MAGISTERARBEIT

Titel der Magisterarbeit

„Finding Critical Success Factors for IT-Projects“

A Case Study on a Light Weight Approach

Verfasserin

Margaret A. Gruen-Kerr BA

angestrebter Akademischer Grad

Magistra der Sozial und Wirtschaftwissenschaften (Mag.rer.soc.oec.)

Wien, 2013

Studienkennzahl lt. Studienblatt: A 066 922

Studienrichtung lt. Studienblatt: Magisterstudium Informatikmanagement

Betreuer: Univ.-Prof. Dipl.-Ing. DDr. Gerald Quirchmayr

Declaration

Schriftliche Versicherung

Ich versichere, dass ich die Masterarbeit selbstständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfe bedient habe. Diese Arbeit wurde bisher weder in gleicher noch in ähnlicher Form einer Prüfungsbehörde vorgelegt.

Guntramsdorf, am 1/5/2013

Acknowledgements

Life has given me some wonderful opportunities over the years, one of which was the chance to return to university to study Information Management. It has enriched my life and brought me a multitude of new friends, ideas and insights; writing this thesis has been an enriching and at times enjoyable task.

I would like to thank my Thesis Advisor Prof. Dr Quirchmayr for his support and guidance. His time and advice has been a great support. His experience and insight on a broad range of topics make working with him a pleasure.

I wish to express my gratitude to the Target Company and special thanks to those who allowed and supported this case study. All my colleagues were patient and wonderfully supportive, without your responses this thesis could not be written.

Thank you all, it has been a privilege to work with you.

Best wishes for now and the future

Margaret

Abstract

Failing IT development projects need fast solutions. Short term improvement can be reached by focussing on a few critical factors. The project management is challenged to find and monitor the applicable success factors for the current context. In this thesis, I applied quantitative research methodology to gather the knowledge of the project team members about critical success factors. The methodology shows that questionnaire data can provide a valid light weight method for ranking ordinal information such as project success. This method was demonstrated using the research and development division of the Target Company, a large scale traffic solutions provider. The results show that it is possible to obtain a statistically reliable result ranking success factors in a project. From this ranking a recommendation was made as to the determinants of project success in the given environment as well as the currently critical success factors. It is possible to formulate the knowledge of the development team about the factors of project success into a statistically meaningful result within the time frame available to an IT development project.

Zusammenfassung

IT Entwicklungsprojekte, welche Probleme verursachen, benötigen innerhalb des vorgegebenen Zeitraums eine Lösung. Durch den gezielten Fokus auf kritische Erfolgsfaktoren kann der Projekterfolg vergrößert werden. Die Herausforderung ist das Erkennen der kritischen Faktoren. Diese Magisterarbeit stellt eine Methode vor, mit welcher man die Kenntnisse der Projektmitarbeiter über die Erfolgsfaktoren sammeln und auszuwerten kann. In einem weiteren Schritt wird das Verhältnis zwischen den Projekterfolgsfaktoren und den vier IT-Entwicklungsprozessmodellen, Agile, CMMI, SWEBOK, und V-Modell XT untersucht.

Die Ergebnisse zeigten, dass die Entwicklungsmannschaft und die Literatur sich einig waren, dass die Faktoren "Spezifikationen und Anforderungen" als auch "Realistischer Zeitplan" kritisch für den Projekterfolg sind. Diese Faktoren kommen als Ziele in den IT-Entwicklungsprozessmodellen vor, aber von der erforschten Liste von Erfolgsfaktoren waren jene, die sich auf die Fachkenntnisse oder Eigenschaften von den Projektmitgliedern beziehen, in den Entwicklungsprozessmodellen nicht vertreten. Die Notwendigkeit von Faktoren wie "motivierte und überzeugte Mitarbeiter", "Fachkenntnis über IT-Entwicklung" oder "Kenntnisse über das Anwendungsgebiet" werden in der IT-Entwicklungsprozess Literatur nicht erwähnt.

Mit der vorgestellten Methode ist es möglich, in kurzer Zeit statistisch relevante Aussagen über die momentan kritischen Faktoren in einem laufenden Projekt zu formulieren. Des weiteren wurde erforscht, wie die kurzfristigen Ziele der Projektverbesserung mit den langfristigen von der IT-Entwicklungsprozessverbesserung übereinstimmen.

Contents

1. Introduction & Motivation

IT projects fail: optimists state a quarter fail, pessimists estimate a third (Table 1-1) of all IT development projects fail completely, even more are compromised.(Ambler 2010; Standish Group International 2010). No one is denies that the failure rate is inacceptable. Classic IT development process models measure change in years, whereas typical IT projects are measured in months, thus the time span to needed to improve the IT development process is longer than the usual project duration. To work within the time span of a development project, a light weight method with a realistic time frame is needed. This thesis introduces a light weight method for identifying the factors driving project success, and presents a case study demonstrating the method.

Success factors in project management were researched by D. Ronald Daniel in the early 1960's. Though success factors do not predate the computer, they were not created for IT projects. Initially success factors were monitored with as much detail as possible; however the speed of a typewriter and interoffice mail limited the number of success factor update reports that could hit a manager's desk. Enterprise Management IT-Systems were introduced in the 1970's and by 1979 it was apparent that, with IT support, project managers had more reports on success factors than they could process, leading to a finer grading of success factors, the "critical" success factors:

> "... for any business, the limited number of areas whose results, if they are satisfactory, will ensure successful competitive performance for the organization. They are the few key areas where 'things must go right' for the business to flourish. If results in these areas are not adequate, the organization's efforts for the period will be less than desired". (Rockart 1979, p. 6)

Rockart proposed that managers make decisions based on a limited number of **critical** success factors, those deemed to make or break project success. "… the underlying premise is that decisions made in this manner should be more effective because they are based on data that is specifically linked to the organization's success factors." (Caralli 2004, p. 9) Thus in 1979 debate was opened as to which factors are the critical success factors and how to find them.

Previous research delivered short lists of 28 (Fortune 2006) or 26 (Nasir 2011) critical success factors for IT development projects. (Nasir, Sahibuddin 2011; Fortune, White 2006). Whereas the title of Rockarts article in 1979 was "Chief Executives Define Their Own", this thesis uses the crowd knowledge of the development team as opposed to the individual executive to find the critical success factors for an IT development project. (Surowiecki 2004). A questionnaire based on the CSF lists was developed and presented to the research and development department of an Intelligent Traffic Solution (aka R&D Team). The section 'Methodology: CSF Questionnaire Design and Justification' shows that the questionnaire is valid and that a reproducible result can be obtained. The 'Results' section presents the findings of using this method for determining critical success factors on a running IT development project.

The first part of the literature review- 'The Critical Success Factor List for Software Development' is dedicated to finding the critical success factors for software development. However it is important to know how the critical success factors align with the goals of IT process improvement. The long term focus must be to cement good development processes so that future projects maintain best processes. Thus the second part of the literature review - CSF's and Development Process Models is devoted four major IT process and project models. The four chosen: V-Modell XT, CMMI, SWEBOK and Agile, are reviewed in light of how their process objectives coincide with the CSF's for software development projects.

The four software development models, V-Modell XT, CMMI, SWEBOK and Agile each have different approaches, all agree on the importance of having clear objectives. Clear objectives were shown to be a priority in the case study as well. However once the objective has been declared further action differs. V-Modell XT attempts to define complete requirement prior to development, which frequently delays development. V-Modell XT is document centric, the project lives on paper, and all activity is triggered by a document (1.4). V-Modell XT is not necessarily an anathema to AGILE, which focuses on having only the documentation necessary, however in practise is. CMMI focuses on the management process, the success of the project is guaranteed by first concentrating on managing the project execution and then continuing on to manage the process of project execution(1.4), this tends to lead to administrative overhead with large amounts of KPI's being collected. The success factor 'Effective monitoring and control' was negated by the research results as being a driving force for project success. The SWEBOK includes the PMBOK (Project Management Body of Knowledge) as its management guidelines. (SWEBOK - Overview; Ó Conchúir 2011). The PMBOK approach leads to a more generic description of what project management is, with general tenets applicable to a variety of projects (including construction), which leaves out concrete information on how to address CSF issues, it is too general to be able to be a concrete base for Development project decisions. Agile on the other hand is focused directly at IT development project. Agile seems to have risen from generations of poor requirements. It postulates that the requirements will not be known at the start and states clearly that at the end of each sprint they will be realigned with reality. Agile IT development processes leave major decisions are to the software development team. (Agile Alliance 2012). Not surprisingly the development team studied agreed that they are competent to make project decisions, this could however be a statement of reality. Any decision that has not been made before it reaches the development team will have to be reached before development can be completed. Thus the short term view dictated by critical success factors is well supported by AGILE project development. It was found that the more technical CSF issues are well aligned with the software development models. The clear definition of roles and responsibilities, the types of documents to produce, which metric to collect is clearly and sometimes pedantically described in the development process models reviewed. However the results of the case study clearly showed that communication and interpersonal factors are in the top three issues critical to project success. None of the development process models mention leadership, team communication or interpersonal issues as topics in the IT development process.

Table 1-1 Chaos or Ambler Project Success and Failure

(Ambler 2010; Nasir, Sahibuddin 2011, p. 2175)

Standish-	Succeeded (%)	Challenged (%)	Failed (%)
1994	16	53	31
2008	32	44	24
Ambler-			
2010	54	32	14

A case study focused on the Research and Development department of the Target Company to demonstrate the use of the CSF questionnaire as a light weight method for determining the current state of the IT development project. The questionnaire is based on the CSF list from (Nasir, Sahibuddin) in (2011) as explained in the ‘Questionnaire Design’ section. Whereas one person’s opinion would not be significant the polling of a relevant number of project members allows a statistically significant statement to be made (see Sample Population section). The results show that the top three factors in the current project are:

- Good Leadership at team level
- Trustful and cooperative working environment exists.
- Skilled and sufficient staff

The weakest three factors were:

- User or client is involved while developing
- Frozen requirements
- Risk management

The Target Company specializes in multi-lane free flow traffic solutions. These are large scale technology based solutions with no end user interfaces. The results confirmed the strength of the company in IT skills, domain knowledge and team commitment. The median response on project success is high; at 70%, which is significantly higher than either Standish or Amblers success rates (Table 1-1). The questionnaire results indicated that the IT process framework should be strengthened to further improve project success. Further details can be found in the ’Results’ section.

1.1 Motivation

There is a twofold motivation for doing this research. The first is the very obvious business case; the second is a human issue. Loss of profitability and efficiency is all too obvious with 20% of IT projects failing;

software projects are endangering business goals. The U.S. Department of Defence (CMMI) and the German Bundeswehr (V-Modell XT) agree that low quality IT-projects are a danger to national security interests (Ahern et al. 2008). The business case and national security interest in software process improvement has been amply established in both the academic as also the popular media and is taken as granted for this thesis. (M.C. Paulk 2009; Das V-Modell® - Home; Paulish, Carleton 1994)

There is a second motivation for an IT professional to want better project success rates. The author of this thesis worked on the project being researched as well as other IT-projects in the past decades, at least 20% of which failed and not more than half were completely successful. For the author of this paper as well as all other member of the teams working on failing projects it is a stressful, mentally and physically draining environment. Work conditions have an impact on both physical and mental health. Around half the time IT development teams work on compromised projects, thus at least half of the time they are under stress to perform better. Each one of these underperforming projects had a manager exerting pressure to be: better, cheaper, faster. Hence, any improvement in project success is major life quality gain for the IT development teams. (Chmiel 2009)

2. Research Approach

The literature review shows that that research on critical success factors generated lists with less than 30 factors (see section: The Critical Success Factor List for Software Development) and that the four researched software development process models take longer to implement than the time available to a IT-project (see CSF's and Development Process Models). The primary focus of this thesis is to investigate a light weight approach based on CSF's to gaining insight into the success factors of a current project, to prioritize process improvement by determining the factors important to the current project. A secondary issue is whether the CSF's are covered by the IT development process models.

The ranking of the CSF's for a current IT development project is a necessary to prioritize the CSF's. This is achieved by gathering and analysing the opinion of the IT development team working on a project. Based on quantitative research methodology a questionnaire was developed to answer following research questions:

RQ1: Given a list of CSF's relevant to IT development projects in general:

> RQ1a: How do the respondents rank the factors in their current project?
>
> RQ1b: What importance do they give the factors in an ideal project?
>
> RQ1c: Is there any difference in the ranking of CSF's due to education, experience or the role the person has in the application development process?

RQ2: From the perspective of the Target Company, what are the critical success factors for the current project?

RQ3: What are the determining factors of project success?

The second issue of the thesis – whether the CSF's can be related to the IT development process is covered by:

RQ4: Is the ranking of the critical success factors related to the application development process being followed? Can the results be related to any particular application development process?

Thus this thesis examines methods for improving project success without losing sight of more important the long term goal of process improvement. It should provide insight into how to deal with the urgent: the project without ignoring the important: the process. The long term goal of permanent process improvement must not be ignored when striving for short term project success. The second section of the literature section (CSF's and Development Process Models) attempts to find shared ground between the success factors and development process objectives. To back up information from the literature review the questionnaire gathered information about the IT development process being used, the results are presented under the heading: 'Development Process'.

To summarize, the thesis investigates a light weight method for improving the success of IT development projects. It does so by obtaining quantitative data from the project team via a questionnaire. The basis for the questionnaire is the found in the 'Literature Survey', the methodological basis is explored in 'Methodology:

CSF Questionnaire Design and Justification'. The approach was then demonstrated using the IT development team of the researched organization, the 'Results' section show that though the data collected was in the most part statically valid, thus the approach was demonstrated to be a plausible method for determining critical success factors for running IT development projects.

3. Literature Survey

To answer RQ1 and RQ2 a significant number of responses are needed to obtain valid results. A first attempt at determining CSFs was made using open answer questions[1] about the causes of project success. It failed, the answers were too diverse for a light weight approach, the time needed to answer a long answer question and resources needed to evaluate the results were too much for the approach to be 'light weight'. A method was found which decreased the time needed to respond, as well as the time needed to evaluate the questionnaire. This method is based on the two most recent summaries (2006) and (2011) of research papers on CSFs in IT development projects. The low number of factors listed suggested using them as items in a direct answer questionnaire. Section 3.1 presents research papers on critical success factors for IT development projects.

To answer RQ3 a connection between CSFs and project success has to be established. The individual success factors used for RQ1 and RQ2 are not independent variables; there is considerable interrelation between factors. Independence is a requirement for finding determinant relations. The two summaries (2006) and (2011) proposed project attribute groups, of which only Nasir's proved reliable. The section 'Project Attribute Groups and Project Success' sketches the mapping of CSF's to project attribute groups. (2011) grouping of CSF's into three attribute groups People-related factors (7 critical success factors), Process-related factors (16 critical success factors), and Technical-related factors (3 critical success factors) was shown to a statistically reliable mapping. The results showed that for the current IT project the process related factors were the determinants of project success.

RQ4 seeks the connection between the CSFs and the IT development process. The literature on four IT development process models is reviewed is the section 'CSF's and Development Process Models'.

3.1 The Critical Success Factor List for Software Development

Although the purpose of this section is to find a CSF list, the definitive CSF list for all software application projects may not exist: "There does not appear to be a consensus of opinion among researchers and authors on the criteria for judging project success and the factors that influence that success." (Wateridge 1995, p. 171) The problem seems to lie deeper than just finding the critical success factors; the problem starts with the definition of project success or rather the lack of a definition for project success. Project success is not the same for all software development projects: "Project managers are putting far too much emphasis on the time and budget aspects for judging project success at the expense of other criteria, leading to a number of project failures." (Wateridge 1995, p. 171) In a similar vein the following year, Belassi and Turkel pointed out that the definition of project success is dependent on who is judging success. "A project which is considered to be a success by the client might be considered a failure by top management, if the project outcome does not meet top management specifications, even though it might satisfy the client. In this case, both of these parties are evaluating project success differently and thus they value the outcome differently..." (Belassi, Tukel 1996, pp. 141–142) Thus the critical success factors for software development will be shown to be finite in number, but the correct CSF list depends on the individual IT development project definition of success.

Despite the lack of consensus on what project success is, people are able to decide whether a project is successful or not. Managers are comfortable with the critical success factor method. It is an easily accessible platform for basing management decisions on, and therefore considerable literature has been written about CSF Factors for software development as well as other areas.

[1] An open answer question is one where the respondent writes her own answer to the question, a directed response question offers a number of responses and the respondent selects from the list of options.

3.1.1 Lists and Groups

As Waterman expected, three literature reviews on CSF's (Belassi, Tukel 1996; Fortune, White 2006; Nasir, Sahibuddin 2011) came up with three different critical success factor lists. Two of these papers are focussed in this section. In 2006 Fortune and White (F&W) reviewed and categorized 67 research reports. (Fortune, White 2006) and in 2011 Nasir and Sahibuddin(N&S)) reviewed and categorized 43 documents on CSFs and conducted several interviews.(Nasir, Sahibuddin 2011) Nasir and Sahibuddin created a convenient table comparing their findings to Fortune and White's, it is included in this thesis as Table 3-1 Comparison of extracted CSF Lists. Nasir and Sahibuddin sorted their findings by the number of mentions in the literature reviewed. In the 43 articles reviewed by Nasir and Sahibuddin, 27 factors were found. The top 5 factors were found in more than half of the articles.

These factors are:

1. Clear requirements and specifications:60%
2. Clear objectives and goals: 56%
3. Realistic schedule: 54%
4. Effective project management skills/ methodologies (project manager): 53%
5. Support from top management: 52%

According to N&S, the benefit of this list is, "There is a good opportunity, if an organisation or project manager is attentive, to control the top five critical factors to drive towards project success since the percentage of frequency of occurrences for each is more than 50%." (Nasir, Sahibuddin 2011, p. 2181) Note however that the top CSF's were only being cited in 60% of the literature. This is an absolute majority, but far from total consensus.

Fortune and White found only one factor "Top Management Support" (61%) that occurred in more than 50% of the articles. They themselves commented that: "although 81% of the publications include at least one of these three (top) factors, only 17% cite all three." (Fortune, White 2006, p. 54) Their top five were

1. Support from top management: 61%
2. Clear objectives and goals: 49%
3. Up-to-date progress reporting: 46%
4. Effective communication and feedback: 43%
5. User/client involvement: 38%

The ability to control project success by paying attention to the top five CSF, as recommended by N&S, is not indicated by the data collected by Fortune and White, as there is no longer in excess 50% support for the top five.

N&S's list of critical success factors had 26 items Fortune and White had 28. The goal is to obtain a list of 5 to maximum 10 factors that will be indicative of project success. A 26 item list is small enough to be presented to a large number of people to gather their opinion which was then done in the questionnaire. A list of factors was created, derived from N&S's list moderately tailored to the environment in which the questions were asked.

Table 3-1 Comparison of extracted CSF Lists

	Critical success factors (Nasir, Sahibuddin 2011, p. 2182)	Nasir&Sahibuddin (n=43)			Fortune&White (n=63)		
		Freq	%	Rank	Freq	%	Rank
1	Clear requirements and specifications	26	60.5	1	0	0	0
2	Clear objectives and goals	24	55.8	2	31	49.2	2
3	Realistic schedule	23	53.5	3	14	22.2	13
4	Effective project management skills/ methodologies (project manager)	23	53.5	4	19	30.2	8
5	Support from top management	22	51.2	5	39	61.9	1
6	User/client involvement	20	46.5	6	24	38.1	5
7	Effective communication and feedback	20	44.2	7	27	42.9	4
8	Realistic budget	19	46.5	8	0	0	0
9	Skilled and sufficient staffs	18	41.9	9	20	31.7	6
10	Frozen requirement	17	39.5	10	0	0	0
11	Familiarity with technology/ development methodology	15	34.9	11	14	22.2	12
12	Proper planning	15	34.9	12	0	0	0
13	Appropriate development processes / methodologies (process)	14	32.6	13	6	9.5	23
14	Up-to-date progress reporting	12	27.9	14	29	46	3
15	Effective monitoring and control	12	27.9	15	12	19	16
16	Adequate resources	11	25.6	16	16	25.4	10
17	Good leadership	11	25.6	17	15	23.8	11
18	Risk management	10	23.3	18	13	20.6	14
19	Complexity, project size, duration, number of organisations involved	10	23.3	19	4	6.3	26
20	Effective change and configuration management	10	20.9	20	19	30.2	7
21	Supporting tools and good infrastructure	9	23.3	21	0	0	0
22	Committed and motivated team	9	20.9	22	0	0	0
23	Good quality management	9	20.9	23	0	0	0
24	Clear assignment of roles and responsibilities	7	16.3	24	0	0	0
25	Good performance by vendors/contractors/ consultants	4	9.3	25	10	15.9	19
26	End-user literacy, knowledge and skills to use the system	2	4.7	26	7	11.1	21
27	Strong business case/ sound basis for project	0	0	0	16	25.4	9
28	Project sponsor/champion	0	0	0	12	19	15
29	Adequate budget	0	0	0	11	17.5	17
30	Organisational adaptation/culture/ structure	0	0	0	10	15.9	18
31	Planned close down /review /acceptance of possible failure	0	0	0	9	14.3	20
32	Political stability	0	0	0	6	9.5	22
33	Environmental influences	0	0	0	6	9.5	24
34	Past experience (learning from)	0	0	0	5	7.9	25
35	Different viewpoints (appreciating)	0	0	0	3	4.8	27

3.1.2 Project Attribute Groups and Project Success

This section goes deeper into what is needed to answer RQ3: What are the determining factors of project success? To determine project success makes it the resultant or dependant variable of a function whose outcome should be maximized. The individual critical success factors are not useful as arguments, or independent variables for determining project success. Larsen and Myers formulated the problems with having a list of almost 30 different factors as:

> "..the factor approach ignores the potential for a factor to have varying levels of importance at different stages in the implementation process. There is a lack of consistency in the research. Very few factors are shown to be important across multiple studies. It assumes that each factor is an independent variable and overlooks the interaction between them." (Larsen, Myers 1999, p. 398)

To reduce the interaction between variables (Larsen's final point), it is necessary to "reduce the number of project variables to those whose behaviour is most in tune with the behaviour of the aggregate success variable. I. e., if one of them increases the other does as well; if one of them decreases, the other does, too." (Wohlin, Andrews 2003, p. 289) Wohlin and Andrews present a general model based on Principle Component Analysis (PCA) for grouping the factors into a smaller number of independent variables that can be directly related to project success. PCA's goal is to calculate the weighting of each factors' contribution to the success variables (project attribute groups). Their work clearly supports the need for aggregating the success factors to a smaller number of reliable variables, however section 3.1.1 showed that there was no consensus about CSF's. "There are some limitations to using PCA. The analysis requires three or more interval variables because it is defined as a linear model …When applying PCA to non-interval variables, it is necessary to be a little cautious. This may, for example, include not accepting the results of the analysis without making sure that the results correspond to intuitive expectations." (Wohlin, Andrews 2001, p. 48) As the limitations could not be proven to be met, this method is not pursued further in this thesis.

3.1.2.1 Fortune and White

Fortune and White put forward the Formal System Model (FSM) to deal with Larsen's criticism that the CSF's are not independent variables and that there is interaction between them. The FSM defines and expects the interaction between the variables. The variables with strong interaction are mapped to the same component of FSM.

FSM was developed as a generic model for large scale industrial projects and processes. It's adaption to IT-Projects is well described in (White, Fortune 2009). "The model can be regarded as an "ideal-type", representing a robust system that is capable of purposeful activity without failure, so that a situation such as a construction project or an IT project that mapped onto it exactly, would also be capable of purposeful activity (that is activity involving decision-making) without failure." (White, Fortune 2009) Further details about the Formal System Model in general can be found in (Peters, Fortune 2003) and its use with IT-Projects in (Fortune, White 2006).

Fortune and White use the FSM because it provides a method where the relevant critical success factors can be integrated into a model representing the complete project process. The places where the CSF's contribute or feed information into the FSM are shown by the mapping of CSF's to FSM Components as displayed in Table 3-2. The critical success factors indicate how well or poorly each component of the FSM is performing. The system performance as a whole is equated to the value given to project success.

In Fortune and White mapping of CSF's to FSM components, there are four CSF's found in the CSF literature that could not be mapped. These are listed at the bottom of Table 3-2. CSF's that were ranked in 5th and 7th place of 27 factors are not relevant when using the FSM. From a software development viewpoint these CSF's are related strongly software process objectives. Effective change management is a critical issue in most process models, V-Modell XT in particular where documenting the change, logging changes and having accountability is a major focus. The CSF - "User/client involvement" is part of the Agile Manifesto (Beck et al. 2001a) For CMMI having a project sponsor is a key factor. (Kraft TM 2010, p. 4) Thus the mapping to FSM components has not included key critical success factors in its software development model.

Table 3-2 Critical success factors mapped onto components of the Formal System Model

Component of FSM/ Project Attribute	**Critical Success Factor from literature**
Goals and objectives	Clear realistic objectives Strong business case/sound basis for project
Performance monitoring	Effective monitoring/control Planned close down/review/acceptance of possible failure
Decision-maker(s)	Support from senior management Competent project manager Strong/detailed plan kept up to date Realistic schedule Good leadership Correct choice/past experience of project management methodology/tools
Transformations	Skilled/suitably qualified/sufficient staff/team
Communication	Good communication/feedback
Environment	Political stability Environmental influences Past experience (learning from) Organisational adaptation/culture/structure
Boundaries	Project size/level of complexity/number of people involved/duration
Resources	Adequate budget Sufficient/well allocated resources Training provision Proven/familiar technology Good performance by suppliers/contractors/consultants
Continuity	Risks addressed/assessed/managed
Not in Model	*User/client involvement* *Different viewpoints (appreciating)* *Project sponsor/champion* *Effective change management*

(Fortune, White 2006, p. 57)

3.1.2.2 Nasir and Sahibuddin

Nasir and Sahibuddin chose a simpler grouping of critical success factors based on work by Zahran. (Zahran 1997) They grouped the critical success factors into 3 groups, the technical factors, process factors and people factors (Zahran 1997) as is shown in table below (Table 3-3).

Table 3-3 Critical Success Factors based on categories

Attribute Group	Nasir&Sahibuddin - Critical success factors of software project	#
People-related factors (7 factors)	Effective project management skills/ methodologies (project manager)	4
	Support from top management	5
	User/client involvement	6
	Skilled and sufficient staffs	9
	Good leadership	17
	Committed and motivated team	22
	Good performance by vendors/contractors/ consultants	25
Process-related factors (16 factors)	Clear requirements and specifications	1
	Clear objectives and goals	2
	Realistic schedule	3
	Effective communication and feedback	7
	Realistic budget	8
	Frozen requirement	10
	Proper planning	12
	Appropriate development processes / methodologies (process)	13
	Up-to-date progress reporting	14
	Effective monitoring and control	15
	Adequate resources	16
	Risk management	18
	Effective change and configuration management	20
	Good quality management	23
	Clear assignment of roles and responsibilities	24
	End-user literacy, knowledge and skills to use the system	26
Technical-related factors (3 factors)	Familiarity with technology/ development methodology	11
	Complexity, project size, duration, number of organisations involved	19
	Supporting tools and good infrastructure	21

There is an imbalance in CSF's in the different categories. 62% are people related factors, 27% are process related issues and a mere 12% are technology. This is a clear reflection that the critical success factors reported in the literature are not evenly distributed across multiple groups. The critical problems are not found in the technology. The major problems of software development are more likely to be sociological than technological. The success or failure of a project is seldom due to technical issues....... If a project does go down the tubes it will be non-technical, human interaction problems that do it in." (DeMarco 1988, p. 59) Nasir and Sahibuddin were writing in 2011, thus the in the intervening 23 years, almost a quarter of a century, the problem is still with humans – not machines

3.2 CSF’s and Development Process Models

The CSF list focusses on short term goals, in an ideal world, these short term measures are aligned with the long term development process objectives. This section compares the CSF list gained from the literature in the previous section 0 with the long term objectives of the software development process models. Table 3-4 CSF’s linked to Software Development Process associates the CSF’s with the objectives stated in software development process specifications. Here the short term goal, to optimize one area now, would coincide with the stated long term objectives of development process improvement.

Table 3-4 CSF’s linked to Software Development Process

#	**Factor**	**V-Modell XT**	**CMMI**	**SWEBOK**	**Agile**
1	Adequate resources, rooms, hardware, test labs			**x**	**x**
2	Appropriate formalized development processes and methodologies	x	**x**	**x**	
3	Clear assignment of roles and responsibilities	x	**x**	**x**	
4	Clear known objectives and goals	x	**x**	**x**	**x**
5	Clear requirements or specifications	x	**x**	**x**	
6	Committed and motivated project team				**x**
7	Domain knowledge(for example of road traffic control)				**x**
8	Change and configuration management	x	**x**		
9	Effective communication and feedback				**x**
10	Effective monitoring and control	x		**x**	
11	Effective management skills at team or project level		**x**	**x**	
12	Experience with the programming languages, technology and hardware				**x**
13	Frozen requirements	x		**x**	
14	Good quality management			**x**	
15	Good Leadership at team level				
16	Good Leadership at project level				
17	Low staff turnover, stable teams are built				**x**
18	Proper planning	x			
19	Realistic schedule	x	**x**	**x**	**x**
20	Risk management	x	**x**	**x**	
21	Skilled and sufficient staff				
22	Support from top management		**x**	**x**	
23	Supporting tools – Project management and SW development tools				
24	Trustful and cooperative working environment exists.				**x**
25	Up-to-date progress reporting	x	**x**	**x**	
26	User or client is involved while developing				**x**

3.2.1 V-Modell XT

From the objectives the V-Modell XT wants to attain, the success factors can be derived. The V-Modell XT is a document-centred process model, there is no activity taken which is not covered by the appropriate document. The objectives of the V-Modell XT as stated in (BMI 2012, pp. 2.3) are:

1. minimization of project risks
2. improvement and guarantee of quality
3. reduction of total cost over the entire project and system life cycle
4. improvement of communication between all stakeholders

This section analyses the V-Modell XT documentation looking at the objectives and thus the problems that V-Modell XT attempts solve.

3.2.1.1 V-Modell XT Objectives - Context

History and Funding

The original V-Modell was developed by the German Ministry of Defence (Bundesministerium für Verteidigung -BMVg). Development was started in 1986; in 1991 it was mandated for use in all military software projects. The V-Modell XT was taken over by the Ministry of the Interior (Bundesministerium des Inneren) who published it in 1992 as the Standard for all government projects. The latest version 1.4 was published in May of this year (2012). The XT in V-Modell XT stands for "eXtreme Tailoring" to denote the need and possibility to adapt the model to the project's needs. The need to tailor the standard to the projects needs was present from the first models, but the addition to the name focussed attention in this area.

Target Group for V-Modell XT

> *V-Modell is targeted at all participants in development projects in particular information systems development. As a process model for project management it is written for executives and project leaders actively guiding a development projects. „The V-Modell supports the handling of projects in enterprises, public and military agencies, and authorities and agencies of the Bundeswehr."(BMI 2012, §§ 2.5)*

Goal

The goal of the V-Modell XT is to have a single process for defining and executing a project. This single process should clear clarify three major areas; what has to be done, how the tasks should be done, and what technology is to be used. The V-Modell XT standardizes the communication between the customer and the producer of software applications, as well as amongst the participants of the software development process, in the expectation that this will improve quality by minimizing misinformation. The V-Modell XT specifies exactly the documents and the contents of the documents for all areas of the application lifecycle; in describing the document's content, the actual items produced are also defined.

The V-Modell XT objectives were listed in the introduction. None of the objectives listed are exclusive to the software industry; all would apply just as well to construction projects, however in stating the four areas as objectives, they have implicitly stated what the success or failure factors have been in past projects. The next sub-sections take a detailed look at the four objectives listed above. The subsection headings are taken directly from the objective in chapter 2 of the V-Modell XT 1.4 and are preceded by a direct quote from the V-model, describing what was to be achieved with the given objective.

3.2.1.2 **Goal – Standardization of Planning Processes and Roles,**

> *The V-Modell improves project transparency and project control by specifying standardized approaches and describing the corresponding results and responsible roles. It permits an early recognition of planning deviations and risks and improves process management, thus reducing the project risk. (pp. 2.3)*

The V-Modell XT minimizes project risk by standardizing planning processes. Projects use the same management and development processes, they are staffed similarly. The roles of the project members fulfil are predefined, the documents produced and the results expected are also agreed upon in advance. The V-Modell describes standardized management processes for all project steps so that efficiency of project execution is given by having predefined guidelines for all project related actions. The steps are not reinvented or renegotiated with each new project. Thus planning omissions or misunderstandings are prevented by project members being able to rely on previous experience with the development process being used. Though the product changes with each project, the production process stays constant.

The keys to a correctly run V-Modell process are the documents which are presented at predefined stages. The stages and the documents associated with the stage are seen as the standard interfaces for information interchange. Thus for V-Modell XT communication takes place with documents. The itemizing of all artefacts needed for a development stage generates transparency and the quality of the application developed is increased.

The standardized document are supported by team members have standardized roles and responsibilities. The process steps are executed by teams or individuals fulfilling carefully defined roles "Part-4: V-Model-Reference Roles" (1.4, §§ 4.0) Risk is minimized with standard roles. Each role is assigned to a specific person or group during the start-up phase of the application lifecycle. Each role definition includes lists of obligations and responsibilities. The possible roles available are quite extensive; there are 33 in total, from account manager to user. Some roles overlap and are performed by a single person in the project. The Project Leader might also be the Data Protection Manager. Many of the roles will not be present in a project at all. The list also includes little used roles such as coach and ergo metric designer, it is not intended that all roles are used in every project, the XT in the model name stands for eXtreme Tailoring. The roles not needed are removed from the model before it is applied**.**

Similarly detailed is the task list for each role. These roles and tasks are tailored to the organization or specific task. This adaptation to the task at hand is the 'tailoring' explicitly mentioned the V-Modell XT name. In addition the role also contains the responsibilities of the role. There is a clear definition of who is accountable for a work product and liable for the decisions made there in. There is a personal responsibility for the deliverables as part of role definition. By performing rigid standardization and definition of roles and processes, V-Modell XT expects that planning mistakes (deviations) will be avoided and project success improved. This rigidity is often seen as the downfall of the model. By carefully defining interfaces and responsibilities it creates a large formalized administrative overhead. By defining what each person is responsible for, it implicitly defines that they are not responsible for everything else. Solely the Project Leader is responsible for the big picture; all other team members can correctly claim it is not their job. There is responsibility for the complete project resting with all other team members.

The objective minimization of project risk through standardization of processes is supported by the success factor: adherence to the process. Roles must be known, responsibilities accepted, and issues documented. Rigid process standardization minimizes risk and is therefore a factor in project success.

3.2.1.3 **Objective: Completeness and Quality**

> *"As a standardized process model, the V-Modell ensures that the results to be provided are complete and have the desired quality. Defined interim results can be checked at an early stage. Uniform product contents will improve readability, understandability and verifiability". (pp. 2.3)*

The completeness of specification and high quality of production seen in large scale manufacturing, such as car manufacturing, not been nearly achieved in the single items produced in the IT development. Unlike large scale manufacturing where masses of identical items are being produced, in software there is only a single item produced. In software each project can however be seen as a produced item, thus the standardization takes place in the planning process, not the production. The V-Modell introduces stringent standardization in the management process in an attempt at attaining the quality and completeness that traditional manufacturing obtains in production. It is an abstraction level (application lifecycle management) and not the product that is being perfected. The objective states the process should ensure that the results to be provided are complete. Product quality is a subjective value; however product completeness should be objective; however in software complete is not necessarily well defined. In addition an objective is improving readability, understandability and verifiability. The mention of these values allows the deduction that, at least in some projects, content is being delivered that is illegible, unintelligible or unverified. A rigid standardization of the software development process is sought to avoid these difficulties.

By creating a uniform process for IT development projects the completeness and quality of the product is improved. Having defined interim results and partial deliveries could catch problems at an early stage.

This object indicates that following factors are indicators of successful software projects:

- Standard management processes will improve the readability, understandability and verifiability of the result.
- A successful project should have incremental deliveries, to catch definition and quality problems early on.

3.2.1.4 **Objective: Reduction of Total Cost over the System Life Cycle**

> *The effort for the development, production, operation and maintenance of a system can be calculated, estimated and controlled in a transparent manner by applying a standardized process model. The results obtained are uniform and easily be retraced. This reduces the acquirer's dependency on the supplier and the effort for subsequent activities and projects. (pp. 2.3)*

The V-Modell XT refers to an 'acquirer'. Having non-standard processes increases the likelihood of vendor lock-in. The cost of switching to a different supplier is prohibitive as the documentation of the product acquired is insufficient to determine what costs will be occurred. The cost of entry to the market is too high for other providers to be competitive. The initial provider has created a monopoly market by concealing needed required information for further cost estimation.

The use case presented in this document is of a company producing its own software, thus vendor lock-in cannot exist in the classic sense, as all software is developed in-house. Nonetheless, the internal cost of maintaining and extending product functionality exists regardless of whether it is external or internal to the organization. The knowledge of how a product functions and what the cost of change is will last only as long as the original project team is available. The cost will increase as team members leave the company or just forget development details over time.

The critical issue is that a standard documentation process is used throughout application lifecycle so that information is concentrated in one spot, is complete and up to date. Effort estimation should be based on this information and formulated in a transparent, reproducible fashion.

3.2.1.5 **Objective: Clarity of Requirements, formalized communication**

The V-Modell XT standard states: *"The standardized and uniform description of all relevant elements and terms is the basis for the mutual understanding between all stakeholders. Thus, the frictional loss between user, acquirer, supplier and developer is reduced." (pp. 2.3)*

Frictional loss (German: Reibungsverluste) denotes the power lost overcoming the friction between two moving surfaces, in the context used above, it refers to information which is misunderstood, lost in transmission or in some form not communicated. The V-Modell XT attempts to overcome information loss by formalizing communication. Written communication acknowledged by both sides attempts to clarify the information transfer. Documents are used not only between the acquirer and supplier, but also within the suppliers' project team. Each software development step is accompanied by documentation officially released by the responsible person for the project phase. .

The process aimed for is a formalized documentation linkage not a face to face meeting. Thus the critical success factors from literature list are those related to formalized interactions and requirements. By having thought of all eventualities in advanced and clearly designated who is responsible for what results risk is minimized, where by project failure is seen to be a direct result of a project risk.

3.2.1.6 V-Model XT CSF's

As the previous sections have shown V-Modell XT seeks to improved project success by minimizing the risk of a poor quality outcome. By controlling the production process the model seeks to control what is produce. The amount of overhead produced in documentation indicates that for the German army that within the iron triangle of scope, time, and budget, erring in favour of scope is preferred. The final factors list is presented in the following table.

Table 3-5 V-Model XT Objectives and CSF's

#	**Factor**
2	Appropriate formalized development processes and methodologies
3	Clear assignment of roles and responsibilities
4	Clear known objectives and goals
5	Clear requirements or specifications
8	Change and configuration management
10	Effective monitoring and control
13	Frozen requirements
18	Proper planning
19	Realistic schedule
20	Risk management
25	Up-to-date progress reporting

V-Modell XT does not mention any form of soft skills or environmental issues. It is not seen as a topic for this model. It describes artefacts and responsibilities, how people fulfil their role or meet the responsibilities delegated to them is not part of this standard. The critical success factors related with the V-Modell XT are very concrete, well defined issues.

3.2.2 CMMI

The Software Engineering Institute (SEI) has a clear statement on what the critical success factor is:

> “*What holds everything together? It is the processes used in your organization. Processes allow you to align the way you do business. They allow you to address scalability and provide a way to incorporate knowledge of how to do things better. Processes allow you to leverage your resources and to examine business trends.*
>
> *This is not to say that people and technology are not important. We are living in a world where technology is changing at an incredible speed. Similarly, people typically work for many companies throughout their careers. We live in a dynamic world. A focus on process provides the infrastructure and stability necessary to deal with an ever-changing world and to maximize the productivity of people and the use of technology to be competitive. (CMMI Product Team 2011, p. 4)*

This section analyses the CMMI-Dev standards and related documents looking at the objective, focus, and thus the problems that CMMI attempts to solve.

3.2.2.1 CMMI Objectives - Context

History and Funding

CMMI was initiated by the American Department of Defence at Carnegie Mellon University; in a sense CMMI is an US parallel to V-Modell XT. Both the Americans and Germans identified poor software quality as a national security threat. Carnegie Mellon founded the Software Engineering Institute (SEI) with the goal of improving engineering processes.

Process is everything to CMMI, establishing the correct application lifecycle management process is the key to successful application development.

The first standard produced by the SEI in process improvement was the Software CMM (Capability Maturity Model) released in 1991. The Capability Maturity Model Integrated (CMMI) followed a decade later; version 1.0 was published in 2001 with 1.1 following in 2002, the current version 1.3 was released in October of 2010. Around half of CMMI certification is occurring outside of the United States. For the past 2 decades the SEI support CMMI as a research project. Fortunately CMMI has been a success economically, thus as of

2013 it will be moved from the SEI and government funding to its own specially created institute the "CMMI Institute". It will be a commercial self-sustaining business unit as opposed to a research institute. It will not totally forgo research; however, the CMMI Institute will no longer be receiving automatic funding from Carnegie Mellon University. "(The) CMMI Institute will be able to evolve CMMI in directions independent of the path it was on while within SEI." (Entinex Inc 2012, p. 1)

Target Group for CMMI

The target group for CMM was defence contracting. According to the CMMI introductory material, "CMMI is used by small and large companies alike in a variety of industries including electronics, health services, finance, insurance, and transportation." (CMMI | Overview) The opinion in the literature is that the process overhead, required by the method, clearly targets it at large, established businesses. The U.S. Defence department requires CMMI certification for its contractors, so these businesses will continue to be important stakeholders in CMMI. The right for an organization to bid for government software contracts is linked to achieving a minimum CMMI maturity level. Maturity levels are further defined in section 3.3.1.4. The amount of training and overhead usually needed to implement CMMI makes it less attractive to small and medium size businesses. "Variants for small organizations, research environments, and so on, were continually being touted, but few survived." (M.C. Paulk 2009, p. 15) The time needed to implement process improvement is a greater issue for small and medium sized businesses as so the time scale needed for improvement discourages many. In 2011 following values were reported: "For organizations that began their CMMI-based SCAMPI effort with V1.2 or later, the median time to move from

- maturity level 2 to 3 is 20 months
- maturity level 3 to 4 is 25 months
- maturity level 3 to 5 is 26 months" (Customer Relations SEI 2012, p. 25)

Goal

The CMMI objective is best stated in their own words:

> *"CMMI (Capability Maturity Model Integration) is a process improvement approach that provides organizations with the essential elements of effective processes, which will improve their performance. CMMI-based process improvement includes identifying your organization's process strengths and weaknesses and making process changes to turn weaknesses into strengths." (CMMI | Overview)*

Process improvement and quality management are the goals of CMMI. Poor quality is seen as leading to high overhead costs, a typical scenario for the defence industry.

There are currently three CMMI 'constellations', for Acquisition, for Development and for Services. The CMMI for Development constellation is applicable to the Target Company, thus the version referred to in this literature review is CMMI® for Development, Version 1.3 (CMMI-DEV) (CMMI Product Team 2011). In CMMI-DEV the more general term engineering and technical engineering is used and not software project, the standard explicitly states that the processes are valid for hardware as well as software projects. "Engineering process areas cover the development and maintenance activities that are shared across engineering disciplines. The Engineering process areas were written using general engineering terminology, so that any technical discipline involved in the product development process (e.g., software engineering, mechanical engineering) can use them for process improvement." (CMMI Product Team 2011, p. 46)

There are 22 process areas in CMMI-DEV, the 16 of the 22 are core process areas common to all CMMI models, of these only 6 process areas are development specific see Table 3-6. The goal is that a fully mature organization has all the 22 process areas implemented.

Table 3-6 CMMI Process Areas

Core Process Areas	Development process areas
1. Causal Analysis and Resolution (CAR) 2. Configuration Management (CM) 3. Decision Analysis and Resolution (DAR) 4. Integrated Project Management (IPM) 5. Measurement and Analysis (MA) 6. Organizational Process Definition (OPD) 7. Organizational Process Focus (OPF) 8. Organizational Performance Management (OPM) 9. Organizational Process Performance (OPP) 10. Organizational Training (OT) 11. Project Monitoring and Control (PMC) 12. Project Planning (PP) 13. Process and Product Quality Assurance (PPQA) 14. Quantitative Project Management (QPM) 15. Requirements Management (REQM) 16. Risk Management (RSKM)	1. Product Integration (PI) 2. Requirements Development (RD) 3. Supplier Agreement Management (SAM) 4. Technical Solution (TS) 5. Validation (VAL) 6. Verification (VER)

The CMMI Maturity denotes the levels attained on “an evolutionary path recommended for an organization” (CMMI Product Team 2011, pp. 21 /year only). The maturity level is a way station along this path to the self-optimizing organization, defined by specific process areas that have been implemented.

Table 3-7 shows which process areas are assigned specific maturity levels. The Initial Level has no processes assigned to it, as it is assumed everything proceeds in an ad-hoc fashion. At Level 2 – Managed it is assumed that planning is taking place, but there is no feedback or update of plans and no verification that what was planned actually happened. At Level-3 Defined a process has been defined to execute projects. Not just a project is being planned, but the process of planning a project has been planned. In addition the information of the current project is being used to further tune the current projects success

The maturity level has significant business value as only organizations assessed at or above a given maturity level are allowed to bid on government contracts.

CMMI How the Goal is achieved - Processes overview

MMI does not define actual steps or methods for the process areas. The technology of the process area is left for the organization to define. Each CMMI process area defines one or more activities or processes that must be done for the process area to be, the process areas does not specify how the process is done. Frequently the names of the process areas are identical or similar to the processes and methods they describe but the two should not be confused. Where V-Modell XT or Agile describe actual methods of performing work CMMI describes process areas where work must be done. Thus CMMI as a process model and Agile or V-Modell XT as development methods are complementary not mutually exclusive. There is a growing movement promoting the use of CMMI and Agile together in application development (Hillel Glazer 2008; Mary Ann Lapham 2012).

Table 3-7 Process Areas by Maturity Level

Maturity Level	**Process Area**	**Category**
Initial		
Managed	Project Monitoring and Control	Project Management
	Project Planning	Project Management
	Requirements Management	Project Management
	Supplier Agreement Management	Project Management
	Configuration Management	Support
	Measurement and Analysis	Support
	Process and Product Quality Assurance	Support
Defined	Product Integration	Engineering
	Requirements Development	Engineering
	Technical Solution	Engineering
	Validation	Engineering
	Verification	Engineering
	Organizational Process Definition	Process Management
	Organizational Process Focus	Process Management
	Organizational Training	Process Management
	Integrated Project Management	Project Management
	Risk Management	Project Management
	Decision Analysis and Resolution	Support
Measured	Organizational Process Performance	Process Management
	Quantitative Project Management	Project Management
Optimizing	Organizational Performance Management	Process Management
	Causal Analysis and Resolution	Support

CMMI implementation involves mapping CMMI process areas to specific processes be performed in a company. Optimally the processes needed to implement a given process area already exist already; the process is planned and performed in the organization so the formal documentation of an existing state suffices to fulfil the process area requirement. In level 1 organization processes are being performed, which means the work is being done, however the process is not planned in advance of doing. As CMMI maturity increases not only the active work must have a defined process definition, the process definition itself must have a defined process of improvement. The feedback from the active processes is used for process management as well as for process improvement. As the processes become aligned the metrics can be used to quantify the process management.

3.2.2.2 **Objective: Management Support**

> *Research has shown that the most powerful initial step to process improvement is to build organizational support through strong senior management sponsorship*. (CMMI Product Team 2011, p. 55)

CMMI clearly states that processes are the foundation of a successful organization and that senior management must be involved in establishing and maintaining them. "Activities performed by the senior management sponsor include but are not limited to the following:

- Influence the organization to adopt CMMI,
- Choose the best people to manage the process improvement effort
- Monitor the process improvement effort personally
- Be a visible advocate and spokesperson for the process improvement effort
- Ensure that adequate resources are available to enable the process improvement effort to be successful

Given sufficient senior management sponsorship, the next step is establishing a strong, technically competent process group that represents relevant stakeholders to guide process improvement efforts."(Ahern et al. 2008; Dymond 2007; CMMI Product Team 2011, p. 55)

"Although this might sound like a cliché, executive sponsorship and support are key drivers for a CMMI program." (Kraft TM 2010, p. 4) The time commitment to gain one CMMI maturity level will take 1.5 to 2.5 years on average. (Humphrey 2006; Customer Relations SEI 2012) A change in management focus over this time frame will end the effort. "Many organizations that start up a process program make the mistake of promising results in the first year." (Reifer Donald J. 2007, p. 25) The key CMMI success factor is 'Continual management support for implementing CMMI defined processes'.

3.2.2.3 **Objective: Planning**

CMMI-Dev process areas add tasks to the current the work process. This is extra work, the value of which value might not be apparent for some time. Gerald Weinberg (Weinberg 1997) argues the change involved in process improvement will make performance get worse before the intended improvements are seen. A UNISYS CMMI implementation reported that:

> "*The foundation for a CMMI program is a good process framework. The one that was put forth by the central team was an over-engineered process framework. There were times when "process bigots" took control and process became the end instead of the means."* (Kraft TM 2010, p. 4)

Philip Ardanuy went so far as to say in a presentation on CMMI that: "The inconvenience is on the developers because the developers do not really benefit. The benefit is downstream." (Ardanuy Philip E.) A key to success is to get all project members to actively support formalized processes. As the results section showed this has not occurred in the organization studied (see 5.7.1.) Process areas that exist only on paper will not exist in the long term.

In the projects this success factor is shown by having a defined process model that is being followed. This model has to be lived, not merely exist on paper.

3.2.2.4 Objective: Metrics

CMMI stresses the importance of metrics in several process areas. The Measurement and Analysis (MA) process area is listed as part of Maturity level 2 – Managed. Researchers are equally convinced that metrics are needed „measurement is part of process and product management and provides the information needed to make decisions that guide product development." (Hillel Glazer 2008, p. 11). Some literature is less encouraging. Ruzhi and Peiyao documented their "Research on CMMI-based Software Process Metrics"(Ruzhi Xu 2006), a model for collecting CMMI process area related statistics such as defect density, lines of code and other process relevant values. The desired LOC and achieved LOC differed by 3.5% in their test project, the recommendation to enforce unit tests and discover more defects in the early project phases will not surprise any project manager and does not justify the effort in collecting the data. (Ruzhi Xu 2006, pp. 395–396) Another group stress the time frame needed to get meaningful data: "After 8 years of data collection we are starting to be able to extract some quantitative indicators of the impact of becoming CMMI Maturity Level 3." (Hihn et al. 2011, p. 9), however, in the meantime: "While waiting for the long term quantitative indicators to become available, short term indicators are needed." (Hihn et al. 2011, p. 2) The process management would like to have quantitative values for measuring development progress; however meaningful metrics take time and honesty to collect. In the case of the 8 years reported by (Hihn et al. 2011) most projects will be close to retirement long before the time series was significant.

The significant factor to look for is the key performance values or indicators that are being collected. The information given by the respondents of the questionnaire showed no support for gathering this long term data (see section 5.2).

3.2.2.5 CMMI CSF's

CMMI does not describe a process for developing software; rather it describes a process for creating software processes. It is a method for improving and then measuring the process of creating an IT-application. Thus the short terms factors that influence this long term goal are:

Table 3-8 CSF's for CMMI

#	**Factor**
2	Appropriate formalized development processes and methodologies
3	Clear assignment of roles and responsibilities
4	Clear known objectives and goals
5	Clear requirements or specifications
8	Change and configuration management
11	Effective management skills at team or project level
19	Realistic schedule
20	Risk management
22	Support from top management
25	Up-to-date progress reporting

The people skills of project management are left out, the details of technological factors are left out.

3.2.3 SWEBOK / PMBOK Objectives and CSF's

"*The objectives of the Guide to the Software Engineering Body of Knowledge project are* to characterize the contents of the Software Engineering Body of Knowledge; provide a topical access to the Software Engineering Body of Knowledge; promote a consistent view of software engineering worldwide; clarify the place of, and set the boundary of , software engineering with respect to other disciplines such as computer science, project management, computer engineering and mathematics; provide a foundation for curriculum development and individual certification and licensing material." (Abran et al. 2004)

The SWEBOK standard has a much broader scope than application lifecycle management. This section analyses SWEBOK (Abran et al. 2004) and the PMBOK (Project Management Institute 2004) it is based on in light of the problems facing application lifecycle management and the factors indicating project success.

3.2.3.1 SWEBOK Context

History and Funding

The SWEBOK guide was first envisioned in 1997 by the IEEE Computer society. The goal was to place Software Engineering on a similar basis to other engineering disciplines, to make it a "legitimate engineering discipline and a recognized profession"(SWEBOK - Overview). The first trial version was published in 2001 (Abran, Moore 2001). The initial funding for the project was by ACM, US and Canadian Research councils as well as large business (Abran, Moore 2001, p. xiii). However before version 1 was published in 2004 the ACM had left the consortium. The purpose of the SWECC[2] was moving towards certification of software engineers. (Cupp 2001)

Target Group and Goal

"The objectives of the Guide to the Software Engineering Body of Knowledge project are to characterize the contents of the Software Engineering Body of Knowledge; provide a topical access to the Software Engineering Body of Knowledge; promote a consistent view of software engineering worldwide; clarify the place of, and set the boundary of, software engineering with respect to other disciplines such as computer science, project management, computer engineering and mathematics; provide a foundation for curriculum development and individual certification and licensing material." (Abran et al. 2004)

SWEBOK is divided into several knowledge areas most pertaining to the architecture and implementation of programs. However there is a Software Engineering Management Knowledge Area as well as Software Engineering Process Knowledge Areas focussed on the understanding project management and project success factors. The SWEBOK guide acknowledges that the majority of the skills needed managing software

[2] SWECC – Software Engineering Coordinating Committee

applications are not software specific, and therefore they, "suggest that the reader interested in project management beyond what is specific to software engineering projects consult the PMBOK (Process Management Body of Knowledge) itself" (Abran et al. 2004). The Project Management Body of Knowledge (PMBOK) (Ó Conchúir 2011) is published by the Project Management Institute (PMI). For PMBOK and SWEBOK, certification and certification trainings are important goals. Thus in reading both guides, SWEBOK and PMBOK, the reader should keep in mind that engineer or project manager certification is a goal, though not the only goal of the person writing. Issues that cannot become part of a certification course will not be given a large amount of space, thus leadership ability and soft skills are not mentioned as knowledge areas.

Process

PMBOK and SWEBOK espouse scientific management. The Merriam-Webster dictionary defines this as

“Planned management of production or other industrial or business activity that is based on the use of codified and verified knowledge of the knowable factors and directed toward the drawing up and carrying out of an overall plan accompanied by detailed instructions for each operation as established from time and motion study standards and research.” (Merriam-Webster 2011)

The focus is on detailed instructions for each operation. Measurement is an integral part of scientific management.

3.2.3.2 **Objective: Management Skills**

The project manager is identified as the key player dictating project success. The position is described does not include the maximum technical expertise; it is administrative and planning skills that define the manager.

"It requires flexibility, good judgment, strong leadership and negotiating skills, and a solid knowledge of project management practices. ... The project manager is in charge of all aspects of the project including, but not limited to:

- Developing the project management plan and all related component plans,
- Keeping the project on track in terms of schedule and budget,
- Identifying, monitoring, and responding to risk, and
- Providing accurate and timely reporting of project metrics." (Project Management Institute 2004, §§ 2.3)

From the description of the project manager’s responsibilities the project success can be seen to depend on effective measurement, planning with a focus on risk management and requirements.

3.2.3.3 **Objective: Metrics**

The definition of management as a science, as opposed to an art or craft, stresses the planning and details aspects of management. The SWEBOK states,

> *“The importance of measurement and its role in better management practices is widely acknowledged, and so its importance can only increase in the coming years. Effective measurement has become one of the cornerstones of organizational maturity” (Abran et al. 2004, §§ 8.6.1).*

The SWEBOK guide underlines it further with *"In essence, management without measurement, qualitative and quantitative, suggests a lack of rigor, and measurement without management suggests a lack of purpose or context." (Abran et al. 2004, §§ 8.0)*

From the viewpoint of the SWEBOK process model, effective measurement is the critical success factor. The importance of measurement is further emphasized by having resources, management and staff committed to the measurement process. Establishing and sustaining measurement commitment is a process point by itself, indicating that measurement commitment is not self-sustaining. In contrast there are no extra steps for maintaining team commitment. Key performance metrics are the driving force behind all decisions and planning.

SWEBOK treatment of metrics has been severely criticized by other software professionals, Cem Kaner states that: "The review of measurement theory in SWEBOK (page 174 and on) … provides a laundry list of metrics, including many that are invalid and unvalidated metrics — to the extent that it is clear what attribute they are actually intended to measure" (Kaner 2012, p. 1). As quality and productivity cannot be measured, secondary indicators are used. These secondary indicators are then used in the management planning process, without acknowledging that they may be a very imperfect representation of the values that should be measured.

3.2.3.4 **Objective: Requirements**

Software Requirements is a separate Knowledge Area in the SWEBOK, its contrast risk management has but a paragraph. It is broken into seven topics, Fundamentals, Process, Elicitation, Analysis, Specification, Validation and Practical Considerations. There is Extensive detail and further reading is offered. Specific steps are offered in how to perform tasks. It is interesting that though a detailed process is offered, it does not preclude implementing it in an agile fashion. "The majority of this chapter is compatible with agile methods concerning software requirements." (Norton 2003)

The planning process is based on the requirements; however, it is not assumed that the requirements are fully known at the beginning of the process. The SW Engineering Management Knowledge area starts with the statement about the problems particular to software projects:

"*The perception of clients is such that there is often a lack of appreciation for the complexity inherent in software engineering, particularly in relation to the impact of changing requirements. It is almost inevitable that the software engineering processes themselves will generate the need for new or changed client requirements." (Abran et al. 2004, §§ 8.1)*

Knowing what to do before doing is a universal success factor amongst all models, one could safely generalize to it being valid in all situations in life.

Objective: Planning

The PMBOK has 44 management processes which are to be followed throughout the project of which almost the half is planning activity. The PMBOK manager spends the majority of effort in planning. Zwikael and Globerson from Victoria University analysed the prevalence of planning in PMBOK and concluded:

"It identifies 21 planning processes, out of the 44 processes required to manage a project. That is to say, planning processes consist of 47% of all processes that should be properly performed by a project manager during the entire lifecycle of a project. However, project literature does not clearly identify which of the 21 planning processes are more crucial than others." (Zwikael, Globerson 2006, p. 3436)

From the description of the project managers focus it can however be inferred that risk management is more crucial than others. It is mentioned as a task that must be done in the Software Project Planning under 2.5. Risk Management. (Abran et al. 2004, §§ 2.5) Once again under quality assurance it is stated that "Risk management can also play an important role in delivering quality software." (Abran et al. 2004, pp. 11–15) In contrast to the complete and detailed information offered in the Requirements Knowledge Area, Risk management is a section (2.5), one paragraph long, with 6 external references for further reading. Though Risk management is identified as a critical area there is little guidance offered.

In general, the success factor can be stated to be planning, with a further focus on risk management. Another perspective is that the amount of planning overhead might be a factor by itself (Zwikael, Globerson 2006)

3.2.3.5 Summary of CSF's for SWEBOK

SWEBOK has the advantage of including PMBOK in its definition. Having a well-established project management standard as part of a software project development process makes a clear statement that good management practises are necessary for software development. Many of the CSFs identified by and the long term objectives of SWEBOK have more overlap than some of the other processes as is shown in Table 3-9.

Once again leadership ability has not been mentioned in a project management standard. There is a difference between management skills, which are described in this standard and taught, and leadership ability that is not.

Table 3-9 CSF's for SWEBOK

#	**Factor**
1	Adequate resources, rooms, hardware, test labs
2	Appropriate formalized development processes and methodologies
3	Clear assignment of roles and responsibilities
4	Clear known objectives and goals
5	Clear requirements or specifications
10	Effective monitoring and control
11	Effective management skills at team or project level
13	Frozen requirements
14	Good quality management
19	Realistic schedule
20	Risk management
22	Support from top management
25	Up-to-date progress reporting

3.2.4 Agile

The Agile objectives statement is on the landing page of the website, it has the advantage of being very clear about what is important:

"*We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:*

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more". (Beck et al. 2001a)

This section analyses the objectives and any other documents in their process definition to define the critical success factors the process is attempting solve.

3.2.4.1 Agile Context

History and Funding

The Agile Manifesto was preceded by growing dissatisfaction with waterfall methods of developing software. As early as the 1940's W. Edwards Deming began promoting PDSA, Gerald M. Weinberg reports he was doing incremental development at IBM in 1957. Winston Royce picked up the torch in his 1970 article reprinted in (Royce 1987) emphasizing that waterfall would only work for very simple projects. In the year 2000 Marco Iansiti, Roberto Verganti and Alan McCormack completed a 2 year empirical study of 29 processes showing that evolutionary development had the best chance of success (MacCormack 2001).

In 2001 a group of developers met and developed the Agile Manifesto quoted in the previous section Agile 3.5. Based on the experiences of the past it was recognized that requirements have to change during software development, and that the development process must be able to cope with it.

In 2003 strong endorsement of Agile techniques came from Watts S. Humphrey, a driving force behind CMMI:

> *"Your initial definition of the requirements will almost certainly be wrong. While you must get as close as possible to a usable solution, you must recognize that the requirements are a moving target that you must approach incrementally. To produce a truly superior and highly-usable product for a realistic cost and on a reasonable schedule, you must follow a development strategy and process that assumes that the requirements will change. This requires that you encourage early changes and that you develop and test*

the program in the smallest practical increments. Also, if possible, get the users to participate in the early testing." (Humphrey 2009, p. 104)

So a founding member of the SEI - CMM propagates customer collaboration and response to change.

The Agile Alliance was formed to promote Agile development. Their website lists 66 different sponsors, without the military or government research grants seen in the other models introduced in this thesis.

Target Group

The definition of Agile assumes that it is applicable to any application development. In reality the focus on small teams and short implementation cycles makes it ideal for small innovative solutions. Larger complex highly integrated projects are less well suited as the amount of effort aligning the separate teams increases with size.

Wysocki (Wysocki 2007) asserts that the best project management type can be determined as a function of ((uncertainty in requirements and solution), (uncertainty in the goal)). The less clear requirements are the greater the benefit of Agile,

Goal

The Agile Manifesto clearly states: "We are uncovering better ways of developing software by doing it and helping others do it." (Beck et al. 2001a)

Agile does not encompass the complete application lifecycle, the system analysis and operations phases are not topic. The requirements, design and testing phases of application development are interwoven where refinement of requirements, design and tests are moving forward in a continuous cycle.

To enrich this here is an Agile definition written by the CMMI product team: *"Multiple Agile approaches exist. The phrases "Agile environment" and "Agile method" are shorthand for any development or management approach that adheres to the Manifesto for Agile Development [Beck 2001]. "Such approaches are characterized by the following:*

- *Direct involvement of the customer in product development*
- *Use of multiple development iterations to learn about and evolve the product*
- *Customer willingness to share in the responsibility for decisions and risk" (CMMI Product Team 2011, p. 58)*
-

3.2.4.2 Objective: Teams and People

Individuals and interactions over processes and tools. (Beck et al. 2001a)

Of the 12 Principles behind the Agile Manifesto (Beck et al. 2001b) following five principles focus on the team:

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

There are many aspects mentioned about what the right culture is, there is agreement that it must include: trust, respect and skills.

Trust is a major factor in Agile methods. There are two areas of trust; management must trust the team and the team must trust each other. Management must be able to delegate technical decisions to the team knowing that the correct decisions will be made and that these decisions must be accepted.

The team members must feel comfortable with sharing their knowledge and weaknesses with the team. "The culture must also be supportive of negotiation, as negotiation is a big part of the Agile culture" (Lindvall et al. 2002, p. 203).

. Principle 8 about sustainable development is an interesting one. Agile teams perform their own time estimation while Sprint planning. The team thus owns and controls the time plan. The toil that insufficient development time takes on the development teams made its way into the Agile principles.

The team is expected to be highly motivated, technically skilled and self-organized. The development team as opposed to the management is the critical factor in Agile development.

3.2.4.3 Objective: Requirements and Communication

Requirements in Agile are seen as a form of communication. They are a method for capturing what the customer wants. The Agile Principle (Beck et al. 2001b) of welcoming change, even late in the development cycle, is a good principle for dealing with what Wysocki observed: “What the customer wants is probably not what the customer needs. Wants are more associated with the solution. Needs are more associated with the underlying problem. “(Wysocki 2007, p. 46)

Communication of user requirements must also be supplemented by direct contact between the users and the developers. "... the distinction between agile and document-centric methodologies is not one of extensive documentation versus no documentation; rather a differing concept of the blend of documentation and conversation required to elicit understanding." (Fowler, Highsmith 2001, p. 4)

The amount of user involvement is fundamental to the success of the project.

3.2.4.4 Objective: Working Software

Working software over comprehensive documentation (Beck et al. 2001a)

3. *Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale*.(Beck et al. 2001b)

Process specialists have been promoting iterative development for ages, empirical studies have shown iterative development to have the greater success rates.(MacCormack 2001; Larman, Basili 2003) This does not necessarily mean a release with each iteration, it suffices that there is an internally available working model for everyone to learn from.(Fowler, Highsmith 2001)

3.2.4.5 Summary of CSF’s for Agile

Nomen est Omen, the time frame for ‘Agile’ development is far shorter than CMMI. Where CMMI has a time frame of years, Agile process are measured in weeks. Thus the time frame of short term goals defined by CSF’s and the long term goals of Agile software development process are far closer together. This should give a greater alignment of short term and long term goals.

Table 3-10 CSF’s for Agile

#	**Factor**
1	Adequate resources, rooms, hardware, test labs
4	Clear known objectives and goals
6	Committed and motivated project team
7	Domain knowledge(for example of road traffic control)
9	Effective communication and feedback
12	Experience with the programming languages, technology and hardware
17	Low staff turnover, stable teams are built
19	Realistic schedule
24	Trustful and cooperative working environment exists.
26	User or client is involved while developing

Table 3-11 Success Factors and Process Models

	CSF's and Process Objectives					
Questionnaire Item Number	**Factors,**	**V-Modell XT**	**CMMI**	**SWEBOK**	**Agile**	**Nasir and Sahibuddin's Process Attribute Group**
6	Committed and motivated project team				x	
11	Effective management skills at team or project level		x	x		
15	Good Leadership at team level					**People**
16	Good Leadership at project level					
21	Skilled and sufficient staff					
22	Support from top management		x	x		
26	User or client is involved while developing				x	
	Count	0	**2**	**2**	**2**	**7 total**
1	Adequate resources, rooms, hardware, test labs			x	x	
2	Appropriate formalized development processes and methodologies	x	x	x		
3	Clear assignment of roles and responsibilities	x	x	x		
4	Clear known objectives and goals	x	x	x	x	
5	Clear requirements or specifications	x	x	x		
8	Change and configuration management	x	x			
9	Effective communication and feedback				x	**Process**
10	Effective monitoring and control	x		x		
13	Frozen requirements	x		x		
14	Good quality management			x		
17	Low staff turnover, stable teams are built				x	
18	Proper planning	x	x	x		
19	Realistic schedule	x	x	x	x	
20	Risk management	x	x	x		
24	Trustful and cooperative working environment				x	
25	Up-to-date progress reporting	x	x	x		
	Count	11	**9**	**12**	**6**	**16**
7	Domain knowledge				x	
12	Experience with the programming languages, technology and hardware				x	**Technology and skills**
23	Supporting tools – Project management and SW development tools					
	Count				**2**	**3**

3.3 Summary

The findings of the Literature Review chapter are summarized in Table 3-11 Success Factors and Process Models. The realistic schedule and clear objectives where the only factors focused on by all IT development process models. This was support by the results obtained. In an ideal project the schedule and objectives are ranked second and third (see CSF Ranking - Ideal Project (RQ1b) for full details). However the top spot was given to 'Effective Communication and Feedback' a factor stressed in Agile development, but not mentioned in other process models.

From the table it is clear that V-Modell XT is only dealing with the process aspects of IT development projects. The skills needed, whether soft skills (People Attribute Group) or hard skills (Technology Attribute Group) are not part of V-Modell. It is a guide to how to standardize a project process, not how to perform IT development.

SWEBOK and CMMI also describe processes, but in contrast to V-Modell they advocate that the manager of a project should have some management training.

Agile spreads itself thinly overall project attribute groups. There is a focus on people but no mention of standardized processes.

There is however a further option. Reinhard Höhn devoted a chapter to describing how V-Modell and Agile are not in conflict with one another. (Höhn 2008) Similarly CMMI can be used in conjunction with Agile methods as is discussed by Hillel Glazer (Hillel Glazer 2008). If these IT development methods are successfully combined then a far greater coverage of the success factors is obtained. However as an opening to his paper Hillel Glazer stated: "Agile development methods and CMMI best practices are often perceived to be at odds with each other" and continues the introduction by explaining why that is so. (Hillel Glazer 2008, p. 1)

In Table 3-12 Success Factors Not Covered IT Process Models are 3 CSFs that are not covered by any software development model.

Table 3-12 Success Factors Not Covered IT Process Models

	CSF's and Process Objectives
15	Good Leadership at team level
16	Good Leadership at project level
21	Skilled and sufficient staff

No IT development process talks of leadership skills and relating to people. SWEBOK was criticized: "It's clearly strongly based on Scientific Management principles, and as such utterly ignores the role of people and human interaction issues in process. Again since Individuals and Interactions trump Process, this is a huge hole in the body of knowledge." (Fowler 2003) V-Modell XT has a focus on clarity of communications however a clear communication, as described by this model, is a document, not a human interaction. The literature review showed that it is human interactions that make or break projects; this was also supported by the ranking of communication in an idea project (see CSF Ranking - Ideal Project (RQ1b). "Because we go about… work in teams, we are mostly in the human communication business, [not the high-tech business]." (DeMarco, Lister 1999)

Leadership is not in scope of CMMI and V-Modell XT, the goal is clearly to define processes not human interaction. Agile states that teams should be a self-organizing team of equals; the scrum-lead is an administrative function. A higher level of leadership is not seen as needed. No one will seriously consider planning a large scale multiple team, multiple location IT project without managers who possess both management training as well as people skills, however this combination seems lacking in the IT process models.

The literature reviewed shows that the best CSF coverage is obtained by combining either V-Model XT or CMMI development processes with Agile development methods. CMMI seems to advocate studying the organizations current processes and choosing the best practise. V-Modell XT offers a list of best practises and has the organization adapt its current process. The success factors found in the literature (The Critical Success Factor List for Software Development) do not clash with any of these models, but the models do not adequately cover all factors in the CSF list.

4. Methodology: CSF Questionnaire Design and Justification

This chapter describes the methodology used to collect and analyse data about success factors in application development at the Target Company. It described the questionnaire designed to capture the knowledge of the professionals working in the research and development department at the Target Company about CSF's in software application development.

The purpose of this chapter is to describe the research methodology chosen; the questionnaire design, the sample population and basis for data analysis.

4.1 Content, CSF List and Grouping

The first step in creating a questionnaire is to decide what questions to ask.

> "For to a folysshe demaunde behoueth a folysshe ansuere." (Caxton 1484)[3]

The goal of this section is to introduce a method to answer the research questions:

RQ1. Given a list of CSF's relevant to IT development projects in general:

a. How do the respondents rank the factors success in their current project?

b. What importance would they give the factors in an ideal project?

c. Is there any difference in the ranking of CSF's due to education, experience or the role the person has in the application development process?

RQ2. From the perspective of the Target Company, what are the critical success factors?

RQ3. What are the determining factors of project success?

RQ4. Is the ranking of the critical success factors related to the application development process being followed? Can the results be related to any particular application development process?

To answer the research questions a questionnaire was designed. The data collected must provide following information by evaluation of the results or by statistical inference from the data.

- RQ1: needs a valid ranked list of success factors, both for the current project as well as in an ideal situation.
- RQ1c: needs context information to the information gained in RQ1a and RQ1b.
- RQ3: needs the respondents to be able to communicate project success such that is numerically quantifiable.
- RQ4: The context information must include the process development model being followed.

The basis for the critical success factor list should be drawn from the literature review, with changes as dictated by expert review.

4.1.1 CSF List for the Research and Development Division

Nasir and Sahibuddin's critical success factors list was chosen as the base to work from. It was chosen for being most recent work found, and for having a sufficiently broad base of research to be creditable (43 research papers plus face to face interviews), as shown in Table 3-1. The original list was adapted in consultation with the project team at the Target Company and drawing on the information gained from the

---

3 Ask a silly question and you get a silly answer.

review of the software process models. Following adaptions were made as listed in Table 4-1 Changes to CSF List

Table 4-1 Changes to CSF List

Adaptions to CSF List	
CSF	Reason
Realistic budget	Removed: Target group does not have budget powers
Complexity, project size, duration, number of organisations involved	Removed: In house applications development, not relevant in context
Good performance by vendors/contractors/ consultants	Removed: Not relevant in the context
End-user literacy, knowledge and skills to use the system	Removed: No end-user contact within the development cycle.
Domain knowledge(for example of road traffic control)	Added: the specific knowledge of road traffic control is complex and specific skill
Low staff turnover, stable teams are built	Added: Important issue in a hire and fire world. See (DeMarco 1988)
Trustful and cooperative working environment exists.	Added: This CSF is directly related to the principles of agile software development. See (Beck et al. 2001b)

The list was then subjected to expert review as described in section 'Content Validity**'.** The final list content presented to the staff of the development team of the Target Company is shown in Table 4-2.

Table 4-2 Final CSF list for the Target Company

#	CSF
1	Adequate resources, rooms, hardware, test labs …
2	Appropriate formalized development processes and methodologies(think of process model)
3	Clear assignment of roles and responsibilities
4	Clear known objectives and goals
5	Clear requirements or specifications
6	Committed and motivated project team
7	Domain knowledge (for example of road traffic control)
8	Change and configuration management
9	Effective communication and feedback
10	Effective monitoring and control
11	Effective management skills at team or project level
12	Experience with the programming languages, technology and hardware
13	Frozen requirements
14	Good quality management
15	Good Leadership at team level
16	Good Leadership at project level
17	Low staff turnover, stable teams are built
18	Proper planning
19	Realistic schedule
20	Risk management
21	Skilled and sufficient staff
22	Support from top management
23	Supporting tools – Project management and ALM tools such as DOORs, ClearCase, SVN and similar
24	Trustful and cooperative working environment exists.
25	Up-to-date progress reporting
26	User or client is involved while developing

4.1.2 Project Attribute Groups

As discussed in 3.1.2, Project Attribute Groups and Project Success individual CSF's are not suited as determinants of project success. To be able to make a statistically meaningful statement about the determinants of project success Project Attribute groups were needed. From the literature analysis two possible methods of creating project attribute groups emerged.

Fortune and White propose that they be grouped in nine process areas which lead to the grouping of factors in process area groups (FSM categories. (Fortune, White 2006) The mapping of the factors is shown in Table 3-2. This grouping has 3 groups that are represented by only one CSF. This makes the group completely reliable from a statistical viewpoint, but increases the variance in the group. The determinants of project success will be less explicit as the impact is spread over many project attribute groups.

Nasir and Sahibuddin proposed a much simpler scheme as shown in Table 3-11. The groups are unequal in size, but do not have single CSF's representing whole categories.

4.2 Questionnaire Design

This questionnaire is designed to quantify the current state and the importance of the selected success factors from the literature. The construct validity, content validity and method bias of this approach is discussed below.

4.2.1 Construct Validity

The questionnaire content was based an overview of CSF's in literature performed by ((Fortune, White 2006). A large sample size was sought to give as much credibility to the research as possible the size needed is discussed in the with the sample population. To obtain the large sample size the questionnaire has to be short. The time needed to reply should be tailored to 10 minutes. The questionnaires are filed out during the respondent's working day. At the time the research was done the researcher performing the survey was working for the Target Company. Management agreed to their staff being surveyed but time per person was limited. The following issues construct validity; content validity and method bias were issues in designing the questionnaire.

Construct validity is needed to establish that mapping feelings to numbers is an acceptable practice. The validation originates in the social and medical sciences, where concepts are and must remain subjective, yet valid research is performed with quantified values gained from the subjective concepts. For instance: pain research is a good example for validly quantifying subjective values. Construct validity is applicable to critical success factors as the questionnaire seeks to quantify a subjective feeling, how good a subjective factor is in a project. These factors are values which are not objectively measureable. The justification can be found in "Nomological" validity. Boudreau explains this concept as follows:

1. "Nomological validity is construct validity that devolves from the very existence of a well-developed theoretical research stream (also called a nomological "network"). If theoretically derived constructs have been measured with validated instruments and tested against a variety of persons, settings, times, and, in the case of IS research, technologies, then the argument that the constructs themselves are valid becomes more compelling. … Assume that one researcher uses a structured interview script to gather data on a construct. Suppose that another researcher in another setting uses a questionnaire instrument. Clearly, the method of measurement is very, even maximally different. Yet, if both studies find significant linkages between the constructs using different measures, then both may be said to be "nomologically" valid." (Boudreau, Gefen 2004, p. 395).

Nomological validity has been established by numerous studies on CSF's such as the 63 analysed by Joyce and White "(Fortune, White 2006). Thus it can be concluded that it is possible to rate a success factor using a visual analog scale (VAS).

4.2.1.1 Scaling the Construct

To scale the answers a numeric visual analogue scale (VAS) was needed. A scale that the majority of the respondents were familiar with was sought to minimize the cognitive overhead. The scale chosen was the Austrian school grading system. Austrian schools grade on a five point numerical scale that has discreet verbally described values associated with it, see Table 4-3. This scale is familiar to the sample population thus reducing the learning time needed to fill out the questionnaire.

Table 4-3 Grading Scale

Number	Discrete value	
1	Very Good	Best possible
2	Good	Next highest
3	Satisfactory	Middle value

4	Adequate	Barely Acceptable
5	Unsatisfactory	Failed

This five point scale is a numeric as well as a category scale, comparable to a Lickert scale. The scale was accepted immediately by most respondents and needed few further explanations. A ten point scale had been tried in a pre-test version and discarded. The extra space needed for the additional points would have forced the questionnaire to a longer format without gaining precision in the results. By hooking into childhood experience the scale was accepted in the vastly altered context of a questionnaire.

The researcher was aware that scale is chosen is subjective to the respondent. The one respondent's answers cannot be compared absolutely to the next respondent thus the results will be imprecise and will preclude using any form of sensitive statistical analysis. This problem is inherent in any categorization of data such as success, or any type of feeling.

4.2.2 Method Bias

Any survey, including this one, inherently has a bias. The primary bias is introduced by choosing which questions to ask. There could be factors relevant to the population that are not part of the questionnaire. To deal with this bias Thomas states: "No general guidelines are likely other than the statement that each researcher must clearly delineate the nature of substantive constructs, potential methods factors, and plan for the measurement and/or control of both in the design and conduct of their research" (Schmitt 1994, p. 397).

Table 4-4 FSM Attribute Groups

Project Attribute Group = FSM Category	CSF's
Communication	6, 9, 24
Continuity	20
Decision Makers	3, 8, 11, 14, 15, 16, 18, 19, 22
Environment	2
Goals	4, 5, 13
Performance	10, 25
Resources	1, 12, 17, 21, 23
Transformation	7

Table 4-5 N&S Project Attribute Groups

Project Attribute Group	CSF's
People Related	6,11,15,16,21,22,26
Process Related	1,2,3,4,5,8,9,10,13,14,17,18,19,20,24,25
Technical Issues	7,12,23

4.2.2.1 *Imprecise Definition of Terms:*

The success or failure of a project is not clearly defined. Thus when rating 'success' it will not mean the same to each respondent. In deed one and the same project can be both a success and failure depending on who is answering. Projects that clients have felt successful for the supplier were failures leading to bankruptcy. The manager's success criterion – within budget or time is not the same as the developer's definition - functionality. This ALM survey deals with a complex topic, however in being concise and simple that some complex relationships are not conveyed. A trade-off is made in the questionnaire design between various levels of complexity. In the pre-test the questionnaire included several open ended questions, allowing the respondent to be precise in their answers. This option fell as the survey could not be completed within the desired 10 minutes in this form. This simplification leads to following issues. The questionnaire only offers the respondent a VAS scale. The answer to all non-contextual questions is a number from 1 to 5. The limits the richness of the respondent's information but was done to gain time and thus a greater number of returned questionnaires.

4.2.2.2 *Grouping*

A further issue that can lead to biased results is the grouping of the questions. "Respondents may be sensing the inherent constructs via the ordering of questionnaire items and they may respond accordingly." (Belassi, Tukel 1996, p. 396) To avoid this issue the lists of questions were sorted alphabetically. This clearly avoided any intended grouping affect. Any bias due to the grouping of the questions was avoided. The however an advantage of being able to reduce the cognitive overhead and thus reducing response time by grouping similar questions was not possible.

4.2.2.3 *Fatigue and boredom*

A further biasing factor is due to respondent fatigue. Questions at the end of the questionnaire will be given less attention than questions at the beginning. This issue was avoided by reducing the content of the questionnaire to fit on one page. The preliminary versions of the questionnaire were taking up to 30 minutes to complete, even well intentioned respondents were losing interest. At 10 minutes this was not happening, however nothing can completely eliminate the difference in attention given to the first answers compared to the last ones.

4.2.2.4 *Desired Responses/Anonymity*

When asking people about work what the management sees as a desirable response can lead to biased answers. The questions being asked in this questionnaire are sensitive information. The solution is obvious: "There are several additional procedures that can be used to reduce method biases, especially at the response editing or reporting stage. One is to allow the respondents' answers to be anonymous" (Podsakoff et al. 2003, p. 888) Protecting respondent anonymity and reducing evaluation apprehension is a necessary consideration so that safe or desirable answers do not predominate. The population has shown suspicions about answering online surveys in the past, the IT professionals seemed to be more paranoid than the average population. To guarantee anonymity and increase acceptance a printed questionnaire format was chosen without any linking context information. The answered questionnaires were collected from central collection points, so that there was no direct link existed between the respondent and the questionnaire. To further reassure respondents following line was added to the questionnaire.

"The information you give will be treated confidentially. Only cumulative results will be published as part of my master thesis. Individual questionnaires are not linked to the respondents."

4.2.3 Content Validity

Content validity or logical validity refers to the extent to which the measurement actually covers the relevant issues. The relevancy of the content to the ALM CSF's must be established for the questionnaire to be valid. The questions asked must be shown to be pertinent and sufficient coverage of critical success factors, "Content validity is established through literature reviews and expert judges or panels. Several rounds of pretesting the instrument, with different groups of experts, is highly advisable" (Boudreau, Gefen 2004, p. 387). Following steps were taken to provide content validity:

4.2.3.1 *Literature Review*

The literature on critical success factors was reviewed in the previous chapter. There has been both theoretical and empirical research over years on critical success factors. The literature on CSF's has provided several lists of success factors well adapted to a survey format. The article "Framing of project critical success factors by a systems model" by Joyce and White (Fortune, White 2006) was used as a basis for this questionnaire content as "through a review of sets of critical success factors from 63 publications it has revealed that there is a lot of overlap between sets" (Fortune, White 2006, p. 65).

4.2.3.2 *Professional Analysis*

Three professionals in software project management performed a review of 70 potential information items on 13 pages. These were S. Gruber – Project Lead EMC2, M. Gelbman – Owner Q-Success Software Consultants and A. Delamer, Professor FH Wiener Neustadt. The cognitive overhead of an open answer questions and lengthier questions placed a too large cognitive load on the respondents. The time needed to answer vastly exceeded the intended ten minutes. The content was relevant but the format had to be tightened. The 70 items were pared down to 34 items.

4.2.3.3 *Pilot Testing*

A reduced set of 32 items was formed to a two page questionnaire and evaluated by 10 staff members at the Target Company. As a result of the pre-testing a further dependant variable querying overall project success was added. This version was pre-tested on 20 staff members. Although the official correspondence language in the Target Company is English, a German translation was added to yet further reduce the time needed to complete the questionnaire.

4.3 Sample Population

4.3.1 Business Context

The Target Company develops large scale, long term intelligent traffic solutions. The majority of their solutions are combined hardware, software and back office solutions to collect highway traffic tolls. These products involve large hardware setups; gantries all toll roads, on board units for all trucks, billing systems. The working lifespan is more than decades. The first system was installed more than 10 years ago. No toll system has as yet been retired; upgrades take place on a rolling basis with as little disturbance to the running system as possible. This environment lacks the security and personal safety issues seen in defence, banking or health care, no is bombed, robbed or dies if a truck toll is incorrectly collected. The operations phase of the product life cycle predominates as the hardware installations have long life expectancies.

4.3.2 Sample Population and Size

The target population was the Research and Development the Target Company Vienna. These people are responsible for the development phase of the product lifecycle. Although the Target Company has multiple locations worldwide this study was limited to the Vienna location. Vienna has the largest R&D department with approximately 160 people working in the research and development phase of the application life cycle.

The total population (160) is relatively small; to obtain reasonable confidence a relatively large sample compared to the total population is needed. A 5% confidence interval (margin of error) at a 95% confidence level would require a sample size of approximately 105. A 75% return on a questionnaire is not realistic, thus the confidence interval was widened so the sample size could be reduced.

To calculate the necessary sample size using the techniques described in (Bartlett James E. 2001, pp. 44–46)

Population Size = ca. 140 (number of developers in R&D Vienna)

Confidence level = 95% (the nature of data is not precise)

Confidence Interval desired = 10%

Sample Size needed= 57 (40% of the population)

Thus a minimum return from 40% of the population was needed. A very good return (80 respondents) was obtained by the researcher personally handing out the sheets and requesting colleagues for help. A second attempt, to reach further respondents by distributing the questionnaires by mail, lead to next to no returns (<10%).

5. Results

This chapter presents the results of the questionnaire, the full questionnaire text is found in the appendix "E. Questionnaire Text". This chapter interprets the data obtained, the data analysis was performed using Microsoft Excel 2010 and SPSS (Statistical Package for Social Sciences) 20.0.0.

The respondent context information (page 1 of the questionnaire) showed a highly educated workforce. The majority of the employees are software developers, followed by testers and managers. Education and experience did not have statistically significant influence on the respondents ranking of project success. The context data gathered on the IT development process being followed showed a very low level of awareness in this area.

The CSF factor information was evaluated to provide 3 different rankings of CSF factors, which combined give a clear list of the top factors governing success in their environment. These are:

1. Clear requirements or specifications
2. Committed and motivated project team
3. Clear known objectives and goals

The least critical factors were

1. Risk management
2. Up-to-date progress reporting
3. Effective monitoring and control

The data indicates that the administrative overhead needed to collect the data is not desired; however the respondents do wish to have the benefits of the planning process, the specifications, clear objects and realistic time plans. This apparently contradictory position underscores the low awareness of development processes.

The final section of this chapter, Project Success, probes the determinants of project success. The results show that the respondents are not that unhappy about project success; they ranked their project as being more than 70% successful on average. This is a very optimistic figure, especially when compared to the Standish Report ratings or Scott Amblers figures which reported at least half of all projects as failing or compromised. (Ambler 2010; Nasir, Sahibuddin 2011) Thereafter a regression analysis is present looking into the determinants of project success. The model presented by Fortune and White presented in "Project Attribute Groups and Project Success", did not prove statistically reliable. Nasir and Sahibuddin's project attribute model was used to calculate the determinants of project success.

The data showed that software project success was determined by the state of the process group of success factors, the technology group and people group was not statistically significant as a determinant of project success. The 'Process' attribute is the largest group, encompassing following 16 items:

Clear requirements and specifications, Clear objectives and goals, Realistic schedule, Effective communication and feedback, Realistic budget, Frozen requirements, Proper planning, Appropriate development processes / methodologies (process), Up-to-date progress reporting, Effective monitoring and control, Adequate resources, Risk management, Effective change and configuration management, Good quality management, Clear assignment of roles and responsibilities.

These are the issues found in V-Modell XT, CMMI and SWEBOK. The Agile issues were mostly centred in the 'People Related' attribute. The factors in the 'People Related' project attribute group were ranked higher in the current project, than the factors in the 'Process' group.

Note! All results are inversely reported compared to how they were scored on the questionnaire

The questionnaire captured the participants opinion about the CSF's using a 5-point Likert scale, ranging from 5 (not at all or insufficient) to 1 (perfect or very good). This scale is standard in the Austrian school system. Thus, in the questionnaire a low value (1) was good and a large value (5) was bad. This scale proved to be misleading for analysing numbers and interpreting graphs. To avoid confusion the answers to the questionnaire were reverse scored. This means that the values in the result section are the inverse of the values entered on the questionnaire: reverse score(x) = max(x) + 1 – x , so good is a high value, bad is a low value and the intuitive meaning of the numbers is restored.

5.1 Preliminary issues

5.1.1 Respondent Context

The questionnaire was distributed in September of 2012. The respondents are employed by the Research and Development division of an intelligent traffic solutions provider. They develop IT projects, approximately two thirds of the team works on a project was started in spring of 2010, estimated for 18-20 months and close to completion. The rest of the team works adapting existing solutions with project durations of 1 to 12 months. The full text of the questionnaire is available in Appendix E. Questionnaire Text

Replies were obtained from 80 of a population of approximately 140 individuals. Of the 80 replies 79 were all or at least partially valid, as complete data sets were necessary for analysis of variance the sample sizes here are smaller as incomplete data series could not be used. The roles and process questions were non-

Table 5-1 Respondents Role in the development process

Role	count	
Dev	45	57%
Q.A.	16	20%
Lead	12	15%
Mgr.	3	4%
Other	3	4%
Grand Total	79	100%

numerical thus analysis is limited to the frequency and relative frequency of each answer. The experience

Table 5-2 Education / Average Years of Experience

Education	N	%	Development Experience	Domain Experience
University	57	72%	10.3	4
HTL	19	24%	16.1	5
Apprentice	3	4%	22.3	14
All Respondents	79	100%	12.2	5

question is a numerical field thus the standard mean, median and variance can be calculated.

5.1.1.1 Role

The respondents show that more than half (57%) work as Dev (Developers). There are less than 3 people doing hardware development, the rest range from developing embedded software to web solutions. The application range from hardware control systems for cameras and microwave readers to back-office billing systems. The QA (quality assurance) team members work in software testing, this ranges from functional testing in a scrum like environment, to release testing prior to delivery to the customer. The Lead is a Group Leader, leading a team of 2-8 members. The group leads will in part be still working hands on either as a developer or software tester. The managers (Mgr.) have a completely managerial function.

5.1.1.2 Education and Experience

The respondents skill set was measured with two indicators. First their formal education, second their experience both in IT development and third their knowledge of intelligent traffic solutions was asked for. The results are summarized in Table 5-3.

The respondents have a high education level. 72% of the staff has some form of higher education compared to 14.9% of the Austrian population as a whole in 2010. A further 24% are HTL graduates (upper secondary school for technology).

The next indication of skill level is experience. There are two aspects of experience in IT-development projects. The development experience indicates knowledge of IT development techniques. This is the time spent with IT development in general. This would be time spent practising IT development skills. The second skill needed is knowledge of the domain in which the project operates. The domain for this development team is intelligent traffic solutions; this includes for example the laws pertaining to traffic tolling or the density and velocity of highway traffic. This market in Austria is limited to this one company, so domain knowledge was equated to years working with the company. The average domain experience is 5 years; however the median domain experience lies by 2 years. The top ten percent has been with the company for an impressive 21 years on average, the maximum value was 35 years.

Not surprisingly the amount of experience drops with the added time spent in higher education, but the company does seem to be good at retaining the university graduates they have. The HTL graduate have 56% more development experience, but have only 25% more domain experience. They are not spending their added employment years with the company.

Thus the data shows that the respondents are highly skilled. They are well educated, have sufficient IT-project experience and are with this employer long enough to learn the problem set being worked on.

5.1.1.3 Development Process

Context items ‘e’ and ‘f’ address the issue of the development process followed. The questions are:

e) Which process model is used in your current project? (Table 5-3)
f) If a process model is used is it (Always/Frequently/Usually/Seldom/Never) followed? (Table 5-4)

As seen in the tables 4 different process models were identified: Agile, V-Modell, Waterfall, and an In-House defined process. In addition 23% of the respondents claimed that no development process was being followed, an additional 3% did know what process model was intended but never followed it. If these two numbers are taken together then 1 in 4 members of the development team are working at CMMI level 1 – no development process used. The official development process documents written by the QA-Manager in 2010 espoused V-Modell XT as the standard development model.

The anticipated answer was V-Modell XT or Agile being known and followed at least usually by 2/3 of the developers. The answers clearly did not support this hypothesis. The obtained results do not permit a statistically valid relationship to be defined between the process being followed and other answers on the questionnaire.

Table 5-3 E) Development Process Used

Process	**N**	**%**
Agile/Scrum	29	37%
V-Modell	20	25%
In House	10	13%
Waterfall	2	3%
None used	18	23%
totals	79	100%

Table 5-4 F) For Agile and V-Model

The given model is followed:	**Count**	**%**
Agile/Scrum	**29**	**37%**
Always	2	3%
Frequent	8	10%
Usually	18	23%
Seldom	1	1%
Never	0	
V-Modell	**20**	**25%**
Always	**0**	
Frequent	5	6%
Usually	8	10%
Seldom	5	6%
Never	2	3%

5.1.2 Precondition: Validity of Project Attribute Groups

The literature review revealed two models for creating independent variables by gathering the CSF's to attribute groups as described in 3.1.2 Project Attribute Groups and Project Success. This section examines the validity of the project attribute models proposed.

Two models of project attribute groups were calculated. The grouping as proposed by Nasir and Sahibuddin as described in 4.1 Content, CSF List and Grouping, as well as the grouping to project attribute groups as proposed Fortune and Whites work (Fortune, White 2006). Both these models recognized the need for independent variables to as determinants of variance in project success. There are therefore two approaches to project attribute groups calculated in this section a three group split as described in (Nasir, Sahibuddin 2011) as well as a 8 group split as described in (Fortune, White 2006).

The first step is to get a feel for how valid the grouping is by verifying the internal consistency and reliability of the groups.

> "When using Likert-type scales it is imperative to calculate and report Cronbach's alpha coefficient for internal consistency reliability for any scales or subscales one may be using. The analysis of the data then must use these summated scales or subscales and not individual items. If one does otherwise, the reliability of the items is at best probably low and at worst unknown. Cronbach's alpha does not provide reliability estimates for single items." (Gliem, Gliem, p. 89)

Thus Cronbach's alpha can provides information about the groups that consist of multiple factors; however the single factor process group's reliability cannot be established. Also of interest is r. R is the "Corrected Item-Total Correlation—this is the correlation of the item designated with the summated score for all other items, these values should be at least .40." (Gliem, Gliem, p. 86).

5.1.2.1 Project Attribute Groups Fortune and White's FME Model

Section Project Attribute Groups and Project Success3.1.2 explained the need for project attribute groups. Fortune and White's (F&W) approach to defining project attribute groups involved mapping them to categories described in FSM. Table 3-2 shows Fortune and Whites the mapping of the questionnaire factors to the attribute groups.

As a first step in analysing project success as being determined by project attribute groups Cronbach's alpha as well as descriptive values were calculated .

The values used for describing the internal consistency using Cronbach's alpha are: "_ > .9 – Excellent, _ > .8 – Good, _ > .7 – Acceptable, _ > .6 – Questionable, _ > .5 – Poor, and_ < .5 – Unacceptable"." (George, Mallery 2010, p. 231)

Table 5-5 shows the calculations performed to determine reliability for the Fortune and Whites project attribute groups. The calculations showed that the internal consistency of the Fortune and Whites Project Attributes Group's was not sufficiently established, thus this model will not be used for further statistical analysis. From a theoretical viewpoint this model was plausible; it holds far more detail and insight about processes; however as it cannot be proven internally consistent the groups cannot be used for calculating determinants of project success.

A problem that could be affecting the reliability of the grouping is the relatively small number of CSFs in some attribute groups. The smaller the number of members causes more variance, a larger number of factors have a greater probability of being reliable.

Table 5-5 Cronbach's alpha: Fortune and White's FSM model

Group	CSFs	N	Alpha	Comment
Communication	6, 9, 24	3	0.70	At a value of 0.70 (rounded to 2 decimal places) the reliability is Acceptable. All items are worthy of retention, reoving any single CSF would decrease the alpha. All items correlated with the total scale to a good degree (lowest r=.45)
Continuity	20	1		not calculated
Decision Makers	3, 8, 11, 14, 15, 16, 18, 19, 22	9	0.85	The value of 0.85 is good reliability. Removing any single CSF decreases the reliability except for 22- Support from top management which left the alpha value virtually unchanged (+0.001). This was also seen in the correct item to scale correlation which was lowest for #22 (r=.40) the other values ranged from .518<r<.693.
Environment	2	1		not calculated
Goals	4, 5, 13	3	0.54	The value of 0.537 is poor, but not yet unacceptable. Number 5 - clear requirements are holding together this group. If this item is left out then alpha falls to .175, The item to total scale correlation for #4 is .26, #13 slightly better at .30 but still not sufficient to pass Gliem's (Gliem, Gliem) recommendation of > .40.
Performance	10, 25	2	0.60	At two digits significance alpha is 0.60 a 'questionable' reliability. The correlation to the total scale is r=.43. As only two values are present, none can be removed to calculate the impact on alpha.
Resources	1, 12, 17, 21, 23	5	0.60	A 'Questionable' result at two digits significance. The item to scale correlation is particularly bad for 23 - Supporting tools where r=.269.
Transformation	7	1		Alpha is not applicable to single items.

5.1.2.2 Project Attribute Groups - Nasir and Sahibuddin Model

Using the grouping shown in Table 3-3 Critical Success Factors based on categories, the reliability of two process attribute groups ('people' and 'process') is acceptable. No statement can be made about the 'Technical' attribute group. The low number of items in the groups is reducing the reliability. Removing a factor T2 (Supporting tools – Project management and ALM tools) from the group improves α to 0.637. Thus all groups were internally consistent.

The internal consistency (reliability) of the Nasir and Sahibuddin Project Attributes Group's was established.

Using Cohen's (Cohen 1988) guidelines for the strength of Pearson correlation coefficient it can be stated that there was a strong positive correlation between the process attributes and the people attributes, r=0.798, as well as between the process attributes and the technology attributes, r=0.559. Pearson's correlation between people attributes and technology attributes showed only a more moderate correlation with r=0.438.

It can be stated that while the attribute groups process and people can be expected to increase in parallel with each other, the technology group will not be as direct in its behaviour.

Table 5-6 N&S - Cronbach's alpha

Project Attribute Group	CSF's	N	α	Comment
People Related	6,11,15,16,21,22,26	7	0.707	acceptable
Process Related	1,2,3,4,5,8,9,10,13,14,17,18,19,20,24,25	16	0.9	Excellent
Technical Issues	7,12,23	3	0.457	0.5 is poor. It improves to questionable if T2 is removed.T2 is weak, improve to .63 if eliminated

Table 5-7 N&S Project Attribute Group – Current Project

Attribute Group	N	Minimum	Maximum	Mean	Standard Deviation
People	66	2.29	4.71	3.4	.57
Process	67	1.75	4.46	3.1	.65
Technology	67	2.00	5.00	3.7	.64
Total possible	66				

5.2 CSF Ranking - Current Project, (RQ1)

The second page of the questionnaire was devoted to the critical success factors and project success. The list of CSF factors was discussed in 4.1.1 CSF List for the Research and Development Division. The respondent evaluated on a scale of insufficient to very good the state of the factor in their current project. In parallel the respondent is asked to evaluate how important the factor is to a hypothetical ideal project. The data here will answer:

RQ1. Given a list of CSF's relevant to IT development projects in general:

a. How do the respondents rank the factors success in their current project?

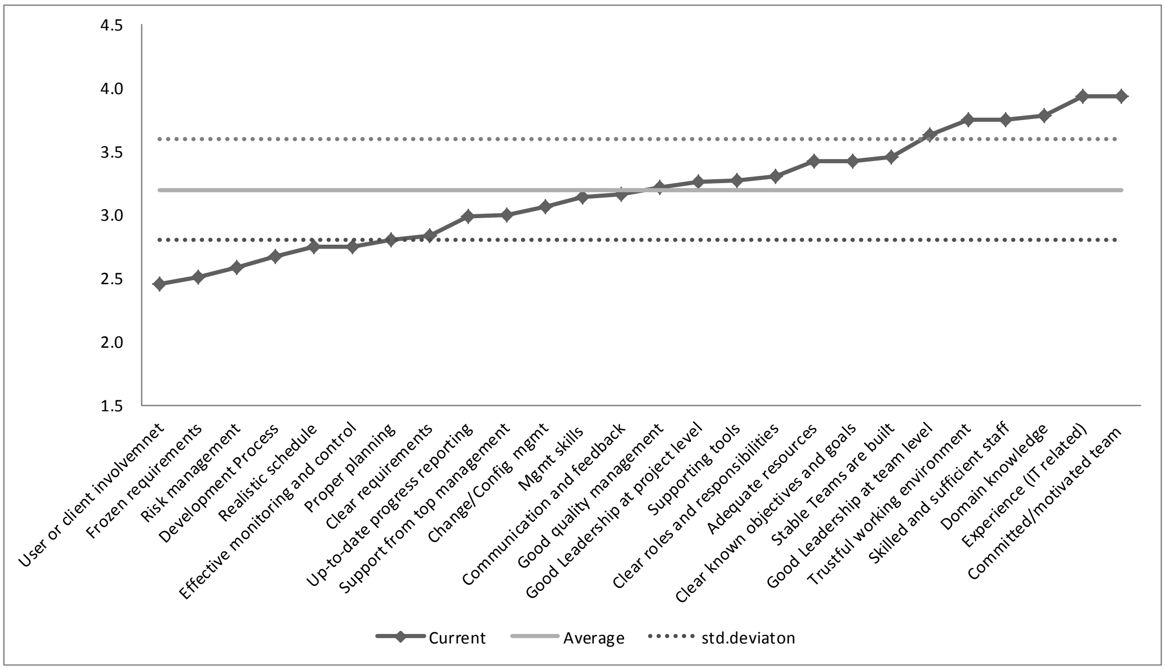
b. What importance would they give the factors in an ideal project?

c. Is there any difference in the ranking of CSF's due to education, experience or the role the person has in the application development process?

Of interest are those factors are at the very top or very bottom of the ranking, those factors that are rated highly or lowly as contributing to project success. Furthermore those factors where the difference or distance between the current state and the ideal state is greatest.

The CSF's are graded using a Licker Scale, this means that the absolute values of the numbers found have no intrinsic value, however the relations between the values are valid, and thus sorting and comparing the values obtained is valid. A ranking based on the obtained values can be performed; however the numbers have no intrinsic value outside of the framework of this survey.

Figure 5-1 CSF Ranking in Current Project



The critical success factor ranking in the current project is shown graphically in Figure 5-1, the complete data is shown in Table 5-8.

The average ranking given to the CSF's in the current project was 3.2 with a standard deviation of 0.4. The six following factors are more than one standard deviation greater than the average rank of 3.2:

- Committed and motivated project team M=3.9
- Experience with the programming languages, technology and hardware M=3.9
- Domain knowledge M=3.8
- Skilled and sufficient staff M=3.8
- Trustful and cooperative working environment exists. M=3.8
- Good leadership at team level M=3.6

At the other end - following factors were rated on average more than one standard deviation less the group average. These CSF's are not contributing to project success in the current project:

- Frozen requirements M=2.6
- Realistic schedule M=2.5
- User or client involvement M=2.5

The data shows that the management skills, planning, requirements, and schedule are seen as being below average in their contribution to the current project. It would not be correct to infer that these areas automatically need attention. It is equally possible that these factors are not relevant in the project environment. For instance, the contact to the client is not relevant in this environment because there is no client interface in the application being written.

The data also indicates that the respondents are quite happy about themselves. They rate their motivation, experience, domain knowledge and working environment as most successful in the current project. Team leadership and trust is also within the top five. The most plausible conclusion is that the Target Company is a good place to work.

Questionnaire: Page 2/ Q1: How would you rate the given critical success factor in your current project? Page 2/Q2: Q2: How important would it be in and ideal project.?

Table 5-8: Critical Success Factor - Results

Item	Critical Success Factors - Ranked by Current Value	Current	Ideally	Diff
6	Committed and motivated project team	3.9	4.5	0.6
12	Experience with the programming languages, technology and hardware	3.9	3.9	0.0
7	Domain knowledge(for example of road traffic control)	3.8	3.5	0.3
21	Skilled and sufficient staff	3.8	4.2	0.4
24	Trustful and cooperative working environment exists.	3.8	4.2	0.5
15	Good Leadership at team level	3.6	3.9	0.2
17	Low staff turnover, stable teams are built	3.5	3.7	0.2
1	Adequate resources, rooms, hardware, test labs	3.4	4.2	0.7
4	Clear known objectives and goals	3.4	4.3	0.8
3	Clear assignment of roles and responsibilities	3.3	4.1	0.8
23	Supporting tools – Project management and ALM	3.3	3.6	0.3
16	Good Leadership at project level	3.3	4.0	0.8
14	Good quality management	3.2	3.9	0.7
9	Effective communication and feedback	3.2	4.1	0.9
11	Effective management skills at team or project level	3.1	3.7	0.6
8	Change and configuration management	3.1	3.7	0.6
22	Support from top management	3.0	3.6	0.6
25	Up-to-date progress reporting	3.0	3.5	0.5
5	Clear requirements or specifications	2.8	4.3	1.4
18	Proper planning	2.8	4.0	1.2
10	Effective monitoring and control	2.8	3.3	0.6
19	Realistic schedule	2.7	4.1	1.4
2	Appropriate formalized development processes and methodologies	2.7	3.4	0.8
20	Risk management	2.6	3.6	1.0
13	Frozen requirements	2.5	3.4	0.9
26	User or client is involved while developing	2.5	3.4	0.9

5.3 CSF Ranking - Ideal Project (RQ1b)

The second half of question 1 requested the respondent to rank: “Which factors, in an ideal project, are contributors to success”.

For 26 CSF’s the respondents graded on a scale of 1 to 5, how important a factor is to the success of an ideal project at the Target Company. The results are shown in Figure 5-5 CSF Ranking in Ideal Project. The raw data is found in the appendix (Ideal Project).

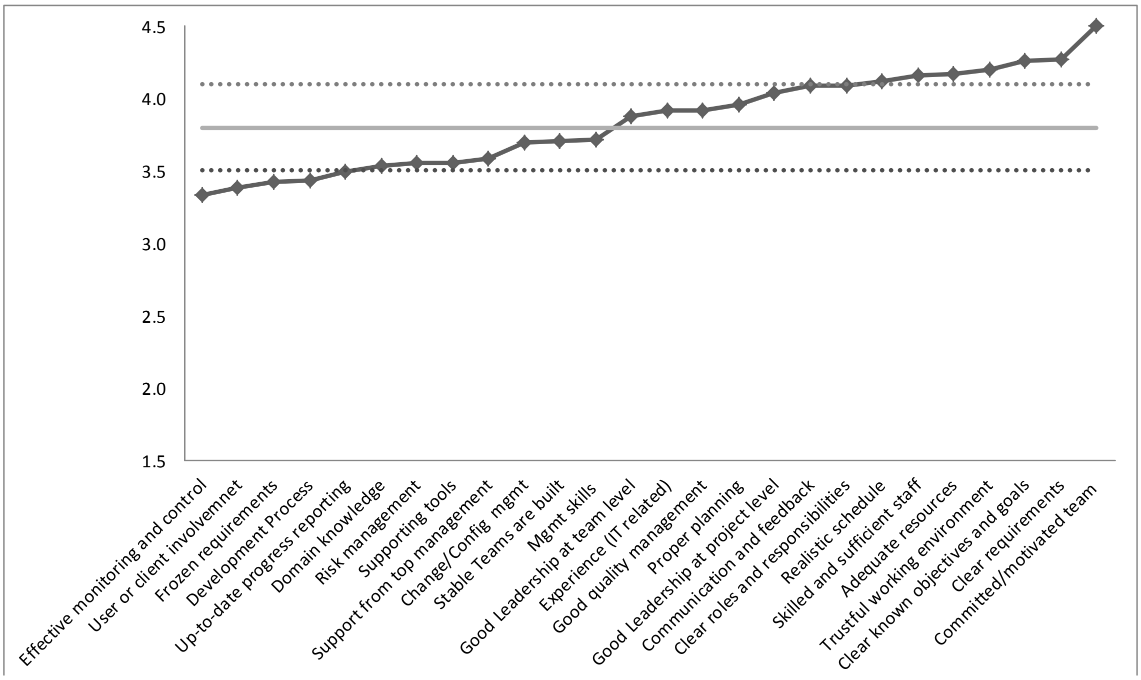
The average ranking given to the CSF’s in the ideal project was M=3.8 with a standard deviation SD=0.3. In comparison with the current project’s values (M=3.6, SD=0.4) ideal factors have a higher average and less variance. “Good leadership at team level” M=3.8 lies exactly on the average rating for an ideal project.

The following factors have factor values M>=4.1, which is more than one standard deviation greater than the M=3.8 + SD=0.3over all factors in the ideal group. The factor “Committed and motivated project teams” predominates at is more than 2 standard deviations from the average value of 3.8

1. Effective communication and feedback M=4.1
2. Clear assignment of roles and responsibilities M=4.1
3. Realistic schedule M=4.1
4. Skilled and sufficient staff M=4.2
5. Adequate resources, rooms, hardware, test labs M=4.2
6. Committed and motivated project team
7. Clear requirements or specifications
8. Clear known objectives and goals
9. Trustful and cooperative working environment exists M=4.2
10. Clear known objectives and goals M=4.3
11. Clear requirements or specifications M=4.3
12. Committed and motivated project teams M=4.5

The following factors have factor values M<=3.5, which is more than one standard deviation less than the M=3.8 - SD=0.3over all factors in the ideal group. These are seen as least important for the success of an

Figure 5-2 CSF in Ideal Project



ideal project.

1. Effective monitoring and control M=3.3
2. User or client is involved while developing M=3.4
3. Frozen requirements M=3.4
4. Appropriate formalized development processes and methodologies(think of process model) M=3.4
5. Up-to-date progress reporting M=3.5
6. Domain knowledge(for example of road traffic control) M=3.5

5.4 CSF Ranking - Distance from Ideal

The distance from the ideal describes how far the current CSF value differs from the value the CSF would have in an ideal world. This distance is simply obtained by subtracting the current ranking of the CSF from the corresponding ideal value.

Distance= (Ideal Project Value) - (CurrentProject value)

In Figure 5-6 Distance Current to Ideal is a plot of the resulting values.

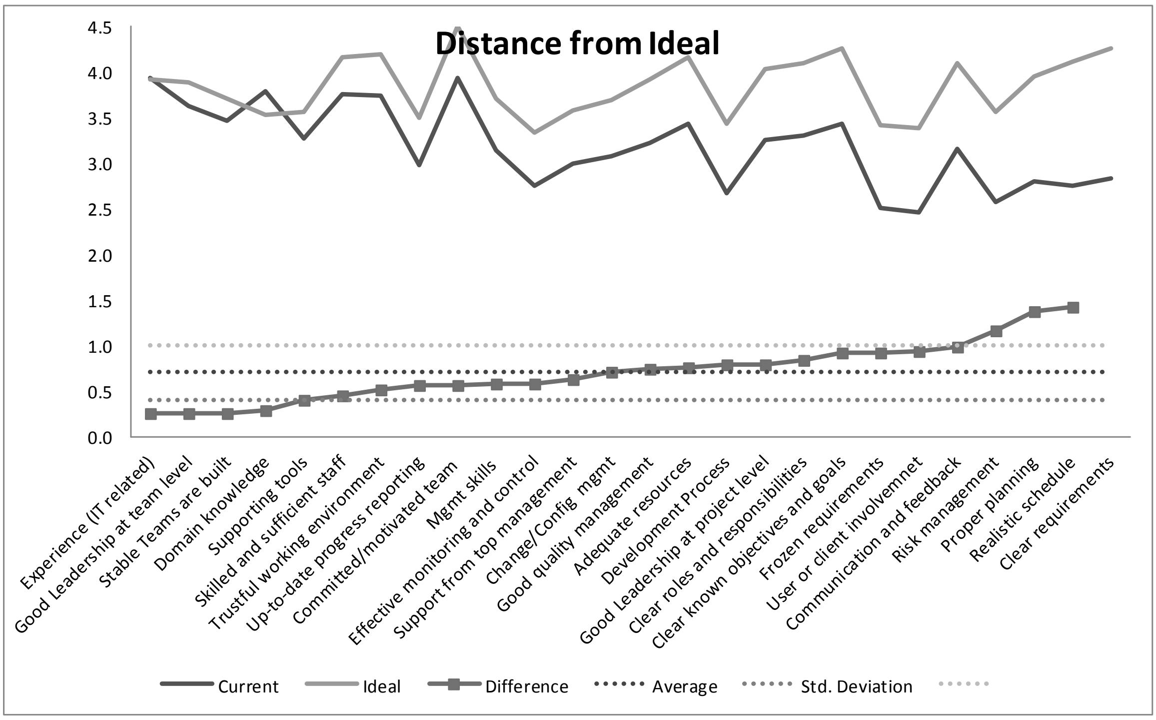
From the data gathered the average, average distance M=0.7, SD=0.3 Following values exceed the mean distance (M=0.7, SD0.3) by more than one standard deviation.

- Risk Management M=1.0
- Proper planning M=1.2
- Clear requirements/ specifications M=1.4
- Realistic Schedule M=1.4

The factors “clear requirements/specifications” M=1.4, and ‘realistic schedule” M=1.4 are both more than two standard deviations distant from the average difference between current and ideal projects.

On a more optimistic note: The respondents feel that the current IT-relevant experience is ideal, their team leads are good and their job is safe. Following factors were also more than one SD from the mean in a positive sense –

Figure 5-3 Distance Current to Ideal



1. Domain knowledge(for example of road traffic control) 0.3
2. Supporting tools – Project management and ALM tools 0.3
    - Low staff turnover, stable teams are built M=0.2
    - Good Leadership at team level M=0.2
    - Experience with the programming languages, technology and hardware M=0.0

5.5 CSF Ranking - Top Issues Vote

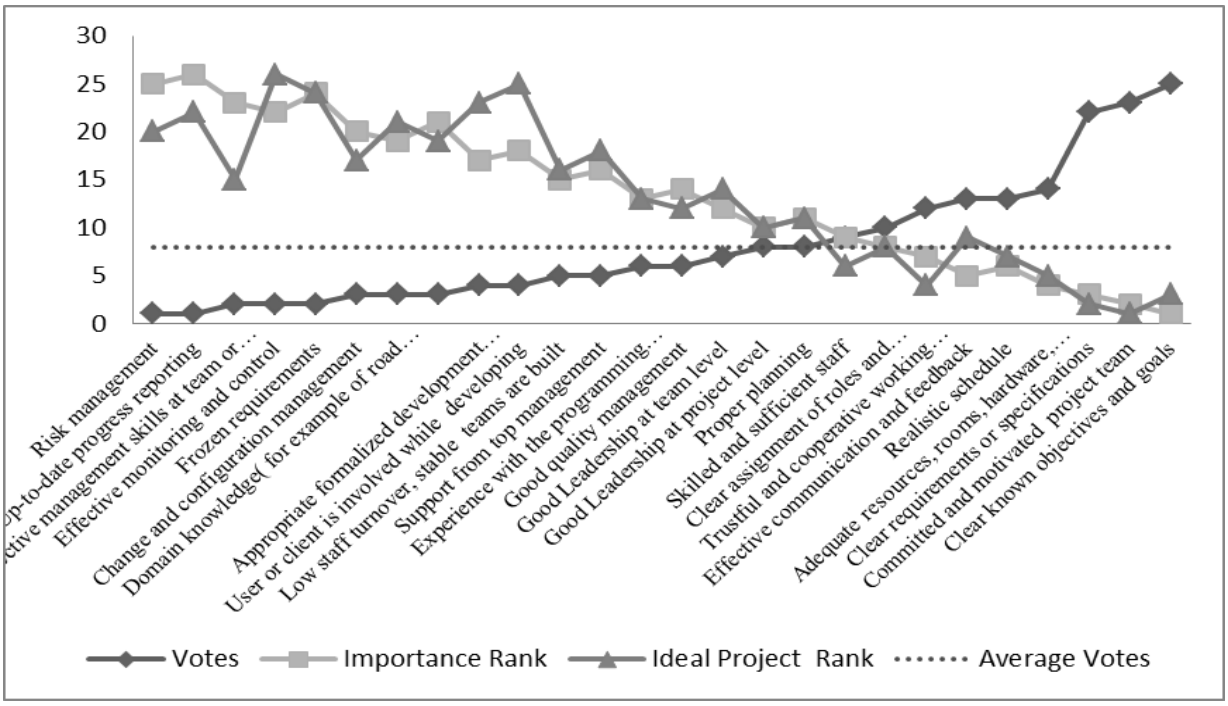
Page 2 Question 2 requested the respondent to. “Mark the 2-5 factors you feel are the most important.”

The mean and standard deviation is not relevant for this question. There should be a top end with a very large number of votes giving a clear indication of what the top issues are. The data shows a clear statement of the top three factors. The top factors have in excess of 20 votes and there is a very clear gap between these factors the next highest on the list. Figure 5-4 Top Factors indicates visual consistency between the ideal and current project factors. The top ranked ideal factor is rated 1 and therefore is a low value in the graph.

The top factors in the list are:

1. Clear known objectives and goals votes=25, 48%
2. Committed and motivated project team votes=23, 44%
3. Clear requirements or specifications votes=22, 42%

Figure 5-4 Top Factors



Though the top factors have a clear lead, not one factor had an absolute majority amongst all respondents, N=52, the top factor had 48%.

The data indicates support for the statement: "There does not appear to be a consensus of opinion among researchers and authors on the criteria for judging project success and the factors that influence that success." (Wateridge 1995, p. 171) All factors got at least one vote for being within the top five; however the following were clearly not universally supported:

1. Risk management votes=1
2. Up-to-date progress reporting votes=1
3. Effective management skills at team or project level votes=2
4. Effective monitoring and control votes=2
5. Frozen requirements votes=2
6. Change and configuration management votes=3

7. Domain knowledge(for example of road traffic control) votes=3
8. Supporting tools – Project management and ALM tools votes=3
9. Appropriate formalized development processes and methodologies votes=4
10. User or client is involved while developing votes=4

Amongst these issues are a large number of process related activities that generate extra administrative overhead.

5.6 Impact of Context-(RQ1.c)

Research question 1.c is "Is there any difference in the ranking of CSF's due to education, experience or the role the person has in the application development process?

The short answer to the question above is: None that are statistically relevant. The rest of this section explains how that statement was reached. It describes the analysis of the influence of the context variables, education, role in the organisation, and process model on Nasir and Sahibuddin's 3 Project Area Groups: Personal, Process, and Technology.

Education

The respondent's education and role in the application development process were investigated for a relationship with project success.

A one-way ANOVA was run to determine the effect of education on the process, people and technology attribute groups, in other words whether education would make a difference to the aggregate critical success factors. The apprenticeship education was ignored in this calculation as the group had two few members (N=3), thus HTL and University remained. Box's test of equality a homogeneity of variance-covariance matrices (p =.638) passed. The HTL graduates estimated all critical success factors more positively than the university graduates as is seen in Table 5-9. There were no significant differences between the answers on project attribute groups. $F(2,60)=1.053$ $p=0.35>.05$; Wilks' $\Lambda = .966$; partial $\eta^2 = .034$).

Table 5-9 Education and Project Attribute Groups

Education of Respondent		M	SD	N
process	1 HTL	3.05	0.81	15
	3 UNI	3.09	0.62	48
	Total	3.08	0.66	63
people	1 HTL	3.49	0.63	15
	3 UNI	3.43	0.55	48
	Total	3.44	0.56	63
technology	1 HTL	3.82	0.69	15
	3 UNI	3.63	0.64	48
	Total	3.67	0.65	63

Table 5-10 Role and Project Attribute Groups

project attribute Role		M	SD	N
process	1,00 - Dev	3.22	0.68	40
	2,00 - Test	2.95	0.47	11
	5,00 - Lead	2.81	0.72	10
	Total	3.11	0.66	61
people	1,00 - Dev	3.51	0.59	40
	2,00 - Test	3.44	0.31	11
	5,00 - Lead	3.16	0.65	10
	Total	3.44	0.57	61
technology	1,00 - Dev	3.72	0.67	40
	2,00 - Test	3.79	0.69	11
	5,00 - Lead	3.53	0.59	10
	Total	3.7	0.65	61

Variance in Project Attribute groups due to Role

The calculation described in the previous page was repeated to determine the influence of role on the project attribute groups that are the aggregated CSF's. One again, Box's test of equality a homogeneity of variance-covariance matrices (p =.806) passed.

The developers score consistently higher in the project attributes than the leads. The mean value for the leads was less than either testers or developers on all project attribute groups. None the less there was no significant difference in the responses based on the respondent's role in the organization. $F(4, 114)=1.260$ $p=0.290>.05$; Wilks' $\Lambda = .917$; partial $\eta^2 = .042$).

Variance in Project Attribute groups due to Process followed

The calculation performed for education was repeated using the process models as determining factors. Once again there was no statistically significant result. The process followed does not affect the respondent's views on the critical success factors, though of better significance than role or education.

$F(8,116)=1.703$ $p=0.105>.05$; Wilks' $\Lambda = .901$; partial $\eta^2 = .105$).

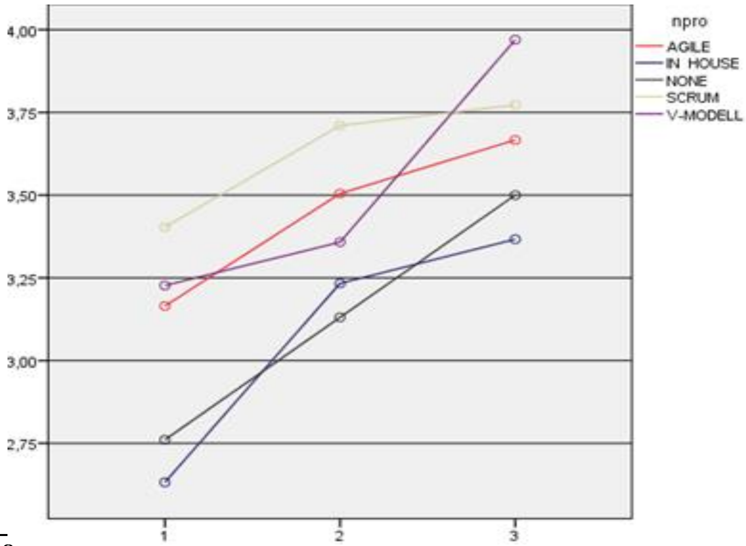
The pattern to the responses is hard to discern from Table 5-11. Figure 5-1 plots the attribute group means by process followed. From the diagram it appears that people answering 'in house' and 'none' to the process model followed are less happy with the state of the project on all measures. The V-Modell XT followers are happiest about technology, and Agile/Scrum followers are generally optimistic. Unfortunately none of these potential differences are statistically significant.

Table 5-11 Process and Project Attribute Groups

Descriptive Statistics

Attribute Group	Process	M	SD	N
process	1 AGILE	3.17	0.41	5
	2 IN HOUSE	2.63	0.69	10
	3 NONE	2.76	0.64	16
	4 SCRUM	3.40	0.52	22
	5 V-MODELL	3.23	0.68	11
	Total	3.07	0.66	64
people	1 AGILE	3.50	0.34	5
	2 IN HOUSE	3.23	0.51	10
	3 NONE	3.13	0.63	16
	4 SCRUM	3.71	0.46	22
	5 V-MODELL	3.36	0.61	11
	Total	3.41	0.57	64
technology	1 AGILE	3.67	0.41	5
	2 IN HOUSE	3.37	0.71	10
	3 NONE	3.50	0.72	16
	4 SCRUM	3.77	0.53	22
	5 V-MODELL	3.97	0.64	11
	Total	3.67	0.64	64

Figure 5- 1-Process, 2-People, 3-Technology



5.7 Project Success

The last question in the survey asked the respondent to judge the success of the current project on a scale of 0-100. The results are shown in Figure 5-5 .

Of the 79 returned questionnaires 56 answered this question (N=56, M=70, SD=17). The diagram shows that the distribution is not normal; however the median value is also 70. There is a very clear clustering between 70 and 100 with a second smaller group between 30 and 50.

A second view of project success is shown in "Table 5-12 Project Success by Role". The respondent's view of project success is broken down by the role that the respondent has in the application lifecycle. The group leads are happiest about the project at 74; the Quality Assurance people are least convinced at 66.

Comparing these numbers to "Table 1-1 Chaos or Ambler Project Success and Failure" indicates that either the project is doing far better than the average IT-project, or that the respondents definition of project success is not the same as what Standish or Ambler were using.

Figure 5-5 Current Project Success

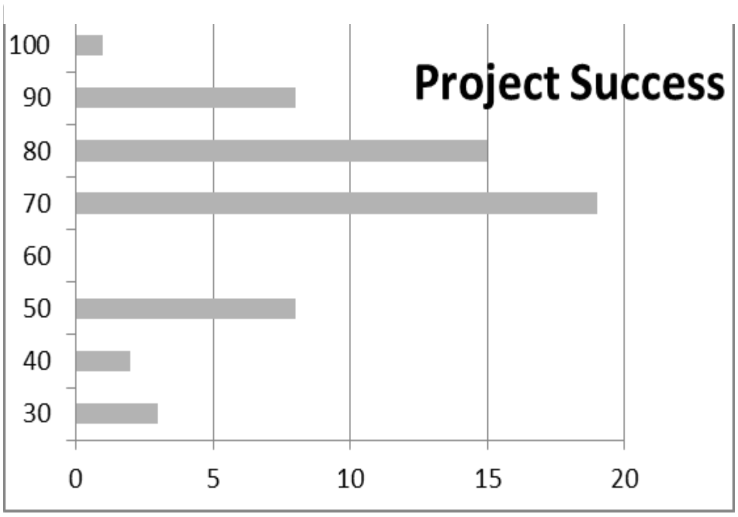


Table 5-12 Project Success by Role

Role	N	Average Estimated Success Value
Lead	9	74
Developer	38	71
Quality Assurance	5	66
Grand Total	54	71

5.7.1 Determinants of Project Success - (RQ3)

This section answers investigates RQ3: What are the determining factors of project success?

The calculations were performed using SPSS release 17. The assumptions for performing regression analysis were met, these are that:

- the sample size is large enough (N approx. 50)
- Section 5.1.2 Precondition: Validity of Project Attribute Groups
- Data was inspected for outlying results
- Pearson's correlation was performed successfully (see

Table 5-13 Pearson's Correlation between Attribute Groups

Correlation between Project Attribute Groups			
Pearson's	Process	People	Tech-nology
Process	1	0.798	0.559
People		1	0.438

The calculation of the determinants of project success was performed the SPSS model summary can be seen on the next page in Table 5-15.

Table 5-14: Calculating the Determinants of Project Success

Model Summary						
Model	R	R Square	Adjusted R Square	Std. Erro of the Estimate		
1	,496[a]	.246	.233	26.020		
a. Predictors : (Constant), Process						
ANOVA[a]						
Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	13014.085	1	13014.085	19.222	,000[b]
	Residual	39944.767	59	677.030		
	Total	52958.852	60			
a. Dependant Variable Variable: ProjSuccess						
b. Predictors : (Constant), Process						
The regression model is statistically significant, F(1,59)=19.222, p<.0005						
Coefficients[a]						
		Unstandardized Coefficients		Standardized Coefficients		
Modell		B	Std. Error	Beta	T	Sig.
1	(Constant)	-10.468	16.403		-.638	.526
	Process	22.849	5.212	.496	4.384	.000
a. Dependant Variable Variable: ProjSuccess						

Using the mapping to project attribute groups as described in section 3.1.2.2 Nasir and Sahibuddin, linear regression established that the 'Process' attribute group was a statistically significant determinant of project success F(1,59)=19.222, p<.0005. The “Process Attributes Group” accounted for 23% of the explained variability in the “Project Success” value. The regression equation was:

projSuccess_percent = -10.468+(22.849xProcess)

The remaining two project attribute groups as well as the context variables were included in the regression analysis but were not accepted as predictor variables.

Table 5-15 Determinants of Project Success

		Process Project Attribute Group
P1	5	Clear requirements or specifications
P2	4	Clear known objectives and goals
P3	19	Realistic schedule
P4	9	Effective communication and feedback
P5	13	Frozen requirements
P6	18	Proper planning
P7	2	Appropriate formalized development processes and methodologies(think of process model)
P8	17	Low staff turnover, stable teams are built
P9	24	Trustful and cooperative working environment exists.
P10	1	Adequate resources, rooms, hardware, test labs
P11	20	Risk management
P12	8	Change and configuration management
P13	14	Good quality management
P14	3	Clear assignment of roles and responsibilities
P15	25	Up-to-date progress reporting
P16	10	Effective monitoring and control

The data indicate that around 1/4 of project success is related to the Process Attribute Group, the factors listed in Table 5-15 above. The technological and people factors were not contributors to the variance in project success.

5.8 Results Summary and Interpretation

The details of the responses have been presented in the previous pages. This section uses these details to compare and contrast the answers given. The first half discusses the critical and the noncritical factors in IT development projects. The critical factors are the few top factors that can make or break a project. The noncritical issues are factors that are either performing well at present or are irrelevant to the work at hand, in both cases, well off or irrelevant, they can be ignored in the current project. In a second step the relationship between the CSFs for the current project and the objective of long term process improvement are discussed.

5.8.1 Issues and Conclusions from Ranking (RQ1b, RQ2)

This section answers RQ2: From the perspective of the research and development division, what are the critical success factors?

The ranking of critical success factors was the focus of sections 5.2, 5.3, 5.4, and 5.5. The respondents were asked to weight factors in the current project, give the factors a theoretical weight in an ideal project, and then vote for the top CSF factors. The results of the 3 approaches for finding the top critical success factors are shown in Table 5-16. The rankings found not only the critical factor but also those factors that can be safely ignored: these are listed in the bottom half of the table.

Table 5-16 CSF Issues and Nonissues

	Top Factor Voting	Distance from Ideal	Current Project Rating
	********* Critical Factors - Manager should watch these ***********		
	Factors voted as being the top CSFs	**Factors where the ideal weight and the current project weight are far apart.**	**Factors that people ranked negatively in the current project.**
1	Clear requirements or specifications	Risk management	Clear requirements or specifications
2	Committed and motivated project team	Proper planning	Proper planning
3	Clear known objectives and goals	Realistic schedule	Effective monitoring and control
4		Clear requirements or specifications	Realistic schedule
5			Appropriate formalized development processes and methodologies
6			Risk management
7			Frozen requirements
8			User or client is involved while developing
	********* Noncritical Factors - Safely Ignore These *******		
	Factors that were eliminated as top CSF	**Factors where the ideal weighting and the current project weighting match**	**Factors that people ranked positively in the current project.**
1	Risk management	Experience with the programming languages, technology and hardware	Committed and motivated project team
2	Up-to-date progress reporting	Good Leadership at team level	Experience with the programming languages, technology and hardware
3	Effective management skills at team or project level	Low staff turnover, stable teams are built	Domain knowledge(for example of road traffic control)
4	Effective monitoring and control	Domain knowledge(for example of road traffic control)	Skilled and sufficient staff
5	Frozen requirements	Supporting tools – Project management and ALM tools	Trustful and cooperative working environment exists.
6			Good Leadership at team level

For projects in general: When asked to vote for the top factors the respondents answers were completely clear: The most critical factors are:

- Clear requirements or specifications
- Committed and motivated project team
- Clear known objectives and goals

For the current project: The ranking of CSFs in the current project was not as clear as for projects in general. Those factors having a large difference from the ideal are shown in Table 5-17. The table is of those CSF (> 0.2 points) from the ideal, sorted by the current ideal value. If the top five are taken from this list the factors for the current project are:

- Effective communication and feedback
- Clear requirements or specifications
- Proper planning
- Realistic schedule
- Risk management

Table 5-17 Proposed CSF List

		Current	Ideally	Difference
9	Effective communication and feedback	3.16	4.09	0.93
5	Clear requirements or specifications	2.84	4.27	1.43
18	Proper planning	2.80	3.96	1.16
19	Realistic schedule	2.75	4.12	1.37
20	Risk management	2.58	3.56	0.98
13	Frozen requirements	2.51	3.42	0.92
26	User or client is involved while developing	2.46	3.39	0.93

Having a committed and motivated team is a CSF in a general project, but is not an issue in this environment.

On a more positive note the bottom half of Table 5-16 contains factors that are noncritical to the current project. Here the third column, the current project noncritical factors is a list of things that work particularly well in this project. The respondents were satisfied with issues closely related to the immediate environment of IT – development. The IT-skills, domain knowledge and team atmosphere are rated highly. Classic management factors, requirements, specifications, planning, monitoring and scheduling were seen as greatest areas of deficit.

5.8.2 Relating Development Process to CSF (RQ4)

RQ4: Is the ranking of the critical success factors related to the application development process being followed? Can the results be related to any particular application development process?

The responses to context questions ‘e’ and ‘f’ relating to the development process section 5.1.1.3 showed that the development process awareness is very weak in the group (see Table 5-18). Thus first half of RQ4 can be negated: The CSF rankings are cannot be statistically attributed to any particular development process.

Table 5-18 Development Process Used

	Never	**Seldom**	**Usually**	**Frequent**	**Always**	**sum**	
Agile/Scrum		1	18	8	2	29	37%
V-Modell	2	5	8	5		20	25%
In House		4	5	1		10	13%
Waterfall			2			2	3%
No Process	18					18	23%
						79	100%

Table 5-19 Current Project CSF and Process Models

	CSF's and Process Objectives						Target Company CSF Ranking (see: Table 5 10)	
question #	**Factor**	**V-Modell XT**	**CMMI**	**SWEBOK**	**Agile**		**Critical**	**Successful**
6	Committed and motivated project team				x			1
15	Good Leadership at team level							6
21	Skilled and sufficient staff							4
5	Clear requirements or specifications	x	x	x			2	
9	Effective communication and feedback				x		1	
18	Proper planning	x	x	x			3	
19	Realistic schedule	x	x	x	x		4	
20	Risk management	x	x	x			5	
24	Trustful and cooperative working environment				x			5
7	Domain knowledge				x			3
12	Experience with the programming languages, technology and hardware				x			2

The second half of RQ4 is whether the answers can be related to a particular process model (see Table 5-19). This table shows top critical factors and successful factors for the Target Company in columns 7 and 8. Column 2-6 shows the presence of the CSF in the objectives of the process models.

The second half of RQ4 is: Can the results be related to any particular application development process? The successful factors are covered mainly by the Agile objectives. In contrast, the project is weak in areas that are covered by V-Modell and CMMI. These factors are mostly project management topics, not specific to IT.

There are two CSFs of interest not covered by any application development process model. ‘Good leadership’ and ‘Skilled and sufficient staff” are factors not covered by any of development process models researched. “Effective communication and Feedback”, it is an objective in Agile, but not mentioned in the IT development process models, the respondents ranked this factor as the top critical issue. The data indicates that a combination of Agile and either CMMI or V-Modell XT would best serve the Target Company. The users of development models should be aware that there are critical issues in IT project management that are not covered by IT development process models. The best implemented IT process will not cover all factors necessary for project success.

In both the literature as well as all rankings of factors, requirements and planning always remain top issues. These topics are found as objectives in all process development models except Agile. Four of the top five factors ranked as being critical in the current project are found as objectives in the IT process models V-Modell XT or CMMI. The noncritical factors in the current project are objectives focused on by Agile development methods. A combination of both a development process as well as method would cover the most CSFs. The integration of the models is seen as cultural mismatch, but none the less a possible and desirable process goal. (Hillel Glazer 2008; Kiesling 2011)

6. Conclusion

6.1 Summary

This thesis sought to find the critical success factors for a running IT development project and investigated how the short term focus on critical factors matched the long term goals and objectives of IT development process models. The need to improve project success is clear given the poor performance of IT projects in general.(Standish Group International 2010; Ambler 2010) A light weight method for finding critical success factors is needed because time is essential. The amount of time needed for IT process improvement is greater than the time available to a single IT project. Critical success factors must be found and acted upon with in the time available to an IT-project. Critical success factors enable the manager to assign priorities to short term improvements which benefit the current project. The short to mid-term improvements must be aligned with the long term goals of process improvement, as an IT development process model is needed to continue successful practises in the long term.

6.1.1 Critical Success Factors

Literature

The literature on critical success factors in IT development projects lead to a list of 26 potential critical success factors as shown in Table 3-1. Only five of the 26 factors were mentioned in more than half of the sources reviewed. These five are:

1. Clear requirements and specifications:60%
2. Clear objectives and goals: 56%
3. Realistic schedule: 54%
4. Effective project management skills/ methodologies (project manager): 53%
5. Support from top management: 52%

Thus the literature does not give a golden list of 5 CSFs that are appropriate for all IT-projects at all times. Though the top five are critical issues for projects in general, the currently critical issues of a running project could differ.

Results

The selection of top issues is usually left to the top manager. In 1979 John F. Rockart wrote an article titled “Chief Executives Define Their Own Data Needs”. The method developed and demonstrated in this thesis uses the project team’s knowledge of the current project. This assumes that the group knowledge is more reliable than individual opinion. (Surowiecki 2004). To tap the knowledge of the project team a questionnaire was developed based on the literature review. The questionnaire design, content and construct validity as well as the necessary sample population size is discussed in “4: Methodology: CSF Questionnaire Design and Justification”. Whereas each individual answer to a question on project success is not valuable, the answers form multiple respondents are. The sum of multiple answers gathers the knowledge of the team; the value of the whole is far greater than the sum of the parts.

To demonstrate the questionnaire, the Target Company agreed to a case study comprising of the complete research and development department (approximately 140 people). The target group for the case study covered all roles present in the development phase of the product lifecycle (managers, developers and quality assurance). To create a wider view of the success factors the respondents were asked about 26 factors, first in the context of the current project, then in a hypothetical ideal project, and thirdly to vote for 3-5 top factors for IT development projects. A summary of the results is shown in “Table 6-1 Conclusions”.

The right side of Table 6-1 shows the factors that are more than one standard deviation from the mean for an idealized project as well as the current project. The 3rd column of the table shows the top ranked factors in the literature.

Key to Success: Having **“**Clear requirements and specifications” and a “Realistic schedule” are found in every ranking, in the literature, an ideal project or in the current project, these two issues are the basis of project success. The key to success is knowing what to do and having the time to do it properly. "This was cute, … so many projects don’t really do any estimating at all. Rather, they propose a goal and then get someone to espouse it as an estimate." (DeMarco 2012, pp. 8–9)

Table 6-1 Conclusions

	CSF's and Process Objectives		**LITERATURE**						**RESULTS**	
			Process Attributes						**Top Factors:**	
#	**Factor**	**Top Ranked**	**V-Modell XT**	**CMMI**	**SWEBOK**	**Agile**	**Process Attribute Group**		**Generally Critical**	**Currently Critical**
6	Committed and motivated project team					**x**	**People**		**2**	
11	Effective management skills at team or project level	4		**x**	**x**					
15	Good Leadership at team level									
16	Good Leadership at project level									
21	Skilled and sufficient staff									
22	Support from top management	5		**x**	**x**					
26	User or client is involved while developing					**x**				
1	Adequate resources, rooms, hardware, test labs				**x**	**x**	**Process**			
2	Appropriate formalized development processes and methodologies		x	**x**	**x**					
3	Clear assignment of roles and responsibilities		x	**x**	**x**					
4	Clear known objectives and goals	2	x	**x**	**x**	**x**			**1**	
5	Clear requirements or specifications	1	x	**x**	**x**				**3**	**2**
8	Change and configuration management		x	**x**						
9	Effective communication and feedback					**x**				**1**
10	Effective monitoring and control		x		**x**					
13	Frozen requirements		x		**x**					
14	Good quality management				**x**					
17	Low staff turnover, stable teams are built					**x**				
18	Proper planning		x	**x**	**x**				**5**	**3**
19	Realistic schedule	3	x	**x**	**x**	**x**			**4**	**4**
20	Risk management		x	**x**	**x**					**5**
24	Trustful and cooperative working environment					**x**				
25	Up-to-date progress reporting		x	**x**	**x**					
7	Domain knowledge					**x**	**Technology**			
12	Experience with the programming languages, technology and hardware					**x**				
23	Supporting tools – Project management and SW development tools									

Legend:
The conclusions table summarizes the results of this thesis. The columns have following meanings

- 'factors': are those CSF's derived from the literature; they are listed with the short text description.
- 'the next 4 columns': show which software development model includes the factor as an objective or goal
- 'Attribute': comes from (Nasir, Sahibuddin 2011) work on project attribute groups
- 'CSF in General'**:** lists five attributes of general importance to an IT development project
- 'CSF for Target': lists those factors that are current critical in the target environment
- 'Success Factors': are those factors making the project successful in the target environment

Differences: The ideal project differed significantly from the literature. Whereas the literature focused on project management skills, the development team rated a "Committed and motivated team" as critical to project success. In the reviewed literature having a committed team was ranked 22^{nd}. (Nasir, Sahibuddin 2011) However from the answers it is seen that respondents are happy about their own contribution to project success.

Positive factors: The ranking also showed the most successful factors in the current project, see Table 6-2. The six factors farthest from average positively, are people related. This group ranks themselves as "Skilled", "Motivated", and "Cooperative". The majority of these success factors can be associated with Agile objectives. The majority are factors from the 'People' process attribute group such as "a trustful working environment", and "committed, highly skilled, motivated teams". The positive success factors indicate that the team judge their skills and motivation more than adequate for the job at hand.

Table 6-2 Positive Success Factors

Success Factors
Committed and motivated project team
Experience with the programming languages, technology and hardware
Domain knowledge(for example of road traffic control)
Skilled and sufficient staff
Trustful and cooperative working environment exists.
Good Leadership at team level

The data indicates that the problematic issues lie within the process attribute group,

- Clear requirements or specifications
- Proper planning
- Effective monitoring and control
- Realistic schedule
- Appropriate formalized development processes and methodologies
- Risk management

From the authors knowledge of the project, frozen requirements and User involvement could be eliminated from the problems list. The end user is not a direct stakeholder in the development of intelligent traffic solutions and having frozen requirements is a factor mentioned in the literature, they have never been part of the reality of this (and other) development teams.

6.1.2 Project Success

Literature

Both Fortune and White (2006) as well as Nasir and Sahibuddin (2011) reviewed the literature on CSF and created lists of the factors found. Ultimately the factors should be determinants of project success. In both papers the individual CSFs were grouped together, into process attribute groups. Nasir and Sahibuddin (2011) created 3 project attribute groups, People, Process and Technology, as shown in the 8^{th} column of Table 6-1. The model presented is simple, however it proved statistically reliable in determining project success which a second method tested did not (see the section on "Precondition: Validity of Project Attribute Groups").

Results

Project success is what interests everyone; it is the bottom line. A regression analysis was run linking the process attribute groups to the reported project success value. It was shown that the 'Process' factors are the determinants of project success. The "People" and "Technology" attribute groups were not statistically linked to project success. This result – "Process" factors are determining project success is supported by the low awareness of the IT development process models as discussed in the results section on "Development Process" Of the people returning the questionnaires, 26% were not following a development process, a further 13% seldom followed the stated IT development process. For process improvement to be sustainable, IT development process awareness must solidify. .

6.1.3 IT Development Process Models and Critical Success Factors

A review of four IT development process models, CMMI, V-Modell XT, SWEBOK and Agile was performed, looking for the critical success factors amongst their goals and objectives. A summary of these finding is shown in columns 4 to 8 of Table 6-1 Conclusions. From the table it can be seen that a number of critical success factors are not covered in all the IT development process models. Of the process improvement objectives there are only two critical success factors are covered as an objective in all process development models, **"**Clear requirements and specifications" and a "Realistic schedule". The same two issues that showed up as top factors in all rankings of CSFs. Knowing what to do and having enough time to do it, is a rather obvious precondition to project success, but as banal and obvious as these factors sound, they are repeatedly identified as being problematic, both in the project management literature and in the development process model objectives. Requirements and scheduling are no less serious for being obvious success factors.

The IT development process models do not address all issues targeted by the CSFs, each focus on specific factors leaving others unmentioned. V-Modell XT completely concentrates on factors found in the process attribute group. It deliberately leaves out People or Technology factors as being not on topic. Agile covers more People and Technology issues, but leaves out the process factors. CMMI, similar to V-Modell, ignores most people related factors.

6.2 Future Work

There are several areas that have not been covered by this thesis. Firstly, this method was demonstrated using a development team, the applications lifecycle however starts with the inception phase and ends in an operations phase. The critical success factors in other project phases would be different. Extending the model to all phases of the application lifecycle would give an indication of if and how CSF importance changes over the application lifecycle.

Secondly, the project attributes groups as described in this thesis (see section "3.1.2 Project Attribute Groups and Project Success") could be tuned to be better indicators of project success. An alternative mapping of factors to project attribute groups based on a more descriptive model would deliver better determinants of project success. At present the project attribute groups are valid and plausible but do not have the explanatory power that the FSM has.

Thirdly, IT development process models should be expanded to address more of the objectives and issues identified as success factors for IT projects. Soft skills such as communication and motivation cannot be captured in a metric, but are no less important to the success of IT projects. The table suggests that a combination of Agile and either CMMI or V-Modell XT would lead to the greatest coverage of success factors. This is not unthinkable. There are voices that point out that Agile is primarily a development method, whereas the two development models, CMMI and V-Modell XT, are process models thus Agile can be combined with them (Höhn 2008; Hillel Glazer 2008). The obstacles to combining a development and a process model are mainly ideological. The Agile followers see V-Modell and CMMI as bureaucratic nightmares. V-Modell and CMMI followers, with stakeholders in the US and German military, believe that Agile projects are totally out of control. There is a strong case for combining the two approaches if ideological differences can be overcome.

6.3 Final Words

The light weight approach to determining critical success factors is viable for determining issues in running IT development projects. The method clearly identified the current projects critical factors. The literature analysis showed the connection, and in places the lack of connection between success factors for IT project success and the IT development process models (section 5.8.2). The answers also showed the teams perception of project success and the factors driving project success are. The development team views their project to be far more successful than average. There is no pessimism in their ratings of the project or their abilities.

The method introduced in this thesis queried all the people involved in a project about the factors affecting their work. There is no better source of knowledge about a project than the people that are working on it.

A. List of Abbreviations

ALM	application lifecycle management
ANOVA	Analysis of Variance- A statistical test for heterogeneity of means by analysis of group variances.
CMM	Capability Maturity Model
CMMI	CMM integration
CSF	critical success factor
FSM	Formal Systems Model
IIDD	iterative and incremental design and development
KTC	Target Company
MANOVA	Multiple analyses of variance, a statistical test procedure for comparing the means of several groups. It is used to test if changes in the independent variables have significant effects on the dependant variable. SPSS provides individual p-values for the dependant variables, indicating where differences are significant and to what proportion they are influencing the results dependant variable.
MFF	Multi-lane Free Flow – collection of highway tolls from vehicles without stopping traffic. These solutions work without tolling plazas using on board units to transmit vehicle information.
PCA	A principal component analysis is conducted to determine which project variables that vary together with each success variable (Wohlin, Andrews 2001, p. 46)
PDSA/PDCA	Plan Do Study/Check Act
PMBOK	Process Management Body of Knowledge
SDLC	software development lifecycle process
SEI	Software Engineering Institute
SPL	Software Product Lifecycle
SWEBOK	Software Engineering Body of Knowledge
SWECC	Software Engineering Coordinating Committee
VAS	Visual Analogue Scale

B. List of Terms

Application lifecycle Application lifecycle management	The application lifecycle includes all phases of solution development the naming of the phases differ according to which model is being used: • analysis of user requirements, • solution design, • implementation, • documenting and testing • Operation and maintenance phase. The terms are used loosely, each term including more or less activities connected with the application. Application lifecycle management is all activities needed to get people together to accomplish the development of a specific application
bureaucracy	A procedure in absence of context.
Capability Maturity Model	A CMM provides state-of-the-art practices to - Determine the maturity of an organization's processes. - Establish goals for process improvement. - Set priorities for immediate process improvement actions. - Plan for a culture of product or service excellence. (Schaeffer 1998)
cross sectional survey	A survey taken at one moment in time on as broad a base as possible.
Document centric	Focusing on the document as the centre of all knowledge. V-Modell XT and CMMI are document centric models.
Domain	"a sphere of knowledge, influence, or activity" (Domain - Definition and More from the Free Merriam-Webster Dictionary)
FSM	Formal System Model
Likert scale	"Likert scales are commonly used to measure attitude, providing a range of responses to a given question or statement. Likert scales fall within the ordinal level of measurement. The response categories in Likert scales have a rank order, but the intervals between values cannot be presumed equal." (Jamieson 2004)
Software development life-cycle	The systems development lifecycle is one part of the application lifecycle
Software development process,	Very similar concepts for the structures imposed on developing software or combined software project.
Visual Analogue Scale	The mapping of non-numerical values such as success or pain to numerical values. It is used to quantify qualitative data.

C. List of figures

D. List of tables

E. Questionnaire Text

Note: all information is on two pages, thus a single sheet was sufficient as hand-out. The literally light weight, minimal visual content was geared at getting the maximum returns. Replies were obtained from 80 out of a population of 120-150 individuals.

Department of Computer Science.

Critical Success Factors in Application Lifecycle Management

I am writing my master's thesis on critical success factors in application lifecycle management. The management permitted me to distribute this questionnaire. The purpose of this questionnaire is to gather the experience of software professionals at The Target Company about what factors their projects need to succeed (critical success factors). The goal is to test if the theory of success factors fits the reality as seen at KTC.

The information you give will be treated confidentially. Only the cumulative results will be published as part of my master thesis. Individual questionnaires are anonymous; they are not connected to a person. If you have further questions or would like a copy of the final results feel free to contact me either at work – my name is in the company directory or via my University of Vienna email address below.

Thanks for your assistance
Margaret Gruen-Kerr,

<u>Note:</u> At times the questionnaire refers to a "current project". Please answer referring to the current project you are working on. If the current project is not typical or you have worked on it less than 3 months then choose a project you have worked on for more than 4 months in the past five years.

<u>Context Information:</u>

a) Which role(s) do you perform?

Developer	Systems Engineer	Group/Team Lead
Quality Assurance		Manager
System Acceptance Testing		
Operations Support	Other: ________________	

b) Educational Background: ____

c) How long have you worked in systems development? ___________(years)

d) How long have you been with the Target Company? ___________(years)

e) Which process model is used in your current project:

- V-Modell XT
- Agile
- Scrum
- Waterfall
- Test driven
- In house defined process
- None Used/don't know
- Other :

f) If a process model is used is it:

- Always followed
- Frequently followed
- Usually followed
- Seldom followed
- Never followed

Critical Factors

1) Below is a list of factors affecting project success. The scale on left measures how it is in your current project, on the right how important this factor is to product success. For instance adequate resources could have a current value of 1 (all that you could wish for) but an ideal value of 3 (not that important to project success). The scale is the Austrian school grades so **5 is poor - 1 is excellent**. Skill staff might currently be 3 – the people are still learning but the importance is a 1, very important to project success. **NA** is for not applicable if the factor does not apply to your work.

Current Project: factor is Poor Excellent							**Factor: importance is** Not at all Very					
						Success Factor						
na	5	4	3	2	1	Adequate resources, rooms, hardware, test labs …	na	5	4	3	2	1
na	5	4	3	2	1	Appropriate formalized development processes and methodologies(think of process model)	na	5	4	3	2	1
na	5	4	3	2	1	Clear assignment of roles and responsibilities	na	5	4	3	2	1
na	5	4	3	2	1	Clear known objectives and goals	na	5	4	3	2	1
na	5	4	3	2	1	Clear requirements or specifications	na	5	4	3	2	1
na	5	4	3	2	1	Committed and motivated project team	na	5	4	3	2	1
na	5	4	3	2	1	Domain knowledge(for example of road traffic control)	na	5	4	3	2	1
na	5	4	3	2	1	Change and configuration management	na	5	4	3	2	1
na	5	4	3	2	1	Effective communication and feedback	na	5	4	3	2	1
na	5	4	3	2	1	Effective monitoring and control	na	5	4	3	2	1
na	5	4	3	2	1	Effective management skills at team or project level	na	5	4	3	2	1
na	5	4	3	2	1	Experience with the programming languages, technology and hardware	na	5	4	3	2	1
na	5	4	3	2	1	Frozen requirements	na	5	4	3	2	1
na	5	4	3	2	1	Good quality management	na	5	4	3	2	1
na	5	4	3	2	1	Good Leadership at team level	na	5	4	3	2	1
na	5	4	3	2	1	Good Leadership at project level	na	5	4	3	2	1
na	5	4	3	2	1	Low staff turnover, stable teams are built	na	5	4	3	2	1
na	5	4	3	2	1	Proper planning	na	5	4	3	2	1
na	5	4	3	2	1	Realistic schedule	na	5	4	3	2	1
na	5	4	3	2	1	Risk management	na	5	4	3	2	1
na	5	4	3	2	1	Skilled and sufficient staff	na	5	4	3	2	1
na	5	4	3	2	1	Support from top management	na	5	4	3	2	1
na	5	4	3	2	1	Supporting tools – Project management and ALM tools such as DOORs, ClearCase, SVN and similar	na	5	4	3	2	1
na	5	4	3	2	1	Trustful and cooperative working environment exists.	na	5	4	3	2	1
na	5	4	3	2	1	Up-to-date progress reporting	na	5	4	3	2	1
na	5	4	3	2	1	User or client is involved while developing	na	5	4	3	2	1

2) Could you please mark the 2-5 factors you feel are the most important to project success?

3) How successful is your current project - 0 is a complete failure, 100 is perfect.

0 10 20 30 40 50 60 70 80 90 100

Thanks once again for your time – Margaret Gruen-Kerr

F.References

CMMI | Overview. Available online at http://www.sei.cmu.edu/cmmi/, checked on 22/10/2012.

Das V-Modell® - Home. Available online at http://www.v-modell.iabg.de/, checked on 11/11/2012.

Domain - Definition and More from the Free Merriam-Webster Dictionary. Available online at http://www.merriam-webster.com/dictionary/domain, checked on 14/12/2012.

SWEBOK - Overview. Available online at http://www.computer.org/portal/web/swebok/overview, checked on 20/11/2012.

Merriam-Webster (2011). Available online at http://www.merriam-webster.com, checked on 11/08/2012.

Abran, Alain; Bourque, Pierre; Dupuis, Robert; Moore, James; Tripp, Leonard (2004): Guide to the Software Engineering Body of Knowledge - SWEBOK. 2004th ed. Edited by Alain Abran, Pierre Bourque, Robert Dupuis, James Moore, Leonard Tripp: IEEE Press. Available online at http://www.worldcat.org/oclc/612153976.

Abran, Alain; Moore, James W. (2001): Guide to the software engineering body of knowledge. Trial version (version 0.95). 0th ed. Los Alamitos, CA: IEEE Computer Society, checked on 20/11/2012.

Agile Alliance (2012): Agile 2012. Agile 2012. Dallas, Texas, 13/8/2012 - 17/8/2012. Available online at http://www.serena.com/docs/repository/alm-benchmark/Serena-Agile-2012-Survey-Infographic.pdf, checked on 20/11/2012.

Ahern, Dennis M.; Clouse, Aaron; Turner, Richard (2008): CMMI distilled. A practical introduction to integrated process improvement. 3rd ed. Upper Saddle River, NJ: Addison-Wesley.

Ambler, Scott W. (2010): 2010 IT Project Success Rates. Ambysoft Inc., updated on 20/02/2013, checked on 20/02/2013.

Ardanuy Philip E.: Success factors and critical lessons learned implementing a repeatable CMMI research-to-operations systems engineering process for NESDIS/STAR. In : 91st American Meteorological Society Annual Meeting. Available online at https://ams.confex.com/ams/91Annual/flvgateway.cgi/id/17197?recordingid=17197.

Bartlett James E., Kotrlik Joe W. Higgins Chadwick C. (2001): Organizational Research: Determining Appropriate Sample Size in Survey Research. In *Information Technology, Learning and Performance* 19 (1), pp. 43–50. Available online at http://www.osra.org/itlpj/bartlettkotrlikhiggins.pdf, checked on 30/10/2012.

Beck, K.; Beedle, M.; van Bennekum, A.; Cockburn, A.; Cunningham, W.; Fowler, M. et al. (2001a): Manifesto for Agile Software Development. Agile Alliance. Wasatch mountains, Utah. Available online at http://www.agilemanifesto.org/, checked on 22/07/2012.

Beck, K.; Beedle, M.; van Bennekum, A.; Cockburn, A.; Cunningham, W.; Fowler, M. et al. (2001b): Principles behind the Agile Manifesto. Agile Alliance. Wasatch mountains, Utah. Available online at http://www.agilemanifesto.org/principles.html, updated on 7/08/2007, checked on 20/11/2012.

Belassi, Walid; Tukel, Oya Iemeli (1996): A new framework for determining critical success/failure factors in projects. In *International Journal of Project Management* 14 (3), pp. 141–151. Available online at http://www.sciencedirect.com/science/article/pii/026378639500064X.

9/09/2009: V-Modell® XT 1.3. Available online at http://v-modell.iabg.de/dmdocuments/V-Modell-XT-Gesamt-Englisch-V1.3.pdf, checked on 26/06/2011.

1.4, 2012: V-Modell XT Gesamt 1.4. Available online at http://www.v-modell-xt.de/, checked on 21/07/2012.

BMI (Ed.) (2012): V-Modell® XT 1.4. Referat IT 6 im Bundesministerium des Innern. 1.4th ed. Berlin, Germany: Bundesministerium des Innern (BMI), checked on 21/07/2012.

Boudreau, Marie-Claude; Gefen, David (2004): Validation Guidelines for IS Positivist Research. In *Communications of the Association for Information Systems* 13 (1). Available online at http://aisel.aisnet.org/cais/vol13/iss1/24.

Caralli, Richard A. (2004): The Critical Success Factor Method: Establishing a Foundation for Enterprise Security Management. Software Engineering Institute, Carnegie Mellon University. Pittsburgh, Pennsylvania, checked on 24/10/2012.

Caxton, W. (1484): Aesop's Fables. Available online at http://books.google.at/books?id=QP9YPAAACAAJ.

Chmiel, Nik (2009): An introduction to work and organizational psychology. A European perspective. 2nd ed. Malden, MA: Blackwell Pub.

CMMI Product Team (2011): CMMI® for Development, Version 1.3 CMMI-DEV, V1.3. Available online at http://www.sei.cmu.edu/reports/10tr033.pdf, updated on 1/09/2011, checked on 25/10/2012.

Cohen, Jacob (1988): Statistical power analysis for the behavioral sciences. 2nd ed. Hillsdale, N.J: L. Erlbaum Associates.

Cupp, J. William (2001): Reviewing the professionalization of software engineering: can small colleges remain viable? In *Journal of Computing in Small Colleges* 17 (1), pp. 132–146. Available online at http://dl.acm.org/citation.cfm?id=772488.772509.

Customer Relations SEI (Ed.) (2012): CMMI for Scampi Class A Appraisal Results. 2011 End-Year Update. Software Engineering Institute, Carnegie Mellon University. Available online at http://www.sei.cmu.edu/cmmi/why/profiles/upload/2012MarV3CMMI.pdf, checked on 17/11/2012.

DeMarco, Tom (1988): Looking For Lost Keys. In *Software Magazine* 8 (5), p. 58. Available online at http://search.proquest.com/docview/218438112?accountid=14682.

DeMarco, Tom (2012): All Late Projects Are the Same. In *IEEE Softw.* 29 (1), pp. 8–11. Available online at http://www.systemsguild.com/pdfs/DeMarcoNov2011.pdf.

DeMarco, Tom; Lister, Timothy R. (1999): Peopleware. Productive projects and teams. 2nd ed. New York, NY: Dorset House Pub.

Dymond, Kenneth M. (2007): A guide to the CMMI. Interpreting the Capability Maturity Model Integration. 2nd ed., rev. for CMMI for development, V1.2, with Dymond diagrams. Annapolis, Md: Process Transition International.

Entinex Inc (Ed.) (2012): CMMI FAQ - brutally honest, totally hip, no nonsense answers (a public service of Entinex). Available online at http://www.cmmifaq.info/#7.01, updated on 12/10/2012, checked on 6/11/2012.

Fortune, Joyce; White, Diana (2006): Framing of project critical success factors by a systems model. In *International Journal of Project Management* 24 (1), pp. 53–65. Available online at http://www.sciencedirect.com/science/article/pii/S0263786305000876.

Fowler, M. (2003): Swebok. Available online at http://martinfowler.com/bliki/Swebok.html, updated on 9/08/2012, checked on 11/08/2012.

Fowler, M.; Highsmith, J. (2001): The Agile Manifesto. Commentary, August 2001. Available online at http://www.pmp-projects.org/Agile-Manifesto.pdf, checked on 20/11/2012.

George, Darren; Mallery, Paul (2010): SPSS for Windows step by step. A simple guide and reference, 17.0 update. 10th ed. Boston: Allyn & Bacon.

Gliem, Joseph A.; Gliem, Rosemary R.: Calculating, Interpreting, And Reporting Cronbach's Alpha Reliability Coefficient For Likert-Type Scales. In : 2003 Conference (Columbus, Ohio : Ohio State University), pp. 82–88. Available online at http://hdl.handle.net/1805/344, checked on 13/12/2012.

Hihn, J.; Lewicki, S.; Morgan, S. (2011). In : System Sciences (HICSS), 2011 44th Hawaii International Conference on, title=Bootstrapping Process Improvement Metrics: CMMI Level 4 Process Improvement Metrics in a Level 3 World, pp. 1–10. Available online at http://ieeexplore.ieee.org/ielx5/5716643/5718420/05718860.pdf?tp=&arnumber=5718860&isnumber=5718420.

Hillel Glazer, Jeff Dalton David Anderson Mike Konrad &. Sandy Shrum (2008): CMMI® or Agile: Why Not Embrace Both! Available online at http://www.sei.cmu.edu/reports/08tn003.pdf, updated on 12/11/2008, checked on 26/10/2012.

Höhn, Reinhard (2008): Das V-Modell XT. Anwendungen, Werkzeuge, Standards. Berlin, Heidelberg: Springer (Springer-Lehrbuch).

Humphrey, Watts S. (2006): CMMI Executive Review. Available online at http://www.sei.cmu.edu/library/assets/cmmi-exec-overview06.pdf, checked on 9/11/2012.

Humphrey, Watts S. (2009): The Watts New Collection: Columns by the SEI's Watts Humphrey. Software Engineering Institute, Carnegie Mellon University. Pittsburgh, Pennsylvania. Available online at http://www.sei.cmu.edu/reports/09sr024.pdf, updated on 19/07/2012, checked on 20/11/2012.

Jamieson, Susan (2004): Likert scales: how to (ab)use them. In *Med Educ* 38 (12), pp. 1217–1218.

Kaner, Cem (2012): SWEBOK Problems, Part 2. Available online at http://kaner.com/?p=27, updated on 22/07/2012, checked on 22/07/2012.

Kiesling, Stephan (2011): Integration agiler Entwicklungszyklen in das V-Modell auf Basis von Informationsflüssen. Available online at http://www.se.uni-hannover.de/pub/File/pdfpapers/Kiesling2011.pdf, updated on 10/11/2011, checked on 13/03/2013.

Kraft TM (2010): CMMI in a Geographically Distributed Organization. UNISYS, checked on 19/11/2012.

Larman, C.; Basili, V.R (2003): Iterative and incremental developments. a brief history. In *Computer* 36 (6), pp. 47–56.

Larsen, M. A.; Myers, M. D. (1999): When success turns into failure: a package-driven business process re-engineering project in the financial services industry. In *The Journal of Strategic Information Systems* 8 (4), pp. 395–417. Available online at http://www.sciencedirect.com/science/article/pii/S0963868700000251.

Lindvall, Mikael; Basili, Victor R.; Boehm, Barry W.; Costa, Patricia; Dangle, Kathleen; Shull, Forrest et al. (2002): Empirical Findings in Agile Methods. In : Proceedings of the Second XP Universe and First Agile Universe Conference on Extreme Programming and Agile Methods - XP/Agile Universe 2002. London, UK, UK: Springer-Verlag, pp. 197–207. Available online at http://dl.acm.org/citation.cfm?id=647276.722332.

M.C. Paulk (2009): A History of the Capability Maturity Model for Software. In *ASQ Software Quality Professional* 12 (1), pp. 5–19. Available online at http://home.comcast.net/~mark.paulk/papers/p2009c.pdf.

MacCormack, Alan (2001): Product-Development Practices That Work: How Internet Companies Build Software. In *MIT Sloan Management Review* 42 (2), pp. 75–84.

Mary Ann Lapham, Ray Williams Charles (Bud) Hammons Daniel Burton Alfred Schenker (2012): Considerations for Using Agile in DoD Acquisition. Available online at http://www.sei.cmu.edu/reports/10tn002.pdf, updated on 3/10/2012, checked on 17/11/2012.

Nasir, Mohd Hairul Nizamand; Sahibuddin, Shamsul (2011): **Critical Success Factor for Software Projects: A Comparative Study. In *Scientific Research and Essays* 6 (10), pp. 2174–2186. Available online at http://academicjournals.org/sre/PDF/pdf2011/18May/Nasir%20and%20Sahibuddin.pdf, checked on 29/07/2012.

Norton, Darell (2003): An Agile Review of the Software Engineering Body of Knowledge (SWEBOK) | Darrell Norton. Available online at http://codebetter.com/darrellnorton/2003/07/09/an-agile-review-of-the-software-engineering-body-of-knowledge-swebok/, checked on 20/11/2012.

Ó Conchúir, Deasún (2011): Overview of the PMBOK® guide. Short cuts for PMP® certification. In *Springer eBooks*. Available online at https://univpn.univie.ac.at/+CSCO+10756767633A2F2F6A6A6A2E66636576617472657976617 82E70627A++/content/nxwv37221hn81853/fulltext.pdf.

Paulish, D.J; Carleton, A.D (1994). In *Computer, title=Case studies of software-process-improvement measurement* 27 (9), pp. 50–57.

Peters, Geoffrey; Fortune, Joyce (2003): The Formal system paradigm for studying failures.

Podsakoff, Philip M.; MacKenzie, Scott B.; Lee, Jeong-Yeon; Podsakoff, Nathan P. (2003): Common method biases in behavioral research: A critical review of the literature and recommended remedies (5). Available online at http://www.personal.psu.edu/jxb14/M554/articles/Podsakoffetal2003.pdf, checked on 1/11/2012.

Project Management Institute (2004): A guide to the project management body of knowledge. (PMBOK guide) ; an American national standard, AN SI/PMI 99-001-2004. 3rd ed. Newton Square, Pa: Project Management Institute.

Reifer Donald J. (2007): Profiles of Level 5 CMMI Organizations. In *CrossTalk, the Journal of Defense Software* 20 (1), pp. 24–28. Available online at http://www.crosstalkonline.org/storage/issue-archives/2007/200701/200701-0-Issue.pdf, checked on 9/11/2012.

Rockart, John F. (1979): Chief Executives Define Their Own Data Needs. In *Harvard Business Review* (Mar 01, 1979).

Royce, W. W. (1987): Managing the development of large software systems: concepts and techniques. In : Proceedings of the 9th international conference on Software Engineering. Los Alamitos, CA, USA: IEEE Computer Society Press (ICSE ’87), pp. 328–338. Available online at http://dl.acm.org/citation.cfm?id=41765.41801.

Ruzhi Xu, Yunjiao Xue Peiyao Nie Yuan Zhang Desheng Li (2006): Research on CMMI-based Software Process Metrics. In : Computer and Computational Sciences, 2006. IMSCCS ’06. First International Multi-Symposiums on, title=Research on CMMI-based Software Process Metrics, vol. 2, pp. 391–397. Available online at http://ieeexplore.ieee.org/ielx5/10901/4673661/04673736.pdf?tp=&arnumber=4673736&isnumber=4673661.

Schaeffer, Mark D. (1998): Capability Maturity Model Process Improvement. Software Engineering Institute, Carnegie Mellon University. Pittsburgh, Pennsylvania. Available online at http://www.sei.cmu.edu/library/abstracts/news-at-sei/backgroundsept98pdf.cfm, checked on 24/10/2012.

Schmitt, Neal (1994): Method Bias: The Importance of Theory and Measurement. In *Journal of Organizational Behavior* 15 (5), pp. 393–398. Available online at http://www.jstor.org/stable/pdfplus/2488211.pdf?acceptTC=true.

Standish Group International (2010): Chaos. Technical Report.

Surowiecki, James (2004): The wisdom of crowds. Why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations. 1st ed. New York: Doubleday.

Wateridge, John (1995): IT projects: a basis for success. In *International Journal of Project Management* 13 (3), pp. 169–172. Available online at http://www.sciencedirect.com/science/article/pii/026378639500020Q.

Weinberg, Gerald M. (1997): Quality software management. New York, N.Y: Dorset House Pub.

White, Diana; Fortune, Joyce (2009): The project-specific Formal System Model. In *International Journal of Managing Projects in Business* 2 (1), pp. 36–52. Available online at http://search.proquest.com/docview/232628259?accountid=14682.

Wohlin, Claes; Andrews, Anneliese Amschler (2001): Assessing Project Success Using Subjective Evaluation Factors. In *Software Quality Journal* 9 (1), pp. 43–70. Available online at http://search.proquest.com/docview/235098813?accountid=14682.

Wohlin, Claes; Andrews, Anneliese Amschler (2003): Prioritizing and Assessing Software Project Success Factors and Project Characteristics using Subjective Data. In *Empirical Software Engineering* 8 (3), pp. 285–308.

Wysocki, Robert K. (2007): Effective project management. Traditional, adaptive, extreme. 4th ed. Indianapolis, Ind: Wiley.

Zahran, Sami (1997): Software process improvement: practical guidelines for business success,: Addison-Wesley.

Zwikael, Ofer; Globerson, Shlomo (2006): From Critical Success Factors to Critical Success Processes. In *International Journal of Production Research* 44 (17), pp. 3433–3449.

G. Lebenslauf

Contact M.GruenKerr@gmail.com

Veltlinerstrasse 1/39

2353 Guntramsdorf, tel. +43 (0)664 550 40 83

born 14.07.1960

3 Children (1990,1993,1995)

Fields of Interest Application Lifecycle Process improvement,

Software Quality Assurance, Requirements Testing, User Acceptance Testing,

End User Training, Third Level support

Education 2009-(planned Winter 2012) **Masters Information Management** University of Vienna Faculty of Computer Science.

Thesis: „Critical Success Factor in Application Lifecycle Management, a Case Study at Kapsch TrafficCom

Honours Bachelor of Arts in Computer Science and Economics

York University Toronto

Foundation Level Testing (AST, Oct. 2008)

IPICS Program for Internet and Communication Security (July 2010)

ISTQB Foundation Level Certification (July 2012)

Work Experience

5/2009 – 4/2010 **Quality Assurance Manager**, Fatfoogoo Vienna, a PCI certified

Online Payment Platform specialized in on-line games.

- Introduced QA and Software Testing to the company
- Performed Risk analysis
- Prepared for PCI-Certification
- Soap/WSDL testing
- Test Automation (SoapUI).
- Responsible for one further tester.

3/2003 - 3/2009 **Quality Assurance Engineer** Altova Vienna

- responsible for XMLSpy und AltovaXML testing
- W3C - XML Technologies Specification testing
- Automation of „compliance tests“ in C# and Java
- Black box tests, scenario tests und performance tests

8/2001 - 11/2002 **Software Developer**, Spardat Vienna (TeleBanking)

- Using C and SQL on a VAX/VMS, Oracle/RDB database to perform support work for a TeleBanking server.
- Used CPP and Sybase under Windows 98/NT/XP to perform support for a TeleBanking client.
- Provided Edifact 3rd Level support

9/1998 - 4/2001 **Software Developer** COLD Project (Bank Archives)

Contractor for Compaq (prev. DEC)

- Performed maintenance programming in C and SQL on a VAX/VMS with an Oracle/RDB database
- Data Archive Support for the Bank Austria

7/1990 - 8/1998 Leave (3 children 1990, 1993, 1995)

1985-1990 **Technical/Industrial Program Development**

- C Programmer on VAX/VMS implementing various mainly industrial oriented projects. Among other a control system for the Heizbetriebe Wien, and a circuit board placement program for Rhode & Schwarz and a user interface for crash tests for BMW.
- Statistical data analysis: IIASA Laxenburg (Unix , Fortran)
- Technical Writing „DOS User Manual" Phillips Austria

Languages English and German bilingual.

Skills ISTQB and AST Foundation Level Software Testing

Programming Languages

XSLT, XQuery, C/CPP, SQL (good)

Fortran, Java, VB.NET, C# (known)

Software:

XMLSpy, Mapforce, DatabaseSpy, AltovaXML,

Eclipse, VS.NET, CVS, ANT, Seapine TestTrackPro, Clear Case,

Clear Quest Doors

Databases:

Access, MySQL, IBM DB/2, Oracle/RDB

XML and other Standards:

XML, XML Schema, Edifact, X12, XBRL

References Prof. Dr. Gerald Quirchmayr

University of Vienna , Faculty for Computer Science 1090 Wien, Währinger Straße 29

T:+43-1-4277-78811
gerald.quirchmayr@univie.ac.at

Dipl. Ing. Arezoo Edrisian

R&D Cluster Manager of Roadside Equipment

Office +43 (0) 50 811 2290 arezoo.edrisian@kapsch.net