

Analyses of an agile methodology implementation

Sylvia Ilieva
Sofia University
Sylvia@acad.bg

Penko Ivanov
Rila Solutions
Penko.Ivanov@rila.com

Eliza Stefanova
Sofia University
eliza@fmi.uni-sofia.bg

Abstract

In the last few years Agile methodologies appeared as a reaction to traditional ways of developing software and acknowledge the need for an alternative to documentation driven, heavyweight software development processes. This paper shortly presents an agile approach for software development of e-business applications. The approach, named eXPERT, is applicable to small teams developing projects characterised by often changing requirements, tight schedules, and high quality demands. The present article describes a case study about eXPERT approach implementation at software developing company. Experiment results based on preliminary defined series of metrics are presented and analyzed.

1. Introduction

In the last few years Agile methodologies appeared as a reaction to traditional ways of developing software and acknowledge the need for an alternative to documentation driven, heavyweight software development processes. Of all the lightweight methodologies, eXtreme Programming (XP) tends to be best accepted by the e-project developers. E-project as defined in [3] is a project, which must be delivered rapidly; is mission critical; and has to be managed in a turbulent business and technology environment. Recent studies show that productivity and software quality increase by applying XP principles. But even projects that have adopted several or all XP practices meet some problems, related mainly to project management, estimation and planning. Bearing this in mind, a new approach was developed based on eXtreme Programming (XP) [1] and the Personal Software Process (PSP) [4].

This approach is applicable to small teams developing projects characterised by often changing requirements, tight schedules, and high quality demands. Its aim is to facilitate all related activities with crucial importance to project success: pure

development, management of time, changes, quality, customer relationships, and professional growth of the employees. The approach is called eXPERT [2] and combines principles of XP and the PSP.

eXPERT approach has the following clearly defined ambitious business goals:

- to increase productivity by approximately 20%.
- to reduce defect rates by 30%.
- to reduce project overruns by about 15%.

Seven pilot projects were run at different SMEs experimenting the application of the defined approach to e-project development. The present case study describes one of those pilot projects developed by the Bulgarian company Rila Solutions.

The article is structured as follows: Section 2 presents eXPERT approach. Section 3 presents the experiment at Rila Solutions. In section 4 series of metrics that had to be collected during experiments are defined. In Section 5 experiment results based on defined metrics are given. Section 6 concludes the paper.

2. eXPERT approach

Although the XP practices seem to be very simple, they require strong individual and team discipline in order to achieve good results. There are a number of reports about XP experiments showing that these experiments have failed due to developers' reluctance or incapability to apply the practices in a disciplined and professional manner. Incapability is related mainly to wrong estimations of individual work and failing to create a correct plan of the tasks that have to be performed. It seems that complementing XP with the PSP is a good way for resolving the problem. Even more, the PSP could also contribute to coping with the problem of reluctance. That is why the eXPERT approach is built on two well-known software development approaches XP and PSP. Table 1 enumerated the practices eXPERT approach is based on.

It is known that the agile methodologies speak about practices. Nevertheless eXPERT approach was described in terms of processes in order to facilitate its

adoption by companies, who have already implemented or are interested in CMM(I) or ISO certificates. eXPERT approach maintains its description comprehensible, motivating and easy to apply, while, at the same time, providing means for good project estimation, planning and management

Table 1. Practices from XP and PSP.

Practices from XP	Practices from PSP
Planning Game	Defect Logs
Simple Design	Time Logs
Metaphor	Effort logs
Coding Standards	Coding Standards
Test before code	
Collective Code Ownership	
Pair Programming	
Continuous Integration	
Refactoring	
Small releases	
Open Workspace	
Customer on-site	
40 hours / week	

The eXPERT approach [2] is described through five processes: Customer Requirements Management, Project management, Design, Code and Test (fig. 1).

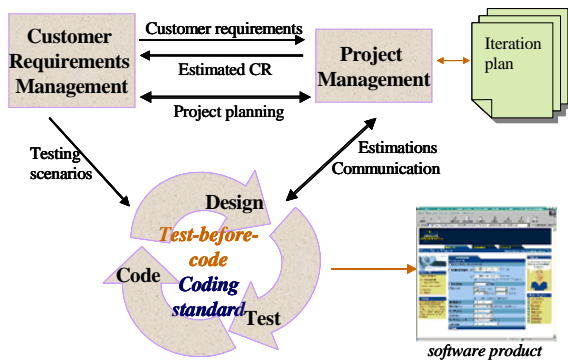


Figure 1. eXPERT Processes

Every process is described in following format:

- Overview (objectives)
- Inputs
- Activities
- Process outputs
- Completion criteria
- Measurements

Activity n:	<Activity description>	
Task n.m:	<Task description>	Responsible role

Figure 2. Activity format

Figure 2 presents the common format for Activity description: *Tasks* that have to be done to complete an activity and the responsible *Role* for performing a task.

Let us take as example the Testing process. Its *objective* is to validate that software product meets the requirements and satisfies the acceptance criteria defined by the customer. As *input* Test process takes Documented Customer Requirements with design elements and Coding standards and as *output* produces Code with fewer defects. Following *activities* describe the Testing process:

- A1. Prepare for testing.
- A2. Describe and implement acceptance testing.
- A3. Perform unit testing in parallel with coding.
- A4. Perform regression testing when integrating.
- A5. Perform acceptance testing after integration, especially before delivery.
- A6. Measure the process (effort, defects)

Completion criteria for Testing process: when defined test cases cover all typical, boundary, and specific cases described by the customer requirements. Identified *Measurements* are: Defects rate, Effort spent on acceptance testing, Defect removal efficiency. For the lack of space we are not going to further details in Testing process description. More information about all defined processes can be found in [2].

The activities described into the eXPERT approach [4] are based on the enumerated XP practices. Certain modifications are introduced mainly in relation to measuring the effectiveness of these activities and the defect rates. This measuring is needed to identify problem causes and to eliminate them in the future. The logs for collecting project data follow the templates and the principles of PSP, but are modified to fit the XP method, and in particular to reflect its specifics, namely that developers work in pairs and that design, testing and coding processes are strongly interrelated and executed in parallel. eXPERT approach defines the following roles: Customer, Project Leader and Developer. They are very close to the roles as defined in XP (Programmer, Customer, Coach, Tracker, Tester, Consultant, Big Boss), with some additional responsibilities coming from the application of the PSP practices. More than one role can be assigned to a single team member into a project. In such case it is important that this team member has the necessary knowledge and skills as well as the time for performing all the roles assigned to him.

The eXPERT approach, if compared to XP, seems to be more structured, easier to understand and to apply by SME's because of the process oriented architecture. At the same time it is not as light as XP (especially for developers) because of the requirement to measure activities and tasks to be done - each of the processes contains one special activity, whose goal is to track effort and time spent on tasks in the process. The idea for tracking came from PSP, but in all other aspects the eXPERT approach and PSP seem to be incomparable.

3. Rila Solutions experiment

3.1. Company description

Rila Solutions is a software development company that is focused mainly on the development of client projects with tight schedule and limited budget. It is common for the company to develop software for clients with no special knowledge in the technology of the IT solutions that Rila provides and even clients that have no IT background. Such projects rely entirely on the vision of the customer of the fully functional system and its features but not the means of achieving desired functionality.

For these reasons Rila Solutions always tries to be on the edge of the modern technologies and in search of agile methodologies to deal with the new requirements in the software development practices.

eXPERT as an approach for flexible and tightly client involvement appeared as a adequate approach for developing such projects. Therefore, eXPERT was presented to the top management of the company as an approach that gives the desired flexibility for completing project on time on schedule on budget. The company management agreed to start a pilot project using the eXPERT and requested comparison results for its effectiveness.

A challenge for the adoption of the eXPERT approach was to the current ISO 9001 certification of the company and steps and changes that should be performed prior to using and adopting any new agile methodology within the company. Each company innovation to the current certified process is a subject of thorough research and modifies the current status quo for the whole project lifecycle.

3.2. Experiment description

Applying eXPERT in Rila Solutions passed several major steps:

- Step 1. Analyse the gap between the current project lifecycle that is already certified;
- Step 2. Implement the new approach within the framework of the current software development company standards;
- Step 3. Use the eXPERT in a real life project according to its practices;
- Step 4. Provide clear and comparable results to the company management about the effect of applying eXPERT.

The first step of applying eXPERT was a typical business reengineering process that did not involve any programming effort. In addition, for the purposes of correct introduction of eXPERT to the company staff, different description documents, standards and a

company developed questionnaire were used as additional sources.

The result of the first step was a “Gap Analysis” document that outlined the differences between the currently certified company process and the requirements of the eXPERT approach. This document was used as a basic document for the second “implementation” step that involved the following sub steps:

- Update the current process and all supporting document to reflect the changes for applying the eXPERT;
- Create a so called “Tailoring Guide” for the company employees and most importantly for the members involved in the pilot project team that describes in detail the adoption of the eXPERT within the company and the changes to the company process;
- Carry out a short course on the principles and practices of the eXPERT and its implementation within the company for the pilot project team members in order to prepare them for the new framework and discipline required by the approach.

As a result of the “implementation” step the new updated company process adopting the eXPERT methodology was achieved. After setting up all the prerequisites for successful implementation of the new methodology, the pilot project that utilizes it was ready to start.

One of the crucial factors for evaluating the usefulness of the new methodology was the adequate selection of a baseline and pilot project for comparing the achieved results and reporting if the enterprise has achieved the desired benefits and what is the actual profit. There was a real need of adequate comparable results from these two projects

Table 2. Measurable criteria for baseline and pilot project selection

	Baseline	Pilot
Technology	J2EE	J2EE
Team Size	4	4
Effort	908 m/h	900 (estimated)
No of requested client functionalities	8	11

Rila Solutions as a previously ISO 9001 certified company has its own developed system for time tracking called IVAN where all previous and current company projects with their total and task effort and durations are listed. Due to differences in size and technology and the limited number of currently starting projects two quite similar projects were selected. Table 2 presents the measurable criteria for the baseline and

the estimated pilot project that were taken into account for selecting the most similar couple of projects. The similarities on these factors gave the chance for accurate comparison of the impact of eXPERT on the development lifecycle.

Several additional criteria like team experience, number of database tables, number of user forms and controls were also used for the correct selection of the two projects.

Baseline Project

Rila has developed a product called Investor Management System that is used as a framework for developing financial management systems for transferring of knowledge for investment decisions, entering different financial data and subsequent monitoring. It operates entirely over the Web, without any additional requirements concerning equipment or software. The technology that stands behind that system is J2EE with Oracle as database and application servers.

A personalized implementation of this system with specific functions and modules according to the client business needs was selected as a baseline project.

Pilot Project

The selected pilot project was a personalization of the product for another client and required similar amount of new functionalities and modifications as the baseline project.

The team that participated in the development of the baseline project also participated in the pilot project thus ensuring as far as possible the same team competence on the technology and the system metaphor. Moreover, the same team had experience gained during the baseline project that was used in the pilot project, but as there is no ideal situation this team selection was the best compromise for correct comparing.

The pilot project was organized in one release that includes three iterations. The user stories that were collected for each iteration included the client vision of the workflow and the use cases for the system.

4. eXPERT metrics

Within eXPERT project a series of metrics are established which had to be collected during experiment [6]. Those metrics are helpful to make conclusions about reaching defined project goals as well as to answer the questions about the approach acceptance. Defined and used metrics are shortly presented below. The tools used for collecting the metrics and the possibilities for the automation of this process are outside the scope of this article.

4.1. Productivity

The productivity is defined in two ways (Table 3):

- size in KLOC (lines of code) of the developed code divided by effort of the team, pair or single programmer;
- velocity over by effort of the team, pair or programmer. Velocity is the sum of the time estimates of user stories implemented within an iteration/release.

Table 3. Productivity metric

Calculation	Size/ Effort		Velocity /Effort	
Unit	Work/Time			
Measured	Size [KLOC]	Effort [Man month]	Velocity	Effort [Man month]
Possible degree of detail	Measure Productivity <ul style="list-style-type: none"> • for the team • for each pair • for each programmer 			

The company has the freedom to select a way to measure productivity, but the chosen manner has to be applied along the whole project. Also it can decide to measure productivity per iteration or per release.

4.2. Defect rate

Two types of defect rates metric are defined (Table 4):

- the number of defects made by a team and by each programmer during each iteration and/or release and during the entire project;
- the percentage of the effort spent for bug fixing vis-à-vis the effort spent on the entire project.

Table 4. Defect rate metric

Calculation	Number of defects	(Effort spent for bug fixing / Effort) x 100	
Unit	Number	%	
Measured	Number of defects	Effort spent for bug fixing [Man month]	Effort [Man month]

The possible degrees of detail for both type of calculation are:

- both for the team and for each programmer;
- for the whole project and for each release and/or iteration.

For first calculation only: measurements could be made for different types of defects.

Each company has to determine the type of defects they are going to record. Second type of measuring defect rates was optional.

4.3. Relative Schedule Deviation

The relative schedule deviation presents how the real time spent on development corresponds to planned time. That deviation could be measured for each user story, iteration, release or for the whole project. The metric is presented in more structured way in Table 5.

Table 5. Relative Schedule Deviation metric

Calculation	((Real time - Planned time) / Planned time) x 100	
Unit	%	
Measured	Real time [months]	Planned time [months]
Possible degree of detail	Measure the times planned and real : <ul style="list-style-type: none"> • For the whole project • For a release • For an iteration • For each user story/feature 	

A positive value means that the deviation is against the planning. A negative value means that the deviation is on project's favour.

4.4. Relative Cost Deviation

The relative cost deviation metric is similar to the Relative Schedule Deviation metric but from cost prospective. Detailed presentation of the metric is given in the Table 6.

Table 6. Relative Cost Deviation metric

Calculation	((Real costs - Planned costs) / Planned costs) x 100	
Unit	%	
Measured	Real costs [K €]	Planned costs [K €]
Possible degree of detail	Measure the costs planned and real : <ul style="list-style-type: none"> • For the whole project • For a release • For an iteration • For each user story/ feature/ 	

The company decides the parameters to include in calculating the cost, e.g. labour, equipment, etc.

As for previous metric the same understanding is valid for positive and negative values.

4.5. Project Cost Change

The project cost metric presents how the cost of the

pilot project corresponds to the cost of the baseline project. The definition of the metric is presented in the Table 7.

Table 7. Project Cost Change metric

Calculation	(1-Cost _{pp} /Cost _{bp}) x 100	
Unit	%	
Measured	Cost _{pp} [K €]	Cost _{bp} [K €]
Possible degree of detail	Measure Relative project cost change <ul style="list-style-type: none"> • For the overall project • For selected modules 	

Cost_{pp} is the cost of the pilot project or a selected module from the pilot project. Cost_{bp} is the cost of the baseline project or a selected its module.

The company decides what parameters to include in calculating project cost, e.g. labour, equipment, etc.

In cases where the modules/projects are not absolutely convenient, it is possible to define inappropriate weighting factor.

4.6. Customer and developer satisfaction

Both customer and developer satisfaction metrics are defined by questionnaires. Those metrics (Table 8) are not directly related to the project goals but they show how the new approach is accepted by the customer and developers and how it affects the quality of the product from customer point of view.

Table 8. Customer/developer Satisfaction metric

Calculation	Customer/developer satisfaction rated between 0-100%
Unit	%
Measured	Questionnaire
Possible degree of detail	Measure the customer/developer satisfaction with the whole project and/or during the project with each release, iteration etc., depending of the company

5. Rila Solutions experiment results

Starting the pilot project required all the team members to accept and use the practices required by the eXPERT. All the practices were sorted in three groups: already adopted, not modified and modified practices.

As displayed on Table 9 three of the practices from the XP were slightly modified. Customer on-site practice is the leading practice and the most significant risk factor for applying the eXPERT and therefore requires to be applied as much as possible. In Rila

Solutions experiment the customer was not on site but had a constant web access to the so called “staging” environment, where the latest version of the system was running and a constant communication with the client was supported. In this way the customer had a permanent view on the project status as a whole. In these terms the company achieved a customer “on-line” practice that worked quite well for the project.

Table 9. eXPERT Practices in Rila Solutions

Origin	Already adopted practices
XP, PSP	Coding Standards
XP	Continuous Integration
XP	Open Workspace
	Not Modified Practices
XP	40 Hours Week
XP	Paired Programming
XP	Small Releases
XP	Release and Iteration Planning (Planning Game))
XP	System Metaphor
XP	Simple Design
XP	Collective Code Ownership
PSP	Defect Logs
PSP	Time Logs
PSP	Effort Logs
	Modified Practices
XP	Customer on-site
XP	Refactoring
XP	Unit testing

The process of refactoring requires code reviews and changes to the source code of the system. Such procedures are time consuming and therefore refactoring is required on certain places. Such places are the system bottlenecks in the business logic of the system.

The test before code practice has been applied for the business logic only but not for the acceptance tests and the presentation part of the application (htmls and jsps). The presentation part of the system is the point of constant customer reviews and changes. For that reason the test before code practice has not been applied for those parts of the system. The acceptance tests were the ones to verify and validate the user interface part.

As an ISO 9000 certified company Rila Solutions has been previously aware of the importance of collecting historical data and using tracking and reporting systems. In that way systems for time and effort tracking and bug tracking were used prior the implementation of the PSP practices required from the eXPERT so that the major metrics for comparing the results were available for the baseline project. Nevertheless some modifications to these systems were

introduced for the PSP practices and especially for the strict reporting required.

5.1. Productivity

On Fig. 3 the productivity calculated by iteration, according to 4.1, is presented. It is measured using the metric KLOC/Effort. The effort is measured in man hours and is gathered for each iteration of the “pilot project” as required by the PSP. The lines of code created are also extracted only for the iteration measured. The last iteration is with less productivity factor due to the fact that client requested minor modifications and bug fixing. This required more effort but less KLOC was produced.

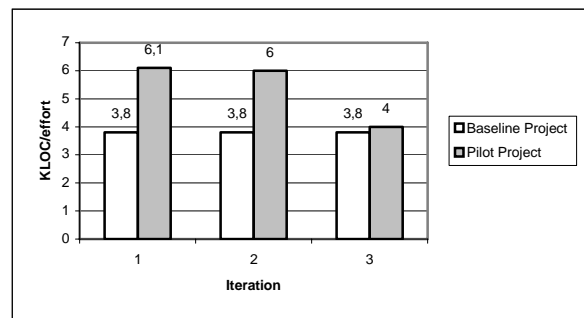


Figure 3. Productivity by iteration

The differences in the productivity between the baseline and the pilot project are also a result of the customer requirements that involved implementation of several new functionalities in the system and quite a few source code was reused. This led to higher amount of code produced as KLOC.

5.2. Defect Rate

The metric used for measuring the defect rate is the number of defects for iteration. To the extent that statistics for defects by iteration were not available for the baseline project but only for the pilot project, comparison by iteration is not available. The measurement is presented in total number of defects (Table 10). This number corresponds to the defects reported by the client or by the quality assurance team and does not include compilation errors produced during development or developers unit tests.

Table 10. Defects rate

	Baseline	Pilot
Total Number of Defects	60	52

5.3. Relative Schedule Deviation

Rila Solutions developers are not attached to work in only one project or client and the project has to be completed within a certain time-frame determined by the project schedule. The pilot project was with fixed duration and the project schedule was presented to the customer. If developers finished the work planned for a certain day with less effort spent on the pilot project (within shorter time) that was not reported as schedule deviation (reduction in time). Developers were instead given the opportunity to work on other projects (which is a normal practice in the company and is based on management business decision). Company management decision was not to make changes in the schedule because the schedule was included in the client's contract and reducing the time for the project would not be cost-effective for the company. Especially in case of using eXPERT, that decision had additional logic because when team members work on several projects at the same time, applying pair programming requires additional effort in coordination of pair members.

For the above reasons the reduction in the schedule is not included in the presented metrics and is not calculated as schedule deviation.

5.4. Relative Cost Deviation

At the very start of the pilot project the estimated work for the project was 900 m/h. The total effort reported on the project was 804 m/h.

For the whole project the cost deviation (CD) was:

$$CD = (900-804) \times CHR = 96 \times CHR,$$

where **CHR** stands for **Company Hour Rate**

5.5. Project Cost Change

Project Cost Change is actually the most valuable metric for the company management. The presented result is not based on real cost deviation but on effort deviation (Fig. 4). Actually the cost deviation can be easily calculated for each company knowing its hourly cost rates. The total cost of a company project of course includes other expenses not immediately related to the effort. However, as a software development company Rila Solutions calculates its project costs based on the effort spend.

Reduction of effort spent on a project leads to real money savings for the company as a whole and provides the flexibility of sharing resources among projects. Overall company costs are reduced and higher profit for the whole organization is achieved.

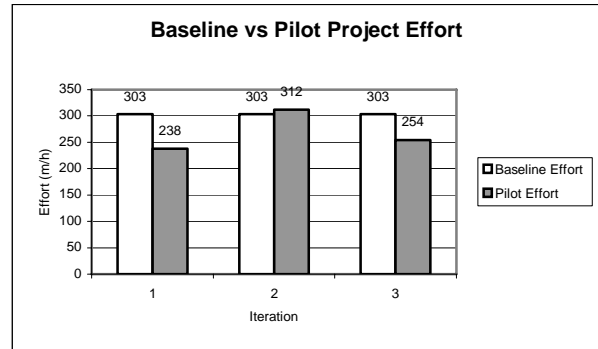


Figure 4. Baseline vs pilot project effort

The most significant on this measure is the less effort spent on the first iteration that accordingly leads to a higher productivity for iteration one (Fig 3). As this was the start of the project fewer changes were required by the client and more new source code were developed. The increased effort on the second iteration also results in very high productivity but effort is also required for bug fixing and client changes. The effort spend on the third iteration is mainly for amendments according to the client requirements and that is why a rate of productivity is measured with almost the same effort spend. The effort spent for the project as a whole is less than the estimated effort with a higher rate of productivity measured.

5.6. Customer and developer satisfaction

The customer had a constant control on the development in progress and everyday feedback from him was received. Such arrangement was highly praised by the client at the project sign-off.

The developers' satisfaction was gathered through a questionnaire at the end of the last iteration. The major effect on the developers was the new discipline imposed by eXPERT. Working 40 hours a week with pair programming requires a lot of concentration and appeared rather exhausting for the developers. Developers also found paired programming a very useful style of working as everyone was strictly conforming to the coding standards.

Developers' common conclusion was that constant development of unit tests for the business logic did not give the desired improvements on the software quality as far the paired programming prevented to a certain extent the introduction of simple logical errors in the code.

5.7. Achieving business goals

Fig. 5 presents the main measurable business achievements after applying eXPERT within Rila

Solutions. The chart presents the percentages of improvement according to the business goals defined in Section 1.

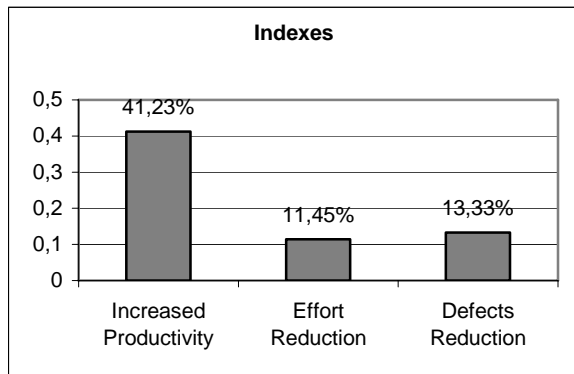


Figure 5. Indexes

The first impression is that the increased productivity gives a real percentage advantage on the other measured components. This of course is a result of the greater amount of the newly developed code for the pilot project compared to the baseline project. The baseline project required more modifications to the existing software modules than the pilot project while the pilot project required the development of more new functionalities. The first business goal – increased productivity by 20%, may be considered as fulfilled.

The second business goal – reduction of defect rates by 30% - is achieved in half with a rate of 13.33%. The defects reported and measured were mainly high level and critical defects reported by the QA team and the client. Client requests for amendments in the system were not included in both projects and for that reason were not compared. Having the client available “on-site” gives this approach an advantage in dealing with client requests as far the client reacts immediately reviewing the system.

Nevertheless the third business goal about effort reduction is not fully achieved the result of nearly 11.5 % is enough encouraging the application of eXPERT in the forthcoming company projects.

6. Conclusion

The eXPERT approach has been designed for projects aiming to deliver quality software, i.e. with low defect rate, in a short time. Rila Solutions experiment shows that it is relatively easy to apply eXPERT in small teams and in organizations with a flat hierarchy, because the direct communication to the management is important for the success of the projects.

The projects that will benefit from using eXPERT are projects that have no clear detail specification in

the beginning. Applying eXPERT is also preferred for projects that have no clear definitions about the final result – when the client only has a broad-brush picture of the envisioned system without knowing its detailed frames and final features.

Prior to applying eXPERT in a software development company the client commitment to the project should be carefully assessed. The benefits achieved with a committed client can be easily lost if communication fails.

In general introducing a new approach is difficult and usually an adaptation period is recommended. The Rila Solutions case study shows significant improvement in productivity, reduced defects rate, reduced efforts spent and is a first step for better results.

The presented case study is a contribution to the experience factory of empirical findings on agile methodologies based on XP. Some of the proposed metrics could be integrated in XP-EF framework [7] which may be used in future projects for more detailed evaluation of eXPERT accepting.

Acknowledgements

This work was the result of the IST-2001-34488 project, called “eXPERT“, that was funded by EC 5FP.

10. References

- [1] Beck K., “Extreme Programming Explained: Embrace Change”, Addison-Wesley, 1999
- [2] Bozeva T., “Deliverable D1 – eXPERT approach”, IST-2001-34488 project, July 2003
- [3] Highsmith J., "e-Project Management: Harnessing Innovation and Speed", e-Business Application Delivery Journal, Vol. 1, No. 1, Cutter Consortium 2000
- [4] Humphrey W., “A Discipline for Software Engineering”, Addison-Wesley, 1995
- [5] Ilieva S., E. Stefanova, “eXPERT approach for e-business software development”, International conference “Basic Technologies for E-business’2002”, Albena, 16-18 September, 2002
- [6] Leire Orue-Echevarria Arieta, “D11 - Final report on IST-2001-34488 project, July 2003
- [7] Williams L., W. Krebs, L. Layman, “Extreme Programming Evaluation Framework for Object-Oriented Languages Version 1.3”, North Carolina State University Department of Computer Science, Raleigh, NC TR-2004-11