

Challenges of user Involvement in Extreme Programming projects

Shahriar Mohammadi Bahman Nikkhahan Sahar Sohrabi
*Information Technology Engineering Group, Department of Industrial Engineering,
K. N. Toosi University of Technology, Tehran, Iran*
smohammadi40@yahoo.com Nikkhahan@sina.kntu.ac.ir shr.sohrabi@gmail.com

Abstract

Extreme programming (XP) is one of the most frequently used methodologies in Agile Software Development. According to various reports it had a significant effect on the software projects that use XP. In this paper, one of the main practices of XP, called "on-site customer" is considered, in detail. Also, various important challenges related to its implementation are considered. In each case, according to reports and researches related to user involvement, suitable solutions for resolving the problems are offered. At the end of each case, having an appropriate alternative that uses "Product Management Team" is proposed.

1. Introduction

There are many studies in IS literature about user involvement in software projects [5, 21, 6]. But they did not consider agile methods like XP, so they have no tendency to have a full time on-site customer. Kujala (2007) believes that early user involvement seems to have a positive value for users and customers as such, but user and customer satisfaction can be considered as improving through better system quality [16].

Butler and Fitzgerald (1997) mentioned that user involvement and user participation were used interchangeably in IS literature, even though they have different meanings in other disciplines. They stated that the development activities and behaviors of users and their representatives during systems development process can be group under the name of user participation, whereas user involvement used to reflect the level of importance of and personal relevance of the information system to users [21, 5]. So in XP, user involvement means user participation, because, if we want to run "on-site customer" practice completely, we must have a full time customer, and he or she must participate in software development, but as we will see future in this paper, most of the time we could not have this practice completely running.

Traditional plan-based approaches tend to favor a large up-front investment in requirements analysis and design having low flexibility in customer dialogue in later phases [13]. As a reaction to this, and other characteristics of traditional approaches, agile methodologies suggest a different approach, emphasizing close and continuous cooperation with the customers throughout a larger part of the development cycle [13].

XP is one of the most frequently used methodologies in agile software development; it performs continuous cooperation with the customers through the "on-site customer" practice [20]. XP involves the customer in the development cycle more than many other structured processes [12]. Usually in XP, "customer" means someone from customer organization that has a direct interest in project. He/she might be a direct user of system, a representative from customer organization or a domain expert in developer

organization. Hansson, Dittrich, and Randall (2004) stated that by ‘user’ we mean the one who actually uses the program. When we use the word ‘customer’ we mean those who have money and the mandate to decide what to buy. Often, the user and the customer are not the same person [14].

If “on-site customer” practice ran successfully, it will have a great effect on the project efficiency. This paper reviews many researches about using an on-site customer in XP projects. According to these researches, optimal solutions for some cases are offered, including characteristic of customer, problems of having an on-site customer, customer location and communication with customer.

Hanssen and Fægri (2006) utilized Product Management team (PMT) to solve the problems of implementing the “on-site customer” practice [13]. This paper, according to various cases, shows that how PMT can reduce a lot of problems of customer involvement.

2. Agile methodologies

Software development methods are attempting to offer once again an answer to the eager business community asking for lighter weight along with faster and nimble software development processes. This is especially the case with the rapidly growing and volatile Internet software industry as well as for the emerging mobile application environment. The new agile methods have evoked substantial amount of literature and debates. Miller (2001) gives the following characteristics to agile software processes from the fast delivery point of view, which allow shortening the life-cycle of projects [2]:

- 1- Modularity on development process level
- 2- Interactive with short cycles enabling fast verifications and corrections
- 3- Time-bound with iteration cycles from one to six weeks
- 4- Parsimony in development process removes all unnecessary activities
- 5- Adaptive with possible emergent new risks
- 6- Incremental process approach that allows functioning application building in small steps
- 7- Convergent (and incremental) approach minimizes the risks
- 8- People-oriented, i.e. agile processes favor people over processes and technology
- 9- Collaborative and communicative working style

3. Extreme programming

Extreme programming (XP) is currently the most well known agile method [15]. Like other methodologies, XP has some practices; most of these practices are suitable for being agile in the projects. Some of the practices of XP are planning game, small/short release, metaphor, simple design, refactoring, pair programming, 40-hour week and on-site customer (Table 1). One of the main differences between XP and other methodologies is having an “on-site customer” practice; because, XP emphasizes on full time customer involving in the team.

Table 1. Practices of Extreme Programming [3]

Practice	Enabling factors	Limiting factors
Limiting factors	<p>Divide and conquer (task breakdown, upfront architecture, use cases, etc.) for large and complex projects</p> <p>More formal communication</p> <p>Rotation of skilled developers among teams</p>	<p>Company's organizational issues (mixed and well defined processes, distributed work across multiple teams, communication and coordination problems)</p>
Pair programming	<p>Creation of a collaborative and supportive environment with collocated teams</p> <p>Continuous rotation of developer pairs</p>	<p>Distributed work access across multiple teams</p> <p>Cultural problems (traditional-agile teams)</p>
Test-first	<p>Communication and collaboration (developers-customer)</p>	<p>Lack of automated testing tools</p> <p>Skilled developers-customers</p>
Refactoring	<p>More up front design and design patterns</p>	<p>Lack of refactoring tools</p> <p>Collision with other quality control systems</p>
Simple design	<p>Unit-testing</p>	<p>Lack of theory-guidance</p> <p>Lack of documentation in distributed development</p> <p>Time restrictions in development process</p> <p>The shift of priorities in development process</p>
Code ownership	<p>Team-oriented practices</p>	<p>Volume and type of projects</p> <p>Cultural problems (traditional-agile teams)</p>
Continuous integration	<p>Team-oriented practices</p> <p>Object oriented analysis</p>	<p>Cultural problems (traditional-agile teams)</p> <p>Type of projects (WEB,</p>

On-site customer	Surrogate customer (when needed) More planning approaches (design flexibility) More team practices	internet, etc.) Difficulties at the beginning of the project Type of projects (WEB, internet, etc.) Lack of capable and knowledgeable customers
Short release cycles	More upfront architecture GUI and interfaces architect Divide and conquer (valuable user stories first, valuable task breakdown, etc.)	Large and complex projects Distributed work across multiple teams
40-hour week	Effective organization and strict implementation of both working tasks (iteration time, release time, etc.) and team meeting schedules (fixed time meetings, etc.) Change project's scope or schedule	Company's organizational issues Type Volume of projects
Coding standards	Team-oriented practices Object oriented practices	Company's organizational issues
Metaphor	Enhanced metaphors (high level designs, UML–diagrams, etc.) More architectural design	Unclear practice theory Limited experience

There are many other agile methods like Scrum, DSDM, Crystal methods, Feature driven development and so on [2]. But XP is one of the first full featured agile methods. XP embraces the four major values of agile methods (e.g., early customer involvement, iterative development, self organizing teams, and flexibility) [20].

4. Benefits

User involvement in the projects has a lot of benefits: improved product quality through better understanding of the users needs, improved knowledge of customers' organization, reduced risk of producing unnecessary or unacceptable functionality, improved ability to negotiate expectations among users, improved ability to resolve conflicts regarding the design of the system, increased feeling of ownership among users, reduction in the natural resistance towards change in work practices, remedies lack of decision capability in management, improved project performance and an increased willingness to experiment and improvise in search for solutions [13]. User involvement on system success concluded that sales and user productivity were increased, whereas training cost and user support calls were decreased with an effective user-involvement approach [21, 24].

Having an on-site customer significantly can reduce the number of errors related to business requirements. For example, one research estimated that with using "on-site customer" practice, rework was reduced by over 60% for the project [24].

5. Characteristics of a good customer

For having a successful "on-site customer" practice, customer must have some characteristics. a good on-site customer understands the domain, understands how software can provide business value in the domain, can make decisions about what is needed now and what is needed later, and is willing to accept ultimate responsibility for the success or failure of the project [15].

A customer representative is expected to be available and to be "committed, knowledgeable, collaborative, representative, and empowered" [7].

Customer must have a mutual understanding of schedule constraints in order to deal with system requirements as well as a 40 hour week and clear feature prioritization, in order to have a useful system up and running in a service from day one onwards [23].

In many studies, the authority of on-site customer to make decision about the project is emphasized [15, 9, 19, 16, 18, 8].

A research suggests that the on-site customer should be competent, knowledgeable, and most importantly, credible, or else the benefits to the development process will be limited [12].

Customer characteristics in XP are different from other methodologies. In XP, customer is responsible for the project, like any other members of the project team. It is important that onsite customers understand that they are team members, not team auditors [8]. So XP customer must be someone who enjoys collaborative efforts, and who is prepared to be available to team members to answer questions, to help with problem solving, to be open-minded, honest, objectively critical and respectful [8].

6. Roles and responsibilities of a good customer

Like any members of project team, customer may have many roles and responsibilities, but here six major roles are considered.

6.1. Participating in the processes

In XP, customer must participate in the planning of iterations and releases of the project, like any other members of the project team [23]. Thus, the first and basic role of the customer is acting like a team member.

6.2. Communicating with end users

A good on-site customer, maintains regular contact with end users, understands their wishes, and balances their potentially conflicting interests [23].

6.3. Defining user stories

After communicating with end users and gathering their requirements, customer must define user stories and prioritize them; it is the main task of a customer in the project team.

The customer writes user stories and then discusses each requirement directly with the programmers [18].

If he/she has the ability of making decision about the triple constraints of the project (resources, scope and schedule [6]), the agility of the system can be maintained adequately.

Increasing the scope is a main concern about bringing the customer on-site [23].

6.4. Communicating with developers

An on-site customer must talk to developers, clarify feature requests when needed, and understand some of the developer's technical concerns [23].

6.5. Running tests

After the developing, customer must specify functional tests for user stories, and verify that these tests run correctly [23]. The customer regularly tests the software to confirm it works as expected [18].

Requirements and testing are the key tasks a customer will undertake [19].

6.6. Communicating with management

One of the other roles of an on-site customer is Maintaining good contact with management, explaining progress, and justifying the time spent with the development team [23].

7. Problems of having on-site customers

As mentioned, "On-site customer" practice has many advantages, but it however, has many problems that must be solved. Otherwise this practice may not generate expected results.

Suitable solutions for different problems of having on-site customer are offered here.

Hanssen and Fægri, (2006) had been assigned a Product Management Team (PMT) that was responsible to select an active stakeholder for a project; if it was not possible, one of the

PMT members may act as a representative of the project's stakeholders [13]. With the existence of this team, many problems of having on-site customer can be solved. This paper shows how the PMT can be utilized for solving these problems.

7.1. Partially on-site customer

Despite the importance of the on-site customer requirement, few XP project studies report that they are actually able to fully implement this practice. In many published reports, this requirement is partially implemented by having a knowledgeable engineer or manager role-play as the customer to supplement a part-time or unavailable customer [12, 11]. Current XP research suggests that the "on-site customer" practice, even when partially adopted, provides observable benefits in the final product [12].

Possible solutions:

- Have a part-time customer or to have a technical manager role-play as the customer [12].
- PMT can play the customer role in the team or if it is possible can select best stakeholder as an on-site customer.

7.2. Frequently changing in requirements

When customers participate in development process, the project requirements change frequently. This was partly due to the fact that new people were brought into the project, bringing new requirements. Another reason was that in several cases the people involved in the project, easily changed their minds. XP in itself is capable of implementing non-stable requirements. However, the resulting system may be an incoherent collection of features, ultimately leading to project failure [23].

Possible solutions:

- Construct a customer team in which it is clear by whom each stakeholder is represented [23], if it is not possible, PMT can perform this task.
- Asking for use cases may be a way to make the customer think more detailed [9].
- Find a way to teach the customer the costs of incoherence [23].

7.3. Semantic gap between customer and developer

Technology and potential customers experienced a significant semantic gap when trying to talk to each other. This difference was amplified by the fact that neither the developers nor the customers considered talking to each other as their "real job", easily considering time invested in talking to each other as wasted [23].

Developers had a quite low threshold level to ask questions when the customer was present on-site [15].

Possible solution:

- Explicitly introduce a training process, concerning customer involvement, both for the customers and the developers [23].

7.4. It is hard to convince management

It is very hard to convince management because putting a customer on-site is expensive. Furthermore, management wants to buy a “whole solution” instead of being forced to run a customization project requiring their involvement [22].

Possible solution:

- Have a customer team and choose a representative from the team. Most of the time this is not an applicable solution, but with the existence of PMT, always a team of customers is available.

7.5. Non-appointed customers

Engaging in this kind of strong cooperation with a small selection of customers also means a reduced capability to capture the needs of other, non-appointed customers [13].

Possible solution:

- Management must clearly establish and communicate priorities and assign the resources so that all the work can be completed, and not just for the customer we can “see.” [24].
- One of the major roles of PMT in the team is gathering information in the entire domain, so, members of PMT must be domain experts and can play as the role of a real active stakeholder. Furthermore, in order to expertise of PMT in the domain, it can select a stakeholder, so that he/she could capture the needs of others adequately.

7.6. Time limitation of the customer

Another problem of customer involvement is time limitation of customer for participating in the project. The representative of the customer team or PMT involves in customer discussions and sales meetings so can't collaborate with the engineering team.

In one research, although the customer would have liked to devote 100% of her time to being the on-site customer and working with the programmers, she also needed to work with the end-users and business stakeholders in the project. As a result, approximately 50 % of her time was spent managing technical integration issues [18].

Possible solution:

- In this situation PMT can help. As mentioned before, PMT is responsible for selecting an active stakeholder and if it is not possible, PMT can introduce a representative from itself to perform customer's tasks.

7.7. Varying motivation of customer

As long as the ongoing work affects the customer, the customer stay engaged. As soon as there is no direct interest, the customer subsides. A fundamental effect is that customers have a strongly varying motivation to contribute. Customer loses interest quickly if the perceived value is reduced [13].

Possible solutions:

- Monitoring and managing the collaboration [13].
- Choosing a stakeholder that has a direct interest in the project, and this is PMT's responsibility.

Risky situations, such as schedule slippages and technical difficulties, are more difficult to hide from the customer in XP, but the net benefit may be positive, since it deters institutional delusion about project status and can open communication about how to manage the situations to best meet the customer's business needs. In practice, the net effect of the high transparency might not be significant, but there is a risk of the customer perceiving the daily chaos of the development process and attempting to assert control over it [12].

As a result, with utilizing the Product Management Team, many problems of "on-site customer" practice can be solved, but this team adds a cost to the project and needs to manage properly.

8. Customer location in the team

XP emphasizes on having a full time on-site customer in team; it will lead to speeding up requirement gathering [15], but some researches had problems with implementing this practice completely.

One of the most important reasons for not having an on-site customer is percentage of customer useful activities in the team.

In one research, empirical results revealed that while customer was an average of 83% present with the development team, only 21% of his work effort was required to assist the development team in the development [15].

Another research revealed that the customer was present on-site on average over 80% of the total working time. However, only 21% of his work effort was required to assist the development team in the development [1].

At the end of another research, specified that, 92 percent of the customer time in the team was idle.

Some studies stated that the customer cannot work in one location with the project team. For example they believe that, pair programming is very noisy [12, 15]. These studies showed that not only was the customer very underutilized throughout the project's duration, but the customer's attempts to perform other job-related tasks were disturbed by the noise of the development team and by irregular interruptions [12].

For addressing these problems, different approaches have been used, like moving the customer's work place nearby XP project room. This solution may also support by the developers. However, developers may emphasize that it should take only at the maximum of couple of minutes to contact the customer. Moreover, customer should visit in the project room daily [15]. In another research, the customer did not work in the same room with the development team, but the customer's office situated in the same building with the development team's premises – this was considered to be "sharing enough". But in this project, the developers suggested to involve the customer more in the team [4].

Mary Poppendieck (2004) suggested using on-site developer instead of on-site customer:

- “In my experience, the best software results when the technical team develops a deep understanding of the domain, rather than working through an intermediary. XP calls for ‘customer on site,’ so that developers have access to someone playing the customer role at all times. In my experience we have generally used the practice of ‘developer on site’ instead; the development team works at the site of the real users of the system. They go to lunch with the people who use their software, understand what users do for a living, and felt their pain when the system is balky.” [11]

Members of PMT must actively gather requirements from end users (working in the customer site) and participate in the development process (working in the project development location); these tasks must be done simultaneously and each member of PMT must have a specific role.

9. Communication

Communication is a major concern in all of the software projects. In XP, communication between developer and customer is so critical that the use of an ‘on-site’ customer, physically located with the development team, is recommended [17]. Therefore, here communication and relationship between on-site customer and developers is considered.

Two parallel knowledge transfer processes constitute the platform of software engineering: Firstly, the knowledge of the particulars about the problem domain, the business functions, and the inherent qualities of these functions must be transferred from the customer organization to the supplier organization. Secondly, pragmatic knowledge related to the realistic fulfillment of these criteria must be transferred from the supplier back to the customer (and other stakeholders) [13]. When these two parallel knowledge transfers effectively issue a suitable relationship between customer and developers, “on-site customer” practice may be successful.

It is claimed that problems related to the customer relationship is the most prominent cause in high failure rate for IT projects. Researchers and practitioners seem to agree that increased user engagement is the answer to the problem [13].

Maybe in this relationship, some barriers exist, like semantic gap between Developers and customer and customer’s resistance on changes [22].

Linda Rising (2004) introduces some recommendation for effectiveness of relationship between customers and developers like “It’s a relationship not sale”, “Customer Understanding”, “trust account”, “aware of boundaries”, “Timely response” and so on. [10]

Although communication is a vital process, it will only benefit the knowledge transfer between supplier and customer if it is employed mindfully. Too much, and it will most likely degrade the performance of the developers. Naturally, if there is too little, developers face the risk of developing the wrong software [13].

Gallivan and Keil (2003) proposed a four-stage communication model for effective user-involvement by integrating project team co-ordination and user participation. They have extracted three lessons from their survey. Firstly, they have concluded that using more links, especially more direct links, increased the success of the project. Secondly,

like Gallivan and Keil (2003), Keil and Carmel (1995) also mentioned the reliance of indirect links that intermediates may change the messages or may not understand the customer needs exactly. Therefore, it is important to choose the right intermediate [21].

Brown and Swartz identified three different types of satisfaction gaps:

- The gap between client expectations and client experiences
- The gap between client expectations and provider perception of client expectations
- The gap between client experiences and provider perception of client experiences

With high degree of communication, rapid feedback, and constant adjustments, XP should prevent expectations gaps from becoming unmanageable, but it depends on the quality of the communication between the customer and the development team [12].

Customers perceive, rightly or not, value in certain process activities and artifacts, such as code documentation, architecture, version management, traceability and auditing. To a developer, the value of these activities may only be perceptual, but to the customer, they can be significant. An XP team should be able to adapt the process to customer needs. Interviews and surveys can be performed to understand how customers perceive various process activities, just as more analysis is required to understand how the various practices of XP interact and affect the overall technical characteristics of the software project [12].

10. Conclusion

This paper considers on-site practice of XP (a common methodology in agile software development). It regards the most important challenges in different researches, including characteristics, roles and responsibilities of a good customer, problems of having on-site customers, location of a customer and communication with him.

This research indicates that having a Product Management Team (PMT) can reduce the “on-site customer” practice’s problems effects. Because this team is responsible for selecting an active stakeholder to participate in the project and however, if it is not possible, PMT acts like a customer and a developer. It means that some members of PMT are “customer on-site” and other members are “developer on-site”.

11. References

- [1] P. Abrahamsson , and J. Koskela, “Extreme programming: A survey of empirical data from a controlled case study”, *Proceedings of the Third ACM-IEEE International Symposium on Empirical Software Engineering (ISESE 2004)*, IEEE Computer Society, Redondo Beach, California, USA, 2004, pp. 73-82.
- [2] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, “Agile software development methods: Review and analysis“, VTT Publications, Oulu, Finland, 2002.
- [3] L. Angelis, and I. Stamelos, “Investigating the extreme programming system—An empirical study”, *Empir Software Eng*, Springer Science + Business Media, 2006, pp. 269-301.

- [4] R. J. Back, P. Hirkman, and L. Milovanov, "Evaluating the XP customer model and design by contract", *Proceedings of the 30th Euromicro Conference (EUROMICRO 2004)*, IEEE Computer Society, Rennes, France, 2004, pp. 18-325.
- [5] T. Butler, and, B. Fitzgerald, "A Case Study of User participation in the Information Systems Development Process", *Proceedings of the 18th International Conference on Information Systems*, Atlanta, USA, 1997, pp. 411-426.
- [6] G. Chin, "Agile Project Management: How to Succeed in the Face of Changing Project Requirements", American Management Association (AMACOM), New York, USA, 2004.
- [7] M. Coram, and S. Bohner, "The Impact of Agile Methods on Software Project Management", *Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)*, IEEE Computer Society, 2005, pp. 363-370.
- [8] C. Farrell, R. Narang, S. Kapitan, and H. Webber, "Towards an effective onsite customer practice", *Proceedings of the Third International Conference on Extreme Programming and Agile Processes in Software Engineering*, Alghero, Sardinia, Italy, 2002, pp. 52-55.
- [9] M. Finsterwalder, "Does XP need a professional Customer?", *XP2001 Workshop on customer involvement*, Cagliari, Italy, 2001, available online at: <http://homepages.cwi.nl/~arie/wci2001/finsterwalder.pdf>.
- [10] S. Fraser, A. Martin, R. Biddle, D. Hussman, G. Miller, M. Poppendieck, L. Rising, and M. Striebeck, "The Role of the Customer in Software Development: The XP Customer – Fad or Fashion?", *OOPSLA '04*, ACM Press, Vancouver, British Columbia, Canada, 2004, pp. 148-150.
- [11] S. Fraser, A. Martin, D. Hussman, C. Matts, M. Poppendieck, and L. Rising, "The XP Customer Role", *Proceedings of Extreme Programming and Agile Processes in Software Engineering*, Springer Verlag, Berlin, Germany, 2004, pp. 342-346.
- [12] P.S. Grisham, and D. E. Perry, "Customer relationships and Extreme Programming", *Proceedings of the 2005 Workshop on Human and Social Factors of Software Engineering (HSSE '05)*, ACM Press, St. Louis, Missouri, USA, 2005, pp. 1-6.
- [13] G.K. Hanssen, and T.E. Fægri, "Agile Customer Engagement: a Longitudinal Qualitative Case Study", *ISESE'06*, ACM Press, Rio de Janeiro, Brazil, 2006, pp. 164-173.
- [14] C. Hansson, Y. Dittrich, and D. Randall, "Agile processes enhancing user participation for small providers of off-the-shelf software", *Proceedings of Extreme Programming and Agile Processes in Software Engineering*, Springer Verlag, Berlin, Germany, 2004, pp. 175-183.
- [15] J. Koskela, and P. Abrahamsson, "On-Site Customer in an XP Project: Empirical Results from a Case Study", *EuroSPI 2004*, Trondheim, Norway, 2004, pp. 1-11.
- [16] S. Kujala, "Effective user involvement in product development by improving the analysis of user needs", *Behaviour & Information Technology*, Taylor & Francis, 2007, pp. 1-16.

- [17] L. Layman, L. Williams, D. Damian, and H. Bures, "Essential communication practices for Extreme Programming in a global software development team" Information and Software Technology, Elsevier B.V., 2006, pp. 781-794.
- [18] A. Martin, R. Biddle, and J. Noble, "The XP customer role in practice: Three studies", *Proceedings of the Second Agile Development Conference (ADC 2004)*, IEEE Computer Society, Salt Lake City, Utah, USA, 2004, pp. 42-54.
- [19] A. Martin, J. Noble, and R. Biddle, "Being Jane Malkovich: A Look into the world of an XP Customer", *XP 2003*, Springer, Genova, Italy, 2003.
- [20] D.F. Rico, "Running head: Agile Methods and the Links to Customer Satisfaction and Firm Performance", 2006, available online at: <http://davidfrico.com/rico06a.pdf>.
- [21] Y. Salihoğlu, "User Involvement in Software Project Development: A Review of Models", Software Project Management Research Paper.
- [22] A. Van Deursen, "Customer Involvement Experiences in a Software Product Line". *XP2001 Workshop on customer involvement*, Italy, 2001, available online at: <http://homepages.cwi.nl/~arie/wci2001/vandeursen.pdf>.
- [23] A. Van Deursen, "Customer Involvement in Extreme Programming, XP2001 Workshop Report", *XP2001 Workshop on customer involvement*, ACM Software Eng, Italy, 2001, pp. 70-73.
- [24] M. Williams, J. Packlick, A. Bellubbi, and S. Coburn, "How We Made Onsite Customer Work - An Extreme Success story", *Agile 2007*, IEEE Computer Society, Washington, DC, USA, 2007, pp. 334-338.

Authors

Dr. S. Mohammadi is a former senior lecturer at the University of Derby, UK. He also used to be a Network consultant in the UK for more than fifteen years. He is currently a lecturer in the Khajeh Nasir Toosi University of Technology(KNTU), Iran. His main research interests and lectures are in the fields of Networking, Data Security, Network Security, E-commerce and its Security. He may be reached at s.mohammadi@kntu.ac.ir and smohammadi40@yahoo.com

Bahman Nikkhahan is a master student of IT engineering in the Khajeh Nasir Toosi University of Technology(KNTU), Iran. He received his B.S. in hardware engineering from Iran University of Science and Technology(IUST). He also is an IT project manager in the New Data Age company. His main research interests is software engineering, E-government, E-commerce and network security. He may be reached at bahman616@gmail.com.

Sahar Sohrabi received his B.S. in computer science from Tabriz university, Iran, in 2006. She is studying M.Sc. in Khajeh Nasir Toosi University of Technology(KNTU) now. Her research interests include mathematical modeling, and heuristic and metaheuristic optimization methods.