Transformational Leadership in Software Projects

MARYAM MOUSAVIKHAF

Master of Science Thesis

Stockholm, Sweden 2013
Transformational Leadership

In

Software Projects

By

Maryam Mousavikhah

Master of Science Thesis INDEK 2013:05
KTH Industrial Engineering and Management
Industrial Management
SE-100 44 STOCKHOLM
Master of Science Thesis INDEK 2013:05

Transformational Leadership
in
Software Projects

Maryam Mousavikhah

<table>
<thead>
<tr>
<th>Approved</th>
<th>Examiner</th>
<th>Supervisor</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013-02-05</td>
<td>Johann Packendorff</td>
<td>Johann Packendorff</td>
</tr>
<tr>
<td></td>
<td>Commissioner</td>
<td>Contact person</td>
</tr>
<tr>
<td></td>
<td>Johann Packendorff</td>
<td>Johann Packendorff</td>
</tr>
</tbody>
</table>
Abstract

Lack of management in software projects is among the most important reasons for the failure of this kind of projects. Considering this fact, in addition to high rate of IS (Information System) projects’ failure, and the lack of leadership studies in IS field, it is necessary to pay more attention to the concept of leadership in software projects. Transformational leadership as one of the most popular leadership theories, although might bring specific advantages for this kind of projects, has not been outlined in this field of study. Therefore this study has tried to understand the meaning of transformational leadership in software projects and outline corresponding ideal pattern of this concept.

Taking into account unique potentials qualitative research has in relation to research questions, Myers model of research design is selected to found whole study on. This model is especially designed to be used in business and management researches. Myers research model has five blocks. First block is related to philosophical assumption which for our study is interpretive. Second block is related to research method which is case study inspired by grounded theory, in our study. Data collection technique, the third block, is semi-structured interview. The fourth block which concerns data analysis approach is grounded theory encompassing analytic induction for our study. Finally the fifth block, written record, took the form of thesis report in current research study.

This study has found required characteristics and behaviors of software project managers and drawn the lines of relationship between those attributes and transformational leadership. Comparing materialization of transformational leadership in software projects with the theory, led to finding of nearby transformational leadership or engaging leadership model as the ideal pattern of such concept in software projects. Finally shortages of that model in relation to software domain are investigated and consequent suggestions are provided.
Keywords

Transformational leadership, nearby transformational leadership, engaging leadership, Software Projects, Software Project Management
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1. Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2. Problem Description</td>
<td>4</td>
</tr>
<tr>
<td>1.3. Aim</td>
<td>5</td>
</tr>
<tr>
<td>1.4. Delimitations</td>
<td>6</td>
</tr>
<tr>
<td>1.5. Disposition of the Report</td>
<td>7</td>
</tr>
<tr>
<td>2. Method</td>
<td>8</td>
</tr>
<tr>
<td>2.1. Scientific Research in General</td>
<td>8</td>
</tr>
<tr>
<td>2.2. Philosophical Assumptions</td>
<td>10</td>
</tr>
<tr>
<td>2.3. Research Method</td>
<td>12</td>
</tr>
<tr>
<td>2.4. Data Collection Technique</td>
<td>17</td>
</tr>
<tr>
<td>2.5. Data Analysis Approach</td>
<td>19</td>
</tr>
<tr>
<td>2.6. Written Record</td>
<td>21</td>
</tr>
<tr>
<td>2.7. Summary of Selected Research Design</td>
<td>22</td>
</tr>
<tr>
<td>2.8. Research Procedure</td>
<td>23</td>
</tr>
<tr>
<td>3. Background</td>
<td>24</td>
</tr>
<tr>
<td>3.1. Leadership Approaches</td>
<td>24</td>
</tr>
<tr>
<td>3.2. Transformational Leadership</td>
<td>28</td>
</tr>
<tr>
<td>3.3. “Nearby Transformational Leadership” or “Engaging Leadership”</td>
<td>32</td>
</tr>
<tr>
<td>3.4. Suitability of Nearby Transformational Leadership in relation to Research Field</td>
<td>37</td>
</tr>
<tr>
<td>3.5. Software Projects and Management</td>
<td>39</td>
</tr>
<tr>
<td>3.6. Case Selection Criteria</td>
<td>43</td>
</tr>
<tr>
<td>3.7. Selection Method</td>
<td>47</td>
</tr>
</tbody>
</table>
4. Results and Analysis ............................................................................................ 49

4.1. Required Characteristics and Behaviors of Software Project Managers .......... 49

4.2. Required Characteristics and Behaviors in Relation to Leadership Theories ....... 59

4.3. Conclusion ................................................................................................................. 67

4.4. Discussion .................................................................................................................. 71

5. References ............................................................................................................. 73
Acknowledgement

First of all, I would like to thank my supervisor who patiently helped me to accomplish different steps of the project one by one. Although sometimes it seemed almost impossible to handle some parts of the study, his positive attitudes and influential suggestions smoothen the path.

Second, I would like to thank my husband, Alireza, who has helped me not only by his intellectual recommendations but also by his continuous devotions, without which my goals and desires would not be achievable at all.
Table of Figures and Tables

Figure 2.1: Research Design Model (Myers, 2009, p.23) .............................................................. 9
Figure 2.2: Selected Philosophical Assumption ........................................................................ 11
Figure 2.3: Selected Research Method .................................................................................... 13
Figure 2.4: Selected Data Collection Technique ...................................................................... 18
Figure 2.5: Selected Data Analysis Approach .......................................................................... 20
Figure 2.6: Selected Written Record ........................................................................................ 21
Figure 2.7: Selected Research Design ..................................................................................... 22
Figure 2.8: Research Procedure ............................................................................................... 23
Figure 3.1: Nearby Transformational Leadership Model ........................................................... 33
Figure 3.2: Software Project Management Competences Pyramid (Rose et al., 2007, p.616) ........................................................................................................................................ 40
Table 3.1: Criteria of Selected Cases ......................................................................................... 48
Table 4.1: Required Attributes of Software Project Managers in relation to Nearby Transformational Leadership ........................................................................................................... 66
1. Introduction

1.1. Background

“Like love, leadership continued to be something everyone knew existed but nobody could define” (Bennis & Nanus, 1985, p.23)

This statement is still valid. Large number of new leadership theories is evident of this claim. Just in last 2 years more than 3000 articles are published with the subject of leadership.

First attempts to define this concept returns to 1800s. (Fairholm & Fairholm, 2009) Majority of studies in this field focused on how the leader should be and tried to define his required attributes in different aspects. Trait approach defines the leader as a person with specific inborn characteristics. Scientists in behavioral or style approach believe that everybody could be a leader by treating according to certain behavioral patterns. (Bolden et al., 2003) Contingency approach considers leadership required behaviors dependent to the situation. A wave of new leadership theories started in 1980s. They all emphasized on the role of manager as a vision creator and disseminator; in spite of the divergences different theories under this approach have. (Parry & Bryman, 2006) Contemporary generation of leadership theories called post-charismatic or post-transformational is assumed to amend flaws in previous theories. (Bolden et al., 2003) Today the emphasis is moved from the leader to the leadership. Emergence of new theories such as distributed leadership, shared leadership and collaborative leadership is the result of such trend. (Crevani et al., 2010) It is noticeable that earlier perspectives still live and new theories are complementary to them.

Leadership in general has attained a distinctive position of being considered as a key contributor to action processes in organizations. To achieve such a performance academy is responsible to continuous introduction of appropriate practices to industry. (Crevani et al., 2010) Information technology as one of the challenging business contexts (Phillips, 1998) in today’s world is not exception to this requirement. Employing best leadership theories could be a remedy to high rate of IS (Information
System) projects’ failure (Flinn, 2010). Based on Thite (2000) there is a strong linkage between failure of an IS project and leadership deficiencies.

In study of management in software projects there is a delicate point which is related to the distinction between management and leadership. Smith (1999) in an article with title “Project leadership: Why project management alone doesn't work” states that it is not the lack of project management which usually leads to the failure of projects but it is the lack of project leadership. Manager is the one whom team members are forced to follow but the leader is whom they want to follow. Management is a function which every business must exercise, but the leadership is a relationship which if exercised; strengthens the team’s vigor. (Maccoby, 2000) Despite the emphasis on the need of project leadership and non-technical competencies in software projects (Pulk, 1990) IS academy has paid little attention to subject of leadership. (Klenke, 1996)

Although project leadership as a distinctive leadership assignment has been under researchers’ attention, (Lindgren & Packendorff, 2009) also different attempts has been performed to describe ideal profiling of project managers, (Muller & Turner, 2010) this is not always acceptable to transfer experiences of one problem domain to the other in general (Kurbel, 2008) and this is often wrong to transfer experiences of other engineering domains to software projects in particular since software projects have distinctive specifications in comparison to the other engineering domains (Paynter & Ahmed, 2000) Such a transference is one of the underlying reasons of software development projects’ failure. (Kurbel, 2008) As a result the need for comprehensive researches on leadership in software projects is evident.

Through recently popular theories of leadership is transformational leadership which is however for the first time proposed at 1978 (Burns) is still under researchers’ attention. Even today transformational leadership concepts and principles are sources of the inspiration for newly emerged leadership patterns such as nearby transformational or engaging leadership. (Alimo-Metcalfe & Alban-Metcalfe, 2001) Although transformational leadership is suggested as a universal paradigm; there is a need to precisely understand what it means in different problem domains. (Bass, 1999) How it could and should be practiced by various professionals. However, suitability of
transformational leadership style for software project management is admitted by Thite and Simmons (1997) and Thite (2000) there is no pragmatic sketch of such style in IS field.
1.2. Problem Description

Existing thesis focusing on the transformational leadership practices tries to explain

- What does transformational leadership mean in software projects?
- What is the ideal pattern of transformational leadership in software project domain?

Profiling of a leader is explanation of idiosyncratic combination of his behavioral, temporal, emotional and mental attributes (Muller & Turner, 2010). First question of this research would be answered by some kind of profiling for transformational leaders in software projects, i.e. describing those attributes of leader which could be considered as instances of transformational leadership materialization. Looking for a proposed model of transformational leadership which has the most congruency with perceived definition of transformational leadership in software projects and distinguishing its disagreements and shortcomings would provide answer for the second question of this research.
1.3. Aim

The aim of this thesis is to gain deep insight into specifications of leadership in software projects. This would enable professionals to have an outline of what they should do as managers of software projects and what aspects of their personality they have to strengthen or weaken.

For researchers, this thesis provides a basis to found their studies about leadership practices in software projects.
1.4. Delimitations

This study is restricted to the selection of a limited number of cases. Such a limitation stresses the needs of examining attained results amongst software projects with divergent criteria.

Another restriction of the study which again is related to the data gathering is that underlying data is gathered based on team members’ perspectives; although any software project has different stakeholders including client, managers of software organizations etc. Furthermore, this study does not differentiate between various expert groups in software projects but it is plausible that transformational leadership materialization is not exactly the same in standpoints of diverse expert groups. Barr & Tessler (1998) says that software team is the combination of various groups of experts comprising analysts, programmers, testers, system architects etc which might have different professional requirements.
1.5. Disposition of the Report

Chapter 1: This chapter starts by a short description of the research area and continues by exact statement of the research problem and the aim of this study. Finally delimitations this study has been confronted with are mentioned.

Chapter 2: This chapter explains the research method of the study encompassing five main sections: philosophical assumption, research method, data collection technique, data analysis approach and written record. This chapter also contains a discussion about the validity of the research. At the end of this chapter a diagram illustrates different steps carried out to accomplish this research project.

Chapter 3: This chapter is dedicated to more precise description of the research area. Such a description includes a review on leadership theories in general, a study of transformational leadership and an accurate definition of nearby transformational leadership. Then, current status of management researches in context of software projects is explored. After that, suitability of engaging leadership model in relation to software projects is investigated and finally the criteria to be considered during selection of sample cases are presented.

Chapter 4: This chapter discusses the results acquired during this research study. First of all, required characteristics and behaviors of software project managers are explained then their required characteristics and behaviors in relation to leadership theories are clarified. This chapter continues with the conclusion part which provides a complementary discussion about outcomes of the research. Lastly, the discussion part by reviewing impacts of limitations on current study and introducing potential subjects in the field, suggests future researches.
2. Method

2.1. Scientific Research in General

Scientific research could be accomplished using two main approaches: qualitative and quantitative. Although some research questions could be responded using both types of approaches there are other research questions which imposes use of either qualitative or quantitative. When the research intends to understand a phenomenon without having a hypothesis or presumption about the final result, qualitative approach is what suits to probe the research problem. (Mack et al., 2005) Qualitative approach is especially beneficial when the researcher tends to underpin experiences, behaviors and emotions in addition to organizational functioning. (Strauss & Corbin, 1998)

Considering our research goal which is to understand transformational leadership behaviors of software project managers, also the fact that we don’t have any presumption what the final result of the research would be qualitative approach is the achievable option.

Taking into account that rigorousness of qualitative research is dependent on the use of a scientific research model, (Myers, 2009) it is essential to select a research model based on which different aspects of the study are directed. Accordingly we found our research on the qualitative research design model suggested by Myers. (2009) This model is intentionally developed to be used to carry out qualitative research in business and management. However, it doesn’t necessarily mean that business and management are the only fields in which it could be favorable. This model provides a clear roadmap of whole steps to be passed among research project. This is an obligatory specification of research designs. (Myers, 2009)

This model has five blocks, namely philosophical assumption, research method, data collection technique, data analysis approach and written record; each of which would be explained in following sections. It should be noticed that although the model consists of five different steps each one comes after the other, they could be done iteratively as well. (Myers, 2009)
Figure 2.1: Research Design Model (Myers, 2009, p.23)
2.2. Philosophical Assumptions

A compulsory prerequisite of every qualitative research is determining its philosophical perspective. Philosophical perspective or assumption, sometimes called philosophical paradigm, specifies the researchers’ viewpoint about how the knowledge could be acquired to arrive at the answer of the research question. (Myers, 2009)

Although there exist a number of different classifications for the philosophical assumption, Chua (1986) have defined three epistemologies for research, namely positivism, interpretive and critical studies. Positivism epistemology usually concerns the predictive understanding of a phenomenon or testing a theory using structured instruments. Interpretive researchers try to understand a phenomenon by studying the meanings people assign to that. Researchers of this epistemology look at the phenomenon through the eyes of their participants instead of enforcing them a priori defined perception of the subject under study. Critical researchers attempt to analyze and critique current situation. Such attempts usually lead to clarifying the contradictions and conflicts in social practices.

Interpretive epistemology has unique benefits for our research. First reason is that, this epistemology is appropriate to understand complex, dynamic and social phenomenon which could be context-dependent. (Orlikowski & Baroudi, 1991) Our research intends to find attributes of transformational leaders in a specific context, namely software projects. Second reason is that, through such epistemology we can arrive at the answer of research question free of any presumption. We don’t need to rely on previously accepted set of suppositions. It gets more importance considering the lack of similar researches in our research context. In simple words, if instead of interpretive epistemology any other one has been selected; we had to take research assumptions from other contexts.
Therefore the chance of a holistic result could be lessened.

Figure 2.2: Selected Philosophical Assumption
2.3. Research Method

Research method is a ‘strategy of enquiry’ (Myers, 2009, p.24), the way to find empirical data in the real world. When indicating the research method one thing to be determined is the unit of analysis which could be an individual, an event, a group, organization etc. Site selection is another activity to be performed during research method definition.

Among different research methods is the case study. Case study method has recently attracted the management researchers’ attention. (Siggelkow, 2007) Based on Eisenhardt (1989, p.534) case study is ‘a research strategy which focuses on understanding the dynamics present within single settings. Case studies are useful to generate knowledge with regard to management field. (Leonard-Barton, 1990) Case study has the distinctive advantage of enabling researcher to interrogate the data within its real context, (Yin, 1984) where a relationship, process, activity etc take place. The purpose of case study could be of two kinds. One is to test a theory and the second is to generate it. (Lee et al., 2010)

Grounded theory on the other hand is very helpful in developing new concepts and theories. It facilitates the emergence of theory or new concepts grounded in empirical data. There is a major difference between grounded theory and other methods. That difference is the emphasis on the continuous interplay between data collection and the analysis. Grounded theory is extensively used by business and management researchers as a way of data analysis. (Myers, 2009) Although this is very fundamental principle of grounded theory not to have a prejudice about the results of data collection and analysis, from following statement of Strauss & Corbin (1998, p.48) this is evident that it is helpful to conduct a literature review before starting the data collection: “The researcher brings to the inquiry a considerable background in professional and disciplinary literature... [But] the researcher does not want to be so steeped in the literature that he or she is constrained and even stifled by it.”

For our research, the case study inspired by grounded theory is a suitable choice. First reason is that, it is the first time to find a meaning for transformational leadership
in software field. Second reason is that, case study is very useful to understand a phenomenon and its context in-depth. (Darke et al., 1998) Third reason lies in the specific advantages grounded theory principles could bring for our study. Considering our research question, without knowing anything about transformational leadership it is impossible to seek for such a concept in software projects and to ask relevant questions in data collection phase. In addition to this, unique techniques grounded theory suggests, facilitates and accredits our data analysis.

![Research Method: Case Study Inspired by Grounded Theory](image)

**Figure 2.3: Selected Research Method**

In spite of the advantages case study may bring for our research, there always have been debates about its rigor. (Zainal, 2007) There are four criteria which are usually mentioned as a basis for measuring the rigor of a research. (Gibbert et al., 2008) They are internal validity, construct validity, external validity and reliability.

On the other hand regarding the validity of the qualitative research, positivists usually asks about its trustworthiness. This is what naturalistic investigators answer by their terminology. Among them is Guba (1981) whom his suggested constructs are admitted by many scientists. He has proposed a construct which is just same as four criteria to measures the validity of case study. (Shenton, 2004) They are:

- Credibility (as a substitute to internal validity)
- Transferability (as a substitute to external validity or generalisability)
- Dependability (as a substitute to reliability)
- Confirmability (as a substitute to objectivity)

Here these four criteria are investigated for this research.

**2.3.1. Validity of the research**
**Internal Validity**

Internal validity or credibility (Riege, 2003) is related to the data analysis phase and the level of logic exists between data and attained results. (Gibbert et al., 2008)

To achieve internal validity we have tried to make use of a well-known data analysis approach which would be explained in coming section and also reflect our lines of reasoning in the results and analysis part. One strategy which could be used to attain this kind of validity is pattern matching. (Gibbert et al., 2008) It means predicting the pattern of results and comparing results with such a predicted pattern. We have used this strategy not exactly as its suggested but in a reasonable way. (Burke, 1997) Since we wanted not to prejudge the results we have not predicted any pattern in results; but after completing the data analysis phase, a comprehensive literature study is performed to find whether results of study are similar to any existing definition of transformational leadership. Here the answer was yes. The results were very similar to the definition of nearby transformational leadership. Then we tried to find whether such a definition could be true in software field and again the answer was yes. Therefore we can claim that we have applied this strategy by some means.

**Construct Validity**

Construct validity or confirmability (Riege, 2003) which should be considered in the data collection phase is the extent to which the researcher has neutrally and accurately observed the reality. (Gibbert et al., 2008)

Regarding this kind of validity and to be more precise, to be neutral, we have tried to develop a set of criteria for selection of the cases. These criteria helped to select appropriate cases. After finding cases with appropriate criteria, interview sessions are held which started with very general questions and only based on interviewees’ answers, more detailed questions were asked. In other words any question has been the child of interviewee’s response to the previous question. Furthermore by writing short memos of interviewees’ answers we have made sure to clarify all aspects of their perspective even if they have been pointed to something very quickly and briefly. This
procedure of handling interview sessions together with systematic way of case selection, guaranteed both the accuracy and the neutrality of data gathering.

Another strategy we have used to assure the confirmability of the data was that even after data collection we have made another literature review in which we were trying to find whether the facts mentioned by our respondents was the same for other software projects. If the answer of this question was yes, the confirmability of the findings was strengthened. Furthermore, during interviews we frequently asked why and how questions. This kind of questions by putting respondent’s ideas in front of him gave him the chance of rethinking about what he has been said which usually led to more enriched answers and sometimes correcting previous answers. Finally we have recorded all interview sessions and take notes of each interview. This is what suggested as a technique to increase construct validity. (Riege, 2003)

**External Validity**

External validity or generalizability (Gibbert et al., 2008) or transferability (Riege, 2003) refers to the capacity of research to generate same results in similar cases. This kind of validity which is related to the sampling issue in qualitative studies could be handled by setting a rationale for the choice of samples and enough description of the samples being studied. (Cook & Campbell, 1979)

To achieve external validity we have done an extensive literature study on different conditions which might impact the emergence of transformational leadership behaviors and management of software teams. Based on the results a set of criteria has been established as a basis for selection of the samples. This set of criteria appears at the end of background chapter. In addition the value of each criterion for selected samples is mentioned.

**Reliability**

Reliability or dependability (Riege, 2003) is about the process of research execution. Attaining reliability needs transparency in explaining procedures used during the research also enough provision of collected data. (Gibbert et al., 2008)
At the end of this chapter we’ll present a detailed explanation about procedure of conducting the research study. Furthermore in the results part we will demonstrate the results of data collection phase. Records of interviews are available for further investigations as well.
2.4. Data Collection Technique

Data collection technique is related to the way the empirical data is going to be gathered. Data collection techniques for a specific research project could be one or a combination of more. Data collection techniques include interview, fieldwork (participant observation), using documents etc (Myers, 2009).

Interview is the most used technique to collect data in qualitative researches. In general interviews are classified into structured, unstructured and semi-structured. Unstructured and semi-structured interview could be used in a qualitative research. The key advantage of interview in qualitative research is its flexibility. In qualitative research the emphasis should be on generality in the formulation of initial research questions and interviewee’s point of view. Asking questions based on respondent’s answers enable researcher to attain deeper insight into the interviewee’s world. (Bryman, 2012)

Turner (2010) has suggested a set of techniques to assist researchers conduct effective interviews. First is to implement a pilot test. Pilot test will disclose deficiencies and limitations with the interview design. Second is to construct effective questions which includes open-ended wording, asking neutral questions, asking questions one at a time and asking questions appropriate to respondent’s culture and business environment. The third is asking follow up questions. It means that researcher should be flexible and ready to ask follow-up questions.

Following these techniques, first we designed a set of questions to be asked during interview and conducted a pilot interview. This pilot interview showed that questions were too complex, sometimes difficult to understand for the respondent and also more detailed than required. Being more detailed than required limited the quality and the amount of acquirable data. Hence a new set of questions including very general questions designed and it is decided to ask detailed questions just based on respondent’s answers to those general questions. During the interview keywords mentioned by respondents were written down and placed as the subject of subsequent questions one by one. Here is an instance of general questions designed for the interview however
sometimes just first question is asked and other questions take another form or sequence based on respondent’s observed interests.

1. How do you describe your manager’s personality?
2. How does such a personality affect your work and performance?
3. How is your relationship with your manager? How does it affect your work?
4. How does your manager increase your motivation?

We can conclude that the data is collected using semi-structured interviews.

Figure 2.4: Selected Data Collection Technique
2.5. Data Analysis Approach

Data analysis approach indicates how gathered data would be analyzed to generate the answer of the research question. In spite of being two different steps in the research, the data collection and the data analysis, they usually are not used in such a sequential manner. This is usual; especially in use of iterative research methods such as grounded theory. (Myers, 2009)

Grounded theory could effectively be used for systematical analysis of the data. (Myers, 2009) Myers explains the process of conducting grounded theory after the data collection is completed. This process consists of three stages:

1. Open coding: The text should be analyzed and summarized in succinct codes. Codes should have descriptions which reveal underlying concept. They should constantly be compared with newly emerged ones and to find similarities and differences. The result is a set of refined codes. Finally codes have to be placed in different categories based on their similarities.

2. Axial coding or selective coding: Categories should be interpreted to find relationships between them.

3. Theoretical coding: after interpreting categories, the theory or the main gist of the data is formulated.

We have started the data analysis phase by accurate listening to each interview record and take memos of that. Based on those memos initial codes are produced. This task is repeated for each interview whilst during code production for every new interview, codes of previous interviews have been reviewed to prevent redundant code production. Sometimes previous codes were edited with new codes since new interview enabled us to select better codes for implied concept. Better code means a better reflection of the meaning in the interviewee’s statement. While listening to new interviews we were sensitive to contradictions between new interviewee’s statements and previous ones; if any contradiction was found, previous codes have been reassessed
and sometimes codes which were related to meanings rejected by others have been omitted. At last, all codes are summarized into different categories. At this stage we established our main category which was the transformational leadership. Other categories which are placed under this category were the list of different temperamental, behavioral, emotional and mental attributes emerged under the main category. To formulate the results we conducted another literature review. This literature review had two purposes. The first one was to make sure that what we have found as transformational leadership is admitted as same concept in literature and second one was to find the closest definition to ours. This literature review ends to a brilliant outcome. We found engaging leadership model, one of the recent definitions of transformational leadership which could very well acknowledge the correctness of our findings however very few dissimilarities and disagreements between that definition and ours should not be ignored.

The notion of abduction very well explains our approach of analysis. This notion which is called analytic induction in grounded theory is the process of constant interplay between induction and deduction, movement between empirical data and theory as and when required. Scientists found that what results in new ideas is not pure induction or pure deduction but a combination of them. This is a well-known notion of data conceptualization in grounded theory. (Suddaby, 2006)
2.6. Written Record

Written record is the form by which the researcher is going to present her findings. It could be a book, a journal article, a conference paper, a thesis report etc. Written record is of great importance in qualitative research. (Myers, 2009) In our case, written record takes the form of master thesis.

![Written Record: Master Thesis Report](image)

Figure 2.6: Selected Written Record
2.7. Summary of Selected Research Design

The research model we have relied on is the qualitative research model which is designed by Myers (2009) especially for the use in business and management context. Each block of the model has taken its value in compliance with other blocks. The final scheme of our research design is shown in following diagram.

![Figure 2.7: Selected Research Design](image)

Figure 2.7: Selected Research Design
2.8. Research Procedure

Following diagram shows different steps performed to accomplish this research.

Figure 2.8: Research Procedure
3. Background

3.1. Leadership Approaches

Concept of leadership is introduced at 1800s. From that time up to now many attempts have been performed to elucidate this concept. Through these attempts different theories have been developed. (Fairholm & Fairholm, 2009) Leadership theories in last decades are divided into five main approaches, (Parry & Bryman, 2006) each of which has been popular in a specific period of time.

1. The trait approach
2. The behavioral or style approach
3. The contingency approach
4. The new leadership approach
5. The post-charismatic and post-transformational approach

3.1.1. Trait Approach

Trait approach was dominant up to 1940s. This approach lists common adjectives and characteristics of leaders. (Bolden et al., 2003) Scientists in this school believe that leaders have congenital properties which separate them from others. They believe that leader’s required qualities could not be learnt. This approach fails to draw relationship between situational needs and leadership traits since it considers a unique set of characteristics for all leaders. (Zaccaro, 2007)

3.1.2. Behavioral or Style Approach

From 1940s to 1960s style approach stood out among leadership theories. Scientists of this school look for behavioral patterns of leaders, unlike previous school which focuses on inborn adjectives among leaders. They believe that leadership is learnable. Since scientists of this school put different behaviors in a group called style, this approach is known as style approach as well. (Bolden et al., 2003) Since in this approach there is no god-given set of characteristics which distinguishes leaders from non-leaders; everybody is capable of being a leader if he learns to behave in an
appropriate style. (Parry & Bryman, 2006) Through different scientists in this school a group of researchers at Ohio University are prominent. They have identified two key components in leaders’ behaviors: consideration and initiating.

Although efforts of scientists in this school of thought considerably affect existing knowledge about leadership, it failed to present a style which suits various criteria of the organizations to be led. In simple words this approach has not sufficiently considered the situational variances. (Lussier & Achua, 2004)

3.1.3. Contingency Approach

This approach was popular between 1960s and 1980s. Contingency theories define leadership as a situation-dependent concept. Based on this approach there is no global style of effective leadership. (Turner & Müller, 2005) Scientists in this school have tried to find different variables which determine required behaviors of leaders and suggest effective leadership style rooted in these variables. (Bolden et al., 2003)

One of the outstanding results acquired by the proponents of this approach belongs to Fiedler. (1967) He has developed Lease Preferred Coworker (LPC) scale. Based on this scale leaders are in two kinds: relationship-oriented and task-oriented. Fiedler explains that appropriateness of these leadership orientations depends on the situation’s favorableness. Relationship-orientated leaders could bring best results in moderate control situations, when task-oriented leaders are more successful in low or high control situations. (Parry & Bryman, 2006)

This approach has many weaknesses through which the most important one is inconsistent empirical results. (Lussier & Achua, 2004) Also there are other critiques on this approach; like that appropriateness of leadership behavior is not always situation-dependent and that none of contingency models could respond to the question of how leader should behave in conflicting situations. (Parry & Bryman, 2006)
3.1.4. New Leadership Approach

‘New leadership’ approach is used to group different leadership theories which share common themes although they have dissimilarities. (Bryman, 1992) First instances of theories in this approach are found in 1980s. (Parry & Bryman, 2006) These instances include transformational leadership (Bass, 1985), charismatic leadership (House, 1977), visionary leadership (Sashkin, 1988) etc. Many of these notions roots in Burns’ studies. Burns (1978) used terms transactional and transforming leaders for the first time for political leaders. All theories in new leadership approach emphasize on the role of vision and consider the manager as the one whose responsibility is to articulate this vision. In their point of view the leader is the manager of meaning. (Parry & Bryman, 2006)

Although new leadership approach attained great importance it suffers from various deficiencies. First one is that its focus is on top level leaders in spite of the fact that the majority of leaders are at middle levels. Second one is that there is the lack of researches about impacts of new leadership theories based on situations’ specifications. (Parry & Bryman, 2006) Third one is that less attention is paid to behaviors of transformational leaders. Finally the fourth one is the lack of studies about influence processes of transformational leadership. (Yukl, 1999)

This school of leadership was results of studies on successful business managers, those who effectively manage their organization in change challenges. (Turner & Müller, 2005) Weber was the first one who introduced charisma. (DuBrin, 1995) He defines charisma as a quality of person who is distinguished from ordinary people by his personal magnetism and exceptional powers. He considers charisma as an innate property which emerges in crisis.

3.1.5. Post-charismatic and Post-transformational Approach

Considering emergence and employment of new technologies in human’s ordinary life, it is likely to emerge new leadership theories, those which might dispel critiques about new leadership approaches such as charismatic and transformational leadership. One instance of this group of leadership theories is presented by Mumford and Van
Doorn. (2001) This theory which is called pragmatic leadership describe the leader as a person who identify and communicate solution to social problems, pay attention to the needs of followers, persuading them about feasibility of the solution and finally creating structures to support implementation of that solution.
3.2. Transformational Leadership

At 1978 Burns for the first time used the term transformational leadership. He has introduced transformational leader as the one who focuses on values and enhancement of the followers in a way they go beyond self-interests and serve well-being of their society. (Olsen et al., 2006) From that time up to now this concept attained considerable popularity because of the large number of studies decisively support benefits of transformational leadership for organizations, followers and leaders. (Brown & Arendt, 2011) Bernard M. Bass (1985) was the first one to publish a formal theory about transformational leadership. He defines transformational leadership as the act of broadening and elevating the followers’ interests, informing and persuading them about missions and purposes of the group as they can devote their self-interests for the good of the group or the organization. (Bass, 1990) He has developed Multifactor Leadership Questionnaire (MLQ) to measure transactional, transformational and laissez-faire leadership. MLQ should be answered by followers to show which pattern of leadership is more consistent with their leader’s behavioral pattern. Better insight could be attained about transformational leadership with contrasting this pattern to transactional pattern of leadership. Transactional leader concerns about maintaining status quo (McGuire & Kennerly, 2006) but transformational leaders try to move the organization from current situation to desired values and norms. (Bennis & Nanus, 1985) Transactional leaders define specific tasks, deadlines and standards and make sure that followers meet those (McGuire & Kennerly, 2006) but contrarily transformational leaders pose high expectations and change it to followers’ aspiration. They encourage questioning instead of inarguably following routine ways of doing things. (Rafferty & Griffin, 2004)

Model of transformational leadership suggested by Bass (1999) has four dimensions including Idealized influence (or Charisma), inspirational motivation, intellectual stimulation and individual consideration. Idealized influence is used to describe the quality of being respectable for the followers as they find the vision and the mission with leader. (Den Hartog et al., 1997) Inspirational motivation is used to refer to the act of motivating followers through symbolic actions and putting examples as the followers focus their effort to achieve the articulated vision. (Den Hartog et al., 1997) Intellectual
stimulation concerns leader’s action to encourage followers looking at each difficulty as a challenge to confront, each routine solution as a subject of reconsideration. This dimension emphasizes the value of creativity and problem solving. (Yammarino & Bass, 1990) Fourth dimension, individual consideration draws attention to the role of leader as a mentor. Leader should concerns personal needs of their followers otherwise they might waste their potential power in struggling with their personal life’s obstacles. (Bass & Avolio, 1990)

Bennis and Nanus (1985) have developed their theory with an emphasis on organizational processes more than Bass (1985). They differentiate between managers and leaders when they state “leaders are people who do right things but managers are people who do things right”. (Bennis & Nanus, 1985) What they introduced as transformational leadership is similar to Bass’s (1985). They have identified four common characteristics for transformational leaders: identifying and transferring the vision of their organization, transforming organization’s standards and values, displaying a consistent and trustworthy character, concentrating on positive points instead of negative ones. (Bennis & Nanus, 1985)

Although based on Bass (1985) charisma is an essential part of transformational leadership, studies conducted by Bennis and Nanus (1985) showed that transformational leaders are not necessarily charismatic people in their followers’ point of view. (Yukl, 1999)

Tichy and Devanna (1986) based on interviews with successful chief executives (Parry & Bryman, 2006) found seven characteristics commonplace to transformational leaders (Simić, 1998); namely:

1. Being change agent: ability to create flexible, adaptive and creative organization by stimulating change
2. Courage: facing realities, selecting appropriate mind set and taking risks of applying required changes
3. Openness and faith in the followers: empowering followers by putting confidence in them
4. Led by values: devising a set of norms and values, and behaving in accordance with them

5. Life-long learning: ready to learn from any new experience and adapt his attitudes to lessons learnt

6. Ability to face complexity: ready to confront with and struggle in uncertain and difficult situations they face

7. Visionary abilities: ability to imagine desired future, plan the way to achieve it and articulate it to the followers

After aforementioned theories which could be considered as the first theories formally published about transformational leadership, this subject has been under the scientists’ spotlight. A delicate point to be noticed is that at the heart of dominance of charismatic and transformational leadership developed in 1980s and 1990s, is the component of heroism. (Calder, 1977; Meindl et al., 1985; Parry & Bryman, 2006) Accordingly critiques arouse which compare new leadership approach theories with great man theories. (Alimo-Metcalfe & Alban-Metcalfe, 2008) Among them is Peter Gronn. (1995) To respond to such critiques a new tradition of leadership paradigms emerged. They emphasized teams instead of leaders. An example of this tradition is Manz and Sims (1991) and Sims and Lorenzi (1992) who developed SuperLeadership archetype which propagates the idea of self-leadership. (Shah & Ali, 2012) The other criticisms posed against heroic paradigms of transformational leadership are listed below:

**Being exclusive to top level managers**

On one hand we have the fact that there are differences between characteristics of charismatic leaders for distant leadership and that for nearby or close leadership. Distant leaders are those who are in top level situations in the organizations, but close or nearby leaders are immediate leaders or those who are in lower levels. (Shamir, 1995) On the other hand most of the studies led to the theories in new leadership approach tradition are based on studies of distant leaders. (Alimo-Metcalfe & Alban-Metcalfe, 2008) Therefore the transformational or charismatic leadership theories could not be generalized to nearby leaders.
Incomplete set of samples

The majority of studies in leadership research which led to transformational leadership theories are based on studies of men; also there is no clear evidence that they have been developed considering the ethnic specifications and variances. (Alimo-Metcalfe & Alban-Metcalfe, 2008)

Based on leaders’ views

In heroic models the source of information is usually leaders not the staff although it is essential to look at leadership not only from the eyes of leaders but also from the eyes of those who are impacted by the leaders’ behaviors and attitudes. (Alimo-Metcalfe & Alban-Metcalfe, 2008)
3.3. “Nearby Transformational Leadership” or “Engaging Leadership”

Alimo-Metcalfe and Alban-Metcalfe at 2001 presented “Nearby Transformational Leadership” or “Engaging Leadership”. This model surmounts defects of previous transformational leadership models by emphasis on engagement and partnership instead of heroism or being visionary (Alimo-Metcalfe & Alban-Metcalfe, 2008), also by considering diversity of factors affecting leadership concepts. These factors encompass sex, ethnicity, job sectors and governance type. This model is founded on the studies about middle to chief executive levels. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

Source of the data led to the engaging leadership model is not only managers but also their subordinates. (Alimo-Metcalfe & Alban-Metcalfe, 2008) Validity of this model is confirmed in different business contexts including health and social care settings, fire and rescue services and police services. Leadership behaviors which could be result of utilizing this model could noteworthy affect well-being of personnel, job satisfaction, commitment, and decreased work-related stress. (Alimo-Metcalfe & Alban-Metcalfe, 2003)

This model contains persistent themes of team-working and connectedness, encouraging questioning and challenging current status. (Alimo-Metcalfe & Alban-Metcalfe, 2008)

Engaging leadership model for private sector consists of three clusters: leading and developing others, leading and developing the organization and finally personal qualities. (Alban-Metcalfe & Alimo-Metcalfe, 2007) Each cluster comprises different dimensions which are displayed in following figure.
### 3.3.1. Leading and Developing Others

1. Showing genuine concern

A Transformational leader treats humanistically which means that he concerns his followers’ needs and feelings. (Alban-Metcalfe & Alimo-Metcalfe, 2007) He plays the role of a mentor. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

2. Enabling
Enabling shows the level of empowerment team members have. (Alban-Metcalfe & Alimo-Metcalfe, 2007) Trusting staff and delegating responsibilities is what leads to empowerment of employees. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

3. Being Accessible

This dimension has two aspects: first establishing not too formal communication, with emphasis on making face-to-face communication and second being approachable and in-touch with staff (Alban-Metcalfe & Alimo-Metcalfe, 2007)

4. Encouraging Change

This dimension is related to encouraging questioning about current solutions and stimulating staff to find new ways of doing things. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

3.3.2. Personal Qualities

5. Being Honest and Consistent

Honesty and consistency in behavior of leader is an essential characteristic of transformational leader. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

6. Acting with Integrity

This dimension encompasses two aspects. One is about honesty in communications and the other is equitableness. (Alban-Metcalfe & Alimo-Metcalfe, 2007) The leader should be open to criticisms and be ready to consult with others in decision-makings. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

7. Being Decisive

Transformational leader should be ready to take difficult decisions even if they’re risky. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

8. Inspiring Others
One fundamental factor in transformational leadership is to inspire others in a way they like to join the group to achieve its objectives. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

9. Resolving Complex Problems

This scale is related to personal abilities of leader to solve difficult problems. It enables him to handle lack of clarity and confidence his team confronts. (Alban-Metcalfe & Alimo-Metcalfe, 2007)

3.3.3. Leading and Developing the Organization

10. Networking and achieving

One crucial aspect of transformational leadership is what Tichy and Devanna calls vision and articulation and Conger calls communicating the vision. (Bennis & Nanus, 2012) Hence networking which means the capability of leader to create effective communications with both his team and external stakeholders, is another dimension of engaging leadership.

11. Focusing Team Effort

Focusing team effort implies actions performed in order to concentrate effort of team on project goals and priorities. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

12. Building Shared Vision

This dimension emphasizes on one important aspect of “nearby” transformational leadership which is about not only the need to articulate the vision but also the need for engagement of different parties in forming that. These parties include team, client and any other party who plays a role in development of team, project, department etc. (Alban-Metcalfe & Alimo-Metcalfe, 2007)

13. Supporting a Developmental Culture
Supporting a developmental culture means being supportive even in case of mistakes and providing feedback to whom made mistakes. (Alban-Metcalfe & Alimo-Metcalfe, 2005)

14. Facilitating Change Sensitively

This dimension is related to the sensitivity of leader on how changes on the environment impacts team, also his capability to survive with this impact in a way that team smoothly continue the work. (Alban-Metcalfe & Alimo-Metcalfe, 2007)
3.4. Suitability of Nearby Transformational Leadership in relation to Research Field

It’s of great importance to consider that not only being responsive to criticisms on heroic models of transformational leadership is the strength of the Nearby Transformational leadership model but also it has unique characteristics with regard to our research field which is software projects. The positive points this leadership model has with regard to our research context lies behind following characteristics of software projects:

1. High rate of staff turnover

   One of persistent concerns facing IT organizations is staff turnover. (Joseph et al., 2007) In one hand lack of IT professionals makes it easy for this population to find new jobs and in the other hand staff turnover has high cost for IT organizations. In case of IT professional leaves, the organization should struggle with different negative consequences such as work quality reduction, tacit knowledge lost, substitute professional recruitment and training and putting the organizations reputation in risk. (Ghapanchi & Aurum, 2011) Engagement as a fundamental element of engaging leadership helps organizations to lessen the rate of absenteeism and turnover of their staff. (Alimo-Metcalfe & Alban-Metcalfe, 2008)

2. High level of team-work requirement

   High level of team-work is essential to reach to a productive and high quality software project. (Elif et al., 1997) In software projects team is the most influential factor to determine the success of the project. (Hazzan & Dubinsky, 2008) Considering this fact and team-working as a persistent theme of engaging leadership (Alimo-Metcalfe & Alban-Metcalfe, 2008), this model is likely to answer some needs of ideal leadership behaviors required for software projects.
3. High level of personnel burnout

There are different reasons which make IT profession a stressful job, reasons like being time-bound, client-oriented and technology intensive. (Rashidi & Jalbani, 2009) Whatever the reason of such phenomenon is, personnel burnout is known as a common dilemma in software projects today. (Ivancevich et al., 1983; Kaluzniacky, 1998; Rutner et al., 2011) Engaging leadership has shown positive impacts on well-being of personnel under such kind of leadership. This encompasses reduced work-related stress. (Alimo-Metcalfe & Alban-Metcalfe, 2003)

4. Being innovative job

Any software problem might have different solutions. Software job could not be routinised. (Ilavarasan & Sharma, 2003) This job is placed among creative professions. Encouraging questioning as another persistent theme of engaging leadership (Alimo-Metcalfe & Alban-Metcalfe, 2008) is in line with this requirement of software job.
3.5. Software Projects and Management

There is a prevalent pessimism about the destiny of software projects. (Cule et al., 2000; Johnson, 2006) This pessimism roots in different reasons. Through them the most prominent one is the lack of effective leadership. (Pulk, 1990) Software projects are placed among technical projects and technical/scientific employees usually suffer inadequate managerial qualities. (Thite, 2000) Unfortunately software project management is mostly looked as management not leadership. (Anon., 1999) Most of the books in this field limited to technical aspects of software projects such as planning, project lifecycle, project monitoring, risk management etc.

Scientists have tried to find an ideal model of leadership which suits software projects. Pulk (1990) with stressing the role of leadership on success of any software development effort, have mentioned project management skills, software development skills and the knowledge of the product to be developed, as the requirements of a software project manager. Also he has highlighted the fact that the prevalent weakness of managers in this field is related to their project management skills. This fact attains greater importance considering the Kerzner’s (1989) statement about the project failures’ common reasons; which is related to human and social aspects of the projects. Pulk (1990) also emphasized on the knowledge of product as a more required quality in software field comparing to other technical fields. Pulk’s (1990) studies are mainly related to software development projects although there is another type of software project which ends to client organization’s acquirement of a new software product; this is package implementation. (Cadle & Yeates, 2008)

Jeff Gooch (1997) mentioned challenges which are particular to IT projects. Those include being less tangible and less familiar comparing to other types of projects, diversity in team members’ skills, experiences and attitudes and being difficult to concretely define scope and requirements. He has mentioned about the importance of goal presentation and clarification for team members in forming their endeavor to arrive at project objectives. He has analogizes the project manager to an orchestra conductor and says that just like an orchestra conductor who should not play a string himself, the
project manager should not act as a chief analyst, coder, designer etc but he should just manage.

Royce (2005) suggests a steering leadership style which is described as active management involvement and frequent course correction. He explains about the uncertainties within software management domain. One required quality of software project managers based on his point of view is mentoring for team members with a lower level of experience.

Rose et al (2007) in contrary to traditional software project management approach which considers the right use of techniques and tools the key to success of software projects, tried to develop a competency model for software project leaders. They stated that software projects differences ask for different set of management competencies. Based on Rose et al (2007), suggested model of management competencies should be considered only as a starting point. Competence pyramid consists of seven competencies:

![Software Project Management Competences Pyramid](image)

*Figure 3.2: Software Project Management Competences Pyramid (Rose et al., 2007, p.616)*
1. Technical management: the ability to manage the software environment
2. Process management: the ability to accomplish traditional project management tasks such as planning and resource management
3. Team management: the ability to form an effective team and manage relationship with its members
4. Customer management: the ability to create a appropriate relationship with customer
5. Business management: the ability to achieve an acceptable financial result
6. Personal management: the ability to use personal skills for the good of the project
7. Uncertainty management: the ability to manage complex problems arouse as a result of software project’s complexities

Among responsibilities of project manager is to create a good project atmosphere, also to protect team members in different situations such as customer or managers’ interference, team members having personal problems and team members being under work pressure. Project manager must be the main motivator and commitment developer and the one who identifies origins of motivation in team members. Another important quality of project manager could be described by sparring partner. It means that he should play the role of a consultant for team members regarding their technical problems, work processes, social interactions and even a personal friend. Being personal friend implies engagement and engagement causes respect and respect is the most effective source of problem solving in any project. Project managers are decision makers who have the ability to taking risks in case of uncertainties. Furthermore they need good communication skills to be able to make an effective relationship among different parties in a software project, i.e. customer managers, internal managers, users and the team (Rose et al., 2007)

Mohan Thite (2000) tried to find the best leadership style for IT projects. His studies have shown that a mixture of transformational leadership with technical leadership and one dimension of transactional leadership suit this kind of projects. The basis of transformational leadership for his studies is the model suggested by Bass and Avolio. (1990) Although the model was validated for IT projects, it could be enhanced in
relation to technical projects’ specifications. Based on this study required characteristics of IT project managers are:

1. Organizational Catalyst: being able to create an atmosphere in which team members can freely explore their creativity.
2. Intellectual Stimulation: encouraging unconventional thinking to find alternative ways of doing things
3. Charisma: being able to create team’s commitment to common goals and values
4. Contingent reward: ensuring that subordinates receive appropriate reward. This dimension is a transactional behavior but necessary for an IT project manager.
5. Active Monitoring of Exceptions: continuously and closely monitor the performance of team members in relation to project goals and standards. This dimension is also a transactional behavior but the point is that if it’s performed within a transformational style it would benefits the project.

Thite and Simmons (1997) has stressed on technical leadership as a very important behavior which should be practiced together with transformational leadership.
3.6. Case Selection Criteria

Pawar and Eastman (1997) in their effort to develop a model of contextual factors affecting transformational leadership receptivity have reviewed organizational factors influencing transformational leadership requirement and emergence. These factors which are listed below would be considered as a basis to select our interviewee software teams.

A. Organizational climate

Bass (1985) has proposed the need for further researches on impacts of organizational climate on transformational leadership emergence. Lale Gumusluoglu and Arzu Ilsev (2006) have mentioned that in a climate which supports innovation, followers are more likely to accept transformational leadership.

Taking into account the focus of our research which is on software teams, organizational climate could be interpreted to team climate. Based on West and Anderson (1996) four factors predicts efficient team functioning. They are:

1. Participative safety: The level of trust felt by team members when they state their opinions
2. Support for innovation: The level of support for original and innovative ideas
3. Team vision: The level of clarity at which team has defined its goals
4. Task orientation: The level of team effort made to reach high quality results (Acuña et al., 2008)

B. Structure

According to Shamir and Howell (1999), organic structures are more suitable hosts than mechanistic structures for emergence and functioning of charismatic leadership.

Burns and Stalker (1961) defined organic structures as structures with more decentralized decision-making processes, less formalization and standardization, more lateral communications rather than vertical communications and fewer hierarchical
distinctions. In comparison to organic structures, mechanistic structures don’t permit their members to act based on their differences. Moreover Garg & Krishnan (2003) found that transformational leadership is positively related to the level of decentralization in host organization.

Decentralized organizations are those in which decisions are made not only at the top level management but also in all levels of management and organization. Hence the level of decentralization is defined as the level of authority distributed in all levels of organization for making decisions. (Khan, 2010)

C. Environmental complexity and scarcity

Bass (1990) said that when organization is under stress or in need of long range planning and anticipation of potential crisis, transformational leadership could be helpful. (Bass, 1990)

Gibbon (1992) has identified environmental complexity and scarcity as external factors to determine required form of leadership. Information complexity is same as the level of uncertainty and organizations can overcome uncertainty by variety of boundary spanners. Resource scarcity necessitates organizations for more integration and efficiency. (Gibbons, 1992)

D. Task scope and subordinate growth need

Resource position impacts nature of leadership challenge and resource condition impact how organizations could implement change. (Gibbons, 1992) Also Griffin(1979) said that followers with high growth need are eager to undertake challenging high scope tasks while followers with low growth need, require their tasks to be well-structured if they are not routine. (Griffin, 1979) Hence task scope and subordinate growth need together affects required leadership activities.

Task scope may range from routine and repetitive jobs to very complex and challenging jobs.
GNS (Growth Need Strength) is defined by Hackman and Oldham (1980) to measure the level of growth each individual likes to achieve in his career. This includes personal needs to development, learning and success in the job. High-GNS individuals showed high interests in undertaking complex, enriched jobs but low-GNS individuals don’t have any interest in performing challenging jobs. (Lawrence, 2001)

E. Organizational size

Rouche (1989) in his research of transformational leadership in American college community found relationship between transformational leader behaviors and the size of organizations. He found that leaders of medium-size organizations got better scores for MCLQ (Multifactor College Leadership Questionnaire)

F. Organizational life cycle

Baliga and Hunt (1988) discussed how organizational life cycle is in relation with the form of required leadership. They divided organizational life cycle into 4 main stages and discussed form of required leadership in each of them.

1. birth stage: external transformational, representational
2. growth stage: external transformational, representational, transactional
3. maturity stage: Internal transformational, representational
4. revitalization/death
5. revitalization via crisis path: representational, external and internal transformational and transactional
6. revitalization via proactive path: representational, and internal transformational
7. proactive death: representational and transactional

Transformational and transactional behaviors refer to the definitions presented by different authors such as Burns and Bass. Representational behaviors are those behaviors which are represents some aspects of organization not to subordinates but to those who are at the same (or close to) organizational level as the leader is. (Hunt et al., 1988)
G. Organizational strategy

Bass (1985) suggests organizational objectives as a contextual factor which affects influences of the transformational leadership. Organizational objectives could result in the need for more transformational leadership. (Pawar & Eastman, 1997) This kind of organizational objectives are strategic ideals which are found challenging by team members. (Conger & Kanungo, 1998) Strategies of organizations could be classified into four different types: defender, prospector, analyzer and reactor. (Berson & Avolio, 2004) Organizations with defender strategy try to produce high quality product which stabilizes its position in the market. Organizations with prospector strategy are those which should always be ready for quick responses to market changes and periodic redefinitions. Analyzer organizations simultaneously try to keep their stable position and seek for new opportunities. Reactor organizations just react to what happens without any specific predetermined direction. Manager’s perception of organizational strategy is determinant to emergence of transformational leadership behaviors.
3.7. Selection Method

Following table is created based on mentioned explanations about each criterion. Available teams have been evaluated by attempting to assign a grade/value to each criterion. Teams with the highest diversity have been selected for interview. The grade/value of each criterion for different teams is assigned based on the viewpoint of company’s HR (Human Resource) manager who is the most informed person about such criteria for all teams. Following table contains grades/values assigned to two selected teams as well.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Team</th>
<th>Scale</th>
<th>Grade/Value</th>
<th>Team Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Team1</td>
</tr>
<tr>
<td>Organizational climate</td>
<td>Participative safety</td>
<td>1-5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Team vision</td>
<td>1-5</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Task orientation</td>
<td>1-5</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Support for innovation</td>
<td>1-5</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Structure</td>
<td>Decentralization</td>
<td>1-5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Environmental complexity</td>
<td>Uncertainty</td>
<td>1-5</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>Resource scarcity</td>
<td>Resource scarcity</td>
<td>1-5</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>Task scope</td>
<td>Originality, Complexity,</td>
<td>1-5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Difficulty</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Criteria</td>
<td>Team</td>
<td>Scale</td>
<td>Grade/Value</td>
<td>Team Status</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Subordinate growth need</td>
<td>Interest to perform challenging, complex and enriched jobs</td>
<td>1-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Organizational size</td>
<td>Number of team members</td>
<td></td>
<td>14</td>
<td>83</td>
</tr>
<tr>
<td>Organizational life cycle</td>
<td>birth, growth, maturity,</td>
<td>birth, growth, maturity, revitalization via crisis path,</td>
<td>Growth</td>
<td>Revitalization via crisis path</td>
</tr>
<tr>
<td></td>
<td>revitalization via proactive</td>
<td>revitalization via proactive</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>path, proactive death</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Organizational Strategy</td>
<td>prospector, analyzer,</td>
<td></td>
<td>prospector</td>
<td>analyzer</td>
</tr>
<tr>
<td></td>
<td>defender, reactor</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Criteria of Selected Cases
4. Results and Analysis

4.1. Required Characteristics and Behaviors of Software Project Managers

Good technical information is a highly required characteristic of software project managers. Having good technical information enables managers to recognize final destination of project and the way to achieve it. Good technical information is the most fundamental management essentials mentioned by our respondents.

He is very competent person... since he closely works with projects, I know other competent managers as well but regarding this work he acts even better than our CEO (Chief Executive Officer) however our CEO might have even more work experience or management experience but he can manage better. The reason is that he is very much involved with technical job, he can observe technical problems, and in all aspects he makes use of it. If you ask him about feasibility study he really sees technical things and conducts study based on that, even if you ask him about financial aspect he looks at technical part of the work and says because of these technical facts the financial aspect is this... I tell him something and he sees everything around it, he has very good technical knowledge.

In the ever-changing customer environment defining user requirements is one of the challenges software projects usually struggle with. This is what usually called as requirement uncertainty and consists of requirement instability and requirement diversity. (Liu et al., 2011) Furthermore each software project has different stakeholders from two sides. One is development team and second is different departments in client organization. Usually there are conflicts about defining project requirements even in client organization. (Liu et al., 2011) These all call for a manager who understands language of both developers and client parties, one who has good technical information. Such a manager is able to distinguish the final destination of project through all these uncertainties and diversities. Having good technical information is required but not enough because the manager should make use of it to be a purposeful manager. It means that manager should define borders of the project; however it’s not
formally mentioned at contract making stage or even initial stages of project. He should be able to identify how the final product of project has to look like and articulate it to developers and client. Otherwise the amount of scrap and rework (Royce, 2005) which is a natural part of software projects would intolerably increase.

At least he knows what he wants to do and this is very influential in IT work... to have a purpose, know what we want to do... this seems very obvious but... if we know what we want to do sub-goals resulted from that would be accurate

Another respondent says:

He knows what he wants. He can very well break the work and then collect it again

Different team members might have different imagination about the final product of the project and consequently different ideas about how to achieve it. When manager has identified final product of the project he needs good communication skills in order to articulate it to whole team members. He should be able to persuade team members about the way to achieve it. Otherwise during project work contradictions raise and since results of everybody’s work is dependent on the others’, people’s attempt is wasted and there will be no final product or a flawed one. In addition software project manager needs good communication skills to create good relationship with client. Effective communication is recognized as a predictor of software project success. (Edwards et al., 2006)

He can very well impress his audience about anything, for example he gives a solution just now, persuade all that this solution is right... after five minutes one point comes into his mind and he can persuade all again...that solution has these problems and all should accept this one. Audience’s acceptance is very high in front of him.

Software manager should have high communication ability because he should present work to client and protect it... he is the point of contact with client, also he should be able to create a bi-directional respect with client organization.
Indirect supervision or what one of the interviewees called internal control is much more effective than direct supervision or external control. It causes team members to supervise their work before being checked by the manager. Considering the intangible output of software projects (Cadle & Yeates, 2008; Kittlaus & Clough, 2009) which makes supervision of these kinds of projects complicated, such a way of supervision is very useful.

Internal control means that you bring yourself in somebody’s world that he accepts you, as a reliable person... he accepts his talks and he has trust in him... external control means that he threatens, puts pressure, shouts... why it’s not done... why is it so... correct it in 2 hours...He should dictates but how he does it is important... when I’m in somebody’s world he accepts me and performs it... because I’m in his ideal world as a manager and he does it to satisfy me because I’m trustworthy for him...External control might have more effects for a construction job rather than a software job.

Another respondent says:

He is both close to work and far from that... it means he acts in such a way that you don’t feel he is interfering in your work but he is such close that he can analyze your work technically

Intelligence is another required characteristic of software project managers. Respondents believe that management of software projects needs such smartness. One of the respondents says:

Manager should be clever since project doesn’t have any sensible output. If he is not clever he cannot measure real progress of the project.

Team members might hide intangible facts from the manager till last moments. Also they might do a low-quality job and if manager does not discern it, defects might disclose even long time after delivery of the software product to client. Based on studies the time between delivery of software product and depicting quality management results is quite long. Also quality management in software projects is result of individual judgment. (Phan, 1998) Taking into account quality management, a human-dependent
job, personal abilities of those who are responsible for quality of software, gets more importance. Considering software project manager as the final responsible person for the quality of software calls for unique properties like meticulousness and intelligence.

*Team under management of meticulous and intelligent person is forced to work with higher accuracy... but the team of the one who’s not meticulous says himself if I don’t do this he doesn’t understand... for example one of the team members is writing a report for a project. If he knows the manager is intelligent he is forced to test cases which might not occur even in close future*

Being **good-tempered** means showing openness and kindness to the others. It is critical for software project managers because software implementation is a very human-dependent job. People are the most important part of software projects. (Bas, 2006) They cannot do qualified jobs if they are irritated by the work environment. Morality of manager as a factor to affect work environment plays a substantial role in software projects. Software project manager is sometimes described as a negotiator because he is the central communicator between users, management and development team. (Bas, 2006)

*Between managers and their subordinates always conflicts occur... there are managers who instead of resolving this problem and return the team to initial state make the relationship worse. It means if today one time put pressure on team member tomorrow make it two times... but our manager hasn’t such personality.*

*He says I’ll be thankful if you can do it and we will do it for him... because of the good relationship exists between us, we give the priority to him... even if it’s holiday.*

**Patience** is also highly required for managing software projects. Stress is very commonplace in IT projects. (Glass, 1997) Also these projects need high level of communication. (Rutner et al., 2011) Hence if manager is not patient when communicating with team members stress of work environment would not be tolerable. To this should be added the human resource turnover which is very usual in software projects (Ghapanchi & Aurum, 2011) which in turn leads to a lot of rework. All these necessitate manager to be patient.
Manager of software project should be able to make friendly relationship with team members. It has different reasons. Having a sense of friendship decreases stress and pressure of work which is an impartible part of software projects. (Kaluzniacky, 1998) Also it’s good to consider one of respondents’ statements who say:

“If manager is friendly it helps team members grow up and it leads team member to be honest so he tries to compensate his mistakes if any, even if he has to work overtime... it makes environment pleasant, also gives self-confidence to team members... sometimes we are tired he comes and we say some jokes... we get free from work for some moments... this positively affect my performance as well... it gives me confidence when I feel a higher level manager has this kind of relationship with me

Being friendly is critical because output of software projects is intangible (Kittlaus & Clough, 2009) and progress of work could not be easily measured so if a team member hide his mistakes or neglects, it’s not quickly and easily revealed. Consequently honesty between leader and team members as a result of friendship could have positive impacts on software project.

in a construction project if team member use stucco instead of cement manager could easily recognize it but in a software project this is not easy to find mistakes and neglects, some of problems shown their negative effects after a long time, for example when a query in not optimum, after having lots of data its impact shows, maybe after a year.

The other reason why the manager of software project has to be friendly with team members is that if having such a relationship team members can easily talk to him and ask about their questions and doubts. It might improve quality of the work and shorten the time required to accomplish it, especially considering the very ongoing changes in software subjects (Kaluzniacky, 1998) which leads to more questions and doubts for those who are working in IT projects.
It is scarce that somebody is at the highest level of organization but you can easily talk to him... there are many times that we have talked to him and arrived at no specific conclusion but that we can easily talk to him makes us very confident... there are other managers that we know if we go to them the problem gets resolved but because we don’t feel openness we preferred to forget about it and tell ourselves let this problem doesn’t get resolved, I don’t want my problem to be resolved at all... I don’t go to him

Sympathizing with team members when they have problems in their personal life, feel sick and similar situations creates a sense of humanism which provides the mind relief which is very helpful to accomplish a software job which is all a brain work. (Glass et al., 1992)

He is a very good tempered person... for example at the work peak if I face a problem or sickness and don’t come to office he calls personally and if a sickness happened, he shows affability and say take rest till you get better then only come... this is very good point, gives the guys energy... other managers might say take your laptop home and do it or I send your laptop home you do this

One of the ways in which the software project manager can cause team members make use of their creativity is to visualize team members’ personal goals. One of interviewees mentioned that

At the start of my job, being a project manager was my desire and my project manager who knew this, used to show relations of tasks assigned to me with being a project manager. It really motivated me

Another way to raise team members’ creativity is to posing questions based on their individual interests. Visualizing team members’ goals and posing questions based on individual interests cause team members to be creative.

Motivation is a fundamental driver for creativity. (Zhang & Bartol, 2010) When a software project manager spends time about technical problems with his team members they gain an insight which helps them to be more creative. Especially if he answers questions of team members by showing them how to find answer of their questions
instead of directly answering them. We call it **indirect answering**. Here it’s important to consider again that software project manager should be eligible in technical aspects of the work as well.

*I think he can answer questions of team members by not simply giving them the answer but showing them the ways to find the answer*

Software project team members attain motivation if their manager set an example for them. If the manager has such a social and technical personality which team members can consider him as an example, it motivates them for more attempts. In such an example person they can see their future. It makes them put more effort to arrive at such a personality and position.

**Indirect stimulation** has a considerable positive impact on encouraging team members. Manager of one of the teams under study, used to introduce team members’ achievements in meetings with client when team member himself is not present in the meeting or with high level managers of company like CEO or HR manager. When aforementioned team member hear about such praise he feels appreciated and attain motivation. This is what we call indirect stimulation.

*He knew what to tell me, what to instill into me... to make motivations in me... he has another positive characteristic as well. I have been done something that only he knew about it. No one else in the company knew I have done it... but I am looking for praise... he in a meeting that I was not present mentioned that, but he didn’t tell me that he has mentioned it... I heard it from someone else... it stirred my motivation*

Manager of software projects should be a **calm person** and a person who can manage stress; because IS projects are stressful in their nature on one hand and software job needs high level of concentration on the other hand. Stress enormously affects concentration. (Glass, 1997) Hence the manager should be able to decrease controllable roots of stress. **He should have high level of team management abilities** which is the ability to keep all members working cooperatively. Rutner (2011) says IT professionals have to be good team players. They are very frequently in contact with their team members and client whereas their relationship with client is usually controversial.
Hence the manager should be able to manage conflicts between team members or team members and client. He should be able to easily manage technical or non-technical argues between different parties; this is an unavoidable part of any team-work.

This is very probable that conflicts occur between team member...he creates an agreement among them... he is a person who wants to solve the problem instead of looking for the guilty

Manager of software project should be flexible and liberal, the one who always embrace new suggestions and solutions. A certain software problem might have different solutions. If the manager persists on the primary solution he might lose the best choice. If team members know that he has a dogmatic personality they will never present him their new suggestions.

Flexibility means that when he says something he doesn’t insist that it should be done this way...if I have found another solution he doesn’t say why you didn’t do what I suggested however there are some other managers who say that because I’m manager and I have responsibility you must do this...if it was so we would lose our creativity... we just waited for what the manager said and acted based on it... we could do creative jobs this way

Honesty and trustworthiness are two other critical characteristics for a software project manager. These properties are important because software job is a brain work (Glass et al., 1992) which needs lots of concentration. In a mistrusted environment it is so difficult to concentrate on the work. Team members’ mind is always busy with irrelevant things. They don’t comfortably and frankly talk to their manager so they hide their questions and mistakes. These mistakes and doubts would not easily be revealed. They usually stay hidden till very final steps of the work.

I can say he is somebody you can count on him. If he says something you can count on it, this is not like that he tomorrow says no, I didn’t say this... maybe after sometimes you go to him and he says yes, I said this, but I made a mistake or it didn’t happen... but it’s not like that he says no, I didn’t say this
Accompany with team members is another behavior which enables manager to smoothly control work progress. Spending time with team members help them to effectively confront complexities when manager helps them in problem solving. The importance of the amount of the time manager spends with his team members is highlighted taking into account the need of integrating the results of work done by different team members. It needs high level of coordination between team members. When manager spends time with his team members he is able to prevent plausible mistakes and inconsistencies. Considering the time shortage most of the software projects should struggle with (Genuchten, 1991), prevention of mistakes and reworks is essential.

Whenever one of team members is unable to do something instead of putting pressure on him he tries to come beside him and accompany with him

Sometimes project fall into complicated situations such as when team members don’t come to an agreement about different solutions. They might spend a lot of time on argumentation about dissimilarities of these solutions. Sometimes the best solution might be sacrificed because of what happens during interactions among team members. In these kinds of situations manager should decidedly make a decision and get the project out of the situation. Hence manager of software projects should be decisive when required.

Project manager can help team members in doing a quality job if he clarifies project goals and limitations. This way the complexity of work both in technical and social aspects is lowered.

You as a manager can collect all team members and explain the project’s importance for the company... how much important it is, how much strict the client is, which problems might arise...what things are important and what siding problems may occur in the project

One of behaviors our interviewees mentioned as what makes them to put extra effort; is that their leader protects them against client or higher level managers of the organization; when they make a mistake. It’s the extreme level of support. In such
situations team members perceive they indebted to the leader and try to deliver a quality job. Regarding some properties of software products like that some errors appear in long term (Phan, 1998) also the complexity of the work (Ilavarasan & Sharma, 2003) which causes software errors to be inevitable this behavior is highly requested by staff.

*He in many cases sacrifice himself for his subordinates whereas he faces himself with his upper hands*
4.2. Required Characteristics and Behaviors in Relation to Leadership Theories

Here under each dimension first we describe what that dimension means, and then we explain to which of required characteristics and behaviors of software project managers it is related to.

4.2.1. Leading and Developing Others

1. Showing genuine concern

A Transformational leader treats humanistically which means that he concerns his followers’ needs and feelings. (Alban-Metcalfe & Alimo-Metcalfe, 2007) He plays the role of a mentor. (Alban-Metcalfe & Alimo-Metcalfe, 2005) Showing concern about somebody needs to make an effective communication with her. Being good-tempered and friendly are prerequisites of an effective communication. It helps team members feel they are looked as humans not as instruments to get work done. Sympathizing is another sign of showing genuine concern because it’s not possible to sympathize with others unless you have been thought about them and their problems. Although other characteristics such as calmness and patience might be in some kind of relation to showing genuine concern but we don’t consider them under this dimension since it’s possible that leader shows genuine concern even if he is not patient and calm. Indirect stimulation of team members shows manager’s consideration about individual’s growth need.

2. Enabling

Enabling shows the level of empowerment team members have. (Alban-Metcalfe & Alimo-Metcalfe, 2007) Trusting staff and delegating responsibilities to them is what leads to empowerment. (Alban-Metcalfe & Alimo-Metcalfe, 2005) Indirect supervision leads to empowerment in two ways. First it causes employees feel their leader trusts in them and second it gives employees the freedom they need to blossom their creativity. By indirect supervision leader delegates the responsibility to staff and by some means apply his supervision. Indirect answering of team members’ questions
also means that the leader believes that team members can find answers by themselves hence increase their self-confidence which itself leads to a higher level of empowerment.

3. Being Accessible

This scale has two aspects: first establishing not too formal communication, with emphasis on making face-to-face communication and second being in-touch with staff (Alban-Metcalfe & Alimo-Metcalfe, 2007) We can place being good-tempered, friendly, spending time about technical problems and accompany with team members under this scale since being good-tempered and friendly implies first aspect of this scale and spending time about technical problems and accompany with team members implies second aspect of this scale.

4. Encouraging Change

Encouraging questioning about current solutions and stimulating staff to find new ways of doing things is an important scale of transformational leadership. This is what we found as posing questions based on staff’s individual interests and indirect answering. Posing questions cause staff start thinking about status quo. Also indirect answering stimulates them to think more, during which they have to find and criticize existing solutions.

4.2.2. Personal Qualities

5. Being Honest and Consistent

Honesty and consistency is about being honest for the good of the organization. We have found same characteristic needed for the leader of software project manager under this dimension: honesty and trustworthiness of leader.

6. Acting with Integrity

This dimension encompasses two aspects. One is about honesty in communications and the other is equitableness. (Alban-Metcalfe & Alimo-Metcalfe, 2007) We also found
honesty and trustworthiness as an important characteristic from team members’ point of view. Another behavior which is noteworthy for software teams is looking all team members the same. This is what Alban-Metcalfe & Alimo-Metcalfe (2007) considered as equitableness.

7. Being Decisive

Transformational leader should be ready to take difficult decisions even if they’re risky. (Alban-Metcalfe & Alimo-Metcalfe, 2005) Under this dimension we found two required characteristics for software project manager. First one is being purposeful. When software project manager is purposeful it means that he has identified the way project must begin and go on and have articulated it to all parties such as client and team members. Second characteristic is what software teams call exactly the same as what our selected model suggests: being decisive when required.

8. Inspiring Others

One fundamental factor in transformational leadership is to inspire others in a way they like to join the group to achieve its objectives. (Alban-Metcalfe & Alimo-Metcalfe, 2005) To inspire others, leader needs to have good communication skills. Also he can visualize team members’ personal goals when he strengthens on those parts of their aspiration which are congruent with team’s objectives.

9. Resolving Complex Problems

This scale is related to personal abilities of leader to solve difficult problems. It enables him to handle lack of clarity and confidence his team confronts. (Alban-Metcalfe & Alimo-Metcalfe, 2007) Intelligence, a required characteristic of software project manager, is essential to resolve complexity exists in software problems. (Ilavarasan & Sharma, 2003) Also flexibility and liberality of software project manager helps him to find the best solution for the problems.

4.2.3. Leading and Developing the Organization

10. Networking and Achieving
One crucial aspect of transformational leadership is what Tichy and Devanna calls vision and articulation and Conger calls communicating the vision. (Bennis & Nanus, 2012) Hence networking i.e. the capability of leader to create effective communications with both his team and external stakeholders is another dimension of the model we have selected. Networking with external stakeholders might be achieved if manager has **good communications skills** and in addition to this if he is **good-tempered** he would be able to create effective communications with team members.

11. Focusing Team Effort

Focusing team effort implies actions performed in order to concentrate effort of team on project goals and priorities. (Alban-Metcalfe & Alimo-Metcalfe, 2005) **Clarifying project goals and limitations** is a prerequisite of focusing team members on project goals. After the goals are clarified and team members start working based on them, **indirect supervision and accompany with team members** in different steps assures that the working process is appropriately following the way to meet such goals. If any contradiction occurs between what team members are carrying out and the appropriate working process, manager needs to interfere. Based on our findings it constitutes one characteristic of software project manager, being **decisive when required**. Focusing team effort requires that manager be very well aware of project borders and goals and the way to achieve those. We have found it as being **purposeful**.

12. Building Shared Vision

This factor emphasizes on one important aspect of “nearby” transformational leadership which is about not only the need to articulate the vision but also the need for engagement of different parties in forming the vision. These parties include team, client and any other party who plays a role in development of team, project, department etc. (Alban-Metcalfe & Alimo-Metcalfe, 2007) Building a shared vision should be an incremental process through which the goal and anticipations of project is defined with engagement of all stakeholders. In order to handle engagement of different parties, manager should have **good communication skills**. Also he should **clarify project goals**
and limitations for team members in order to make sure that there is an agreement about project goals and anticipations.

13. Supporting a Developmental Culture

Supporting a developmental culture means being supportive even in case of mistakes and providing feedback to whom made mistakes. (Alban-Metcalfe & Alimo-Metcalfe, 2005) Using indirect supervision leader can get informed about mistakes team members might have made and pilot them to the correct way. Also one characteristic which is highly required to be supportive even in case of mistakes made by team members, is patience. If the leader shows patience also the flexibility and liberality, team members would easily talk about their questions and mistakes and ask help from him. In addition if team members perceive that their manager protects them against client or higher level managers they feel such a support which should exist in developmental climate.

14. Facilitating Change Sensitivey

This dimension is related to the sensitivity of the leader on how changes on the environment impacts team, also his capability to survive with this impact in a way that team smoothly continue the work. (Alban-Metcalfe & Alimo-Metcalfe, 2007) Under this dimension we didn’t find any specific behavior or characteristic for software project manager. The reason behind such a result might be twofold. First one is that scope of the leadership on which we have focused is a software implementation project. Definition of the project usually means defining different aspects of project such as time, budget, partners etc. Hence leading a project is not like leading an organization which has undefined circumstances which might frequently change. The second reason is that, if any of project aspects changes it would be considered as a major change because of which negative impacts on project and team is inevitable and project leader cannot prevent them. However we should consider that we have looked on transformational leadership from team members’ point of view not from software organization managers’ (managers of project managers’) point of view. It means that if we have been asked software organizations’ managers about required characteristics
and behaviors of their subordinate managers, i.e. software project managers we might observe that they expect specific behaviors by project managers even in case of major changes in project.

<table>
<thead>
<tr>
<th>Required Attributes of Software</th>
<th>Project Managers</th>
<th>Leading and Developing Others</th>
<th>Personal Qualities</th>
<th>Leading the Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Showing genuine concern</td>
<td>Being honest</td>
<td>Resolving complex problem</td>
</tr>
<tr>
<td>Transformational</td>
<td></td>
<td>Enabling</td>
<td>Encouraging change</td>
<td>Networking and achieving</td>
</tr>
<tr>
<td>Leadership Dimensions</td>
<td></td>
<td>Being accessible</td>
<td>Acting with integrity</td>
<td>Focusing team effort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Inspiring others</td>
<td>Being decisive</td>
<td>Building shared vision</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Being patient</td>
<td>Resolving complex problem</td>
<td>supporting a developmental culture</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>good technical information</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>purposeful</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>good communication skills</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>indirect supervision</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>intelligence</td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>good-tempered</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>patient</td>
<td></td>
<td></td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>friendly</td>
<td></td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>posing questions based on individual interests</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>visualize team members’ personal goals</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>spend time about technical problems</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

64
<table>
<thead>
<tr>
<th>Required Attributes of Software</th>
<th>Transformational Project Managers</th>
<th>Leading and Developing Others</th>
<th>Personal Qualities</th>
<th>Leading the Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Leadership Dimensions</td>
<td>Showing genuine concern</td>
<td>Encouraging change</td>
<td>Acting with integrity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Enabling</td>
<td>Being accessible</td>
<td>Being honest and consistent</td>
</tr>
<tr>
<td>indirect answering</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>set an example</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>indirect stimulation</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>calm person</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>high level of team management</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>abilities</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>flexible and liberal</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>honesty and trustworthiness</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>accompany with team members</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>decisive when required</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sympathizing</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>looking all team members</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>the same</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>clarify project goals and</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>limitations</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>protects team members</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>against client or higher</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Facilitating change sensitively</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Required Attributes of Software Project Managers</td>
<td>Transformational Leadership Dimensions</td>
<td>Leading and Developing Others</td>
<td>Personal Qualities</td>
<td>Leading the Organization</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>--------------------------------------</td>
<td>------------------------------</td>
<td>-------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td></td>
<td>Showing genuine concern</td>
<td>Enabling</td>
<td>Being accessible</td>
<td>Being honest and consistent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Encouraging change</td>
<td>Acting with integrity</td>
<td>being decisive</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Inspiring others</td>
<td>Resolving complex problem</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Networking and achieving</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Focusing team effort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Building shared vision</td>
</tr>
<tr>
<td>Table 4.1: Required Attributes of Software Project Managers in relation to Nearby Transformational Leadership</td>
<td></td>
<td></td>
<td></td>
<td>Supporting a developmental culture</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Facilitating change sensitively</td>
</tr>
<tr>
<td>level managers</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.3. Conclusion

We have found related characteristics and behaviors under each dimension of transformational leadership model proposed by Alban-Metcalfe and Alimo-Metcalfe. (2001)

Most characteristics and behaviors required for software project manager are in line with transformational leadership dimensions. Five characteristics or behaviors required for software project manager do not fall into any of the dimensions; but it is noteworthy to consider that four of these five ones, could be considered as transformational leadership properties based on other models, such as those which have looked at transformational leadership style as a “heroic leadership” style.

Good technical information is not a part of engaging leadership model but still highly required for software managers. This result conforms to the results of studies by Pulk (1990) which mentioned knowledge of product as one of the three necessities of software project managers. One point to be considered is that, studies by Pulk (1990) is limited to software development projects but our study broadens the scope to package implementation projects. In simple word our research has shown that some parts of Pulk’s finding could be generalized to other types of IT projects. Thite and Simmons (1997) are the other scientists who have emphasized on the need of technical information for IT project managers when they have included technical leadership to be practiced in parallel with transformational leadership behaviors. Rose et al (2007) have found technical management as the lowest layer of their pyramidal model of software project management competencies.

Some of IT project challenges mentioned by Jeff Gooch (1997) are strongly confirmed by our study. Those include intangibility of software product and difficulty in definition of scope and requirements. We have identified the lines of relationship between these challenges and required characteristics of software project leaders. Jeff Gooch (1997) recommends that software project manager should be a single-tasked person whose only task is the management. Based on this recommendation he should not interfere in any other task such as analysis, design, coding etc. Although we have
not directly tested this assumption, our findings by stressing on active involvement of the leader, implies to probable need of his engagement even in tasks which are not formally assigned to him. Many of the characteristics and behaviors we have found, ask for such an involvement. They include indirect supervision, spend time about technical problems, accompany with team members and posing questions based on individual interests.

Steering leadership style which is proposed by Royce (2005) as the appropriate style for the software manager is very close to our findings. He has described steering leadership as active management involvement; also the leadership model we found valid for software projects, is engaging leadership. Mentoring of team members, another essential quality of software manager based on Royce (2005) is demonstrated in the qualities we found under the developing others cluster.

Three layers of the software project management competencies pyramid (Rose et al., 2007) are reflected in those dimensions of engagement leadership model which are valid for software projects. There are other similarities between what we have found as required qualities of software project managers and those based on Rose et al (2007). These qualities consist of:

1. protecting team members against client or higher level managers in our results corresponds to protecting team members in different situations such as customer or managers’ interference in results of studies by Rose et al
2. posing questions based on individual interests and indirect stimulation in our results corresponds to being main motivator and commitment developer in results of studies by Rose et al
3. accompany with team members and spend time about technical problems in our results corresponds to sparring partner in results of studies by Rose et al
4. being decisive when required in our results corresponds to decision maker in results of studies by Rose et al
5. Good communication skills just same as each other in results of both studies
Hence we can state that although Rose et al (2007) have not used any transformational leadership theory as the basis of their research, the results they have attained are in line with our findings; which found engaging leadership a suitable style for software project managers.

What Mohan Thite (2000) have arrived at; has the most similarity with our findings. Both ascertained the value of transformational leadership behaviors and technical leadership for software projects. The major difference between two set of findings is that, Thite have used transformational leadership model of Bass and Avolio (1990) but we have used engaging leadership model suggested by Alimo-Metcalfe & Alban-Metcalfe (2001). The model they have used is among heroic models of leadership whereas ours is among non-heroic models. Furthermore they have found contingent reward and active monitoring of exceptions as two transactional behaviors required for software projects. In our findings contingent rewards has no place. It points to the fact that our studied teams don’t recognize financial agreements among the responsibilities of the project manager.

Active monitoring of exceptions however is called as a transactional behavior by Thite, is shown itself as indirect supervision in our results. The reason why we have not perceived it as a transactional behavior is hidden in two facts. First reason is related to the way this behavior should be performed, which is indirect kind of supervision. This is noticeable that even Thite has mentioned that active monitoring is a transactional behavior but it benefits the team when it’s performed in a transformational style. Second reason is related to the difference between underlying transformational leadership models. In engaging leadership we have two dimensions, supporting a developmental culture and enabling, which could be fostered by indirect supervision but it seems that in the model suggested by Bass and Avolio (1990) no dimension supports active monitoring of exceptions. This difference in its turn might roots in the fact that engaging leadership or nearby transformational leadership is developed based on studies of middle to chief executive managers. This means that active monitoring of exception might be considered as a transactional behavior for a top level manager but a transformational one for a direct or middle level manager.
One of the requirements we have found for a software project manager is related to his potential to be considered as an example. Although in the engaging leadership model there is no dimension directly related to this attribute; Alban-Metcalfe & Alimo-Metcalfe (2005) have indicated a set of characteristics for nearby or close charismatic leaders which could convert him to an example for his team members. The reason why this attribute gets specific importance in the field of software may lie in the fact that majority of people in this field suffer from the lack of experience and knowledge. (Baccarini et al., 2004) Hence scarce knowledgeable and experienced people are highly respected and considered as professional examples. Less-experienced people need to be nurtured with experiential knowledge from mentors (Hawk et al., 2012) and leader is the one who could play the role of a mentor.

Being a calm person is another required characteristic for software project manager which is not directly related to any dimension of engaging leadership model. Rose et al. (2007) explains that manager of a software project should be able to manage his own stress and remain approachable in any difficult situation he confronts with. This ability should be well-stabilized as a part of manager’s personality; since being placed in stressful situation is very commonplace in IT projects. (Rutner et al., 2011)

Team management ability is another prerequisite for a software project manager since team has a specific position in complex problem domains such as software. (Demirors et al., 1997) Team management is among management competencies for software project managers in competencies pyramid (Rose et al., 2007) Manager of a software project is responsible for qualified knowledge transfer and creating an appropriate project atmosphere; which are critical to success of software projects.

Put it in a nutshell; we suggest that three new attributes be considered essential for software project managers. These attributes encompass setting an example, calm personality and high level of team management abilities.
4.4. Discussion

Engaging leadership model although suggests an appropriate pattern for managing software projects, could be revised to be more specific to software projects. It should be noticed that appropriateness of engaging leadership model should be extensively examined both quantitatively and qualitatively; since this study tried to find the materialization of transformational leadership in software projects for the first time. Furthermore this study is limited to a particular set of samples. Hence results of this study should not be generalized for whole software society. The generalization could be done only in case of similar criteria as mentioned in case selection criteria section (section 3.6).

While this study is limited to description of leadership ideals from team members’ point of view, other researches should focus on the leadership requirements and definitions, provided by other groups of stakeholders such as client team and higher level management of software organizations.

The outcome of this study could be considered as the starting point of more attempts to remedy the chronic ill of software project management, the ignorance of leadership studies in this field.

Future studies should look at leadership essentials in software projects taking into account particular specifications of software projects; since these specifications might considerably affect the climate of software teams and behaviors of software experts; those specifications which discern software projects from other types projects. Furthermore, very rapid changes in software world including emergence of new software implementation technologies and the growing need of new software products should be reflected upon; in all leadership researches within this field. Without such a consideration results of studies in this field might easily become obsolete.

Although this study expands the borders of software leadership researches to package implementation projects, software development and package implementation are only two types of software projects. Other types of IS projects such as system
enhancement, consultancy, system migration, infrastructure implementation etc might need different leadership requirements. (Cadle & Yeates, 2008) Thus other researches should look at leadership from the eyes of teams working on those types of IS projects as well.

One fact to be considered is that however grounded theory is a credible data analysis method, it might be subject of criticisms, especially when this method is utilized by a single researcher. Themes selected at the analysis phase of this study, are based on personal interpretation of researcher from the data. This fact reinforces aforementioned necessity of examining findings of this study by other researchers.
5. References


