The future of e-Service provision in Marginalised Areas: Online Negotiation

Nobert Rangarirai Jere[1], Mamello Thinyane[2] and Alfredo Terzoli[3]

[1] Computer Science Department, University of Fort Hare, Telkom Centre of Excellence in ICTD Alice, 5700, South Africa

[2] Computer Science Department, University of Fort Hare, Telkom Centre of Excellence in ICTD Alice, 5700, South Africa

[3] Computer Science Department, Rhodes University, Telkom Centre of Excellence in Distributed Multimedia Grahamstown, South Africa

Abstract

As the amount of commercial transactions carried out on t he Internet increases, the interest to partially or totally automate the negotiation of the terms of these transactions has rapidly become an interesting research topic. Several e-marketing strategies through various online websites have been developed. In rural communities e-services applications such as e-commerce, e-health, e-judiciary and e-government have been developed. However, most of these applications are failing to fully benefit the people in the societies because they are no modifications which are done to the applications as technology changes. This paper discusses the development of an intelligent negotiation module to enhance an e-commerce platform. The paper explains a negotiation application that has been developed and plugged to any existing e-Commerce shopping portal for the Dwesa Community in the Eastern cape province of South Africa and proposes an architecture for all other e-Services. This project explained is undertaken within the Siyakhula Living Lab.

Keywords: e-services, e-commerce, negotiation, DRP

1. Introduction

The application explained in this paper is based on the existing case study called the Siyakhula Living Lab[1] (SLL) undertaken in Dwesa. The name Siyakhula, means that we are "growing together" [1]. The University of Fort Hare (UFH) and Rhodes University (RU) run it jointly, both Universities are located in the Eastern Cape Province of South Africa. The mission of the Siyakhula Living Lab is to develop and field-test the prototype of a multi-functional, distributed community communication platform for deployment in marginalized and semi-marginalized communities in South Africa [1]. SLL aims to develop the marginalized community by equipping people in the area with the necessary technological skills to be able to support projects deployed. It shows how marginalized communities that are very difficult to reach, may in future be joined with the greater South African and African communities to the economic, social and cultural benefit of all [2]. Some of the current examples of the current projects and e-Services are explained below:

The initial objectives of the ICT4D intervention in Dwesa were to develop a prototype of an e-Commerce [2] platform for the arts and crafts entrepreneurs in the community, and also for the possible exploration of micro-tourism potential in the area. The application explained in this paper is part of the ICTD application which is undertaken within the SLL. The application is referred in this paper as Dwesa Rewarding Program (DRP). The aim of the development of this additional module discussed in this paper is to be able to attract more customers buying online and give them rewards based on different factors such as buying behavior, value of order and location [3].

Many people across the world shop online. However, some of the customers may be discouraged by the fact that they would not be able to negotiate for prices when shopping online. Therefore the paper explains an intelligent application meant to allow customers all over the world, buying from www.dwesa.com, a shopping portal which sells artifacts products [3]. The products are some of the outputs of the rural entrepreneurs in the Eastern Cape Province in South Africa.

[1] *www.dwesa.org*

[2] *www.dwesa.com*

1.1 Definition of Negotiation

Bichler and other authors, define negotiation as an iterative communication and decision making process between two or more negotiators (parties or their representatives) who [4]:

- Cannot achieve their objectives through unilateral actions,
- Exchange communicative acts comprising offers, counter-offers and arguments,
- Deal with interdependent tasks,
- Search for a co nsensus which is a co mpromise decision.

Traditionally negotiation has been done through face–to-face interactions. However, with the changes in technologies, advances in the telecommunications and development in e-Commerce systems, negotiation is now possible using electrical gadgets such as computers. Customers are now able to negotiate when buying products online. With the advent of e-Commerce, online businesses have become more popular than before. Placing online orders, making payments electronically and finding information about the products and the vendors have become easier [5]. Despite the comfort that e-Commerce has brought with it, humans are still involved in most of the important processes of business, for example, in making decisions in all phases of buying and selling [5].

Negotiation is one of the key factors in commercial systems involving a lot of decision making and tradeoffs between various factors. Some of the practical applications that assist users in negotiation are Auctionbot, Kasbah, Tete-a-tete, e-Bay [6]. Most business transactions in e-Commerce involve negotiation to settle on the most suitable price for both parties [3]. During negotiation, individuals or organizations have to make decisions of a varied nature to attain their objectives. The purpose of the negotiation module discussed in this paper is to reward loyal customers, give customers the opportunity of a counter-offer on prices and to enhance the e-Commerce platform in a marginalized area.

The functions that the negotiators use to increase or decrease their offers and counter-offers as a buyer and as a seller are defined as follows [5]:

For Buyers

If max_price > std_price then
Offered price := std_price – D
Else
Offered price := max_price – D

For Sellers

If min_price > std_price then
Offered price := min_price + D
Else
Offered price := std_price + D

Formula 1 For buyers and sellers

Where, *std_price* is the market price provided by the matchmaker and *D* is a constant. It is calculated based on the current situation of the market [5]. For both the buyer and the seller the values of *D* should not exceed their preferred prices (R). Therefore, it can be concluded that buyers always aim to pay less when buying. In contrast, sellers aim to sell at higher prices.

2. Negotiation architecture

In designing a negotiation module, the following negotiation terms are crucial:

- *Negotiation protocol* consists of a set of rules that govern the interaction among the negotiation parties. Some examples of the rules are: permissible types of participants: negotiators, third parties; negotiation states: accepting bids, negotiation closed; valid actions of the participant in particular states [7]
- *Negotiation objects* are ranges of issues over which agreement must be reached.
- *Reasoning model* is the apparatus that participants employ in order to achieve their negotiation objectives. For instance, it is a m echanism by which the next counter-offer is calculated. Some of the strategies developed are argumentation, persuasion and heuristics-based. Obviously, the selection of the reasoning model depends on both the protocol and the negotiation object [7].

In order to have the required feedback, the architecture of the negotiation server needs a co mponent to record the history of negotiations [8]. For example, the history of an on-going negotiation transaction (i.e., proposal exchanges) can be used to determine the speed of concession applied by the counterpart (counterpart concession). In addition, the negotiation system can “learn” from experience (i.e., the history of previous negotiation transactions with a counterpart) to generate new policies (i.e., mappings from negotiation contexts to negotiation goals). And also strategies (i.e., mappings from negotiation goals to negotiation plans to be explained in the next section) and/or to modify the existing policies and strategies dynamically at run-time [8].

The proposed negotiation application follows the basic structure of a negotiation server with different components such as these explained below [8]. Negotiation Transaction Manager includes:

Negotiation Scheduler: Responsible for initiating a new negotiation transaction / session when the Negotiation Server receives a transaction / session message.

Negotiation Session Processor: Responsible for processing a negotiation session. A *negotiation transaction* is defined as a sequence of negotiation steps carried out by a pair of negotiation servers that lead to an agreement or disagreement in a negotiation process [Haifei *et al*., 2005]. Each negotiation step is called a *negotiation session*. These sessions are managed by the shop owners.

- *Event/Trigger/Rule (ETR) Server*: Responsible for receiving events from the Negotiation Transaction Manager and triggers the proper decision-action rules to relax constraints, to inform the user [Haifei *et al*., 2005].This is a connection link to the database and the customers.
- *Cost/Benefit Module (CBM)*: Responsible for performing cost-benefit evaluation of alternatives based on the pre-registered preference scoring and aggregation methods provided by the negotiation parties [Haifei *et al*., 2005]. This is the intelligent module that is responsible for decision making.
- *Negotiation Messaging:* Provides the sender and receiver for Negotiation Servers to communicate with each other using a push communication model.
- *Negotiation Repository:* Provides a persistent storage to store a variety of negotiation data. This makes use of the database. These terms have been explained since they have assisted us in understanding the negotiation module.

3. Negotiation process

The diagram below shows a FlexFlow modeling of a simple negotiation between a buyer and a seller [9]. The process is as shown in Figure 1 below:

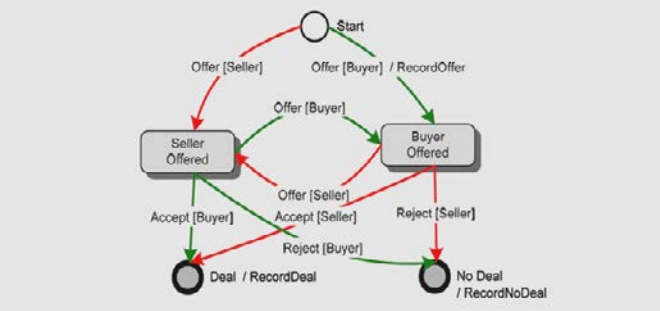

Figure 1 Negotiation Process [10]

Figure 1 shows that for every negotiation there is an offer and an acceptance. Figure 1 shows the communication process between the system and the customers. Just like any negotiation application, the module in this research is similar to the other online negotiation processes. There should be an agreement between the parties, and if there is no agreement it means there is no deal. Again, if there is agreement then there is an acceptance of the offer.

The negotiation process can be divided into four stages which include information collecting, searching/offer gathering, negotiating, and evaluating [10]. There is a need to gather the necessary information required to start the negotiation process. Information such as customer details, location, customer age, customer buying behavior and the order history of the customer are gathered to have clear negotiation goals. In decision making, these factors have to be analyzed and the process may take a while when trying to resolve the conflict [10]. Therefore, to overcome the problems in delay, in the development of this application, the processing time has a time span and different sessions used to manage the time frames.

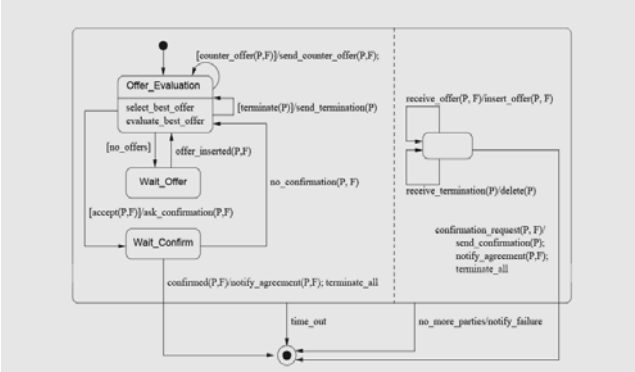

Figure 1 Negotiation Control Module [11]

Figure 2 shows how the control module of the negotiation application works. One needs to wait for offers and make evaluation on the available offers, and then make a final decision to be communicated between the negotiation parties. Notice that it is not the buyer but the sellers who are responsible for generating the initial offers [11]. Accordingly, we use the term "offer" to designate the prices proposed by the sellers, and the term "counter-offer" to refer to the prices proposed by the buyer. The reasoning module of the negotiator is composed of two separate logical programs: one for selecting the next offer to be evaluated, and the other for evaluating an offer [11].

We consider a negotiator responsible for negotiating a price with several potential sellers on behalf of a buyer. Offers sent by the sellers are arranged in a queue. The outcome of the evaluation of an offer can be one of three actions: accept the offer, propose a counter-offer, or terminate the negotiation with the corresponding party. During the negotiation process, if an offer is accepted, the negotiator asks the other party to confirm the agreement and upon receiving confirmation, the negotiator terminates the whole negotiation process (i.e. all the threads). If on the other hand a message confirming the refusal is received, or the confirmation message is not replied to after an agreed delay, an occurrence of event "no confirmation (P,F)" is generated, and the negotiator continues to negotiate with the other parties [12].

4. Negotiation context and goal

Every business enterprise operates in a mini-world of business, in which the enterprise has access to some of the information, material, financial, and personnel resources that exist in the real world [8]. All these resources have to be identified and acquired so that the negotiation process starts. The information required may be on the inventory details, customer information and the current market conditions, all of which are important for setting the goals of negotiations. Some of the information may be stored in the enterprise's local database and/or application systems. Others may be accessible from remote databases or

application systems, for example, calling the methods of remote objects that encapsulate these databases and application systems [8].

During the negotiation process, it is necessary to have a negotiation goal. In the development of this e-negotiation module, the goal is to allow customers to counter-offer the prices and have a chance to negotiate for desired prices. Of course, the idea is to sell the products above the cost price and make customers get some rewards through negotiation. The negotiation goal is aimed at creating a customer loyalty relationship, thus the idea is to let those customers who deserve to succeed during the negotiation process get a fair deal based on their order history and all the necessary details from the database.

5. Negotiation strategies

There are various techniques and strategies used for coming up with the negotiation rules, goals and planning. Thus, having gathered some facts, the following are some common negotiation strategies [13]. These strategies enhance the decision making in the negotiation process. According to Rahwan et al, these are some of the negotiation strategies [13]:

- Desperate Strategy: This is a very simple strategy in which the time constraints may be important and the negotiator wants to close a deal fast. In this strategy, as soon as a sub-negotiator finds an acceptable offer, the coordinating negotiator accepts it and sends messages to all other sub-negotiators to terminate their negotiation [13].
- Patient Strategy: In this strategy, even if an acceptable deal is found by one or more sub-negotiator(s), these negotiators are asked to wait while other negotiators are asked to resume their negotiations. Once all sub-negotiators complete their negotiation process (whether with success or failure), the best offer is chosen [13].
- Optimized Patient Strategy: In this strategy, the coordinating negotiator uses information about one negotiation outcome to influence the performance of other sub-negotiators. The constraints on the utility for the other sub-negotiators is updated in order to avoid unnecessary deals which are not as good as the one already found [13].

An understanding on some of the negotiation strategies allows the developer to have an overview of all the alternatives available during the negotiation process. These different strategies were used during the development of the negotiation module and gave an understanding of how the negotiation strategy should process and send messages.

6. Negotiation rules

For the negotiation application to function fairly and properly, the negotiation rules were defined and set right. These rules determine how the overall decision is computed and used by the reasoning module to provide feedback. For the development of the DRP, we came up with rules that are used for the development of the negotiation module. We aim to make the negotiation process as real as if shopping face-to-face. We have also designed simple rules that can be modified at any time. Different factors were considered for negotiation. Most of the factors are captured when the customer is completing the registration form. All these details are stored in the database. Some of these factors include the customer's age, income, order history and number of points. However, all these factors are dynamic and can be modified at any time.

Some of the negotiation key rules are:

- Negotiation can only be done after adding items to cart;
- Most items can be negotiated for;
- Customers should negotiate between the Cost price and the Selling price;
- No items can be negotiated below the cost price;
- Negotiation makes use of functions, for example, negotiation by age;
- Customers have to make offers;
- A valid offer is automatically calculated. This should be greater than cost price;
- A certain percentage to be deducted per customer is calculated;
- The final price that each item could be negotiated to is calculated per customer.

In coming up with these rules we have considered the nature of the business and its size. We have noticed that rural entrepreneurs in Dwesa wish to sell as many items as possible at the same time to earn a living from the sales. Therefore, the rules we listed above are meant to keep the business going and allow the shop owners to reward customers and also get profits. Hence, it should be clear that for the DRP there is no item that is sold below the cost price after negotiation. Having considered the rules above, different formulas were used for the development of the negotiation modules. Some of the formulas to support the above mentioned rules are:

For making any offer:

Offer_Price > Cost_price (Valid offer)
If Offer_Price < Cost_price then
Offer_Price is invalid

Formula 2 validating offer

Percentage discount to be deducted:

per_final = (per_final/high_poss_val) * 100.

This gives the value in percentage that should be considered for the valuation of the offers made by the customer. *High_poss_value*, is obtained from the database after getting all the activated factors which are in the database.

Negotiation prices

diff = Selling_price - Cost_price

Since the *Offer_Price* should be between the *Cost_Price* and the *Selling_Price*, calculating the difference between these prices help in the final price calculated during negotiation.

Final Acceptable negotiation price

*final_price = Cost _price + (diff - (diff * per_final/100))*

Formula 3 Acceptable price

Since the value of *diff* is explained above, and *per_final*, the *final_price* which would be accepted by the system is calculated from the recent formula.

From the formula:

If Offer_Price >= final_price
Then Accept offer
Else Send a message offer is too low

Formula 4 Low offer

Table 1 Negotiation factors

Factor Name	LOW	MEDIUM	HIGH
Age	5	15	20
Income	0	10	15
Points	5	20	30
Order History	10	15	20

In setting up the negotiation rules, the above table shows the details that could be in the database with the factors to be considered and the different values. The values differ from customer to customer. The details, according to which customers' inputs are calculated and ranked differently, are listed in the above table. For example, for a customer with age greater than 65, the value of 20 is given, indicating that greater priority is afforded to pensioners. Again, someone with points greater than 1000, is regarded as a loyal customer and we aim to promote such people, thus their value is 30. This means customers with higher points are given higher priority and percentage discounts during negotiation. In this paper, several factors were considered for the development of the negotiation module. These factors form most of the rules to be used during the development of the negotiation application or module.

6.1 Analysis of negotiation factors

We have identified different factors to be used for negotiation. These factors could be used in setting up the negotiation rules. However, those listed below are some of the examples of the negotiation factors we have used for the DRP negotiation module. Therefore, listed below are some of the factors that could be used for negotiation.

- Negotiation based on Age

Those customers between 18 years to 25 years are classified as the youth and may not be working. All the customers above 65 are old. These two groups of customers stand a higher chance of negotiating for lower prices and more rewards. Customers between 26 to 64 years form part of the working class and customers in this age group have the lowest percentage discount when negotiating on the basis of age. In negotiating with these customers various other factors like the points available and the order history are considered.

- Negotiation based on Cost Price / Selling Price

In making the negotiation rules, we have considered that there is no product that is sold below the cost price or no reward will be offered at the expense of the business. Thus, there is a need to carefully consider the counter-offer prices from the customers. The moment that the offer price is less than the cost price the negotiation process ends. Hence, the relationship between the cost price and the selling price is very critical in making the decision.

- Negotiation based on Profit Margins

The volume of sales and the profit margins also come in during the negotiation process. If the business is making a loss due to low sales volumes, then chances of getting rewards and paying less if negotiation is initiated are very high. Again, if the business is just breaking even with low sales, having a positive attitude during the negotiation may also allow customers to get more. However, in times of high sales volume and high profit margins, it may be difficult to negotiate and win much. Of course, rewards are there, but a lot of factors on the customer details and the order history are to be considered, which makes it difficult to win during the negotiation.

- Negotiation based on Order History

Considering the frequency of the customers, value of the order and the highest amount which customers have spent also form part of the negotiation process. Those customers who are always buying in bulk and spending more have higher chances of winning when negotiating.

The above are the main factors we have considered for the development of the negotiation module. However, there is a level of randomness in some cases when the negotiation decision is made. The system may add a random element to simulate the non deterministic nature of real life negotiation decision based on the previous decisions available in the negotiation application knowledge base. The above factors were considered to explain how the rules explained earlier in the paper have been derived from. Of

course, these factors are dynamic and can be modified at any time by shop owners.

6.2 Negotiation overview

The unique function provided in this paper, is for customers to be able to negotiate for rewards or for a discount with the system. DRP aims to create dialogue between the system and its customers. Customers make offers on the marked selling prices on each item after adding the items to their cart. Negotiation is a two-way process between the customer and the system and instant feedback is expected. An intelligent negotiation module is developed in this research.

Different factors are considered for the negotiation process before the decision on how to give a reward is made. Some of the examples that the system has to check include the transaction history, the number of points available, the season, overall sales during the negotiation time and the value of the order. Customers' details are also considered. Some of the details include age, income, location and number of dependants. Through negotiating, customers increase their chances of paying lower prices. As noted by MoneyhighStreet Staff, one of the best ways of saving money, if you are buying an expensive item or service, is through negotiation with the supplier [14]. Below are some of the best ways of negotiating which may enable those who are making use of b*uy at Dwesa (Dwesa Shopping portal)* to get rewards through negotiating.

- Rural shops need to sell the products to their customers

Having goods, but no buyers, is a huge cost to shop owners so they need to shift their stock, maintain cash flow and replace old stock with new stock [14]. Customers play an important role in the success of any business. Therefore, even on *buy at Dwesa,* customers are critical and thus the project intends to allow them to negotiate for rewards using points. Shops always compete for customers and very few good shop owners like to see customers walking away empty handed and in turn buy from their competitors [14].

- All shops need to make profits

Having shown that shops want and need to sell to customers, one needs to remember that shops and service suppliers need to make a profit. There is no point in asking for a big discount or reward if it means that the shop does not make any money from the deal. Good negotiation should benefit both the customers and the shop [14]. This means that the shop benefits from a customer, by making a profit on the sale and by you, the buyer, spreading the word to your friends and family about the said shop. Customers have to benefit from the negotiation too, by getting a r educed price or discount of some sort and rewards. This supports the objective of Siyakhula Living Lab projects, where this project was deployed, which is to generate income for the poor people in the community.

- Customers should be reasonable when making offers

Negotiating for a 10% discount will probably be more effective than striking out for a larger reduction as this is more likely to make the deal unprofitable for the shop or supplier. To be successful in negotiations, customers have to present a good reason for their request [15]. By backing up a discount request with sound facts, customers are making a stronger negotiation stance and are more likely to succeed [14]. The above negotiation tips are applicable for all the customers purchasing on *buy at Dwesa* and increases their chances of getting the desired rewards. Therefore, the project gives points to customers, allows them to negotiate with the system and calculates the discount or reward to be communicated to the customer instantly.

6.2.2 Examples of a customer negotiating

The diagram below shows the page that is displayed to the customer after adding the intended items to their cart. The customer is given an option to negotiate for the price on some of the items added to the cart. The total amount of the order and the total points for buying the items are displayed

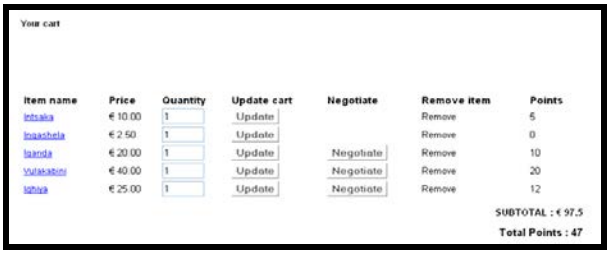

Figure 3 Negotiation after adding items to Cart

This menu allows the customers to begin negotiating after adding some items to their cart. The points added to the customers' points which account for buying the items added to the cart are displayed as shown above (total points 47). For some items, customers can negotiate allowing them to give some offers for the items added. However, some items are not negotiable, as already highlighted. When the customer clicks on negotiate, the offer button appears as indicated below. The offer button allows the customer to make an offer on the price displayed. The customer can make offers on the items added to their cart.

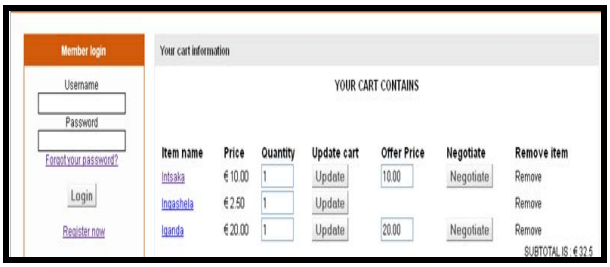

Figure 4 Customer Making Offers

During the negotiation process customers have the chance to offer different prices. There are different messages displayed to the customers depending on the offer price made by the customer. In cases where the customer makes an offer that is too low, a corresponding message is communicated to the customer. The messages are aimed at assisting the customers on the offers the system might accept for successful negotiation.

If the customer's offer price is too low the customer is asked to make another offer until the given number of negotiation attempts are used up. The system gives a counter-offer and the customer can accept or reject the counter-offer. By accepting the deal, the difference between the selling price and the negotiated amount has to be subtracted from the original total price of the items in the cart.

Figure 6 shows the message communicated to the customer after making a low offer. Customers are limited in relation to the number of offers they are able to make, to minimize the negotiation tasks and to improve the efficiency of the system. We have used the bar graph to guide the customer when making offers during negotiation and improve customer feedback.

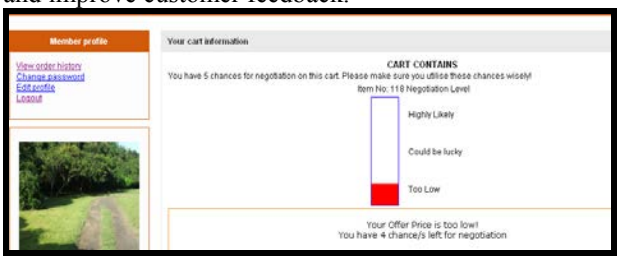

Figure 6 Customer offer too low

When a customer makes a good offer acceptable to the system, a different message is displayed. The message is as displayed below.

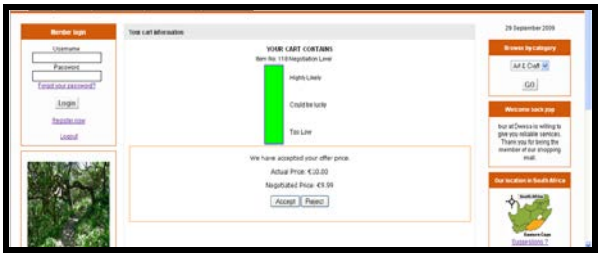

Figure 5 Customer offer accepted

The screen shot above shows that the system has accepted the offer made by the customer. The customers have two options, either to accept or reject the deal, after that. Choosing to accept deducts the negotiated price from the selling price and reduces the subtotal price. Rejecting the offer allows the customer to exit the current page.The system can send a message containing a counter-offer to the customer during the negotiation process. It should be noted that the counter-offer message is only displayed in cases where the customer's offer is above the cost price. The customer may choose to accept the counter-offer and complete the negotiation process. Rejecting the offer gives the customer a chance to make a new offer.

The negotiation factors can be modified and updated at any particular time. During the negotiation process the system gets the updated values and factors from the database. Using this interface allows the shop owners to add or edit the negotiation factors. There is an option to select from the already listed factors or to type in a new factor. The factors are given different values and have different levels. The shop owners determine these levels and values.

Figure 7 Setting negotiation factors

7. Proposed Architecture

Based on the current state of the e-Service applications which are currently available for the Dwesa community: The following have been noticed:

- Each application is a standalone
- Community members feel the applications are too many to master all of them
- Training on ICTs and deployed application on a daily basis
- The network is still managed with the universities
- Expense incurred by the universities to the Dwesa community
- The other 3 schools relies on one school where the V sat is deployed
- No access to computers when schools are closed
- The future was not considered during the deployment of the e-Service e.g all the applications are computer web based none runs on mobile phones.

There is need to have a basic architecture which could be used to empower rural communities with services that could assist them in the near future. An example of such architecture is like the one that has been proposed for the e-Government services by the New Zealand government. In this paper we have borrowed the same architecture and propose that for the future e-Service applications within the Dwesa rural community. This could also be a good framework for other rural communities in Africa.

The networking Government in New Zealand came up with some ideas which could be used in developing the architecture. The reason for the proposal of the architecture

is to provide a platform for the e-Services deployed for rural communities. In the architecture we have classified the components into six categories or building blocks. The categories are a u seful construct to identify the required business functionality in a generic service delivery process, together with the components that need to exist to provide that functionality [16]. These categories are:

- user access;
- user services and guidance;
- service enabling tools;
- connection tools;
- business delivery systems; and surrounding e-government environment - governance, policy and management regime.

The architecture requires that some elements of agencies' service delivery will in future be developed from an all-of-government perspective (i.e. 'develop once, use many times') [16]. In particular, there are benefits to having a common architecture for:

- how services are presented to people (User services & guidance);
- how service delivery is actually electronically enabled (Service enabling tools); and
- how agencies connect to one another and their customers (Connection tools).

The architecture has been designed so that it applies to the public sector (all of government), individual organisations, or business units within organisations. Having deconstructed the generic components of service delivery processes, we can more clearly define the role of different actors in these processes. Any service delivery process comprises all five "boxes", from user access through to business system. In a loosely coupled environment different actors may have different roles for defining and providing the different boxes. They may come from separate agencies.

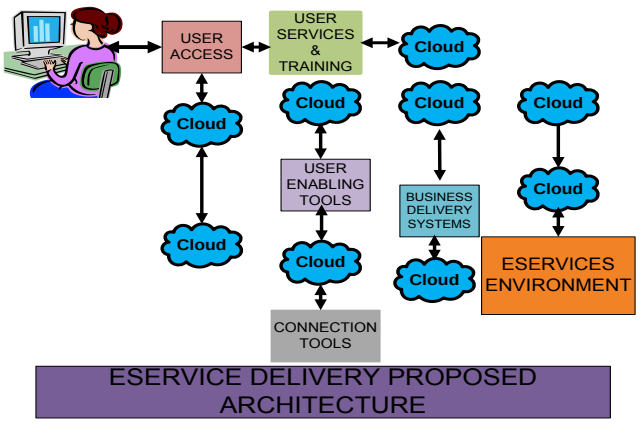

Figure 8 e-Services Architecture

The overall implication of this architecture for stakeholders in e-Service deliveries is increased flexibility in the way services are delivered. It will allow single agencies to potentially provide more varied services than they do today (say, by providing over the counter or web-based access to services provided by other agencies). It will also allow new types of integrated services to be delivered by groups of agencies working together to make use of the components that will be built to support the architecture. The cloud in this case represent that the services would be accessible through the internet. The proposed architecture is attached separately after the references section and is Adopted from the Networking Government in New Zealand [16]

8. Conclusion

The negotiation module forms the greater part of the rewarding module discussed in this paper. It describes an intelligent application that allows customers to give their price offers and get instant feedback while shopping online. The negotiation module is dependent on the other modules, for instance it relies on the points allocation module and the database for its reasoning. However, the other modules were not discussed in this paper. The final reward is determined by the power of the customer to negotiate. Therefore, this work reported in this paper is part of the other three main modules namely points allocation, negotiation and the rewarding modules. As indicated the other modules were discussed in other papers. However, we decided to concentrate on giving details on the negotiation module to fully explain how e-Commerce businesses in marginalized areas could be improved in an attempt to improve e-Services deployed in rural areas. As part of enabling the future of these services, the proposed architecture explained in this paper could be useful to stakeholders who might have some interest in deploying e-Service applications in rural areas. Considering such architectures could assist in providing sustainable e-Services to rural communities. The idea of this paper is to find innovative ways of improving the services from e-Service applications and to develop e-Services which could improve rural livelihoods in the long run.

10. References

[1] Siyakhula project. Dwesa e-Commerce platform project: 2007. Available at: www.dwesa.org

[2] P. Tarwireyi, A. Terzoli, and H.N Muyingi. Adapter-based revenue management system for the exploration of non-conventional billing options in new markets for telecommunications. SATNAC conference Wild Coast, Eastern Cape Province, South Africa. 2008. Available at: www.satnac.org.za/proceedings/2008/management.htm

[3] N.R. Jere, M. Thinyane and A. Terzoli. Augmenting an e-Commerce service for Marginalized Communities: A Rewards Based Marketing Approach. SATNAC conference 2009 R oyal Swazi Spa Swaziland.

[4] M. Bichler., J. Kalagnanam., K. Katircioglu., A.J.King., R.D. Lawrence., H.S. Lee., G.Y. Lin,., and Y. Lu. Application of flexible pricing in business-to-business electronic commerce, IBM Systems Journal,2002. 41 (2): 287-302.

[5] M. Chhaya, and J. Vassileva, Bilateral Negotiation with Incomplete and Uncertain Information: A Decision-Theoretic Approach using a Model of the Opponent. Cooperative

Information Agents IV, LNAI. 2004.
[6] W.Y. Wong, D.M. Zhang, M. K. Ali. Negotiating With Experience CSIRO Mathematical and Information Sciences Locked Bag 17, North Ryde NSW 2113 Australia. 2007.
[7] C. Sierra, P. Faratin, and N.R. Jennings. A Service-Oriented Negotiation Model between Autonomous Agents. Proc. 8th European Workshop on M odeling Autonomous Agents in a Multi-Agent World (MAAMAW-97), 1997. Ronneby, Sweden, 17-35.
[8] L. Haifei, Y. Stanley, and L. Herman,.Automated E-business Negotiation, Model, Life Cycle and System Architecture. Database Systems Research and Development Center, University of Florida. 2005.
[9] A. Reuter, and F. Schwenkreis. ConTracts – A Low-Level Mechanism for Building General-Purpose Workflow Management Systems: IEEE Computer Society, Bulletin of the Technical Committee on DataEngineering 2005.
[10] M. Rakesh, A. Mitchell, J.S. Cohen. FlexFlow: Workflow for Interactive Internet Applications. 2002. Available at: http://www2002.org/CDROM/poster/194.pdf
[11] M. Dumas, G. Governatori, M.P. Wil, M. Hofstede, and P. Oaks. A Formal Approach to Negotiating Agent. Development Centre for Information Technology, Innovation 2 School of ITEE, Queensland University of Technology, Australia. 2003.
[12] B.V. Elsevier Science. A Formal Approach to Negotiating Agents Development. 2002. Available at: www.elsevier.com/locate/ecra
[13] I. Rahwan, R. Kowalcyczyk, H.H Pham, Intelligent Agents for Automated One-to-Many e-Commerce Negotiation. Australian Computer Society. 2001. 4: 197-203.
[14] MoneyHighStreet Staff. How to Negotiate A Discount And Save A Fortune. 2007. Available at: www.moneyhighstreet.com/feature/68/
[15] K.W. Dickson, S.C. Chiu, and C.K. Patrick, Facilitating e-Negotiation Processes with Semantic Web Technologies. Department of Computer Science and Enginnering, Conference paper, Chinese University of Hong Kong. 2003.
[16] Networking Government in New Zealand e-Government, Designing of the Architecture, 2010. Available at http://www.egovt.nz

Nobert Rangarirai Jere is currently pursuing his PhD in Computer Science degree at the University of Fort Hare, Alice, South Africa. His area of interest is e-Commerce development, rural development through ICTs and t he future of ICTs. Has published 6 papers for conferences and 2 for journals

Mamello Thinyane is currently the Head of Center of Excellence in ICTD and Senior Lecturer within the Computer Science Department at the University of Fort Hare. His academic interest are ICTs in rural development, Mobile applications, IKS and Social Networking. He has published many papers in both journals and conferences.

Alfredo Terzoli is a Professor in Computer Science Department and the project Director of the Rhodes University Centre of Excellence in Distributed Multimedia. His research areas of interest are real-time multimedia over packet networks and developmental ICT. He has published many papers in both journals and conferences.