

An Integrated Taxonomic Model of Gastronomic Offers Using Web Services and Databases

Paul Dayang*, Donald Didier Sepele Hlamna

Department of Mathematics and Computer Science, Faculty of Science, University of Ngaoundere, Ngaoundere, Cameroon

Email address:

piusday@gmail.com (P. Dayang), dsepele@yahoo.com (D. D. S. Hlamna)

*Corresponding author

To cite this article:

Paul Dayang, Donald Didier Sepele Hlamna. An Integrated Taxonomic Model of Gastronomic Offers Using Web Services and Databases. *International Journal of Information and Communication Sciences*. Vol. 6, No. 3, 2021, pp. 55-61. doi: 10.11648/j.ijics.20210603.11

Received: July 7, 2021; **Accepted:** July 19, 2021; **Published:** July 27, 2021

Abstract: Data integration is the process of combining data from several different sources into a unified view, making it more actionable and valuable to access by different users with diverse interests. Among data integration approaches, figure prominently databases, cloud computing and web services. With the progression of the technology, mostly in the internet domain, we are facing a proliferation of websites with a high level of data redundancy. Several of those websites evolve in the same domain of activity. To limit the proliferation of websites that content mostly the same information, we propose in this paper, a data integration model based on web services and databases. This double data integration based on a services registry that content main web services in relationship with information contained in a database, respect a specific taxonomy for a specific domain. In this paper, we focus our research on the domain of gastronomy. In order to construct our model, we revisited some main architectures and models that exist so that we can use them as building blocks in the new approach. As building blocks, we can name service-oriented architecture, Universal Description, Discovery and Integration and taxonomy. Revisiting those established models have help us to build the taxonomic data integration model.

Keywords: Web Service, Data Integration, Taxonomy, Gastronomy

1. Introduction

With the last year's technological mutation, sharing information become crucial. Sharing information is essential to improve data access time. It allows also the management of complex applications requiring access to large volumes of data often stored on remote servers. Development of internet induced the proliferation of websites. Several of these websites evolve in the same sector of activity. Most often, these websites contain the same information with some minor differences. Furthermore, this information must be deployed on different platforms and managed by different programming languages. To use effectively the information disseminated through a network environment managed by heterogeneous and autonomous systems, the approach of data integration becomes indispensable [1, 2].

Data integration is the process of combining data from several different sources into a unified view, making it more actionable and valuable to those accessing it. Data integration allows the data to be used as if they were stored locally or on

a unique disk. It allows also to broadcast data rapidly, simply, efficiently and economically. Lined to data integration models, we have: databases federation [3], cloud computing and web services [4].

Recent years have witnessed a software development paradigm transformation from component-based to service-based due to the broad adoption of service-oriented architecture (SOA) and its related technologies [5]. Among the SOA technologies, we have web services. A web service is a software application identified by a URI, whose interfaces and binding are capable of being defined, described and discovered by XML artefacts. Moreover, such an approach supports direct interactions with other software applications using XML based messages via Internet-based protocols [6].

To limit the proliferation of websites that content hardly the same information, it's advisable to create a central platform that content all information concerning a specific domain. All the other websites that operate in the field can simply integrate these data in their platforms and complete

them with their specificities. This approach has the advantage of not requiring users to provide the same information several times on different platforms.

How can we combine web services and databases to define a data integration model that respect a given taxonomy? To illustrate this model, we will focus on the field of gastronomy.

The rest of the article is structured as follows. In the first section, we revisit some publications and basic elements on the functioning of web services. Afterwards, we study some concepts of gastronomy which constitute the first step of your contribution. Subsequently, the main contribution will be the taxonomic data integration model of gastronomic offers. Finally, we draw some conclusions and perspectives for future research.

2. Prerequisites on Web Services Required of Our Model

Web services are an implementation of the Service Oriented Architecture (SOA). To understand the functioning of web services, firstly we must understand the logic underlying the approach of SOA.

2.1. The Service Oriented Architecture

The SOA is a model use preferably in distributed applications. The SOA is not a technology but it represents a way of designing and deploying applications as described in Figure 1. Precisely, the SOA allows to structure our projects based on services approach [7, 8]. A service is a mechanism enabling and supporting a machine-to-machine communication through a prescribed interface under some

constraints and policies as specified by the service description [9]. This architecture allows to link up many applications working together to carry out a global functionality.

All applications link up in SOA, play different roles. These applications act as actors. There are two main actors:

- 1) The service provider is an application that performs an activity whose results are directly or indirectly exploitable by other applications, possibly distributed over a network. This application runs on a server and includes modules (methods or functions) that can be invoked from the outside, thanks to an interface described respecting a specification or a precise format.
- 2) The service requester, also known as service consumer, is an application that benefits from the results of a service provider's activity. To benefit from the services of a service provider, the service requester must invoke the methods of the service provider as there are described at the level of the interface and if possibly by providing parameters.

Results from the activity of a service provider are called service delivery which can be grouped into three categories:

- a) Information: The application provider performs on behalf of the client, treatments whose results are communicated to the service requester.
- b) States: The application provider manages states and state changes, represented by datasets.
- c) Side effects: The application provider performs interactions with the environment that is to say with a set of devices that allows the input and output of data from the system.

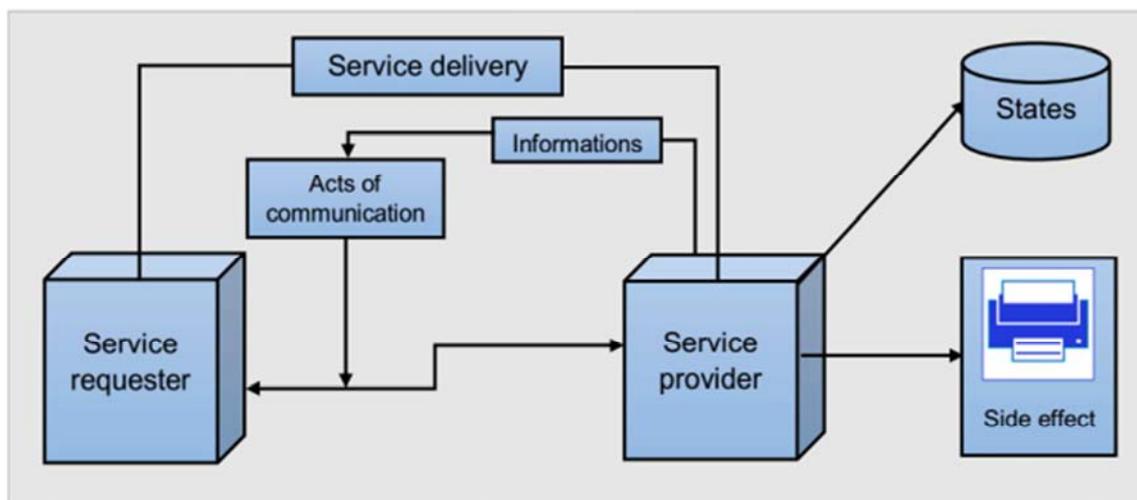


Figure 1. The basic elements of SOA.

A service relationship is a relationship between a service provider and a service requester. The service relationships are created and consumed during message exchanges. A message is constituted of one or more acts of communication. The relationship between two applications is therefore based on an exchange of communication acts. Communication acts are actions performed by each application intervening in a

service relationship. These acts of communication are used to implement the service or are directly part of the service delivery.

In order to enable, the service provider and the service customer to exchange communication acts, they must first comply with a service contract. The service contract describes the reciprocal commitments of the service provider

and the service requester [9].

2.2. The Service Oriented Architecture

We remind the three main actors: the service provider, the service registry and the service requester. The description of this web service is made in Web Service Description Language (WSDL). The WSDL is based on XML. After this description the web service is published on the Universal

Description, Discovery and Integration (UDDI). When the web service is published in the UDDI, the service requester now, can have access to the URL of this web service. To find the service, the service requester can use the quality of services (QoS) property [10, 11] or the Web service discovery based on goal-oriented query expansion [12, 13] event if it presents some limits. Figure 2 shows the overall functioning of the web service.

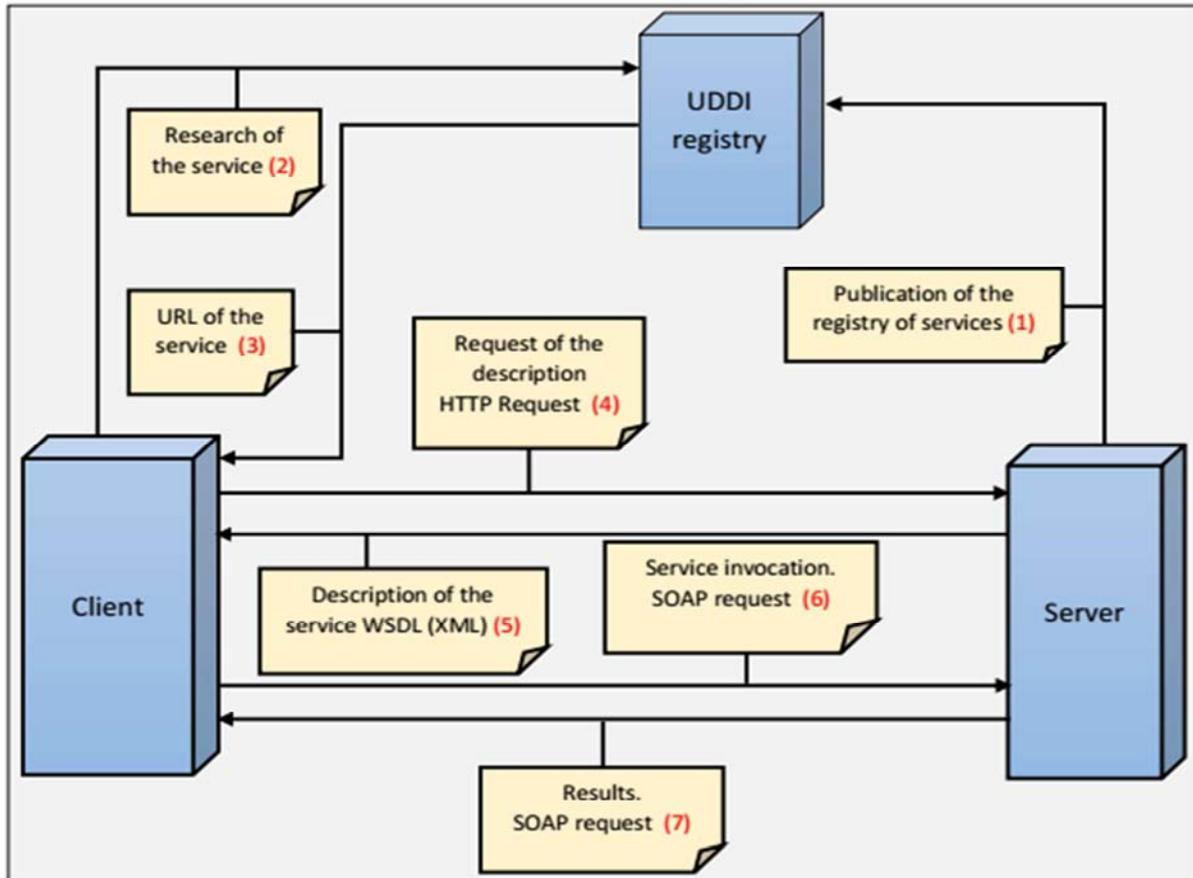


Figure 2. Overall functioning of web services.

The exchanges between these three actors are therefore governed by the following technologies.

The HTTP protocol is a client-server communication protocol developed for the web.

The XML is a language for formalizing documents using tags and rules.

The SOAP communication protocol described in XML aims to exchange structured information in a decentralized and distributed environment.

The WSDL is a language, also based on the XML for describing web services. The web service description in WSDL represents the service contract [14-17].

The UDDI is a web service that contains technical and administrative information about the companies and services they publish. This registry enables the publication and discovery of Web services by requester application [18-20].

3. Taxonomy of Gastronomic Offers

A restaurant is a structure that prepares and serves food and beverages in exchange of a payment. There are several types of restaurant. To classify these gastronomic structures, several criterion and methods are used. Moreover, these criterion and methods differ from one country to another or from one region to another. These classifications are generally made according to the quality of services and facilities offered by the restaurants.

Several other factors will guide us towards a gastronomic taxonomy. Among these factors, we can mention the development of cities, the search for a better quality of life, the employment and working time of women or the globalization of gastronomy. Each gastronomic structure is generally governed by a gastronomic formula or a combination of gastronomic formula, as shown in Figure 3.

The diet behaviour of consumers belongs of the package of the classification criterion.

According to the eating habits, we may identify three major types of meals:

- 1) The nutrition meal: The consumer seeks above all to cover his physiological needs. It is the meal taken quickly between noon and one o'clock.
- 2) The business meal: It's happening mostly in diverse prestigious institutions.

- 3) The leisure meal: It is taken during private outings, with friends or with family.

These consumption habits have led to the emergence of new formulas and catering offers. The following diagram (Figure 3) gives an overall view on the gastronomic classification. This classification is the basis structured data that will be used for the demonstration of the taxonomic integration in the next sections.

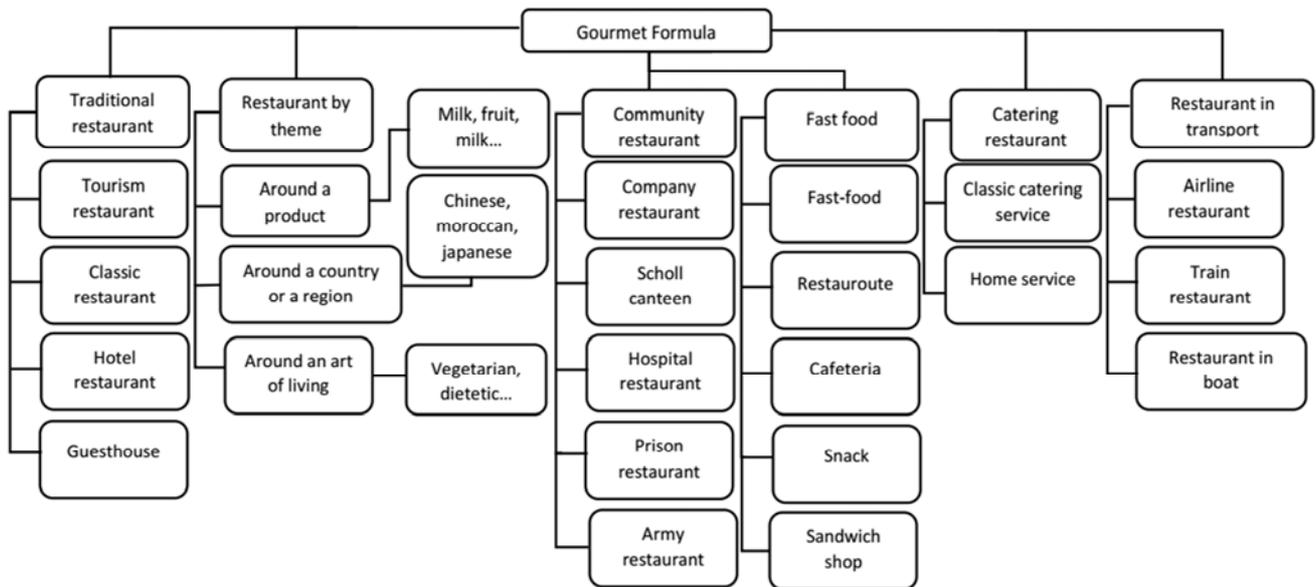


Figure 3. Taxonomy of gastronomic Offers.

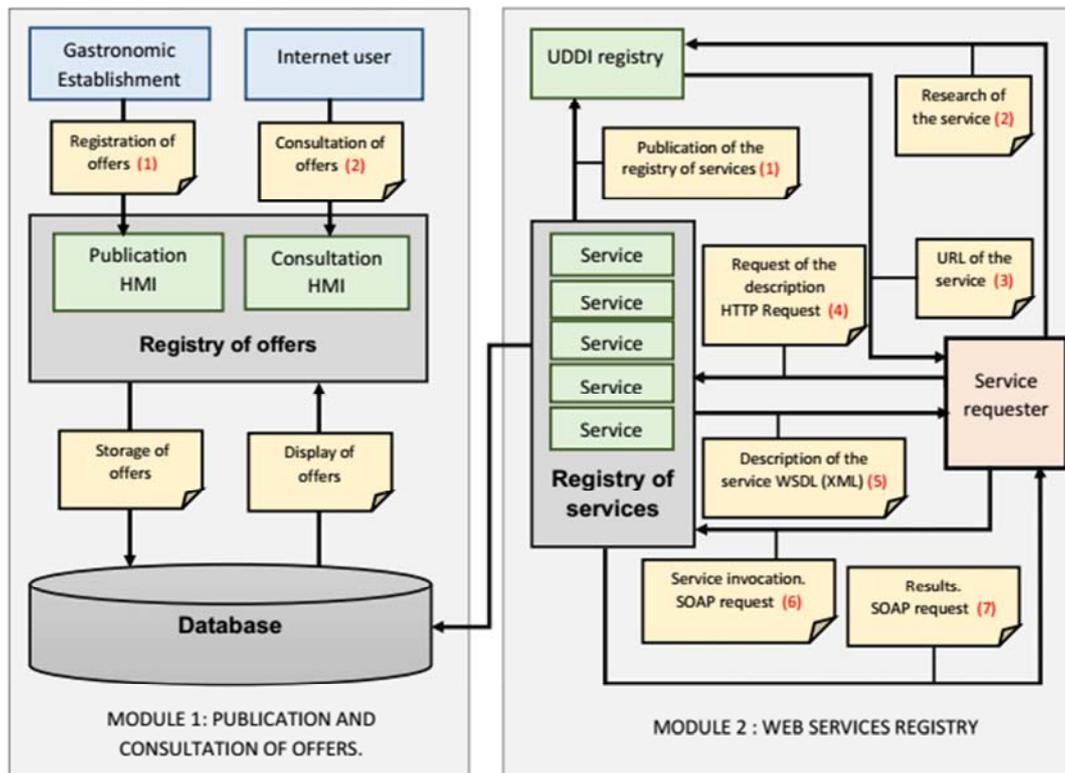


Figure 4. Taxonomic data integration model.

4. Presentation of the Model

The taxonomic integration model of gastronomic offers is divided into two modules (Figure 4): the module for publishing and discovering offers from the restaurants and the module for publishing and discovering web services. The second module provides access to the information stored in a database. The basic data stored in the database are previously provided by owners of gastronomic establishments from the first module.

4.1. Module for Publishing and Discovering of Gastronomic Offers

Each gastronomic establishment has an account. The account is used to publish or delete offers and to update its account through Human Machine Interfaces (HMI) provided for this purpose. All data provided by a gastronomic establishment are stored in a database. For each gastronomic establishment the main information that will be stored in the database are: general information about the restaurant (name of the restaurant, name of the director etc...), address of the restaurant (country, city, district, phone number, email address, website etc...), opening days, opening time, closing time, capacity, description, average budget, type of restaurant, contact details geolocation, culinary specialties, suggestions, services offered, the peculiarities of the restaurant, the menu, some pictures etc...

In addition to the publishing interfaces, another HMI allows Internet users to view the offers presented by catering establishments. For that purpose, the user has a front-end that contains information on all the gastronomic establishments stored in the database. The users may the list of restaurants, by searching according to a criterion or a combination of criteria (neighborhood, type of restaurant, menu etc...).

4.2. Module for Publishing and Discovering of Web Services

The central element of the second module is a directory containing a set of web services related to the gastronomic offers. These web services can be written in diverse object-oriented programming languages such as Java, Net or PHP. This registry is also published in the UDDI in order to be discovered by other applications. By publishing this directory in the UDDI, a dual possibility is offered to the various service requesters. Indeed, when the URL of the model's registry is not known by the service requesters, they can issue an HTTP request to acquire this information. When client applications know the service directory URL in relation to the gastronomic offers, they do not need to connect to UDDI before invoking one of the different services contained in the directory.

The advantage of that model is that, the set of web services contained in the directory respect a precise taxonomy. All applications in the same field no longer need to create or constructed their own databases. In addition, client applications in the domain will not have to rewrite a set of functions or methods to access information that already exist,

which will reduce a lot of work. It also has the advantage of not requiring users to provide the same information several times on different platforms.

5. Modelling Approach for a Mobile Application

With the recent rapid development of mobile devices in terms of processing power, memory and storage capabilities coupled with the advancements of wireless technology in terms of high data transmission rates such as 3G and 4G, it has now become feasible to host Web services on mobile devices [21]. To use web services on a mobile device, many frameworks have been developed [22, 23].

The mobile app alerting the proximity of a restaurant allows you to locate a restaurant by moving. When a restaurant establishment is registered in the directory, this application can locate it even if it is in Indeed, all the technologies used in this model exist and are functional. The contribution of this model is to gather within a single entity a set of functions or methods relating to a set of information previously entered by users. The data and the services that use them respect a certain taxonomy for a specific area.

This model (Figure 5) actually allows for a double data integration. When the offers providers store the information in the database, the users can then consult them using a set of function to query linked to this database. This module is therefore reserved purely for human actors. The second module is reserved for software agents because it is a box containing a set of Web services that can be evoked by various client applications.

Indeed, there are catering establishments that we do not suspect exist because they operate in the informal sector. Generally, such gastronomic structures are not published in the database.

In order to get the mobile working, the activation of the geolocation system of the telephone system is a prerequisite. After this activation, the mobile application performs an invocation of the access service to geolocation coordinates to perform a correspondence test between coordinates of the phone and the coordinates contained in the database of the publication module and the discovery offers.

6. Summarizing the Contribution of our Model

Indeed, all the technologies used in this model exist and are functional. The contribution of this model is to gather within a single entity a set of functions or methods relating to a set of information previously entered by users. The data and the services that use them respect a certain taxonomy for a specific area. In our case study, the illustration was done for the gastronomic area.

This model actually allows for a double data integration. When the offers providers store the information in the

database, the users can then consult them using a set of function to query linked to this database. This module is therefore reserved purely for human actors. The second

module is reserved for software agents because it is a box containing a set of Web services that can be evoked by various client applications.

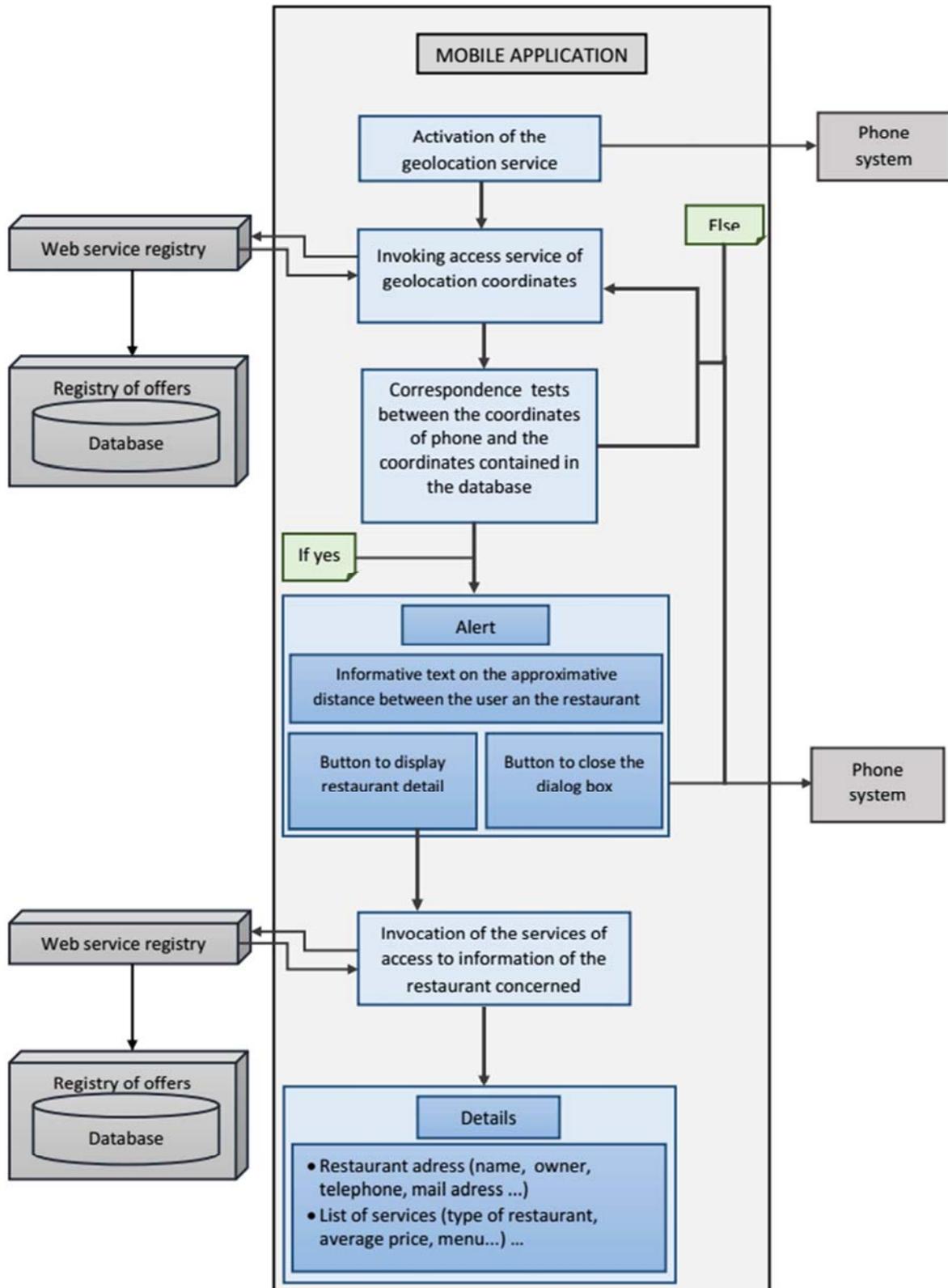


Figure 5. Mobile application using the model suggest in this paper.

7. Conclusion

In this paper, we suggested a taxonomic data integration model based on web services and databases. This integration is twofold. On the one hand, we have an integration for human actors and on the other hand for software agents. As an illustration, the field of gastronomy has been chosen. This model has been proposed to bring together on a single platform a set of information on a specific field. The part played by web services in this model is to allow to all web sites evolving in the same domain to integrate data coming from diverse platforms. The taxonomic data integration model, that allow a double data integration, is based on standards like HTTP, XML, SOAP and WSDL. All these technologies or building blocks exist and are functional, which show and ensure that the approach proposed in this paper is reliable. To avoid, the use of UDDI registry always, the URL of this service registry must be known by all applications that invoke it. Future work will consist in migrating the model to other domains in order to validate it and also prove its portability which is an important feature of the nowadays software solutions.

References

- [1] X. Wei, "Heterogeneous database integration middleware based on Web services", *Physics Procedia*, vol. 24, pp. 877-882, 2012.
- [2] W. Shao, H. Zhang, J. Liu, G. Yang, X. Chen, Z. Yang and H. Huang, "Data integration and its application in the sponge city construction of CHINA", *Procedia Engineering*, vol. 154, pp. 779-786, 2016.
- [3] L. M. Haas, E. T. Lin, and M. A. Roth, Data integration through database federation in *IBM Systems Journal*, vol. 41, no. 4, pp. 578-596, 2002.
- [4] W. Kima, J. Hwan Lee, C. Hong, C. Han, H. Lee, and B. Jang, An innovative method for data and software integration in SaaS, *Computers and Mathematics with Applications*, vol. 64, pp. 1252-1258, 2012.
- [5] Q. He, J. Yan, H. Jin, and Y. Yan, Quality-aware service selection for service-based systems based on iterative multi-attribute combinatorial auction. *IEEE Transactions Software Engineering*, vol. 40, no. 2, pp. 192-215, 2014.
- [6] W3C Working Group Note, Web Services Architecture, February 2004. <https://www.w3.org/TR/ws-arch/> [Online: 13 November 2018].
- [7] Z. M. Aljazzaf, M. Perry, and M. A. M. Capretz, Trust-based Service-Oriented Architecture, *Journal of King Saud University – Computer and Information Sciences* vol. 28, pp. 470-480, 2016.
- [8] H. Yue, and X. Tao, Web Services Security Problem in Service-oriented Architecture, *Physics Procedia*, vol. 24, pp. 1635-1641, 2012.
- [9] M. Lopez-Sanza, C. J. Acuna, C. E. Cuesta, and E. Marcosa, Modelling of Service-Oriented Architectures with UML, *Electronic Notes in Theoretical Computer Science*, vol. 194, pp. 23-37, 2008.
- [10] S. Shrivastavaa, A. Sharmab, and D. Shrivastavac, An Approach for QoS Based Fault Reconfiguration in Service Oriented Architecture, *Procedia Computer Science*, vol. 46, pp. 766-773, 2015.
- [11] Z. Aljazzaf, Bootstrapping quality of Web Services, *Journal of King Saud University – Computer and Information Sciences*, vol. 27, pp. 323-333, 2015.
- [12] N. Zhang, J. Wang, M. Yutao, H. Keqing, L. Zheng and L. Xiaoqing, Web service discovery based on goal-oriented query expansion, *The Journal of Systems and Software*, vol. 142, pp. 73-91, 2018.
- [13] E. Hustada, and C. de Lange, Service-oriented architecture projects in practice: A study of a shared document service implementation, *Procedia Technology*, vol. 16, pp. 684-693, 2014.
- [14] Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language <https://www.w3.org/TR/2007/REC-wsdl20-20070626/> [Last consultation: 06 September 2020]
- [15] Web Services Description Language (WSDL) 1.1 <https://www.w3.org/TR/2001/NOTE-wsdl-20010315> [Last consultation: 06 September 2020]
- [16] Understanding WSDL <https://msdn.microsoft.com/en-us/library/ms996486.aspx> [Online: 06 September 2019]
- [17] WSDL and Service Contracts <https://docs.microsoft.com/en-us/windows/desktop/wsw/wsdl-support> [Online: 06 September 2020].
- [18] UDDI <https://msdn.microsoft.com/en-us/library/aa286530.aspx> [Last consultation: 13 November 2019].
- [19] UDDI (Universal Description, Discovery, and Integration) <https://searchmicroservices.techtarget.com/definition/UDDI-Universal-Description-Discovery-and-Integration> [Online: 22 December 2020].
- [20] Relationship between UDDI and WSDL <http://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.jst.ws.consumption.ui.doc.user%2Fconcepts%2Fwscwslud.html> [Online: 22 December 2020].
- [21] K. E. Mohameda, and D. Wijesekera, Performance Analysis of Web Services on Mobile Devices, *Procedia Computer Science*, vol. 10, 744-751, 2012.
- [22] Kim, Y. and Lee, K., A Light-weight Framework for Hosting Web Services on Mobile Devices, *Web Services, 2007. ECOWS '07. Fifth European Conference*, pp. 255-263, 26-28 Nov. 2007.
- [23] F. AlShahwan and K. Moessner, Providing SOAP Web Services and RESTful Web Services from Mobile Hosts, *Internet and Web Applications and Services (ICIW)*, pp. 174-179, 2010.