Case Study:  
Factors for Early Prediction of Software Success & Failure

J. D. Procaccino, J. M. Verner, and S. P. Overmyer  
College of Information Science & Technology,  
Drexel University,  
3141 Chestnut Street,  
Philadelphia, Pa. 19104-2875 USA

ABSTRACT
Factors such as poor requirements, lack of management support, and customers and users who do not make themselves readily available significantly affect software development success. The earlier in the development process problems arise the more serious is the outcome [2]. When there are limited resources, a project manager needs to know which factors are likely to have the most severe impact on the project. In order to investigate some of these early factors and their effect on the success or failure of software projects we develop an organizational case study. After engaging in structured discussions with twenty-one IT professionals from a large North American financial institution, we asked each participant to respond to a survey addressing two development projects with which they had recently been associated. Our investigation treats these software development projects as a composite case. In addition, we investigate differences in management and developer perceptions of the success or otherwise of our projects and the prediction of project success from the viewpoint of both groups. We find that both managers and developers consider the level of customer and user involvement contributes most to project success. Developers perceive the next most important factors to be (1) that customers and users have realistic expectations and (2) the scope of the project is well defined. On the other hand, to predict success from management’s viewpoint, participation of customers and users in estimating project schedules is significant.

Keywords
project management, project success and failure, requirements impact, customer and user impacts, management impact

1 INTRODUCTION
Project success can be defined as a project that “meets its budget, delivery and business objectives”. Project failure refers to a project that has either “been cancelled or … does not meet its objectives” [6]. In 1994, 31% of all corporate software development projects resulted in cancellations [6] even though there is extensive research into (and many guidelines for) successful software development [e.g., 2, 7, 12, 9]. Clearly, systems development can present problems to an organization with money, time and talent at risk. Recent research that focuses on factors contributing to the failure of information systems projects takes a practitioner view (such as that of an IT manager, project manager or software developer) [5, 13, 7, 14]. Management, who is paying for the project, has its own view of what constitutes a successful project. In some cases this view is different from that of the developers, while users may take another view (note: we will not be considering the user’s view of project success here, as our research has a different focus). Major categories of factors contributing to the success or failure of software systems are: (1) requirements, (2) management, (3) customers and users, (4) estimation, and scheduling, (5) the project manager, (6) the software development process, and (7) development personnel.

The focus of the research described here is to use a case study approach to provide insight into the most influential early success or failure factors within a particular corporate environment. More insight into these factors will help project managers and other project stakeholders to predict the likelihood of project success, early enough for the development of both a risk mitigation strategy and a plan for corrective action. The earlier a project manager discovers that the project is going off course, the more effectively and efficiently can adjustments be made. In most business environments, conditions for software development projects are less than ideal, with many pressures and constraints on the project manager and development team. A project manager may need to choose which factors to concentrate on to ensure the best chance of project success.
This study is a first step to the establishment of guidelines for project risk prioritization in the early phases of software development. Because of the number of generic risks the project manager has to contend with, we would eventually like to provide a severity roadmap showing the consequences of early project actions (taken or imposed). In addition, we wish to determine if there are any differences between management and developer’s perception of project success. To be successful in a corporate environment the project manager needs to be aware of success and failure factors from all stakeholder viewpoints, and be able to respond to these perceptions either directly or indirectly [10].

2 BACKGROUND

The following factors emerge as central, reoccurring risks encountered early in nearly all software development projects:

2.1 Management¹ Support:
Inadequate management practices have far reaching implications for project success [1]. A project needs a committed sponsor throughout, who must encourage the same commitment from other stakeholders as well as participation in the decision-making process.

Project stakeholders include [3]:

- Project management (who want to keep the project within time and monetary constraints)
- Project sponsor or champion
- Developers
- Customers.
- Users (who want to be kept informed of developments that may effect them)
- External consultants (who want to “finalize” user specifications as quickly as possible).

2.2 Customer/User:
A recent survey found that one of the most important contributors to successful project development is end-user involvement [8]. Further, involvement by all appropriate users should occur in all phases of system development [1]. Expectations of customers and users are also an important contributor towards the perceived success or failure of a software project. Even given strong buy-in, stakeholders of large-scale system development projects often have vastly different expectations and goals regarding the completed system [3]. Realistic expectations can reduce conflict and this in turn supports the perception of project success, from both the developer and managerial standpoint [7].

2.3 Requirements
A clear understanding of the problem to be solved assists in well-defined requirements, and is essential to the development of a successful system [11, 15]. However, “we fail to use requirements management to surface (early) errors or problems” [4]. Recent research noted that poor requirements are involved in most project failures [5] and that “requirements continue to be a huge problem for IS development” [11]. Unrealistic expectations that customers and users have regarding their software projects often arise because projects start with incomplete requirements [9]. Specific project problems related to requirements include:

- Changing and/or unstable requirements during development [7].
- Fuzzy project scope with user indecision regarding what problem(s) should be solved [11].
- Incomplete requirements (data, and reports), resulting in incomplete information about the proposed system [11].
- The various goals of the stakeholders cannot be met simultaneously and some may even be in conflict [3].
- Misunderstood requirements and feature creep can result in misallocated resources, re-work, late delivery (if at all) and excessive monetary cost.

Given these as yet unsolved software project problems, our investigation combines a number of software development efforts as a composite case. The idea is to shed some light (from two stakeholder perspectives), on the relative importance and severity of a number of these risk factors, as they are perceived to impact project success.

3 OUR INVESTIGATION
In late 1999 we engaged in structured discussions with 21 IT professionals, including developers, technical support personal, project leaders and project managers (referred to in the body of this paper as “respondents” or “developers”), from a large financial institution. Our discussions focussed on the issues within their organization that they identified as associated with software project success and failure. Based on these discussions and through the development of a questionnaire that targeted software development practices within the organization, we attempt to discover which development practices are most important for software project success (from both developer and management viewpoints). We asked our IT professionals to complete two questionnaires, with one related to a successful project, and the other, to a less successful, or unsuccessful, project. While are results are specific to this institution we believe that they are generalisable to other similar organizations.

Our questionnaire was arranged under the following headings: (1) requirements, (2) management, (3) customers and users, (4) estimation, and scheduling, (5) the project manager, (6) the software development process, and (7)
development personnel. We also asked (a) was the project pleasant to work on? (b) did management consider the project a success (MSUCCESS) and, (c) from the respondent’s point of view, was the project a success (RSUCCESS)? Although we addressed all the above seven topics in the questionnaire, the research reported here is focused on issues central to the early phases of project development - namely management support, requirements, and customer and user involvement. While some of the questions may appear ambiguous to an outside reader, because of our earlier discussions the questions were not ambiguous to our respondents. The results of our investigation are presented next. We follow this with a discussion section and conclude with lessons learned.

4 RESULTS
We received data from 42 recent software development projects. These projects were all in-house MIS development (rather than maintenance) projects. The average number of full-time developers working on these projects was eleven (minimum of one and maximum 80) with an average of five contractors (plus consultants) (minimum 0 and maximum 10). We list each question and report on our results under the following headings: Management support, Customer/users, and Requirements. We follow this with sections on Multivariate analysis, and Management and developer’s views of success. Our discussion is mainly focussed on those results that are important in the prediction of either MSUCCESS or RSUCCESS. We used Chi-square tests at the .05 level to determine which of the results are significant; Table 3 presents those variables. Overall, while management thought 78% of our projects were successful the respondents considered only 49% of the same projects a success.

4.1 Management support
We began with a number of assumptions regarding management support and project success including: (1) projects whose project managers are given full authority to manage their project are more likely succeed; (2) a project with a committed sponsor(s), is more likely to succeed than a project without a committed sponsor. A project that begins without a committed sponsor(s) is already in trouble; (3) a project whose sponsor leaves or withdraws support during the development process will not succeed. Based on these assumptions the following questions were asked and answers obtained:

Q1.2 Did the project start with a committed sponsor (or champion)? Both management and respondents’ perception of success decreased when the project began without a committed sponsor; a Chi-square test was not significant with MSUCCESS but was significant with RSUCCESS (0.028). Respondents considered 80% of the projects that began without a committed champion a failure though only 33% of these projects were considered a failure by management.

Q1.3 Did this commitment last right through the project? A project will be more likely to be successful if it has sponsor commitment that lasts throughout the project. Both management and respondents’ perception of project success decreased when sponsor/management commitment did not last throughout the project and Chi-square tests were significant with both MSUCCESS (0.032) and RSUCCESS (0.008). Not surprisingly, there is a significant correlation between Q1.2 and Q1.3. Overall, 63% of our projects had sponsor commitment from beginning to end (though 20% had no commitment at all and 7% began without a sponsor but managed to pick one up along the way). For projects with ongoing commitment, 90% were viewed as a success by management, and 74% a success by respondents. Where the commitment for a project did not last, the number of successful projects from management’s view dropped to 50%, and respondents considered only 17% of these projects a success. With no sponsorship at any stage of the development only 33% of the projects failed in management’s view, though a whopping 86% failed as far as the developers were concerned.

Q1.4 Was the sponsor involved in project decisions? Both management and respondents’ perceptions of success remained relatively constant whether or not the sponsor was involved in project decisions. A Chi-square test was not significant with MSUCCESS or RSUCCESS.

Q1.5 Were the other stakeholders committed and involved? Management’s perception of success was relatively consistent whether or not the other stakeholders were committed and involved. However, the respondents’ perception of success dropped by approximately one-half if there was a lack of commitment and involvement by the other stakeholders. A Chi-square test was not significant with either MSUCCESS, or RSUCCESS.
As can be seen in Figure 1, the significant issues associated with manage support for software projects are reflected in Questions 1.2 and 1.3. That other issues were not statistically significant does not diminish our ability to glean useful information.

4.2 Customer/users

Earlier, we summarized research that has shown that customers and users can contribute significantly to project success. The assumptions behind our customer/user questions are: projects will succeed if they have (1) customer/users who are highly involved and who stay involved throughout the project, (2) customer/users who contribute to project estimates, (3) customer/users with realistic expectations, and (4) a low level of customer/user turnover. With these points in mind, we asked the following questions.

Q2.1 What was the level of involvement of customer/users? As customer/user involvement increases, so do the chances of a project being viewed as a success by managers and developers. Management categorized as successful, across all levels of customer/user involvement, a higher percentage of projects, than did respondents. Chi-square tests were significant with both MSUCCESS (0.024) and RSUCCESS (0.000).

Overall, 43% of our projects had high customer/user involvement, 22% had a reasonable level, 7% had some involvement, 26% had little involvement and only 3% had no customer/user involvement. Figure 2 illustrates the relationship between management and respondents’ views of success across the varying levels of customer/user involvement. Both groups’ perceptions of success increased as customer/user involvement reached “reasonable” and “high” levels.

Q2.2 Did the involved customer/users stay right through the project? Both developers and management will be more likely to view the project as a success if customers/and users stay involved right through a project. Both perceptions of success decreased noticeably when the customer/users were not involved throughout the project. Chi-square tests were significant with both MSUCCESS (0.007) and RSUCCESS (0.010).

Overall, 79% of projects had their customer/users attention throughout the project (5% had no customer/user involvement at all). Almost 90% of projects where the customer/users stayed involved throughout were considered a success by management. Respondents considered 60% of such projects a success. For those projects where the involved customer/users did not stay throughout the project, 43% were considered a success by management, but only 14% were considered a success by developers.

Q2.3. What level of confidence did the customer/user have in the project manager/team members? The higher the level of confidence that customers/users have in the project manager and development team the more likely developers are to consider the project a success. Chi-square tests were not significant with MSUCCESS but were with RSUCCESS (0.001).

Forty-eight percent of projects had high or very high customer/users confidence in the project team, 26% reported an average level, 21% low and 5% very low. If we consider only those projects reported with “high” or “very high” levels of customer/user confidence in the project manager/team, management viewed 84% of these projects a success, while 75% were considered a success by respondents. For only those projects reported to have an “average” level of confidence, management still considered 72% successful, though respondents’ views of success for these projects dropped to 36%. When viewing projects where there was a “low” level of confidence management still considered almost 70% of these projects a success, though respondents’ views of the success rate dropped like a stone to 10%. No projects with very low customer
confidence in the development team were perceived as a success. Figure 3 illustrates the relationship between management and respondents’ views of success across the varying levels of confidence customers and users had in project team.

Q2.4 Were the customer/users involved in making schedule estimates? Both management and respondents’ perception of success decreases when customer/users are not involved in making schedule estimates. A Chi-square test was significant with MSUCCESS (0.044) but not significant with RSUCCESS. Forty-one percent of the projects had customer involvement and of these 94% were considered a success by management and 56% a success by developers. We conclude that the involvement of customers/users in making schedule estimates increases the likelihood that management will consider the project a success but does not make any difference to developer perceptions.

![Figure 3. Level of Confidence in Project Team for Successful Projects - Q2.3.](image)

**Q2.5 What level of customer/users staff turnover did you have to contend with?** Essentially, the respondents’ perception of project success decreased as customer/user staff turnover increased. Management categorized a higher percentage of projects as successes than did developers across all levels of staff turnover. A Chi-square test was not significant with MSUCCESS or RSUCCESS.

Q2.6 Did the customer/users have realistic expectations? Developers are more likely to consider the project a success if the customers/users have realistic expectations. Customer expectations do not make any difference to project success as far as management is concerned. A Chi-square test was not significant with MSUCCESS, but was with RSUCCESS (0.007). Both management and developer’s perceptions of success decrease noticeably when the customer/users do not have realistic expectations.

Twenty-three percent of projects were reported as having unrealistic customer/user expectations though for 33% of our projects, the respondents did not know what the customer/user expectations were. Eighty-three percent of projects having customer/users with realistic expectations were considered a success by management, while 72% of these projects were considered a success by developers. When considering only those projects where the customer/users did not have realistic expectations, 75% were considered a success by management, while only a quarter were considered a success by developers. Developer views of the success rate dropped from 72% to 25% as expectations became unrealistic.

Q2.7 Did you run into problems due to the large number of customer/users involved? Although it appeared from our results that success might increase with a large number of customer/users, the statistical evidence did not support this, as a Chi-square test was not significant with either MSUCCESS or RSUCCESS.

As Figure 4 illustrates, the significant issues associated with customer/user involvement in a software project are reflected in Questions 2.1, 2.2, 2.3, 2.4, and 2.6. Again, that other issues were not statistically significant does not diminish our ability to glean useful information.

![Figure 4. Significant Results Related to Customer Involvement](image)

4.3 Requirements

We expected that those projects viewed as successful would (1) begin with complete and accurate requirements, that were (2) gathered using a specific methodology with (3) a well defined scope that (4) remained consistent throughout the project. With these points in mind, we asked the following requirements-related questions.

Q3.1 Were requirements gathered by using a particular method? A Chi-square test was not significant with either MSUCCESS or RSUCCESS. Management and developers views were similar.

Q3.2a Were the requirements complete and accurate at the start of the project? Management’s perception of success

---

2 Q2.1 and Q2.3 have been recoded into two new variables, above average, and not above average.
was relatively consistent whether or not requirements were complete and accurate. However, developers’ perceptions of success decreased noticeably (from 78% to 41%) when requirements are not complete and accurate at the start of a project. A Chi-square test was not significant with MSUCCESS (0.982), but was with RSUCCESS (0.049).

Only 22% of our projects began with complete and accurate requirements. Of those projects that began with incomplete requirements, 40% had their requirements completed later. Seventy-eight percent of projects where the requirements were complete and accurate were considered a success by both management, and developers. When considering those projects where the requirements were not complete and accurate at the start of the project, 77% were considered a success by management, but only 41% were considered a success by developers.

Q3.2b If requirements were not complete and accurate, at the start were they completed adequately? Both management and developers’ perception of success decrease markedly when requirements are not completed adequately and Chi-square tests were significant with both MSUCCESS (0.013) and RSUCCESS (0.004).

Overall, 100% of projects where the requirements were not complete and accurate at the start of the project, but were later completed adequately were considered a success by management, while 69% of such projects were considered a success by developers. When considering those projects where the requirements were not completed adequately, while 63% were considered a success by management only 21% were considered a success by developers.

Q3.3 Was the scope of the project well defined? Management’s perception of success dropped somewhat if the project scope was not well defined (82% to 70%). However, developer’s perception of success dropped more severely for this factor (65% to 28%). Chi-square tests were not significant with MSUCCESS but were significant with RSUCCESS (0.017).

Fifty-five percent of projects had a well-defined scope and of these 82% were considered a success by management and 65% were considered a success by developers. When considering those projects whose scope was not well defined, 70% were considered a success by management, while only 28% were seen as a success by developers.

Q3.4 Did the scope change during the project? Both management’s and developer’s perception of success remained relatively constant whether or not the scope changed during a project and a Chi-square test performed on the responses was not significant with MSUCCESS or RSUCCESS.

Q3.5 Did the customer/users make adequate time available for requirements gathering? Though management’s perception of success remained relatively constant when customer/users did not make adequate time available for requirements gathering developers’ perception of success decreased substantially (63% to 9%) for such projects. A Chi-square test was not significant with MSUCCESS, but was significant with RSUCCESS (0.002). When customers/users do not make adequate time available for project requirements gathering developers are not likely to consider the project a success.

Customer/users made adequate time available for requirements gathering in 74% of our projects. Seventy-nine percent of these projects were considered a success by management, and 63% were considered a success by developers. For those projects where customer/users did not make adequate time available for requirements gathering, 73% were considered a success by management, while only 9% were considered a success by developers.

Q3.6 Was there a single central repository for the requirements? Management’s perception of success remained relatively constant whether or not there was a central repository for requirements. However, developer’s perception of success decreased dramatically (60% to 31%) if the project was without a central requirements repository. A Chi-square test was not significant with either MSUCCESS, or RSUCCESS.

Sixty percent of our projects had a single central repository for their requirements; 81% of these projects were considered a success by management, while 60% of such projects were considered a success by developers. For those projects that did not have a single repository for their requirements, 75% were considered a success by management, and only 31% were considered a success by developers.

Q3.7 Did the requirements result in well-defined software deliverables? Management’s perception of success remained relatively constant whether or not requirements resulted in well-defined software deliverables. However, developer’s perception of success decreased by more than half for projects lacking well defined software deliverables (64% to 25%). A Chi-square test was not significant with MSUCCESS, but was with RSUCCESS (0.016).

Sixty percent of our projects had requirements that resulted in well-defined software deliverables; 75% of these projects were considered a success by management, while 64% were considered a success by developers. For those projects whose requirements did not result in well-defined software deliverables, 81% were considered a success by management, though only 25% were considered a success by developers.

Q3.8 Did the size of project negatively affect requirements elicitation? Management’s perception of success increased
with large projects whose size negatively affected requirements elicitation. However, developer’s perception of success decreased notably (68% to 44%). A Chi-square test was not significant with MSUCCESS, but was with RSUCCESS (0.041).

Overall, 46% of our projects were large enough for their size to negatively impact requirements elicitation. When considering these large projects 88% were considered a success by management, but only 44% were considered a success by developers. In contrast 68% of the projects whose size did not have a negative impact were considered a success by management, while 55% of such projects were considered a success by developers.

As Figure 5 illustrates, the significant issues associated with requirements in a software project are reflected in Questions 3.2a, 3.2b, 3.3, 3.5, and 3.7. Each question, however, reveals interesting information, whether statistically significant or not.

Figure 5. Significant Results Related to Requirements

4.4 Multivariate analysis
Next, we conducted a multivariate analysis to explore the impacts that the variables generated from the management support, customer/users and requirements questions have on developer and management perceptions of project success (RSUCCESS and MSUCCESS). Since the success variables are coded as zero for “unsuccessful” and one for “successful,” we were unable to use ordinary least squares regression, which relies on a continuous dependent variable. Instead, we conducted a logistics regression analysis. We also recoded variable 2.1 (What was the level of involvement of customer/users?), from its original five-point scale to a two-point scale (above average level of involvement = 1; not above average level of involvement = 0), to avoid having to use dummy variables.

(1) Prediction of success from the developer viewpoint
Three variables entered our logistics regression equation ($R^2 = 0.60$) namely, (a) What was the level of involvement of customer/users? (b) Did the customer/users have realistic expectations? and (c) Was the scope of the project well defined? The parameter estimates for project success, all of which are significant at the 10% level, are shown in Table 1 below. Our logistic equation predicts correctly 83.78% of projects (82.35% of the unsuccessful projects are predicted correctly and 85.71% of the successful projects are predicted correctly). Only three of the unsuccessful projects are erroneously predicted as successful, and two successful projects are erroneously predicted as unsuccessful.

(2) Prediction of success from management viewpoint
Two variables entered our regression equation ($R^2 = 0.29$) namely, (a) What was the level of involvement of customer/users? and (b) Were the customer/users involved in making schedule estimates? The parameter estimates for project success, all of which are significant at the 10% level, are shown in Table 2 below. Our logistic equation predicts correctly 82.50% of projects (66.67% of the unsuccessful projects are predicted correctly, and 87.10% of the successful projects are predicted correctly). Three of the unsuccessful projects are predicted erroneously as successful and four successful projects are erroneously predicted as unsuccessful.

Table 1. Project success from developer perspective

<table>
<thead>
<tr>
<th>Variable</th>
<th>parameter estimate</th>
<th>standard error</th>
<th>probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of involvement of customer/users</td>
<td>3.14</td>
<td>1.34</td>
<td>0.02</td>
</tr>
<tr>
<td>Realistic customer/users expectations</td>
<td>1.91</td>
<td>1.11</td>
<td>0.84</td>
</tr>
<tr>
<td>Scope of the project was well defined</td>
<td>1.96</td>
<td>1.11</td>
<td>0.77</td>
</tr>
<tr>
<td>constant</td>
<td>-4.06</td>
<td>1.58</td>
<td>0.01</td>
</tr>
</tbody>
</table>

Table 2. Project success from management perspective

<table>
<thead>
<tr>
<th>Variable</th>
<th>parameter estimate</th>
<th>standard error</th>
<th>probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of involvement of customer/users</td>
<td>1.64</td>
<td>0.8565</td>
<td>0.06</td>
</tr>
<tr>
<td>Schedule estimates involved customers/users</td>
<td>1.89</td>
<td>1.15</td>
<td>0.10</td>
</tr>
<tr>
<td>constant</td>
<td>-0.16</td>
<td>0.61</td>
<td>0.79</td>
</tr>
</tbody>
</table>

The above results, will help project managers to predict the likely success or otherwise of their project and to be able to do so with reasonable accuracy early in the project. From everyone’s perspective, the level of involvement of the customers and users is the factor most critical to project success. Our results show that with a high level of involvement, the project has a much greater chance of success. From the developer perspective, the next most important factor for success is that the scope of the project is well defined; customer and users’ expectations follow in importance. Q2.4, which asked if customers/users are involved in schedule estimates, is an important predictor
for management success. This is an interesting finding as it means that if customers/users are involved in these estimates management cannot be blamed if the schedule not met. It is somewhat surprising that we are able to predict the success or otherwise of software development project with this degree of accuracy from so few project variables.

5 DISCUSSION AND CONCLUSIONS
This study leads us to a number of interesting conclusions for both management and developers, based upon our interpretation of the results in each category of analysis.

5.1 Management support
The following is a summary of the important results:
• A project can be considered successful even when a project manager is not given full authority to manage the project.
• Starting a project with a committed sponsor is important to development staff, but not necessarily to management.
• Having a project sponsor who is visibly committed for the duration of a software project is key to project success from both management and developer perspectives. A committed sponsor can make life immeasurably easier for the project team. Committed customers and users, who make adequate time available for requirements gathering will be likely to follow.
• A project sponsor who drops out of their role in the project has a more detrimental effect on perceptions of project success than starting without a sponsor and picking up one later. With no sponsor commitment developers perceive little chance of project success.
• Projects can be perceived as successful without the sponsor’s involvement in project decision-making. This helps define the role of project sponsor as political, rather than production support.
• If other stakeholders are both committed and involved with a project, the developers will feel better about it, but not significantly so. This commitment appears to be irrelevant to managers.

5.2 Customer/users
The following is a summary of the important results:
• If customers/users are involved to at least a reasonable level, a project is more likely to be viewed as a success by management and developers.
• Both developers and management will be more likely to view the project as a success if the customers/and users stay involved right throughout the project.
• The higher the level of confidence that customers/users have in the project manager and development team the more likely developers are to consider the project a success. Managers’ perceptions of project success do not seem to be impacted by stakeholder confidence.
• The involvement of customers/users in making schedule estimates does not increase the likelihood that developers will consider the project a success, however, their input appears to be important to managers.
• Low customer turnover appears to increase the perception of project success by developers, but not managers. This is easily explained by observing that it is the developers who must contend with customer/user turnover, not the managers.
• Developers are more likely to consider a project a success if the customers/users have realistic expectations. Customer/user expectations also appear to make a difference in management perceptions of project success, but not significantly so.
• Large numbers of customers/users did not affect anyone’s perception of the success of these projects. Having large user populations can be a double-edged sword, creating as much conflict as consensus.

5.3 Requirements
The following is a summary of the important results:
• If requirements are not complete and accurate at the start of a project developers are less likely to view that project as a success. The lack of accuracy and completeness of requirements, at the start of a project does make any difference to management’s perception of project success.
• If requirements are not completed adequately during the project neither management nor developers are likely to consider the project a success.
• Using a requirements gathering methodology did not appear to increase the perception of project success by either developers or management.
• If the scope of a project is well defined, management and developers are more likely to view the project as a success.
• Changing the scope of a project during development does not change either management’s or developers’ perception of project success.
• A central repository for requirements was important for success for developers but made no difference to management’s view of project success.
• When customers/users do not make adequate time available for project requirements gathering developers are likely to consider the project less successful. It does not change management’s perception of the success of projects if customers and users do not make adequate time available for requirements gathering.
• Well defined software deliverables were important for the success of a project in developers view but did not matter to management.
• Projects so large that the requirements elicitation is impacted were likely to be less successful so far as developers were concerned but this did not matter to management.
5.4 Manager’s and Developer’s Views of Success

As noted earlier, of our 42 projects, 78% were considered a success by management (MSUCCESS), and 49% were considered a success by developers (RSUCCESS). A Chi-square test indicated that there is a significant difference between MSUCCESS and RSUCCESS (0.001). Table 3 summarizes questions whose responses showed significant differences between MSUCCESS and RSUCCESS with Chi-square tests.

We believe that the difference in viewpoint is attributable to the different perspectives, motivation, and responsibilities typically associated with the respective roles of management and developers on projects.

Table 3. Questions Showing Significant Differences Between MSUCCESS and RSUCCESS

<table>
<thead>
<tr>
<th>Question</th>
<th>MsUCCESS</th>
<th>RsUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2 Did the project start with a committed sponsor (or champion)?</td>
<td>0.028</td>
<td></td>
</tr>
<tr>
<td>1.3 Did this commitment last right through the project?</td>
<td>0.032</td>
<td>0.008</td>
</tr>
<tr>
<td>2.1 What was the level of involvement of customer/users?</td>
<td>0.024</td>
<td>0.000</td>
</tr>
<tr>
<td>2.2 Did the involved customer/users stay right through the project?</td>
<td>0.007</td>
<td>0.010</td>
</tr>
<tr>
<td>2.3 What level of confidence did the customer/user have in the project manager/team members?</td>
<td>0.001</td>
<td></td>
</tr>
<tr>
<td>2.4 Were the customer/users involved in making schedule estimates?</td>
<td>0.044</td>
<td></td>
</tr>
<tr>
<td>2.6 Did the customer/users have realistic expectations?</td>
<td></td>
<td>0.007</td>
</tr>
<tr>
<td>3.2a Were the requirements complete and accurate at the start of the project?</td>
<td></td>
<td>0.049</td>
</tr>
<tr>
<td>3.2b If requirements were not complete and accurate, at the start were they completed adequately?</td>
<td>0.013</td>
<td>0.004</td>
</tr>
<tr>
<td>3.3 Was the scope of the project well defined?</td>
<td>0.017</td>
<td></td>
</tr>
<tr>
<td>3.5 Did the customer/users make adequate time available for requirements gathering?</td>
<td></td>
<td>0.002</td>
</tr>
<tr>
<td>3.7 Did the requirements result in well-defined software deliverables?</td>
<td></td>
<td>0.015</td>
</tr>
</tbody>
</table>

For example, considering Q2.4, while management may view customer input into schedule estimates as politically important (or expedient), developers might view this input as professionally irrelevant or even intrusive, since the typical customer input may not be based on technical factors. Another example of varied perspective is reflected in Q3.5, wherein the developers are very concerned with sufficient user access for requirements purposes, while managers are not. Since requirements gathering is the responsibility of the developers, it is no surprise that they consider customer availability as key. Management on the other hand, is concerned with the final product, and considers requirements as a deliverable from the developers, not from the customer. This difference in both perspective and responsibility potentially explains this effect. One particularly interesting result was that of Q3.2a, in which management appeared to be unconcerned with having accurate requirements at the beginning of a project, while developers were very concerned. In the following question (Q3.2b), both groups were concerned with completing the requirements effort. One possible explanation for this difference is that managers want to see code as soon as possible, while developers, guided by a lifecycle model expect a period of discovery and requirements development as a normal course for the project. In the final analysis, it appears that most of these differences in success factors can be attributed to respective project responsibilities with developers concerned primarily about things affecting their job performance throughout the projects, and management concerned only with the final products (and its possible effect on them).

6 LESSONS LEARNED

One of the most compelling reasons to measure the perceptions of those involved with software development is that participant perceptions of project success, consistent or not with reality, can dramatically affect the health of a project manager, a project team, and thus the health of the project. One of the most interesting lessons learned from this study involves the perceptions that the software development staff has about management’s view of project success and failure. As Table 3 shows, and as we noted earlier, there is a vast difference between developers’ perceptions of project success factors, and their perceptions of how management personnel view project success and failure.

Developers take a mainly inward-looking view of project success concentrating on the things that affect them and their ability to do their job properly. From their point of view, a successful project (and project manager) has (a) customers/users who make adequate time available for them (a project sponsor will help here), (b) a project scope that is well-defined (this will follow on from (a)), and (c) carefully managed customer/user expectations.
The management viewpoint appears to be more politically orientated - to keeping customers and users, rather than developers, happy, and passing (a share of) the blame to someone else if the schedule is not met. With the current level of IT shortages management needs to be more aware of what keeps developers happy in order to avoid staff turnover and loss of motivation.

Further questions that have emerged from this analysis include:

- How consistent are management's perceptions of project success and developer's views of those perceptions in both this, and other organizations? If they are consistent, then remedy is straightforward. If not, management must do a better job of manipulating developers’ perceptions to increase productivity and job satisfaction.

- Are customer's perceptions of project success consistent with both management's and developer's perceptions? If not, then perhaps recommendations can be made to achieve a better working relationship between customers/users and developers, so that views may be more consistent.

- How influential are specific perceptual variables in predicting or changing the course of unsuccessful projects? What are the critical success factors? We have suggested that on-going sponsor support may result in a number of positive actions such as users making sufficient time available for requirements gathering. Additional data and path analysis techniques should help to provide some answers to these question.

REFERENCES


