Implementation of Automatic Hotel Reservation System (AHRS) on J2EE Framework

Moh Moh Khaing, Khin Mar Soe
University of Computer Studies, Yangon
moh2khaing@gmail.com

Abstract

Nowadays, hotels are in the large-scale distributed environments where each site is completely autonomous and offers services to its customer through transparent external communication. In the development of the World Wide Web, web searching strategies are evolved in many application areas. A web application is designed to provide large travel agencies with all the functions required to manage their activities. This system implements automatic hotel reservation system (AHRS) in a transparent manner on J2EE Framework. J2EE Framework is supported by Java which makes it easy to build locations and access in a distributed environment. In this paper, we present possible plans and make reservation by negotiation model according to user requirements. Faster change information will help the hotel's activities integrate their processes over the internet so that they can achieve greater than efficiency.

Keywords- *J2EE Framework, automatic hotel reservation system, negotiation model, travel agencies, user requirements*

1. Introduction

Automatic hotel reservation system (AHRS) is rapidly emerging as a new paradigm to develop complex, intelligent and adaptable negotiation systems over the Internet This system is built upon the foundations of agent research with a strong emphasis on the automatic negotiation, security issues of agent systems and travel models of reservation systems [5]. The network of computers become essential to manage the hotel information and the distributed computing is also necessary to exploit the hotel's organization. Distributed system supports computation that is running in different address spaces, practically on different machines, must be able to communicate between one machine to another [7].

With growth in distributed computing, internet, www, e-commerce, more and more applications must operate in open systems, where the components change over time, and there is a lot of information available from multiple sources, much of it unstructured [6]. The proposed hotel reservation system is a cross-platformed, dynamically distributed system that is modular, self-healing, plug-and-play enabled, and that allows for adaptive parallelism. Therefore, it is only appropriate where the marginal cost of providing a service via technology is less than the marginal cost of providing an equivalent service via non-technological means [4].

There are many ways to increase revenue or decrease costs in existing hotels. International five-start hotels chain with their own hotel reservation systems with features such as centralized reservations will probably show little interest in this system. This system neither details a cost-effective way to integrate with their legacy systems, nor does it purpose a solution that takes advantage of the same economies of scale as they do. Thus, the focus is to provide a solution for the middle range of hotels, such as four-star hotel chain, for whom currently available technology is affordable and can also provide valuable services to bring them on par with the five-star hotels. The solution is web-enabled and scalable to allow for a hotel to grow into a hotel chain.

In this system, both individual customers and travel agencies will be able to benefit from the usage of the system, since its negotiation strategies will not depend on price only, but several attributes, like the number of the rooms, the required facilities. The proposed system will use J2EE compatible agent development framework, developed during the implementation with negotiation and secure transaction capabilities, which generates domain dependent framework from a generic agent framework. Our architecture is targeted for individual customers and travel agencies, and it is based on three main abstractions: (i) hotel information enquiry like room rates, room types, facilities available etc, (ii) negotiated between retrieved information and calculating the total cost, (iii) reservation and cancellation of the plan for hotels.

In this paper, this section presents the introduction of automated hotel reservation system. The next section will discuss development methodology and Section 3 mentions the architecture of proposed system. In Section 4, we

describe the negotiation model for the proposed system. The next Section 5 presents the database design for hotel reservation system and we mention the implementation of automatic hotel reservation system in Section 6 and the last section describes conclusion of proposed system.

2. Development Methodology

The proposed multi_agent system will be based on AHRS based agent framework, which is being developed in three town Yangon, Mandalay and Taunggyi. The framework has a layered architecture, which consists of three layers. The principle purpose of this layered architecture is to separate the domain related behaviors from agent specific behaviors. This feature emphasizes the system objectives from the point of providing seamless integration of domain behaviors into the generic agent architecture. The layers of the generic architecture are shown in Figure 1.

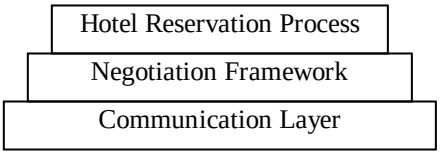

Figure 1: Layers of the Framework

The communication layer provides the classes and the methods, which will enable the agent to communicate with AHRS. It also communicates between the hotels for each town to get guest's requirements. The second layer, which is called the "Negotiation Framework", contains actions, which are common in all agents. For example, registering or deregistering itself to the agent platform is a standard action common in all AHRS complaint agents. However, this layer can be enriched with extra common actions, such as negotiation, security and ontology related behaviors. It negotiates hotel type, room type, number of rooms, and number of days which entered user's inputs with information from the data stored in database for each town and shows possible plans to guest. Hence, this layer enables agents to collaborate in the open environments such as web.

The upmost layer, which is named as "Hotel Reservation Process" contains the actions which are specific to a certain application domain or to a specific agent type. For example, the actions that are related with providing directory service to other agents on the platform are specific to directory facilitators. Thus such actions are used to create specific types of service agents. As another example, the action related with finding a hotel room as requested is specific to the customer domain. These kinds of actions create domain specific agents from a generic agent framework using the agents from layers. Hotel reservation system is implemented on negotiation framework. The system accepts user's enquiry and register user information and stores the reservation information in each related hotel containing in plan that is users' choice.

The generic framework will be used to instantiate the required agents and the collaborative between agents which will be identified during the realization of the methodology to create generic travel agent framework [3]. For example, an agent that is responsible of integration of legacy hotel reservation system to the agent system will be built upon the generic framework by specification of the domain specific layer. Since generic framework supports web standards like XML, SOAP etc., domain specific agents, AHR service agents and legacy systems will be able to collaborate easily in web environment.

3. Architecture of Proposed System

With the evolution of the Internet, hotel customers can gain in-depth information about hotel prices and quality before they step into a hotel. Hotel customers, now well informed about price and quality through information sharing on the internet, have more choices of lodging. Therefore, the current trend in the hotel trend is to gain market share not by competing on costs, but by services. Figure 2 shows the architecture of the proposed system. All users or customers search the plans for hotel reservation at the client side. In server site, AHR system implements and puts together three databases of towns in Mandalay, Taunggyi and Yangon. In these databases, all hotel information includes within each town.

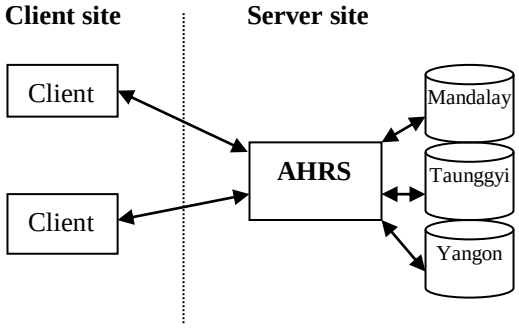

Figure 2: Architecture of Proposed System

The system is automatic hotel reservation system for helping guests to get required hotel type, room category and type, hotel location and other requirements such as internet access or restaurant on desire time period. It will be compared user requirement with the fact in database and retrieved

information that user wants. In this system, the customers will be negotiated between retrieved information and calculating the total cost. The system will make possible plans and to show the guest. The user can choose a plan and the system automatically makes booking to all hotels of three towns in that plan. Figure 3 shows the process flow of automatic hotel reservation system on J2EE framework.

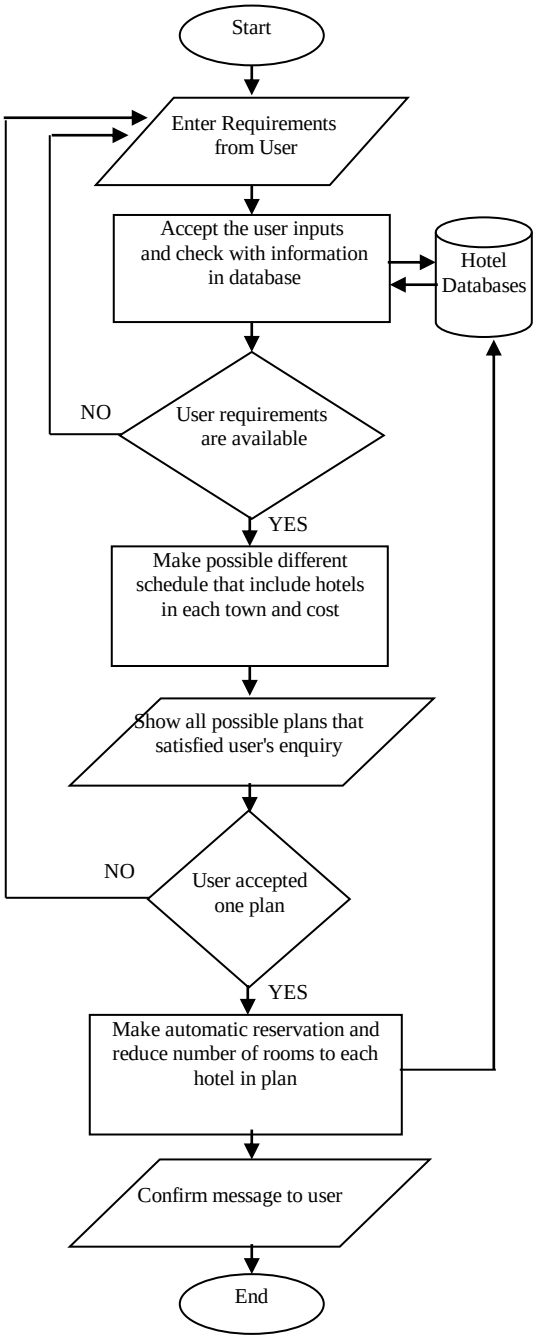

Figure 3: Process Flow of Automatic Hotel Reservation System

System will find and match the data in database according to user requirements such as number of days, number of rooms, hotel type, room category and type, hotel location and other facilities (including restaurant gym and internet access) in three town Yangon, Mandalay, and Taunggyi. System shows possible plans and one that is including number of days to stay hotel type, room category and type, cost for each town and total cost for whole plan. This system also registers the guest information in guest list when user is not been a member. But user is already member, user enter in Login. The system reserves the information in each related hotels in plan which is accepted by the user.

4. Negotiation Model for the Proposed System

Agent-based technology appears to offer solutions to many new challenges faced by information infrastructure. Agent communication languages and protocols, matchmaker and facilitator agents, could be the glue that allows these applications configure themselves to work together dynamically. A negotiation protocol represents the exchange of messages during the negotiation process. The term negotiation strategy represents the logic used by a partner to decide which provider or consumer satisfies his needs best [1].

Any negotiation setting will have four different components: a negotiation set, a protocol, a collection of strategies and a rule. There are three obvious possibilities: one-to-one negotiation, many-to-one negotiation and many-to-many negotiation. In this system, we use many-to many negotiations as defined as meta-negotiation.

Meta-negotiation is defined by means of a meta-negotiation document where participating may express: the pre-requisites to be for a negotiation. Meta-negotiation infrastructure can be employed in the following manner [2]. A service provider publishes descriptions and conditions of supported negotiation protocols into the registry. Service consumer performs a lookup on the registry database by submitting their own documents. The registry discovers service providers who support the negotiation process that a consumer is interested in and returns the documents published by the service providers. Finally, after an appropriate service provider and a negotiation protocol is selected by a customer using his/ her private selection strategy, negotiations between them may start according to the conditions specified in the provider's documents. We present negotiation model for the proposed system in Figure 4.

In negotiation model, the system accepts all user inputs such as hotel type, room type, no of rooms and no. of days. These input data are retrieved from each hotel in each town of three databases. It will be compared user requirements with the fact in database and retrieved information that user wants.

This system will also be possible negotiated between retrieved information and calculating the total cost. It will be made possible plans and to show the guest. Each plan includes possible one hotel satisfied with their needs in each town and total charges. The user must be chosen one plan from possible plan and after accepting by the user, the system automatically saved to all hotels in that plan.

Begin

- Accept all user inputs *(Hotel Type, Room Type, No. of Rooms, No. of Days)*
- Retrieve available data from each hotel in each town
- Compare user data and make reasonable schedules
- Show possible plans to user

End

Figure 4: Negotiation Model

5. Database Design for Automatic Hotel Reservation System

Every hotel operates in a different environment with different wants and needs. This section serves as a database system design with software modularity; hotels can customize their own hotel reservation system by picking and choosing the features that best meet their requirements. Following recent trends, business travelers now demand larger hotel suites than smaller hotel rooms. With the faster-paced working environment and the demand for connectivity, business travelers are spending more time in their room, and using the hotel room as an office. Therefore, demand is increasing for amenities such as room service, secretarial service, laundry service, transportation service, Internet connectivity, video conferencing and activities reservation. However, computers can help keep track of the guest's needs and complaints, provide the guest with administrative services, and help to plan the hotel employee workflow. While there are the other methods to obtain service personalization, computers can potentially allow hotels with a smaller budget to complete with hotels that spend considerably more on service personalization. There are two database designs in automatic hotel reservation system. In first database design, we construct a database of each site (Yangon,Mandalay, Taunggyi) for this system. These three sites or cities have the same feature of databases. The availlist class, the room class, hotellist class, reservation class, and guest class are set in database design of a site for AHRS as shown in Figure 5. The availlist class contains avail_id, hotel_id, room_id ,available and price. The room_id of the availlist class links to room class. The room class includes room_id, room_cat and roomtype. The hotel_id in availlist class links to hotellist class. Hotellist class consists of hotel_id, hotelname. The hotel_id in the hotellist class links to the reservation class. The reservation class has res_id, guest_id, room_id, no_of_room, no_of_days, hotelid, hotel_id and totalcost. The guest_id in the reservation class links to the guest class. The guest class takes in guest_id, name, gender, address, email, fax, telephone no, password. It stores guest information of each site.

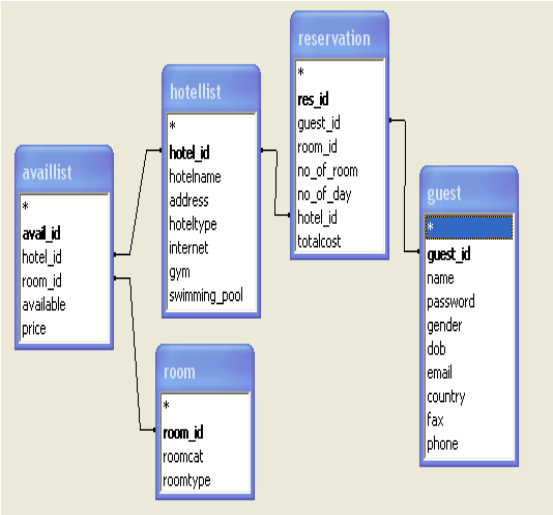

Figure 5: Database Design of a site for Automatic Hotel Reservation System

In second database design, we build for storing of all guests and hotel reservations of three sites. The guest class and reservation class are set as shown in Figure 6. The guest class contains guest_id, name, password, gender, dob, email, country, fax, and telephone no. The reservation class has res_id, guest_id, room_id, no_of_room, no_of_days, hotel_id, travelplan and totalcost. The guest_id in guest class relates to the reservation class. Guest database includes all guests and hotel reservations for three sites in automatic hotel reservation system. These data are used to allow registration to enter the agent platform. This system makes use of traveling and reaching on these data. Customer(guest) chooses the desire hotel and its information is shown to the viewing on web page. After that, this system evaluates the available price and traveling plan through reservation process.

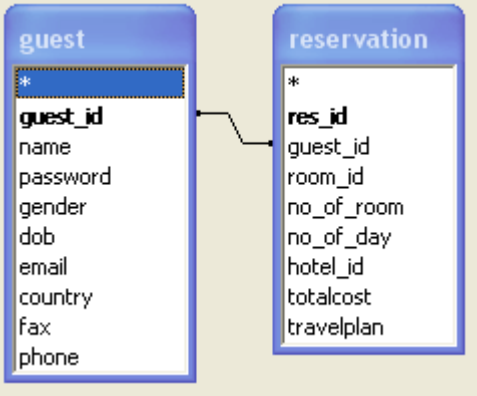

Figure 6: Database Design of Guest for Automatic Hotel Reservation System

6. Implementation of Automatic Hotel Reservation System

The implementation of AHRS uses inexpensive off-the-shelf components, such as inexpensive desktops and servers, to keep hardware costs low. The system is a plug-and-play solution with a modular design to keep installation, maintenance, and upgrading costs low. The solution is also designed for distributed computing with each computer working in parallel so that an isolated failure will not bring the demise of the entire system, thus providing the robustness demanded by hotels. .

In this system, hotels inform on-line user about their hotel's services on their own site. Each hotel has many branch hotels in same country or different country. The key to success in the networked economy is the ability to create and modify process to automate value chains in concert with changing requirements. The key success of automatic hotel reservation on distributed architecture is the ability to reduce the use of processing time and resources, produce a better performance than traditional reservation.

Every guest must have user name and password to enter into the AHR System. After that, he/ she can sight the hotel information including hotel type, hotel facilities, and region's view. When every guest enters this system, he/she fills up concern with his/her information and selects the desire hotels within three towns correctly for reservation. If a guest is wrong for writing and selecting, the system displays the error message for guest. In Figure 7, we illustrate the filling guest's data on reservation.

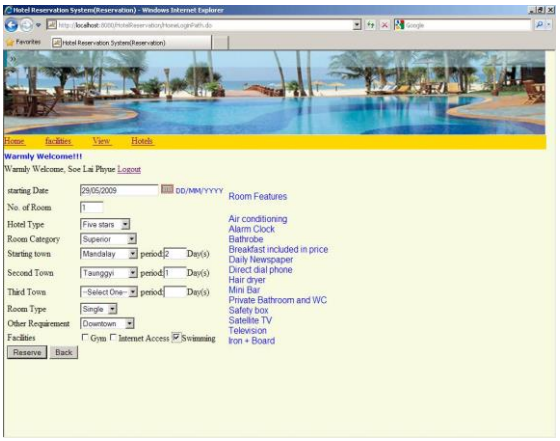

Figure 7: Filling Guest's Data on Reservation

In negotiation process, the system evaluates the appropriate plan on the choice of hotel information. In Figure 8, the guest chooses the desire and suitable plan in this system. Figure 9 is shown as successful of the confirmation plan for guest in AHR system.

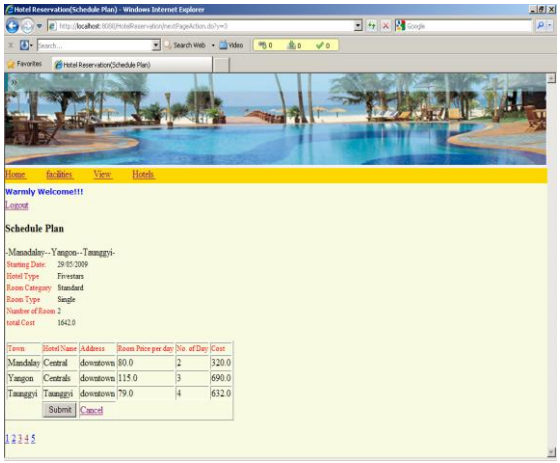

Figure 8: Choosing Appropriate Plan

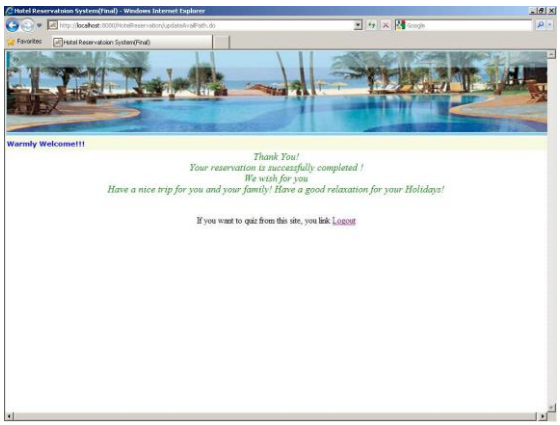

Figure 9: Confirmation Plan for Guest in AHR System

7. Conclusion

The proposed system can provide the data consistency and access and location transparency in hotel reservation system. According to the system, users can view the hotel information from three places and can access the required reservation in their desire hotels. The system may distributed provide the flexible, access for multiple user. Efficient implementation on J2EE is really needed when the applications are interactive and require better performance and low response time. It also addresses efficient access transparency and location transparency in three-tired distributed system. The system may reduce the user's processing time and reduce customer's overheads. It also supports better performance, faster response time and data consistency to guest.

8. References

[1] A. R. Lomuscio, M. Wooldridge, N. R. Jennings, "A Classification Scheme for Negotiation in Electronic Commerce", *Agent-Mediated Electronic Commerce:A European Perspective(eds. F.Dignum and C.Sierra*), Springer Verlag, 19-33.

[2] H. S. Nwana et al., "Agent-Mediated Electronic Commerce: Issues, Challenges and some Viewpoints", *Autonomous Agents'98*, MN,USA, 1998.

[3] M. Woodridge, N. R. Jennings and D. Kinny, "The Gaia Mehodology for Agent-Oriented Analysis and Design", *Autonomous Agents&Multi-Agent Systems*, Volume 3, Issue 3, September, 2000.

[4] N. R. Jennings, "An Agent-based Approach for Building Complex Software Systems", *Communications of the ACM*, April 2001, Vol.44, No.4.

[5] N. R. Jennings and M. Woodridge, "Agent Technology: Foundations", *Applications and Markets*, Springer Verlag, 1998.

[6] R. Guttman, A. Moukas, and P. Maes, "Agent-mediated Electronic Commerce: A Survey", *Knowledge Engineering Review*, 13(2):147-159, 1998.

[7] S. Gritzalis, J. Iliadis, S. Oikonomopoulos, "Distributed Component Software Security Issues on Deploying a Secure Electronic Marketplace", *Information Management and Computer Security* journal, Vol.8, No.1, pp.5-13, 2000, MCB University Press.