

EXPERT APPROACH AND FRAMEWORK FOR ANALYSING EXPERIMENTS WITH IT

Eliza Stefanova, Sylvia Ilieva

*Department of Information Technologies, Faculty of Mathematics and Informatics,
Sofia University,*

5, James Bourchier Blvd., 1164 Sofia, Bulgaria

Abstract: The paper introduces a light method for software development applicable to small teams developing projects characterised with often changing requirements, tight schedules, and high quality demands. The method is called eXPERT and builds on the principles of eXtreme Programming (XP) and the Personal Software Process (PSP). The paper describes the objectives and logics of the eXPERT approach, as well as framework for analyzing experiments with it.

Key words: software development process, eXtreme Programming (XP), Personal Software Process (PSP).

1. INTRODUCTION

All SME developing e-commerce and e-business software have similar business objectives, namely “Faster-Better-Cheaper”. Reports from the field show that productivity and software quality increase by applying XP principles [1]. However, even projects that have adopted several or all XP practices meet project management problems, namely related to estimating and planning the project, both in terms of time and costs. To overcome this obstacle *eXPERT* approach focuses on combining XP and PSP practices [3]. *eXPERT* approach is agile and integrated and provides a novel set of practices for development of e-projects.

eXPERT approach complements XP, PSP, and some common sense software engineering practices. Since PSP is oriented to individuals, the practices are adapted as to suit the XP ones, where software developers work in pairs and the design, coding and testing activities are highly coupled instead of being executed in consecutive phases. The paper is organized as follows: Section 2 presents *eXPERT* objectives. Section 3 discusses the criteria which such lightweight method should meet. Section 4 describes the framework for analyzing experiments with *eXPERT* approach. Section 5 concludes the paper.

2. EXPERT OBJECTIVES

The main aim of the European project named *eXPERT* is to define a light approach for software development - *eXPERT*, combining the principles of XP and PSP, which will address the goals and the specifics of e-projects (e-business and e-commerce software projects). The project will teach developers in different SMEs how to work effectively in an atmosphere of often changing requirements; to construct the simplest system that has business value; to accurately estimate development work to avoid schedule slips, overcosts; to work effectively with the customers; to maintain software code in top condition. The project consortium establishes a network of Centers of Expertise (CE) in Spain, Germany and Bulgaria, and User companies associated to each CE. Seven pilot projects will be run at the User companies experimenting the application of the defined approach to e-project development. The Framework for Analyzing Experiment is a means for defining experiment objectives, reviewing experiments and estimating their achievements based on the *eXPERT* approach for e-project development.

The results aimed to be achieved by the end of the eXPERT project by the SMEs performing the experiments are as follows:

- Increase productivity by approximately 20% due to the design of simpler, easy to understand systems and avoiding rework
- Reduce defect rates by 30%.
- Reduce their project overruns by about 15% due to avoiding schedule slips.
- Increase their competitiveness by producing higher quality products in shorter time and better prices

3. EXPERT APPROACH

Proposed method is called *eXPERT* and is build on the principles of XP and PSP. The criteria which eXPERT approach has to satisfy are:

- To achieve defined business goals.
- To maintain the approach easy to apply.
- To provide all data necessary to perform good estimations, project planning and management activities.

The first step in the process of creation of lightweight method for software development applicable to small teams developing projects characterised with often changing requirements, tight schedules, and high quality demands is to define the business goals to be pursued. The eXPERT business goals are:

- Faster development
- Better quality
- Reduction of additional costs

Next step is to define how the most popular activities in software development are related to and contribute for reaching those business goals:

Faster software development (which is not leading for reduced quality) could be based on:

- *Management of customer requirements*. This means that the real client demands are comprehended instead of the supposed client needs. *Active participation of the client* who in every moment could say his expectations for the developed software. Those two elements provide possibility not to spare time for realizing features, which are not needed.
- *Simple design*, which usage simplifies the future changes in software and thus contributes to its faster development.
- *Development of iterations*, it allows to trail the time needed for implementation of given functionality and to define more precisely the time needed for future ones. In this respect, it speeds up software development.
- *Test driven development*, which means that tests for validating given software are developed first i.e. before software itself. Once created those tests support automatic checking of code correctness in subsequent changes.
- *Pair programming*, i.e software is developed from two programmers working together at the same time at one machine and on the same task. This XP practice could bring to faster development mostly because of the permanent control of the produced code, the reduced defects and as a result to reduced time to eliminate them in future steps.
- *Continuous integration* – maintaining of common integration machine where at every moment the whole team could see the working version of the software developed.

Increased quality could be achieved with:

- Test driven development – risks of defect appearance are reduced by creating tests for checking the code before it has been written itself. Acceptance tests, which automatic answer the question if the developed system is the required one.

- *Coding standards*, which creation and keeping results in better quality.
- *Pair programming* influences quality by permanent mutual control
- *Registering and analyzing defects*, which leads to avoiding their future appearance and further to increased quality.

Reduction of additional costs could be achieved by:

- Project planning
- Project observation and control
- Effort estimation
- Registering and analyzing defects. This indirectly influences reduction of additional costs (from reducing defects to reducing time to detect and remove defects).

The approach could be easily applied if:

- Moderate usage of additional measurements. This could be very important to make the method practical – it has to be approach, used from software developers, for whom it is difficult to adopt every additional requirement not directly related to coding. These considerations pull especially with small, young software companies, where there are no approved software development processes.
- It is described in the terms of processes. The processes have short overview, process input, process output, completion criteria, and measures. This type of description will facilitate the easily application the approach in SMEs with already established software development process.

4. FRAMEWORK FOR ANALYZING EXPERIMENTS WITH EXPERT APPROACH

The developed Framework for Analyzing Experiments [2] can be used by software development projects applying the *e*XPERT approach. The particular implementation of the FAE depends on project peculiarities.

The suggested Framework for analyzing experiments (FAE) addresses the following issues:

- Activities to be performed to settle an e-project
- Activities to be performed to review e-projects and estimate their achievements
- Forms and templates for performing the activities
- Questionnaires used to determine project status.

Fig. 1 depicts the main concepts related to FAE.

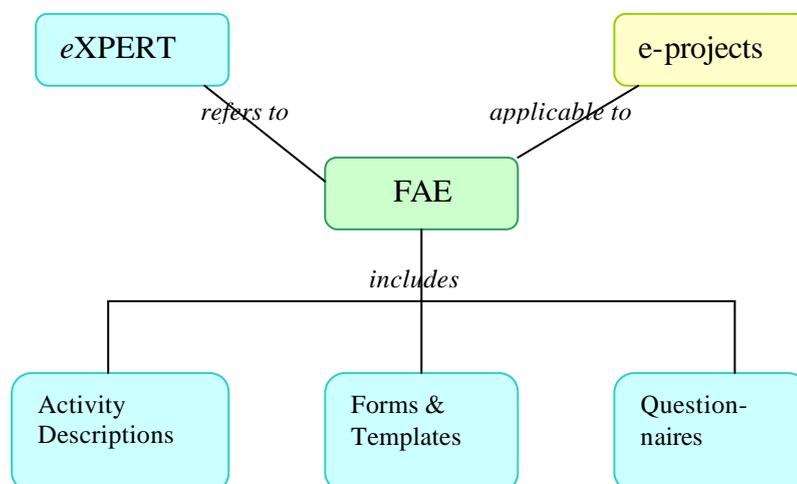


Fig.1. Main concepts of FAE

The Framework for analyzing experiments includes three types of activities:

- Plan & Conduct Project
- Review Project
- Analyse & Report Project Results.

Because of the lack of space we will give short descriptions of activities and we will not discuss forms & templates and questionnaires.

4.1. Plan & Conduct Project

This activity could be divided in two sub activities – Description of project baseline and Project planning.

The project baseline is used for comparing project results with a previously completed or running in parallel (baseline) project. Other step within Description of project baseline activity is establishing the scope of the project. The portion of software development process, XP and PSP practices, which will be specially observed during a project, is its scope. The scope has to include one or more software development processes, one or more PSP principles, and one or more XP practices.

Planning a project passes through the following steps:

Schedule project activities relates to developing the project breakdown structure and planning the task execution in the time. The project plan has to show how many releases are foreseen to be made, approximately how long each release is, and how many iterations are planned for each release.

Select team involves determining the people, who will be involved into the project. All team members must be trained in the *e*XPERT approach.

Assign resources means determining who team member is going to perform which task. All tasks must have resources assigned to them.

Plan costs involves making an estimate about the current iteration cost, and hence for the whole project. It would be helpful to analyse cost per iteration (labor and other types of expenditures), accumulated cost from the beginning of the project and expected total project cost.

Determine data to be collected in the project run. The data to be collected in the project run must provide a basis for making conclusions related to the business goals of the *e*XPERT project, namely: rate of defects, team productivity, project costs and schedule execution.

4.2. Review Project

Related to the XP practices it is recommended to review a project near releases. These reviews could be performed either internally by the project teams or by other supervising organization. In *e*XPERT project reviews are planned to be performed by Centers of Expertise.

The activities here are:

Selecting and preparing a review team which involves identifying the review team leader, selecting each of the team members, and providing the team with instructions about performing the review as well as with orientation in the problem domain of the project to be reviewed.

Plan and prepare review. The review team has to prepare a document, which: identifies review goals, identifies review activities to be performed, provides a schedule for the activities and determines a mechanism for estimation and communicating the review results.

Conducting a review focuses on collecting data about a project. Different techniques could be used for collecting data. Within the *e*XPERT project the most meaningful data that has to be collected during a review is:

- Rate of errors in the current iteration/release/project
- Team productivity
- Iteration/Release/Project cost
- Schedule deviations (in terms of man-days or planned vs. implemented features)

- Strengths of the *e*XPERT approach
- Difficulties met during applying the *e*XPERT approach
- Aspects of the *e*XPERT approach that have to be improved

4.3. Analyse & Report Project Results

Analyze&Report Project results activity consists of several consecutive activities:

Consolidate data. During a review all team members collect data in the form of notes. The data collected has to cover all the aspects of the *e*XPERT approach and the business goals of the *e*XPERT project. After the review data are consolidated and observations are defined. Observations primarily represent information concerning the software processes and practices performed by the reviewed team that describes its correspondence with the *e*XPERT approach.

Rate observations. Two aspects of the *e*XPERT approach have to be rated: application of *e*XPERT processes and *e*XPERT project business goals completion. The scale for rating is following: satisfied, not satisfied and not rated. It is possible other rating scale to be explicitly defined, if it differs from the one described in this activity

Project success rate. A project is defined to be “*successful*”, if it achieves defined objectives with at most 10% negative deviation, or exceeds the objectives.

Identify additional improvements achieved. Additional factors determining project success will be observed and estimated like communication between development team and customer, adjustment to changes into Requirements into the project run, coding standards and others.

Develop Case Study. The Case Study document represents the results achieved into the project and respective lessons leant.

5. CONCLUSIONS

The paper is subsequent in series of articles presenting *e*XPERT approach - a light method for software development applicable to small teams developing projects characterised with often changing requirements, tight schedules, and high quality demands. The method is built on the principles of XP and PSP. The objectives and logics of the *e*XPERT approach, as well as framework for analyzing experiments with *e*XPERT approach are described. The *e*XPERT organizational structure (roles and responsibilities) and detailed processes description will be presented in next papers.

Acknowledgments

This work was the result of the IST-2001-34488 project, called “*e*XPERT “ that was funded by EC 5 Framework Programme.

REFERENCES

1. Kent Beck. The XP Series: Extreme Programming Explained, Addison Wesley, 2000
2. Deliverable D2 of *e*XPERT project IST-2001-34488. Framework for analyzing experiments, 2002
3. Watts S. Humphrey. A Discipline for Software Engineering; Addison Wesley, 1995