Predicting Good Requirements for In-house Development Projects

June Verner, Karl Cox, Steven J. Bleistein
Empirical Software Engineering
National ICT Australia
Sydney, Australia
june.verner, karl.cox, steven.bleistein@nicta.com.au

ABSTRACT
We surveyed software practitioners regarding software development practices used during recent projects. Five categories of questions broadly related to requirements were asked: the sponsor, customer/users, requirements issues, the project manager and project management, and the development process. Relationships between project factors and good requirements were investigated. We developed requirements prediction equations by dividing our response data into two data sets. Using binary logistic regression on each set we tested the equations developed. Such analysis provides us with insight into which variables are significant predictors of good requirements. The best predictors were 1) the customers/users had a high level of confidence in the development team, with 2) risks were controlled and managed by the project manager.

Categories and Subject Descriptors
D. Software, D.2 Software Engineering, D.2.1 Requirements/Specification

General Terms
Management

Keywords
Requirements, management, risk management, prediction

1. INTRODUCTION
Good requirements are significantly related to successful software project outcomes [35] and organizations that implement effective requirements engineering (RE) practices reap multiple benefits [37]. A Standish Group Report revealed that two of the top ten reasons for challenged failed projects were lack of user involvement, and unstable requirements [31].

Concern has been expressed about the lack of RE in industrial projects; there is a need for more practical RE research [38]. It is important to examine what methods are used in practice and which practices lead to good requirements. In a world of constrained resources our contribution is to provide project managers (PM) with insight into: 1) which practices have the greatest impact on RE and hence a project's success, 2) where to focus attention when resources are constrained and 3) insight into identifying projects at risk early.

Our paper is organized as follows: in Section 2, we describe our survey and discuss some overall details of the data collected. In Section 3, we present our results in detail. Section 4 presents our conclusions and suggestions for further research.

2. OUR STUDY
Based on discussions with a large number of software practitioners who develop in-house software, and the software engineering literature, we developed a questionnaire to investigate software development practices that lead to successful project outcomes. Our questionnaire was distributed to practitioners from a large US financial institution (Group 1) who responded by answering it twice, once for a recent project they considered successful and once for a recent project they considered a failure. The questionnaire was later distributed to a second group of practitioners from various companies in North Eastern U.S.A. (Group 2), who each answered the questionnaire once. Our sample is not random but rather a convenience sample.

The survey was organized into a number of sections covering the software development process. We asked respondents if they considered the project they referenced when answering 1) was a success and 2) if it had good requirements at some stage during the development process. Only questions relating to the development of good requirements are considered in this paper. Sections of the questionnaire not considered here are discussed elsewhere e.g., [33, 34, 35].

In all we received completed questionnaires from 102 respondents, reporting on 123 distinct projects. As noted earlier, the majority of our respondents were developers involved with software for use within their own organizations. The responses to the first set of 42 questionnaires (Group 1), described 42 separate projects, 21 regarded as successful and 21 unsuccessful. The second set of responses (Group 2) included descriptions of 81 unique projects.

Overall 62% of projects were regarded as successful and 38% unsuccessful, 87% were development projects (55% successful), and 13% were large maintenance/enhancement projects in terms of effort (66% successful). Forty four percent of Group 1 projects had good requirements while 78% of the Group 2 projects had good requirements. Group 2 had significantly more successful projects, and projects with good requirements than Group 1.

ISESE'06, September 21–22, 2006, Rio de Janeiro, Brazil.

Group 2 respondents reported on one project only; the majority chose a successful project.

The percentage of projects by number of full-time IT employees is 1-4 = 44%; 5-9 = 19%; 10-19 = 21%; 20-29 = 7%; 30-39 = 3%; 40-99 = 5%; and 100-180 = 1% (range 1-180, mean 13, median 6). There was no difference between the groups for number of full-time staff working on a project.

3. RESULTS AND ANALYSIS

The percentages of "yes" responses overall to the questions are shown in Table 1. Table 2 provides details of "yes" responses to the questions for projects with good requirements, by group. Table 3 presents significant associations between variables for projects with good requirements by group, and overall. We performed Chi Square tests to determine the degree of association between variables, and correlation analyses to provide the direction of that association. In the following analysis, if a pair of variables is significantly associated (<0.05) and positively correlated, we refer to them as *significantly associated*. When it occurs, we mention negative association explicitly. In all tables our questions are categorized as follows: "S" refers to questions that deal with the project sponsor/senior management, "C" refers to questions related to customers and users, "R" refers to questions directly related to requirements issues, "M" to questions related to the PM and project management, and "P" to questions related to the development process.

3.1 Project Sponsor/Senior Management

A powerful political sponsor can assure that a project is adequately resourced, that customers and users make sufficient time available for requirements gathering [27] as well as supporting realistic scheduling preventing schedule changes, or other undermining changes, and resource planning [20, 21]. Lack of commitment of top management is the highest ranked risk factor by "experienced software project managers" [17].

Overall, all the "S" variables were significantly associated with good requirements, though S4 was negatively associated. The variables S1, *the project began with a committed sponsor*, S2, *sponsor commitment lasted through the project*, and S3, *the sponsor was involved in project decisions,* showed a high degree of multi co-linearity. Surprisingly, there was no correlation of S1, S2, or S3 with S4, *senior management negatively impacted the project.* This finding appears to contradict practitioners' initial comments that powerful sponsorship protects a project against interference from competing interests. This may be because of changed organizational priorities or because the project sponsors were not senior enough to protect the project from external interference.

In summary:

Our results support prior research that suggests having a committed sponsor who lasts the distance lends stability to requirements. However, this sponsorship may not always protect the project from external interference.

Predicting good requirements using "S" variables

Using Group 1 as the basis for developing a prediction equation, we obtained a single variable, S2 (*sponsor commitment lasted through the project*), which predicted 78% of projects with good requirements, 65% of projects without good requirements and 71% of projects overall on its own data set. We were unable to develop a satisfactory prediction equation for good requirements using the "S" variables with the Group 2 data set. This may be because there were few projects without sponsor support. Our results suggest that none of the "S" variables is an acceptable predictor of good requirements on its own. However a committed, politically powerful sponsor is unlikely to actively take part in the requirements elicitation process.

3.2 Customer/users

We have combined customers and users into one category because the practitioners who participated in discussions for this study did not distinguish between the two. Unrealistic customer and user expectations can arise because projects start with incomplete requirements [36]. An explicit user-inclusion strategy should be adopted for effective requirements gathering as user support and enlightened involvement are important for ownership, "buy in" and for managing user expectations [11, 20, 25, 27, 29, 31, 32]. Customer/user participation can reflect confidence in the development team, positive expectations, and the desire to contribute knowledge of the business needs. We found a high degree of multi co-linearity among C1, C2, C3, C4, C5, and C6, suggesting that (C1), *a high level of customer/user involvement,* may result in (C4), *commitment and involvement of other stakeholders,* (C2)*, customers and users having a high level of confidence in the development team*, and (C3), *involved customers and users will then stay right through the project*. Our analysis also suggests that C1 is very important as *a high level of customer/user involvement* may lead to (C6), *customers/users have realistic expectations,* and (C5), *they will make adequate time available for requirements gathering,* thus implying good requirements. The importance of user involvement in requirements gathering (C5) supports observations of Clavadetscher [7], Glass [12] and Procaccino et al. [27]. Contrary to what practitioners had suggested in our initial discussions, we did not find that large numbers of customers and users impacted the development of good requirements.

The relationship between customer/user involvement (C1) with level of confidence in the development team (C4) raises an interesting question about causality. Are customers/users involved because they are confident in the development team or do they become more confident in the development team because of their involvement?

In summary:

We were surprised that large numbers of customers and users did not impact establishing good requirements, though this may reflect the relatively small size of the projects in our sample. Further research will clarify the effects of large numbers of customers and users on the requirements elicitation process.

It is interesting that a high level of confidence in the development team is so related to good requirements. The confidence that the customers and users have in the development team is not an area typically addressed in the RE literature. Our research supports customers/users making adequate time available for requirements gathering as an important requirements determinant. This is one of the most frequently identified factors for the development of good requirements. The relationships between all of the "C" variables and S2 are in agreement with research that stresses the importance of a committed sponsor whose commitment lasts throughout the project [30].

Table 1: Percentage "Yes" Responses to Questions

ID	Question	With good requirements[1] % Yes	Without good requirements[2] % Yes	All projects[3] % Yes
S1	The project began with a committed sponsor	90	63	80
S2	Sponsor commitment lasted through the project	82	35	66
S3	Sponsor was involved in project decisions	77	40	64
S4	Senior management negatively impacted the project	19	38	26
C1	High level of customer/user involvement	73	43	62
C2	Other stakeholders were committed and involved	73	36	60
C3	Involved customers/users stayed right through project	82	57	73
C4	Customers/users had high level of confidence in development team	73	14	52
C5	Adequate time made available for requirements gathering	80	43	68
C6	Customers/users had realistic expectations	63	17	46
C7	Customer/user's expectations managed throughout	82	40	65
C8	Problems caused by large numbers of customers/users	23	38	28
R1	There was a central repository for requirements	69	35	59
R2	Project size impacted elicitation of requirements	26	51	35
R3	Requirements gathered using specific methodology	56	35	49
R4	Project had a well-defined scope	84	33	67
R5	Project scope increased during the project	58	80	66
R6	Requirements were managed effectively	69	23	51
M1	PM given full authority to manage project	70	50	63
M2	PM was above average	68	23	54
M3	PM related well to staff	69	26	56
M4	PM had a clear vision of the project	83	49	72
M5	PM really understood the customers problem	75	46	65
M6	PM communicated well with staff	65	23	52
M7	PM was experienced in the application area	68	68	68
M8	Years of experience of the PM < 10	66	82	71
M9	PM's background (IT, Business, other)	47, 38, 15	65, 26, 9	53, 34, 13
P2	Development methodology appropriate for project	62	23	47
P3	PM able to choose the methodology	34	39	36
P4	Risks identified at the beginning of the project	80	33	62
P5	Risks incorporated into the project plan	66	32	53
P6	Risks controlled and managed by the PM	63	10	43
P7	Project had effective change control	73	30	60
P8	An approach to control quality used	76	37	59
P10	Other projects negatively impacted this project	25	57	38

[1] This column represents the percentage of "yes" answers to questions for projects that had good requirements

[2] This column represents the percentage of "yes" answers to questions for projects that did not have good requirements

[3] This column represents the percentage of "yes" answers to the questions for all projects.

Table 2: Percentage "Yes" Responses to Questions by Group

ID	Question	Group 1 projects with good requirements[4] % Yes	Group 2 projects with good requirements[5] % Yes
S1	The project began with a committed sponsor	71	86
S2	Sponsor commitment lasted through the project	55	73
S3	Sponsor was involved in project decisions	52	73
S4	Senior management negatively impacted the project	24	27
C1	High level of customer/user involvement	60	64
C2	Other stakeholders were committed and involved	52	64
C3	Involved customers/users stayed right through project	74	74
C4	Customers/users had high level of confidence in development team	41	61
C5	Adequate time made available for requirements gathering	61	70
C6	Customers/users had realistic expectations	29	58
C7	Customer/user's expectations managed throughout	64	68
C8	Problems caused by large numbers of customers/users	24	33
R1	There was a central repository for requirements	56	61
R2	Project size impacted elicitation of requirements	50	26
R3	Requirements gathered using specific methodology	50	46
R4	Project had a well-defined scope	42	78
R5	Project scope increased during the project	71	65
R6	Requirements were managed effectively	39	57
M1	PM given full authority to manage project	64	64
M2	PM was above average	45	61
M3	PM related well to staff	54	58
M4	PM had a clear vision of the project	65	74
M5	PM really understood the customers problem	70	63
M6	PM communicated well with staff	48	57
M7	PM was experienced in the application area	78	63
M8	Years of experience of the PM < 10	90	83
M9	PM's background (IT, Business, other)	66:21:13	44:43:13
P2	Development methodology appropriate for project	42	56
P3	PM able to choose the methodology	42	33
P4	Risks identified at the beginning of the project	52	69
P5	Risks incorporated into the project plan	46	58
P6	Risks controlled and managed by the PM	24	57
P7	Project had effective change control	41	67
P8	An approach to control quality used	50	67
P10	Other projects negatively impacted this project	56	30

[4] This column represents the percentage of "yes" answers to questions for projects that had good requirements

[5] This column represents the percentage of "yes" answers to questions for projects that had good requirements

Table 3 Significant Associations of Questions with Good Requirements by Group

ID	Question	Direction	Group 1	Group 2	Overall
S1	The project began with a committed sponsor	+	0.024	0.043	0.001
S2	Sponsor commitment lasted through the project	+	0.006	0.000	0.000
S3	Sponsor was involved in project decisions	+	0.045	0.004	0.000
S4	Senior management negatively impacted the project	-		0.023	0.034
C1	High level of customer/user involvement	+	0.004		0.001
C2	Other stakeholders were committed and involved	+		0.005	0.001
C3	Involved customers/users stayed right through project	+	0.044	0.037	0.007
C4	High level of confidence in development team	+	0.001	0.000	0.000
C5	Adequate time made available for requirements gathering	+		0.000	0.000
C6	Customers/users had realistic expectations	+		0.000	0.000
C7	Customer/user's expectations managed throughout	+	0.006	0.004	0.000
C8	Problems caused by large numbers of customers/users	-		0.019	
R1	There was a central repository for requirements	+		0.001	0.000
R2	Project size impacted elicitation of requirements	-		0.006	0.006
R3	Requirements gathered using specific methodology			0.006	
R4	Project had a well-defined scope	+	0.024	0.000	0.000
R5	Project scope increased during the project	-		0.017	0.015
R6	Requirements were managed effectively	+		0.000	0.000
M1	PM given full authority to manage project	+		0.029	0.043
M2	PM was above average	+	0.003	0.004	0.000
M3	PM related well to staff	+	0.000	0.035	0.000
M4	PM had a clear vision of the project	+	0.037	0.008	0.000
M5	PM really understood the customers problem	+	0.024	0.009	0.003
M6	PM communicated well with staff	+	0.019	0.005	0.000
M7	PM was experienced in the application area				
M8	Years of experience of the PM < 10				
M9	PM's background (IT, Business, other)				
P2	Development methodology appropriate for project	+		0.019	0.000
P3	PM able to choose the methodology				
P4	Risks identified at the beginning of the project	+	0.002	0.007	0.000
P5	Risks incorporated into the project plan	+	0.030	0.021	0.000
P6	Risks controlled and managed by the PM	+	0.000	0.027	0.000
P7	Project had effective change control	+		0.002	0.000
P8	An approach to control quality used	+	0.028	0.032	0.001
P10	Other projects negatively impacted this project	-		0.048	0.001

Predicting good requirements using "C" variables

The prediction equation based on the Group 1 data set and the "C" variables comprised a single variable C4 (*customers/users had a high level of confidence in the development team*). This equation predicted 67% of projects with good requirements, 82% of projects without good requirements and 76% of projects overall on its own data set, but predicted all Group 2 projects as successful. This suggests that C4 (*customers/users had high level of confidence in development team*) does not predict good requirements on its own.

We were unable to develop a satisfactory prediction equation for good requirements using the "C" variables and the Group 2 data set. This is surprising considering the amount of literature that states the customer/users must be included in the project throughout the lifecycle, not just in the requirements phase. However, this could be an artefact of the number of successful projects in the group.

3.3 Requirements Issues

The results in Tables 1 and 2 support the view that requirements continue to be a problem for software development [13, 22] and agree with [23], whose respondents thought their companies did not do enough RE. Although definition of a requirements development process at the start of a project will normally include the use of a RE methodology [37] we found that R3, *requirements gathered with a specific methodology*, was not significantly associated with good requirements. This may be because, in 79% of the projects, respondents did not know what requirements methodology was used. For the remainder of projects, interviews and questionnaires were the main requirements gathering method. For the ones that did know, three projects used prototyping and six used JAD sessions with prototyping. Eight of the nine projects using prototyping and/or JAD had good requirements.

Eight projects used UML for documenting requirements, but only four projects were successful. The use of UML as a requirements notation has been criticized [14, 18], "most UML models are not appropriate for requirements work; while they are good design models they are lousy requirements models" [28]. However, using UML was found to be better than using no methodology.

It is essential to manage requirements throughout the development process [37]. As a group, the variables associated with requirements management exhibit the greatest overall impact on project success [27].

We found that R1, *there was a central repository for requirements,* was significantly associated with good requirements, indicating that a central repository supports effective requirements management. That only 59% of the projects used a central repository suggests that there is still significant room for improvement in requirements management.

Practitioners suggested that large projects, in terms of functionality, are less likely to be successful than smaller projects. R2, *project size impacted elicitation of requirements,* was significantly negatively associated with good requirements. This result agrees with [12], suggesting that project size hampers requirements gathering, and can lead to unclear, incomplete, and potentially unstable requirements.

Wiegers [37] addresses a number of good RE practices including the need for a well-defined project scope. R4, *the project had a well-defined scope,* and R5, *project scope increased during the project,* were both significantly associated with good requirements, R5 negatively. An increase in scope and unstable requirements pose major risks to software projects [15]. R5, *project scope increased during the project,* was also significantly associated with R2, *project size impacted elicitation of requirements.* This supports the view that the larger the project and the longer it goes on, the more growth in scope developers are likely to experience [16].

Predicting Good Requirements using "R" variables

In developing an equation on the Group 1 data set with the "R" variables we found that R4 (*the project had a well-defined scope*) was the best predictor of good requirements, predicting 61% of projects with good requirements, 74% of projects without good requirements and 68% of projects overall on its own data set. Using logistic regression with the Group 2 data set, R4 (*the project had a well-defined scope*) predicted 90% of projects with good requirements, 59% of projects without good requirements, and 83% overall on its own data set. This variable (R4), in a single variable equation was the best predictor of good requirements for the Group 2 data set. However, the best Group 2 predictor of good requirements was an equation comprising two variables, R4 (*the project had a well defined scope*) with R6 (*the requirements were managed effectively*), which predicted 97% of projects with good requirements, 70% of projects without good requirements, and 91% of projects overall on its own data set. For the Group 1 data set, the same equation (R4 + R6) predicted 69% of projects with good requirements, 75% of projects without good requirements and 73% of projects overall.

3.4 Project manager and project management

In situations where they worked with PMs who were not given full authority to manage a project (M1) our practitioners reported that senior management constantly interfered with and second-guessed the PM. This led to lack of motivation and, in many cases, project failure. In agreement with our practitioners, the data shows a significant relationship between M1, *PM was given full authority to manage the project,* and good requirements. We were surprised to find that a good PM was just as likely to suffer from interference as a poor PM.

M4, *the PM had a clear vision of the project*, was associated with good requirements. Defining project vision is good engineering practice [37]. A project that is without a clearly defined and well-communicated direction invites disaster [37]. Lack of a clear vision leads to poorly defined goals and specifications, poor requirements, insufficient project planning, lack of a project plan, and unrealistic deadlines and budgets [9]. This underscores the importance of understanding requirements beyond micro-level user needs [29]. M4 is significantly associated with M5, *the PM really understood the customer's problem*, and both are significantly associated with good requirements. A clear vision is necessary for a PM to really understand the customer's problem.

Communication between the PM and the project team is also important [2]. M5, the *PM communicated well with staff*, was significantly associated with good requirements.

Our practitioners suggested that M7, *the PM is experienced in the application area*, will increase the chances of a project's success. However, our data did not support this. M7 was not significantly associated with either project success or good requirements.

Years of project management experience ranged from under 6 months to 22 years, with over 60% of PMs having more than three years experience, and 15% more than 10 years. Our practitioners suggested that an experienced PM is more likely to be associated with a successful project. Unexpectedly, the data did not support this as M8, *PM's years of experience,* was not significantly associated with either project success or good requirements.

Our practitioners also suggested that a PM with an IT background was more likely to be associated with a successful project. However, the results did not support this either, as M9, *PM's background,* was not significantly associated with either project success or good requirements. PMs with business or other backgrounds were just as likely to be successful.

There was a high degree of multi co-linearity between most "M" variables. Analysis suggests that a project with a PM who is given full authority to manage the project (M1), who is above average (M2), relates well to staff (M3), has a clear vision of the project (M4), really understands the customer's problems (M5), and communicates well with staff (M6), is likely to have good requirements. These results show that, for PMs, vision, communication and relationships with team members are more important than any particular background, underscoring research that stresses the need for a PM to have good interpersonal skills [9, 10].

In summary:

Because the project management literature generally assumes that a PM has full authority to manage a project, we did not expect that M1, *the PM was given full authority to manage the project*, would enter into the prediction equation for good requirements. In initial discussions, practitioners suggested that interference threatened project success. We were surprised that in more than one third of projects, PMs were subjected to interference. Analysis suggests that, when interference occurs, it is mainly related to staffing issues, and adequate staffing is significantly associated with good requirements.

The importance of M4, *PM had a clear vision of the project*, reinforces the importance of project scope, but includes an extra dimension; the importance of knowing expected business outcomes beyond just the project parameters. Effective communication is also frequently suggested as a key to good requirements, and our analysis supports this.

Predicting Good Requirements using "M" variables

In developing a prediction equation on the Group 1 data set, we found that M6 (*PM communicated well with staff*) with M5 (*PM really understood the customer's problem*) predicted 71% of projects with good requirements, 91% of projects without good requirements, and 82% of projects overall on its own data set. Using just M6 (*PM communicated well with staff*) we were able to predict 77% of projects with good requirements, 77% of projects without good requirements, and 77% of projects overall on its own data set.

Using the Group 1 equation to predict Group 2 projects we found that all projects were predicted to have good requirements, suggesting that neither of the one variable or two variable equation is a valid predictor in this case.

The prediction equation for "M" variables based on the Group 2 data set included M2 (*PM was above average*), which predicted fairly poorly (100% projects with good requirements, 8% of projects without good requirements and 83% of projects overall) on its own data set.

The Group 2 equation predicted the Group 1 projects without good requirements surprisingly well, correctly predicting 71% of projects with good requirements, 77% of projects without good requirements and 74% projects overall.

3.5 Development Process

Good development practices include processes such as selection of an appropriate lifecycle methodology, managing risks, specifying quality attributes, and change control processes [37].

Table 4. Prediction Equations

Variable Category	Respondent group	Question Number	Prediction Equation	Prediction % on own group **Good requirements: not good requirements : overall**	Prediction % on alternate group **Good requirements : not good requirements : overall**
S	**Group1**	**S2**	**Sponsor commitment lasted through the project**	**78 : 65 : 61**	**Not meaningful**
	Group 2	**NA**	**-**	**-**	**-**
C	**Group1**	**C4**	**Customers/users had a high level of confidence in the development team**	**67 :82 : 76**	**Not meaningful**
	Group 2	**NA**	**-**	**-**	**-**
R	**Group1**	**R4**	**Project had a well defined scope**	**61 : 74 : 68**	**90 : 59 : 83**
	Group 2	**R4** **R6**	**Project had a well defined scope** **Requirements were managed effectively**	**97 : 70 : 91**	**69 : 75 : 73**
M	**Group1**	**M6** **M4**	**PM communicated well with staff** **PM really understood the customer's problem**	**71 : 91 : 82**	**Not meaningful**
	Group 2	**M2**	**PM was above average**	**100 : 8 : 83**	**71 : 77 : 74**
P	**Group1**	**P10**	**Risks were identified at the beginning of the project**	**85 : 70 : 76**	**Not meaningful**
	Group 2	**P4**	**Other projects negatively affected this project**	**85 : 40 : 74**	**50 : 85 : 72**
NA = No prediction equation could be developed					

Overall Prediction Equations

Respondent group	Question Number	Prediction Equation	Prediction % on own group **Good requirements : not good requirements : overall**	Prediction % on alt. group **Good requirements : not good requirements : overall**
Group 1	C4 P6	**Customers/users had a high level of confidence in the development team** Risks were controlled and managed by the PM	100 : 85 : 91	84 : 67 : 81
Group 2	R6 C4	**Requirements were managed effectively** **Customers/users had a high level of confidence in the development team**	100 : 80 : 96	69 : 85 : 79

Using a *methodology appropriate for the project* (P2) is significantly associated with good requirements. An appropriate methodology and a well-defined scope allow for well-defined deliverables. While the practitioners suggested that P3, *the PM is able to choose the development methodology*, was important for a successful project outcome, our results did not support this. Some organizations forced PMs to use a specific life-cycle development methodology, irrespective of the problem. However, when PMs were given a choice, their projects were no more successful.

Change happens; change is unavoidable. It is very difficult to define all requirements in a single discrete requirements stage [37]; hence, managing requirements successfully includes effective change control. Our results support this with P7, *the project had effective change control,* significantly associated with good requirements.

The quality of software project management is characterized by active risk management [12, 24]. This observation is supported by the correlation between responses to questions P4, *risks were identified at the beginning of the project*, P5, *risks were incorporated into the project plan* and P6, *risks were controlled and managed by the PM*, and M2, *the PM was above average*. Even though risk management practices are significantly associated with good requirements [1], most developers and PMs perceive risk management activities as extra work and expense [13]. Just identifying the risks without doing something about them is not enough.

P10, *other projects impacted this project*, a recognized risk factor, was significantly negatively associated with good requirements and P6, *risks were controlled and managed by the PM.*

In summary:

Our results suggest that when risks are controlled and managed by the PM, we will get good requirements. However, there are risks outside the control of the PM, such as concurrent projects, that can compete for scarce resources.

Predicting Good Requirements using "P" variables

We found that with Group 1, P4 (*risks were identified at the beginning of the project*) predicted 85% of projects with good requirements, 70% of projects without good requirements and 76% of projects overall on its own data set.

This equation predicts all Group 2 projects as having good requirements, suggesting it is not a meaningful predictor on its own. The prediction equation developed from the Group 2 data set, included P10 (*other projects negatively affected this project*) which correctly predicted 85% of projects with good requirements, 40% of projects without good requirements, and 74% of projects overall on its own data set.

Applying the Group 2 equation to predict Group 1 projects, we correctly predicted 50% of projects with good requirements, 85% of projects without good requirements and 72% of projects overall.

3.6 Prediction of Good Requirements

The goal of this research is to develop a prediction equation for good requirements. Table 4 summarizes the significant predictors for each variable type. For example, the prediction equation developed from the Group 1 data set for the Sponsor (S) variables, included a single variable S2, (*sponsor commitment lasted through the project*). The column "Prediction % on own group" shows that S2 predicted for Group 1, 78% of projects with good requirements, 65% of projects without good requirements and 71% overall. S2 predicted that all Group 2 projects had good requirements, so is not a meaningful predictor on its own. No "S" variables in the Group 2 data set predicted good requirements (NA in Figure 4).

In developing a prediction equation with the Group 1 data set that considered all variable categories (S, C, R, M, and P), we tested variable combinations that were significantly related to good requirements. The best combination was C4 *(customers/users had a high level of confidence in the development team*) with P6 (*risks were controlled and managed by the PM),* which predicted 100% of projects with good requirements, 85% of projects without good requirements, and

91% of projects overall on its own data set. When this equation was tested on Group 2, it successfully predicted 84% of projects with good requirements, 67% of projects without good requirements and 81% overall (refer to the second last row of Table 4). The percentage of correctly predicted projects overall falls within the boundary set by Boehm of 70% [3]. Though P6 (*risks controlled and managed by the PM*) is recognised in the software engineering literature as critical to project success [19, 24], C4 (*customers/users had high level of confidence in development team*) is rarely discussed. Yet this study found it to be highly significant.

Curiously, the best predictor of good requirements from the Group 1 data set does not include any explicit requirements engineering practices, indicating that the success of requirements engineering does not rely solely on requirements practices. It is critical to manage risks and it is also critical that customers/users have a high level of confidence in the development team. This means that requirements engineers, project sponsors, managers and developers should establish good, open relationships with the customers/users and maintain this relationship throughout the whole project.

Developing a prediction equation with Group 2 data from all variable categories, results in R6 (*requirements were managed effectively*) with C4 (*customers had a high level of confidence in the team*), which predicted 100% of projects with good requirements, 80% of projects without good requirements, 96% of Group 2 projects overall. Applying this equation to the Group 1 data set predicts 69% of projects with good requirements, 85% of projects without good requirements and 79% overall.

The appearance of C4 in both overall prediction equations suggests it is a very significant factor. Though many organisations engage in customer relationship management, or the management of customer expectations, this is not the same as C4. There is little consideration of customer confidence in the development team in the RE literature. An area for further research is to investigate what makes customers confident in their development team. Unsurprisingly, R6, *the requirements were managed effectively*, is a significant attribute in the prediction equation. Good requirements management is well recognized as a key factor in project success [19, 37].

Overall, R4, *the project had a well-defined scope*, is the most influential single factor, as alone it predicts 68% of Group 1 projects and 83% of Group 2 projects with good requirements correctly. Our results support the hypothesis proposed by Davis and Zowghi [8] that good requirements practices are not sufficient for success and show that practices beyond the scope of RE, such as the commitment of the sponsor throughout the project, the confidence of the customer/users in the development team, a skilled PM, and project processes that include risk management not only lead to good requirements but, ultimately, to project success.

4. CONCLUSIONS AND FURTHER RESEARCH

Overall, *the project had a well-defined scope*, was shown to be the most influential single factor as alone it predicted whether or not a project had good requirements in 68% of Group 1 projects and 83% of Group 2 projects. Other significant factors <u>in combination</u> were better predictors for good requirements and included *customers/users had a high level of confidence in the development team*, *requirements were managed effectively* and *risks were controlled and managed by the PM.* Implications for practice are that a PM who has limited resources should ensure that the above factors are fully addressed. Most of these factors should be acted upon early in the project lifecycle. Defining project scope, setting up a risk management structure, and management plans for defined requirements management procedures, such as a central requirements repository, are critical actions in setting up software projects. The confidence that the customers/users have in the development team is rather more problematic and requires further research as discussed below.

Summarizing, we also found that for good requirements:

1) it is not having a sponsor that is important, per se, but having a sponsor whose commitment lasts throughout the project;

2) it is not the number of users involved that is important, but rather the size of the project in terms of functionality;

3) it is not having a PM with years of experience, or a PM experienced in the application area, but rather a PM who manages requirements effectively;

4) it is not just the identification of project risks, but doing something about them, after they have been identified; and

5) it is projects that have one, and only one, central repository for requirements.

Table 1 indicates that current practices can be scored as fair at best. Analysis of our results suggests further research is required in order to investigate:

1) The value of distinguishing more clearly between requirements scope, problem scope and project scope. Does a good definition of scope at the outset of a project enable project teams to better manage requirements that change or evolve over the course of a project?

2) Customer involvement and customer confidence in the project team indicate an improved likelihood of success. Do customers become more involved because they are confident in the team, or are they confident because they are involved? What motivates customer involvement with the development team? What instils customers with confidence in the development team?

3) How generalizable are the factors identified in this study?

While we believe that the results of this initial study are significant on their own, we intend to compare against factors important for good requirements in other environments such as across national borders.

The major contributions of our study are to:

1) reinforce the importance of grounding RE research in industry practice,

2) show that good requirements are determined by more than requirements practices and

3) emphasize that in a world of constrained resources, it is essential that PMs understand which practices generate the highest return.

This research serves as a starting point for motivating our continuing research into requirements practice in industry.

5. REFERENCES

[1] Addison T. and Vellabh S. (2002), "Controlling Software Project Risks- an Empirical Study of Methods Used by Experienced Project Managers" Proc. of SAICSIT, pp128-140.

[2] Bandinetti S., Di Nitto E. and Fuggetta A., “Supporting Co-operation in the SPADE-1 Environment, IEEE TSE, Vol 22 No 12 pp 841-865.

[3] Boehm B. W. (1981), Software Engineering Economics, Prentice Hall, Englewood Cliffs, NJ.

[4] Boehm B. W. (1991), “Software Risk Management Principles and Practice”, IEEE Software Vol. 8 No 1, pp32-41

[5] Boehm, B. W. and Basili, V. R., (2001), “Software Defect Reduction: Top 10 List”, IEEE Computer, 34(1): 135-137.

[6] Brooks F. P. Jr. (1975), The Mythical Man Month. Essays on Software Engineering, Addison Wesley, USA.

[7] Clavadetscher, C.(1998) “User Involvement: Key To Success”, IEEE Software. Vol. 15, Issue 2 (Mar/Apr) p. 30, 32.

[8] Davis A and Zowghi D (2006) “Good requirements practices are neither necessary nor sufficient” Requirements Engineering Journal Vol 11 No 1, pp 1-3.

[9] DeMarco T. and T. Lister (2003), Waltzing With Bears, Dorset House Publishing New York NY.

[10] DeMarco T and Lister T., Peopleware: Productive Projects and Teams, (1987) Dorset House Publishing Co, NY.

[11] Gause D and Weinberg G, Exploring Requirements: Quality Before Design (1989), Dorset House, NY.

[12] Glass, R. L. (1998) “How Not To Prepare For A Consulting Assignment And Other Ugly Consultancy Truths”, Communications of the ACM. Volume 41, Issue 12 (December) pp11-13.

[13] Glass R. L. (2001), “Frequently Forgotten Fundamental Facts about Software Engineering” IEEE Software, May/June, pp112, 110, 111.

[14] Jackson, M. (2001), Problem Frames, Addison Wesley.

[15] Jones T.C. (1994), Assessment and Control of Software Risks, Prentice Hall, Englewood Cliffs NJ.

[16] Jones T.C. (1996), “Strategies for Managing Requirements Creep”, IEEE Computer, Vol 26, No 6, pp. 92-94.

[17] Keil, M., P.E. Cule, K. Lyytinen and R.C. Schmidt (November 1998) A framework for identifying software project risks, Communications of The ACM. 41(11): 76-83.

[18] Kovitz B., (1999), Practical Software Requirements, Manning

[19] Leffingwell, D. and D. Widrig (2000) Managing Software Requirements: A Unified Approach, Addison-Wesley.

[20] McConnell, S. (1996), Rapid Development, Redmond, Washington: Microsoft Press.

[21] McKeen, J.D. and T. Guimares (1997) “Successful strategies for user participation in systems development”, Journal of Management Information Systems. 14(2): 133-150.

[22] Moynihan T. (1997), “How experienced Project Managers Assess Risk”, IEEE Software, Vol 14, (MAY/JUNE) pp35-41.

[23] Neill C. J. and Laplante P. A. (2003), “Requirements Engineering: State of the Practice” IEEE Software, November/December, pp 40-45.

[24] Osmundson J. S., Michael J. B., Machniak, M. J., and Grossman M. A. (2003), “Quality management metrics for software development” Information and Management 40, pp799-812

[25] Paulk, M., B. Curtis, M. Chrissis, C. Webster (1993) Capability Maturity Model for Software, Technical Report, CMU/SEI-93-TR-024, Software Engineering Institute, Carnegie Mellon.

[26] Pressman, R. (2000), Software Engineering: A Practitioners Approach, adapted by Darryl Ince, 5th edition, McGraw Hill, London.

[27] Procaccino, J.D., Verner, J. M., and M. Darter, “Toward Predicting Software Development Success from the Perspective of Practitioners: An Exploratory Bayesian Model” to appear in JIT.

[28] Robertson J. and Robertson S. (2000), “Requirements Management: A Cinderella Story”, Requirements Engineering Journal, 5 (2), pp134-136.

[29] Roy V., and Aubert B. (2003), “A Dream Project Turns Nightmare: How Flawless Software Never Got Implemented”, Annals of cases on Information technology Idea Group Publishing pp 98-111

[30] Schmidt, R., Lyytinen, K., Keil, M., and Cule, P., (2001) “Identifying Software Project Risks: An International Delphi Study”, Journal of Management Information Systems, Spring, Vol 17, No 4, pp 5-36.

[31] Standish Group (1999), “Chaos: A Recipe for Success”, Standish Group International.

[32] Tackett, B. and van Doren B., (1999), “Process Control for Error-Free Software: A Software Success Story”, IEEE Software 16(3): 24-29.

[33] Verner J. M. and Evanco W. M. (2003), “An Investigation into Software Process Knowledge” in Managing Software Engineering Knowledge, (eds) A. Aurum, R. Jeffery, C. Wohlin and M. Handzic, Springer-Verlag, pp29-47.

[34] Verner J. M. and Evanco W. M. (2005), “In–house Software Development: What Software Project Management Practices Lead to Success?” IEEE Software Jan/Feb vol. 22, issue 1, pp86-93.

[35] Verner J. M, Cox. K. Bleistein S and Cerpa N. (2004) “Requirements Engineering and Software Project Success: An Industrial Survey in Australia and the U.S. Australia Journal of Information Systesm Vol 13, No 1 pp225-238.

[36] Verner, J. M., Overmyer, S. P. and McCain, K.W. (1999) In the 25 years since the mythical man-month what have we learned about project management?, Information and Software Technology. 41(14): 1021-1026.

[37] Wiegers, K. (2003), Software Requirements, Second Edition Microsoft Press, Redmond, WA.

[38] Weiringa, R. and Ebert, C. (2004), “RE’03: practical Requirements Engineering Solutions” IEEE Software March/April, pp. 16-18.