



Computer Science

Opponent(s):

Zak Blacher

Respondent(s):

Christian H. Becker

**Using eXtreme Programming in a Student
Environment: A Case Study**

1 A General Evaluation of the Project

Overall I remain unconvinced of the study performed. The conclusions do not seem well supported by the understanding and interpretation of the surveys. To be blunt, it seemed to me as if the students made use of their standard development model and called it eXtreme Programming without a proper understanding for the differences. See the use of the repository to store documentation as an example of this.

2 Comments on the Project in Relation to the Dissertation

This project seems more related to sociology than computer science, but reflects the broadeng.

The sample data is too small to be used statistically. 1 class of 10 students from similar backgrounds and socioeconomic group working on 1 project for 5 weeks can't generate enough data. The conclusions will be drawn from a sample size of 1.

Student evaluations seem to rely too heavily on self-awareness.

2.1 Title

The title is fine.

2.2 Dissertation Layout

No complaints.

2.3 Scientific Method

There is a section devoted to evaluative research methodology, but is not mentioned further. Hypotheses and conclusions are presented, but there is no well-formed chain between them. This will be detailed where more relevant.

2.4 Argumentation and Conclusions

Well established arguments, but conclusions do not seem solidly supported by evidence. Final summary conclusion seems to disagree with self-survey.

2.5 The Abstract

No comments.

2.6 Language Aspects

A few 'denglish' expressions. Some word ordering is minced. In some summaries, the language took on a very different, more informal attitude,

2.7 References and Sources

(Relevant comments in chapters)

2.8 General Comments on the Project

Interesting topic, but possibly more suited to sociology than computer science.

3 Chapter by Chapter Evaluation of the Dissertation

3.1 Chapter 1

No Comments

3.2 Chapter 2

S2.4 -- a history of case studies. is this necessary?

I don't quite understand what's being said with tables 2-1 & 2-2

S2.6 "some kind of an instant expert" -- different tone from the rest of the paper

3.3 Chapter 3

Capitalization issues

in title.

Section 3.2 -- 'Caused by that problem'

S 3.3.3 -- Why is it surprising that these averages are similar? Shouldn't they be? Should they not be?

should the talk about these results be in chapter 4 or 5 instead?

3.5.1 -- 11 rules -- #8 should be 'extent'

Language: "Understand" "Undecided" "Not Understand" --- should be "Understood"

3.6.3 -- answers within one question (team) --- perhaps a better word than 'team'

3.7 -- 10 swedish male students may not be representative enough. Yes, it is the whole population, but nowhere does the study say it is limited to male swedish CS students in their mid-twenties.

3.8 -- "snapshot"

3.4 Chapter 4

Why didn't people understand 'Slack'?

4.3.2.2 -- is LOC count a proper productivity metric? why is this here?

this seems to suggest that most of the work was done by 2-3 people

4.3.2.3 -- "practice has not been applied fully."

SVN single check-in could fault entire conclusion by actions of one person.

was it made clear that documentation / help files break XP practice? this is standard practice.

4.5.2 -- (just above item 4) "this will be discussed more detailed"

Item 4 -- perhaps a miscommunication as to the strictness of the rules enforced by the null hypothesis?

Item 5 -- title is strange.

Under Fig 4.9 -- "XP did not work well" instead of "XP Worked Not Well"

4.5.3 -- "students not influenced by observers" how about each other? herd mentality?

"supports the process of drawing the big picture" <- language change (formality issue)

3.5 Chapter 5

5.2 -- distributed sit-together? teleconferencing?

Deloitte & Touche is now just 'Deloitte'

5.6 -- 'Stories is' <- reword.

5.16 -- How are Slack and Negotiated Scope Contract different?

3.6 Appendix

Typos on page 88 - “JAR - Java Archive”

“TDD – Test-Driven Development”

3.7 General Comments on the Dissertation

Much of these conclusions seem to suggest language being the issue, rather than the implementation of a development practice. This suggests that either XP wasn't properly explained in terms that could be understood. Cognitive bias may have been a factor in the results from the surveys.

This is made obvious in these sections of the conclusion chapter: {Shared Code v. SVN Check-ins, Slack v. Feature reduction, Weekly Cycle}

4 Final Comments

I get the feeling that most of the disparities between the observed and the reflected results stem from language issues rather than implementation problems. From what's discussed, it would appear that most of the relevant XP paradigms are implemented, even when the students are not aware of it. The development structure of the students seems to stem from self-learned styles rather than strict XP. This is not well reflected in the final conclusion.