Design and Implementation of Airline Reservation Web Services Using Service-oriented Architecture

Fang-Fang Chua, Syahrul Amri Bin Ngazizan, and Musa Bin Hassan

***Abstract*—In the past, application developers were to deal with issues with distributed system technologies in building applications mainly due to its lack of agility towards business process, costly revisions and low reusability. Service-oriented Architecture (SOA) is an architecture solution that underlined loose coupling and dynamic binding between services and one of the approaches to embrace SOA is through web services. Traditional approach in designing and implementing software is compared with the service-oriented approach in realizing web services. We adopt the SOA methodology (SOMA) from IBM and realizing its applicability in the domain of airline reservation services due to its high reusability. We identified and implemented the service engineering process and created the client program to invoke and composite the deployed airline reservation web services.**

***Index Terms*— Service-oriented Architecture (SOA), web services, airline reservation, reusability.**

I. INTRODUCTION

Responding directly to economic and competitive pressures, companies in the air services sector are working to closely manage costs while at the same time improving operational performance and overall customer service. Application consolidation and modernization are now seen as a key requirement for air carriers seeking to transform themselves into more agile and competitive enterprises. For this reason, a growing number of air services firms are evaluating the requirements, capabilities and benefits of a service-oriented architecture (SOA) for adoption consideration.

Powerful trends are forcing major changes in the air services industry. Competition from low-cost carriers, the growth in e-commerce and new safety and regulatory requirements are reshaping both the industry and individual carriers. To survive and prosper, airlines must focus on costs, price and yield. To meet those requirements, many are now exploring ways to use their IT infrastructures to transform their companies into more agile, profitable businesses.

Manuscript received January 17, 2010.

Fang-Fang Chua is with the Faculty of Information Technology, Multimedia University, Malaysia. (phone: 60383125406; fax: 60383125264; e-mail: fang2x81@gmail.com).

Syahrul Amri Bin Ngazizan was with the Faculty of Information Technology, Multimedia University, Malaysia. (e-mail: activ8ions@yahoo.co.uk).

Musa Bin Hassan is with the Faculty of Information Technology, Multimedia University, Malaysia. (e-mail: musahassan@gmail.com).

Airlines Reservation Web Services aim to provide reusable services based on service-oriented architecture principles and approaches. The service is realized through web services implementation, published into service registry, discovered and then to be composited into a complete web-based airline reservation services. We aim to increase the system flexibility and interoperability in the airline industry context.

II. LITERATURE REVIEW

A Service-oriented Architecture (SOA) is a specific type of distributed system in which the agents are 'services' [1]. SOA implementation is heavily utilizing the standards such as XML (eXtensible Markup Language), WSDL (Web Service Description Language), SOAP (Simple Object Access Protocol) and UDDI (Universal Description, Discovery and Integration). The service provider is required to publish a WSDL description of its developed service into the service registry using UDDI standard. The service requester can then access, invoke and composite the required published web services using the client interfaces. Compared to the service processes proposed by [2], [3], [4] and [5], we have utilized the proposed SOA methodology by IBM [2] in modelling, designing and implementing the web services due to its proven feasibility. As shown in Figure 1, the SOA layer comprises of

Operational: This layer comprises database, service registry and applications approach.
Service Components: One or more independent components can be integrated to develop a service.
Services: The complexity of the services can be varies. Some of the developed services are required to be combined to perform the whole process or operations.
Business Process: The developed services should be reusable to cater the business operations.
Presentation: This layer is where the user will invoke all the published services.

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

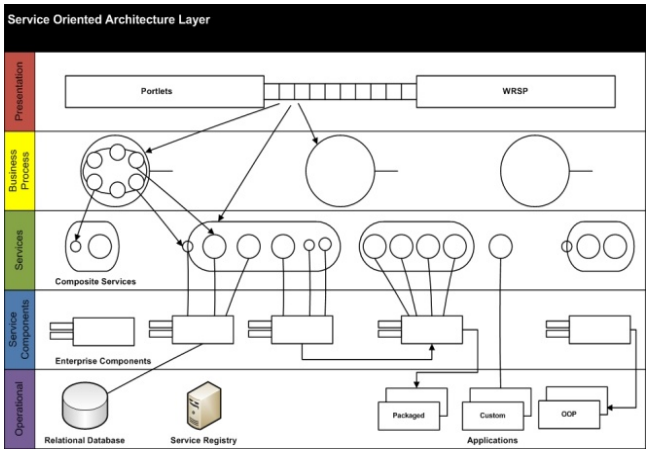

Figure 1. Service-oriented Architectural Layer

To fully leverage the significant benefits of a service-driven infrastructure, airline companies should overcome the limitations of the previous generation of legacy processing solutions. The comparison of traditional software-oriented approach and SOA approach in developing the airline reservation web services is shown in Table 1 and Table 2.

Table 1. Traditional Software-oriented Approach

Traditional Approach	
Speed	Slow
User Choice Flexibility	The traditional airline reservation system does not offer flexibility to offer alternative routes or other connecting routes from other systems since the systems are not connected with other airlines systems.
Simplicity	Complicated user interface
User-Friendliness	Yes.
Reusability	No.
Interoperability	The system is not platform and language independent.
Cost of Implementation	Very costly as well as maintenance costs incurred.

Table 2. Service-Oriented Approach

Service-Oriented Approach	
Speed	Considerably fast.
User Choice Flexibility	The new approach reservation services offer more routes selection even that routes are not offered by the respective airline company. The airlines reservation services are interconnected with other airline reservation services.
Simplicity	The layout interface is simple.
User-Friendliness	Yes.
Reusability	Yes. The service can be reused to develop other similar or improved services.
Interoperability	It is platform and language independent.
Cost of Implementation	Development and maintenance cost can be reduced.

III. PROPOSED SOLUTIONS

The service requirements for our airline services include User Management, Reservation Management and Payment Management. From the generic service requirements, detailed service specification can be identified. The designed service functionalities in our airline system 'VivaAir' include Customer Service, Flight Service, Reservation Services and Payment Service. We have chosen to use Macro Flow Diagram [5] to model our business processes such as Check Flight, Book Flight and Pay Ticket. Some of the examples of designed business processes are shown in Figure 2, 3 and 4.

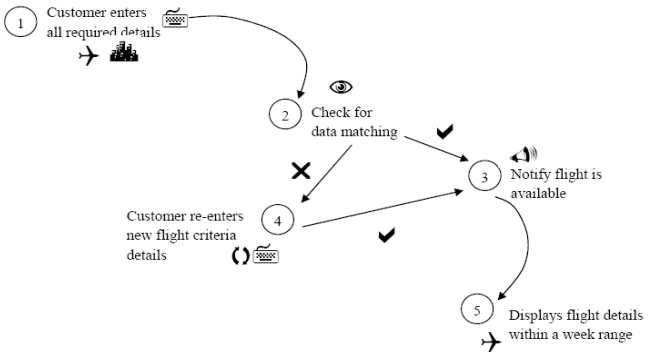

Figure 2. Macro Flow Diagram: Check Flight

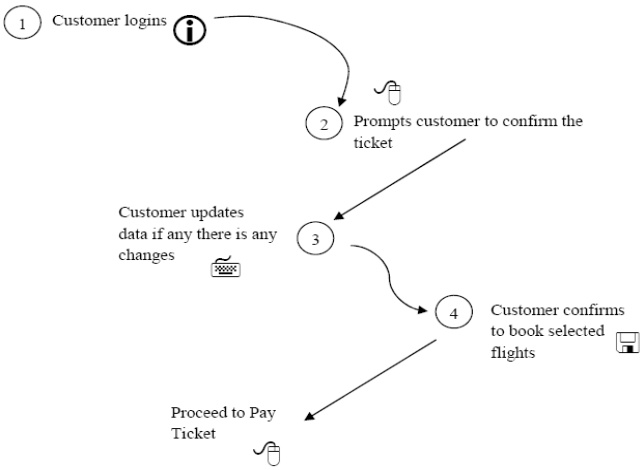

Figure 3. Macro Flow Diagram: Book Flight

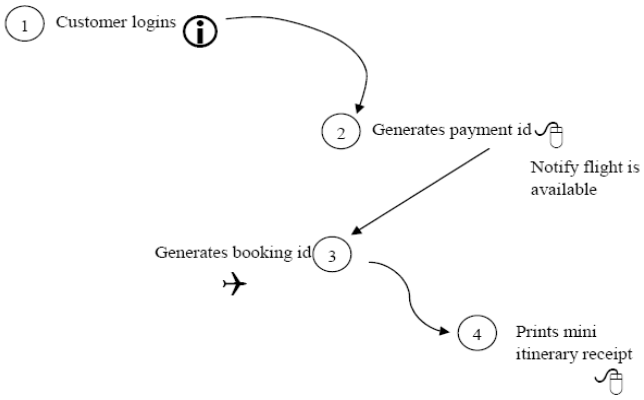

Figure 4. Macro Flow Diagram: Pay Ticket

The detailed design of services are being described in service model diagram [3] as shown in Figure 5 and the detailed descriptions of some of the web services operations are shown in Table 3.

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

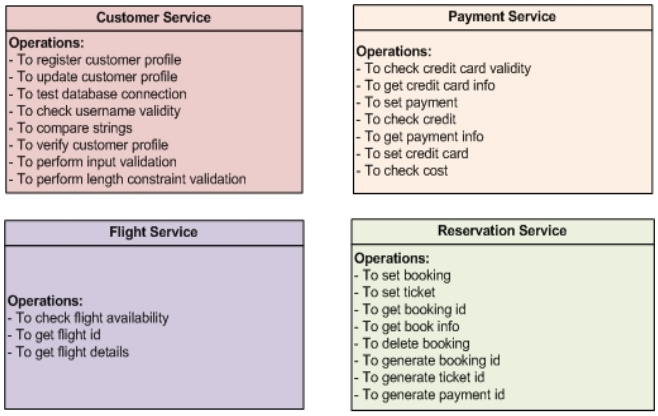

Figure 5. Service Model Diagram

Table 3. Web Services Operations

Web Method Name	Web Service Description
checkAvailability	This web service is developed to check the availability of the flights on the specified dates.
deleteBooking	This web service is developed to delete or cancel confirmed ticket.
dbICustomer	This web service is developed to register customer's profile.
dbUCustomer	This web service is developed to update customer's profile if any changes to the existing data.
setBooking	This web service is developed to book flight ticket.
setPayment	This web service is developed to make a payment for the ticket that has successfully confirmed or bought.
checkCreditValidity	This web service is developed to check credit card validity whether it has been declined or not.

IV. IMPLEMENTATION RESULTS

We have used NetBeans 6.5 [6], mySQL and Photoshop to design and implement our airline reservation system called 'VivaAir' using web services approach. The web services will then be published into UDDI registry and clients may invoke and compose the services. xHTML, JSP, CSS, JavaScript, AJAX and XML technologies are being integrated in creating the client interfaces. The seven main business scenarios which are being implemented are described in details as below.

The business scenario (Register Profile) is as follows:

- The customer should register himself in order to proceed to book ticket service.
- The customer needs to input all the required particular details during the registration process.
- The web service will perform validation checks on customer input and length constraints.
- Upon successful login, the customer will be registered officially to the web service and he can login using his username and password.
- The guest is only permitted to check flight availability.

The business scenario (Update Profile) is as follows:

- The customer should login by himself in order to perform update profile process.
- The customer will only change those fields that need to be updated.
- Then, any changes will be updated respectively.

The business scenario (Check Flight) is as follows:

- The customer is permitted to search available flights based on the origin city, destination city, departure date and return date.
- The web service will display any matching records based on the search criteria entered.
- The web service will notify the customer about the flight availability.
- If the searched flights are available, then web service will display flights which are within a week. Otherwise, the web service will prompt to ask the user to re-enter new searching criteria.

The business scenario (Book Ticket) is as follows:

- From the Check Flight, the customer is required to log in and the web service will prompt the customer to confirm the flights.
- The web service will then ask the customer whether to update his profile details or not.
- Subsequently, the customer will be asked to purchase and confirm the selected flights.

The business scenario (Cancel Ticket) is as follows:

- The customer is required to be logged in before he has privilege to cancel his confirmed tickets.
- The customer will select which ticket to be cancelled in the booking history.
- Once the confirmed ticket has been selected then the web service will delete the data off from the database.
- The customer is given cancellation ID upon successful transaction.

The business scenario (View Booking Status) is as follows:

- The customer is required to be logged in before he has privilege to cancel his confirmed tickets.
- The customer will select details in the booking history to view booking status.

The business scenario (Pay Ticket) is as follows:

- After Book Ticket, the web service will generate payment ID upon successful transaction is made.
- Once it has been confirmed, the web service will generate booking id to the customer and require the customer to print out the mini itinerary receipt.

Some of the system screenshots are being shown in Figure 6, 7, 8 and 9. Figure 6 allows the customer to search available

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

flights that matches the criteria by the customer. If the searched flight is available then, the customer can proceed to Select Flight and so forth. Figure 7 displays all the available flights within a week from the date entered earlier. The customer can click Next button to proceed to Flight Price. Figure 8 displays all the flight details with the fare costs and the customer will need to login in order to proceed to Book Ticket page. Figure 9 displays all the payment details and the customer may click Purchase button to buy the flight ticket.

Figure 6. Book Flight: Search Flight

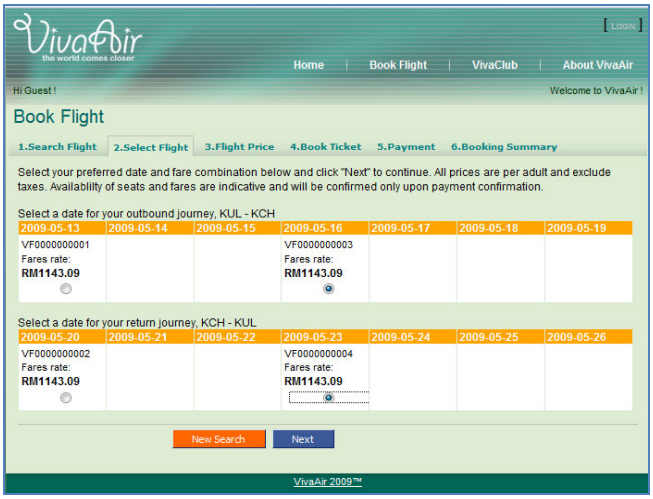

Figure 7.Book Flight: Select Flight

Figure 8. Book Flight: Flight Price

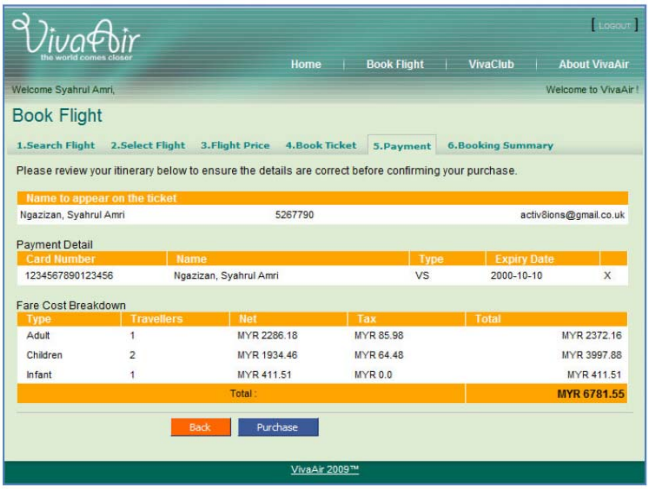

Figure 9. Book Flight: Payment

V.CONCLUSION

As a result of implementation, SOA approach cut-off development time of the web services. The services created are reusable and flexible to be integrated with other web service applications. Conclusively, SOA is a buzzword today and many organizations and industries are in race to adopt SOA in order to have competitive advantages for services delivery. However, it is important to ensure that the right approach is selected and the right capabilities are provisioned to ensure successful realization. It is recommended to select an approach or methodology based on the primary business drivers for adoption practice.

REFERENCES

[1] W3C (2003, May 14). Web Services Architecture. Available: http://www.w3.org/TR/2003/WD-ws-arch-20030514/

[2] Arsanjani, A. (2004, Nov 9). Service-oriented modeling and architecture – how *to* identify, specify, and realize services for your SOA. Available: https://www.ibm.com/developerworks/library/ws-soa-design1

[3] Erradi, A., Anand, S., Kulkani, N. (2006) SOAF: An Architectural Framework for Service Definition and Realization,On the proceedings of IEEE international conference on services computing (SCC'06), IEEE.

[4] Erl, T. (2005) Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall.

[5] Zimmermann, O. (2004). Elements of Service-Oriented Analysis and Design, IBM Developerworks.

[6] NetBeans Platform 6.5 (2008) Available: http://www.netbeans.org/

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)