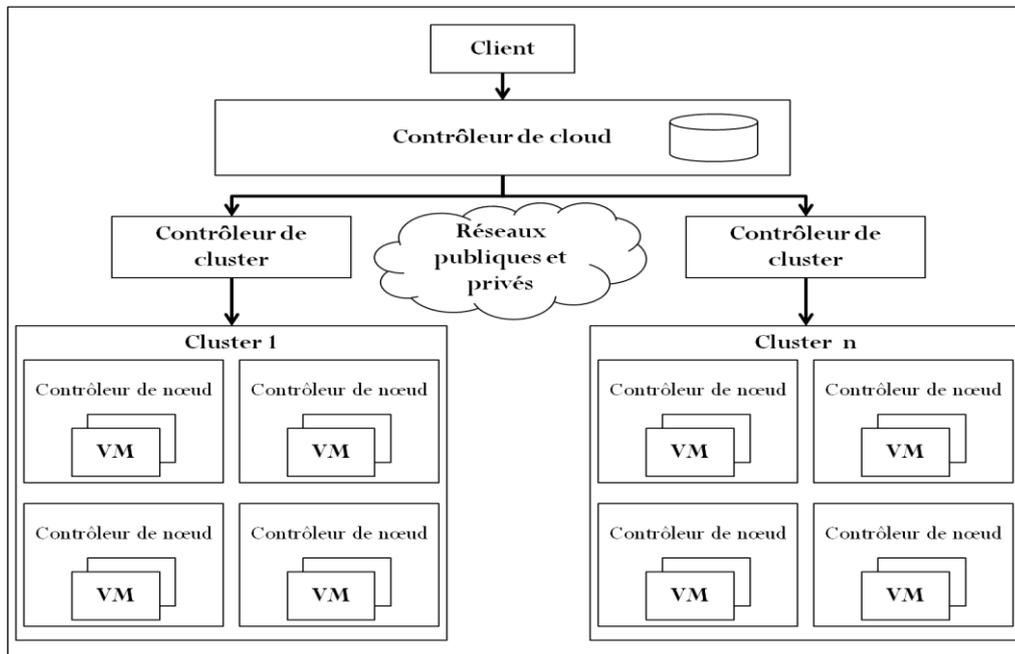


Figure 3.3. Principe de fonctionnement d'eucalyptus [31]

8.2. OpenStack

Openstack est un logiciel gratuit, son but est de fournir toutes les briques nécessaires à la mise en place d'un service IAAS. L'IAAS est l'une des facettes du cloud computing : il s'agit de fournir des ressources informatiques sous forme de machine virtuelles prêtes à l'emploi, et ce directement à la demande des utilisateurs, sans passer par la mise en place à la main des VM (Virtual Machine) par un administrateur système. Le système IAAS gère la mise à disposition de la ressource en fonction de la capacité actuelle de la plateforme, et assure le suivi de consommation à des fins de facturation. OpenStack, développé initialement par Rackspace et la NASA, propose un regroupement de logiciels open source sous licence Apache. Ce regroupement permet de mettre à disposition des ressources de calcul (des machines virtuelles) ainsi que des espaces de stockage, en reprenant les principes de l'IAAS[30].

8.3. Synthèse des solutions de cloud open-source

Les solutions open-source du cloud computing présentées dans ce chapitre sont dotées d'une architecture de supervision des nœuds d'exécution des machines virtuelles composée d'un nœud central appelé cloud controller (CLC), de plusieurs nœuds d'exécution des services du cloud appelés Node controllers et parfois d'un ensemble de nœuds intermédiaires de supervision appelés cluster controllers. Le principe de fonctionnement est le suivant [30] [31]:

- Le cloud controller représente le point central de supervision et de gestion de toute l'architecture.
- Le rôle du cluster controller consiste à gérer un ensemble de nœuds regroupés. Il sert d'intermédiaire entre le cloud controller et les différents nœuds d'exécution de services (node controllers). L'intérêt consiste, entre autres, à désengorger le cloud controller dans sa tâche de supervision de toute l'architecture en collectant les informations au niveau des différents nœuds et en les envoyant au cloud controller suivant les requêtes de ce dernier. On peut alors avoir plusieurs cluster controllers dans une architecture de cloud computing mais seulement un unique cloud controller.
- Les contrôleurs de nœuds (node controllers) représentent les points d'exécution des services du cloud. Ces nœuds sont généralement équipés des outils nécessaires pour la création et l'exécution de machines virtuelles.

9. Cloud computing et sensibilité au contexte (context-awareness)

Comme indiqué précédemment, les services de communication et d'information fournis par les TIC apportent un aspect intelligent à la gestion des villes en transformant les données en connaissances utiles et en informations exploitables. Le défi de la réalisation de villes intelligentes grâce aux TIC réside dans l'intégration de données provenant de sources disparates et leur traitement en informations utiles fournies par le biais de services, utilisés par les citoyens et les administrations publiques. Ce défi revêt différentes dimensions, notamment la collecte d'énormes quantités de données, l'agrégation de données sous différents formats, la pertinence de telles données par rapport à des problèmes et scénarios pratiques, l'analyse des données pour déduire des informations utiles et la gestion des données historiques qui augmentent de manière continu [6].

Les sources de données relatives aux environnements urbains continuent à augmenter, allant des services de collecte de données et des plateformes de détection déployées par les autorités administratives aux données collectées à partir des citoyens via des mécanismes de détection participatifs et opportunistes. Cependant, l'absence d'une plateforme commune et standardisée pour la collecte, le stockage, l'analyse et la diffusion de telles données restreint considérablement les perspectives d'informations et de services pouvant être développés et bénéficiant aux infrastructures sous le thème de la ville intelligente. Le cloud computing - basé sur les concepts d'infrastructure convergée et de services partagés - peut être utilisé pour relever certains des défis de ce domaine, principalement ceux liés à la collecte, au stockage et au traitement des données du domaine urbain [6].

L'un des éléments qualitatifs important des TIC pour les villes intelligentes est la gouvernance inclusive par des citoyens informés grâce à la participation, c'est-à-dire un aspect de la science citoyenne qui permet leur participation à la collecte et à l'utilisation d'informations. D'autre part les citoyens seront probablement mieux informés et plus engagés si les services qu'ils consomment sont liés à leur contexte social et environnemental [6].

La sensibilité au contexte (context-awareness), qui est une discipline établie dans les domaines de l'informatique et les interactions homme-machine, est donc pertinente pour l'emploi des TIC dans les villes intelligentes. Cependant, l'intégration de telles informations contextuelles n'est pas simple en raison d'un certain nombre de facteurs. Ceux-ci incluent la disponibilité d'informations contextuelles pertinentes, l'hétérogénéité des informations disponibles, l'interconnexion de différents types de contextes provenant éventuellement de sources différentes, l'agrégation d'informations contextuelles provenant de sources indépendantes pour inférer un contexte d'ordre supérieur sur les citoyens et leurs environnements, ... etc. La fourniture de services sensibles au contexte dépend également de l'intégration transparente des systèmes de collecte, de traitement et de diffusion des données[6].

10. Systèmes de recommandation

Un système de recommandation est un système qui fournit une recommandation, prédiction, opinion, ou une liste configurée d'articles qui aide l'utilisateur à les évaluer. [32]

Il est considéré comme une forme particulière de filtrage d'informations conçue pour présenter des éléments d'information (films, musique, livres, images, pages Web,...) susceptibles d'intéresser l'utilisateur. [33]

Habituellement, les systèmes de recommandation essaient de prédire la note qu'un utilisateur donnerait à un article (musique, livres, ...) qu'il n'a pas encore considéré, en utilisant un modèle construit à partir des caractéristiques d'un article (approches basées sur le contenu) ou de l'environnement social de l'utilisateur (approches de filtrage collaboratif). [33]

10.1. Définition formelle d'un système de recommandation

Un modèle de recommandation a pour but de suggérer à l'utilisateur un contenu informationnel susceptible de répondre à ses besoins à long terme en se basant sur un processus de filtrage d'information. Les besoins en informations à long terme de l'utilisateur, que le système essaie de découvrir peuvent être vus comme une requête (ou un ensemble de requêtes).

Le principe de base du filtrage est de traiter les documents provenant de sources dynamiques et de décider, si le document correspond ou pas aux besoins en information des utilisateurs, besoins exprimés à travers le concept du profil utilisateur [34].

Formellement, le problème traité par un système de recommandation peut être décrit comme suit [35]:

soit $\phi = \{S, U\}$ l'univers des informations de base manipulées par le modèle, tels que :

- $S = \cup_{j=1}^n s_j$ est l'ensemble des services de la collection et n le nombre total des services.

-Le profil est la structure informationnelle représentant l'utilisateur à travers trois dimensions corrélées, évoluant dans le temps, noté $U = (Ds, Ps, Cs)$. Tels que Ds représente les informations démographiques de l'utilisateur, Ps les préférences des services de l'utilisateur jusqu'à l'instant s et Cs représente le contexte de l'utilisateur à l'instant s .

Pour chaque requête soumise, l'utilisateur, les requêtes et les services utilisés sont collectés comme un triplet d'informations concurrentes dans le profil. Où l'utilisateur $u \in U \{u_1, u_2, \dots, u_n\}$, la requête $q \in Q \{q_1, q_2, \dots, q_m\}$ et les services associés $s \in S \{s_1, s_2, \dots, s_l\}$.

Le traitement de ces données s'effectue de deux manières. Si la requête de l'utilisateur a déjà été soumise, le système est en mesure de retrouver dans le profil les services pertinents les plus fréquemment visités ou utilisés par l'utilisateur. Par contre, si c'est une nouvelle requête, le problème qui se pose est : Etant donné les services déjà sélectionnés, quelle est la pertinence des services qui devraient être sélectionnés en réponse à cette requête.

Étant donné un ensemble T contenant les données collectées représentées par des triplets (u, q, s) , et une fonction de mapping $f : U * Q \rightarrow S$ doit être apprise par le modèle. La résolution de ce problème est donnée en appliquant l'approche de recommandation.

Il est à noter que lors d'un scénario de recherche, une observation est un triplet (u, q, s) correspondant à l'événement qu'un utilisateur u soumet une requête q à un moteur de recherche, et sélectionne un service s à partir des résultats retournés. La fonction f permet de sélectionner pour chaque paire d'information "utilisateur, requête" notée (u, q) contenue dans le profil, une liste de nouveau services ou chaque service s' n'ayant pas été déjà collectée dans le profil (c-à-d, que le triplet $(u, q, s') \notin T$), tel que le service s' sélectionnée est celui ayant une grande valeur d'évaluation.

Le processus de recommandation peut s'effectuer selon deux approches de filtrage [34] :

_ *Un filtrage par le contenu* (où filtrage cognitif). Il tient compte seulement des contenus du document et du profil.

_ *Un filtrage collaboratif* (où filtrage social). Il se base sur les évaluations attribuées par des groupes d'utilisateurs aux documents.

Ces deux approches sont décrites dans ce qui suit.

10.2. Recommandation basée sur le contenu

Dans la technique de filtrage basée sur le contenu, la recommandation est faite en fonction du profil de l'utilisateur en utilisant des caractéristiques extraites du contenu des documents que l'utilisateur a évalués dans le passé. Les documents qui sont principalement liés aux documents positivement évalués sont recommandés à l'utilisateur.

Le système accompagne un utilisateur en analysant en arrière plan les documents consultés. Il utilise pour cela un ensemble d'heuristiques (sauvegarde d'un document par l'utilisateur, sélection, abandon d'une procédure, etc.) pour identifier les documents ayant

intéressé l'utilisateur et en déduire ses objectifs. Il explore ensuite dynamiquement les documents utilisés pour évaluer les documents correspondants.

Les documents sont enfin ordonnés en estimant leur intérêt pour l'utilisateur, cette liste est présentée à l'utilisateur lorsque ce dernier demande une recommandation.

En outre, l'accent peut être mis sur les processus utilisés pour apprendre les caractéristiques des documents que l'utilisateur juge intéressants [36].

10.3. Recommandation basée sur la collaboration

Dans cette approche les utilisateurs du système participent activement à l'alimentation d'une base de données gérée par le filtre, contenant des informations sur eux-mêmes, et sur les documents qu'ils ont consulté. Les recommandations pour le nouvel utilisateur sont basées sur ces prédictions. Ainsi, tous les utilisateurs peuvent tirer profit des évaluations des autres en recevant des recommandations pour lesquelles les utilisateurs les plus proches ont émis un jugement de valeur favorable, et cela sans que le système dispose d'un processus d'extraction du contenu des documents [36].

Le système observe le comportement de l'utilisateur et lui recommande les documents pertinents en fonction de son profil et des profils similaires des autres utilisateurs. Chaque utilisateur définit en quelques termes son objectif, puis fait la recherche en utilisant le système, ce dernier utilise ces définitions pour attribuer des poids aux documents associés. Ces poids sont ensuite utilisés dans le processus de recommandation en les comparant avec les préférences initialement exprimés par l'utilisateur. Le système propose les documents jugés les plus en rapport avec l'objectif énoncé par l'utilisateur [36].

10.4. Système de recommandation sensible au contexte

Traditionnellement, les systèmes de recommandation collaboratifs et basés sur le contenu, traitent deux types d'entités, les utilisateurs et les articles. Cependant le système de recommandation sensible au contexte est une approche où le paradigme traditionnel utilisateur/article a été étendu pour prendre en charge des dimensions supplémentaires capturant le contexte dans lequel les recommandations sont faites [37].

Le contexte est l'information sur l'environnement d'un utilisateur et les détails de la situation dans laquelle il se trouve. Ces détails peuvent jouer un rôle important dans les recommandations ainsi que les évaluations des articles. Ces données peuvent être utilisées

pour adapter les recommandations en fonction des caractéristiques des utilisateurs, telles que le temps disponible, l'emplacement, les personnes à proximité, etc.

Certaines recommandations peuvent être plus adaptées à un utilisateur le soir et ne correspondent pas du tout à ses préférences le matin et il / elle aimerait faire une chose quand il fait froid et complètement une autre quand il fait chaud dehors. Les systèmes de recommandation qui prêtent attention et utilisent ces informations pour formuler des recommandations sont appelés systèmes de recommandation sensibles au contexte [38].

Contrairement aux informations de contenu qui sont enregistrées dans les profils, le contexte change dynamiquement et il est souvent enregistré de manière permanente, car il est plus susceptible de perdre sa crédibilité après une certaine période de temps. C'est pourquoi il est très important de rafraîchir périodiquement les informations. Les systèmes de recommandation tenant compte du contexte ont retenu beaucoup l'attention, car ils ont sensiblement amélioré la qualité des recommandations et les approches sont devenues plus spécifiques à utiliser dans certains domaines [38].

Ces informations contextuelles peuvent être obtenues de plusieurs façons; les informations peuvent être obtenues de manière explicite en interagissant directement avec l'utilisateur lui demandant de remplir un formulaire et de réaliser une enquête. Une autre façon consiste à collecter des informations implicitement en utilisant des sources telles que le GPS, pour obtenir un emplacement ou un délai de transaction [38].

Différents paradigmes ont été proposés pour incorporer des informations contextuelles dans le processus de recommandation [37]: pré-filtrage, post-filtrage et modélisation contextuelle.

- Dans une approche de pré-filtrage contextuel, les informations contextuelles sont utilisées pour filtrer l'ensemble de données avant d'appliquer un algorithme de recommandation traditionnel.
- Dans une approche contextuelle de post-filtrage, des recommandations sont générées sur l'ensemble des données. L'ensemble de recommandations qui en résulte est ajusté à l'aide des informations contextuelles.
- Les approches de modélisation contextuelle utilisent des informations contextuelles directement dans la fonction de recommandation comme prédicteur explicite d'une note pour un élément.

Alors que les approches contextuelles de pré-filtrage et de modélisation peuvent utiliser des algorithmes de recommandation traditionnels, l'approche de post-filtrage contextuel utilise des algorithmes de recommandation multidimensionnels [37].

10.5. Problème de scalabilité dans les systèmes de recommandations

Généralement, le terme « scalabilité » désigne la capacité d'un système à performer de manière adéquate et à gérer de manière raisonnable les grandes charges de travail (workloads)[36]. De ce fait, le problème réside dans les difficultés d'administration de données dans la création et la maintenance des systèmes à grand échelle ; le processus d'administrer une base de données sur cloud devient alors de plus en plus incontrôlable avec la croissance du nombre des bases de données et des administrateurs et surtout lorsque l'intervalle de demandes devient suffisamment large [41].

Dans les systèmes de recommandations sensibles au contexte, il est nécessaire de séparer entre le transfert et traitement des données contextuelles et l'application qui les utilisent, en introduisant une couche intermédiaire dont ces principales fonctionnalités sont: la collecte, le traitement, et la dissémination des informations pour l'application en question. La manière la plus simple pour réaliser cette séparation est d'utiliser un serveur centralisé de contexte qui se charge de fournir les informations contextuelles pour les applications [42].

Cependant, la quantité de données utilisées en entrée augmente rapidement à fur et à mesure que de nouveaux utilisateurs et éléments sont ajoutés. La taille des données relatives au comportement de l'utilisateur stockées peut facilement atteindre les TBs par jour. Malgré la grande quantité de données, la plupart des systèmes de recommandations tentent de répondre de manière interactive en moins d'une seconde afin de maintenir l'engagement des utilisateurs. Un défi clé ici consiste à concevoir des systèmes efficaces capables de gérer de tels ensembles de données à grande échelle. [43]

Pour cette raison, l'approche backend as a service peut servir comme une bonne solution du problème de scalabilité des systèmes de recommandations.

11. Synthèse

L'un des plus grands défis des systèmes de recommandations c'est la scalabilité. Traditionnellement, les systèmes de recommandation ont été conçus en utilisant une architecture centralisée. Cela empêche une scalabilité rapide et entraîne des goulots lorsque

les données deviennent très volumineuses. Une technologie de la nouvelle génération comme le Cloud Computing peut gérer des données massives et prendre en charge un énorme traitement.

Dans la partie suivante nous allons aborder le problème de scalabilité et ses stratégies ainsi que l'approche backend as service comme solution à ce problème.

12. La scalabilité et ses solutions

La scalabilité, c'est la capacité de changer, à la demande et de manière transparente pour l'utilisateur, l'ordre de grandeur (accroître ou réduire les capacités) d'un équipement, d'un logiciel ou d'un processus informatique. Une capacité qui prend toute sa dimension avec l'évolution des technologies, des volumes de données et du trafic. La scalabilité est désormais une caractéristique cruciale pour les systèmes informatiques et surtout dans le contexte de smart city. Dans cette partie nous allons étudier le problème de scalabilité de données ainsi que les différentes solutions adoptées afin de réaliser un système scalable.

12.1. Définition d'un système scalable

La métrique qui permet de déterminer qu'un système est probablement scalable ; c'est-à-dire qu'il soit capable de continuer à fonctionner avec performance acceptable lorsque la charge de travail augmente, c'est la mesure du rapport demande/réponse d'une certaine ressource [40].

Formellement, un système scalable idéal est un système dont le comportement commence avec des petits pas et reste scalable tout au long jusqu'au futur indéfini ; la Figure 4.1 illustre un tel système [39].

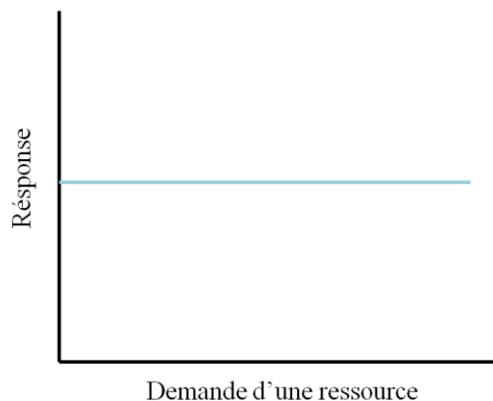


Figure 4.1. Un système scalable idéalisé [39].

Cependant, en réalité, un tel système idéalisé ne peut pas être atteint ou bien au moins pas jusqu'au bout, tout système commence toujours à rencontrer des ressources insuffisantes qui ne peuvent pas être augmentées de manière rentable. Lorsque ce point est atteint, la courbe commence à augmenter. Pour un tel système, nous disons que la région de scalabilité se termine là où la courbe commence à monter. Toutefois, une région de scalabilité n'a pas à démarrer lorsque le système est déployé, il risque de ne pas bien démarrer et de s'améliorer progressivement à mesure que la demande augmente [39].

Lorsqu'un système est scalable, ça signifie qu'il évolue dans un sens et pour une période donnée, et qu'aucun système ne peut être scalable à l'infini.

La figure 4.2 suivante illustre les courbes de réponse en fonction de la demande pour deux systèmes A et B. Clairement, pour tout les niveaux de demande, la métrique de réponse est plus grande (pire) pour B que pour A. S'il existe une valeur maximale tolérable pour la métrique de réponse, le système B l'atteindra avant le système A. C'est pourquoi nous déduisons que le système A est plus performant que le système B. Si les deux courbes continuent d'augmenter de façon monotone, il est clair que, sans une intervention quelconque et face à une demande toujours croissante, les deux atteindront éventuellement un point où la demande de ressources dépasse leur disponibilité et la métrique de réponse devient insatisfaisante. À ce stade (différent pour chaque système), le système aura atteint sa limite de scalabilité [39].

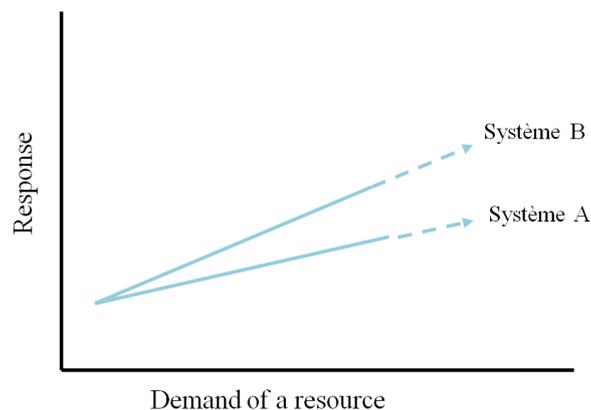


Figure 4.2. Réponse versus demande [39].

La figure 4.3 montre un système imaginaire qui tolère une demande accrue jusqu'à ce qu'il se heurte à une limite de conception importante. Plus le coude de cette courbe se trouve à droite, plus le système peut être scalable. Un système fonctionnant près du coude de la courbe

n'est plus scalable car pratiquement toute augmentation de la demande de la ressource entraînera une mesure de réponse excessive.

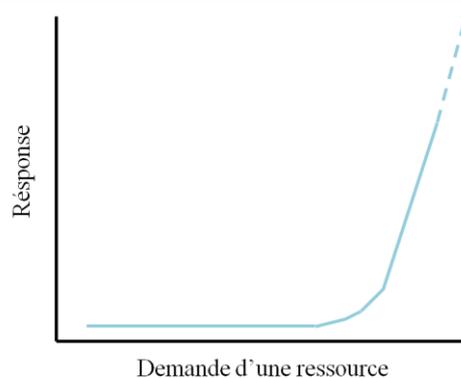


Figure 4.3. Echech de scalabilité [39]

Bien évidemment, les exemples de la figure 4.1, 4.2 et 4.3 sont idéalisés, mais les systèmes réels présentent un comportement hybride. La figure 4.4 dessous illustre cette situation. Lorsque le système est en cours d'exécution dans la zone inférieure, il répond bien. Quand il commence à fonctionner dans la zone centrale, la réponse commence à devenir inacceptable. Ensuite, quand il passe dans la zone supérieure, la réponse est totalement inacceptable. Dans l'exemple, nous voyons que les administrateurs du système ont été en mesure de réagir rapidement à deux reprises pour que la réponse soit acceptable, alors qu'une fois, ils ont pris plus de temps. Finalement, le système a atteint un point où aucune action raisonnable n'a empêché la métrique de réponse de prendre une valeur inacceptable [39].

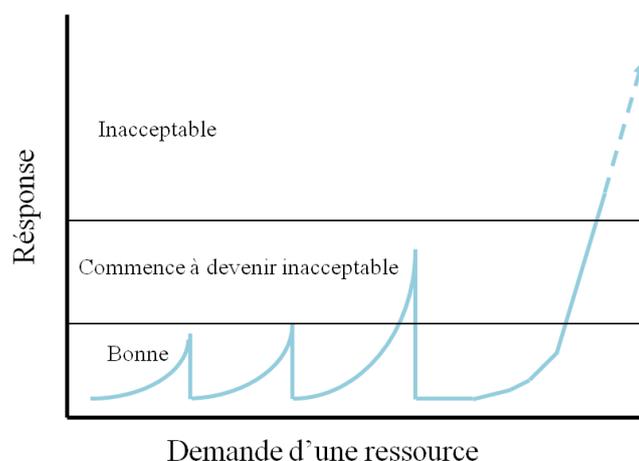


Figure 4.4. Métrique de réponse par rapport à une demande pour une ressource [39]

Si le système continue de répondre aux niveaux les plus élevés de la demande, il est alors extensible jusqu'à ce qu'aucune action ne puisse résoudre les problèmes de métrique de réponse. Inversement, si le système initial est construit et manque de ressources supplémentaires, le système n'est pas scalable [40].

12.2. Strategies de scalabilité

Dans sa forme la plus simple, la solution de scalabilité consiste à ajouter plus de ressources afin de gérer la charge accrue qu'un système peut supporter. Pour cela, deux grandes méthodes sont connues: la scalabilité verticale et la scalabilité horizontale, permettant d'ajouter plus de ressources pour un système particulier [40].

12.2.1. Scalabilité verticale

La mise à l'échelle verticale ou scalabilité verticale fait référence à la maximisation des ressources d'une seule unité afin d'accroître sa capacité à gérer une charge croissante. Sur le plan matériel, cela inclut l'ajout de puissance de traitement et de mémoire à la machine physique exécutant le serveur. En termes logiciels, la mise à l'échelle peut inclure l'optimisation des algorithmes et du code d'application. L'optimisation des ressources matérielles, telle que la parallélisation ou l'optimisation du nombre de processus en cours, est également considérée [44].

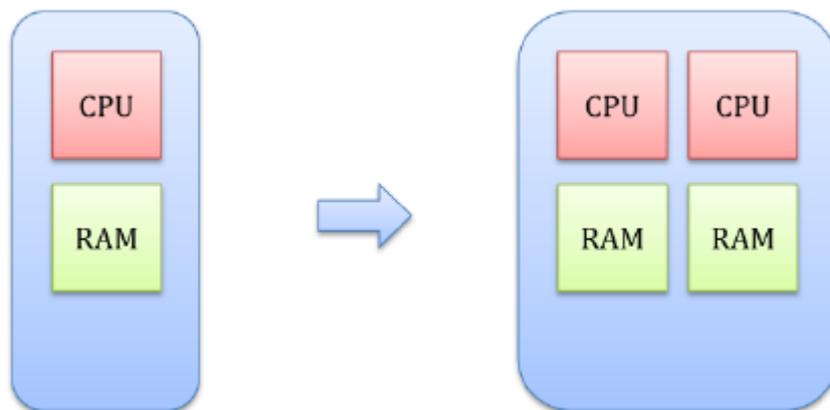


Figure 4.5. Exemple de scalabilité verticale [44]

Bien que la mise à l'échelle verticale peut être relativement simple, cette méthode présente plusieurs inconvénients. Premièrement, l'ajout de ressources matérielles se traduit par des rendements décroissants au lieu d'une scalabilité linéaire. Le coût de l'expansion augmente également de manière exponentielle. La courbe de coût du traitement informatique

est une mesure de puissance dans laquelle le coût commence à augmenter de façon disproportionnée par rapport à l'augmentation de la puissance de traitement fournie par des serveurs plus grands [40].

De plus, il y a l'inévitable exigence de temps d'arrêt pour la reproduction à une plus grande échelle. Si tous les services et les données de l'application Web résident sur une seule unité, l'échelle verticale de cette unité ne garantit pas la disponibilité de l'application [40].

12.2.2. Scalabilité horizontale

La scalabilité ou la mise à l'échelle horizontale fait référence à l'incrément de ressources par l'ajout d'unités au système. Cela signifie l'ajout de plusieurs unités de plus petite capacité au lieu d'ajouter une seule unité de plus grande capacité. Les demandes de ressources sont ensuite réparties sur plusieurs unités, réduisant ainsi la charge excédentaire sur une seule machine [44].

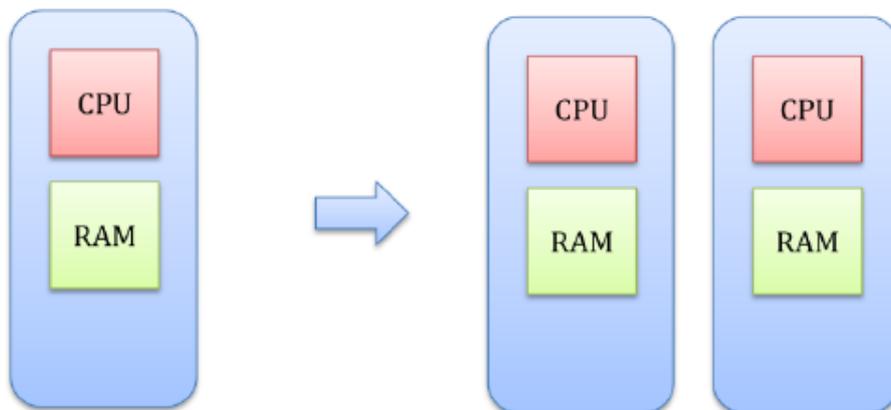


Figure 4.6. Exemple de scalabilité horizontale [44]

Avoir plusieurs unités nous permet de garder le système en service même si certaines unités tombent en panne, évitant ainsi le problème du «point de défaillance unique» et augmentant la disponibilité du système. De manière générale, le coût total engendré par plusieurs petites machines est inférieur au coût d'une seule unité plus grande. Ainsi, la mise à l'échelle horizontale peut être plus rentable que la mise à l'échelle verticale [44].

Cependant, la mise à l'échelle horizontale présente également des inconvénients. L'augmentation du nombre d'unités signifie que plus de ressources doivent être investies dans leur maintenance. De plus, le code de l'application elle-même doit être modifié pour permettre

le parallélisme et la répartition du travail entre différentes unités. Dans certains cas, cette tâche n'est pas futile et la mise à l'échelle horizontale peut s'avérer difficile [40].

12.3. L'approche back-end as a service

BAAS est une approche permettant de connecter des applications au stockage et au traitement du cloud back-end [45], ainsi que d'autres fonctionnalités telles que la gestion des utilisateurs et des périphériques, les notifications push et l'intégration aux services de réseaux sociaux.

Ainsi, BAAS fournit un ensemble de fonctions pour connecter les applications aux services disponibles dans le cloud, et une manière cohérente de gérer les données mobilisées.[46]

Les services BAAS sont fournis via des kits de développement logiciel (SDK) personnalisés et des interfaces de programmation d'applications (API).

Il s'agit purement d'un back-end, fournissant une infrastructure qui évolue et optimise automatiquement, associée à un ensemble de ressources essentielles dont les développeurs ont besoin, comme le contenu, les données, le stockage, la messagerie et l'authentification. Les API ont permis d'identifier et de mettre ces ressources à la disposition des développeurs. Dans le cadre de BAAS, une progression naturelle des API à partir d'utilisations uniques vers des piles significatives de ressources. [45]

Un BAAS permet de stocker des données et de coupler un data store de base avec des outils d'approvisionnement et d'authentification expressifs. C'est comme une base de données NoSQL dans le cloud où l'accès se fait via des APIs REST. Le stockage basé sur le cloud ne représente qu'une partie des offres BAAS. [47]

Comme les technologies du cloud computing, BAAS se caractérise par son élasticité. Ce qui signifie que les ressources matérielles sont allouées et peuvent être étendues en fonction des besoins du système, ce qui donne à un système l'opportunité de s'adapter à l'échelle (scale up) uniquement en branchant plus de serveurs et sans aucune modification du code; en revanche, il utilise également un ensemble des services RestAPI qui contrôlent l'accès aux données, ce qui permet d'éviter les surcharges de demandes. Nous pouvons donc en déduire que l'approche BAAS offre à la fois une solution matérielle et logicielle au problème de l'extensibilité.

La scalabilité de l'approche BAAS est basée sur l'architecture des services API Rest et assurée via deux contraintes: l'interface uniforme et les interactions sans états (uniform interface and Stateless interactions constraint). [48]

- interface uniforme: cela signifie que toutes les ressources exposent les mêmes interfaces au client dans l'architecture REST. Comme le plus connu REST est formé sur la base du protocole HTTP qui est utilisé comme un protocole de transfert d'état en plus d'un protocole de transport de données. HTTP peut non seulement localiser une ressource de façon unique, mais aussi nous dire comment faire fonctionner la ressource et exige qu'une requête et une réponse soient effectuées via quatre opérations http: GET, POST, PUT et DELETE. [45]

-GET récupère l'état actuel d'une ressource dans une représentation,

-PUT crée une nouvelle ressource,

-POST transfère un nouvel état sur une ressource,

-et DELETE supprime une ressource existante.

Chaque méthode a une sémantique attendue pour décider quelles méthodes conviennent à chaque ressource. GET, PUT et DELETE sont idempotents, car les trois méthodes utilisées plusieurs fois pour une ressource ont le même effet qu'une seule fois.

Une interface uniforme devient la base de la définition de l'interface, elle aide à découpler le client et le serveur et leurs permet d'évoluer indépendamment. Tant que l'interface reste inchangée, le client et le serveur peuvent interagir normalement.

Une contrainte d'interface uniforme déplace la variabilité d'une représentation d'interface à une représentation de ressource, ce qui rend notre préoccupation plus proche de l'objectif ultime des données. [48]

- Les interactions sans état améliorent également l'évolutivité de l'architecture RESTful. La contrainte des interactions sans état exige que la requête de chaque client contienne tous les états d'application nécessaires pour cette requête. Aucune information d'état n'est conservée sur le serveur et aucune n'est impliquée par les demandes précédentes. Les interactions sans état réduisent le coût de l'agrandissement du système. Comme toutes les conversations sont sans état, lorsque l'échelle du système augmente, la seule chose nécessaire

est de brancher plus de serveurs à charge équilibrée et la coordination entre différents serveurs n'est pas nécessaire. [48]

12.3.1. Comparaison de l'approche BAAS par rapport aux services du cloud computing

En tant que forme de services cloud, l'approche BAAS peut être classée quelque part dans la région entre les types de logiciels et de plate-forme en tant que service (SAAS et PAAS). Selon un article de l'Institut national des normes et de la technologie (NIST) [49], les types de services cloud peuvent être approximativement distingués en fonction du contrôle que le consommateur et le fournisseur du cloud ont sur l'environnement informatique.

Dans le type SAAS, le consommateur du cloud n'a pas beaucoup de contrôle ou de tâches à accomplir avant d'utiliser le cloud. Il n'a que le contrôle de l'application, tandis que le fournisseur de cloud doit gérer les installations, le matériel, l'infrastructure virtualisée, l'architecture de la plate-forme et éventuellement une partie de l'application.

En ce qui concerne le type PAAS, il permet le consommateur cloud de garder le contrôle sur l'architecture de la plate-forme et de l'application pendant que le fournisseur aurait le contrôle sur le matériel de l'infrastructure virtualisée et sur les installations du service, il inclut également la puissance de stockage et de calcul et une large gamme de services [49].

Cependant, certains fournisseurs BAAS permettent aux consommateurs de choisir leur propre base de données et d'utiliser des services push pré-implémentés ou un autre support applicatif, tandis que d'autres fournisseurs n'offrent qu'une base de données et aucun support applicatif pré-implémenté. Étant donné que les applications frontales sont totalement indépendantes des offres BAAS et qu'il fournit principalement des options d'arrière-plan universellement importantes, les offres BAAS permettent aux développeurs de se concentrer uniquement sur les données de l'application.

Sous l'aspect contrôle, l'utilisateur contrôle l'application et le fournisseur BAAS contrôle tout le reste, y compris la disponibilité et d'autres exigences importantes pour les services de cloud. Le service back-end est capable de stocker, modifier et supprimer des objets dans la base de données; il peut être capable d'exécuter des fonctions écrites par le développeur, en fonction de l'offre. Cela permet aux fournisseurs BAAS d'adapter leurs offres aux besoins des développeurs frontaux [50].

13. Conclusion

Dans ce chapitre nous avons étudié les différentes stratégies de scalabilité parmi lesquelles l'approche back-end as a service. C'est une nouvelle branche de cloud computing qui permet de fournir un moyen de connecter des applications au stockage et au traitement cloud back-end. L'approche BAAS peut servir comme solution d'une scalabilité verticale pour les systèmes dans les grandes charges de travail.

Dans le chapitre suivant nous présentons les travaux relatifs au domaine de smart city.

CHAPITRE 03 :

Travaux Relatifs

1. Introduction

Afin d'acquérir des connaissances à partir des expériences précédentes dans le domaine de smart city nous avons choisi de faire l'étude de non seulement des travaux de recherche dans ce domaine mais aussi des exemples du monde réel. Il existe un grand nombre d'initiatives de smart city dans le monde parmi lesquelles nous avons choisi les deux exemples d'IBM Smarter city et Seoul smart city afin de profiter des approches utilisées dans ces deux projets. D'autre part nous allons étudier quelques travaux de recherche dans le domaine, plus précisément dans les systèmes de recommandations afin de bénéficier et exploiter les avantages de leurs approches.

2. Projets de smart city existants

Dans la section suivante nous allons introduire deux grands projets de smart city déjà existant celui d'IBM smarter city et Seoul smart city :

2.1. IBM smarter city

IBM définit une ville plus intelligente (smarter city) en termes de trois caractéristiques principales: instrumenté, interconnecté et intelligente. Ces caractéristiques suggèrent qu'une ville doit être en mesure de détecter et répondre à son environnement. Elle peut observer son environnement avec des capteurs et d'autres mécanismes de collecte de données (instrumenté) et communiquer et partager des informations recueillies avec d'autres entités gouvernementales (interconnectés), et peut aussi utiliser les données pour faire des choix de services optimaux (intelligent). Pour prendre des décisions vraiment intelligents exigeant des données à collecter et analysées à partir d'un groupe diversifié d'organismes régionaux et locaux. [51]



Figure 5.1. Solutions d'IBM smarter city [51]

2.1.1. IBM intelligent water

Le système d'eau intelligent d'IBM optimise les opérations de services publics d'eau et crée des opportunités supplémentaires pour l'innovation et la valeur commerciale. Il offre un aperçu des actifs et des activités d'un service public. Cette solution utilise des analyses avancées et des technologies d'optimisation pour transformer les énormes quantités de données provenant des systèmes de gestion d'actifs, de comptage et de capteurs en informations utiles. Les connaissances recueillies à partir de ces informations peuvent être utiles pour guider les décisions qui impliquent des actifs stratégiques et tactiques, la satisfaction du client, les processus d'entreprise et des modèles d'affaires [52].

La figure suivante montre le diagramme de composants de haut niveau pour les solutions d'eau intelligente IBM.

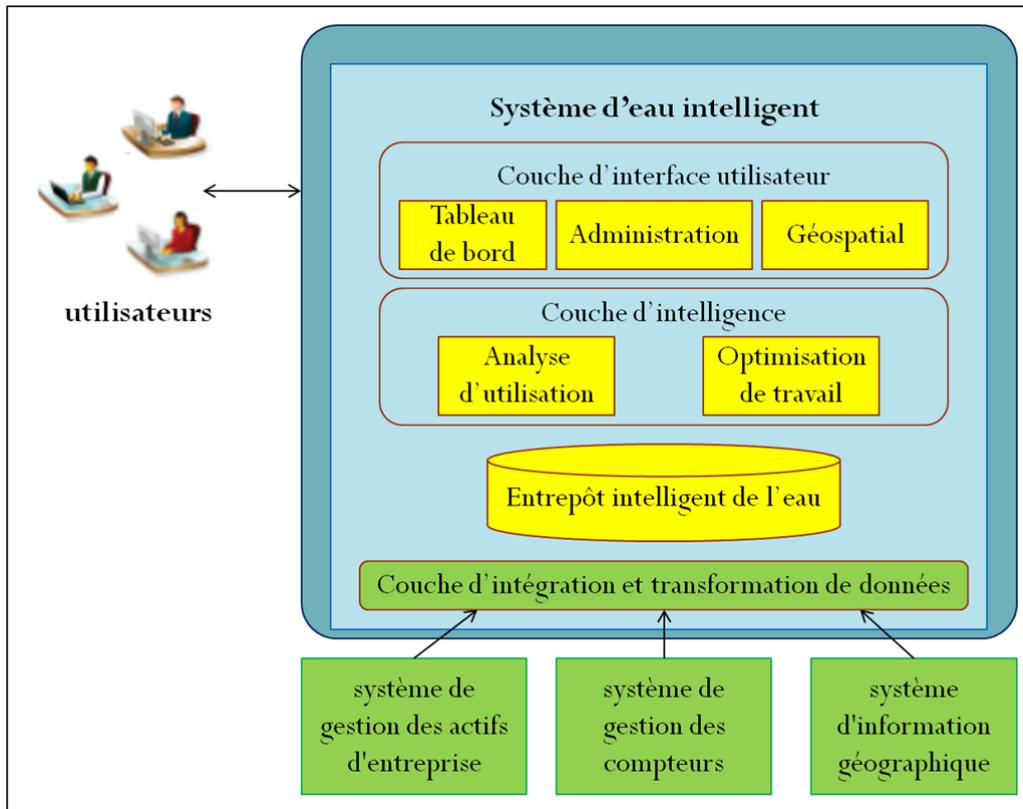


Figure 5.2. Les composants d'IBM intelligent water [52]

Les composants de la solution d'eau intelligente IBM, sont les suivants [52] :

- Centre Intelligent d'opérations

Le centre Intelligent d'opérations constitue la base des services d'eau intelligents d'IBM.

- Les sources de données

Actuellement le système d'eau intelligent accepte les données provenant des systèmes suivants:

- système de gestion des actifs d'entreprise fournit des informations sur les actifs, leur état et les processus de travail.
- système de gestion des compteurs fournit des données en temps réel pour chaque client, ces systèmes recueillent les données d'utilisation d'eau pour chaque client pendant une intervalle de temps donné qui peuvent être utilisées pour mesurer la consommation d'eau résidentielle et commerciale.
- Système d'Information Géographique (SIG) qui intègre, stocke, analyse et affiche des informations géographiques, spatiales (localisation), et topographiques.

- la couche d'intégration et transformation de données

Elle recueille des données à partir des différentes sources de données, synchronise, nettoie et transforme les données en un format standardisé. Cette couche fournit des données sécurisées, fiables et cohérentes pour le stockage dans le référentiel.

- Entrepôt intelligent de l'eau

Les données provenant des diverses sources de données sont collectées et regroupées dans un référentiel de gestion de données centrale, connue sous le nom d'Entrepôt intelligent de l'eau. Il contient un assortiment varié et complet de données de gestion de l'eau actuelles et historiques [52].

- couche Intelligence

La couche d'intelligence comporte l'analyse d'utilisation et l'optimisation du travail

- L'analyse d'utilisation

Avec l'utilisation des modèles analytiques, l'utilitaire peut mieux contrôler l'utilisation de l'eau et la base de clients, ce qui permet une meilleure planification en utilisant les prévisions de demande et des revenus futurs. Les données de consommation d'eau et système SIG de l'utilitaire sert comme entrée des analyses, qui effectuent la segmentation des clients en fonction de critères tels que l'utilisation. Les résultats de segmentation peuvent devenir la base pour d'autres activités, telles que la prévision de la demande, les efforts de conservation et de fixation des taux.

- l'optimisation du travail

Dans le but de répondre à des situations d'urgence, la maintenance préventive et des inspections chaque élément de travail implique l'envoi d'une équipe à partir d'un poste central à l'emplacement des travaux. Dans de nombreux cas, chaque élément de travail se compose d'un ensemble de sous-tâches dépendantes, où chaque sous-tâche pourrait nécessiter des compétences et des équipements spécifiques. Les Analyses peuvent être utilisées pour répartir le travail et optimiser l'utilisation des équipes et du matériel, ce qui minimise les pertes de temps en transit à l'aide des algorithmes d'optimisation avancés [52].

- la couche d'interface utilisateur

Elle comporte un portail Web qui peut être consulté par Internet. Les utilisateurs peuvent avoir accès à un environnement sécurisé qui fournit des informations essentielles à leur rôle de travail. Elle comporte les éléments suivants:

- Des tableaux de bord visuels permettant l'accès aux informations tels que : indicateurs de performance, rapports et résultats d'analyse, infrastructures d'eau représentés dans un contexte géospatial et des ordres de travail superposées sur carte.
- Des services d'administration en charge de processus de vérification
- Un système géospatial intègre fournit les données du SIG affiché dans le tableau de bord.[52]

2.1.2. IBM intelligent transportation v1.0

IBM Transport Intelligent V1.0 fournit de nouveaux renseignements et des idées, avec une grande transparence et une visibilité accrue à travers un réseau de transport entier. Il permet d'analyser le comportement et les événements de trafic à travers les entrées divers du réseau de transport. Il contribue également à optimiser le débit, l'efficacité, le temps de réponse, et l'expérience de voyage. [53]

La figure suivante montre le diagramme de contexte du système de gestion de trafic globale avec la solution IBM intelligent transporattion v1.0 :

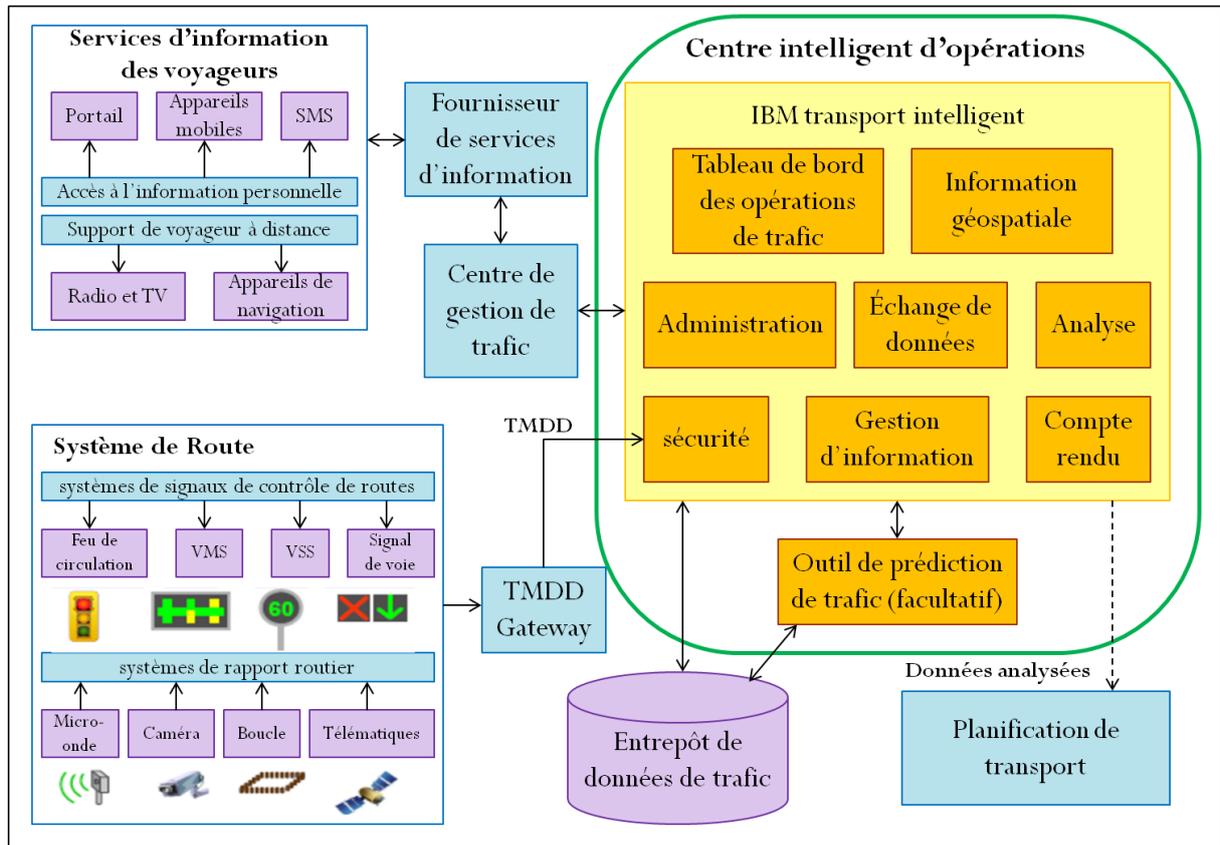


Figure 5.3. Système de gestion de trafic avec IBM intelligent transportation [53]

Le système de transport intelligent d'IBM comprend les composants suivant [53] :

- Systèmes de routes

Ces systèmes comprennent les catégories suivantes:

- systèmes de signaux de contrôle de routes

Les feux de circulation, panneaux à messages variables (PMV), panneaux de vitesse variable (VSSs) et les signaux de voie sont des systèmes de signalisation de contrôle routier. Ils sont utilisés pour gérer les flux de trafic et informer les voyageurs de l'évolution des conditions routières ou donner des instructions afin qu'ils puissent prendre les mesures appropriées.

- systèmes de rapport routier

Matériel routier, y compris les détecteurs de boucle, des capteurs à micro-ondes et des caméras vidéo fournissent des données de trafic en temps réel. Ces données comprennent le volume du trafic, la vitesse moyenne, et les conditions de la chaussée. Avec des données provenant des capteurs routiers fixes, les véhicules sur la route sont également exploités en tant que source de données, en particulier en ce qui concerne le trafic, les files d'attente, et les données de congestion connexes.

- Centre de gestion de trafic

Un TMC prend en charge les systèmes de routes et d'autoroutes. Des informations en temps réel sont collectées à partir des systèmes d'information de la chaussée, le personnel de la sécurité publique, et d'autres rapporteurs de circulation. Avec ces informations, l'équipe de TMC et les gestionnaires du trafic peuvent prendre des décisions et des mesures appropriées. Ils allouent des ressources, comme les travailleurs de la route et le personnel de maintenance, et interagissent avec les responsables de la sécurité publique dans le traitement des incidents spécifiques. Ils peuvent aussi donner des directives aux systèmes de signalisation de contrôle du chemin de roulement et de fournir des informations aux services d'information voyageurs[53].

- La planification de transports

L'équipe de planification de transport utilise des données de trafic actuelles et historiques pour évaluer et concevoir les systèmes de transport (tels que les rues, les autoroutes, les chemins ou voies cyclables, et les lignes de transport en commun). En outre, l'équipe de planification des transports analyse des données d'incident et génère des plans de réponse aux incidents pour une utilisation par le TMC.

- Fournisseur de services d'information

Le fournisseur de services d'information est le mécanisme qui communique des informations de trafic entre les différents services d'information voyageurs et TMC.

- Service d'information des voyageurs

Un service d'information de voyageurs informe les voyageurs sur les préoccupations telles que les incidents de la route, les travaux routiers et les conditions de conduite. Cette information est disponible sur les sites Web publics ou des portails et des dispositifs (tels que la télévision, la radio, mobile et appareils de navigation) ou est transmis en utilisant un service de messages courts (SMS).

- Dictionnaire de données de gestion du trafic (TMDD) Gateway

Les données de trafic sont générées par des sources multiples qui peuvent être fournis par plusieurs fournisseurs. Les données de trafic (dans différents formats) sont nettoyées et normalisées dans le format TMDD par la Gateway [52].

La solution IBM de transport intelligent comprend les fonctions suivantes [53]:

- Tableau de bord des opérations de trafic

Le tableau de bord de la circulation routière fournit des données relatives à la circulation et les principaux résultats de l'analyse dans le Centre Intelligent d'Operations. Cette plateforme de gestion du trafic comprend des services, tels que la surveillance des informations de trafic, la gestion des événements de circulation, et l'analyse de trafic pour les opérateurs et le public.

- L'information géospatiale

Il prend en charge le stockage et la présentation du système d'information géospatiale (SIG) pour des affichages graphiques dynamiques dans le tableau de bord tels que: cartes basées sur les SIG pour visualiser le réseau routier, capture manuelle et automatique des événements de circulation et incidents et icônes de périphériques.

- Protocoles d'échange de données à base de standards

L'échange de données prend des données structurées sous un schéma source et crée une instance dans le schéma cible avec le plus de précision possible. Il transforme les données de trafic multi-sources dans un modèle de données de trafic uniforme.

- L'analyse

L'analyse comporte des outils analytiques qui utilisent des données spécifiques au trafic et liés à la circulation à partir de diverses sources comme entrée du traitement. Les résultats de cette analyse fournissent au personnel et d'autres responsables de la ville des connaissances et informations clés afin qu'ils puissent prendre des décisions éclairées.

- Compte-rendu

Les rapports pour les responsables de la ville, la planification des transports, et le personnel peuvent être générés en utilisant des données de trafic en temps réel et historiques. Les rapports peuvent être créés dynamiquement et personnalisés en utilisant des requêtes et des filtres sur les données de trafic [53].

- Gestion de l'information

La gestion de l'information basée sur des normes recueille, gère, et stocke les données relatives au trafic pour une utilisation dans les rapports, le partage et l'analyse. En raison des

quantités massives de données de trafic générées, Elle gère les données actuelles et archivées pour une utilisation au besoin.

- L'Administration

L'Administration permet la définition des groupes d'utilisateurs, la personnalisation de l'interface visuelle de l'utilisateur, et l'attribution des rôles et accès aux différentes fonctionnalités et des données dans la solution de transport intelligent IBM.

- la sécurité

La sécurité inclut les fonctionnalités suivantes : l'authentification, l'autorisation, le niveau de sécurité et des données, les Applications et le niveau de sécurité de services, la protection de la vie privée des voyageurs et des informations spécifiques au véhicule. [53]

2.1.3. Solutions d'IBM smarter City sur Cloud (SaaS)

IBM estime que les solutions IBM Smarter City sur Cloud aideront les gouvernements à économiser les coûts et de l'habilité opérationnelle en combinant les technologies de Cloud Computing, les services améliorés de citoyens et l'efficacité de gestion de trafic et d'eau des.

IBM a le support technologique pour transformer un gouvernement d'un centre de données traditionnel en un fournisseur de services de Cloud, en créant ainsi un environnement de services Cloud pour le gouvernement, offrant ainsi une variété de services rapides et à moindre coût et fournir des produits sécurisés, des portails en service libre où les clients obtiennent, de gèrent et de payent pour leurs services [51].

En fournissant des solutions sur le Cloud et de servir en tant que fournisseur des services de cloud, les grands gouvernements peuvent [51]:

- Aider les petites villes et les communautés dans la réalisation des avantages avec des services faciles à mettre en œuvre.
- Utiliser efficacement les informations supplémentaires fournies par les multiples régions pour améliorer la prise de décision et les services aux citoyens.
- La gestion des transports et d'eau de façon unique pour aider à attirer et à retenir de nouveaux commerces pour la communauté.
- Connectez le gouvernement avec les entreprises.

2.2. Seoul Smart City

Séoul Smart City est un projet centrée plus sur les citoyens, il vise à mettre en œuvre autant de technologies intelligentes que possible, mais aussi de créer une relation plus collaborative entre la ville et ses citoyens.

Comme toutes les villes intelligentes smart Séoul définit ses propriétés par les trois traits suivants [54]:

- Infrastructure des TIC: la sécurisation de l'infrastructure des TIC de la prochaine génération est essentielle pour le succès des services de smart city émergents. Les efforts visant à développer l'infrastructure des TIC doivent anticiper les futures demandes de services, plutôt que de répondre uniquement aux personnes les plus apparents.
- Framework intégré de gestion de la ville: Un «Framework intégré de gestion de ville» bien défini est essentiel. Les nombreux sous-systèmes intégrés, les méta-systèmes et individuels d'une ville intelligente ne travailleront en harmonie que par le respect strict des normes communes.
- Utilisateurs intelligents: Les TIC sont des outils permettant à améliorer une ville intelligente, mais ne sont d'aucune utilité sans les utilisateurs de smart-tech capables d'interagir avec des services intelligents. Accroître l'accès aux appareils intelligents et l'éducation sur leur utilisation, à travers les niveaux de revenu et les groupes d'âge, doit rester l'une des plus grandes priorités d'une ville intelligente. [54]

2.2.1. U-Seoul Net

L'établissement d'un réseau de communication dédié aux services intelligents a été une priorité pour Smart Séoul, Avec le réseau u-Séoul Net les citoyens ont accès aux services administratifs à tout moment, partout.

u-Séoul net est divisé en trois communications sous-réseaux: un réseau Wi-Fi utilisée pour servir des fonctions administratives; un réseau de télévision en circuit fermé permettant l'échange de données vidéo générées par des installations de vidéosurveillance de Séoul; et le réseau u-service, qui relie les sites Web de tous les bureaux publics sous le gouvernement métropolitain de Séoul, en permettant aux citoyens de contourner les réseaux des fournisseurs de services Internet, et à la place accéder à u-Séoul net pour obtenir des informations gratuites sur les services de la ville. [54]

2.2.2. Open Governance 2.0

A cause de L'importance du partage de l'information du gouvernement avec le secteur privé, L'information publique mise à la disposition du secteur privé lui permet de produire des solutions innovantes aux besoins publics.

Reconnaissant la valeur socio-économique de l'information publique, Séoul vise à rendre toutes les informations administratives à la disposition de ses citoyens. La stratégie de Séoul dans Open Governance 2.0 encourage la gouvernance urbaine transparente et une communication ouverte entre le gouvernement de la ville et de ses citoyens. Informations Open Square, ajoutée au site Web de Séoul, est un mécanisme par lequel Séoul révèle des documents administratifs, y compris les travaux en cours. Les citoyens et le secteur privé sont par conséquent encouragés à utiliser des informations administratives de la ville pour découvrir de nouvelles possibilités d'emploi et d'affaires. [54]

2.2.3. U-Seoul Safety Service

Le service de sécurité u-Séoul utilise des services de localisation et technologies de vidéosurveillance pour informer les autorités et les membres de la famille des situations d'urgence impliquant des enfants, les handicapés, les personnes âgées et les personnes souffrant de la maladie d'Alzheimer . Séoul a mis au point un dispositif intelligent dédié à cet effet et lorsque son titulaire quitte une zone de sécurité désignée ou pousse le bouton d'urgence, une alerte d'urgence est envoyé aux gardiens, la police, les services d'incendie et de centres de contrôle de vidéosurveillance.

Pour utiliser le service de sécurité u-Séoul, les citoyens sont tenus d'enregistrer avec des supports mobiles spécialement conçus à cet effet.

“U-Children Safety System” fournit des «zones de sécurité des enfants» en utilisant un réseau sans fil d'infrastructure maillée avec multiple entrée et multiples sorties (MIMO). Le réseaux de vidéosurveillance en temps réel et les périphériques intelligents pour enfants permettent à Séoul de faire usage des réseaux sans fil pour localiser les enfants disparus le plus rapidement possible. [54]

2.3. Synthèse des projets de smart city

Dans ce chapitre nous avons vu deux exemple de ville intelligente chacune possédant sa propre méthode :

- IBM smarter city : son fonctionnement est basé fortement sur les nouvelles technologies (capteurs, senseurs, caméra) d'où les données sont collectés et analysés afin de soutenir le processus de prise de décision, cette approche fournit aux citoyens un ensemble de services. De ce fait, son avantage réside dans l'indépendance des citoyens ce qui permet une gestion totalement automatique de la ville et qui permet de fournir des services en temps réel aux citoyens. Cependant sa forte dépendance sur les TICs représente un inconvénient majeur car la participation des citoyens semble négligée.
- Smart Seoul : est basé sur le concept de « smart people », elle considère les citoyens de la ville capable d'utiliser les nouvelles technologies c'est pour cela que cette approche permet la participation des citoyens dans le processus de prise de décision, son but est de créer une communication continue entre le gouvernement et les citoyens et une transparence grâce au partage d'informations. L'avantage de cette approche c'est le fait qu'elle permet aux citoyens de participer à la gouvernance de leur ville, c'est-à-dire la transparence gouvernementale. Et l'inconvénient c'est la dépendance de smart people ; elle se base totalement sur la capacité et la connaissance d'utilisation des appareils intelligents par les citoyens.

3. Les Travaux de Recherche

L'application de solutions de recommandation pour l'administration de la ville intelligente a été un nouveau domaine pour les systèmes de recommandation avec de nouveaux défis et opportunités de recherche, c'est pour cela que le travail dans ce domaine a été relativement faible, Afin de bénéficier d'autres travaux, nous avons choisi d'étudier deux exemples qui ont utilisé une approche de recommandation pour fournir un ensemble de services électroniques gouvernementaux :

3.1. Recommandations proactives et réactives

Ayachi et al. [55] ont proposé un nouveau framework de E-gouvernement pour la personnalisation des services en utilisant deux moteurs de recommandation différents, à savoir la recommandation réactive et la recommandation proactive; la partie réactive utilise une technique basée sur les contraintes pour fournir des recommandations en fonction des besoins de l'utilisateur, du profil et des informations contextuelles, alors que la partie proactive fournit des recommandations prévisionnelles à partir du profil du citoyen dans les réseaux sociaux. Le fonctionnement de l'approche proposée est présentée par la figure suivante[55].

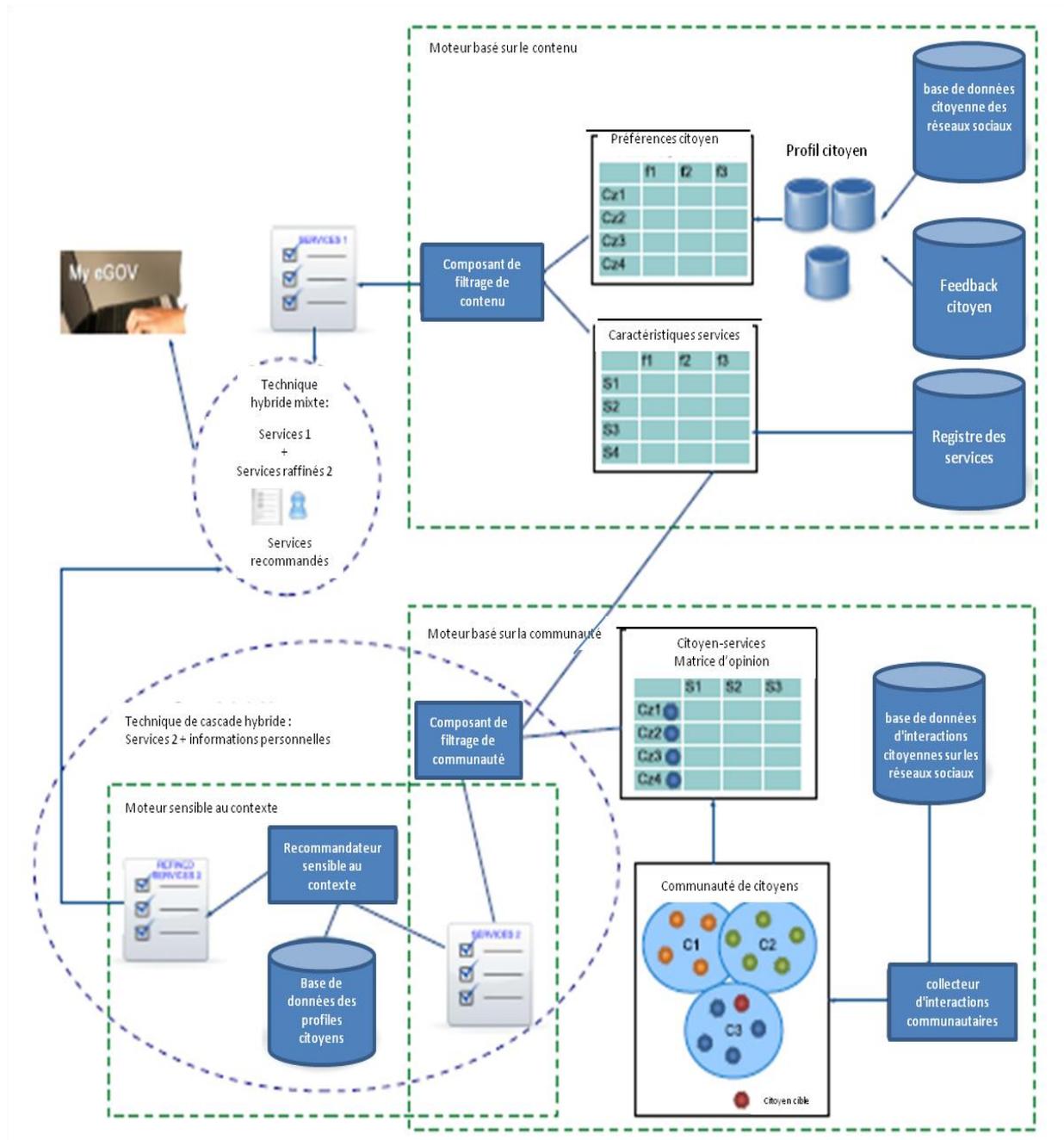


Figure 5.4. Fonctionnement de l'approche proactive et réactive [55].

3.1.1. Moteur de recommandation réactif

Le moteur réactif fournit des services en fonction des besoins du citoyen formulés par le biais d'un ensemble de questions et de réponses. Il représente principalement un système de recommandation basé sur la connaissance avec une technique de contrainte. Cette technique établit une relation entre les besoins des citoyens et les fonctionnalités des services pour fournir des recommandations personnalisées adaptées aux profils des citoyens, un système de recommandation tenant compte du contexte est utilisé. En effet, il propose des services

adéquats en fonction du profil de l'utilisateur (par exemple, ses informations démographiques, sa situation sociale, ses informations temporelles et géographiques) [55].

Une méta-technique hybride de recommandation est proposée dans laquelle le moteur basé sur les contraintes ne filtrera pas les services de la base de données du registre de services, mais plutôt de la liste sélectionnée de services fournis par le moteur sensible au contexte.

Une telle technique d'hybridation permet au recommandataire basé sur les contraintes d'exploiter le résultat du recommandataire sensible au contexte afin de fournir des recommandations personnalisées pertinentes correspondant au profil du citoyen [55].

3.1.2. Moteur de recommandation proactif

Le moteur proactif recommande des services en fonction du profil du citoyen construit à partir de la base de données des réseaux sociaux et du feedback des citoyens. Pour garantir la correspondance entre les fonctionnalités des services et les préférences des citoyens, un recommandataire basé sur le contenu, qui commence par créer un profil de citoyen dans son noyau, sera utilisé [55].

Néanmoins, ce type de recommandation souffre du problème de démarrage à froid ; lorsque le système ne dispose pas de suffisamment d'informations sur les préférences des citoyens dans le passé et ne peut donc pas recommander de services. Pour surmonter ce problème, un système de recommandation basé sur la communauté sera utilisé, suggérant des services appréciés de la communauté des citoyens et affinés par un recommandataire sensible au contexte.

La technique hybride proposée repose sur trois composants: un moteur basé sur le contenu, un moteur basé sur la communauté et un moteur basé sur le contexte. La liste finale des recommandations proactives personnalisées provient de la fusion des services issus du moteur de contenu et de ceux produits par les moteurs de communauté et de contexte. Une hybridation en cascade est utilisée entre les deux moteurs pour affiner les recommandations produites par la communauté. La tâche de fusion utilise une technique d'hybridation mixte qui expose les recommandations côte à côte dans une liste combinée [55].

3.1.3. Synthèse

L'approche générale de l'ensemble du système suit une approche de filtrage collaboratif, qui consiste à calculer les similarités entre les citoyens afin de les classer dans des groupes en fonction de leurs préférences. Le filtrage collaboratif présente deux inconvénients majeurs: le coût en temps d'exécution; à chaque recommandation, la base de données doit être entièrement consultée, Le deuxième inconvénient est que ce type d'approches souffre du problème de démarrage à froid lorsque l'état initial du système de recommandation ne contient pas suffisamment d'informations de l'utilisateur pour générer des recommandations susceptibles de provoquer une panne du système.

Bien que cette approche soit réellement efficace et présente de nombreux avantages à exploiter les réseaux sociaux dans les recommandations, elle n'est pas adaptée à notre cas car elle évalue les citoyens selon leurs similitudes, ce qui ne sert pas à notre objectif d'amélioration de la qualité des services en fonction des besoins des citoyens, c'est pourquoi nous avons besoin d'une approche permettant l'évaluation entre les services et les citoyens, c'est-à-dire une approche basée sur le contenu.

3.2. ASSO the Assistant to Citizens for Service Selection

L'assistant des citoyens pour la sélection de services (ASSO) est un outil développé pour exploiter et fournir les avantages des services groupés en paquets. Déployé en tant qu'application Web accessible par plusieurs appareils, ASSO présente un scénario dans lequel un citoyen-utilisateur ayant besoin de changer d'adresse résidentielle pour un nouveau lieu, recommande un ensemble de services obligatoires par la loi et un ensemble d'autres services pertinents (par exemple, un service qui soutient la recherche d'une nouvelle école pour les enfants du citoyen) [56].

ASSO fonctionne avec un assistant intuitif, étape par étape. L'interaction citoyen-utilisateur commence par la formulation d'une demande (par exemple, je dois changer d'adresse résidentielle) et se termine par une liste classée des fournisseurs de services lui permettant de demander l'exécution du service.

ASSO exploite le réseau de relations pour répondre aux questions des citoyens: Qu'est-ce que mon gouvernement me demande de remplir lorsque je change d'adresse résidentielle? Quels autres services puis-je utiliser si je suis marié ou si j'ai des enfants? Quel est le meilleur fournisseur de services pour mes besoins?

L'interaction avec ASSO est composé en 6 étapes: 1) demande du citoyen, 2) sélection d'un service abstrait, 3) profil du citoyen, 4) composition d'un paquet, 5) résumé du paquet, et 6) classement des services dans le paquet lorsque plusieurs fournisseurs sont disponibles [56].

3.2.1. Profil du citoyen

À cette étape, le citoyen-utilisateur a effectué une demande dans la page d'accueil décrivant son événement: «Je dois changer mon adresse résidentielle », et c'est tout ce que le système prend en entrée. À la deuxième étape, le citoyen-utilisateur sélectionne un élément parmi un nombre de services abstraits dans la collection correspondant à sa demande.

Le système navigue entre l'événement de la vie sélectionné par l'utilisateur et d'autres éléments pour obtenir des segments d'utilisateur de haut niveau (appelés catégories) qui concernent le citoyen-utilisateur: couple marié et / ou parent. Avec ces informations ASSO entre dans l'étape de raffinement du profil citoyen, en formulant un questionnaire court adapté avec la catégorie à laquelle appartient l'utilisateur, qui a pour but de rassembler des détails importants, par exemple pour déterminer si le citoyen est propriétaire de sa nouvelle maison, des enfants, un compte en banque, un travail, un permis de conduire, etc [56].

Les résultats du questionnaire permettent d'extraire les services obligatoires et les autres services pertinents. Dans l'étape du profil du citoyen, le système exploite déjà les liens entre les services et les événements de la vie à travers des catégories d'utilisateurs, et non pas à des services isolés référant les besoins ponctuels des utilisateurs.

Pour affiner la catégorie auquel le citoyen-utilisateur appartient, ce qui a été déduit à la deuxième étape, ASSO ne demande pas à l'utilisateur de parcourir une longue liste de catégories auxquelles il peut appartenir, mais il collecte ces informations à partir de questions intuitives, par exemple: est-ce que vous possédez l'ancienne ou la nouvelle propriété? ". Le processus suivant utilise les réponses positives pour déduire la catégorie d'utilisateurs à laquelle appartient l'utilisateur; ces catégories sont liées à une liste de services pertinents, préparant ainsi la composition du paquet [56].

3.2.2. Composition d'un paquet de services

A partir de l'étape précédente, un profil du citoyen-utilisateur est obtenu à partir d'un scénario de «changement d'adresse résidentielle pour plusieurs personnes», c'est-à-dire l'événement de la vie. La connaissance de la loi administrative, associée aux catégories

d'utilisateurs et aux services capturés dans ASSO, permettent de déterminer quels services abstraits sont obligatoires, ainsi que ceux qui intéressent le citoyen en fonction de son profil.

L'interaction avec l'utilisateur est maintenue au moyen d'éléments de colorés pour les services obligatoires prévus par la loi, que les utilisateurs ne peuvent pas éliminer de la composition du paquet (par exemple, le contrat d'approvisionnement en eau à la nouvelle adresse), et d'autres services facultatives que ASSO considère pertinents pour le profil de l'utilisateur (par exemple, lancement d'un nouveau contrat Internet). Par conséquent, l'étape de composition des paquets évite au citoyen de connaître les contraintes imposées par la loi (elle fournit en fait un environnement d'auto-apprentissage), tout en lui demandant de confirmer les services optionnels sélectionnés par ASSO qui interprètent le scénario et le profilage de l'utilisateur avec une liste d'autres services [56].

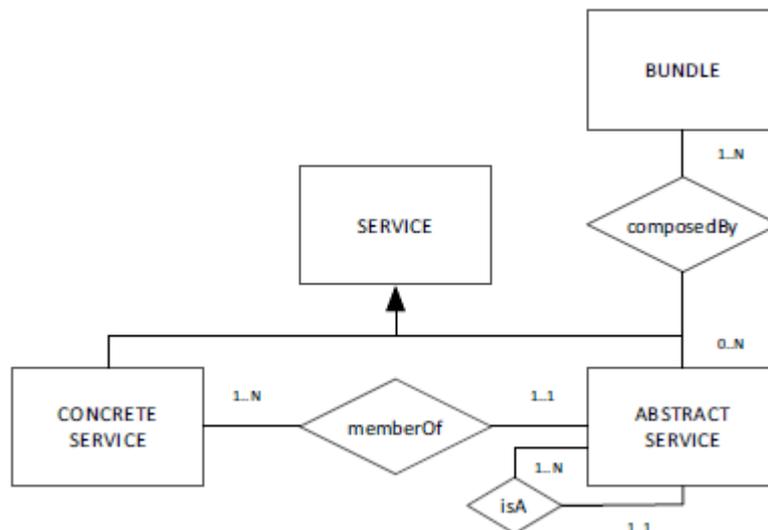


Figure 5.5. Composition d'un paquet de services [56].

3.2.3. Evaluation et classement des fournisseurs de services

Alors que l'utilisateur compose et confirme son ensemble de services abstraits, ASSO entre dans l'étape dans laquelle il liste les fournisseurs (concrets) des services sélectionnés. Puisque les fournisseurs de services peuvent être des administrations publiques ou des entités privées, l'utilisateur peut souvent choisir parmi d'autres fournisseurs alternatifs, dans un scénario concurrentiel [56].

ASSO inclut un support pour classer les fournisseurs de services en fonction des préférences exprimées par les utilisateurs. En demandant à l'utilisateur d'exprimer ses

préférences concernant les propriétés du service, telles que le prix, le temps, ...etc. ASSO peut classer les fournisseurs qui correspondent le mieux aux préférences de l'utilisateur. En utilisant un formulaire qui enregistre un poids de pertinence que l'utilisateur associe à chaque propriété du service et évalue un score correspondant pour chaque fournisseur et renvoie un classement [56].

3.2.4. Synthèse

Baldassarre et al. [56] a proposé un système qui assiste les citoyens pour la sélection de services (ASSO). Il s'agit d'un système basé sur la connaissance qui fournit des recommandations de services en fonction des besoins du citoyen en utilisant des liens sémantiques pour la description des services. Cette approche est basée sur le profil et la demande du citoyen qui est exprimée en termes d'événements de la vie, les liens sémantiques entre événements de la vie et services constituent un avantage certain de l'approche proposée. Toutefois, l'inclusion des préférences des citoyens et des informations contextuelles qui manquent dans ce système reste nécessaire pour offrir de meilleures recommandations.

4. Conclusion

Dans ce chapitre nous avons réalisé une étude des projets déjà existant en plus des travaux de recherche dans le domaine de smart city. L'étude des deux projets nous a permis de constater l'existence de deux approches différentes de gestion de smart city celle qui est centrée sur les citoyens et l'autre qui est centrée sur les TICs, pour cela dans notre travail de recherche nous allons adoptée une approche centrée sur les TICs mais aussi qui prend en charge les citoyens et leur contribution dans la gestion de la ville.

L'étude des deux autres travaux de recherches ci-dessus nous a orientés vers les approches de recommandations et a mis en évidence leur rôle important dans le contexte de la ville intelligente et la manière dont cela pourrait influencer sur la productivité de tels systèmes.

**CHAPITRE 04 : MODELISATION DE
L'APPROCHE PROPOSEE**

1. Introduction

Le chapitre suivant est dédié à la modélisation de l'approche proposée qui représente une étape essentielle afin de bien concevoir notre système de l'administration d'une smart city, nous allons détailler le fonctionnement de l'approche ainsi que les différentes étapes menant à la réalisation de l'objectif, par la suite nous allons décrire la conception de notre système et la modélisation des différentes entités qui le composent.

2. Fonctionnement et objectif général

L'objectif de notre travail de recherche consiste à réaliser un système de l'administration de smart city qui a comme objectif principale de fournir des services adaptés aux besoins des citoyens, Pour cela nous avons choisi de développer un système de recommandation sensible au contexte, cette approche qui représente un moyen très efficace de personnalisation fait face à de nombreux problèmes parmi lesquels le problème de scalabilité de ce fait, nous avons opté pour l'approche back-end as a service afin de surmonter ce problème.

Notre système présente un scénario dans lequel le citoyen ayant besoin d'effectuer une démarche administrative quelconque, reçoit comme réponse des recommandations regroupant un ensemble de services obligatoires et un ensemble d'autres services qui sont jugés pertinents. L'interaction avec le citoyen commence par la formulation d'une demande (à partir d'une liste de démarches administratives proposées) et se termine par une liste classée de services lui permettant la réalisation de sa requête.

Les étapes de la modélisation de notre système peuvent être résumées comme suit :

- La définition du problème à résoudre
- La modélisation du citoyen
- La présentation de l'approche de recommandation
- La conception du système

3. Formalisation du problème

Formellement, le problème traité par un système de recommandation peut être décrit comme suit [35]:

soit $\phi = \{S, C\}$ l'univers des informations de base manipulées par le modèle, tels que :

- $S = \cup_{j=1}^n s_j$ est l'ensemble des services de la collection et n le nombre total des services.

-Le profil est la structure informationnelle représentant le citoyen à travers trois dimensions corrélées, évoluant dans le temps, noté $C = (D_s, P_s, C_s)$. Tels que D_s représente les informations démographiques du citoyen, P_s les préférences des services du citoyen jusqu'à l'instant s et C_s représente le contexte du citoyen à l'instant s .

Pour chaque requête soumise, le citoyen, les requêtes et les services utilisés sont collectés comme un triplet d'informations concurrentes dans le profil. Où le citoyen $c \in C \{c_1, c_2, \dots, c_n\}$, la requête $q \in Q \{q_1, q_2, \dots, q_m\}$ et les services associés $s \in S \{s_1, s_2, \dots, s_l\}$.

Le problème qui se pose est : Etant donné les services déjà sélectionnés, quelle est la pertinence des services qui devraient être sélectionnés en réponse à cette requête.

Étant donné un ensemble T contenant les données collectées représentées par des triplets (c, q, s) , et une fonction de mapping $f : C * Q \rightarrow S$ doit être apprise par le modèle. La résolution de ce problème est donnée en appliquant l'approche de recommandation.

Il est à noter que lors d'un scénario de recherche, une observation est un triplet (c, q, s) correspondant à l'événement qu'un citoyen c soumet une requête q , et sélectionne un service s à partir des résultats retournés. La fonction f permet de sélectionner pour chaque paire d'information "*citoyen, requête*" notée (c, q) contenue dans le profil, une liste de nouveau services ou chaque service s' n'ayant pas été déjà collectée dans le profil (c-à-d, que le triplet $(c, q, s') \notin T$), tel que le service s' sélectionnée est celui ayant une grande valeur d'évaluation.

4. La modélisation du citoyen

La modélisation du citoyen est au cœur de la mise en œuvre du processus de recommandations contextuelle. Elle consiste à décrire les caractéristiques informationnelles

des citoyens à travers un modèle de profil. Dans le cadre d'un système de recommandation, la modélisation de ce dernier est un processus caractérisé par trois phases [35]:

1. La première phase regroupe des techniques d'acquisition des données du citoyen.
2. La deuxième représente la définition et la construction du profil citoyen qui peut être effectué selon différentes approches pour organiser les informations collectées selon la structure représentative définie.
3. La troisième phase concerne l'évolution du profil au cours du temps qui nécessite la mise à jour du contenu informationnel du profil.

4.1. L'Acquisition des données

Cette phase consiste à collecter les informations pertinentes pour instancier le profil du citoyen. Le processus d'acquisition des données du citoyen implique différentes formes de diagnostic ou d'évaluation. Ce processus permet de collecter ces informations soit directement à partir de l'utilisateur ou indirectement à partir de l'application. Ce processus d'acquisition peut être explicite et/ou implicite [35] :

- l'approche explicite : Cette technique constitue une approche simple pour obtenir des informations sur l'utilisateur. On interroge directement l'utilisateur ou on lui demande de remplir des formulaires pour collecter les données personnelles et démographiques tels que sa date de naissance, son statut marital et son activité professionnelle, qui seront stockées pour créer le profil utilisateur.
- l'approche implicite : L'acquisition implicite consiste à collecter les données de l'utilisateur, en observant son comportement et en examinant ses activités pour inférer son profil. Cette approche ne nécessite aucune implication directe de l'utilisateur lors de sa recherche, en effet, toute interaction de l'utilisateur avec le système est considérée comme une estimation de ses intérêts.

En étudiant la nature des données relatives au citoyen nous avons constaté le besoin d'utiliser les deux approches ; à notre avis, un profil contient des informations personnelles nécessaire à identifier le citoyen et ne peuvent être acquises que d'une manière explicite à travers par exemple des formulaires. Cependant d'autres informations telles que les informations contextuelles et comportementales doivent être extraites implicitement lors de l'utilisation du système sans l'intervention directe du citoyen.

4.2. Construction du profil citoyen

La bonne construction du profil du citoyen en exploitant ses données informationnelles permet l'adaptation des résultats selon ces besoins en gérant efficacement les meilleures recommandations. Pour cela, Il existe principalement trois types de représentation de profil [57]:

- la représentation ensembliste : le profil y est généralement formalisé comme des vecteurs de termes pondérés ou classes de vecteurs.
- la représentation sémantique : la représentation du profil met en évidence les relations sémantiques entre les informations le contenant. La représentation est essentiellement basée sur l'utilisation d'ontologies ou réseaux sémantiques probabilistes.
- la représentation multidimensionnelle : le profil est structuré selon un ensemble de dimensions, représentées selon divers formalismes [57].

En plus des informations factuelles décrivant le citoyen, la notion de profil est aussi souvent liée à celle de préférences et de contexte [58]:

- Un profil utilisateur représente l'ensemble des informations décrivant l'utilisateur.
- Un contexte représente les données décrivant l'environnement d'interaction entre un utilisateur et un système.
- Une préférence est une expression permettant de hiérarchiser l'importance des informations dans un profil ou un contexte [58].

En effet, les préférences du citoyen représentent un aspect important dans la description du citoyen et peuvent changer en fonction du contexte dans lequel il évolue.

L'objectif de construction réside à proposer un modèle générique capable de décrire le citoyen, le contexte et les préférences par un ensemble d'attributs, ou plutôt une liste de concepts pouvant être spécialisés, affinés et instanciés aux besoins.

Afin d'exploiter la classification des informations relative au citoyen selon les notions de profil, contexte et préférences dans la construction du profil, nous avons abordé la représentation multidimensionnelle qui est la plus appropriée a ce besoin afin de diviser le profil en dimension relatives à ces trois notions; de ce fait, nous avons identifié trois

principales dimensions qui constituent le profil citoyen: dimension démographique, dimension contextuelle et dimension des préférences, chacune est composée d'un ensemble d'attributs associés à leurs valeurs et éventuellement organisés en entités.

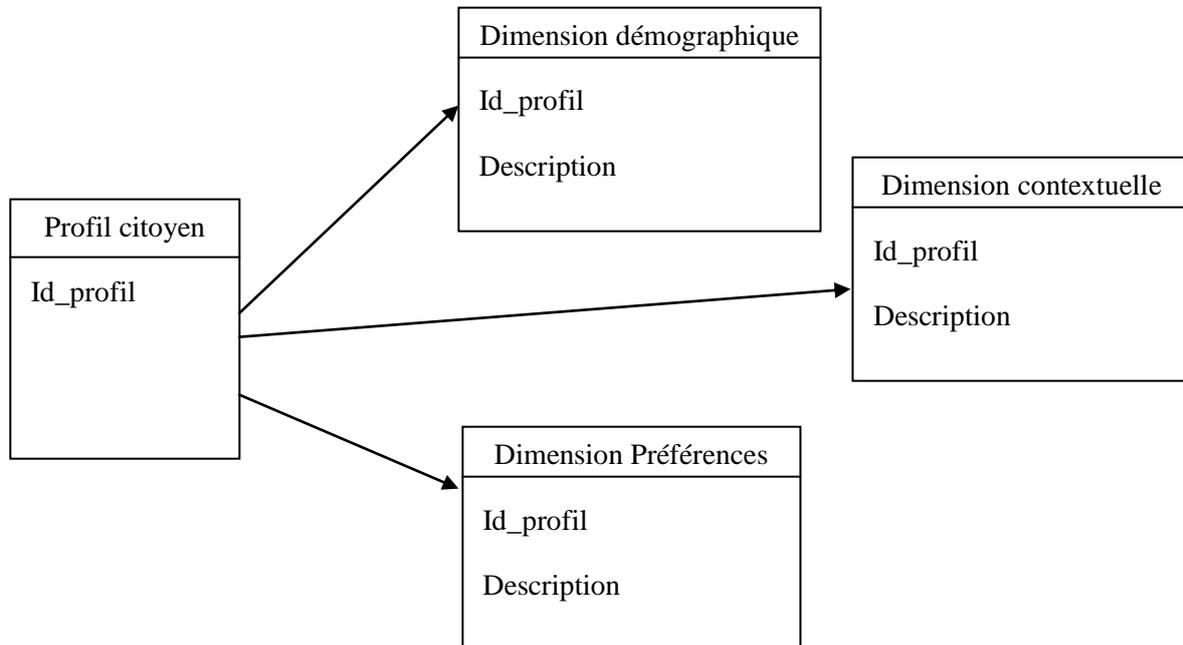


Figure 6.1. Schéma décrivant les dimensions d'un profile citoyen

4.2.1. Dimension démographique

Cette dimension représente la partie statique du profil obtenu de manière explicite par le citoyen à l'aide d'un formulaire lors de la première inscription. Elle regroupe toutes les informations qui décrivent le citoyen et ne dépendent pas du système, classés en deux catégories de données : l'identité du citoyen et les données personnelles. La première catégorie (identité) est composée d'un ensemble d'attributs d'identification de l'utilisateur. Des exemples de tels attributs sont le nom et le prénom du citoyen, son adresse, etc. La seconde catégorie de données personnelles contient un ensemble d'attributs comme par exemple la date de naissance du citoyen, son genre, etc.

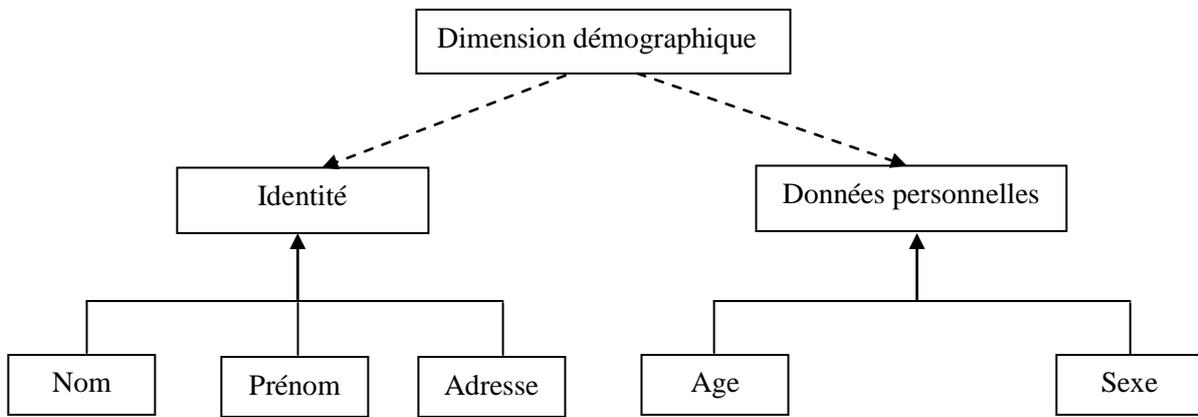


Figure 6.2. Raffinement de la dimension démographique

4.2.2. Dimension contextuelle

Cette dimension représente la partie dynamique du profil, elle contient toutes les données permettant de décrire le contexte du citoyen elle concerne principalement les deux entités : l'entité spatiale permettant de décrire la localisation géographique actuelle du citoyen et l'entité temporelle représentant le temps de la requête, chacune d'eux représente une entité composée ; la localisation géographique contient deux attributs permettant de tracer les coordonnées de latitude et longitude, ainsi que l'entité du temps est composée de trois attributs représentant l'heure, les minutes et les secondes.

A cause de la nature délicate de ces informations et les variations continue en temps réels de ses valeurs, une mise à jours est effectué à chaque requête afin d'éviter de mettre à jours les informations qui seront inutilisables et optimiser le nombre des mises à jours pas forcément nécessaires.

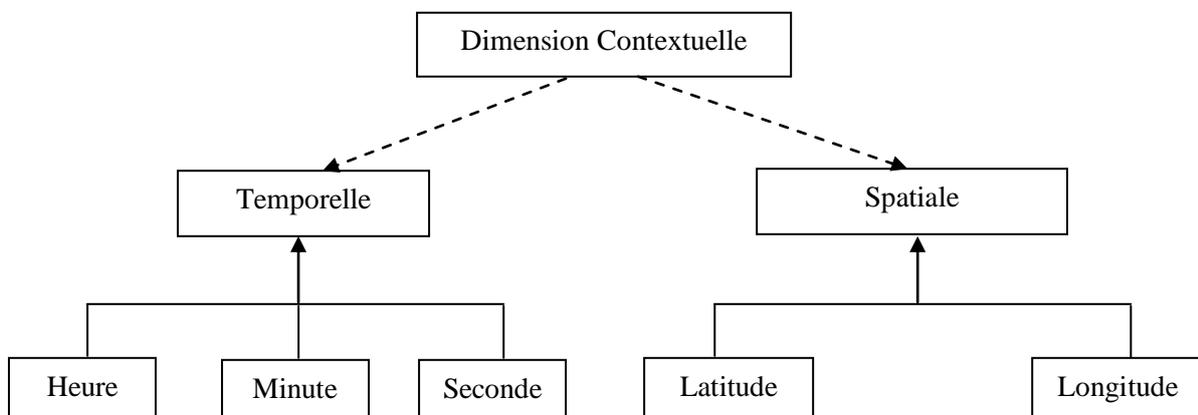


Figure 6.3. Raffinement de la dimension contextuelle

4.2.3. Dimension de préférences

Dans la plupart des systèmes d'informations personnalisés on distingue trois types de préférences [59] :

-Les **préférences de résultat** concernent le contenu lorsque l'utilisateur peut choisir des résultats parmi ceux obtenus après l'exécution des fonctionnalités et le format préféré des résultats des fonctionnalités (par exemple, de la vidéo, du texte, ou des images).

-Les **préférences d'affichage** concernent la manière dont l'utilisateur souhaite que l'information soit affichée. Ceci recouvre d'un côté l'apparence, le style, les polices de caractères, etc., et, de l'autre, les caractéristiques des formats d'affichage (c'est-à-dire, les caractéristiques de la vidéo, des images, ou du son).

-Les **préférences d'activité** concernent les activités qu'un utilisateur souhaite et peut accomplir dans le système. Du côté du système, une activité consiste en un ensemble de fonctionnalités c'est à dire, l'invocation d'un service, qui doivent être exécutées afin d'accomplir cette activité. Une fonctionnalité peut être liée à d'autres fonctionnalités qui s'exécutent de manière séquentielle, concurrente ou conditionnelle par rapport à celle-ci. Une fonctionnalité de consultation est définie par une identité (un nom), la description (commentaire) de ce qu'elle permet, et par la spécification des requêtes qui traduisent et activent cette fonctionnalité. [59]

En ce qui concerne notre système ont va exprimer uniquement le type de Préférences d'Activité dans le profil citoyen par la dimension de préférences. C'est une dimension qui dépend de l'historique de l'utilisation du citoyen (le feedback), elle contient une liste de préférences décrivant les services administratives que le citoyen a utilisés le plus, ainsi que les scores qu'il a déjà attribué à ces services. Nous définissons la préférence comme un tuple composée des paramètres de la manière suivante :

Préférence (service, institution, {critère}, {valeur}, évènement)

Chaque préférence dans la liste est composée des éléments suivants:

-le nom du service utilisé et l'institution responsable de ce service,

-{critère} représente un ensemble qui contient les critères d'évaluation de la préférence, il peut contenir l'un des deux critères du score et nombre d'utilisation ou bien les deux au même

temps ; le score est la note donné par le citoyen, en plus du nombre de fois que le citoyen a utilisé ou visité le service en question,

-l'ensemble {valeurs} contient les valeurs de ces critère.

-Et finalement l'évènement contient l'évènement social qui à déclenché le service.

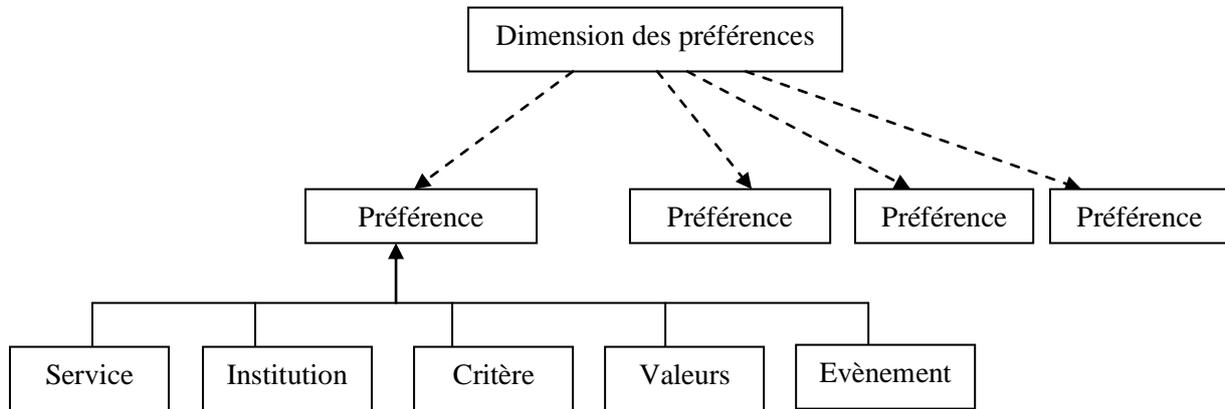


Figure 6.4. Raffinement de la dimension des préférences

4.3. Evolution du profil citoyen

La gestion de l'évolution du profil en fonction des changements des préférences du citoyen dans le temps est une caractéristique distinctive des performances des systèmes de recommandations. La prise en compte de cette évolution peut s'effectuer lors de la modélisation du profil par deux modèles : un modèle à court terme et un modèle à long terme des préférences du citoyen [35].

Dans la majorité des systèmes l'évolution du profil a été presque exclusivement limitée à l'addition de nouvelles informations. En d'autres termes, la représentation des préférences du citoyen est augmentée avec la connaissance de nouvelles données d'utilisations, tandis que le plus souvent l'ensemble des préférences reste intact. La technique utilisée dans ce cas est similaire à celle de la construction [33].

Néanmoins, adapter le profil utilisateur implique des changements au niveau des préférences eux mêmes (leurs contenus) qui conduisent éventuellement à la suppression de quelques unes ou à l'émergence d'autres selon l'historique d'utilisation.

De ce fait, afin de faire face aux changements de préférences dynamiques, l'évolution consiste à adapter la structure et/ou le contenu du profil aux variations des préférences du citoyen.

Cependant afin que l'évolution soit prise en considération le profil doit exister de manière permanente, ce qui est notre cas et non pas celui dans la majorité des systèmes où, le plus souvent à chaque connexion de l'utilisateur, un nouveau profil est instancié.

Dans notre système, le profil utilisateur évolue en ajoutant de nouveaux éléments aux préférences du citoyen ce qui augmente le nombre des préférences. Ainsi, le processus de mise à jour recalcule pour chaque élément, les valeurs des critères déjà existantes :

- Si l'élément appartient déjà au profil, le système modifie cet élément et met ainsi à jour les valeurs de critères.
- Sinon, le système crée un nouvel élément traduisant une nouvelle préférence et met ainsi à jour le contenu du profil.

La mise à jour du contenu du profil traduit la notion de cycle de vie d'une préférence. En effet, on associe aux préférences du profil d'utilisateur une valeur de critère traduisant le degré d'importance d'une préférence par rapport à une autre. Elle augmente à chaque fois quand les citoyens utilisent le service associé à la préférence, et elle diminue pendant une période constante.

Les préférences qui ont une grande valeur d'utilisation restent dans le profil et respectivement, les préférences qui rencontrent peu d'utilisations seront soustraites graduellement et auront finalement tendance à disparaître. Basée sur les valeurs des critères, la structure de la vue personnelle peut être modulée pendant que les préférences des citoyens changent.

5. Modélisation des services et requête du citoyen

Dans cette partie nous allons décrire la requête du citoyen ainsi que le modèle de services utilisé :

5.1. Requête du citoyen

Afin de simplifier les demandes des citoyens, le système propose une liste d'événements sociaux parmi lesquels le citoyen est invité à choisir en fonction de ses besoins, parmi les événements sociaux proposés par le système on peut citer:

- Postuler pour un emploi
- Voyage en dehors du pays

- Ouvrir une nouvelle entreprise
- demande d'un logement

5.2. Modélisation des services

L'entité de service représente le concept de base de notre système; chaque entité est exprimée par une série d'activités administratives que le citoyen peut effectuer et est composée de deux types de services: événement social et service administratif.

Le but de notre modèle est de définir la relation entre l'événement social du citoyen et les services administratifs offerts par la ville. Chaque événement social est donc associé à un ensemble de services administratifs. La structure du modèle est expliquée dans la figure suivante:

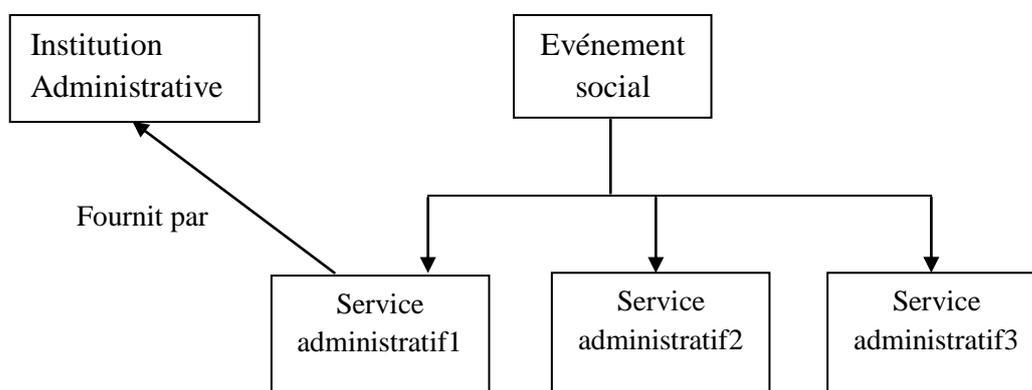


Figure 6.5. Modèle général de l'entité service

Le modèle de service change en fonction de l'événement social et du nombre de services administratifs associés. La figure 6.6 montre un exemple d'application pour un événement social professionnel:

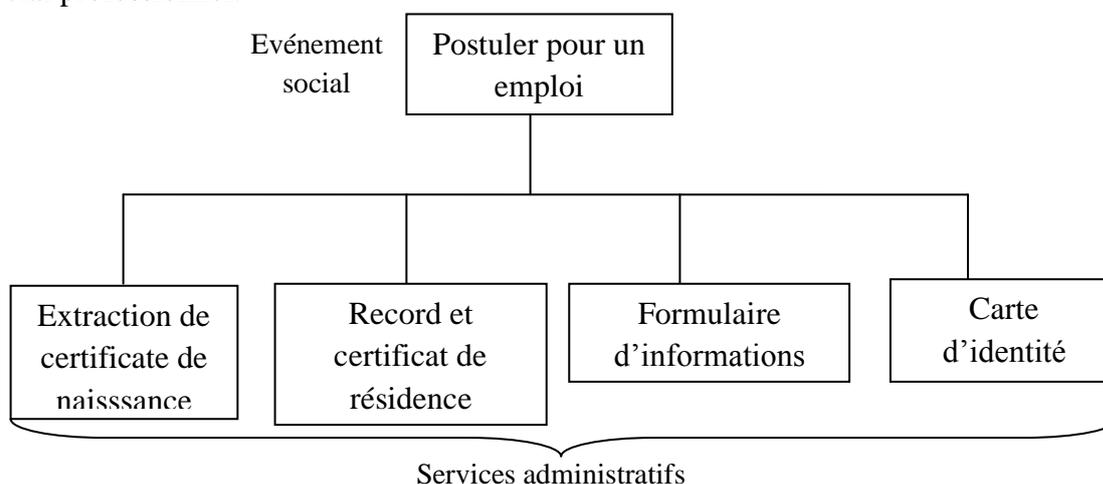


Figure 6.6. Modèle d'un exemple d'événement social

6. L'approche de recommandation

Notre approche consiste à proposer aux citoyens un ensemble de recommandations contextuelles en fonction de leur demande. Pour ce faire, le processus de recommandation passe par les étapes suivantes [2]:

1. Obtention d'informations contextuelles: les informations contextuelles peuvent être obtenues de plusieurs manières, y compris explicitement en s'adressant directement aux personnes concernées et à d'autres sources ou implicitement à partir des données ou de l'environnement.
2. Modélisation des informations contextuelles dans le système
3. Utilisation des informations contextuelles: le processus de recommandation sensible au contexte peut prendre l'une des trois formes suivantes: pré-filtrage, modélisation ou post-filtrage contextuels.

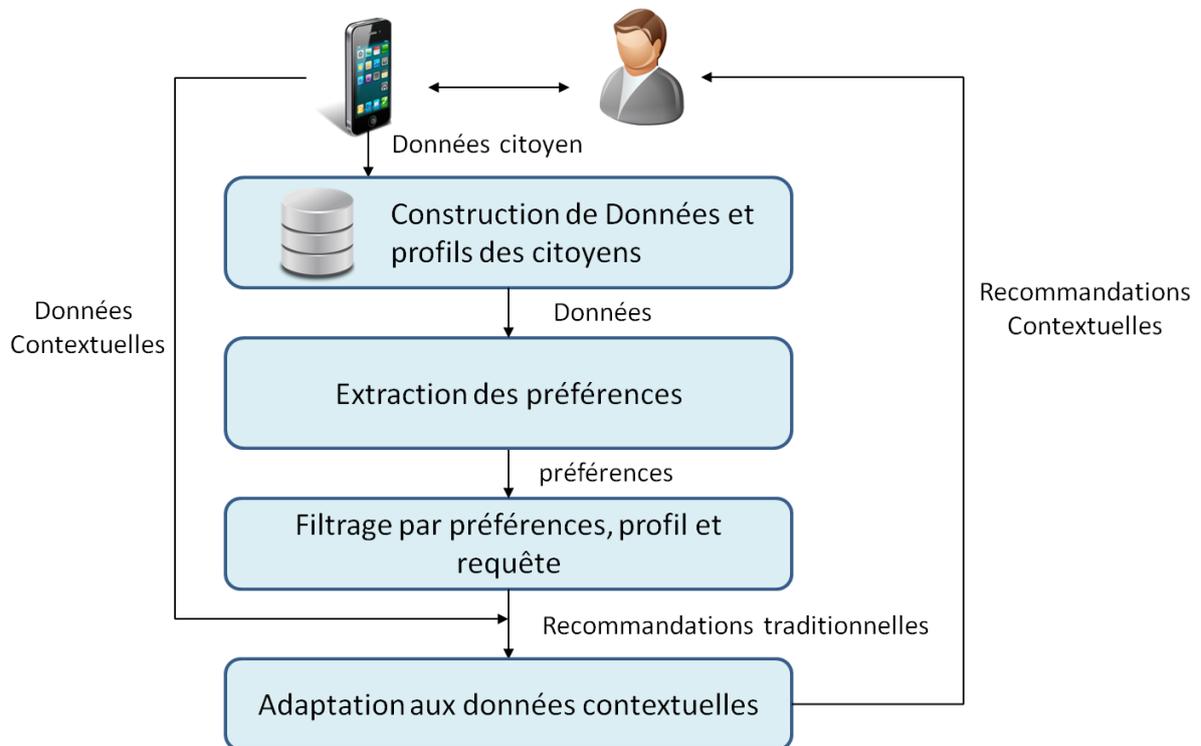


Figure 6.7. Les étapes de l'approche de recommandation

6.1. Les sources de données

Dans notre travail, deux types de données sont utilisés pour le système: les données sur les institutions de la ville et les données relatives du citoyen, chacune d'elles provenant d'une source différente, à savoir [2]:

- Les données présentées concernant les institutions administratives de Biskra ont été obtenues à partir d'une procédure de collecte de données que nous avons effectuée à partir des différents sites Web de cette institution et des coordonnées géographiques de certains SIG tels que GoogleMaps et OSMdroid.
- Quant aux données relatives aux citoyens qui sont obtenues implicitement à partir des données ou de l'environnement, telles qu'un changement détecté de la localisation du citoyen ou des informations de profil, la source des informations contextuelles implicites, qui dans notre cas est le smartphone du citoyen, est accessible directement et les données en sont extraites.

6.2. Modélisation et évaluation des préférences

Les systèmes de recommandation contextuels prennent en charge la modélisation et la prévision des goûts et des préférences des utilisateurs en incorporant les informations contextuelles disponibles au processus de recommandation en tant que catégories supplémentaires de données. Ces préférences et goûts à long terme sont généralement exprimés sous forme des évaluations ou notes (ratings) et sont modélisés comme une fonction non seulement des articles et des utilisateurs, mais également du contexte. En d'autres termes, les évaluations sont définies par la fonction d'évaluation (rating) comme suit:

$$R: \text{Utilisateur} \times \text{Article} \times \text{Contexte} \rightarrow \text{Rating}$$

Où Utilisateur et Article sont respectivement les domaines des utilisateurs et des éléments, Rating est le domaine des évaluations et Contexte spécifie les informations contextuelles pouvant être de différents types, chaque type définissant un aspect du contexte, tel que l'heure, le lieu, etc. [60].

En ce qui concerne notre cas, nous avons adapté la fonction d'évaluation en fonction de nos besoins et le résultat est légèrement différent. Elle a comme entrée le citoyen qui est représenté par un profil contenant ses informations personnelles, l'institution qui représente

l'institution administrative de Biskra et le contexte qui, dans notre cas, est le lieu et l'heure du citoyen; la fonction d'évaluation serait donc modélisée comme suit:

$$R: \text{Citoyen} \times \text{Institution} \times \text{Contexte} \rightarrow \text{Rating}$$

Afin d'intégrer les informations contextuelles (emplacement et heure) du citoyen dans le processus d'évaluation, chaque contexte est défini comme une dimension de notre fonction d'évaluation multicritères [2].

6.3. La fonction de rating

Puisque les éléments recommandés dans notre cas sont des institutions administratives, la localisation géographique représente une caractéristique importante du processus d'évaluation.

Pour notre approche, nous utilisons une évaluation multicritères, dans laquelle chaque établissement est évalué selon quatre critères: distance, points de séjour (stay point), note donnée par les citoyens et heure d'ouverture, chacun étant calculé comme suit [2]:

- Distance: calculer la distance entre la position réelle du citoyen et la localisation de l'institution selon la loi des cosinus sphériques [61], définie par la formule suivante:

$$\text{Distance} = \text{acos}(\sin \phi_1 \times \sin \phi_2 + \cos \phi_1 \times \cos \phi_2 \times \cos \Delta\lambda) \times R$$

Où ϕ_1 et λ_1 sont respectivement la latitude et la longitude de la position du citoyen, et ϕ_2 et λ_2 sont la latitude et la longitude de l'emplacement de l'institution, alors que R représente le rayon de la Terre, ce qui équivaut à $R = 6371$ km.

- Points de séjour: l'institution administrative est mieux notée si elle représente un point de séjour pour un citoyen; ce qui signifie que plus le citoyen visite une des institutions administratives, plus le classement devient plus élevé, le séjour est détecté en fonction des trajectoires GPS en faisant correspondre la localisation du citoyen à chaque instant avec la localisation des institutions administratives.
- Evaluation du citoyen: la note que le citoyen a explicitement attribuée à l'institution sur une échelle de 1 à 5, le citoyen ayant le choix entre attribuer une note ou non.
- Heure d'ouverture: c'est une valeur binaire qui compare l'heure de la demande du citoyen à l'heure d'ouverture de l'institution administrative si celle-ci est ouverte à ce moment-là. Le critère prend la valeur 1 si ce n'est pas un 0.

Ainsi, chaque évaluation d'une certaine institution i par un citoyen c est représentée par un vecteur R où chaque élément r_j correspond à la valeur de l'une des critères ci-dessus, comme suit [2]:

$$R(c, i) = (r_0, r_1, r_2, r_3)$$

Citoyens	Critère d'évaluation			
	Distance	Point de séjour	Note du citoyen	Heure d'ouverture
Citoyen1	1.33 km	6	3.5	1
Citoyen2	1.84 km	0	-	0
Citoyen3	3.06 km	16	4	1

Tableau 6.1. Exemple d'évaluation d'une institution administrative par trois citoyens [2]

Les institutions sont présentées aux citoyens afin d'être notées. Il existe donc des cas où les institutions n'ont pas été évaluées par le citoyen, de sorte que la valeur du critère d'évaluation soit vide, comme indiqué dans le tableau 6.1.

6.4. Utilisation des informations contextuelles par l'algorithme de post-filtrage

Le post-filtrage est l'un des paradigmes existants qui permettent l'utilisation des informations contextuelles dans un système de recommandation. Dans ce paradigme de recommandation, les informations contextuelles sont initialement ignorées et les évaluations sont prédites à partir de l'ensemble des données. Ensuite, l'ensemble de recommandations résultant est ajusté pour chaque utilisateur à l'aide des informations contextuelles [60].

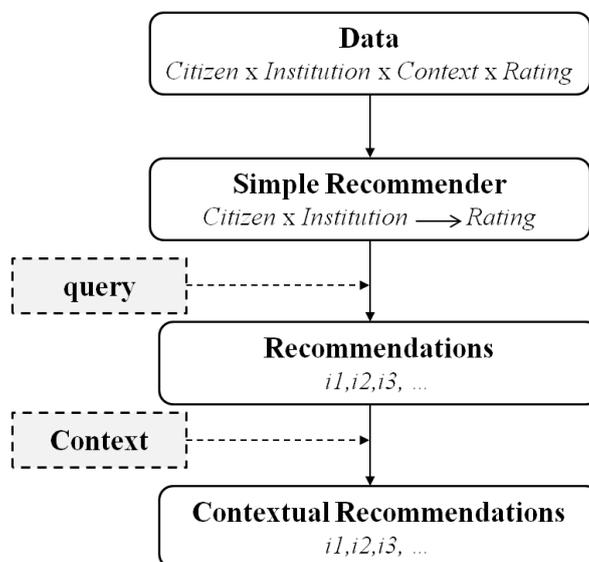


Figure 6.8. Paradigme de post-filtrage contextuel [60]

Notre approche utilise d'abord le paradigme de post-filtrage sur les données de la ville afin d'extraire et de filtrer les réponses en fonction de la requête et du contexte du citoyen [2].

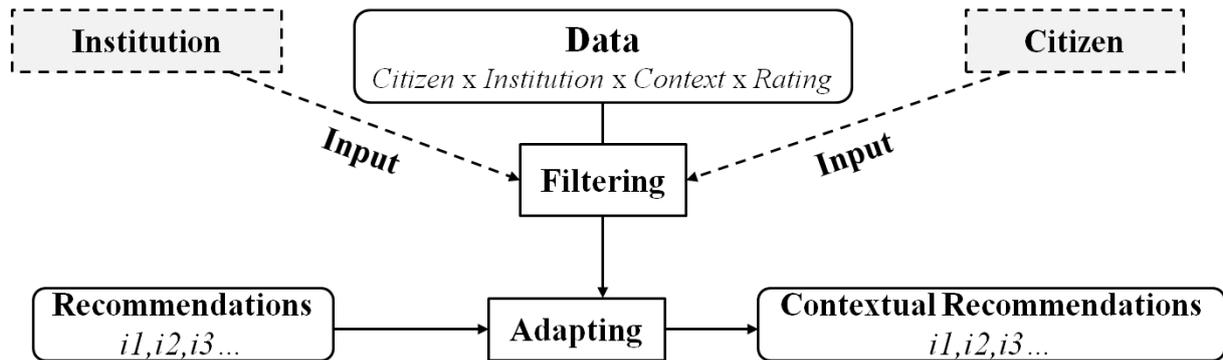


Figure 6.9. Le processus de post-filtrage contextuel [2]

- Filtrer les institutions recommandées qui ne correspondent pas au profil de la requête du citoyen ou ne sont pas pertinentes dans le contexte donné.
- Adapter la liste des institutions recommandées correspondant à la requête du citoyen en fonction de leur pertinence par rapport au contexte et Ajuster le classement des recommandations finales sur la liste [2].

6.4.1. Algorithme de filtrage

L'étape de filtrage représente la première forme de recommandation où le système fait la sélection parmi les services ceux qui répond à la requête formulée par le citoyen. La requête du citoyen peut être de trois types différents : soit qu'il s'agit d'une requête simple de recherche d'une institution administrative, ou bien, l'exécution d'un service directe, ou alors choisir l'un des événements sociaux proposés par le système. Dans chacun des cas, notre système fait le filtrage par rapport au service et aussi l'évènement social associé ; A partir de la requête reçue du citoyen, le filtrage choisit d'abord les services en relation directe avec la requête, par la suite une recherche est faite pour trouver les événements sociaux qui peuvent être reliés à la requête afin de recommander les services de ces derniers. Par la suite la liste des services (directe ou reliés) est envoyée à la deuxième étape d'adaptation qui prend en considération le contexte du citoyen que nous avons appelée recommandation sensible au contexte. L'algorithme suivant décrit en détail la procédure de filtrage :

Algorithme filtrage**Input** liste des services S, liste des évènements E, Liste des citoyens C**Output** liste finale de recommandations**BEGIN**

```

c = Service_Authentification_citoyen () ;

```

```

If ( c  $\notin$  C ) // le cas d'un nouveau citoyen

```

```

    Service_création_du_compte () ;

```

```

    Service_formulaire_informations_profile () ;

```

```

Else if ( c  $\in$  C ) // le cas d'un ancien citoyen

```

```

    Requête = service_requête_citoyen () ;

```

```

    If ( requête  $\in$  S ) // la requête du citoyen est un service directe

```

```

        Services = Recherche_service_dans la Base de Données (requête) ;

```

```

        Execution_service (services);

```

```

        Mise_à_jours_préférences_citoyen (requête, c);

```

```

        Evènements = Recherche_évènement_relié (services) ;

```

```

        Recommandations = Recherche_services_de_l'évènement (Evènements) ;

```

```

    Else if ( requête  $\in$  E ) // la requête du citoyen est un évènement

```

```

        Sevices = Recherche_services_de_l'évènement (requête) ;

```

```

        Evènements = Recherche_évènement_relié (requête) ;

```

```

        If (Evènements  $\neq$  NULL)

```

```

            Recommandations= Recherche_services_de_l'évènement(evènements) ;

```

```

        End

```

```

    Else // la requête est une recherche d'institution

```

```

        service_recherche_institution (requête) ;

```

```

        Sevices = Recherche_services_de_l'institution (requête) ;

```

```

End

```

```

Preferences = Recherche_préférences_citoyen (c) ;

```

```

Contexte = Recherche_contexte_citoyen (c) ;

Recommandations = Recommandation_algorithme (c, requête, services,
préférences, contexte) ;

Affichage des recommandations finales (recommandations) ;

END
END

```

6.4.2. Algorithme de recommandation

Cette phase représente la recommandation sensible au contexte, elle reçoit comme entrée une liste de recommandations déjà filtré selon la requête à l'étape précédente, afin de calculer les évaluations (ratings) de chacun des services de la liste et par la suite faire le tri croissant des services pour obtenir la liste finale de recommandations. L'algorithme suivant explique en détail les étapes de cette phase :

Context-aware Recommendation Algorithme

Input List des citoyens C , le citoyen $c \in C$, requête citoyen q ,
Context du citoyen $context$, Liste des préférences du citoyen $preferences$,
Liste des services filtrés S ,
Nombre de services à recommander N ,
Matrice des évaluations (ratings) citoyen-service R

Output Liste des Top- N services recommandés $Rec(c)$

```

BEGIN
  Rec(c) ← Empty-List;
  For  $i \leftarrow 1$  to longueur( $S$ ) do/ Foreach  $s \in S$  do
    If ( $s \notin preferences$ ) do // le service n'appartient pas aux preferences du citoyen
      Calcule-rating ( $s, c, q, context$ );
    Else do // le service existe déjà dans la liste de préférences
      Mise-à-jours-rating ( $s, c, q, context$ );
    end
    Ajouter ( $s, List$ );
  end
  Rec(c) ← Tri_services_selon_ratings(List);
  Afficher_Top_N_recommandations(Rec(c));
END

```

7. Le modèle conceptuel

L'objectif de notre travail de recherche est de réaliser un système d'administration distribuée utilisant la technologie du cloud computing pour offrir aux citoyens un ensemble de services et de recommandations basés sur le concept de reconnaissance de contexte. Nous allons présenter le modèle conceptuel suivant qui est inspiré de l'approche BAAS, car il convient à notre contexte de recherche de deux manières. D'une part, une ville intelligente a besoin d'un système distribué avec un serveur centralisé où toutes les données et services de la ville peuvent être stockés pour être livrés ultérieurement aux entités de la ville et c'est exactement ce que l'approche BAAS offre; un ensemble de services de cloud exposés à travers une Rest-API qui assure des interactions en temps réel. D'autre part, les systèmes de recommandation sont révélés être une approche très efficace pour adapter les services aux besoins et préférences des citoyens, Cependant ces systèmes souffrent d'un problème de scalabilité lié aux grandes quantités de données. Pour traiter ce problème, nous avons choisi l'approche BAAS caractérisée par son auto-dimensionnement, son architecture et ses contraintes REST, qui aident à améliorer et à maintenir la scalabilité du système [2] [3].

Conçu pour une ville intelligente, notre système possède une architecture centralisée divisée en deux parties logiques: le backend (essentiellement la partie principale) et le front-end (la partie application), qui communique via un ensemble de services API. La figure 6.10 suivante illustre la conception générale de l'ensemble du système [2] [3].

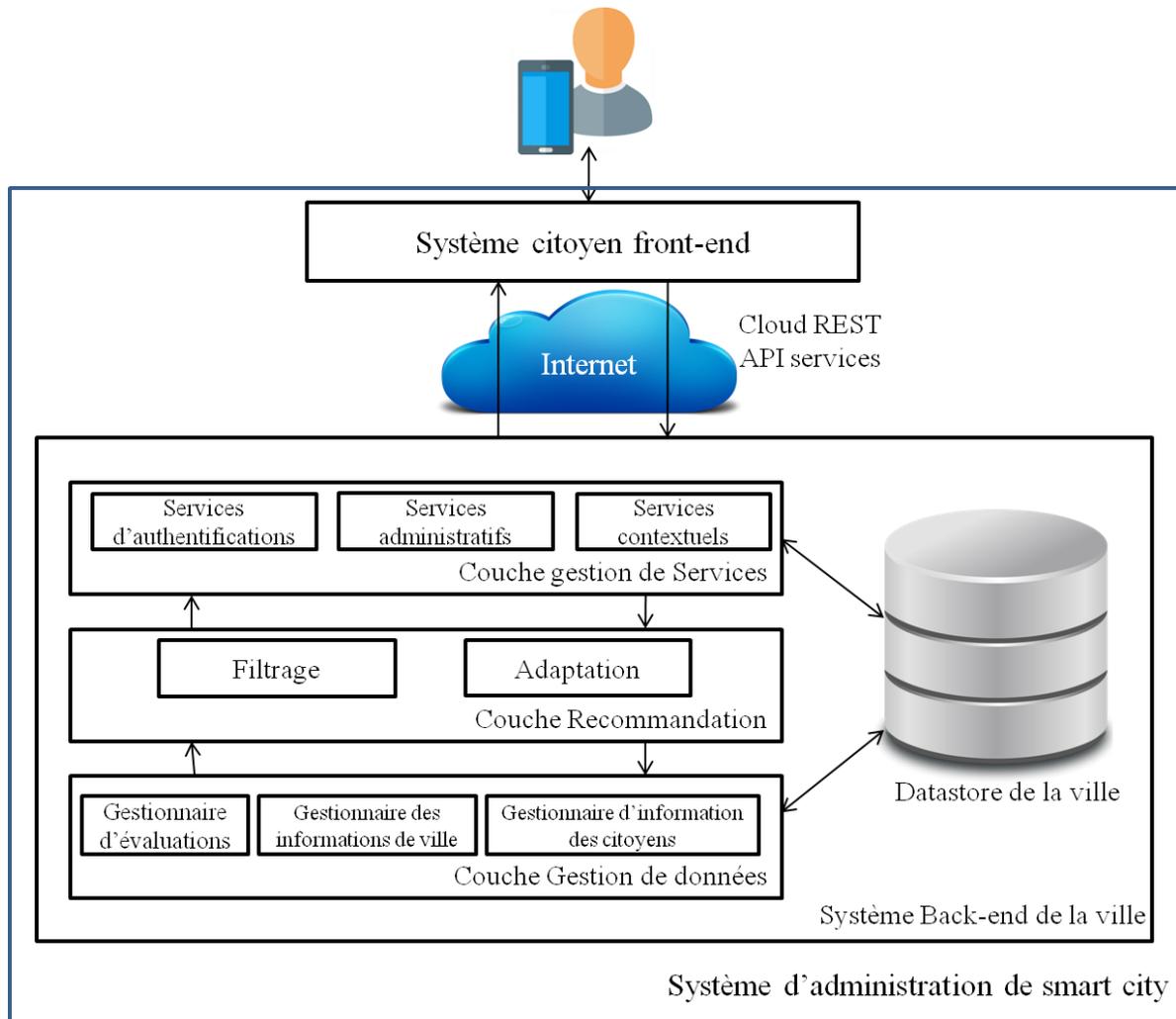


Figure 6.10. Cadre conceptuel global du système proposé de ville intelligente [2].

7.1. Le système backend

La partie Backend est le système central de la ville déployé sur le cloud qui effectue les fonctions principales et fournit des services de recommandations adaptées aux demandes des citoyens. Il contient une série de couches qui coopèrent pour atteindre l'objectif général du système [2].

- La couche de gestion des données: cette couche est responsable de toutes les procédures de collecte et de gestion des données.
- La couche recommandation: elle permet une exploitation intelligente des données du système et contient les fonctions essentielles nécessaires pour recommander des résultats et le filtrage des données. Elle est responsable du processus de recommandation sensible au contexte.
- La couche de gestion de services: son rôle est de gérer les différents services du système [2].

7.2. Le système frontend

La partie frontale représente le système d'application mobile du citoyen qui performe des tâches relatives et interagit directement avec les citoyens; Il joue le rôle d'intermédiaire entre les citoyens et le système backend de la ville, qui communique via les appels de service appropriés en fonction des demandes [2].

8. Fonctionnement de BAAS

Notre approche consiste à créer un Framework pour les services administratifs et le partage d'informations selon l'approche Backend as a service.

Nous utilisons la partie backend comme système de serveur pour la ville et la partie frontale est représentée par l'application mobile du citoyen permettant de ne consommer que les services proposés et d'envoyer des données sur les citoyens. En ce qui concerne les données de la ville, elles sont gérées et enregistrées dans le datastore. Chaque enregistrement est représenté sous forme de ressource à laquelle il ne peut accéder que par les services Rest API du système, de sorte que le protocole Rest et les contraintes contrôlent l'accès aux données et garantissent donc sa scalabilité, même en grande quantité.

Nous avons également utilisé l'approche BAAS afin de mettre en œuvre un ensemble de services Rest API permettant de transmettre des données de requête et de réponse entre le backend et la partie front-end [2] [3].

La création de services administratifs est automatisée en fonction du type de l'institution. c'est-à-dire si l'institution est une municipalité, des services d'extraction de documents d'identité sont automatiquement créés dans le système backend. Après la création des services, les API associées sont générées automatiquement.

Chaque fonction du système a des services associés et chaque demande de citoyen déclenche un appel de service Rest API associé au type de demande. La figure 6.11 suivante montre le fonctionnement de l'approche BAAS [2].

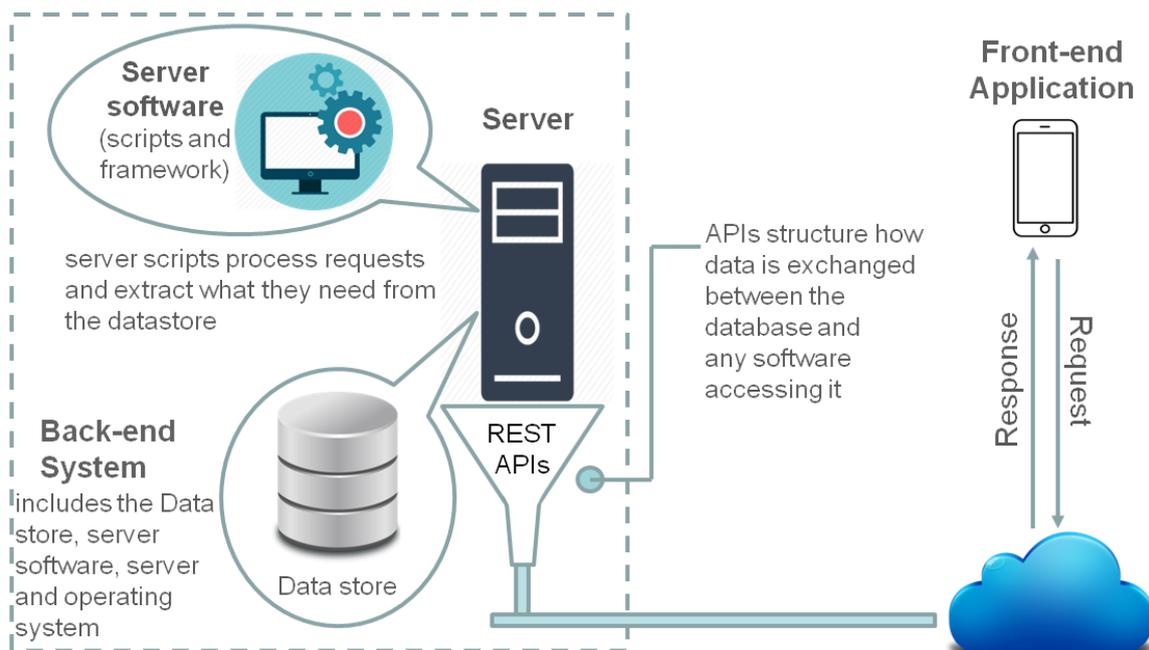


Figure 6.11. Le fonctionnement de l'approche BAAS [2]

9. Conception détaillée

Dans la section suivante nous allons présenter la conception détaillée de tous les composants du système ainsi que les systèmes back-end et front-end.

9.1. La partie back-end

Le système back-end représente la partie centrale du système de la ville, elle regroupe trois couches qui coopèrent entre elles afin d'offrir un ensemble de services à la partie front-end.

9.1.1. La couche de gestion de services

La couche de gestion de services est conçue spécialement pour gérer les interactions entre les deux parties back-end et front-end à l'aide des services définis dans notre système, Elle interagit directement avec la partie back-end lors des appels de services afin de recevoir ou transmettre les informations nécessaires, elle joue donc le rôle d'une interface pour la partie back-end.

Lors de requêtes compliquées, cette couche renvoie la requête ainsi que les données nécessaires à la couche suivante de recommandations pour qu'elle s'occupe du reste de la procédure et par conséquent, transmet les résultats obtenus vers l'application du citoyen.

Les services dans notre système sont classés en trois types différents : services contextuels, services administratifs et services d'authentification.

a) Les services d'authentification

Cette catégorie regroupe les services responsables des différentes procédures reliés à l'identification du citoyen, parmi les fonctions de ces services:

- Authentification du citoyen à l'aide d'un couple d'identifiant et mot de passe
- Création d'un nouveau compte citoyen
- Création d'un nouveau profile à l'aide d'un formulaire rempli par le citoyen
- Modification d'un profile citoyen déjà existant

b) Les services administratifs

C'est les services responsables de traitement des requêtes du citoyen et permettant l'échange d'informations sur les démarches et les institutions administratives. Chaque institution possède un ensemble de services qu'elle offre aux citoyens selon sa fonction, on peut distinguer deux types de services administratifs :

- des services simples de partage d'informations nécessaire pour effectuer une démarche administratives tels que : les formulaires à remplir, les conditions et les papiers nécessaires.
- Des services permettant d'effectuer des opérations administratives enligne. Parmi lesquelles on peut citer: la demande de papiers enligne, consulter son compte...etc.

c) Les services contextuels

C'est le type de services responsable d'obtenir les informations contextuelles spatiale et temporelle du citoyen, ces informations sont obtenus d'une manière implicite ; ils sont collectées automatiquement à partir de l'application citoyen afin d'être stockés, traités et utilisés par le reste du système dans le processus de recommandation. Ce module est composé des deux services suivants:

- **Le service temporel**

Le rôle de ce service est d'extraire la date et le temps actuel du citoyen lors d'une requête pour vérifier par la suite si l'institution administrative responsable du service est

ouverte en ce moment ou fermée et mettre à jours la valeur du temps dans la matrice d'évaluation par 1 si elle est ouverte et 0 si non.

- **Le service de localisation**

Afin d'obtenir la position géographique du citoyen ce service reste à l'écoute jusqu'à ce qu'il détecte un changement de la localisation, lors de cet évènement il se connecte au serveur de localisation GPS pour recevoir les coordonnées géographique de cette localisation, et par la suite fait la vérification si cette position représente la localisation d'une institution administrative ou non, si c'est le cas ça veut dire que le citoyen a visité cette institution administrative donc la valeur de point de séjours (stay point) de cet institution doit être incrémenté dans la matrice d'évaluation .

SERVICE DE LOCALISATION

Input Liste des institutions Institutions, Matrice des évaluations R, Liste des services S.

BEGIN

```
locationManager=(locationManager)this.getSystemService(Context.LOCATION_SERVICE);
```

```
location listener lc= new Location Listener () {
    public void onLocationChanged (location location){
        position = location;
        latitude = position.getLatitude() ;
        longitude = position.getLongitude() ;
        for each (inst ∈ Institutions ) do {
            If (inst.latitude = latitude and inst.longitude = longitude) {
                For each service do {
                    If (service.Institution = inst) {
                        R[service,c].staypoint ++;
                    end
                end
            end
        end
    end
end
```

```
LocationManager. Request Location Updates (locationManager, GPS_Provider,0,0,lc);
```

END

9.1.2. La couche recommandation

En étant la couche principale du système, elle regroupe toutes les opérations nécessaires pour le traitement des données selon l'approche de recommandation sensible au contexte, pour cela elle est définie par les deux sous-systèmes correspondant aux deux étapes de filtrage et d'adaptation.

a) Le sous système de filtrage

L'objectif du sous système de filtrage est d'obtenir une liste de recommandations composée de services qui convient à la requête du citoyen et sans tenir en compte son contexte. Afin d'atteindre cet objectif, il est composé de deux modules comme suit :

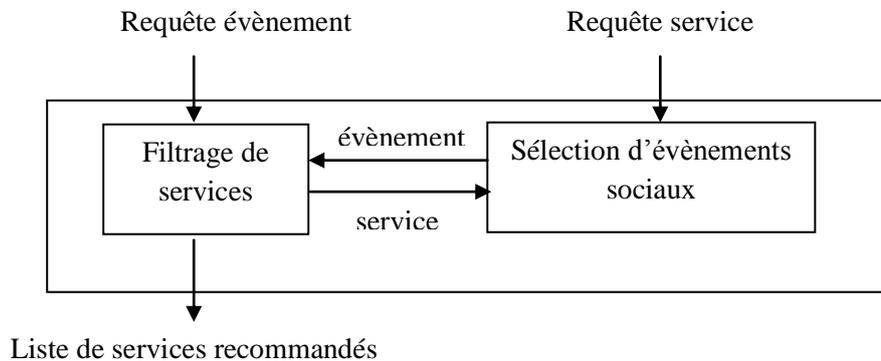


Figure 6.12. Modèle conceptuel du sous système de filtrage

- Le module filtrage de services : déclenché par une requête du citoyen de type évènement social, ce module effectue une recherche dans la base de données afin d'extraire les services reliés à l'évènement qu'il a reçu.
- Le module sélection d'évènements sociaux : Lors d'une requête service ce module fait une recherche pour obtenir les évènements sociaux qui ont une relation avec le service en question.

b) Le sous système d'adaptation

Le rôle de ce sous système est d'adapter la liste de services reçue du sous système de filtrage en fonction du contexte du citoyen pour obtenir la liste finale de services recommandés sensible au contexte. Ce rôle important est distribué sur plusieurs modules comme suit:

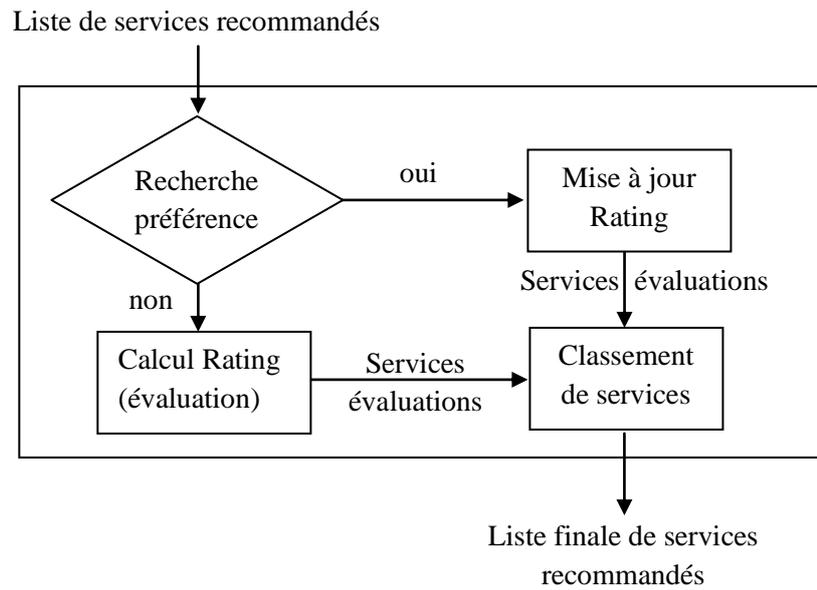


Figure 6.13. Modèle conceptuel du sous système d'adaptation

- Le module recherche préférences : fait la recherche du service parmi les préférences du citoyen, si le résultat est positif elle fait l'appel du module mise à jour rating, si non elle appelle le module calcul rating.
- Le module calcul rating (évaluation) : fait le calcul de l'évaluation des services selon la fonction de rating expliqué à la section précédente.
- Le module mise à jour rating : ce module fait la mise à jour de l'évaluation déjà existante des services.
- Le module classement de services : ce module fait le tri croissant de services selon la valeur de leurs évaluations.

9.1.3. La couche de gestion des données

Cette couche est responsable de toutes les opérations de gestion des données et informations du système. Elle est divisée en trois modules :

- Gestionnaire des informations des citoyens : c'est le module permettant le stockage, la modification et l'utilisation de toutes les données relatives aux citoyens, son profile et ses préférences.
- Gestionnaire des informations de la ville : c'est le module permettant le stockage, la modification et l'utilisation de toutes les données relatives aux institutions administratives de la ville ainsi que leurs services.
- Gestionnaire d'évaluations : Elle est responsable de toutes les procédures de sauvegarde et mises à jour des évaluations entre services et citoyens ainsi que les évènements sociaux.

9.2. La partie front-end

Elle joue le rôle d'interface du système, c'est l'application mobile qui interagit directement avec le citoyen. Son fonctionnement se résume en deux points ; la collecte les données fournit par le citoyen et lance un appel vers les services de la partie back-end. Elle est composée de deux modules suivants :

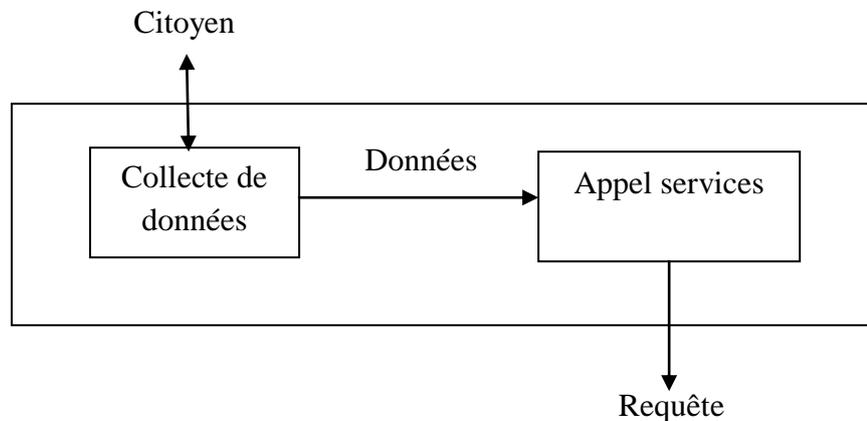


Figure 6.14. Modèle conceptuel du système front-end

- Collecte de données : c'est le module responsable de la procédure de la récolte de données fournit par le citoyen à travers des formulaires à remplir.
- Appel services : le rôle de ce module est de traduire la requête du citoyen obtenu du module collecte de données vers un appel de l'un des services du système back-end.

10. Fonctionnement du système

Afin de décrire le fonctionnement de notre système nous allons faire recours dans cette section au langage UML pour bien expliquer le fonctionnement à l'aide d'un ensemble de schémas illustratifs.

10.1. Diagramme de classes

Dans notre système, on peut distinguer huit classes différentes chacune représentant une entité du système et reliées par deux types de relations : relation d'activation et relation d'association. Parmi ces classes, nous avons trois de type service et le reste sont des entités.

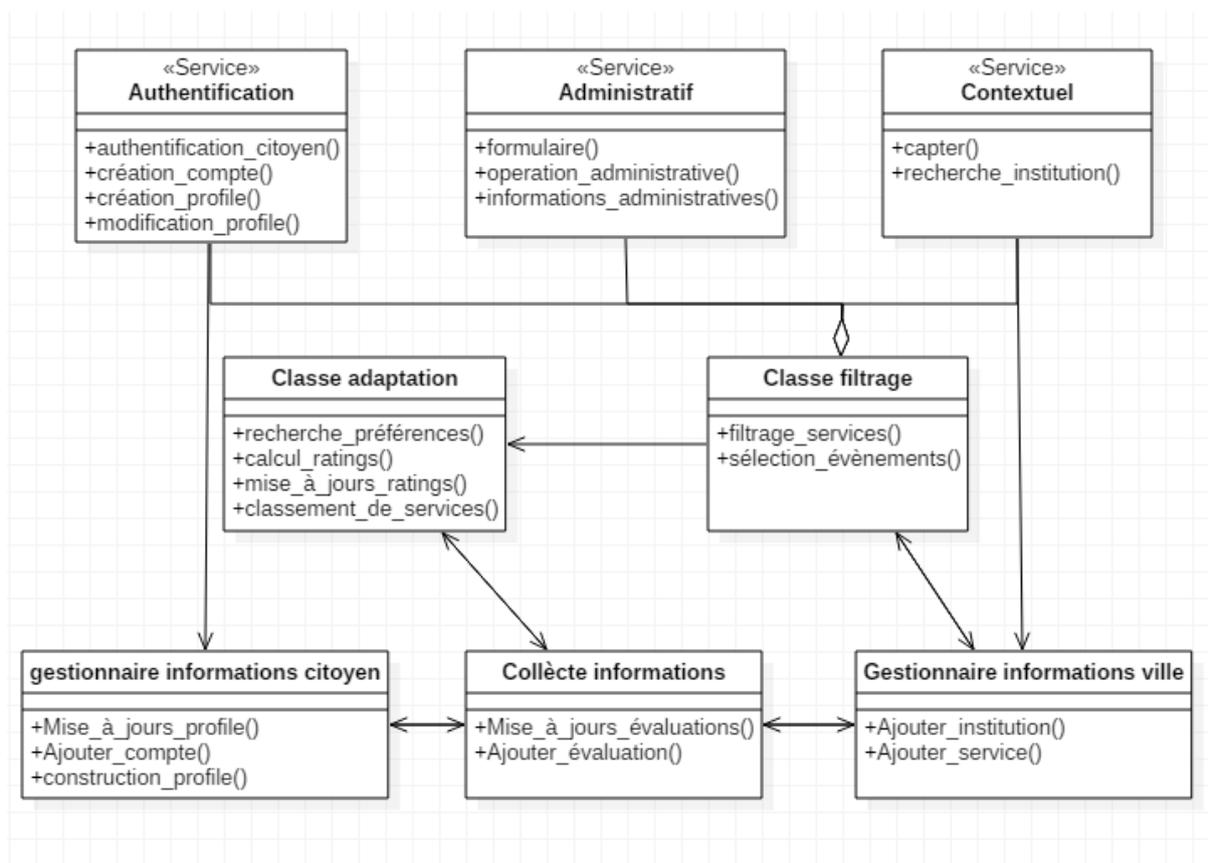


Figure 6.15. Diagramme de classes

10.2. Diagramme de cas d'utilisation

Nous constatons dans notre système quatre différents cas d'utilisation offerts au citoyen et reliés par des relations de dépendances, le premier cas est celui d'une requête d'inscription lorsqu'il s'agit d'un nouveau citoyen où on doit lui créer un nouveau compte et profile, le reste des cas représente un ancien citoyen alors une authentification est nécessaire. La deuxième requête est une recherche d'une institution administrative précisée par le citoyen, dans ce cas, le filtrage et exécution de services doivent être activés. Le troisième cas d'utilisation s'agit du choix d'un événement social qui active par la suite les cas de filtrage de services, sélection d'évènement, évaluation rating et classement de services. Le quatrième et dernier cas est une requête de service qui active par la suite le reste des cas nécessaires.

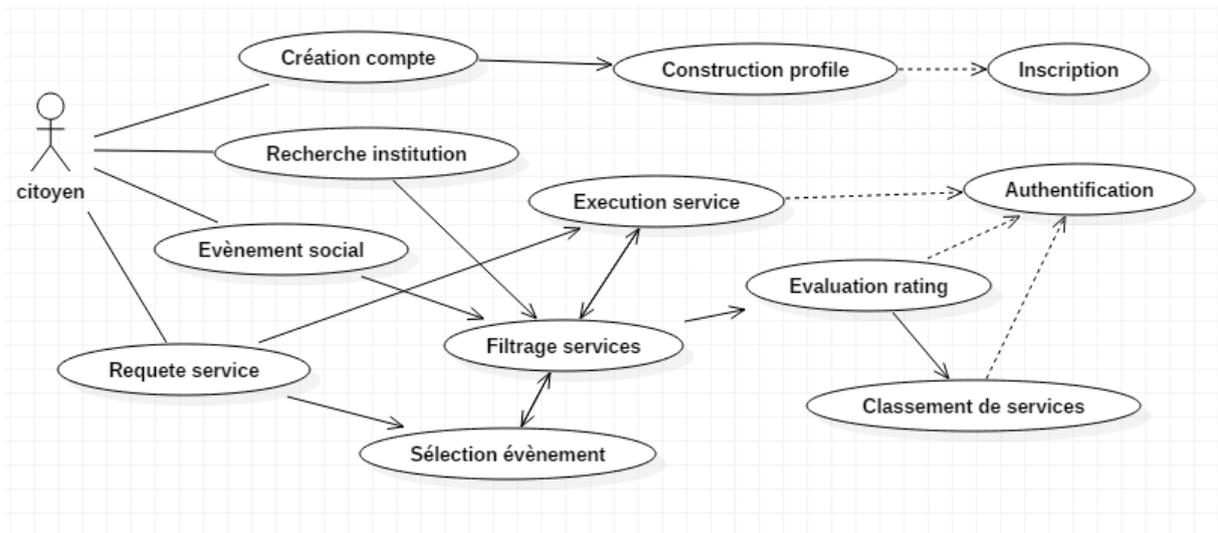


Figure 6.16. Diagramme de cas d'utilisations

10.3. Diagrammes de séquences

Toutes séquences de notre système commencent par une requête du citoyen qui interagit directement avec l'application qui effectue par la suite un appel vers un service administratif de la partie back-end, après cette étape initiale, commence le traitement de la requête reçu par l'étape de filtrage où le module sélectionne l'évènement social approprié et par la suite envoie l'évènement choisi au filtrage de services pour extraire la liste de services compatible avec la requête, pour passer à l'étape suivante d'adaptation, le schéma suivant décrit cette séquence en détail.

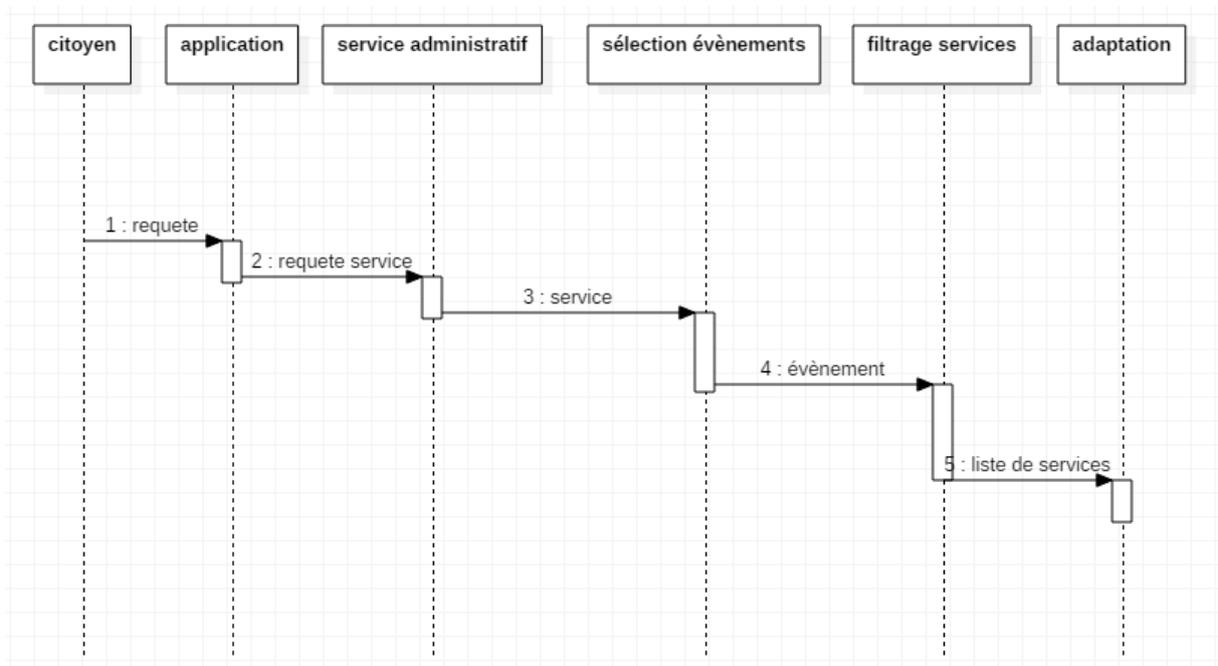


Figure 6.17. Diagramme de séquence du filtrage

La troisième étape d'adaptation commence par la réception des services à partir du filtrage et faire la recherche des services parmi les préférences du citoyen selon laquelle le choix est fait entre le calcul ou la mise à jour des évaluations (rating) et par la suite faire le nouveau classement des services dans la liste de recommandations finale.

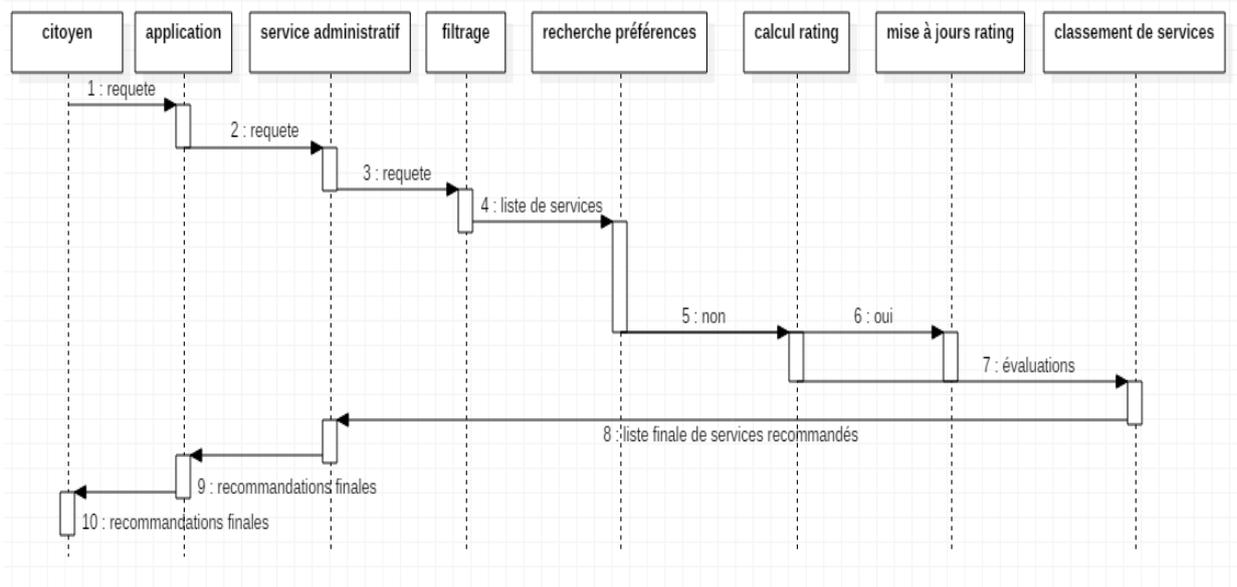


Figure 6.18. Diagramme de séquence d'adaptation

Et finalement, Afin d'acquérir les données nécessaires au processus de recommandation le module adaptation interagit avec les gestionnaire d'information du citoyen, de la ville et d'évaluation pour qu'ils effectuent l'extraction des informations stockées dans le-data store et retourner les résultats cette étape est illustrée par le schéma suivant.

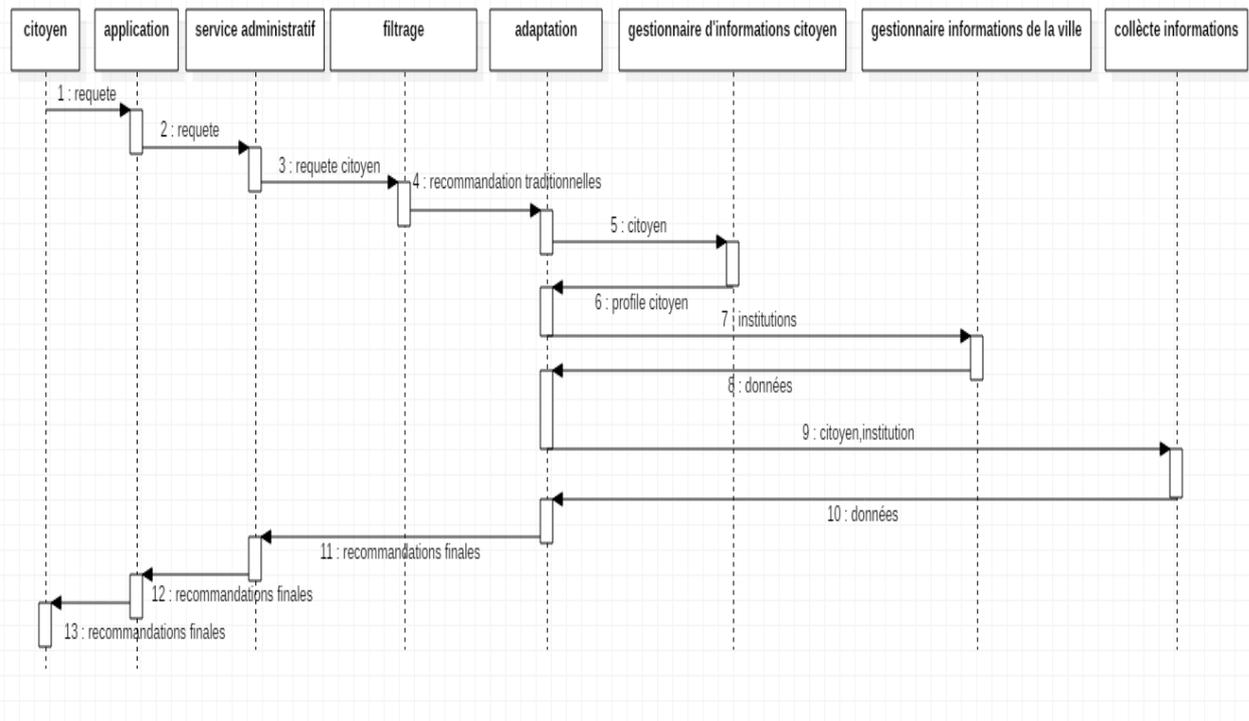


Figure 6.19. Diagramme de séquence des gestionnaires d'informations

11. Conclusion

Nous avons parvenu dans ce chapitre à concevoir en détail un système d'administration de smart city en se basant sur l'approche de recommandations sensible au contexte qui représente l'approche principale du système et aussi nous avons décrit l'approche back-end as service sur cloud en plus du fonctionnement globale du système à travers l'explication de ses modules et sous systèmes et aussi les différentes interactions entre eux.

Après l'étape de modélisation, le prochain chapitre sera consacré à la réalisation de notre système.

CHAPITRE 05 : VALIDATION DU MODELE PROPOSE

1. Introduction

Afin de bien illustrer notre approche de système d'administration de smart city nous avons choisi comme cas d'étude l'administration de la ville de Biskra ainsi qu'un ensemble d'outils et technologie de développement que nous allons présenter dans ce chapitre. Suivi des différentes étapes et captures de l'implémentation du système, et finir par un ensemble de tests permettant l'évaluation de notre système selon les résultats obtenus.

2. Etude de cas

Dans cette section nous allons étudier l'impact des nouvelles technologies d'informations et de communications sur l'Algérie ainsi que son infrastructure technologique selon les aspects essentiels qui compose notre approche proposée pour vérifier la possibilité que notre système peut être réalisé ou non selon l'état actuel du pays.

2.1. L'Infrastructure technologique des TICs en Algérie

La mise en œuvre des connexions : l'extension des services satellites et des données pour renforcer la connectivité

Le développement technologique dans les dernières années a joué un rôle fondamental dans la promotion d'une nouvelle culture digitale en Algérie. Les efforts du gouvernement pour accroître le taux de pénétration d'Internet et étendre l'utilisation des technologies de communication ont renforcé la compétitivité du secteur. Selon l'Autorité de régulation des postes et télécommunications (ARPT), le pays avait un taux de pénétration d'Internet de 71,2% à partir de 2016. Les efforts en cours du gouvernement visent à promouvoir une plus grande diversification dans le secteur et à suivre les progrès réalisés ces dernières années grâce au dévoilement des services 3G et 4G, qui ont accru la portée des paquets de données dans tous les segments de la société algérienne. Les opérateurs de téléphonie fixe et d'Internet mobile ont enregistré un chiffre d'affaires combiné de 3,7 milliards d'euros en 2016, selon l'ARPT, soit une croissance de 2,5% par rapport à 2015, avec l'expansion des services 3G et la mise en œuvre des technologies 4G créditées comme principaux facteurs à l'origine de la croissance. A la fin 2016, le pays comptait 29,5 millions d'utilisateurs d'Internet fixe et mobile. L'augmentation rapide de l'utilisation d'Internet a été largement attribuée à une évolution globale vers la numérisation dans l'Algérie [62].

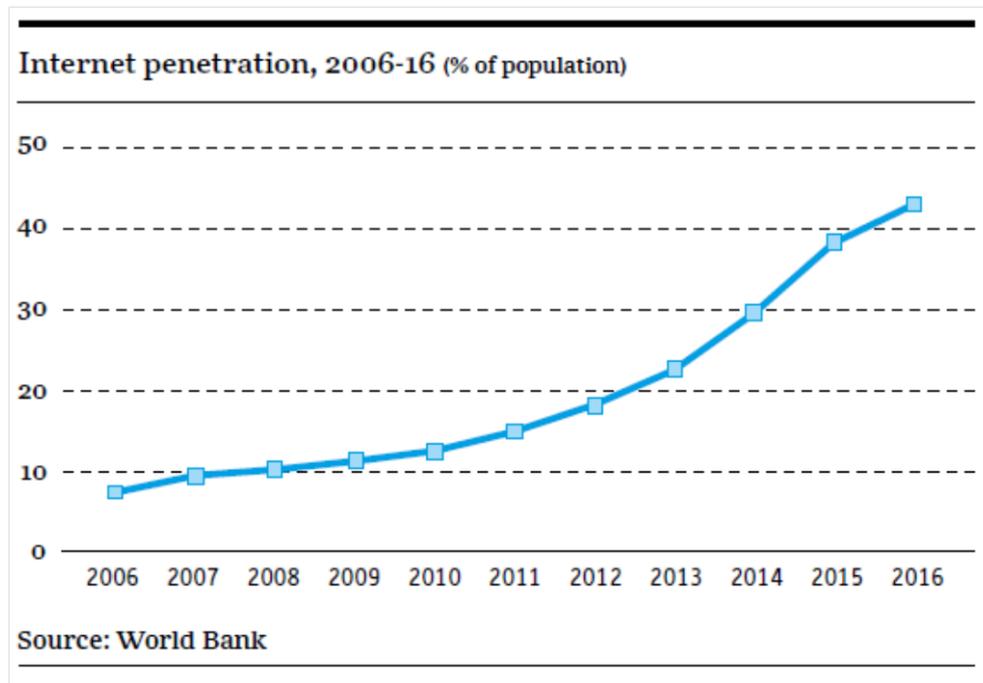


Figure 7.1. Schéma représentant l'évolution de la pénétration d'internet en Algérie [62]

Tendances de déplacement: Le lancement de la 3G en 2013 a été crucial pour transformer le secteur. Les chiffres de 2016 montrent une migration de masse du GSM vers la 3G, qui s'est accompagnée d'une baisse du volume des appels vocaux. Après son lancement à la mi-2016, les services 4G avaient un taux de pénétration modeste de 3,5% à la fin de cette année. L'utilisation d'Internet est passée du fixe au mobile, soutenue par le déploiement étendu des appareils 3G.

À la fin de 2016, 56,7% des utilisateurs de téléphones mobiles étaient abonnés à des services 3G ou 4G. Dans le premier rapport trimestriel de l'ARPT pour 2017, il était estimé que 33,8 millions de personnes accédaient principalement à Internet sur un appareil mobile. Avec un taux de pénétration mobile de 113,4% en 2016 l'Algérie a un segment mobile mature[62].

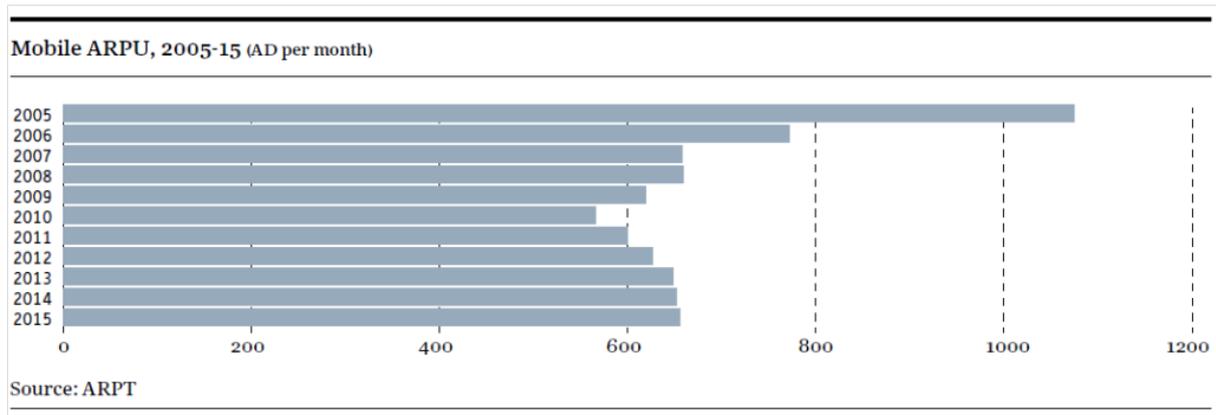


Figure 7.2. Schéma représentant l'évolution de la pénétration de l'utilisation mobile en Algérie [62]

Vitesse de connexion: Il existe actuellement deux câbles sous-marins internationaux avec une bande passante de 1040 Gbps reliant l'Algérie au réseau mondial; cependant, la consommation réelle est de 840 Gbps. Le déploiement des services 3G et 4G a changé l'approche vers l'amélioration de l'expérience client. Ces packages incluent une gamme de services à valeur ajoutée qui fournissent un accès haut débit par paquets et des connexions sans fil.

Internet à ligne fixe: Pour améliorer l'accès à Internet, le gouvernement met en œuvre un plan FTTX pour connecter 7 millions de foyers à l'internet fixe d'ici 2022. Algérie Télécom visait à déployer plus de 500 000 lignes de fibre optique jusqu'au domicile d'une capacité de 100 Mbps et à moderniser les lignes existantes.

Service universel des TÉLÉCOMS: Un élément clé du programme de télécommunications du gouvernement est le service universel des télécommunications annoncé en 2016, qui vise à combler les disparités régionales en matière de connectivité. Le service universel de télécommunications vise à fournir des connexions de télécommunications fixes et sans fil aux régions éprouvant des difficultés d'accès à ces services. Ce programme améliorera les services dans 97 municipalités dans 28 wilayas du sud de l'Algérie.

Tandis que l'expansion des réseaux à fibres optiques et des câbles sous-marins internationaux solidifiera les connexions Internet du pays, ouvrant à son tour des possibilités d'innovation. Dans l'ensemble, les plans à long terme du gouvernement pour embrasser la numérisation sont bien avancés [62].

Adaptations numériques: le développement des TIC pour améliorer les performances dans plusieurs secteurs

Internet des objets: L'amélioration de l'infrastructure numérique devrait avoir un impact significatif sur les utilisations existantes de la technologie. Le cloud computing est devenu une caractéristique majeure des opérations commerciales ces derniers temps, tandis que les coûts croissants associés à l'hébergement poussent les investisseurs à rechercher des marchés alternatifs et moins chers. Compte tenu de sa capacité de bande passante et des vitesses de connexion fournies par les câbles sous-marins, l'Algérie a le potentiel pour être une destination attrayante pour les entreprises internationales. Le déploiement de services de fibre optique améliorés stimulera le développement d'applications locales, ainsi que d'autres services numériques tels que les cartes et les appareils GPS.

Outils de connexion: Au-delà du courrier GPS et intranet, les entreprises s'orientent de plus en plus vers l'interconnexion des activités grâce à l'utilisation de services centralisés. Dans des secteurs tels que l'agriculture, la santé et les transports, cette tendance soutient une diversification économique accrue, avec des services d'irrigation intelligents et le déploiement de capteurs d'optimisation parmi les exemples où les services interconnectés ont aidé les entreprises.

Administration électronique: L'expansion rapide des services 3G et 4G en 2016 et 2017 a eu un impact significatif sur les plateformes électroniques. Après le lancement des systèmes de paiement électronique en octobre 2016, le taux d'activité en ligne a augmenté rapidement. Des représentants du gouvernement ont déclaré que 5000 transactions de paiement électronique avaient été effectuées en décembre 2016. Le développement a permis le paiement électronique des factures des services publics, des télécommunications, des assurances et d'autres problèmes administratifs. Renforçant ainsi les plans du gouvernement visant à améliorer les services d'administration en ligne, à développer de tels systèmes et à faciliter l'accès du public à ces derniers [62].

Smart Cities: La capitale, Alger, a été au centre des efforts pour concevoir des villes plus modernes, efficaces et durables pour s'adapter aux conditions futures, avec l'initiative «Smart City Algiers» qui cherche à profiter des avantages des TIC pour améliorer la qualité de vie des habitants de la ville. L'initiative a attiré l'attention de la société américaine Xerox, qui collabore avec deux fournisseurs Internet dans deux projets distincts. Le premier consiste en un plan d'installation de connexions de télévision par câble dans toute la ville, réduisant

ainsi la pollution associée à la production d'antennes paraboliques. Cela permettrait aux opérateurs d'utiliser des câbles à fibres optiques pour fournir des services de télévision par câble grâce à l'installation d'un seul appareil pour l'ensemble du bâtiment. Le deuxième projet concerne la fourniture d'une connexion Wi-Fi urbaine dans toute la ville pour améliorer le flux de données et la connectivité dans la capitale. Le plan a toutefois été confronté à des défis, avec des contractions budgétaires, des problèmes de licences bureaucratiques, et l'immensité et la complexité du paysage urbain d'Alger qui représentent des défis pour la mise en œuvre du projet. En 2016, la société locale de mobilier urbain AD Display a commencé à installer des abribus numériques dans le centre-ville d'Alger, équipés d'écrans tactiles LCD, de connexions Wi-Fi et de ports USB pour accroître la connectivité dans toute la ville. Parallèlement aux plans visant à étendre l'installation de ces conceptions intelligentes à d'autres villes, AD Display travaille à l'intégration d'un système GPS dans le réseau de bus d'Alger, tandis que d'autres projets incluent le développement d'initiatives d'éclairage intelligent dans les grandes villes.

Stockage de données: Un autre domaine qui a été identifié comme ayant un potentiel de croissance important est celui du stockage de données, avec des efforts en cours pour établir des centres de données tiers à travers le pays. Il existe une opportunité majeure d'externaliser le stockage en Algérie compte tenu de la capacité de rayonnement du pays. Le développement constant de l'hébergement de stockage de données représente une priorité dans la stratégie du pays pour stimuler l'économie numérique [62].

2.2. Synthèse

L'Algérie est dotée d'un potentiel d'infrastructure technologique très important et qui est toujours en développement ; l'extension du réseau internet ainsi que les services d'internet mobile 3G et 4G représente une bonne opportunité d'innovation, ainsi l'opportunité du développement d'une initiative de ville intelligente au sein de ce pays peut prendre place dans le future, et peut être réalisable selon l'émergence des nouvelles technologies.

Dans cette section nous avons étudié l'infrastructure technologique de l'Algérie selon plusieurs facteur (connexion internet, connexion mobile, administration électronique ...) à partir desquels nous avons conclu que l'infrastructure technologique présente de l'Algérie peut servir comme fondation afin d'élaborer notre approche proposée.

2.3. Démarches administratives

Le gouvernement algérien offre une liste de démarches administratives, qui représente un ensemble d'étape que le citoyen doit suivre afin de lui permettre de réaliser une tâche administrative qu'il souhaite effectuer. Chaque étape étant confirmée par l'obtention d'un papier caché et signée par les administrateurs permettant au citoyen de passer à l'étape suivante. La liste suivante représente les démarches administratives implémentées dans notre système et présentes sur le site officiel du gouvernement d'intérieur [63] :

- Etat civil :
 - Acte de naissance : Déclaration de la naissance, Délivrance de l'acte de naissance, Modification de l'acte de naissance.
 - Acte de mariage : Déclaration du mariage, Délivrance de l'acte de mariage.
 - La fiche familiale de l'état civil
 - Le livret de famille
 - Acte de décès : Déclaration d'un décès, La délivrance d'un extrait de décès.
 - Le certificat et fiche de résidence
- Vie Associative :
 - Association
 - Parti politique
- Permis de conduire : Permis probatoire, Permis définitif, Renouvellement du permis de conduire, et Conversion du permis de conduire.
- Carte nationale d'identité
- Vie citoyenne
- Carte d'immatriculation des véhicules
- Passeport biométrique
- Activités réglementées :
 - Commerce non sédentaire
 - Equipements sensibles
 - Cachets et griffes
 - Établissements de divertissements et de spectacles
 - Débits de boissons
 - Confection et fabrication de l'emblème national

- Procédure d'obtention de l'autorisation d'exercice d'activités de gardiennage et de transport de fonds et produits sensibles
- Inhumation et transport des corps Nationaux et Etrangers
- Passeport d'urgence
- Logement

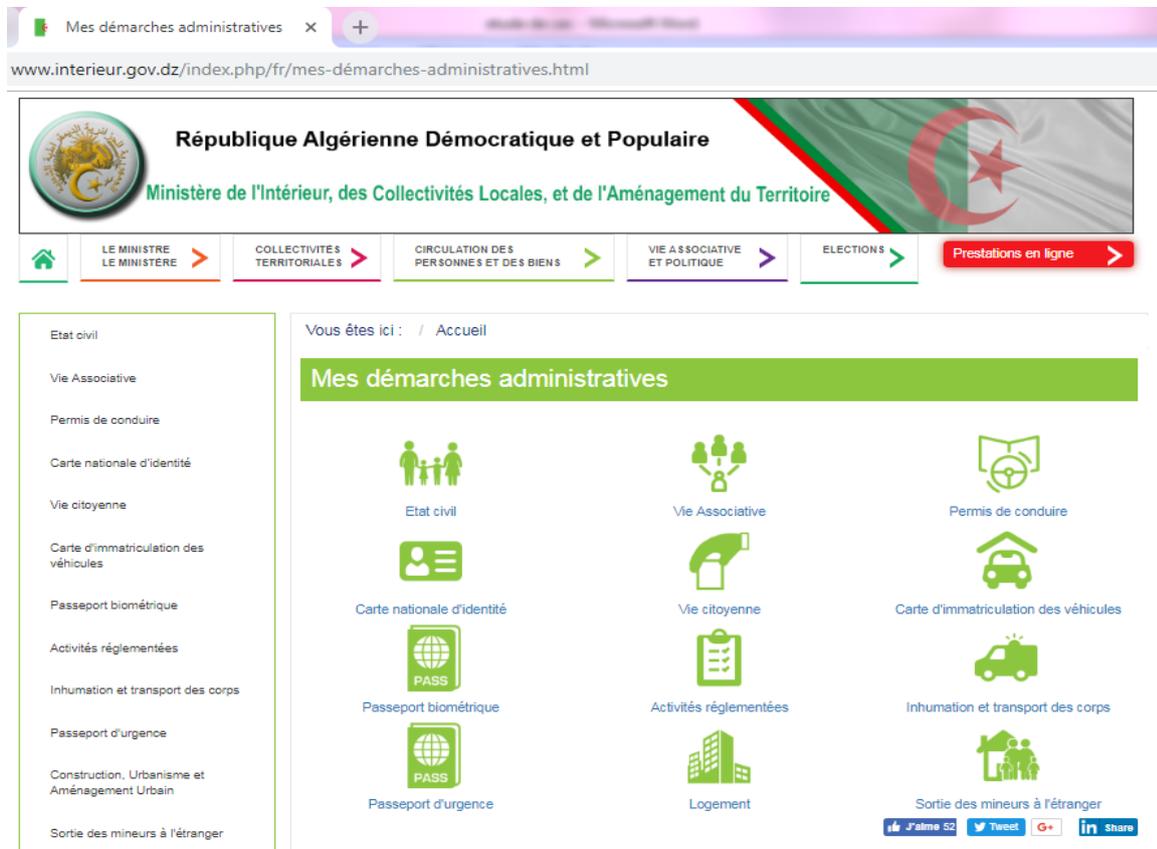


Figure 7.3. Capture de la page des démarches administrative sur le site web du gouvernement d'intérieur de l'Algérie [63].

2.4. Les institutions administratives de Biskra

Les institutions administratives en Algérie sont organisées selon la décomposition du pays en wilaya, commune et dayra chacune de ces régions à sa propre administration ou le citoyen peut se rapprocher afin de déposer des demandes ou de faire des opérations administratives de différents types.

En étant une ville récente et encore sous évolution, la ville de Biskra est une petite ville situé dans le sud-est de l'algérie, elle est composée en elle-même d'un ensemble de 12 Daïra et 33 Communes. les communes qui sont cité par : Aïn Naga, Aïn Zaatout, Besbes, Biskra,

Bordj Ben Azzouz, Bouchagroune, Branis, Chetma, Djemorah, Doucen, Ech Chaïba, El Feidh, El Ghrous, El Hadjeb, El Haouch, El Kantara, El Mizaraa, El Outaya, Foughala, Khenguet Sidi Nadji, Lichana, Lioua, M'Chouneche, Mekhadma, M'Lili, Ouled Djellal, Oumache, Ourlal, Ras El Miaad, Sidi Khaled, Sidi Okba, Tolga, Zeribet El Oued [64].

3. Outils de développement

Afin de développer notre système nous avons fait recours à la technologie Google Cloud endpoints comme framework de base pour notre système pour développer le système back-end et les services REST API, ainsi que l'environnement de développement de l'application mobile du citoyen est celui d'Android studio avec le langage de programmation JAVA.

3.1. Android Studio

Android studio est un environnement de développement (IDE) dédié au développement des applications mobiles sous le système d'exploitation android, proposé par Google, il exploite les capacités intelligentes d'édition de code de IntelliJ IDEA (communauté d'édition). Cet IDE permet également d'éditer et de pré-visualiser les mises Android à travers plusieurs tailles d'écran, différents langages et même des versions différentes de l'API Android. Il comprend trois composants essentiels [65] :

- Android SDK : Le kit de développement (SDK) d'Android est un ensemble complet d'outils de développement.
- Android Virtual Device qui est un émulateur de terminal optimisé pour tester les applications, et intégrer un ensemble de modèles de code.
- La fonction Memory Monitor, un outil d'analyse de la performance pour mieux gérer la mémoire utilisée par l'application au fil du temps.
- Un système de compilation basé sur Gradle qui offre beaucoup de flexibilité et d'extensibilité. Certaines de ses principales caractéristiques est de compiler plusieurs supports pour soutenir les différents types de construction (le débogage debug et la libération release) ou différentes versions de la même application (payantes et gratuites), manipuler des multi-apks, et gérer la dépendance pour les bibliothèques.

- Android DDMS (Dalvik Debug Monitor Service) est un puissant outil de débogage qui permet d'interroger les processus actifs, examiner la pile et le tas, surveiller et mettre en pause les threads actifs et explorer le système de fichier de n'importe quel matériel Android connecté. DDMS fournit également un accès simplifié aux captures d'écran de l'émulateur et aux journaux générés par le LogCat. Le DDMS est complètement intégré via le plugin ADT [65].

3.2. Le langage JAVA

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de réaliser des logiciels compatibles avec de nombreux systèmes d'exploitation. Java offre aussi la possibilité de réaliser des programmes pour téléphones mobiles et assistants personnels PDA. Il possède plusieurs caractéristiques qui le rendent parmi les meilleurs outils de développement des logiciels: simple, distribué, interprété, indépendant de l'architecture, portable, robuste, sûr, dynamique, multithreadé [66].

3.3. Les APIs publiques

Les API publiques sont désormais considérées comme le mécanisme le plus efficace pour intégrer deux applications. En générale, les API sont hébergées dans une application côté serveur et exposées aux clients à consommer. Les clients pourraient être d'autres applications côté serveur, des applications mobiles natives ou même des applications de navigateur (mobiles ou de bureau). Lorsque nous parlons d'une API publique, il y'a quelques caractéristiques communes [67]:

- Ils sont disponibles via un protocole populaire et bien défini, généralement HTTP.
- Il utilise JSON ou XML comme format de données.

Le protocole REST utilise des verbes HTTP divers de manière efficace pour effectuer des opérations de base CRUD ou en termes plus simples:

- Liste (ou obtenir) tous les enregistrements
- Obtenir un ou plusieurs enregistrements en fonction de certains critères de recherche ou identifiant (s)
- Créer un enregistrement
- Mettre à jour un enregistrement

- Supprimer un enregistrement

Un point clé des services Web est qu'ils offrent une couche de séparation entre le client et le serveur en termes de technologie utilisée. Les applications client sont totalement indépendantes du langage / environnement de programmation utilisé côté serveur. HTTP est le langage commun qui lie les deux ensembles et, par conséquent, l'implémentation d'API côté serveur peut être écrite en Java, mais il y'a toujours la possibilité d'invoquer la même API HTTP via un client Java, un client iOS ou même un client Web [67].

3.4. Google Cloud Endpoints

Google Cloud Endpoints, est une solution de Google pour aider à créer une API publique pour une application App Engine. Il s'agit d'un système de gestion d'API qui aide à sécuriser, surveiller, analyser et définir des quotas sur les API en utilisant la même infrastructure que celle employée par Google pour ses propres API. Une fois l'API déployé sur Endpoints, le portail Cloud Endpoints peut être utilisé pour créer un portail des développeurs, un site Web auquel les utilisateurs de l'API peuvent accéder pour consulter de la documentation et interagir avec l'API. Cloud Endpoints fournit également des installations pour générer des bibliothèques clientes pour Android et iOS, facilitant ainsi la tâche pour intégrer la fonctionnalité de l'application back-end dans les applications mobiles sur Android et iOS.

Le but de Cloud Endpoints est de relier l'ensemble des applications (web, mobile...) en générant une API. C'est une fonctionnalité récente de Google App Engine. Google résume de façon très synthétique le principe dans un seul schéma [67]:

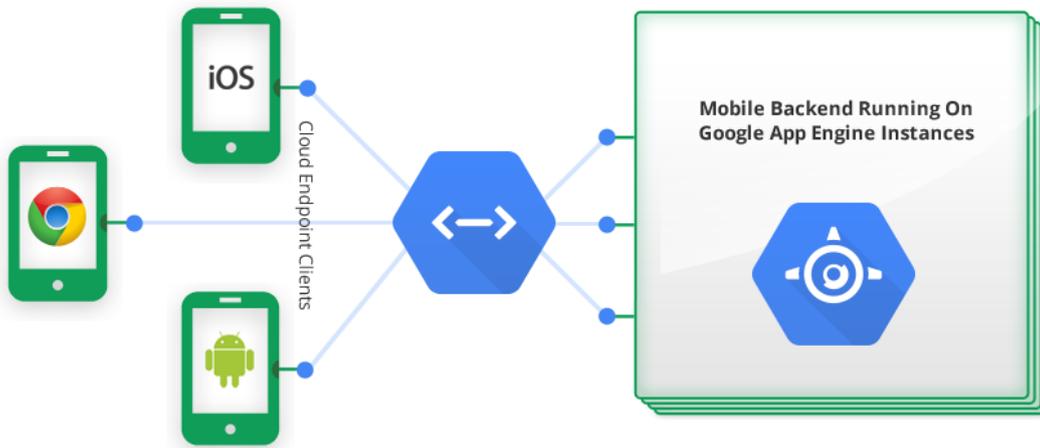


Figure 7.4. schéma décrivant la solution Google Cloud Endpoints [67]

- Sur la droite, il y'a le *backend*. C'est l'application qui tourne sur App Engine. C'est elle qui effectue tous les calculs et tout le stockage (sur le Datastore par exemple).
- Sur la gauche, il y'a les *frontends*. Ce sont les interfaces de l'application dont se servent les utilisateurs. Ces dernières années, les interfaces se sont multipliées : Web (navigateur), application Android, application iOS...

L'objectif ici est de séparer une application en deux parties logiques : le backend (en gros le modèle et le contrôleur) et le frontend (en gros la vue).

Le Framework Cloud Endpoints est un framework Web pour les environnements d'exécution Python et Java standards App Engine. Il fournit les outils et les bibliothèques requis pour permettre de générer des API REST et des bibliothèques clientes pour l'application.

À l'instar d'autres frameworks Web, Endpoints Frameworks est compatible avec les détails de communication de bas niveau des requêtes et réponses HTTP pour l'application. Lorsqu'un client envoie une requête vers l'API, Endpoints Frameworks achemine l'URL de la requête vers la fonction ou la méthode du code qui traite la requête. Endpoints Frameworks convertit la valeur d'affichage en JSON et envoie la réponse. Ajouter des métadonnées au code source (en utilisant des annotations en Java). Les métadonnées définissent la surface des API REST pour l'application [67].

4. Implémentation

Notre système est basé sur la technologie Cloud Back-end as a service qui relie les applications au stockage et au traitement du back-end cloud par génération et offre de services REST API comme méthode de communication, alors que l'application utilise le GPS pour obtenir la localisation géographique de l'appareil mobile du citoyen et utilise les cartes du fournisseur OpenStreetMaps.

Nous avons choisi l'administration de la ville de Biskra comme domaine d'application pour illustrer notre système; c'est une ville situé au sud-est de l'Algérie.

L'objectif de ce système est d'offrir des services administratifs aux citoyens, dans ce but, nous avons utilisé comme exemple les institutions administratives de la ville de Biskra, les services offerts et les données relatives telles que les heures d'ouverture / fermeture, les emplacements ...etc. Toutes ces données sont stockées dans le data-store du système back-end et transféré au front-end (application) via les appels des services Rest API [2].

4.1. Le système back-end

La partie Back-end de notre système représente le côté serveur s'exécutant sur l'instance de Google AppEngine. Elle gère un data-store de données contenant une grande quantité d'informations sur la ville (voir figure 7.5) ainsi qu'une liste des services API REST disponibles et accessibles à partir des appels des applications mobiles connectées des citoyens, comme le montre la figure 7.6 ci-dessous [2]:

Key	Write Ops	ID/Name	latitude	longitude	name	type
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYBgw	10	6	34.8478329	5.7114564	daira of biskra	daira
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYBww	10	7	34.8508899	5.703101	public education direction	public education
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYCAw	10	8	34.8524925	5.723468	biskra post office	post office
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYCQw	10	9	34.8524925	5.723468	wilaya of biskra	wilaya
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYCgw	10	10	34.8463995	5.6957839	national employment agency	employment
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYCww	10	11	34.8647295	5.7150744	public works direction	public works
ag9teUFwcGxpY2F0aW9uSWRyCgsSBHNpdGUYDAw	10	12	34.8519995	5.7315668	apc Biskra	municipality

Figure 7.5. Le data-store du système back-end [2]

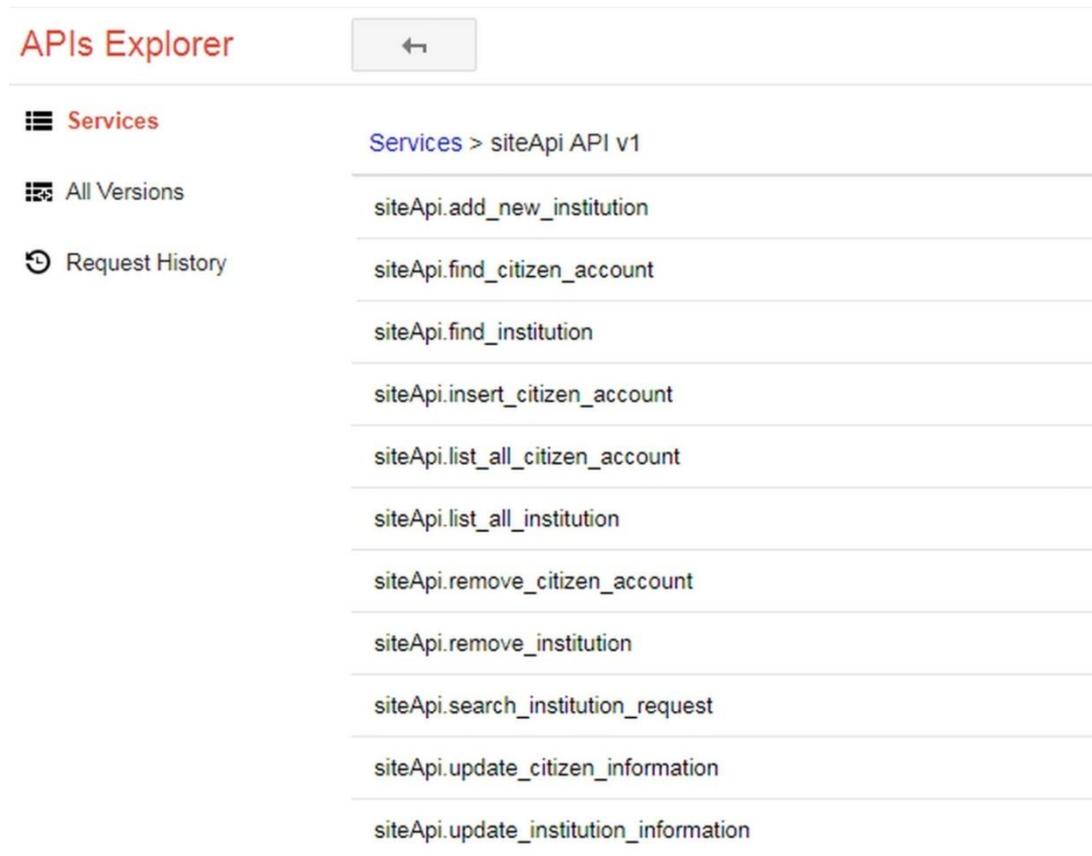


Figure 7.6. Les services REST API disponibles du système back-end [2]

L'une des étapes essentielles de notre système consiste à la création d'une end-point qui fonctionnera comme service Rest API du back-end de notre système, le code suivant représente la déclaration d'une API en utilisant les annotations:

```
package com.example.myapplication.backend;

import com.google.api.server.config.ApiClass;
import com.googlecode.objectify.ObjectifyService;

@Api(
    name = "siteApi",
    version = "v1",
    namespace = @ApiNamespace(
        ownerDomain = "backend.myapplication.example.com",
        ownerName = "backend.myapplication.example.com",
        packagePath = ""
    )
) // Annotation qui indique que la classe fournit des fonctionnalités d'une API

@ApiClass (resource = "site")

public class siteEndpoint {

    private static final Logger logger = Logger.getLogger(siteEndpoint.class.getName());

    private static final int DEFAULT_LIST_LIMIT = 20;

    static {

        ObjectifyService.register(site.class);

    }

}
```

Chaque API Rest à un ensemble de méthodes qui représentent des fonctions qu'elle effectue, présenté ci-dessous le code d'une méthode de l'API qui consiste à rechercher une institution à partir de son identifiant dans le datastore, elle utilise la méthode GET du protocole HTTP:

```
@ApiMethod(  
    name = "get",  
    path = "institution/{id}",  
    httpMethod = ApiMethod.HttpMethod.GET)  
  
public institution get(@Named("id") Long id) throws NotFoundException {  
    logger.info("Getting site with ID: " + id);  
    institution = ofy().load().type(institution.class).id(id).now();  
    if (institution == null) {  
        throw new NotFoundException("Could not find institution with ID: " + id);  
    }  
    return institution;  
}
```

Chacun des services API dispose d'un ensemble de méthodes. Les méthodes peuvent être directement exécutées à partir de l'explorateur d'API Google et le résultat s'affiche comme suit:

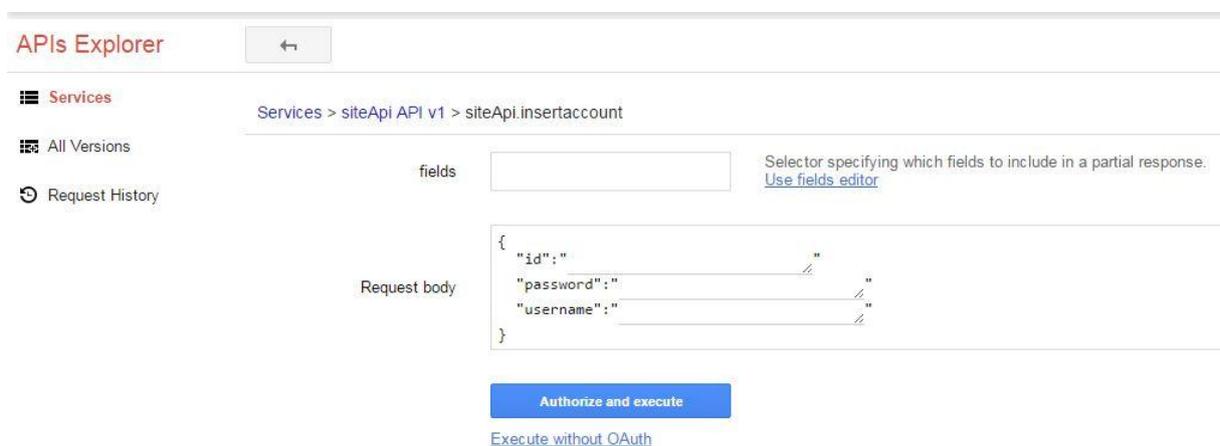


Figure 7.7. Capture d'écran de l'exécution de la méthode

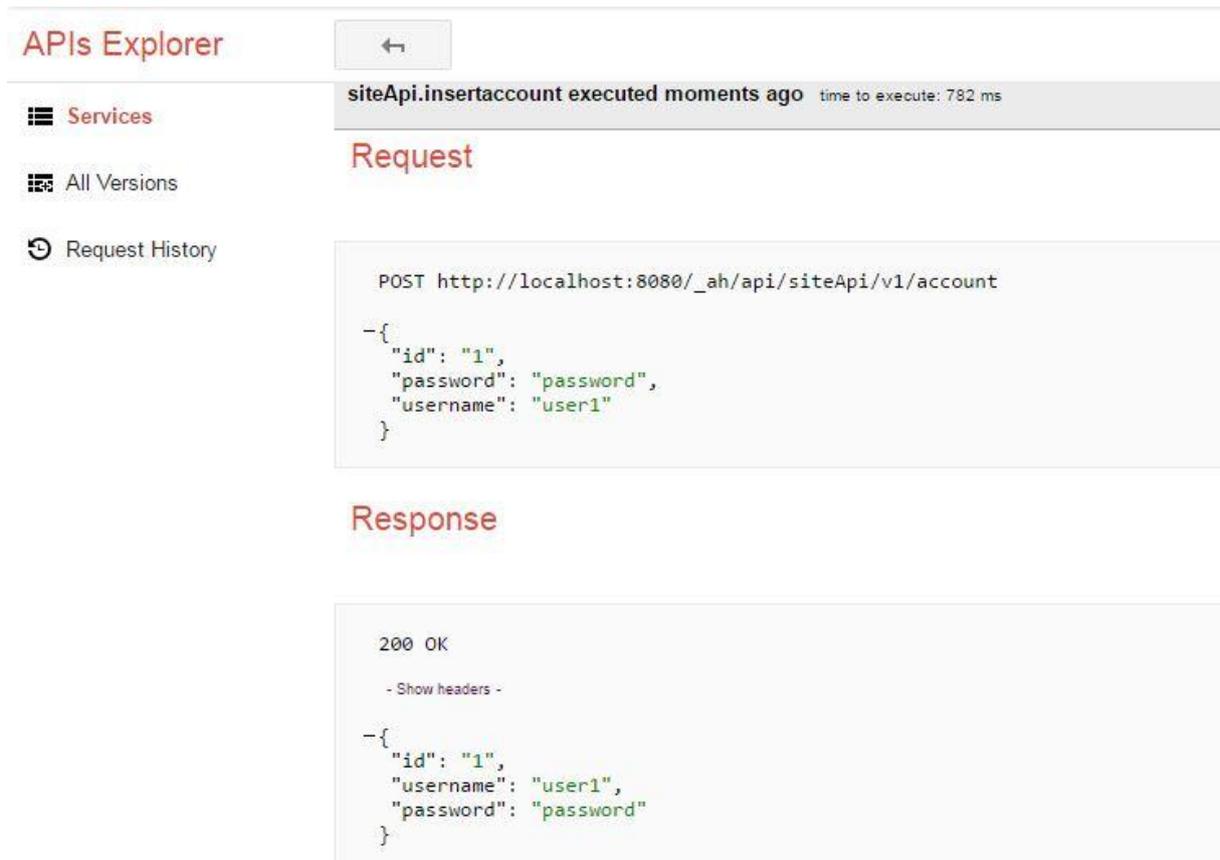


Figure 7.8. Capture d'écran du résultat de l'exécution

4.2. Le système front-end

La partie front-end du système est, en fait, une application mobile installée sur l'appareil mobile connecté du citoyen.

Notre application possède une activité d'accueil qui représente la première activité permettant l'identification et la création de profils qui peuvent être cruciaux pour sécuriser le système et exploiter pleinement ses capacités. Dans le processus d'authentification, l'application envoie les informations du formulaire rempli via le service appelé au système back-end pour vérification et renvoie les résultats.

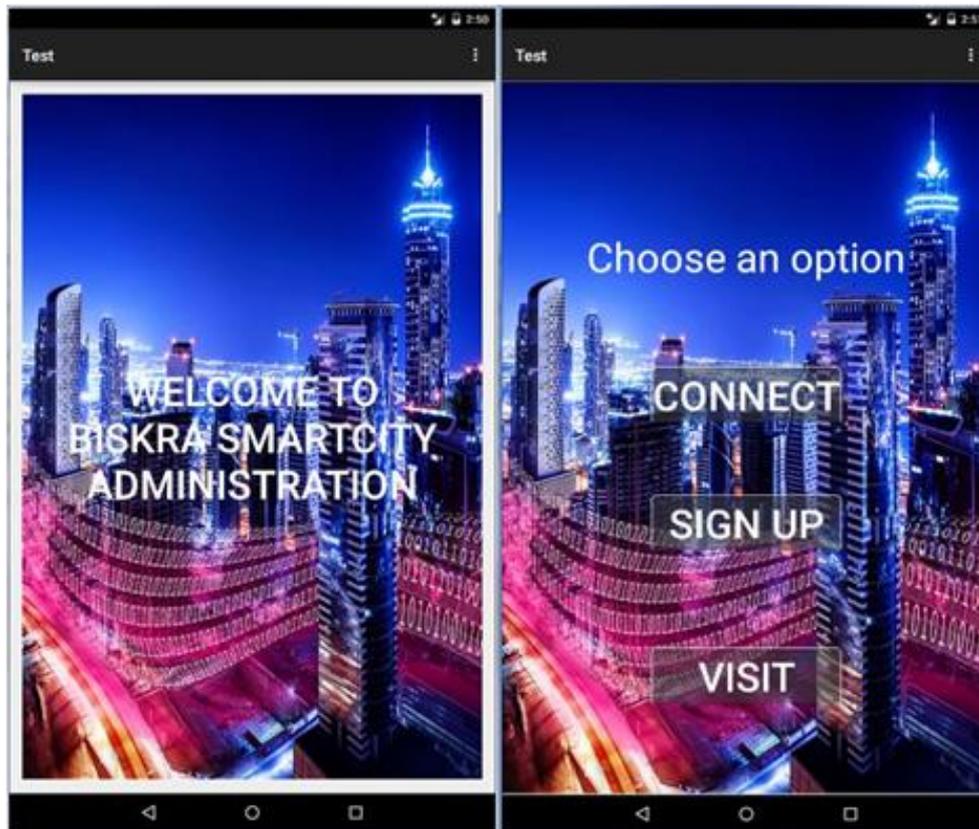


Figure 7.9. Capture d'écran de l'activité d'accueil

L'application permet le citoyen tout d'abord de créer un compte ainsi qu'un profil afin de l'identifier, cependant son objectif principal réside dans la localisation et le partage des informations des institutions administratives, à cet effet un citoyen peut choisir entre sélectionner l'un des événements sociaux présentés pour lui sous forme d'une liste déroulante afin d'obtenir comme réponse des recommandations de services comme le montre la figure 7.10.

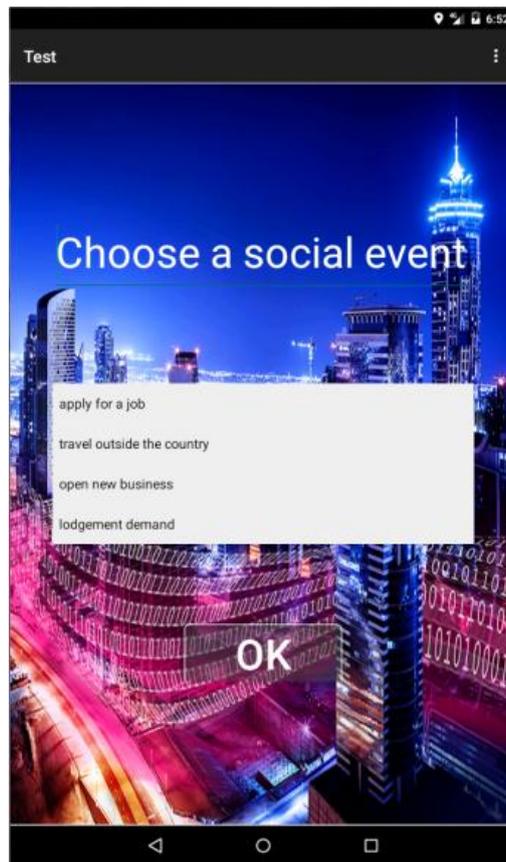


Figure 7.10. Capture d'écran de l'activité des événements sociaux

Ou bien, le citoyen peut choisir de visualiser l'ensemble de la carte de la ville ou rechercher une institution précise, la recherche peut être effectuée selon le nom de l'institution, sa localisation, ou son type ex: municipalité, bureau de poste....etc. Tout en détectant en même temps la position géographique du citoyen par GPS à chaque événement de changement de position, comme le montre la figure 7.11 [2]:

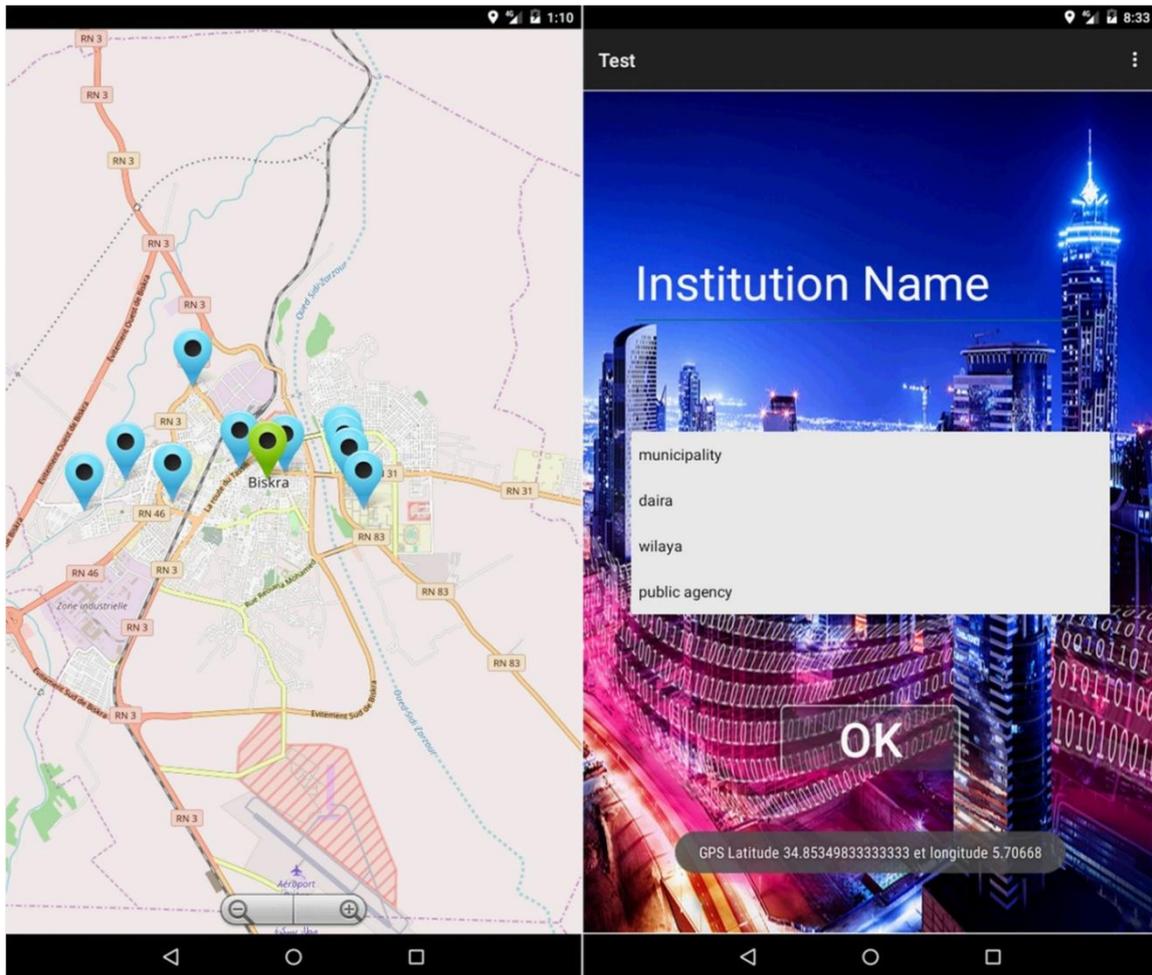


Figure 7.11. Capture d'écran de l'activité de la carte et de recherche pendant un événement de changement de position [2]

Dès que l'activité de recherche est affichée, chaque changement de l'emplacement géographique du téléphone mobile sera capturé par l'application et enregistré comme emplacement actuel du citoyen.

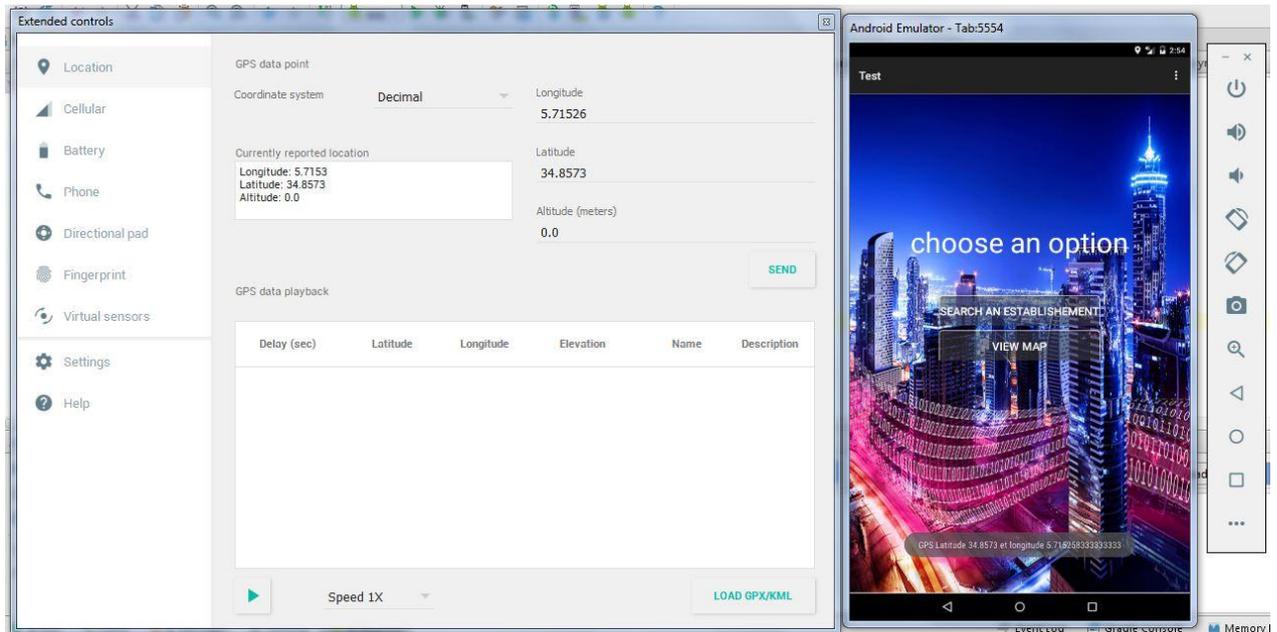


Figure 7.12. Capture d'écran d'un changement de localisation

À chaque requête, le citoyen effectue la demande et les données de résultat sont reçues et transmises via les API Rest, y compris la création de profil, la recherche d'institution, la visualisation de cartes...etc., l'application appelle le service API Rest approprié du système backend, alors que d'autres fonctions telles que le filtrage des données et les recommandations sont toutes exécutées sur la partie back-end fonctionnant sur le cloud.

Le code suivant représente un appel d'un service à partir du serveur back-end :

```

protected List<Site> doInBackground(Context ... params) {
    if(myApiService == null) {
        SiteApi.Builder builder = new
            SiteApi.Builder(AndroidHttp.newCompatibleTransport(),
                new AndroidJsonFactory(), null)
            // Options pour exécuter sur le serveur de développement d'applications local
            // 10.0.2.2 est l'adresse IP de l'hôte local dans l'émulateur Android
            .setRootUrl("http://10.0.2.2:8080/_ah/api/")
            .setGoogleClientRequestInitializer(new GoogleClientRequestInitializer() {
                public void initialize(AbstractGoogleClientRequest<?>
                    abstractGoogleClientRequest) throws IOException {
                    abstractGoogleClientRequest.setDisableGZipContent(true);
                }
            });
        // Fin des options pour le serveur de développement d'applications
        myApiService = builder.build();
    }
    context = params[0];
    try {
        TrafficStats.setThreadStatsTag(0x726F6FFF);
        try {
            // Faire une demande réseau à l'aide du client http.
            return myApiService.list().execute().getItems();
        } finally {
            TrafficStats.clearThreadStatsTag();
        }
    } catch (IOException e) {
        return Collections.EMPTY_LIST;
    }
}

```

Les institutions sont présentées sous forme de marqueurs sur la carte ou sous forme de liste de recommandations; un citoyen peut également consulter les informations sur l'institution et les services en ligne recommandés et disponibles (figure 7.13) [2].

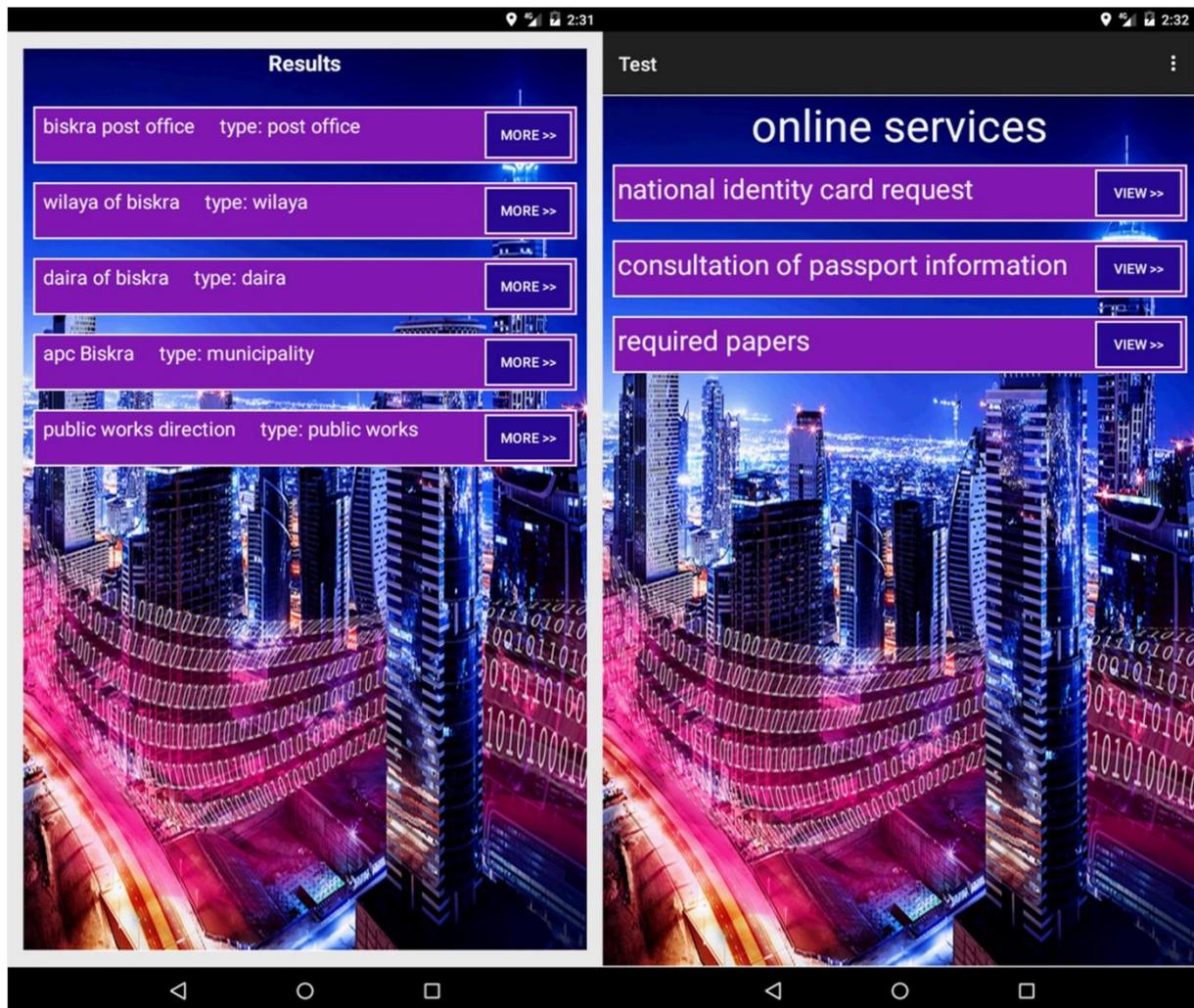


Figure 7.13. Capture d'écran des informations sur l'institution administrative et ses services en ligne [2]

Après avoir consulté les informations détaillées, le citoyen peut également consulter les services en ligne recommandés pour lui afin d'exécuter les services offerts par cette institution, la figure 7.14 montre un exemple de services en ligne appartenant à l'institution de commune de la ville de biskra.

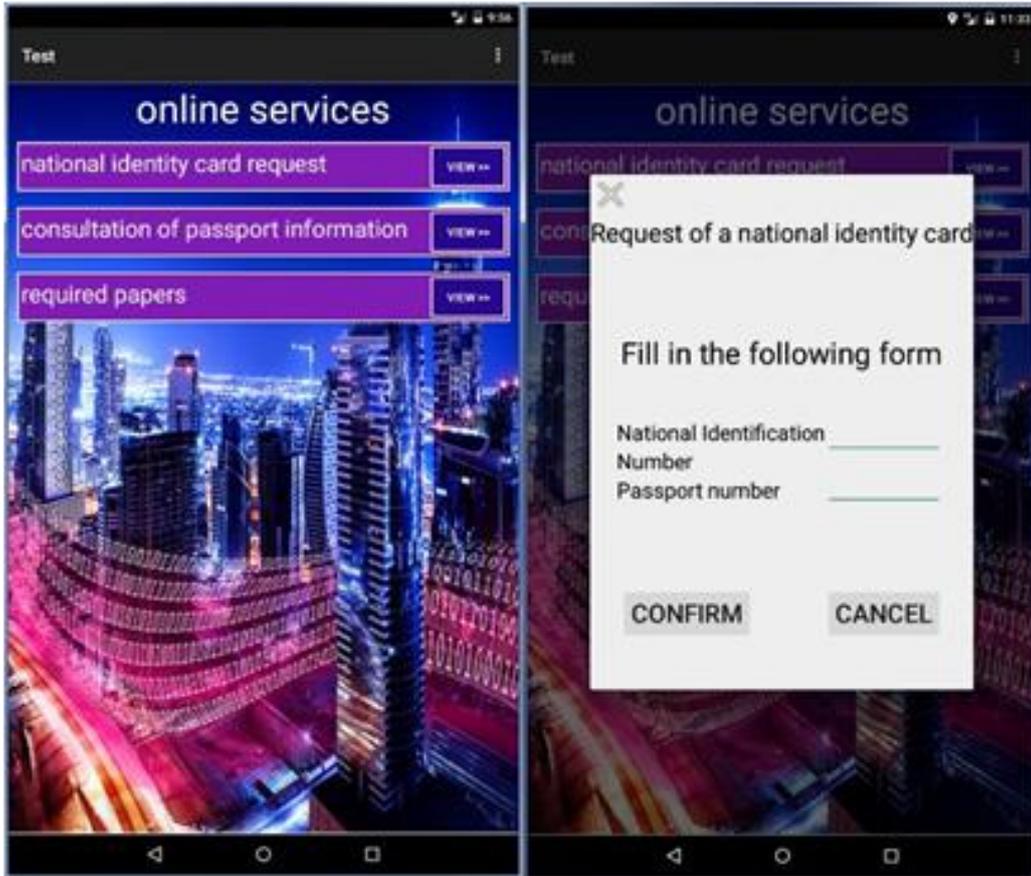


Figure 7.14. Capture d'écran d'un service en ligne

5. Évaluation des résultats et des performances

Cette section illustre les tests que nous avons effectués pour évaluer l'approche proposée sur la base de deux aspects critiques: l'interactivité et la scalabilité, les expériences présentées sont réalisées pour estimer et étudier le comportement du système dans des cas différents.

5.1. Estimation de la scalabilité

Dans le but d'évaluer la scalabilité de notre système, nous avons choisi d'étudier le rapport entre le temps de réponse et le nombre de requêtes réparties dans le temps à l'aide d'un outil de benchmark d'APIs, à cet effet nous avons réalisé des séries de tests de charge pour notre système sur des échantillons de requêtes en augmentant progressivement le nombre de requêtes, afin d'étudier le comportement du système dans les grandes charges de travail changeantes, Les résultats obtenus sont présentés dans les figures suivantes qui montrent les valeurs du temps de réponse en secondes par rapport au nombre de requêtes soumises [2].

Comme phase initiale, nous avons spécifié un échantillon de 10 requêtes avec 2 requêtes concurrentes; les résultats que nous avons obtenus sont présentés dans la figure 7.15 [2].

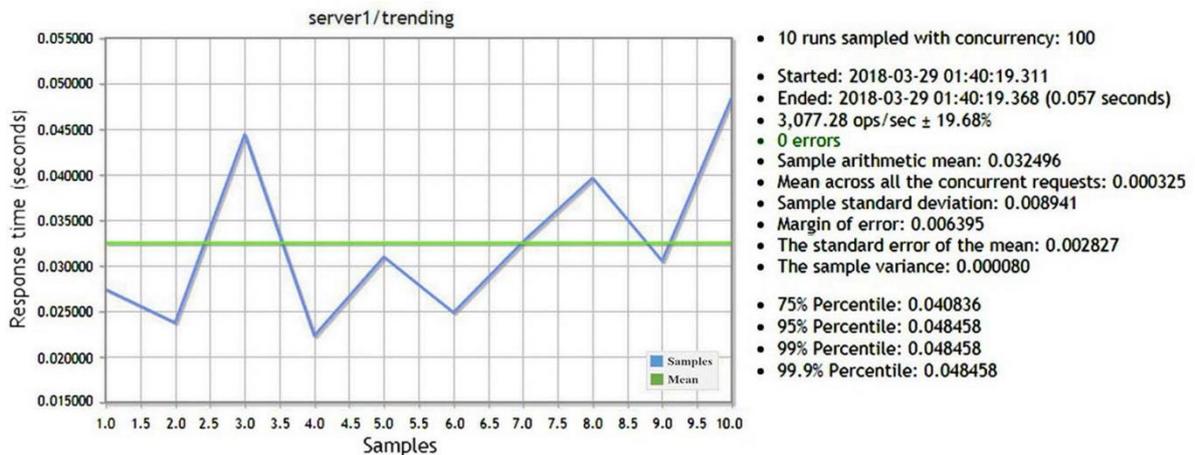


Figure 7.15. Scalabilité du système lors du premier test de charge [2]

À partir de l'analyse de la figure obtenue, nous avons remarqué que les valeurs du temps de réponse augmentent, la moyenne du temps de réponse sur tous les échantillons a été mesurée par 0,032496 secondes et la moyenne sur les demandes concurrentes était de 0,000325 secondes.

Dans un deuxième test, nous avons augmenté le nombre d'échantillons jusqu'à 100 requêtes, avec 10 requêtes concurrentes à exécuter en parallèle, la figure 7.16 montre les résultats [2]:

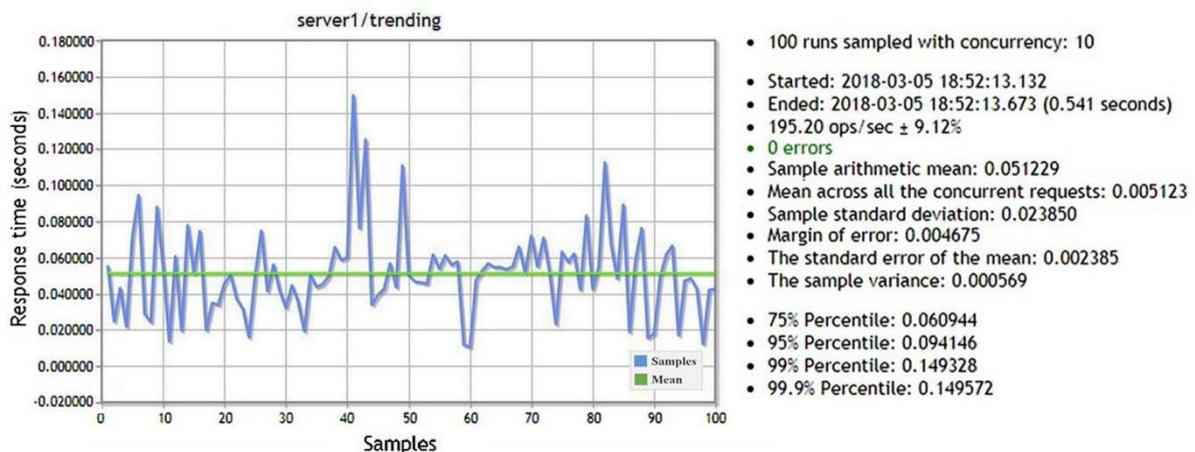


Figure 7.16. Scalabilité du système lors du deuxième test de charge [2]

Les résultats obtenus montrent que le temps de réponse moyen a augmenté par rapport au cas précédent, pour tous les échantillons, il a été mesuré de 0,051229 secondes et pour les demandes concurrentes de 0,005123 secondes [2].

Et pour le test final, nous avons augmenté jusqu'à 200 requêtes avec 20 requêtes concurrentes; les résultats finaux sont présentés dans la figure 7.17:

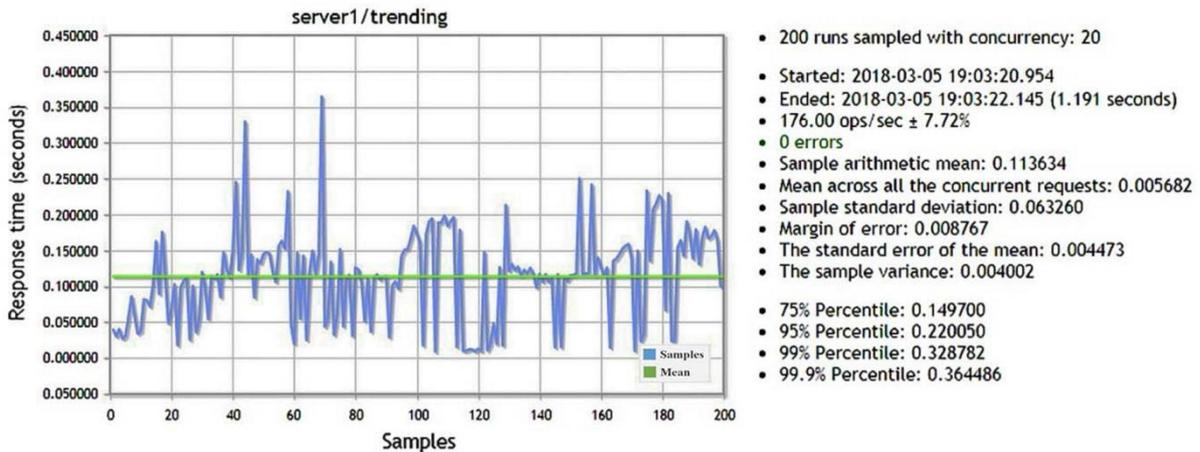


Figure 7.17. Scalabilité du système lors du troisième test de charge [2]

Les résultats obtenus montrent que le temps de réponse moyen a augmenté jusqu'à la valeur de 0,113 634 secondes pour tous les échantillons et de 0,005682 secondes pour les demandes concurrentes [2].

La répartition des requêtes à travers le temps a été générée automatiquement et correspond aux résultats obtenus; le temps de réponse est conforme au nombre de demandes, la figure 7.18 présente la proportion des temps de réponse dans chacun des cas précédents:

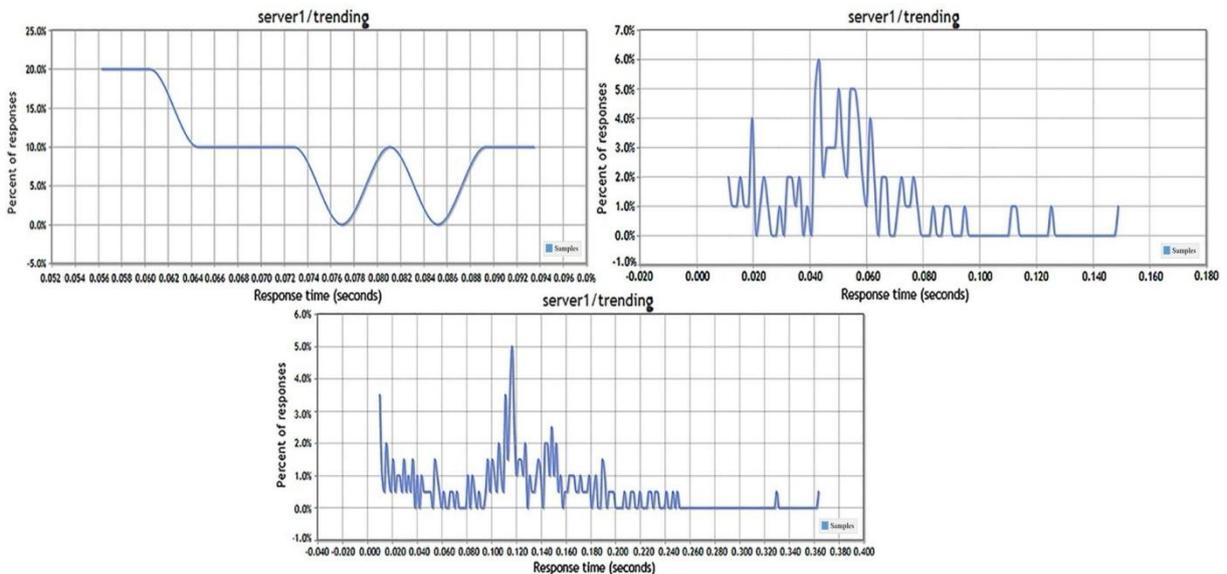


Figure 7.18. Répartition des réponses dans le test initial (en haut à gauche), le deuxième test (en haut à droite) et le test final (en bas) [2]

On peut remarquer que les valeurs élevées dans les tableaux de distribution et les tests de charge correspondent à l'exécution des requêtes simultanées, aussi, la mesure de réponse était en augmentation progressive mais la valeur restait acceptable dans tous les tests: elle ne dépassait pas 0,45 secondes pour toutes les requêtes y compris les requêtes simultanées (voir les figures 7.15, 7.16, 7.17 ci-dessus) [2].

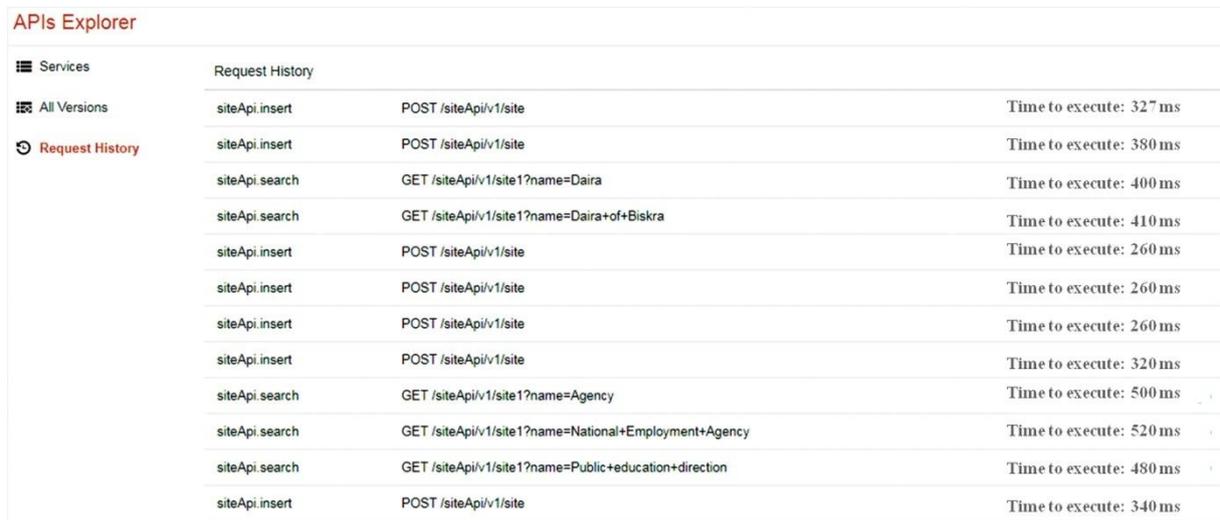
A partir des résultats obtenus et en comparant les performances de notre système dans les différents cas nous constatons une diminution évidente des performances qui s'exprime par une augmentation du temps de réponse, ce comportement est attendu dans les circonstances données et les charges toujours croissantes, même ainsi, la diminution n'était pas majeure, ce qui était encourageant pour le comportement futur, et le système semblait toujours avoir tendance à se stabiliser à un certain point, ce qui garantissait une réponse acceptable et donc un certain degré de scalabilité.

Comme remarque secondaire, nous avons également constaté que notre système a bien traité les demandes simultanées; la moyenne à travers les différents cas n'augmentait pas beaucoup et les valeurs étaient toujours proches [2].

5.2. Évaluation de l'interactivité

Nous avons choisi le temps de réponse et les données transférées comme critères pour évaluer l'interactivité de notre système. Ce système opte pour proposer des réponses le plus rapidement possible afin de maintenir l'engagement citoyen.

Dans notre cas, l'interaction entre le système central et l'application citoyenne est assurée par des services Rest API qui garantissent un temps de réponse raisonnable, la figure 7.19 suivante montre l'historique des requêtes déjà effectuées et le temps d'exécution de chacune d'entre elles [2][3].



APis Explorer			
Services	Request History		
All Versions	siteApi.insert	POST /siteApi/v1/site	Time to execute: 327 ms
Request History	siteApi.insert	POST /siteApi/v1/site	Time to execute: 380 ms
	siteApi.search	GET /siteApi/v1/site1?name=Daira	Time to execute: 400 ms
	siteApi.search	GET /siteApi/v1/site1?name=Daira+of+Biskra	Time to execute: 410 ms
	siteApi.insert	POST /siteApi/v1/site	Time to execute: 260 ms
	siteApi.insert	POST /siteApi/v1/site	Time to execute: 260 ms
	siteApi.insert	POST /siteApi/v1/site	Time to execute: 260 ms
	siteApi.insert	POST /siteApi/v1/site	Time to execute: 320 ms
	siteApi.search	GET /siteApi/v1/site1?name=Agency	Time to execute: 500 ms
	siteApi.search	GET /siteApi/v1/site1?name=National+Employment+Agency	Time to execute: 520 ms
	siteApi.search	GET /siteApi/v1/site1?name=Public+education+direction	Time to execute: 480 ms
	siteApi.insert	POST /siteApi/v1/site	Time to execute: 340 ms

Figure 7.19. Historique des requêtes d'appels des services API [2]

Comme le montre la figure précédente, les demandes simples telles que l'insertion de compte, la modification de profil, etc. ne peuvent prendre que 0,3 s à 0,5 s, mais les plus compliquées telles que les recherches d'institutions ou les exécutions de services en ligne peuvent varier de 0,4 s à 0,6 s selon besoins, de cela nous pouvons conclure que le temps de réponse du système global est raisonnablement décent par rapport aux fonctions complexes qu'il offre [2][3].

En ce qui concerne les données transférées, étant déployées sur le Cloud Data Store, elles sont accessibles uniquement via des requêtes HTTP, c'est pourquoi nous avons utilisé l'outil de studio Android le Network Profiler afin d'inspecter le trafic réseau. Il permet d'afficher les statistiques du réseau sur les canaux de communication entrants et sortants entre l'application et le serveur.

La figure 7.20 ci-dessous représente les statistiques de réseau obtenues lors du fonctionnement de notre système. L'axe y dans le diagramme du réseau représente la vitesse de transfert de la transmission mesurée en Kilo Octet / seconde, tandis que l'axe x représente le temps écoulé en secondes. RX / TX réfèrent aux données transmises / reçues, les données présentées sur le diagramme ont trois étiquettes afin de différencier les processus comme suit: les espaces rouges représentent la recherche d'institutions, le bleu représente la demande d'extraction de toutes les institutions de la ville, et le gris représente le chargement des tuiles de carte [2][3].

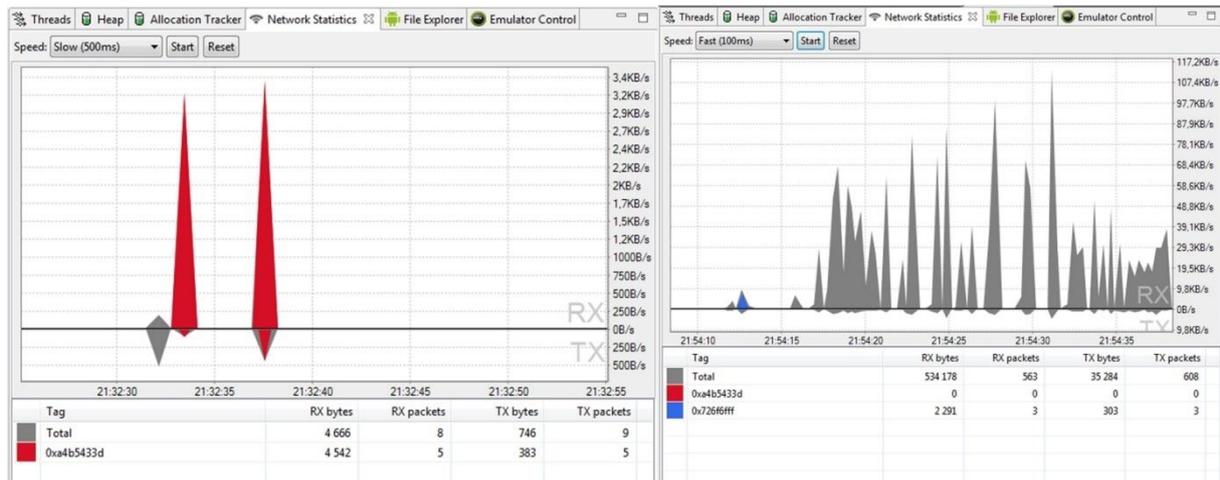


Figure 7.20. Captures d'écran des statistiques du réseau du système dans les activités de visualisation de la carte (à droite) et de recherche (à gauche) [2]

Comme nous pouvons conclure du diagramme précédent, les interactions sont brèves et instantanées, ce qui contribue à diminuer le comportement de décharge de la batterie. En outre, nous pouvons observer que les données transmises ne dépassent pas 117 Ko/s, ce qui représente une quantité acceptable.

Comme il y a toujours de la place à l'amélioration, notre système répond aux exigences essentielles d'un système de ville intelligente en termes de temps de réponse et de transfert de données mais pourrait bénéficier d'autres technologies afin d'améliorer de plus en plus ses performances [2][3].

6. Conclusion

Dans ce chapitre nous avons présenté les étapes du développement de notre système d'administration de smart city, les outils utilisés ainsi que les différents résultats que nous avons obtenus à partir des tests de scalabilité et d'interactivité (temps de réponse et données échangées). Les résultats obtenus étaient bonnes et satisfaisante à un certains degré et ont peut dire qu'ils ont atteint les exigences essentielles d'un système d'administration pour une ville intelligente.

CONCLUSION GENERALE

1. DISCUSSION

Le contexte de smart city est un domaine multidisciplinaire où un système devient complexe en raison de ces domaines multiples et volumineux en raison de l'énorme quantité de données et connaissances nécessaires à sa fonctionnalité, et par conséquent la scalabilité dans ce contexte est indispensable, une ville doit développer un système qui grandit avec son développement. Dans ce contexte, la scalabilité avec de grands ensembles de données du monde réel représente un problème vital et principal pour les systèmes de recommandation des villes intelligentes aujourd'hui; tandis que la quantité de données utilisées en entrée augmente rapidement à mesure que de nouveaux citoyens et services sont ajoutés et que la taille du comportement citoyen stocké peut facilement atteindre les Téra octets par jour, il devient difficile de gérer des ensembles de données énormes et dynamiques produites par les interactions citoyens-services telles que les préférences, les notes et les avis. Il est même possible que lorsque certains algorithmes de recommandation sont appliqués à des ensembles de données relativement petits, ils fournissent les meilleurs résultats, mais reflètent un comportement inefficace ou pire sur de très grands ensembles de données tout en essayant de répondre de manière interactive en moins d'une seconde pour maintenir l'engagement des citoyens. Le défi qui se pose ici est de concevoir des systèmes efficaces capables de gérer de tels ensembles de données à grande échelle.

Dans cette intention, nous avons développé un système de recommandation sensible au contexte avec l'approche back-end as a service sur cloud afin d'assurer la bonne utilisation des données de la manière la plus scalable via des services REST API.

Afin de tester l'efficacité de cette approche, nous avons réalisé une série d'expériences sur le système avec différentes quantités d'échantillons de requêtes, les résultats obtenus n'étaient pas parfaites mais globalement acceptables; A partir des résultats précédents nous pouvons déduire que l'approche back-end as a service a beaucoup amélioré la scalabilité de notre système, les performances dans tous les cas avec des quantités différentes de requêtes parallèles ont eu tendance à stabiliser la réponse après tout, ce qui signifie que le système est resté scalable et pourrait gérer une grande quantité de charges de travail, mais au meilleur de nos connaissances, peu importe la durée de résistance d'un système centralisé, il finira par une réponse excessive à un moment donné, de sorte que le besoin d'une architecture décentralisée

reste nécessaire et une solution définitive de la scalabilité dans ces grands systèmes afin d'éviter les problèmes de surcharge du serveur.

Le partage des informations et des services administratifs représente un aspect important dans le contexte d'une ville intelligente, il peut affecter la qualité de vie du citoyen en diminuant le temps que le citoyen perd en recherchant une institution ou un service spécifique, une personne peut prendre des heures voire toute une journée de navigation et d'interrogation juste pour trouver un endroit même s'il connaît déjà les routes de la ville. En fonction des résultats obtenus et à cet effet notre système vise à réduire l'effort et le temps de la vie et les tâches quotidiennes du citoyen.

En termes d'interactions, notre système offrait une interaction en temps réel avec les citoyens comme on peut en déduire des mesures de temps de réponse obtenues, on peut également noter que l'approche de recommandation contextuelle avait vraiment optimisé les données échangées ce qui est un avantage pour notre système notamment la front-end étant conçu comme une application mobile, cela pourrait réduire l'utilisation de la mémoire pour les smartphones, ce qui représente une limitation connue dans les appareils mobiles.

A partir de ces résultats on peut dire que nous avons atteint notre objectif principal de cette approche qui était également de permettre à la ville d'interagir directement avec ses citoyens via des services d'accès instantané et des interfaces numériques plutôt que via des bureaux avec des formulaires papier et de longues files d'attente.

Ainsi, nous pouvons en déduire que la communication, le partage d'informations et les données en temps réel sont tous des exigences importantes pour le développement d'applications et de services pour les smart cities. L'administration publique reste donc un objectif prometteur que les villes intelligentes tentent aujourd'hui d'atteindre de la manière la plus optimale possible.

2. PERSPECTIVES

Le contexte de la ville intelligente étant un domaine récent et intéressant a fait l'objet de cette thèse, nous nous sommes concentrés sur la partie administration dans laquelle la section introductive contenait un ensemble de définitions et de notions de base sur les villes intelligentes, l'administration intelligente et les citoyens, suivi d'une analyse des travaux connexes, une description des approches adoptées et du système réalisé.

Dans ce travail de recherche, nous avons présenté une approche de recommandation contextuelle pour une administration de ville intelligente, notre objectif était d'améliorer le partage d'informations administratives et la qualité des services offerts aux citoyens en les adaptant à leur contexte et en même temps à résoudre le problème de scalabilité dont souffre ce type de systèmes en proposant un framework basé sur l'approche back-end as a service pour exploiter ses fonctionnalités, notamment les services API REST et la mise à l'échelle automatique dans le contexte de la ville intelligente.

Notre système peut être considéré comme une initiative pour un système administratif de ville intelligente, il a encore un espace pour plus d'amélioration, c'est pourquoi nous avons défini un ensemble de perspectives qui peuvent être résumées dans les points suivants:

- Utilisation de plusieurs sources de données et permettre la collecte automatique des données et des mises à jour à partir du Web
- Proposer des approches pour l'analyse des données, l'agrégation et la combinaison de plusieurs services sur le cloud
- Bénéficiez des technologies IOT en incluant plusieurs types de senseurs ou capteurs pour améliorer la qualité et les données d'entrée en temps réel.

Références

- [1] Kumar, B., & Sharma, N. (2016). Approaches, issues and challenges in recommender systems: a systematic review. *Indian Journal of Science and Technology*, 9(47), 1-12.
- [2] Hoadjli, A., & Rezeg, K. (2019). A scalable mobile context-aware recommender system for a smart city administration. *International Journal of Parallel, Emergent and Distributed Systems*, 1-20.
- [3] Hoadjli, A., & Rezeg, K. (2018). A Smart City System using Backend as a Service Approach: Biskra City Case Study. In 2018 Fifth International Symposium on Innovation in Information and Communication Technology (ISIICT) (pp. 1-7). IEEE.
- [4] FG-SSC, I. T. U. T. (2014). Technical report on smart sustainable cities: An analysis of definitions. United Nations, International Telecommunication Union (ITU-T), Focus Group on Smart Sustainable Cities (FG-SSC).
- [5] Ishkineeva, G., Ishkineeva, F., & Akhmetova, S. (2015). Major approaches towards understanding smart cities concept. *Asian Social Science*, 11(5), 70.
- [6] Khan, Z., & Kiani, S. L. (2012). A cloud-based architecture for citizen services in smart cities. In 2012 IEEE Fifth International Conference on Utility and Cloud Computing (pp. 315-320). IEEE.
- [7] Albino, V., Berardi, U., & Dangelico, R. M. (2015). Smart cities: Definitions, dimensions, performance, and initiatives. *Journal of urban technology*, 22(1), 3-21.
- [8] Smart cities – Preliminary Report 2014. (2015). ISO/IEC JTC 1, Information technology, Switzerland.
- [9] Smart cities: Applications and requirements. (2011). NetWorks European Technology Platform, Rapport Technique.
- [10] Monzon, A. (2015). Smart cities concept and challenges: Bases for the assessment of smart city projects. In 2015 international conference on smart cities and green ICT systems (SMARTGREENS) (pp. 1-11). IEEE.

- [11] Bastien L., (2017) Cloud Computing – Définition, avantages et exemples d'utilisation, LE BIG DATA. 10 février 2017.
- [12] Girardot, J. J. (2000). Principes, méthodes et outils d'intelligence territoriale. Évaluation participative et observation coopérative (No. halshs-00539811).
- [13] Roscia, M., Longo, M., & Lazaroiu, G. C. (2013, Octobre). Smart City by multi-agent systems. In 2013 International Conference on Renewable Energy Research and Applications (ICRERA) (pp. 371-376). IEEE.
- [14] Su, K., Li, J., & Fu, H. (2011, Septembre). Smart city and the applications. In 2011 international conference on electronics, communications and control (ICECC) (pp. 1028-1031). IEEE.
- [15] Mizuki, F., Mikawa, K., & Kurisu, H. (2012). Intelligent water system for smart cities. *Hitachi Review*, 61(3), 147-151.
- [16] Smart water management in cities. (2014). Rapport Technique, ITU-T Focus Group on Smart Sustainable Cities.
- [17] Hoadjli, A., Kazar, O., & Rezeg, K. (2017, Mai). A layered design approach for mobile tourism. In 2017 8th International Conference on Information Technology (ICIT) (pp. 110-115). IEEE.
- [18] Su, K., Li, J., & Fu, H. (2011, September). Smart city and the applications. In 2011 international conference on electronics, communications and control (ICECC) (pp. 1028-1031). IEEE.
- [19] Kumar, T. V. (2015). E-governance for smart cities. In *E-governance for smart cities* (pp. 1-43). Springer, Singapore.
- [20] Keta, M. (2015). Smart city, smart administration and sustainable development. *Romanian Economic and Business Review*, 10(3), 43.
- [21] Rowley, J. (2011). e-Government stakeholders—Who are they and what do they want?. *International journal of Information management*, 31(1), 53-62.
- [22] Cortés-Cediel, M. E., Cantador, I., & Gil, O. (2017, Aout). Recommender systems for e-governance in smart cities: State of the art and research opportunities. In *Proceedings of the international workshop on recommender systems for citizens* (pp. 1-6).

- [23] Desprez F., (2010, mars). Cloud Computing. LIP ENS Lyon/INRIA Grenoble Rhône-Alpes.
- [24] Cloud computing nouveaux modèles!. (2012, mars). Syntec-numerique. Livre blanc.
- [25] Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L., & Leaf, D. (2011). NIST cloud computing reference architecture. NIST special publication, 500(2011), 292.
- [26] De Giusti, A. E., Erl, T., Zaigham, M., Puttini, R. (2013). Cloud computing. Concepts, technology & architecture. Journal of Computer Science & Technology, 13.
- [27] Hennion, N. (2011, Janvier). Introduction aux technologies cloud. Disponible sur : <http://docshare01.docshare.tips/files/5197/51978913.pdf>.
- [28] Kherbache, V., Moussalih, M., Kuhn, Y., Lefort, A. (2010). Cloud Computing, dissertation. IUT Nancy Charlemagne.
- [29] Laribi, I. (2014). Etude et mise en place d'une solution cloudcomputing privée pour une entreprise (Doctoral dissertation).
- [30] Grassa, N. (2014, Aout) cours virtualisation et cloud, ISET de kairouan.
- [31] Paterne, P. (2012). Conception d'une solution de Cloud Computing privé basée sur un algorithme de supervision distribué : Application aux services IaaS. Ecole Polytechnique d'ABOMEY-CALAVI (EPAC).
- [32] Adomavicius G., Tuzhilin A. (2011) Context-Aware Recommender Systems. In: Ricci F., Rokach L., Shapira B., Kantor P. (eds) Recommender Systems Handbook. Springer, Boston, MA
- [33] Dameri R. P. and C. Rosenthal-Sabroux (eds.). (2014). Smart City, Progress in IS, DOI: 10.1007/978-3-319-06160-3_5, c Springer International Publishing Switzerland.
- [34] Negre E., & Rosenthal-Sabroux C. (2014). Recommendations to Improve the Smartness of a City. In: Dameri R., Rosenthal-Sabroux C. (eds) Smart City. Progress in IS. Springer. Doi: 10.1007/978-3-319-06160-3_5.
- [35] Zemirli, W. N. (2008). Modèle d'accès personnalisé à l'information basé sur les Diagrammes d'Influence intégrant un profil utilisateur évolutif. Thèse de doctorat, Université Toulouse III-Paul Sabatier.

- [36] Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3), 261-273.
- [37] Verbert, K., Manouselis, N., Ochoa, X., Wolpers, M., Drachsler, H., Bosnic, I., & Duval, E. (2012). Context-aware recommender systems for learning: a survey and future challenges. *IEEE Transactions on Learning Technologies*, 5(4), 318-335.
- [38] Asanov, D. (2011). Algorithms and methods in recommender systems. Berlin Institute of Technology, Berlin, Germany.
- [39] Weinstock, C. B., & Goodenough, J. B. (2006). On system scalability (No. CMU/SEI-2006-TN-012). CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.
- [40] Bondi A. B., (2000) Characteristics of Scalability and Their Impact on Performance. Proceedings of the 2nd international workshop on Software and performance. DBLP. DOI: 10.1145/350391.350432.
- [41] Rosenthal, A., & Seligman, L. (2001). Scalability issues in data integration. In Proceedings of the AFCEA federal database conference.
- [42] Chen, G., & Kotz, D. (2000). A survey of context-aware mobile computing research. Dartmouth Computer Science Technical Report TR2000-381. ACM. 2000;1(2.1): 2-1.
- [43] Xin, Y. (2015). Challenges in recommender systems: scalability, privacy, and structured recommendations (Doctoral dissertation, Massachusetts Institute of Technology).
- [44] Khare, A., Huang, Y., Doan, H., & Kanwal, M. S. (2012). A Fresh Graduate's Guide to Software Development Tools and Technologies. Chapter 6 scalability.
- [45] Lane K. (2015) Overview Of The Backend as a Service (BaaS) Space. In: integrove. Disponible:<http://www.integrove.com/wpcontent/uploads/2014/11/apievangelistbaaswhitepaper.pdf>.
- [46] Araujo J., Dantas J., Silva F. A., Maciel P. R. M. (2015) Availability Evaluation and Sensitivity Analysis of a Mobile Backend-as-a-service Platform, in Quality and Reliability Engineering. DOI: 10.1002/qre.1927

- [47] Carter B. (2015). grow your own backend-as-a-service platform. In: GOCICT 2015 Conference College of Information & Computer Technology.
- [48] Feng, X., Shen, J., & Fan, Y. (2009, Octobre). REST: An alternative to RPC for Web services architecture. In Future Information Networks, 2009. ICFIN 2009. First International Conference on (pp. 7-10). IEEE.
- [49] Jansen, W. A., & Grance, T. (2011). Guidelines on security and privacy in public cloud computing. Information Technology Laboratory, National Institute of Standards and Technology; 2011; p. 52.
- [50] Kohl, W. (2013). Backend As A Service Using the Example of Enginio a Cloud Service for Qt.
- [51] Naccarati, F., & Hobson, S. (2011). IBM Smarter City Solutions on Cloud. IBM Global Services White Paper-Government Solutions; IBM: Somers, NY, USA.
- [52] Ruano, J., Chao, T., Hartswick, P., Havers, B., Meegan, J., Wasserkrug, S., & Williams, P. (2011). Smarter cities series: Understanding the IBM approach to water management. An IBM Redguide publication.
- [53] Schaefer, S., Harrison, C., Lamba, N., & Srikanth, V. (2012). Smarter Cities Series: Understanding the IBM Approach to Traffic Management. IBM Corporation.
- [54] Hwang, J. S., Young, O. G., & Choe, H. (2013). Smart Cities Seoul: a case study ITU-T technology watch report. Seoul, Feb. 1-20.
- [55] Ayachi, R., Boukhris, I., Mellouli, S., Amor, N. B., & Elouedi, Z. (2016). Proactive and reactive e-government services recommendation. Universal Access in the Information Society, 15(4), 681-697. Doi: 10.1007/s10209-015-0442-z.
- [56] Baldassarre, C., Cremaschi, M., & Palmonari, M. (2013, Aout). Bridging the gap between citizens and local administrations with knowledge-based service bundle recommendations. In 2013 24th International Workshop on Database and Expert Systems Applications (pp. 157-161). IEEE. Doi: 10.1109/DEXA.2013.25.
- [57] Tamine, L., & Bahsoun, W. (2006). Définition d'un profil multidimensionnel de l'utilisateur: vers une technique basée sur l'interaction entre dimensions. Actes de la

Conférence francophone en Recherche d'Information et Applications (CORIA 2006) (pp. 225-236).

[58] Kostadinov, D. (2007). Personnalisation de l'information: une approche de gestion de profils et de reformulation de requêtes. Thèse de doctorat, Université de Versailles-Saint Quentin en Yvelines.

[59] Carrillo-Ramos, A., Villanova-Oliver, M., Gensel, J., & Martin, H. (2006). Gestion des préférences utilisateurs pour les Systèmes d'Information ubiquitaires. Actes de la Conférence Inforsid.

[60] Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In Recommender systems handbook (pp. 217-253). Springer, Boston, MA. Doi: 10.1007/978-3-319-06160-3_5.

[61] Hasan, K. S., Rahman, M., Haque, A. L., Rahman, M. A., Rahman, T., & Rasheed, M. M. (2009, March). Cost effective GPS-GPRS based object tracking system. In Proceedings of the international multiconference of engineers and computer scientists (Vol. 1, pp. 18-20).

[62] The report Algeria 2017. Oxford business Group.

[63] Mes démarches administrative. Disponible sur :
<http://www.interieur.gov.dz/index.php/fr/mes-démarches-administratives.html>

[64] Hammi Abderahmene (2003). Mode de production d'un outil d'aide à la décision et à la conception destiné aux autoconstructeurs. Masters thesis, Université Mohamed Khider Biskra.

[65] Android studio user guide, disponible sur: <https://developer.android.com/studio/intro>

[66] Picard G., Vercouter L. (2013). Initiation à la programmation orientée-objet avec le langage Java. Ecole nationale supérieure des mines. Pole informatique.

[67] Documentation Cloud Endpoints. Disponible : <https://cloud.google.com/endpoints/docs/>

Liste des publications

Revue internationale

HOADJLI Abir, REZEG Khaled. 'A scalable mobile context-aware recommender system for a smart city administration'. International Journal of Parallel, Emergent and Distributed Systems, 2019. DOI: 10.1080/17445760.2019.1626855

Conférences internationales

HOADJLI Abir, REZEG Khaled. 'A Smart City System using Backend as a Service Approach: Biskra City Case Study'. 2018 Fifth International Symposium on Innovation in Information and Communication Technology (ISIICT) (pp. 1-7). IEEE. 2018.

DOI: 10.1109/ISIICT.2018.8613725

HOADJLI Abir, KAZAR Okba, REZEG Khaled. 'A layered design approach for mobile tourism'. 2017 8th International Conference on Information Technology (ICIT) (pp. 110-115). IEEE. 2017. DOI : 10.1109/ICITECH.2017.8079986