Small Lodging Online Marketing & Reservation System

Félix E. Lópes López
Master of Engineering in Computer Engineering
Alfredo Cruz, Ph.D.
Electrical and Computer Engineering and Computer Science Department
Polytechnic University of Puerto Rico

Abstract — *Many industries have changed their way of operating due to the surge of the Internet, determined to capture a larger percent of the economy on the Web. One of the most solid industries transformed by the Internet is Tourism, especially processes such as hotel lodging and flight booking. Even with so many advances, many entrepreneurs are not taking advantage of the benefits that the Internet represents for their businesses. This project is a cost-effective alternative for the small hotel and lodging industry in Puerto Rico. The project is based on developing a local repository that can be integrated into an industrial Global Distribution System. The program will increase local tourism by exposure and outreach to other countries through the Internet. The solution permits the system to grow without impacting operations, making it a cost-effective solution. This project is made possible by Service Oriented Architectures and the benefits these offer.*

__Key Terms__ — Booking, GDS, Reservation, Reuse, SOA.

INTRODUCTION

This paper describes the development of a global distribution service (GDS) for small lodges and hotels in Puerto Rico. The GDS is an alternative for boosting, facilitating, optimizing and improving the efficiency of small hotel business management and marketing strategies. A GDS optimizes business processes related to the loss of reservations, overbooking, lack of information, or inefficient processes. Normally, buying and integrating a GDS will result in a 60% increase in income [1, 2]. In the Background and Acknowledge the reader can learn what some of the common booking problems are in the small hotel and lodging industry in PR, and why they are not maximizing their offers. In the Design section a design is presented to develop a GDS for small hotels and lodges. The reader will learn what are the architectural aspects used to support a SOA. Another area of the project discuss are the Application areas where the use cases are explain in detail as web services, and how they are organized for successfully making a reservation.

BACKGROUND AND ACKNOWLEDGE

There currently exist hundreds of guest houses, B&Bs, small hotels and other kinds of short term rental properties in Puerto Rico that do not outreach potential clients through the Internet. Reasons can go from having no website to lack of a good website that promotes their rental offers, to having a website that is not user-friendly or easy to interact with, The majority of these small businesses also lack a Global Distribution System (GDS) because of the high management costs this kind of system implies. In an attempt to increase their exposure and outreach on the web, small hotels, B&B's, guest houses and small lodges promote their services relying on marketing sites such as Clasificados Online among others.

Today, these marketing services are free or very cheap, being this the reason why they are so popular among the industry; but the truth is that they do not provide a mechanism to coordinate lodging effectively, They lack the necessary mechanisms to track and find the prospective clients in terms of customer profile, such as the number of times they seek room availability or complete a reservation [1, 2].

Another common problem that these small hotels encounter is that they create websites that are not user-friendly, intuitive, or business oriented. They lack the main asset this kind of website should have: the convenience of making online reservations [1, 3]. Another problem that even some

big hotel chains have is that when they do have a website, the reservation online process is not intuitive enough, and the payment process is way too complicated for the common user. This traduces to client loss because potential clients habitually feel insecure and tempted to abandon the reservation process. Some big hotel chains and their subsidiaries promote successfully over the internet because they manage their bookings through GDS. There are some GDS that are the main pillars of the online reservation business. Some of these are: Sabre, Pegasus, Expedia, among others; that have been operating for the past 20 years [1, 2, 4]. With the growth of the internet big hotel chains invest to transform their normal reservation process of selling through travel agencies towards reserving completely online. This allows the users to handle the reservation process by themselves, through an interactive experience. Clients can explore their dreams and travel alternatives by doing their own research on-line and choosing the best offers. Other things offered in this new procedure is being able to compare, share and choose among the offers, This will bring about an increase in online booking [1, 2]. Today, GDS manages multiple services such as: property and service management, amenities, services, tour guide services, day spa, golf car rentals, restaurants and restaurant reservations, tours and eco tours, customer relations and reservations. These services are implemented in diverse systems and environments that could be: web based app, client servers, web services or standalone applications. The implementation of any of the systems is a complicated and time consuming task which makes it expensive and difficult to manage for small hotels. It has been proven that internet listing is the most effective way to rank, promote, advertise and market products. This is based on research done on the growth of the internet since its establishment, and of the economy that has emerged and revolved around it, particularly in the last years. The rise of social media is an example of such economies. Mobile devices are an example of one the fastest growing market usages of applications at present, and with

excellent future growth projections. Today this market represents a sixty percent (60%) of the online users connected to the internet; and it is expected to continue growing in the near future [5]. The implementation of a GSD in the small lodge & hotel industry will result in an economic benefit to this industry because they will promote themselves on the most effective booking website applications available: VRBO, Kayak, Expedia, Travelocity, Hotel.com, Home Away, etc. This exposure can increase annual revenue in 60%. The creation of a good marketing strategy in the overall business can also increase the ROI in another 30% [3, 4].

DESIGN

To accomplish our objective we combined a methodology and an architectural model. To meet the requirements we followed the use cases methodology and defined the main user cases of the system [6]. Figure 1 shows the main use cases and their interactions.
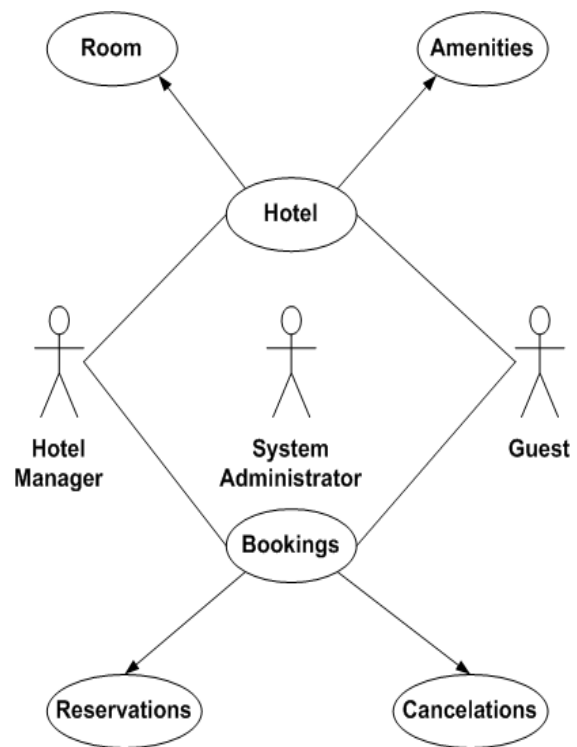

Figure 1
System Use Case

We can see that all of the actors (users) interact with all of the use cases at some moment, but they follow different courses. The main use cases defined are: Hotel, Room, Reservation, Payment and Availability. Some of the main functionalities of the use cases or modules provided are the Create, Read, Update and Delete (CRUD) operations. The information defined by each use case are available for web sites and by other systems that may interact with our system.

A brief description of the scope of each main use cases identify are follow here:

- **Hotel Use Case -** through this module the hotel personnel will be able to manage the hotel information.
- **Room Use Case** – through this use case or module the hotel personnel can manage the room information. The information we define here will be part of the information that will be available to the guest and other systems.
- **Amenities Use Case** - this use case bring the accessibility of the room amenities. This can be extra thing that the room have available for the guest.
- **Booking Use Cas*e*** - this service takes the dates supplied by the user and looks in the system to show all the available rooms within the date parameters.
- **Reservations Use Case** - the reserve use case will search for available rooms in a particular hotel and the selected dates and mark the room as reserved.
- **Cancelations Use Case** - this service's scope is to process the cancellation request of any reservation made by the guests, and to validate if any cancellation fee applies.

For the application development we followed the agile methodology and subscribed to the SOA. Our intent with the Service Oriented Architecture is to establish our application as a service in which all artifacts and components are exposed following the SOA principles [2, 4]. The main purpose of the Service Oriented Architecture is to develop all the artifacts as pure web services, assuring they are designed and developed following the SOA principles, to later reuse and recompose them as the business process requires.

With SOA the artifacts are exposed as a decoupled contract, contract mean web services description language (WSDL), this way system can interact with the services without worries about changes in the interfaces. Figure 2 show how the service contract are expose as decouple from the service logic. Figure 2 also show how the pattern bring the opportunity to leverage many contract version to support different service consumers needs. Decoupling the service contract from the service implementation give the business the chance to evolve without directly impacting any service consumer when service logic changes. This increase the refactoring service opportunities and the potential reuse of the service [7, 8, 9].
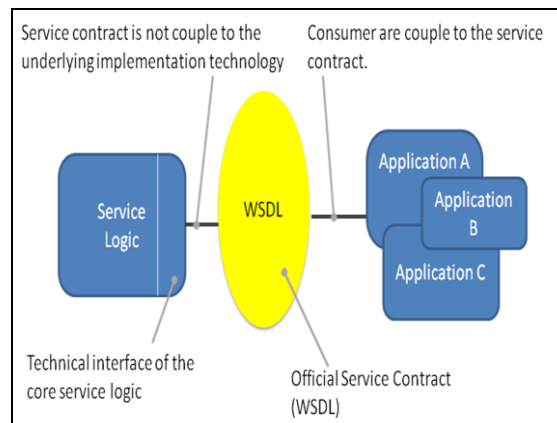

Figure 2
Decouple Contract Pattern

Some main strategy goals of SOA are to allow and facilitate the simultaneous interchange of data between programs of different or specific vendors and technologies, loose coupling, without additional programming or changes.

This level of service is highly reusable and makes it possible to reduce not only the level of dependency, but the development cycles and the maintenance costs. Reusability benefits business, and represents an increase in ROI. Having reusable services readily available also results in quicker time to market. Some of the benefits of establishing, achieving or accomplishing the

architectural model's strategy goals are the reduction in burden, increased ROI, and greater business agility. As mention in the book SOA: Principles of Service Design the main benefits of establish a SOA include [8]:

- Increased Intrinsic Interoperability
- Increased Federation
- Increased Vendor Diversification
- Increased Business and Technology Alignment
- Increased ROI
- Increased Organizational Agility
- Reduced IT Burden

The principles that SOA establishes as a framework (and which we propose to work with) are:

- **Standard service contract:** the services adhere to agreements defined by one or more service description documents.
- **Service reusability:** logic is divided into services with the intention of promoting reuse.
- **Service autonomy:** services that have control over the logic without any dependency of the encapsulated logic.
- **Service discoverability:** services that are supplemented with informative metadata or notes by which they can be effectively discovered for reuse and interpretation.
- **Service loose coupling:** this maintains a relationship with the client that minimizes dependencies.
- **Service abstraction:** aside from what is described in the service contract, the logic of the module is hidden from the outside world.
- **Service statelessness:** this minimizes the service resource needs or use by distributing the management of the state of information when necessary.
- **Service composition:** the services are composed as participants during a service activity no matter the dimension or the complexity of the composition.
- **Service granularity:** is a SOA design pattern consideration that provides an optimal scope of the information given and the granular level of

the business functionality in a service operation.

In our project we follow this principle to position the business logic as pure web services. This allow the application to provide information to multiple system and evolve with the business. Also by apply this principle also we assure the quality of the services as their reliability and the availability of the service.

The integration model in Figure 3 shows how the service design is integrated with different systems, devices, clients and applications in a reliable way. The SOA approach diversifies the technical environment and makes it possible for businesses to interact with each other without being concerned with the technology each one uses. Figure 3 lets us observe the different users and systems that interact with the different apps, even when they are developed and designed by a third party, using either the same technology or a different one.
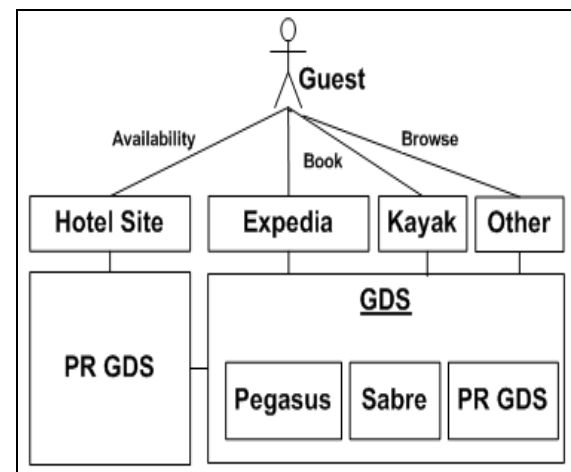

Figure 3
System Integration Model

The exchange between systems is made possible because of the security service layer placed to grant access to the resources. This project focuses on the creation of a scalable architecture where the GDS can be strategically positioned in order to increase the evolution of the different business needs. The architecture strives to enforce the usage of the SOA's principles, design and

standards in order to increase the evolution of the system. To be able to standardize and successfully position the architecture model we develop an internet application that automated the code generation. The code generated it's bases on the database schema define for the system. This approach increased the efficiency of the system life cycle by decreased the time and effort needed to develop new requirements or any adjustments needed. As shown in the Code Generation Model (see Figure 4), the programs take any value as a parameter to generate the code, or it is read from the db schema. These parameters are applied to a standard template to generate the final results (code classes). This drastically reduces the time and effort taken, as well as the risk involved in any future development.
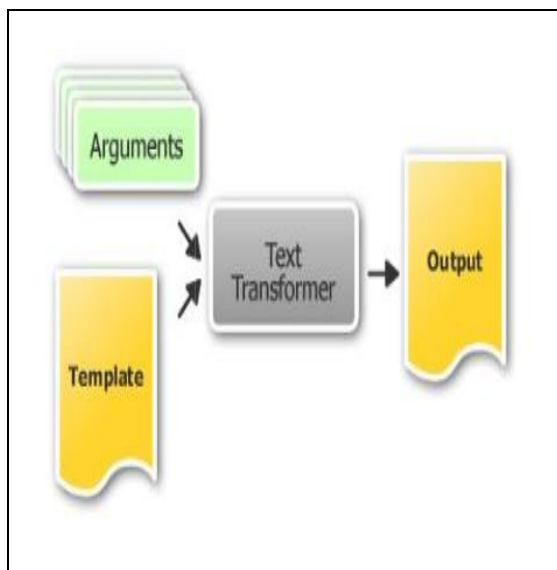

Figure 4
Code Generation

The application can manually take the parameters or it can taken from the database schema where we define the system tables. As mention before the outcome of this application utility are object classes. The templates that the application created were: object interface, object, business layer object, data access layer object, service interface and the service skeleton.

DATABASE

To meet the objective of the client, we identified the need of creating data storage where the business can keep their data. The service data storage was designed based on the services context. Most of these services are agnostic and they can be represented in a single database table.

Because of the business relationship between services, components and interfaces, it was concluded that the best way to represent the storage is with a relational database. This project uses the SQL Server database [4, 9]. Alternative database research was done using a non-structured database known as NoSQL database.

Following the author's recommendation found in the book "Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", what is emerging today is the integration of different kinds of db architectures in order to accomplish a specific task [9]. The use of NonSQL databases in the future is not discarded.

Once the technology was chosen, we move on to define ten tables in the database. There are relationships between the tables and the services or entities they represent. For each requirement or service the corresponding entities or database tables are defined.

The ERD model shows all the tables defined for the project (see Figure 5 a, b and c). The model depicts the different tables and relationships. It shows the main relationships with primary and reference keys among each entity. The tables defined in the ERD model are:

- **Hotel:** This table will contain all the hotels available to offer to the users.
- **Room:** This table will contain the main room information of the hotel available rooms. Some of the data fields available for the room entity are room quantity, prices and availability.
- **Guest:** The table has the data that defines the main demographic information of the guests among other data. An example of some main information today is the email address.
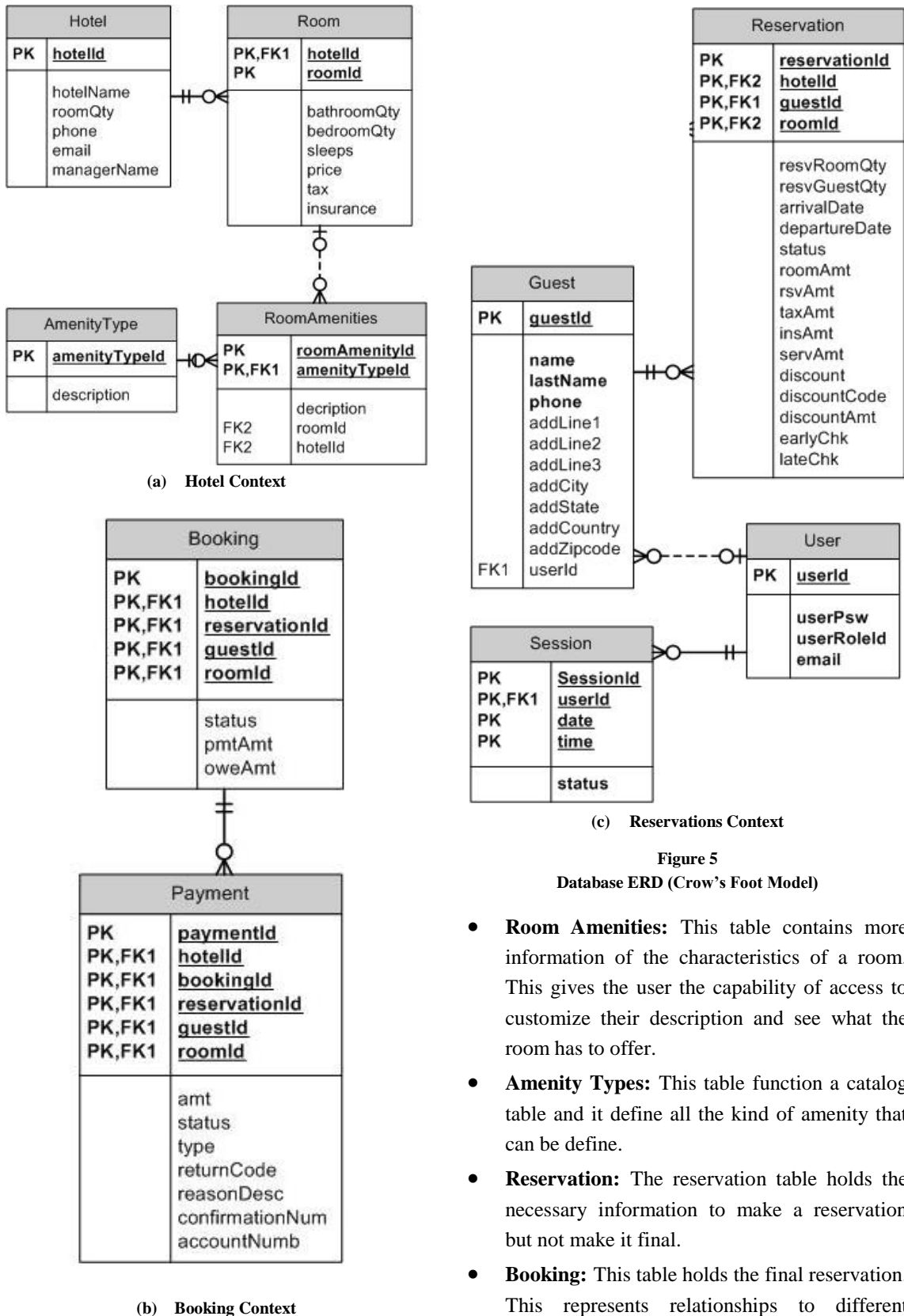
Hotel

PK	hotelId
	hotelName
	roomQty
	phone
	email
	managerName

Room

| PK,FK1 | hotelId |
PK	roomId
	bathroomQty
	bedroomQty
	sleeps
	price
	tax
	insurance

AmenityType

PK	amenityTypeId
	description

RoomAmenities

| PK | roomAmenityId |
PK,FK1	amenityTypeId
	decription
FK2	roomId
FK2	hotelId

(a) **Hotel Context**

Booking

PK	bookingId
PK,FK1	hotelId
PK,FK1	reservationId
PK,FK1	guestId
PK,FK1	roomId
	status
	pmtAmt
	oweAmt

Payment

PK	paymentId
PK,FK1	hotelId
PK,FK1	bookingId
PK,FK1	reservationId
PK,FK1	guestId
PK,FK1	roomId
	amt
	status
	type
	returnCode
	reasonDesc
	confirmationNum
	accountNumb

(b) **Booking Context**

Reservation

PK	reservationId
PK,FK2	hotelId
PK,FK1	guestId
PK,FK2	roomId
	resvRoomQty
	resvGuestQty
	arrivalDate
	departureDate
	status
	roomAmt
	rsvAmt
	taxAmt
	insAmt
	servAmt
	discount
	discountCode
	discountAmt
	earlyChk
	lateChk

Guest

PK	guestId
	name
	lastName
	phone
	addLine1
	addLine2
	addLine3
	addCity
	addState
	addCountry
	addZipcode
FK1	userId

User

PK	userId
	userPsw
	userRoleId
	email

Session

PK	SessionId
PK,FK1	userId
PK	date
PK	time
	status

(c) **Reservations Context**

Figure 5
Database ERD (Crow's Foot Model)

- **Room Amenities:** This table contains more information of the characteristics of a room. This gives the user the capability of access to customize their description and see what the room has to offer.

- **Amenity Types:** This table function a catalog table and it define all the kind of amenity that can be define.

- **Reservation:** The reservation table holds the necessary information to make a reservation but not make it final.

- **Booking:** This table holds the final reservation. This represents relationships to different

payments that may be related to one single reservation.

- **Session:** This table defines the mechanism to safeguard the usage of services. Sessions are passed to the service parameter to authorize service capability of usage.
- **User:** This table holds the information of the users that have access to the service resources. All table resources are related to a particular user.
- **Payment:** This table holds the transaction detail of each monetary transaction requested by the system.

APPLICATION

The main objective of this project is to demonstrate that the GDS can support and improve the functionality of reserving or booking a room in a lodge or hotel. To achieve the goal of the project two types of functionalities has been identified: the agnostic services which support the business entity or task resources, and the orchestrated business services which support the main business tasks such as room availability or room reservation by using a mixture of multiple services. One important step in SOA is to understand well the business process. To be able to have a deep understanding of the process the SOA Design Process as establish in the book, Service-Oriented Architecture (SOA): Concepts, Technology, and Design, recommend us to document or model each of the process that we need to automated [2].

Follow the recommendation we define the process to support the main business task. First users must authenticate their identity in the application. If the guest or user is successfully authorized then they can use their capabilities to request the operations provided by the resources. When a hotel manager subscribes to the services, they are authorized to publish offers and information on the hotels that he/she operates. Main information includes room status, prices, contact information, etc. (see Figure 6 and 7). After hotel and room information has been created, guests can start browsing the services catalog to see the offers and search for available rooms and packages.

Once a guest access the system and browse for a room he can proceed to make a reservation (see Figure 8). To complete the reservation, the guest has to interact with the reservation process established by the business standards and identified by the best practices.

Figure 6
Hotel Description and Room Listing

Figure 7
Hotel Listing and Contact Information

The reservation and the booking process flow as: The user selects a room. Once the room is selected, the guest user information and room

availability is validated. With the valid ID of the chosen room and the guest's ID information the system can invoke the payment process, and proceeding with the payment. If the payment is successful then the room reservation will be booked and completed. If the payment process is declined for any reason, then no reservation will be completed by the system.

Figure 8
Room Description
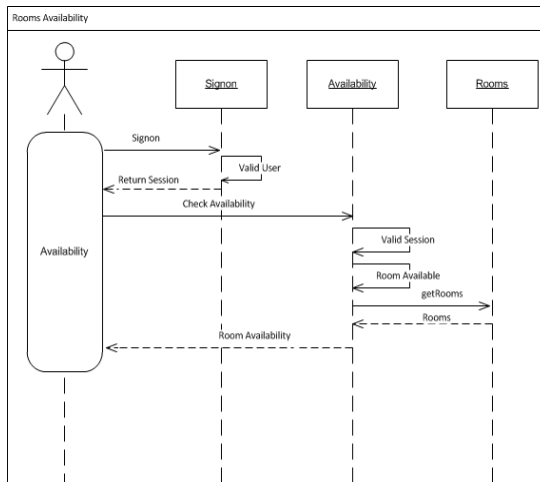
Figure 9
Availability Sequence Diagram

The availability service has the purpose of showing the client the available rooms in a specific time frame. Figure 9 shows the service activity to supply room availability. First, the service user has to sign in to the service provider. Once the authorization is granted the client can request the service to check room availability. The availability service performs three tasks: it validates the authorization granted; checks the room availability, and finally; gets the available room detail. A list of rooms with details is returned to the client as a service response.
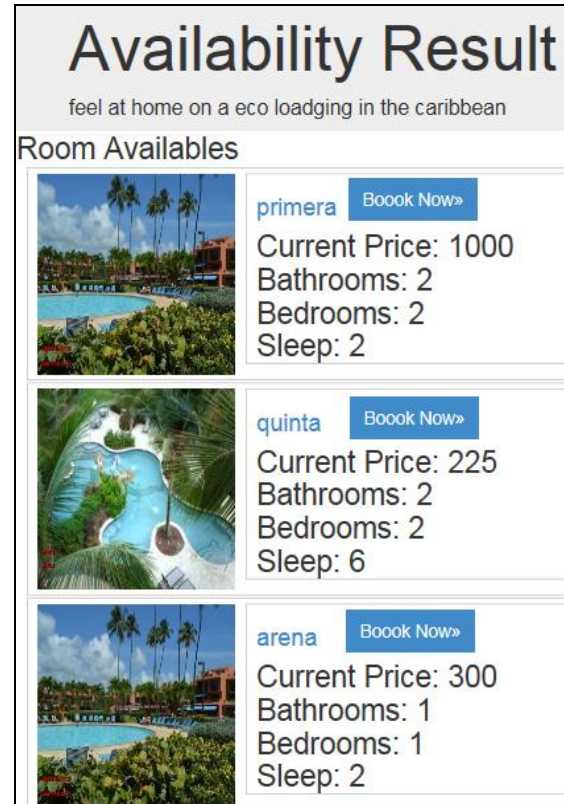
Figure 10
Availability Result Page

Figure 10 depicts how the application shows the available offers to the user. The Figure shows a list of available rooms and their amenities. Some of these amenities are room quantity, number of persons allowed, and price per night.

The next images Figure 11 (a) and (b) show the service activities that the client has to fulfill to make an active reservation. The model shows the interaction between the signing in process, the availability, the guest, the reservation, the booking and the payment services. The process is controlled by the consumer.
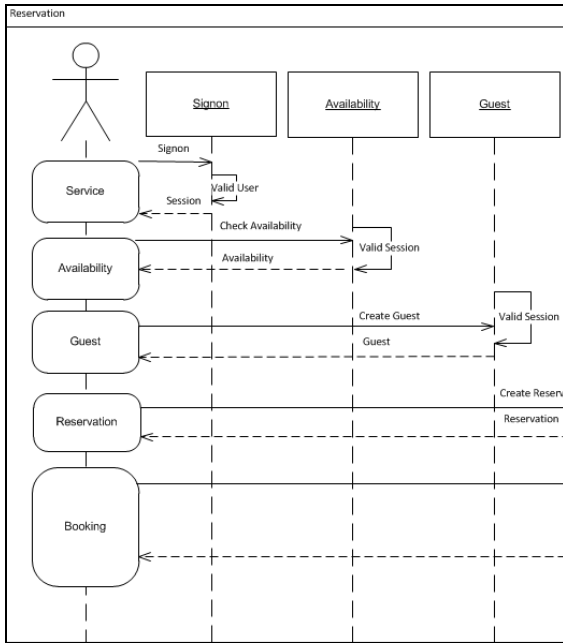
(a) Signon
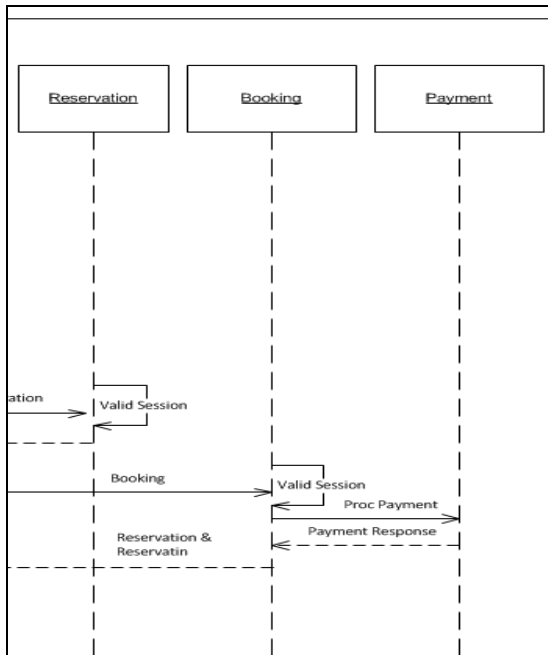
(b) Reservation and Payment

Figure 11
Reservation Process

The "Availability Dates Confirmation Page" shows how the information is requested by the client (see Figure 12). Once the dates are entered the service validates room availability; guest information can be defined for the reservation, as shown in Figure 13.

Figure 12
Availability Dates Confirmation Page

After the user enters the guest's information a summary of the reservation is shown to the client. Some of the information shown to the guest is: room prices, taxes, insurance fees, service fees and the reservation total. Figure 14 shows how the information is presented to the user.

Figure 13
Guest Information Page

Figure 14
Payment Information Page

If the client agrees with the reservation fee he must provide the credit card number, the name that appears in the card, expiration month and year, and the card security code. After entering the credit card information, the service processes the payment and if successful, a confirmation number is given to the client.

FUTURE WORK

Future work of the system can continue by defining the administrative module that will interact with the established service in order to allow the hotel administration to interact with its own data and manipulate it. In addition, integrating a main GDS such as Sabre, Expedia, Pegasus, among others, should be explored to maximize hotel offers. Another ongoing point is to explore the pattern implementation, best practices and standards of the travel industry. One of these standards is the Open Travel Alliance which has been working in the past year with the standardization exchange of the travel industry.

CONCLUSION

The main business objective of this project was to develop a promotional platform within the small hotels and lodges that could increase the income with a minimal effort and investment. The proposed service seeks the automation of the current hotel services; ease of use, availability, reservation process, and publication of the hotel information. The delivery service works as a decentralized data repository where other systems or GDS can integrate within it to look for hotel information and increase hotel exposure. Although there is still much to do to create a final product, this project is a start point future analysis and exploration of small hotel and lodging promotions and reservation systems. With this project we have been able to improve and apply different concepts, patterns, best practices, and standards. We have gained a broader experience and understanding of SOA applications. The project has been a challenge, but it has also been a great experience.

REFERENCES

[1] R. D. Reid and D. C. Bojanic, *Hospitality Marketing Management*, John Wiley & Sons, 2010.

[2] O. Lee, *Internet Marketing Research: Theory and Practice*, Ebrary, 2001.

[3] R. Conrady and M. Buck, *Trends and Issues in Global Tourism*, Springer, 2008.

[4] R. Dholakia, N. Dholakia, W. Fritz and N. Mundorf, *Global e-commerce and online marketing: Watching the evolution*, Quorum Books, 2002.

[5] P. J. Sadalage and M. Fowle, *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*, Pearson Education, 2012.

[6] H. Podeswa, *UML for the IT Business Analyst - A Practical Guide to Object-Oriented Requirements Gathering*, Thomson Course Technology PTR, 2005.

[7] T. Erl, *Service-Oriented Architecture (SOA): Concepts, Technology and Design*, Pearson Education, 2005.

[8] T. Erl, *SOA: Principles of Service Design*, Pearson Education, 2007.

[9] T. Erl, *SOA Design Patterns*, Pearson Education, 2009.