Int. J. Open Problems Compt. Math., Vol. 6, No. 4, December, 2013
ISSN 1998-6262; Copyright © ICSRS Publication, 2013
www.i-csrs.org

Web based Calculator of Genetic Algorithms and Modeling of Packets Forwarding Mechanism

Valentin Hristov

66 Ivan Michailov str., Blagoevgrad 2700, Department of Computer Systems and Technologies, Southwest University "N. Rilski", Bulgaria.
e-mail: v_hristov@hotmail.com

(***Communicated by Firas Ibrahim Alzobi***)

Abstract

Development of a Web-based application for modeling and calculation of genetic algorithms with advanced functionality is proposed and experimentally realized. Brief description of the nature of genetic algorithms is given. The content of XML (eXtensible Markup Language) file for modeling of the genetic algorithms is represented. The framework of the proposed system is given and technologies for its realization are specified. The applicability of the proposed system is discussed. Working screens of the developed model of packets forwarding mechanism are represented and achieved results are verified.

Keywords: *Genetic Algorithms, Packets forwarding mechanism, Evolutionary Computing, XML.*

1 Introduction

Inspired by the Darwin's theory of evolution, genetic algorithms are part of evolutionary computing, which is a rapidly growing area of artificial intelligence.

The purpose of this work is to develop means for modeling and calculation of genetic algorithms in the Web environment that allows solving different tasks, without requiring special qualification of the users (they only need to know the basic ideas and terminology of genetic algorithms). Easy usage of the realized system is provided by appropriate interactive tools for modeling and editing of genetic algorithms. XML technologies enable to integrate and adapt the system work with other systems (e.g. Matlab) without requiring creation of additional processing modules of the system.

As a whole the Web based calculator assures the overcoming of the negative effects of the geographical and temporal communication fragmentation. Such one practical realization economizes time and costs.

Artificial intelligence creates a wide range of methods, which have found application in practice, including genetic algorithms [4].

Some fundamentals of genetic algorithms are represented in [7]. The biological foundations of the genetic algorithms are explained very briefly below.

The complete set of genetic material (all chromosomes) is called genome. Chromosomes consist of genes, blocks of DNA, each gene encodes a specific protein (can be said that each gene encodes a different feature, such as eye color). Alleles are called the possible combinations of values for individual feature (e.g. blue and brown for eye color). Each gene also has its place (a position) in the chromosome.

During the reproduction, the genes of the parents formed an entirely new chromosome by recombination (or crossover). New produced offspring then undergo mutation, i.e. elements of DNA change.

Adaptability of the organism is measured by the success in his life.

These algorithms are used when pursuing a specific result (objective), when the solution requires a relatively large time resource or in cases where the solution is not known or has no solution. Algorithm starts with a set of solutions (represented by chromosomes with specific information about genes) called initial population. According to their viability are chosen solutions to form the next population (offspring). To more appropriate decisions (decisions are compared in terms of pursued result/goal) are given better chances for reproduction. New population is expected to be better than the old. This is repeated until some condition (for example: a number of generations or a sufficiently good solution) is satisfied.

The sequence in the genetic algorithm can be represented as follows:

1) generate initial random population of n chromosomes (solutions);

2) calculating the viability f (x) of each chromosome in the population n (in the target function - called "fitness function") and identification of chromosomes with priority for the next population (m in number, m <n);

3) establishing a new population by repeating following steps until the new population is completed:

3.1) preserving the predetermined number m of the best solutions (according to their fitness - the fitness function values);

3.2) election of two parental chromosomes of m chromosomes;

3.3) using of crossover to cross the parents to form the next generation (children);

3.4) using of mutation to mutate the newly created chromosomes;

3.5) pasting the new generation in the new population (adding n-m new chromosomes and filling the population);

3.6) replacement- using newly generated population for the further implementation of the algorithm;

4) stop and return the report if the final check-condition is satisfied;

5) loop, go to step 2).

In [6] are represented several interactive Java applets for the demonstration of the performance of genetic algorithms. To illustrate the performance of genetic algorithms in [6] is shown solving a problem for extreme of a function. A genetic algorithm tries to find the minimum of the function of the type z=f(x,y). The applet visualizes the graphics of the function (search space) and decisions. Presented applet allows to the user to input different z=f(x,y) functions, i.e. except for the demonstration it can serve to solve problems with genetic algorithms in which fitness function can be represented as function of two parameters (i.e. the chromosome with two genes), which significantly limits its functionality.

This restriction of the applet is related to the idea for the realization of the calculator, proposed in this paper. The purpose is to implement Web-based calculator of genetic algorithms without limitation in the number of genes in the chromosome and with advanced functionality in the description of the fitness function and other parameters of the algorithm (the number of chromosomes in the population and generations) as well as to modeling a packets forwarding mechanism.

2 Framework of Web Based Calculator and XML Model of Genetic Algorithms

As already stated purpose of the work is the development of Web-based calculator of genetic algorithms with advanced functionality. The basic algorithm of the calculator [5] follows steps 1) to 5), shown above. Framework of the proposal is shown on Fig. 1.

Many web technologies [6,7,8,9] are used for the development of the specialized XML editor, to save XML model on the server and to generate working file for calculation of the genetic algorithm as well as to perform genetic algorithm calculations and send the results of calculations back to the user.

The work with Web-based calculator [5] presented in Fig.1, takes place in the following sequence:

1) the user creates a description of the specific problem model (genetic algorithm) that is stored in an XML file;

2) under certain rules, from this XML file, is generated a PHP file to calculate the specific genetic algorithm, i.e. PHP1 reads XML file and generates PHP2 file named with the name of the particular project (genetic algorithm);

3) the execution is redirected to PHP2 (generated file) that performs calculations and generates a response to the user according to selected by him specific form.

On this way developed experimental system assures remote creation of models of genetic algorithms and receiving the results for very wide field of cases.

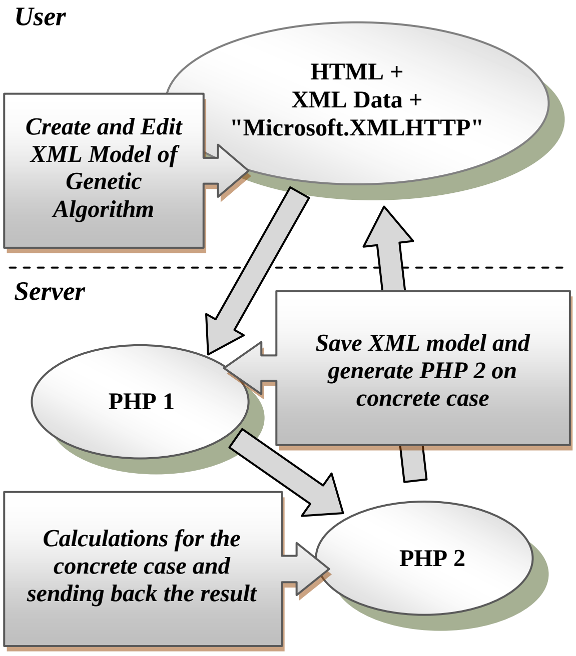


Fig. 1 Technologies used in development of the experimental system and realized functions

As mentioned above, we focus on XML modeling of genetic algorithms and this model is directly related to the user interface (means for creation and editing). The developed user interface is shown on Fig. 2 and the information about creating a personal model is given.

As it is set out below the fields, which form the description of the parameters of genetic algorithm and chromosomes, are as follows:

A) Description of the parameters of genetic algorithm:

- Name - name of the project (under this name the project is stored on the server);

- Type - minimum or maximum (a type of genetic algorithm calculations);

- Fitness calculations of the Chromosomes (fitness function of genetic algorithm) - objective of the genetic algorithm;

- Population - number of chromosomes (individuals);

- Generations - number of generations after which the genetic algorithm stops and provides the results of calculations;

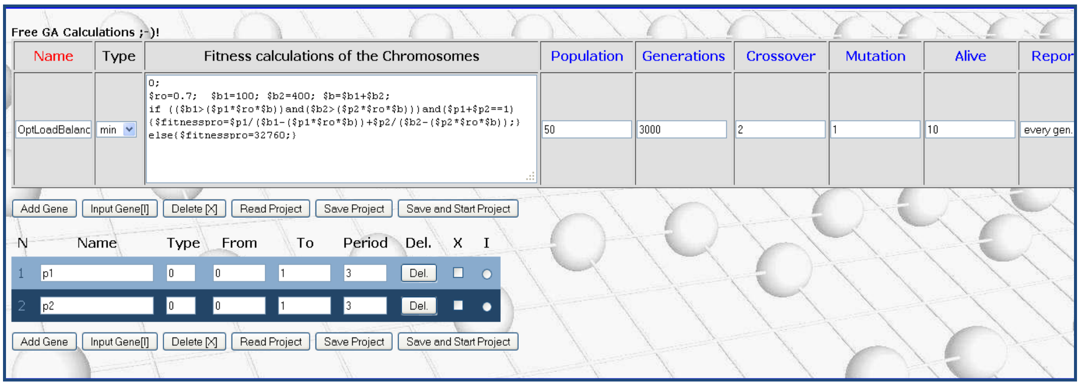
- Alive - number of chromosomes (organisms) m, which "remain alive" (preserving a predetermined number of best solutions), after the fitness check;

- Crossover - in our case shows how many genes will be inherited from the father during the generation of new chromosome (individual) - the remaining number of genes to the completion of their total number in the chromosome are inherited from the mother;

- Mutation - determines the number of genes that can mutate after the formation of new chromosomes;

- Report - specifies the type of report (in our case: whether the statement to include the best obtained result from every generation or only the result of the last generation).

The description of the fitness function is under the rules of PHP [7]. The only requirement is that variables begin with $, and the name is the name of one of the genes, for example, if genes are named GN1 and GN2, the expression $GN1 + $GN2 returns the sum of both genes and pow($GN1, $GN2) the value of $GN1 to the power of $GN2 and etc. (See also next section).



Free GA Calculations ;-)!

Name	Type	Population	Generations	Crossover	Mutation	Alive	Repor
OptLoadBalanc	min	50	3000	2	1	10	every gen.

Fitness calculations of the Chromosomes

```
0;
$ro=0.7;  $b1=100; $b2=400; $b=$b1+$b2;
if (($b1>($p1*$ro*$b))and($b2>($p2*$ro*$b)))and($p1+$p2==1)
{$fitnesspro=$p1/($b1-($p1*$ro*$b))+$p2/($b2-($p2*$ro*$b));}
else{$fitnesspro=32760;}
```

N	Name	Type	From	To	Period	Del.	X	I
1	p1	0	0	1	3	Del.	☐	○
2	p2	0	0	1	3	Del.	☐	●

Fig. 2 Creating and editing the XML model

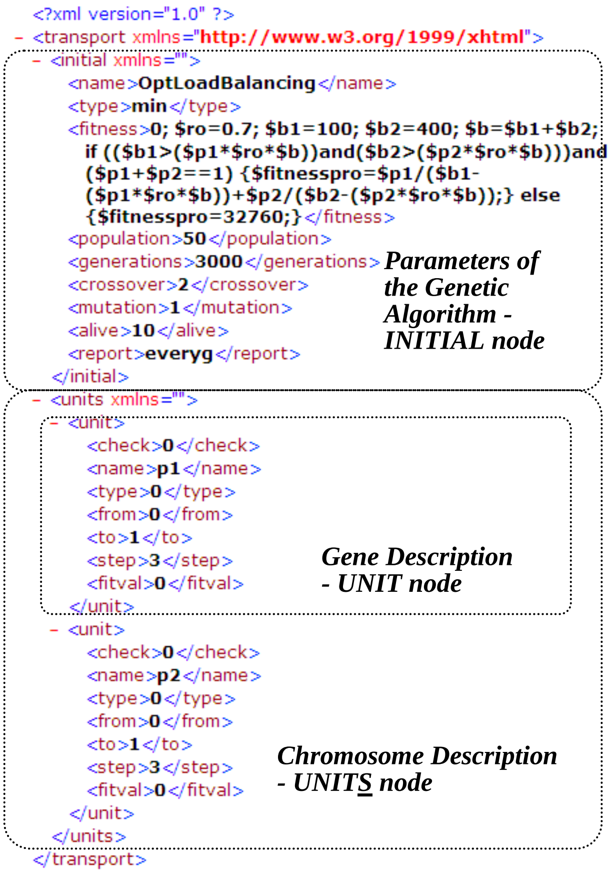
B) Description of chromosomes (genes in the chromosome):

- Name - name of the gene;

- Type - type of information to describe genes (“0” - digital);

- From-To - set boundaries to define the field to search for a solution for that gene (assuming the two boundaries are the same values From=To when it comes to genes whose value is previously known to remain unchanged in calculations);

- Period/Step - determine the accuracy of the calculations by showing the number of significant digits after the decimal point.



```
  <?xml version="1.0" ?>
- <transport xmlns="http://www.w3.org/1999/xhtml">
  - <initial xmlns="">
      <name>OptLoadBalancing</name>
      <type>min</type>
      <fitness>0; $ro=0.7; $b1=100; $b2=400; $b=$b1+$b2;
        if (($b1>($p1*$ro*$b))and($b2>($p2*$ro*$b)))and
        ($p1+$p2==1) {$fitnesspro=$p1/($b1-
        ($p1*$ro*$b))+$p2/($b2-($p2*$ro*$b));} else
        {$fitnesspro=32760;}</fitness>
      <population>50</population>
      <generations>3000</generations>
      <crossover>2</crossover>
      <mutation>1</mutation>
      <alive>10</alive>
      <report>everyg</report>
    </initial>
  - <units xmlns="">
    - <unit>
        <check>0</check>
        <name>p1</name>
        <type>0</type>
        <from>0</from>
        <to>1</to>
        <step>3</step>
        <fitval>0</fitval>
      </unit>
    - <unit>
        <check>0</check>
        <name>p2</name>
        <type>0</type>
        <from>0</from>
        <to>1</to>
        <step>3</step>
        <fitval>0</fitval>
      </unit>
    </units>
  </transport>
```

Fig. 3 XML model contents

To illustrate the contents of the XML model presented on Fig. 2 and 3 is used an example (shown on Fig.4) for optimization of packet forwarding in computer networks.

3 Optimization of packet forwarding process in networks with heterogeneous communication channels

Below is briefly described earlier proposed [2] packet forwarding mechanism titled Optimized Load Balancing mechanism that allows minimizing the waiting time of packets in network facilities interconnected with heterogeneous communication channels.

On Fig. 4 is shown the forwarding process of PDUs (Protocol Data Units) packaged as packets $D = \{dk\}$, $k \in (1, 2... \infty)$ which entering at input port of the network facility. The proposed mechanism manages N channels $C = (c_1, c_2... c_N)$ forming virtual channel among source and destination.

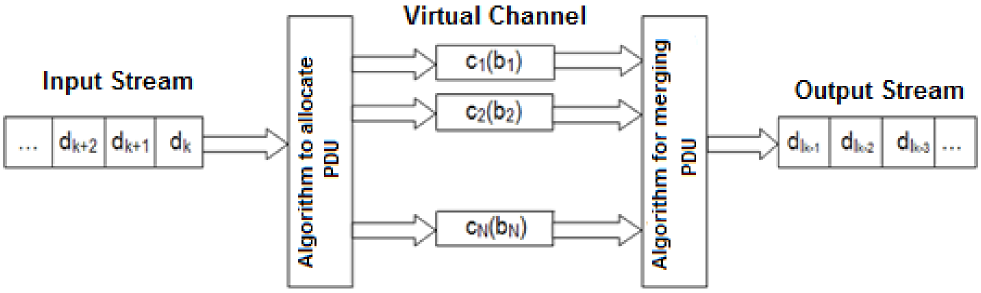


Fig. 4 Transfer of packets (PDUs)

For each channel $c_i \in C$, $i \in \{1,2,.. N\}$ is known its speed (in bps) and is denoted by b_i. The overall speed of the virtual channel is denoted by b. The input packet stream is forwarded to each channel with intensity λ_i, respectively, with probability $p_i=\lambda_\iota/\lambda$, $i \in \{1,2,.. N\}$. Moreover $\sum\lambda_i=\lambda$, $i \in \{1, 2 \dots N\}$, and $\sum p_i=1$.

The aim is to determine probabilities p_i so that waiting time of packets in network facilities to be minimal. This optimization problem might be solved [1] by using proposed calculator of genetic algorithms [5].

In order to determine the mean waiting time of forwarded packets in network facilities with heterogeneous communication channels is used the well-known formula [3] in queuing system theory:

$$(1) \quad T=\sum p_i t_i \ ,$$

where p_i is the probabilities packet to be forwarded to channel $c_i \in C$, $i \in \{1,2,.. N\}$, and t_i is waiting time in the M/M/1 queuing system [3], respectively:

$$(2) \quad t_i = \frac{1}{b_i - \lambda_i}$$

where b_i - the transmission speed of the packets and λ_i - the intensity of arrival of packets in channel c_i, $i \in \{1, 2\dots N\}$ and $b_i>\lambda_i$

The mechanism for traffic forwarding allocates PDUs, Vd_k, $k \in (1,2 ,..., \infty)$ for each channel that is available, $c_i \in C$, with probability p_i, so as to minimize the waiting time of packets:

$$(3) \quad \sum p_i t_i \text{ -> min.}$$

Below is given an example for using proposed calculator of genetic algorithms [5]. Calculations are made with the following input data:

- Number of channels is N=2;
- Speed of the first channel is four times slower than the speed of the second channel (b1=100Mbps and b2=400Mbps);

- Intensity of arrival of packets λ is chosen 50, 150, 250, 350, and 450 Mbps.

Note, intensity of arrival of packets varies between 0.1 and 0.9 of the overall speed of the channels (b=500Mbps), i.e. utilization is 0.1, 0.3, 0.5, 0.7, and 0.9.

User interface for creating and editing the XML model is shown on Fig. 2. The utilization of virtual channel- ro, as well as b1 and b2 are constants, thus they might not be declared as chromosomes. They are defined in fitness declaration field (Fig. 2). If the probabilities of forwarding the packets through the corresponding channels are given by p1 and p2, the intensities will be λ_1= p1.ro.b and λ_2=p2.ro.b respectively.These probabilities are defined as chromosomes, because they vary between 0 and 1 and the optimal result is obtained under an exact combination of them. The genetic algorithm is working by comparing of the best Fitness values of neighboring generations Fitness function is represented by variable $fitnesspro (in PHP syntax) and gives the waiting time (see formula 1):

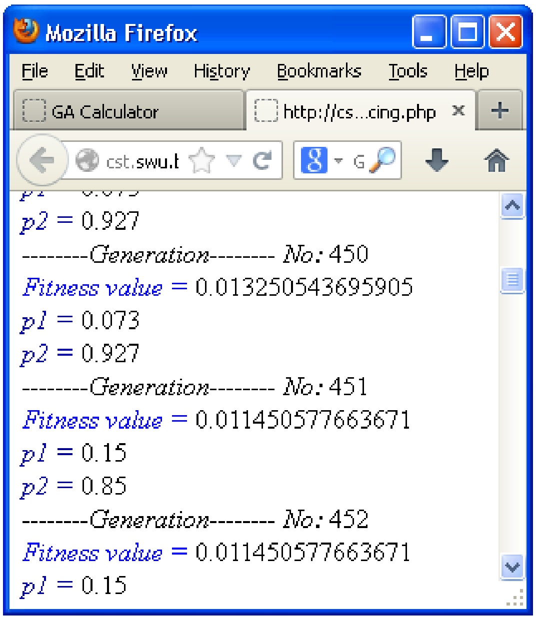
$$(4)\quad \$fitnesspro = \frac{p1}{b1-(p1*ro*b)} + \frac{p2}{b2-(p2*ro*b)}$$

Note fitness function returns estimated waiting time just when the conditions are met: 1) each of the addends in (4) must be nonnegative and 2) the overall probability must be 1 ($\sum p_i=1$), or otherwise returns certain very big value (Fig. 2).

The obtained results for waiting times are shown below.

4 Web Based Calculator of Genetic Algorithms: An Experiment

Experimental system is developed and tested. On Fig. 2 is represented the module for creation and editing of the model of genetic algorithm, on Fig. 3 is given main information about the XML file and on Fig. 5 - a part of generated, after calculations, report. As it can be seen from the Fig. 2 the user can create and edit the model information very easy by using the checkbox for pointing on different fields (rows) and buttons for to delete or insert information. Other buttons, shown on the figure mentioned above, allows saving, reading and calculating the genetic algorithms.



```
p2 = 0.927
--------Generation-------- No: 450
Fitness value = 0.013250543695905
p1 = 0.073
p2 = 0.927
--------Generation-------- No: 451
Fitness value = 0.011450577663671
p1 = 0.15
p2 = 0.85
--------Generation-------- No: 452
Fitness value = 0.011450577663671
p1 = 0.15
```

Fig. 5 A part of calculation report

It can be seen on Fig. 5 how the genetic algorithm is working by comparing of the best Fitness values of neighboring generations. Under every Fitness value there is shown the information about the contents of genes, which values correspond to the Fitness value.

Table 1: Comparison of values for waiting time, derived with method of Lagrange[2] and with genetic algorithms calculator [5]

Utilization	Delay, GA	Delay, by[2]	Error
0,1	0,0028571	0,002857143	4,29E-17
0,3	0,004	0,004	0
0,5	0,0064623	0,0064	6,23E-05
0,7	0,0114285	0,011428571	8,57E-09
0,9	0,0355560	0,035555556	4,99E-07

As it has been already explained the work with the system starts by the preparation of the XML model of the genetic algorithm calculations as well as the chromosomes, their genes and their fields of searching of solution. XML file is stored on the server and PHP1 file produced the script (related to the concrete task) PHP2 by using of special rules. PHP2 is started, calculations are done and report is sent back to the user.

Obtained with the calculator [5] of genetic algorithms results for the mean waiting time- delay are shown in Table 1. They are compared with the numerical results, earlier obtained (by using the method of Lagrange with undetermined coefficients) for same problem (delays for Optimized Load Balancing mechanism) with same input data [2]. Table 1 shows that results obtained by using the calculator of genetic algorithms and the method of Lagrange are almost identical. The maximal error (greatest difference) is observed when the utilization is 0,5 and it is within the range 10^{-5}. This analysis verifies the correctness of the proposed method for modeling of packet forwarding mechanisms by using the calculator of genetic algorithms.

5 Conclusion

The present paper proposes a Web based calculator of genetic algorithms [5] as a remote easily usable and easily integrated tool (it is based on XML). The contents of XML file for modeling of genetic algorithms is defined and represented (it allows extensibility in description of the genes in the chromosomes). Suitable technologies have been chosen for the realization of the modules of the system. Web based calculator with advanced functionality is realized experimented and discussed.

Also, in this paper is represented a packet forwarding mechanism titled Optimized Load Balancing mechanism that allows minimizing the waiting time of packets in network facilities interconnected with heterogeneous communication channels. Working screens of the developed model of this packets forwarding mechanism are represented. The corresponding numerical results are obtained and verified.

6 Open Problem

In this article, Web based Calculator of Genetic Algorithms and Modeling of Packets Forwarding Mechanism have been presented. As a future work I plan to use the calculator [5] for solving of different problems, especially in the fields of:

- Optimization of performance of packet aggregation mechanism: In earlier work [1] I’ve proposed an approach for determining the optimal number of packets which will be aggregated so that their delays to be minimal. Numerical results were obtained for optimization of this aggregation mechanism only with SMTP traffic. The next step will be to use the calculator of genetic algorithms to investigate the influence of the parameters and characteristics of the other types Internet traffic on delays of this aggregation mechanism;
- \- Using Genetic Algorithms for Routing protocol- Routing consists in finding a route from a source to a destination hosts which are in different networks. The routing protocols find a minimum-cost route between a given pair

of nodes. But when the network is very big, then it becomes inefficient since a lot of computations need to be repeated, so it can’t be implemented in the permitted time. The purpose will be to propose a Routing protocol based on genetic algorithms which to solve above problem..

References

[1] Hristov, V., B. Tudjarov, Using Genetic Algorithms to Optimize the Performance of Adaptive Mechanism for Aggregation with Fragment Retransmission, Bulgarian Journal for Engineering Design ISSN 1313-7530, vol.17, 2013, pp. 25-32.
[2] Hristov V., F. Ibrahim, Improvement of forwarding process with multiple network links, Proceedings of the Conference FMNS’2013, Blagoevgrad, Bulgaria, 12 - 16 June 2013, Volume 2, pp. 100-105.
[3] Keshav S., Mathematical Foundations of Computer Networking, Addison-Wesley Professional, 2012.
[4] Popov D. Application of methods of artificial intelligence in design, planning, production and management. Machinery, Technologies, Materials, No I-2007, Sofia, p.23-29.
[5] http://cst.swu.bg/ga/index.html (accessed May 18, 2013)
[6] http://support.microsoft.com (accessed May 18, 2013).
[7] http://www.obitko.com (accessed May 18, 2013).
[8] http://www.php.net (accessed May 18, 2013).
[9] http://www.w3.org/XML (accessed May 18, 2013)
[10]http://www.w3schools.com/html (accessed May 18, 2013)