

TEAM–CLIENT RELATIONSHIPS AND EXTREME PROGRAMMING

John Karn

*Department of Computer Science
University of Sheffield,
UK*

Joseph. J. Ninan

*Department of Computer Science
University of Sheffield,
UK*

Marian Gheorghe

*Department of Computer Science
University of Sheffield,
UK*

Abstract: *This paper describes a study that examined the relationship between software engineering teams who adhered to the extreme programming (XP) methodology and their project clients. The study involved observing teams working on projects for clients who had commissioned a piece of software to be used in the real world. Interviews were conducted during and at the end of the project to get client opinion on how the project had progressed. Of interest to the researchers were opinions on frequency of feedback, how the team captured requirements, whether or not the iterative approach of XP proved to be helpful, and the level of contextual and software engineering knowledge the client had at the start of the project. In theory, fidelity to XP should result in enhanced communication, reduce expectation gaps, and lead to greater client satisfaction. Our results suggest that this depends heavily on the communication skills of the team and of the client, the expectations of the client, and the nature of the project.*

Keywords: *XP, software clients, requirements analysis, communication, satisfaction.*

INTRODUCTION

This paper describes a qualitative study that aimed to observe the relationship between software engineering (SE) teams and the clients who had commissioned the software, how the interaction progressed over the course of the project, and whether the clients were satisfied with the final software product and the team. The teams in this study were supposed to adhere to the extreme programming (XP) methodology (Beck, 2000). Therefore a specific aim was to examine the claim that XP is a people-oriented methodology that should, in theory, lead to greater client satisfaction.

Viewed in the context of SE history, agile methodologies such as XP represent a fairly recent development. XP is the most well-known agile method and is said to be light on documentation,

follows an iterative approach, is incremental, and claims to favor human communication and collaboration over clearly defined stages as mechanisms for developing software (Lycett, Macredie, Patel, & Paul, 2003). There are other agile methodologies in addition to XP, such as Dynamic Systems Development Method (Stapleton, 1995), and Scrum (Janoff & Rising, 2000).

The rationale behind the research described in this paper was the realization that more work was needed that focused specifically on the role of the client in XP projects. Increased client feedback, communication, and involvement in the development process, as stated by XP leads to greater transparency. This had led to concerns that the greater transparency of XP allows clients to have too much insight into how the team is working, which may lead to dissatisfaction on the client side because the client will see problems in software development as they are occurring (Murru, Deias, & Mugheddu, 2003). A recent survey (Ambler, 2007) found that clients were, on the whole, happy with the results of agile teams, and that there was a high success rate for agile projects. In terms of XP specifically, the studies that have been carried out tended to focus primarily on the role of the on-site customer (Koskela & Abrahamsson, 2004; Martin, Noble, & Biddle, 2003; Murru et al., 2003). The results of these studies suggest that the on-site customer is a problematic feature of XP and can be difficult to manage.

Other XP-related research looked at, among other things, the role of novice customers in XP projects and how a team can be lulled into a false sense of security when the customer seems to be very quiet and satisfied early in the project (Elssamadisy & Schalliol, 2002). This sense of security is shattered at a later stage of the project when the team is bombarded with complaints about failing to meet requirements. Another recent publication unequivocally called for more studies that focused on the social issues of the developer-customer relationship during XP development (Grisham & Perry, 2005) and stated that “Customer satisfaction and customer relationships tend to be a sorely unexplored and largely misunderstood aspect of software engineering” (p. 5).

Therefore this work aims to build on these earlier studies by focusing on the relationship between clients and XP teams over the course of seven entire projects. The research aimed to ascertain client opinions on several factors of XP methods, such as the frequency of feedback, whether teams kept them informed, whether they were happy with the communication that took place, the SE and contextual knowledge of themselves and the team, and whether or not they were happy with the overall project, particularly with how they interacted with the team and the final software system they received.

The remainder of the paper is laid out as follows: The next section describes XP in greater detail and the research environment, followed by the methodological procedure, the results, discussion, and then the limitations of the study. Finally there is a conclusion that summarizes the key findings and suggests avenues for further research.

EXTREME PROGRAMMING

XP (Beck, 2000) is an agile methodology centered on 12 core development practices: planning game, small releases, metaphor, simple design, testing, refactoring, pair programming, collective ownership, continuous integration, 40-hour week, on-site customer, and coding standards. More information pertaining to these practices, drawn from Morris, 2001, is given in Table 1.

Table 1. XP Practices (Morris, 2001).

Practice	Description
On-site customer	The customer's job is to write and prioritize stories (tasks from a user's perspective that the software must perform), assist with acceptance testing, and be on hand to answer questions from the development team as they arise.
Metaphor	The project metaphor is, more or less, an informal architecture of the system. The metaphor describes the system in simple concepts. The concepts can be literal or figurative, depending on the clarity of the actual system.
Small Releases	<p>Small releases are a key part of generating feedback and making a project resilient.</p> <p>An XP project is a series of iterations, each lasting 2 to 4 weeks. Each iteration starts with the Planning Game, an activity that determines the tasks for the current iteration, and ends with a "finished" product: All tests pass and the product is as functional as possible.</p>
Planning Game	An iteration begins with the Planning Game, an informal process that sets the agenda for the iteration. The game starts with the customer defining requirements, or the "user stories." Technical members work with the customer to normalize these stories into manageable chunks and break them down into specific tasks, as well as introduce technical tasks needed to support the customer's requests (e.g., upgrading development software, automating builds, etc.).
Pair Programming	All programming on an XP team is done in pairs, two people at one machine. Each task from the Planning Game is owned by an individual. When the day starts, pairs form up, each person either pairing to help someone else, or requesting help on his/her own tasks. Pairs stay together until a logical break comes up. While paired, one takes a turn "driving" while the other actively participates verbally. As ideas flow between the two, the keyboard can be swapped off as often as necessary to get the best code on the screen. Pair assignments are fluid and change throughout the course of a day.
Collective Ownership	Collective ownership refers to the code. Collective ownership allows anyone on the team at any time to work with any piece of code. If a pair working with object A needs object B to change, that pair can go immediately make the change in object B to accommodate the needs of object A.
Testing	Testing is a crucial practice on an XP project. XP succeeds by making a project resilient. Resilience means accurate and frequent feedback; testing provides this. In XP, there are two categories of tests: unit tests and acceptance tests. A unit test is a piece of code that exercises one aspect of a piece of production code. Acceptance tests are distinguished from unit tests in a couple of ways. First, they should test the system end-to-end. Second, the customer is involved in creating the acceptance tests.
Refactoring	Refactoring is the process of improving the design of code without changing the functionality. The code should be clean and readable. Any duplication should be consolidated. Refactorings should be done on an ongoing basis throughout development of the code. As soon as structural improvements make themselves known, they should be done.

Simple Design	To help ensure frequent feedback, it is important that the application's design be kept simple and kept to providing business value. While there will always be tasks that are primarily technical and necessary to support providing business value, these tasks should be kept as simple as possible.
Continuous Integration	Mixing the latest code from each programmer together can be a difficult process, especially if this task is not done often. To stay resilient, newly written code that passes all tests locally must then be integrated with the latest code base by the programs and then ensure all the tests still pass. If not, fixes must be made right away until all tests again pass.
Coding Standard	Having a coding standard for a project is a commonly accepted practice in most projects regardless of methodology. This practice is equally important within an XP team, especially in light of Collective Ownership and Refactoring practices.
40-Hour Week	XP promotes a well-rested team. Its founders do not believe in the sweatshop mentality. Tired workers make mistakes and start desiring a new job.

XP stresses a highly incremental and iterative development process, starting with a simple design that aims to meet an initial set of requirements defined at the start of the first iteration. This design should evolve as the project progresses. XP is aimed at small- to medium-sized teams. The physical environment is also very important, as it should facilitate communication between team members and allow them to coordinate their activities.

THE RESEARCH ENVIRONMENT AND SUBJECTS

The context for this study is the Software Engineering Observatory at the University of Sheffield. The Department of Computer Science's Verification and Testing (VT) research group run this research facility. The observatory was designed specifically to aid those working in the field of empirical software engineering by allowing researchers to observe, question, and interview students taking part in industrial SE projects. Several projects take place within the observatory. These range from a Software Hut project that is taken by second year bachelor's students to the Genesys and Maxi, which are Master of Science (MSc)-level projects.

The research subjects were second- and fourth-year bachelor's and MSc students. The bachelor's students tended to be domestic students and, unless they were mature students returning from industry, lacked any significant SE project experience. The fourth-year students had completed all of their prior higher education at the University of Sheffield and had already completed one group project, thereby having gained invaluable experience. The MSc students tended to be predominantly from overseas (mostly from Asian countries, with particularly large contingents from India and China, and to a lesser extent the Arab world and Greece) and had not completed any of their previous education at the University of Sheffield.

The academics responsible for running the group projects act as managers and meet teams on a weekly basis. The managers have a great deal of experience in managing student projects with real industrial clients and in some cases managing external software projects. The Sheffield students take pride in the fact that they are producing software that will be used by a real-world client. This has led over the years to many satisfied customers, who have been impressed with the students' professionalism and also the final software system produced.

METHODOLOGICAL PROCEDURE

In order to gather data for this study, researchers observed meetings between SE teams and their project clients. We also conducted interviews with representatives of each team, engaged in informal discussions with clients before and after meetings, and conducted formal interviews with clients at the conclusion of the project.

The research was explained to all of the students and clients at the start of a particular project. The students were also reassured that the observations and interviews had no bearing on their final mark and that there would be no instances of students being reported to project management by the researchers. The teams were typically made up of 4 to 6 students. When the students agreed to take part, they were instructed to inform researchers of any client meetings that were to take place, whether on campus or at the client's business premises.

During the client meetings, a researcher would sit in the corner of the room to take field notes. Afterwards the clients would be asked how they felt the project was progressing and clarification was sought if some aspects of the observations were unclear. This continued for the duration of the project, in the case of Software Hut for one semester (4 months), and, in the case of Genesys, for the academic year (8 months).

In terms of interviews, this research adopted a semistructured interview process for the software developers, with the data recorded by hand. This was achieved through a mixture of open-ended questions and specific questions to get not only information perceived to be important by the interviewer, but also unseen information. Before each interview, the interviewee received a brief description of the aims of the research. The goals were made clear and, in particular, the interviewee was informed how this study would help in understanding the XP methodology and the factors that contribute to it, and thus the importance of their answers. The client interviews were more formal in structure, with preplanned questions generated by the research team. This was primarily because of time constraints and the need to gather specific information. When need arose, clarifications were sought on the replies given by the clients. The use of a scribe was employed during these interviews to maintain the even pace of the interview period and to ensure that no important data were missed.

This triangulation of the data through various collection techniques was beneficial because it provided multiple perspectives on an issue, and supplied more information on emerging concepts. The data collection focused on the environment, the culture of the organization in which the developers were working, the history of systems development within the organization (Genesys and Software Hut), managerial expectations and commitment to the development teams, training, and finally individual and team experiences with XP.

The data collection, coding, and analysis proceeded iteratively: The early stages of the research were more open ended, while later stages were directed more toward emerging concepts. This prompted the creation of more structured interview protocols as the project progressed.

RESULTS

This section describes both the positive and negative results found during this research period. There was a wide variety of projects, including an e-commerce system for an outdoor sports business, a text-based system informing customers of offers for tickets to football matches,

special deals for high street stores, and for certain night spots within Sheffield; a Web site for children suffering from cystic fibrosis to help them adhere to a healthy diet; and a system for managing exam results in the Department of Electronic Engineering at the University of Sheffield. For each project, a table illustrates the positive and negative perceptions regarding each team's use of XP and how this influenced the relationship with the project client. For the purposes of anonymity, each team member is referred to by a code (i.e., 4F, 2A, etc., with the number representing a specific team and the letter reflecting the team member).

Project 1

The client for this particular project had a good idea of what he wanted at the outset, and claimed to have a good level of SE knowledge. This assessment was not shared by the team, who strongly disagreed with the notion that the client had a good level of SE knowledge. During the project's lifespan, there were several problems between the team and client, which reached a critical point when the disenchanted client threatened to walk out halfway through the project. Table 2 provides an overview of the project.

Table 2. Overview of Project 1.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
6	A mixture of 4 th year and MSc students. Two were experienced with the XP methodology and very strong technically.	Single client with prior project experience.	Training period for 2 weeks before the start of the project.	New mobile communications 8 months	PHP

"I have been thinking if this is worth the hassle. I may even walk out, as we are not getting anywhere." (Project 1 client)

Although the client claimed to have a very clear understanding of what he wanted, this understanding was not conveyed to the team, who felt the client spoke in vague and unrealistic terms. In addition the client had a blunt way of expressing himself and sometimes wished to take a more hands-on role. The team also experienced problems trying to document the requirements while attempting to adhere to XP. An attempt was made to understand the scope of the project by whiteboarding at the beginning. Additional requirements modeling may have been useful at this stage, in addition to the story-cards. A story-card represents a brief description of a specific user requirement; each implemented function in an XP iteration represents a story-card.

An alleged confidentiality breach also seriously damaged team–client relations. After these events, the team attitudes towards the client went from distrust and apathy to downright hostility and anger. It would be unfair to blame the team alone for this. The client, with his bellicose and heavy-handed attitude, was certainly not without fault, as the quotes below illustrate.

“Someone in this team is a liar. They have disclosed sensitive information about this project to a third party. No one has had the guts to own up. Whoever did it should be purged from the organization.” (Project 1 client, on an alleged confidentiality breach)

“He [the client] is a liar. He has lied about many things throughout the project. Now he is making these unfounded allegations.” (1F, Project 1 team member)

One reason why the client was so intransigent for periods of the project was that he had a preference for detailed documentation to be done upfront, which is at odds with the XP approach. There was much debate about this issue, which carried on into the latter stages of the project. It was suggested by one team member that the team should use a formal document to supplement the XP story-cards, to provide the extra documentation. The team went along with this idea but encountered more problems when the client modified the requirements document. This forced the team to complain that the client’s version of the document had no structure.

“I feel that my requirements document is more comprehensive and does a better job of capturing the essence of the project.” (Project 1 client, on the decision to introduce additional requirements documentation)

Further problems were evident when the client expressed the desire to modify the structure of the database. The team did not trust him to make these changes. However, the client felt it was his right to modify parts of the project and expressed anger when these wishes were not granted. Having meetings and listening to ideas from the client was another source of turmoil in part because of the problems, such as the confidentiality breach, but also due to the perception from certain team members that the client was being unreasonable with his demands and accusations. The fact that this team was following XP created confusion for the client, who was clearly unaccustomed to the agile way of working. In some ways this served to prejudice the team against documentation and modeling. This contributed to the antagonism between team and client. The outcomes of Project 1 are provided in Table 3.

Table 3. Positives, XP Experience, Negatives, and the Outcome for Project 1.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ Story-cards and whiteboard modeling used effectively early in the project. ▪ Regular contact with client. 	<ul style="list-style-type: none"> ▪ Pair programming done consistently only by two members of the team. ▪ Persistent confusion about the exact role of documentation in an XP project. ▪ Test-first not adhered to. ▪ Did not discuss changes with client. 	<ul style="list-style-type: none"> ▪ Voiced a strong opinion that the XP approach was not sufficient to capture the requirements for this particular project. ▪ Urged team to produce a detailed requirements document in addition to story-cards. ▪ Made unilateral decisions to alter requirements documentation. ▪ Expressed dissatisfaction with the process. 	Incomplete system

Project 2

The client for this project, the characteristics of which are provided in Table 4, had a clear idea of what the project objective required at the start. The requirements were clear, but the

client lacked SE knowledge. This led to complaints by the team that the project was too complex and the client was being unrealistic in his goals. In some of the earlier meetings, the client took on a largely passive role and allowed the team members to drive the meeting, only occasionally interjecting with comments about technical aspects of the task at hand. Because the team members all came from a traditional SE background, they found it difficult to adhere to the tenets of XP with its emphasis on fluidity. For example, one team member did not want to make changes to any story-cards, which made it difficult to adapt to changing requirements. The reason for the aforementioned difficulties was because the member in question was used to completing requirements documentation and then moving on. The iterative approach to XP and modification of story-cards was a new way of working. Story-cards are supposed to be updated on a regular basis to represent the requirements from the most recent iteration.

Table 4. Overview of Project 2.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
5	Second-year bachelor's students, but two members had extensive knowledge of different programming languages and of working on tight deadlines.	Single client with prior project experience.	Training period for 1 week before the start of the project. Further help available on demand.	New Web-based database 4 months	PHP, various scripting languages

Personal work preferences made it difficult to do pair programming; the team never got used to this method and worked by running from one machine to the next. Several members expressed a preference for working at home, often during twilight hours. This can be seen from the following quotes:

"This situation is far from ideal. There is too much noise in this lab. How can we get any work done?" (2A, Project 2 team member)

"I prefer to work at home, rather than here." (2B, Project 2 team member)

"I like to stay at home and shut myself off from the world." (2D, Project 2 team member)

One problem that condemned XP from the start in the eyes of this team was the admission that they all enjoyed Discovery (Simons, 1999). Discovery is a documentation centric methodology with defined phases and is similar to the Waterfall method in that it is a linear methodology that goes through the phases of analysis, design, coding and testing. Members of this team found the Discovery module of their studies very interesting, and would have liked to use it in this project. There was some debate about abandoning XP and switching to Discovery, which led to the response from 2B that the team was not really following XP, as can be seen from the quote below. This shows that the team experienced problems remaining faithful to the XP methodology.

“We should stop being so worried about following XP practices. We are essentially following an unholy amalgamation of methodologies.” (2B, Project 2 team member)

None of this helped the team to foster a genial relationship with the client, since the process they were following was not clear, but nor did it lead to a breakdown. The argument from 2B that was expressed during interviews and discussions with researchers went along the lines of the team not doing XP correctly and that they should make it look as if they are using XP in meetings with the project managers. The outcomes for Project 2 are provided in Table 5.

Table 5. Positives, XP Experience, Negatives, and the Outcome for Project 2.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ No conflict with the client. ▪ Regular contact with client. ▪ Professional public image presented during discussions with client and project manager. 	<ul style="list-style-type: none"> ▪ Pair programming not done due to personal working habits. ▪ Test-first seen as counterintuitive. ▪ Admissions that team were not concerned about adhering to XP practices. ▪ Product-driven, as opposed to process-driven, team. 	<ul style="list-style-type: none"> ▪ Failed to understand that project was very ambitious, given the timescale and experience of the developers. ▪ Indifferent to the work produced until the later stages of the project. The client was satisfied with sitting back and letting the team get on with the development work. 	<ul style="list-style-type: none"> ▪ The system was incomplete, a skeleton system was created that provided the basics but lacked all of the core requirements.

Project 3

Some work had already been done by developers from the client’s business prior to the commencement of this project: A Web site already existed in a skeleton form. There were some positives to take out of this XP project, which is introduced in Table 6. The team did present iterations of the software, and the client noted that the working models presented by the team were useful. The level of understanding between the team and the client was improved by intense whiteboard modeling and sketching in the early stages of the project. This also helped with the subsequent production of story-cards relating to specific requirements. The client was happy that the team had a good understanding of the requirements and that there was a working model up and running early in the project.

However, there were also problems to take into account. The client experienced problems conveying the requirements to the team, hence the need for intensive whiteboard modeling. When the client was interviewed at the end of the project, he offered his opinion on why this was the case.

Table 6. Overview of Project 3.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
5	Bachelor's students. Two members were mature students and had prior experience of working as programmers, but lacked any XP experience. Work roles were clearly defined; one member produced documentation.	Single client with prior project experience.	Training period for 1 week before the start of the project. Further help available on demand.	Online database and content management system for a research project 4 months	PHP, CMS, CSS

“The environment made it difficult to communicate: A dedicated meeting room should have been used. There was too much background noise in the main lab and this contributed to the mutual lack of understanding in some of the early meetings.” (Project 3 client)

The client had a very rudimentary knowledge of SE, and felt that the team took too much for granted in this regard. The team had trouble pitching ideas at the client's level of SE understanding.

“They took too much for granted with regards to my technical knowledge. No one asked if I understood certain points. I couldn't get them to pitch things in laymen's terms.” (Project 3 client)

XP stresses regular and informative feedback between the team and client. In this project, the feedback was infrequent and incomplete. This was not entirely the team's fault; the client also canceled several meetings. This client was informed before the start of the project, as are all clients, about the importance of attending meetings and of providing regular feedback. In this particular case, serious problems of a personal nature prevented the client from attending several meetings; therefore the project managers could not apply much pressure. Nonetheless the client had nagging doubts that the team was not showing enough initiative.

“I was available at other times, and they could have called. I was happy for them to have my number.” (Project 3 client)

To their credit, the team was open about what they could and could not do, given the time scale and the level of technical complexity involved in meeting some of the requirements. It took a while for the team to convince the client of the validity of the argument that concentrating on the core requirements was the best way forward. This honesty was appreciated towards the end of the project. Because XP promotes communication and respect, this was a minor drawback. On the positive side there was little conflict between team and client and useful discussions took place. Table 7 presents the outcomes of the project.

Table 7. Positives, XP Experience, Negatives, and the Outcome of Project 3.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ Whiteboard modeling and sketches helped team to gain early understanding of requirements. ▪ Client impressed with iterations. ▪ Honesty was appreciated. 	<ul style="list-style-type: none"> ▪ Heavy emphasis on refactoring and cleaning up code. ▪ Adapted well to the idea of turning out different iterations of the software. ▪ Used story-cards, coupled with whiteboard modeling, to capture the requirements. ▪ Difficult to enforce pair programming and test-first due to working habits of two members of the team. 	<ul style="list-style-type: none"> ▪ Complaints about the noise and environment. ▪ Hinted that the team could have contacted him more often. 	<ul style="list-style-type: none"> ▪ Successful, as core and additional requirements were satisfied.

Project 4

This particular project, indicated in Table 8, differed in that the team did not have access to the main clients who had commissioned the software, but instead communicated with an ambassador who acted as a bridge between the two parties. All of the clients were from the same organization.

Table 8. Overview of Project 4.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
5	A mixture of 4 th year and MSc students. Half the team had XP experience.	Multiple clients; clients' ambassador had prior project experience.	Training period for two weeks before the start of the project.	Maintenance project 8 months	PHP

The ambassador was chosen to represent his organization primarily because he had a higher level of SE knowledge than the primary clients. One positive outcome from this process was that the level of SE knowledge by the clients' representative had increased by the end of the project. During meetings with the team, the ambassador would probe and ask many questions if he did not understand a certain point. This helped to clarify matters for both parties. On a positive note, the team encouraged the ambassador to ask questions. The ambassador reciprocated this attitude and urged the team to ask as many questions as necessary.

The requirements were said to be concrete from the start of the project and the project involved replicating existing systems. For the client, the project was also a learning experience, as the quote below illustrates.

“I have come out of this with a greater understanding of the whole SE process, and how developers go about their tasks.” (Project 4 client ambassador)

The team’s professional approach was also impressive. It was clear that they had prepared well for meetings: No time was wasted and they asked several pertinent questions during each meeting. The client acknowledged this professional approach, and was also impressed by three more things: the team’s high level of technical competence, the genuine team effort that was not dominated by one or two prominent individuals, and the helpful interim releases. The client saw that the various revisions gave a clear picture of how things were progressing. However, not everything was straightforward, particularly a voiced concern about the lack of feedback during certain periods towards the latter stages of the project.

“I had no indication that you were stalling. It is almost as if I have been kept in the dark.” (Project 4 client ambassador, to team after hearing about problems)

“The problems are due to other commitments. This project is still at the forefront of my thinking.” (4D, Project 4 team member)

Despite the openness of both the team and ambassador, problems were still uncovered with the process of gathering the requirements for the software. This highlights the importance of initial requirements envisioning, since this would, in theory, result in the client having a better understanding of the way in which the software will be developed, as well as the project’s needs and possible constraints. The lack of understanding with regards to the constraints led to problems that delayed the final release of the software. Although the project had overrun by a month, in the end, the outcome was positive, as Table 9 and the quote below indicate.

Table 9. Positives, XP Experience, Negatives and the Outcome of Project 4.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ Professional approach from team. ▪ A team effort. ▪ High level of technical competence. ▪ No serious problems caused by not meeting primary clients directly. ▪ Sufficiently knowledgeable ambassador acted as an effective bridge between team and primary clients. ▪ Ambassador suitably satisfied with delivered product. 	<ul style="list-style-type: none"> ▪ Pair programming used throughout by all of the team. ▪ Successful adaptation of test-first. ▪ Refactoring of code. ▪ Effective whiteboarding and use of story-cards. ▪ Effective communication and respect between team members. ▪ Effective XP adaptation. 	<ul style="list-style-type: none"> ▪ Complaints about being kept in the dark for a period. ▪ Criticized the team for not shadowing the existing process. 	<ul style="list-style-type: none"> ▪ The client was happy on the completion of the project. There was recognition that significant progress had been made.

“All in all I am happy with the progress that has been made. The software is not quite there, some tweaks are needed, but significant progress has been made.”
(Project 4 client ambassador)

Project 5

This project, presented in Table 10, got off to a promising start but, by the end of the project, there was an acrimonious split between the team and client. The client felt he was clear with his requirements from the beginning, and that he had provided very detailed and relevant documentation about a specific hierarchy of products. Additionally, the client felt that he was open with regard to meeting and allowing the team to contact him.

Table 10. Overview of Project 5.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
6	MSc and 4 th year members. Two excellent developers. Three members of the team had prior XP experience.	Single client with prior project experience.	Training period for 2 weeks before the start of the project.	New project 8 months	PHP, various scripting languages

“The main requirements were clear, but not the nitty-gritty details. The idea of iterative development is good in theory.” (Project 5 client)

As previously stated, this project got off to a promising start. However, as time passed, the client noted that the team had seriously underestimated the complexity of the project. Worse still was the observation and claim by the client that the team’s body language gave the impression that they did understand what was going on, and they acted as if they understood, but did not. In retrospect, the client felt the team did not probe enough in the early stages, and allowed one or two members to do all of the talking.

“The team did not understand the requirements, and they underestimated the complexity of the project. They gave the impression that they understood, but as the time passed, it was clear that they didn’t.” (Project 5 client)

Additional concern was expressed about the professionalism of the team during meetings. The client said they should have had a clear agenda, a scribe, and a chairperson for each meeting, and that the venue could have been better. The client complained on several occasions about background noise and an uncomfortable environment.

A more serious problem was that the team was forced to start all over halfway through the project. According to the client, this was because they underestimated the initial complexity of the project and they failed to convey their concerns to him. The team argued that the problem was due to the client trying to bring additional requirements in and aiming for increasingly complex features. The implementation of such features would have been very time consuming. This highlighted the need for more requirements envisioning and

scoping when producing story-cards in the early stages of the project. Due to this, there was some degree of slippage, requirements were missed as time passed, and the team claimed this was due to changing requirements. The client disagreed.

“The reason we are not moving forward is because he [the client] keeps trying to change the requirements or add new ones. He is trying to ride roughshod over the contract.” (5B, Project 5 team member)

“You are panicking, and the meltdown has nothing to do with changing requirements. The problem is you underestimated the initial requirements.” (Project 5 client)

This breakdown in communication resulted in irreparable damage to the team–client relationship. The team felt that it would have been wiser to focus on either an eCommerce or stock control system, and that it was unwise to pursue both aims simultaneously during the same project. The client challenged this point.

“It would make more sense to focus on one aspect of the project. Trying to incorporate the stock control system is unrealistic.” (5B, Project 5 team member)

“The discussion has been focused on the Web site. You have neglected the stock control system, which was always part of the project.” (Project 5 client)

Upon completion of the project, the client expressed the opinion that it would have been more useful if the team had produced a detailed requirements and design document. This may have been too extreme and may not have been necessary had the team effectively modeled the requirements early on and paid more attention to the scope of the project. Another point made was that a specialist analyst would have come in useful and there should have been a detailed specification. This is akin to the traditional approach in which systems analysts determine and document detailed user requirements before developers get to work on coding.

Another reason for the client’s anger was that he felt he did not have enough time to explore the look and feel of the system during the project process. Furthermore, he did not get the chance to suggest any changes that could have been made. This was another source of conflict.

“Not having a chance to play with the system and explore the look and feel in enough detail has soured the whole experience for me.” (Project 5 client)

“This is not true. The contract explicitly stated that that any changes after a specific period—30 days before the end of the project—would not be accommodated.” (5B, Project 5 team member)

“I have not been given a fair chance to review the system.” (Project 5 client)

The client was unhappy at the end of the project, expressing his dissatisfaction with the quote below. It also is reflected in the project outcomes, as indicated in Table 11.

“This project has been a bittersweet experience. So much promise and things were going well for a period. I am not happy with the final system: Important functionality is missing.” (Project 5 client)

Table 11. Positives, XP Experience, Negatives, and the Outcome of Project 5.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ Regular meetings with client. ▪ High level of technical competence shown by main spokesman for this team. 	<ul style="list-style-type: none"> ▪ Adapted well to the idea of iterations as specified by XP. ▪ Pair programming was done effectively and pairs were rotated at regular intervals. ▪ Test-first adhered to. ▪ Code was refactored at regular periods. ▪ High degree of respect and communication among team members. ▪ Lack of modeling early on led to misunderstandings later in the project. 	<ul style="list-style-type: none"> ▪ Bitter arguments towards the end of the project. ▪ Felt that not enough scoping took place early in the project. ▪ Over time, the client suspected the team had underestimated the complexity of the project. ▪ The client felt that the contract did not fairly represent his requirements. 	<ul style="list-style-type: none"> ▪ Team and client were in conflict at the end of the project. ▪ Differing opinions over whether or not core requirements were met. ▪ Incomplete system, in the client's opinion.

Project 6

The client for this project, the overview of which is presented in Table 12, had a greater level of SE knowledge than any of the others, and came from a very technical background. The client also believed that the requirements for this project were clear from the beginning. However, the team working on this project had to continue on what had been done by another team the previous year. This took a lot of time, and, for the first few months of the project, the team was seeking to understand what the previous team had done because the code from the previous year was poorly commented, full of bugs, and the project lacked documentation. As a result of these problems, the team was unable to address the client's requirements immediately. The following quotes shed some light on this particular set of problems.

“Do you understand what we are trying to achieve with this system?” (Project 6 client)

“As there is no documentation, I am confused about what the system does.” (6D, Project 6 team member)

Table 12. Overview of Project 6.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
4	One fourth-year student and three MScs. All strong technically, with prior experience.	Single client; prior project experience.	Training for 2 weeks before the start of the project.	Maintenance project 8 months	Java

There was a sense of frustration in the first few months of the project because the client felt there were problems transferring knowledge, and that only one or two people on the team, at most, had any understanding of what was required for the current project. Interestingly the team felt that the client was being unfair by expecting them to start working on the latest version of the system without gaining an adequate understanding of what had preceded it. On the client side, there was a feeling that the team was going round in circles without understanding what to do.

“Well the system should generate reports.” (6D, Project 6 team member)

“They were there last year. As yet you haven’t created anything!” (Project 6 client)

“We have attempted to clean up the UI [user interface] and make it friendlier.” (6D, Project 6 team member)

The knowledge transfer problem was exacerbated by the claim from the client that there was only periodic contact during the early stages of the project. The team disagreed with this perspective by stating that they arranged meetings when they had something to present. Because so much time was spent trying to make sense of the old code, the team argued that there was precious little in the first few months to show the client that he had not already seen. Both sides seemed to be irritated during this period.

“There are a lot of bugs in the first version of the software. The rules were totally wrong.” (6D, Project 6 team member)

“So have you actually written anything?” (Project 6 client)

“No, just reworked existing code.” (6D, Project 6 team member)

“The comments and the code are very poorly structured.” (6D, Project 6 team member)

Things picked up in the second half of the project and the team made good progress towards meeting the client’s requirements. However, there were still problems. Towards the end of the project, the client remarked that he did not think the (new) code was well structured, and voiced doubts about the validity of the XP approach.

“XP seems to involve team members looking over each other’s shoulders. It doesn’t seem to be terribly structured to me. Due to the absence of a detailed requirements specification document, the use of iterations is the only option.” (Project 6 client, on the implementation of an XP process)

A common theme in several projects, as was here, was a desire for documents. This was confusing for the teams involved, since the XP rhetoric motivates teams to provide instead evidence of working functions by way of presenting work done on each iteration. Interestingly, both the team and client remarked that detailed documentation from the previous year would have been helpful in this particular project.

“Are you enjoying the project? I understand that it is difficult to take on someone else’s system.” (Project 6 client)

“I wouldn’t say we were enjoying it. We were not given any documentation.” (6D, Project 6 team member)

One possible way of alleviating this need for documentation in an XP project is by completing an initial envisioning, which will generate high-level models. This would allow stakeholders to communicate their overall desires, to set direction, and to see that the team understands what needs to be built. Two agile practitioners with a wealth of project experience have produced what has been described as a seminal book describing how software developers following agile methodologies should approach modeling and documentation (Ambler & Jeffries, 2002). A relevant argument for this research is that modeling and documentation are important aspects of any software project, including XP projects. Without the high-level models generated during the initial requirements envisioning phase, clients may be tempted ask for detailed documentation without realizing that there is a middle ground.

Ultimately, the client acknowledged that the team had worked hard throughout the project, even though he was not totally satisfied with the final product. Nevertheless, the outcome of the project provided a starting point, a basic database, and a tool for manipulating data. By the end, as the outcomes listed in Table 13 indicate, the client had a realistic view of the project and was not too surprised by the outcome.

“This project was par for the course as far as software failures are concerned; there are no great surprises.” (Project 6 client regarding project outcome)

Table 13. Positives, XP Experience, Negatives, and Outcome of Project 6.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ Very strong team technically. ▪ All members worked hard throughout the project. ▪ Client appreciated their effort and said they conducted themselves in a professional manner. 	<ul style="list-style-type: none"> ▪ Initial difficulties when attempting to understand previous work. ▪ Not enough focus on story-cards and modeling early in the project. ▪ Adapted well to the idea of iterations as specified by XP. ▪ Pair programming was done effectively and pairs were rotated at regular intervals. ▪ Test-first adhered to. ▪ Code was refactored at regular periods. ▪ High degree of respect and communication among team members. 	<ul style="list-style-type: none"> ▪ Frustration expressed due to perceived slow progress and, at times, the lack of meaningful contact. ▪ Awareness that at times the team were not enjoying the project. ▪ Unsatisfied with the final product but conceded it was a good starting point. ▪ Noted that although all members worked hard, not all were forthcoming during meetings. 	<ul style="list-style-type: none"> ▪ Another iteration needed, more tweaks to fix. ▪ Client was realistic, was aware that such problems are likely to occur during SE projects. ▪ Par for the course for projects to overrun.

Project 7

A booklet dealing with personal medical issues concerning children triggered the idea behind this project: The clients of this project, characterizations of which are presented in Table 14, agreed that a Web site could be created based on the booklet. The clients were based in different UK cities and had no knowledge of SE. Overall, the client based in Sheffield expressed a positive opinion about the project, since initially they started with concepts as opposed to firm ideas. He and the other clients found the iterative approach and interim releases to be very helpful, particularly since the scope of the requirements changed from the start of the project and the iterations put the concepts into focus. The interim releases provided a realistic picture of what was possible, as well as allowing the clients to see the progress being made.

“The iterative approach was helpful. As time went by it became clear that more could be done with the software. As a result, it was possible to change the scope of the requirements without creating too many problems.” (Project 7 main client)

Table 14. Overview of Project 7.

Team Size	Team Characteristics	Client Type	XP Mentoring	Project Type and Duration	Main Languages Used
6	Six members in the team, split between 4 th year and MSc students.	Multiple clients; no prior project experience.	Training period for 2 weeks before the start of the project.	New project 8 months	PHP, Flash, scripting languages

Another positive was that the clients were surprised at the professionalism of the team, and how many people turned up for meetings. If a specific time was agreed upon to deliver a part of the project, the team kept their side of the bargain. This was another aspect of the team’s praiseworthy behavior. Finally, the team managed to pitch concepts at a level the clients could understand and refrained from bombarding them with technical jargon.

The Sheffield-based client admitted that he was ignorant of SE and that, in the future, he would put more work into preparing for meetings and learn more about the development project. The lack of SE knowledge was not a major problem for the team, who were happy to explain how to develop ideas in layman’s terms. They also explained if something was infeasible given the technical complexity, time scale, and copyright issues (when using images), and so on.

“Timelines were set, which all parties agreed to. I could understand what the team was trying to convey.” (Project 7 main client)

“It was important to make it clear that there were certain restrictions on having sounds and animations on the Web site. In particular, it would have cost the client thousands of pounds for the bandwidth if large numbers of people started to download videos from his site.” (7C, Project 7 team member)

The team gave no false assurances that things could be done that were not possible. This did not mean that the team was inflexible, since the scope of the project did change. The original idea was to have a site with a lot of animations and games to be used by younger children. This option and the idea behind the project changed into a more educational tool for a broader age range. Honesty proved to be the best policy in this project, and the clients were happy with what was done. This may not have been the case, however, if the clients were too dogmatic or outlandish with unreasonable requirements.

This project is interesting because, on the whole, the clients were satisfied with how things had developed. However, the team was unsatisfied with the frequency of feedback from the client side. What this shows is that clients in an XP project also have a responsibility to maintain contact with their development teams. However, it must be added that, as the outcomes in Table 15 indicate, upon completion of the project, the infrequent contact with the client(s) did not cause as many problems as first feared. The successful outcome suggests that the frequency of contact and feedback was deemed to be sufficient from the client side.

“We need more interaction with all of the clients. The meetings are useful, but more feedback is needed.” (7C, Project 7 team member)

“At times it seems as if the client is not bothered about the project. We need more face-to-face meetings to hammer things out.” (7C, Project 7 team member)

“I am delighted with what has been done, and impressed with the team’s performance.” (Project 7 main client)

Table 15. Positives, XP Experience, Negatives and the Outcome for Project 7.

Positives	Team experience with XP	Client Negatives	Outcome
<ul style="list-style-type: none"> ▪ Not the strongest team technically, but enthusiastic and hard working. ▪ Clients were impressed with the openness and professional approach of the team. ▪ Led well by one very technically gifted member. 	<ul style="list-style-type: none"> ▪ No test-first due to time constraints and the technology employed. ▪ Pair programming only adhered to by one pair. ▪ Iterations worked well. ▪ Effective use of story-cards and whiteboarding to capture requirements before and after initial meetings. 	<ul style="list-style-type: none"> ▪ None expressed by the clients. 	<ul style="list-style-type: none"> ▪ Clients were delighted with what had been done, upon completion of the project. ▪ Positive impression of the team and the methodology used.

SUMMARY OF RESULTS

This research aimed to investigate the nature of the relationship between project clients and SE teams attempting to adhere to the XP methodology. To this end, there are both positives and negatives to take from the results. Table 16 illustrates the positive and negative aspects of the relationship between clients and teams in these observed XP projects.

Table 16. Summary of Results.

General Positives	XP Positives	XP Negatives
<ul style="list-style-type: none"> ▪ Professionalism of teams. ▪ Praise for high level of technical competency. ▪ Most teams pitched ideas and comments at an understandable level for a layman. 	<ul style="list-style-type: none"> ▪ Emphasis on keeping in contact with clients kept communication breakdowns to a minimum. ▪ Pair programming, when used, was beneficial and increased team understanding of the project, which led to greater input from all members during client meetings. ▪ The greater focus on testing meant that the software was relatively bug free. ▪ Several clients commented that they find the idea of iterations to be very helpful since it helped them to see how the system was progressing. ▪ The need for excessive documentation was alleviated by effective whiteboarding and probing done in the early stages in some of the teams. ▪ Story-cards, coupled with the whiteboarding, captured the requirements. 	<ul style="list-style-type: none"> ▪ Story-cards lacked sufficient details. ▪ Insistence on detailed requirements document by some clients. ▪ An “unholy amalgamation of methodologies” at times, as opposed to XP. ▪ Time wasted at the beginning of projects due to lack of scoping. ▪ Confusing feedback from some clients that the teams had not followed requirements. ▪ Test-first seen to be counter-intuitive by several developers. ▪ Glitches unfixed at the end of some of the projects, due to lack of testing within allotted time period. ▪ Lack of prior documentation for maintenance projects caused problems. ▪ Pair programming not fully functional: incompatible pairs and incompatible working hours.

Overall the results show that greater client involvement in a project may well be a double-edged sword. In some cases, a client may feel pressured to exert more authority over the development process and start to take unilateral decisions. There is also a greater risk of open conflict breaking out, especially if a client feels the team is not putting enough effort in or is attempting to throw up a smoke-screen to hide the reality of the situation (whether or not this perception reflects the reality of the team). Ultimately, what this research has shown is that, even with XP and its supposedly greater emphasis on human factors and communication, there are still serious problems with expectations and satisfaction on the part of clients within the produced system.

LIMITATIONS

There are a number of limitations of this study that need to be discussed in light of the results obtained. Firstly, it must be kept in mind that the team subjects were students, not professional practitioners. As always when this is the case, one must be careful about generalizing from

students to practitioners. In defense of our work, however, it must be added that the students were working on projects for real-world clients who actually needed a piece of software, and the students were working in an environment that is as close to an industrial software house as is possible in academia. This means the work carried out for this study is more valid than would have been the case if teams were working on projects defined by academics with researchers playing the role of a client. Another point worth mentioning is that team members were primarily master's students who had previously completed other group projects or, on some occasions, worked in industry before returning to higher education.

Another argument in defense of the use of students for this study relates to the authority of the client. As was previously reported, the clients were diverse in terms of SE project experience. The fact that the clients were working with students may have empowered them to make more decisions and to take on a more active role throughout the project. This could be contrasted with how clients may have conducted themselves in a project in which the development team was made up of seasoned professionals. In such a situation, clients may not be so forthright and direct due to fear of rejection or ridicule. Although we have no evidence that this was the case, it is nonetheless a plausible perspective. A client without any SE experience would be more open and authoritative with students, that is, those still learning the trade of SE, as opposed to seasoned professionals.

Secondly, the work aimed to explore team–client relationships when teams were following XP. It would be unfair to state that all of the teams were all adhering to a purest version of XP as described by Beck (2000).

A further threat is the relatively small sample size of seven teams, although this is an improvement on previous studies that have looked at clients in XP projects. Nonetheless, the sample size and the fact that all projects took place at the University of Sheffield still constitute a threat to the external validity of the study.

The final threat was the Hawthorne effect: Did the presence of researchers have any affect on the behavior of the research subjects, whether they were teams or clients? It cannot be said with 100% certainty that the researchers' presence did not induce socially acceptable behavior. However, the fact that heated discussions and arguments took place when the researchers were present suggests that research subjects were not overly concerned about acting in a socially acceptable manner. Additionally, the often blunt responses from the clients when they were interviewed and the fact that they had commissioned a piece of real-world software supports the notion that they had nothing to gain by behaving in a manner they thought would be more appropriate for the researchers benefit.

CONCLUSIONS AND FUTURE WORK

As stated earlier, XP is a popular agile development method that explicitly defines a role for a client working in close conjunction with the development team (Beck, 2000). So how does this research build on the existing literature? It adds to the knowledge base concerning human and social factors of XP by focusing specifically on how teams interact with clients and sheds light on an area that has been overlooked by other XP researchers in the main. This work is important because XP attempts to increase and improve client–development team communication, both in quality and quantity over traditional methods. In theory, this should

prevent serious breaches between teams and clients, as expectation gaps should not become too large. This theory needed to be tested in realistic situations.

However, this and other related research have shown that success in XP projects depends, to a large extent, on the quality of the communication between teams and clients. In reality it may also depend on how dogmatic the teams, clients, and managers are with regards to XP practices, and whether they are willing to tailor other practices to scale XP to meet the needs of the project teams. This particular study has attempted to fill the knowledge gaps on how clients communicate with teams and how they convey their requirements, the effect of their expectations, and if and when they change. Carrying out more work in this area will provide a greater insight into client needs in XP projects, and whether or not teams following XP actually meet those needs.

This work provided more evidence to suggest that the effective management of client needs and expectations is of crucial importance regardless of the methodology being followed. However, this takes on greater importance with XP, due to its transparency, emphasis on fluidity and rapid feedback, and the explicit call for the client to play an active role in the development process. While placing greater pressure on the team, XP also calls for the client to devote more time to the project.

More work is needed in this area to confirm the findings of this and earlier papers. Future studies of XP should aim to devote more attention to the role of the client. This may involve expanding the scope of current XP studies or carrying out research specific to the role of the client. In addition, future work should also focus on how the team found the experience of working with a specific client. This could involve conducting interviews with software developers and managers.

This line of research is important for SE as a whole, particularly since we are now living in an age of competing methodologies. Agile methodologies suggest that greater client involvement should lead to fewer expectation gaps and greater satisfaction; more work is needed to confirm whether or not this is actually the case. We hope that the work described in this paper will stimulate other researchers to look into issues such as client satisfaction and expectation gaps during XP development projects. Finally, once an XP knowledge base has been established, future work need not focus purely on XP but may incorporate other agile methodologies.

REFERENCES

- Ambler, S. (2007). Survey says...agile has crossed the chasm: Examining the effectiveness of agile practices. Retrieved on June 11, 2008, from <http://www.ddj.com/architect/200001986?pgno=1>
- Ambler, S., & Jeffries, R. (2002). *Agile modeling: Effective practices for extreme programming and the unified process*. Hoboken, NJ, USA: Wiley.
- Beck, K. (2000). *Extreme programming explained: Embrace change*. Reading, MA, USA: Addison Wesley, Longman.
- Elssamadisy, A., & Schalliol, G. (2002). Recognizing and responding to “bad smells” in extreme programming. In *Proceedings of the 24th Conference on Software Engineering* (pp. 617–622). Heidelberg, Germany: Springer.
- Grisham, P. S., & Perry, D. E. (2005). Customer relationships and extreme programming. In *Proceedings of the 2005 Workshop on Human and Social Factors of Software Engineering conference* (pp. 1–6). Heidelberg, Germany: Springer.

- Janoff, N., & Rising, L. (2000). The Scrum software development process for small teams. *IEEE Software*, 17, 26–32.
- Koskela, J., & Abrahamsson, P. (2004). On-site customer in an XP project: Empirical results from a case study. In *Proceedings 11th Conference on Software Process Improvements* (Euro SPI 2004; pp. 1–11). Heidelberg, Germany: Springer.
- Lycett, M., Macredie, R. D., Patel, C., & Paul, R. J. (2003). Migrating agile methods to standardized development practice. *IEEE Computer*, 36, 69–93.
- Martin, A., Noble, J., & Biddle, R. (2003). Being Jane Malkovich: A look into the world of an XP customer. In *Proceedings of the 4th International Conference on eXtreme Programming and Agile Processes in Software Engineering* (XP 2003; pp. 234–243). Heidelberg, Germany: Springer
- Morris, C. (2001). The case for XP. Retrieved July 20, 2008, from <http://clabs.org/caseforxp.htm>
- Murru, O., Deias, R., & Mugheddu, G. (2003). Assessing XP at an European Internet company. *IEEE Software*, 20, 37–43.
- Simons, A. J. H. (1999). The discovery method for object-oriented software engineering. Retrieved May 4, 2007, from <http://www.dcs.shef.ac.uk/~ajhs/discovery/>
- Stapleton, J. (1995). *Dynamic systems development method*. Boston: Addison Wesley.

Author's Note

All correspondence should be addressed to:

John Karn
Department of Computer Science
University of Sheffield
Sheffield, S1 4DP
UK
J.Karn@sheffield.ac.uk

Human Technology: An Interdisciplinary Journal on Humans in ICT Environments
ISSN 1795-6889
www.humantechnology.jyu.fi