Information & Management 46 (2009) 388–396

Contents lists available at ScienceDirect

Information & Management

journal homepage: www.elsevier.com/locate/im

Antecedents and consequences of team memory in software development projects

Halit Keskin *

Science and Technology Studies, Gebze Institute of Technology, Turkey

ARTICLE INFO

Article history:
Received 11 September 2007
Received in revised form 18 June 2009
Accepted 19 July 2009
Available online 5 August 2009

Keywords:
Software development
Product development
Team memory
Declarative memory
Procedural memory
Team culture

ABSTRACT

Team memory is important, yet it is rarely addressed in papers on software development. We investigated the antecedents and consequences of team memory (both declarative and procedural) in software development projects. By examining 67 projects in the IT departments of 38 firms, we found, using PLS that customer orientation and innovation orientation was positively associated with both declarative and procedural memory, social responsibility was positively related to declarative memory, and systematic management control were negatively associated with declarative memory but positively associated with procedural memory. We also found that: declarative memory was positively related to the market success of the software, and procedural memory was positively related to speed-to-market (launching software faster) to the extent that memory was dispersed throughout the project team. Managerial and theoretical implication were further discussed.

1. Introduction

Software development is a complex problem-solving process involving individuals and teams with different and often competing goals, interests, and responsibilities, and a variety of team practices about the communication and coordination of knowledge within the teams. However, team memory (TM), which involves storing knowledge, skills and procedures for future use is rarely discussed in the software development literature [26]. Reich [30] identified five broad principles of knowledge management in IT projects: a climate for learning, knowledge levels, knowledge channels, *TM*, and knowledge risks. However, Haseman et al. [11] suggested that the effectiveness of TM for IT groups remained an open question.

TM may be either *declarative* (facts, events, and the inventory of knowledge) or *procedural* (group skills or how things are done) [21]. TM is needed in interpreting and filtering information, and guiding and influencing actions, beyond those of traditional resources.

Although it is imperative to investigate the role of the contents of TM on the software development project's success, from a managerial point of view, the antecedents of the *contents* of TM should also be investigated. Results of such investigations could help managers to understand how to increase the project teams' knowledge and skills, and apply them for a success in development. Berthon et al. [2] stated that "Culture specifies the value and assumption content of collective memory, while memory development delimits memory potential or capacity." Thus, we argued that if team culture is the context in which software development occurs, it is likely to influence the TM during software development projects.

Finally, because the usage of the TM depends on environmental factors [1] as well as organizational and team processes, the moderator roles of TM characteristics and project success should be investigated. Most prior studies of TM investigated the moderating roles of environmental turbulence and change [3], ignoring team characteristics.

As software development activities are performed in rapidly changing environments, we investigated the *dispersion of memory* (the extent to which the knowledge is diffused throughout the project team [33]), as a characteristic of TM, rather than environmental turbulence or change. Because the required knowledge and skills for successful software development is dispersed through the team and organization, shared understanding and integration of information, knowledge and skills among project team members becomes a key factor for the relationship between how memory is manifested and project outcome. For instance, Dew et al. [8] stated that "The dispersion of knowledge matters in the production of new knowledge because when different people know different things the combination of dispersed bits of incomplete and frequently contradictory knowledge by an individual, or group of individuals,

* Correspondence address: Gebze Yüksek Teknoloji Enstitüsü, Isletme Fakultesi, Istanbul Cad. No. 101, 41400 Gebze-Kocaeli, Turkey. Fax: +90 262 653 8490.
E-mail addresses: keskin@gyte.edu.tr, keskin_28@yahoo.com.

0378-7206/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.im.2009.07.003

sometimes results in the creation of new knowledge that no individual had previously foreseen. As a result, new knowledge enters into the world."

Our study elaborated on research dealing with declarative and procedural memory and their antecedents and consequences to enhance the socio-cognitive view of software development teams and project management. In particular, as shown in Fig. 1, we investigated how teams enhance their information/knowledge stock, skills and behaviors, the way they routinized procedures; and the impact of TM on software development project outcomes. These issues have been long ignored.

1.1. Team memory and software development

Declarative memory includes knowledge of customer preferences, products features (e.g., packaging), organization's goals and business objectives, its market conditions and strategies, competitive position, and technological and programming knowledge. As it involves factual knowledge, it allows software project personnel to analyze new problems, see cause and effects, draw conclusions based on past events and choose a suitable response to a problem, and apply the knowledge in a variety of domains from customized software products to large scale radical ones.

Procedural memory helps people to form skills tied to the program coding, testing and software proficiency, and forms or establishes routines to aid in team member cooperation, interaction, and other project activities. It contains:

- procedures to handle routine program coding, and identify customer needs, deal with customer complaints and employee training, and
- decisions and actions, and technology, such as intranets.

Accordingly, without understanding the memory aspect of software development, it would be difficult to improve software development processes and the product's market success. Indeed, TM:

- reduces duplication of efforts. D'Adderio [6], for instance, pointed out that "Keeping track of past and evolving configurations can help reconstructing the 'whys' and 'hows' behind a part's design, test or manufacturing..."
- provides information that reduces transaction costs, and contributes to effective and efficient decision making. Substantial time and costs of software development can be saved by reusing parts of the software processes knowledge or process models [29].
- supports the coordination of work during software development. Since, software related tasks are highly interdependent, infrastructure activities, and other support services, are knowledge intensive, requiring sequential and reciprocal interdependence between team members [24]. TM integrates information and knowledge over time among team members, and thus the team can maintain cohesion [13].
- supplies guidelines for interpreting and managing ambiguous situations. As standard policies and procedures reflect practices that were validated by collective experience and interpersonal examination and discussion, project teams can apply technical and market knowledge more easily to a variety of complex and novel situations [12]. TM helps project teams recombine their successful experiences to provide better understanding of the problem definition and a superior solution. Zack [37], for instance, noted that ambiguity can be resolved by acquiring contextual or explanatory knowledge either from others or from learned experience.

However, we should know more about how TM affects the project outcomes (e.g., speed-to-market, software development cost, and market success), and how we can increase the TM to produce a successful project.

1.2. Hypothesis development

1.2.1. Antecedents of team memory contents in software development

Team culture reflects deeper aspects of team life and artifacts, such as rituals and procedures, arise from it [9]. We argue that team culture can act as a facilitator of TM contents; TM contents are likely to be revealed in a team's culture [23]. De Long and Fahey [7] mentioned that culture shapes the processes by which new knowledge is created, legitimated and distributed. However, culture may be conceptualized in different ways, such as a collection of values, behavioral norms, or myths and stories. We adopted the definition of O'Reilly et al. [27]: *team culture* is a set of core values shared by group members. Also *team values* are socially oriented and provide unique constructs that describe the characteristics of teams [20]. It also provides justification for appropriate member behavior and we note that they serve to differentiate declarative and procedural memory development in software development projects. Tsui et al. [34] identified them as employee orientation, customer orientation, systematic management control, innovativeness orientation, and social responsibility.

Employee orientation is the degree to which managers show concern, warmth, empathy, and individualized consideration for the ongoing needs and welfare of employees [16] – software development projects are staff – rather than equipment-oriented [28], the introduction of new skills, communication and interpersonal skills training, and the development of group dynamics, will enhance the gathering and retaining of factual knowledge and procedural skills of the team members. Therefore:

H1. Employee orientation in a team will be positively related to the development of (a) declarative and (b) procedural memory in a software development project.

Customer orientation is to the extent to which a customer is involved in the software development process and improvements, including the extent to which customer feedback is used in continuous improvement [31]. As customer orientation manifests itself in close communication, reciprocity, trust and commitment with customers, project teams are more able to receive information and accumulate knowledge about customers [36]. Software development teams with a customer orientation store knowledge accumulated through experience and develop a knowledge base of them for future use. Wang and Lo [35] noted that customers are critical co-developers of knowledge and competence. Therefore:

H2. Customer orientation in a team will be positively related to the development of (a) declarative and (b) procedural memory in a software development project.

Innovativeness orientation is a measure of a team's openness to new ideas and propensity to change through adopting new technologies, resources, skills, and administrative systems. A project team with an innovativeness orientation is open to new ideas and values creativity and adaptability, and actively communicates information and ideas. A predisposition to openness provides room for new viewpoints, allowing a constant renewal and improvement of group knowledge. Also, it helps project teams develop new skills and procedures. Indeed, project teams facilitate the dissemination of common beliefs, values, and understanding and promote team-wide understanding of acquiring, transferring, and using knowledge; team members will: develop appropriate skills and regulate procedures, processes and activities to ensure that customer and technical issues are solves in a timely fashion. Therefore:

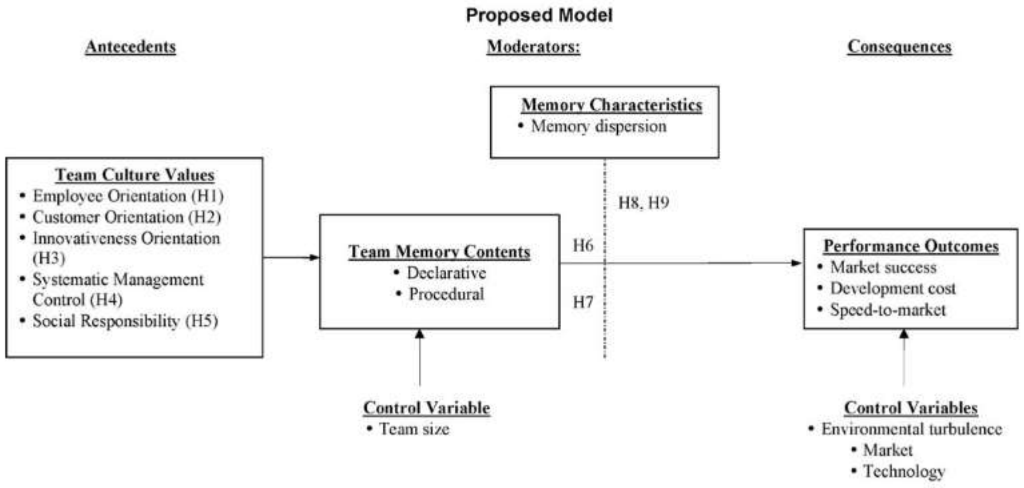

Fig. 1. Proposed model.

H3. Innovativeness orientation in a team will be positively related to the development of (a) declarative and (b) procedural memory in a software development project.

Systematic management control involves activities that encourage employees to have a set of project goals, and to explicit rules that reduce any perceived ambiguity. Accordingly, it motivates people to be more open to receive knowledge and to disseminate it, and to follow established procedures [22]. Jiang et al. [15] noted that management control lead to defining and documenting how the work was to be done and establishing performance guidelines and standards through task feedback. Therefore:

H4. Systematic management control in a team will be positively related to the development of (a) declarative and (b) procedural memory in a software development project.

Social responsibility is to the obligation of a project team member to contribute to social betterment above and beyond their role in the marketing of their products. Social responsibility provides implicit guidelines for employee conduct; encourages employees to become involved in their communities, provides a basis for dialogue between management and the team, and leads to bonding of stakeholders [32]. Project teams develop a knowledge base to communicate with environmental goals and to accomplish the customer needs. Therefore:

H5. Social responsibility in a team will be positively related to the development of (a) declarative and (b) procedural memory in a software development project.

1.2.2. Consequences of memory contents

In a software development context, given that TM increases efficiencies and decreases possible repeated errors, and because of its potential in extracting marketing and technological synergy between the new product and the firm's existing competencies, we argue that it can reduce development cost; and increase speed-to-market and market success. For instance Kyriakopoulos and Ruyter [19] stated that, because of procedural memory, routines and standards speed up execution, reduce costs, and ensure reliable organizational action and that well-exercised processes increase product quality and customer satisfaction, and that procedures, which facilitate habits and motor action, improve exchange and mutual understanding across functions for enhanced project outcome. Also, declarative memory helps teams in identifying patterns in external events and in selecting actions producing outcomes that are coherent and suitable to market conditions. Therefore:

H6. Declarative memory will be positively associated with (a) less development cost, (b) increased speed-to-market, and (c) a higher market success of software products.

H7. Procedural memory will be positively associated with (a) less development cost, (b) increased speed-to-market, and (c) a higher market success of software products.

Even though the TM impacts the performance of software development projects, we argue that memory dispersion throughout the project teams moderate the relationship between TM contents and project outcomes. The way memory is shared in organizations and teams is more important than its absolute amount; a lack of knowledge sharing is a major barrier to the effective management of knowledge and skills in team. Memory dispersion reduces unnecessary time-consuming activities, enhances efficiency, and promotes the dissemination of innovative ideas, creativity, and skills throughout the teams. Therefore:

H8. As memory dispersion increases, the positive effect of declarative memory on (a) development cost will be reduced, (b) speed-to-market, and (c) market success of the software products will be increased.

H9. As memory dispersion increases, the positive effect of procedural memory on (a) development cost will be reduced, (b) speed-to-market, and (c) market success of the software products will be increased.

1.3. Research design

1.3.1. Measures and sampling

To test the hypotheses, multi-item scales were adopted from prior studies. Each construct was measured using a 5-point Likert scale ranging from "strongly disagree" (1) to "strongly agree" (5). However, as team size affects memory, the project team size, a control variable, was assessed with a ratio scale. Also, consistent with the literature, environmental turbulence was used as a control variable for project outcome variables. Definitions of each construct are provided in Table 1.

By using the parallel-translation method, items were first translated into Turkish by one person and then retranslated into English by a second to make sure that the *meanings of question*

Table 1
Operational definitions of research construct.

Construct	Operational definitions
Declarative memory	The degree of knowledge and expertise of the project team.
Procedural memory	The degree of skills, abilities, and procedures of the project team.
Memory dispersion	The degree of consensus or shared knowledge among the project team members.
Market success	The extend which the software product meets (or exceeds) managerial, profit, and market expectations.
Development cost	The extend which the project was developed and launched at the pre-determined cost.
Speed-to-market	The extend of a team's ability to develop and launch a new software product rapidly.
Employee orientation	The degree to which the managers and team members show concern, warmth, empathy, and individualized consideration for the ongoing needs and welfare of other people in the team.
Customer orientation	The extent to which customer focus and orientations is emphasized during the process.
Innovativeness orientation	The degree of a team's openness to new ideas and propensity to change through adopting new technologies.
Systematic management control	The degree to which the managers' activities to set rules and goals during the development process.
Social responsibility	The degree of project team's focus on the social profits.

items were correctly transformed from English to Turkish. The two translators then jointly reconciled all differences. The suitability of the Turkish version of the questionnaire was then assessed by five part-time graduate students working in industry and involved in at least one software development project. After refining the questionnaire, based on interviews with the pre-test subjects. The questionnaire was modified accordingly, and the questionnaire items can be found in Appendix A.

The initial sample consisted of 100 firms located in Istanbul, which have affiliations with European and American companies. First, the firms' IT department managers were contacted by telephone and the aim of our study was explained to them. Of the 100 firms contacted, 84 agreed to work with us on this study. To assess the software product performance more accurately, the software products used in the study must have been commercialized and launched into the marketplace or used in the organization at least 6 months prior to the assessment. We asked that *at least two respondents*, who were *the most knowledge about the projects*, fill out our surveys to avoid single-source bias. Those respondents were expected to serve as key informants for others who worked on the same project team, representing the beliefs or attitudes of the team. These respondents were also members of other "teams" within the same IT department. Here we avoided collecting information from the same respondent for different projects to prevent mixing perceptions of the different projects. After qualifying the respondents, each was informed that his/her responses would remain anonymous and would not be linked to them either individually, their companies, or software products. This was done to assure anonymity, increasing the motivation to cooperate without fear of potential reprisals. Also, we assured respondents that there were no right or wrong answers and that they should answer questions as honestly and forthrightly as possible. Further, we developed a cover story to make it appear that the measurement of the predictor variable was not connected with or related to the measures of the criterion variable. These procedures reduced people's evaluation apprehension and made them less likely to edit their responses to be more socially desirable, lenient, or consistent with how they thought the researchers wanted them to respond.

Of the 84 firms' IT departments that agreed to participate, 38 completed our questionnaires, returning 142 surveys from 67 software development projects. Thus, usable data for our analysis was the 67 software development projects with average of two respondents from each. In the sample: the projects were related to the information and communication technologies (48%), business services (33%), and financial services (19%). The respondents were: engineer/programmer (62%), senior engineer/technical leader (14%), IS specialist/analyst (11%), product/project managers (10%), owner (2%), and department managers (1%). The duration of the projects were: 4–6 months (48%), less than 3 months (21%), 7–9 months (14%), 10–18 months (12%), and over 18 months (5%).

1.4. Analysis and results

We used the partial least squares (PLS Graph 3.0,) approach to path modeling to estimate the measurement and structural parameters in our structural equation model (SEM). Before doing any analysis, since our unit of analysis was the "*project team*," we aggregated the team scores of *each question item*. The inter-rater agreement (r_{wg}) on team level measures had to be demonstrated and all r_{wg} values ranged from 0.72 to 0.90. This is well above the 0.60 benchmark, indicating a satisfactory level of inter-rater agreement for each aggregate measure in a project team.

1.4.1. Measurement validation

To assess the psychometric properties of the measurement instruments, we used a null model with no structural relationships and evaluated reliability by means of composite scale reliability (CR) and average variance extracted (AVE) [18]. For all measures, PLS-based CR is well above the cut-off value of 0.70, and AVE exceeds the normal 0.50 cut-off value. In addition, we evaluated convergent validity by inspecting the standardized loadings of the measures on their respective constructs and found that all measures exhibited standardized loadings that exceeded 0.60. We next assessed the discriminant validity of the measures; the square root of AVE for each construct was greater than the latent factor correlations between pairs of constructs (see Table 2).

1.4.2. Hypothesis testing

We used the PLS Graph 3.0 and Bootstrapping resampling method to test the statistical significance. This procedure entailed generating 500 subsamples of cases randomly selected, with replacement, from the original data. Path coefficients were then generated for each randomly selected subsample. *T*-statistics were calculated for all coefficients, based on their stability across the subsamples, indicating which links were statistically significant.

We employed a hierarchical approach to test our hypotheses, in this we first estimated a model with the main effects only and then added the memory dispersion and interaction effect, for which we normalized each question item of memory dispersion, declarative and procedural memory, and then multiplied the results of these normalized question items. This product approach, suggested by Chin et al. [4] resulted in the addition of 15 product indicators for the latent variables. These represent the moderators of declarative memory and its memory dispersion, and procedural memory and its memory dispersion. The result for the final model includes the interaction effect, as shown in Table 3. The results illustrate that many of our hypotheses were confirmed. With regard to antecedents, we found that a value of customer orientation ($\beta = 0.40$, $p < 0.05$), innovativeness orientation ($\beta = 0.38$, $p < 0.01$) and social responsibility ($\beta = 0.32$, $p < 0.05$) was positively associated with declarative memory, and systematic management control ($\beta = -0.28$, $p < 0.05$) was negatively associated with

Table 2
Correlations of latent variables.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 Market success	(0.83)													
2 Speed-to-market	0.55***	(0.84)												
3 Development cost	0.51***	0.60***	(0.88)											
4 Declarative memory	0.51***	0.31***	0.24***	(0.81)										
5 Procedural memory	0.45***	0.39***	0.27***	0.57***	(0.91)									
6 Employee orientation	0.33***	0.41***	0.26***	0.31***	0.44***	(0.75)								
7 Customer orientation	0.40***	0.45***	0.34***	0.54***	0.56***	0.50***	(0.79)							
8 Innovativeness orientation	0.50***	0.46***	0.45***	0.49***	0.55***	0.63***	0.52***	(0.87)						
9 Systematic management control	0.43***	0.45***	0.36***	0.34***	0.69***	0.50***	0.63***	0.49***	(0.76)					
10 Social responsibility	0.40***	0.35***	0.27***	0.48***	0.59***	0.47***	0.58***	0.51***	0.59***	(0.85)				
11 Memory dispersion	0.52***	0.39***	0.34***	0.49***	0.57***	0.30***	0.58***	0.52***	0.55***	0.51***	(0.81)			
12 Technology turbulence	0.22***	0.18**	0.13	0.14	0.07	0.40***	0.22***	0.31***	0.19**	0.23***	0.30***	(0.80)		
13 Market turbulence	0.39***	0.29***	0.36***	0.44***	0.44***	0.39***	0.46***	0.44***	0.40***	0.47***	0.43***	0.20**	(0.84)	
14 Team size	−0.30***	−0.29***	−0.29***	−0.09	0.02	−0.22***	−0.19**	−0.28***	−0.07	−0.22***	−0.18**	−0.21**	−0.34***	–
Mean	3.42	3.29	3.06	4.00	3.53	3.71	3.82	3.78	3.02	3.47	3.42	4.17	3.64	12
Standard deviation	0.74	0.84	0.78	0.55	0.77	0.65	0.63	0.80	0.77	0.85	0.82	0.71	0.72	30

Diagonals show the square root of AVEs.
* $p < 0.1$.
** $p < 0.05$.
*** $p < 0.01$.

declarative memory, supporting H2a, H3a, H5a, but not support H4a. Also, we could not find any significant statistical association between employee orientation and declarative memory ($\beta = -0.10$, $p > 0.1$); thus this did not support H1a. Regarding the role of team culture values on procedural memory, we found that customer orientation ($\beta = 0.25$, $p < 0.05$), innovativeness orientation ($\beta = 0.31$, $p < 0.1$) and systematic management control ($\beta = 0.31$, $p < 0.01$) were positively associated with procedural memory, supporting H2b, H3b, and H4b. However, we could not find any significant association between procedural memory and employee orientation ($\beta = -0.13$, $p > 0.1$) and social responsibility ($\beta = 0.20$, $p > 0.1$), thus not supporting H1b and H5b.

Regarding the consequences of declarative and procedural memory, we only found that declarative memory was positively associated with market success of software products directly ($\beta = 0.29$, $p < 0.05$), supporting H6a. However, we could not find any direct relationship between declarative memory and development cost, and speed-to-market; thus H6b and H6c were not supported. Also there was no statistical association between procedural memory and any project outcome, Thus H7 was not supported.

In addressing the moderating hypotheses, we found a significant statistical association between interaction effect, moderator, of memory dispersion and procedural memory (memory dispersion × procedural memory), and speed-to-market ($\beta = 0.38$, $p < 0.1$), supporting H9b. Also, the significance of the difference in the R^2 statistics of between the main model and full alternative model reflected the increased explanation of the dependent variable by the inclusion of the direct link. Specifically, effect size (f^2) was calculated as $(R^2\text{full} - R^2\text{excluded})/(1 - R^2\text{full})$. The f^2 statistic was computed as multiplying f^2 by $(n - k - 1)$, where n was the sample size and k was the number of independent constructs, which provided a pseudo F test for the significance of the f^2 statistic with 1 and $(n - k)$ degrees of freedom. The R^2 value for speed-to-market is 0.26 for the main effect model and 0.42 for the alternative full model. Then, f^2 was 0.276 and $F(1, 58) = 3.82$, $p > 0.05$. However, we could not find any statistical relationship between the interaction effect of memory dispersion and declarative memory, and market success and less development cost; thus H8 was not supported. Finally, our results, including the memory dispersion variable as moderator (Table 3), also showed that team culture values explained the 45% of the variance ($R^2 = 0.45$) in declarative memory, and 65% of variance ($R^2 = 0.65$) in procedural memory; declarative and procedural memory and control variables explained 25% of the variance ($R^2 = 0.25$) in development cost, 42% of variance ($R^2 = 0.42$) in speed-to-market, and 43% of variance ($R^2 = 0.43$) in market success in the final model.

2. Discussion and implications

In our study, we presented a model to understand the interrelationships among team culture values, declarative and procedural memory, and project performance in software development project teams.

First, we empirically showed the role of declarative and procedural memory on project outcomes. Our results demonstrated that declarative memory was important to enhance market success. Specifically, when knowledge and expertise was broadly distributed among team members, that the software product may have a greater chance to meet or exceed sales, volume, market share and customer expectations. However, we could not find any association between declarative memory, and speed-to-market or development cost.

Second, we empirically investigated the role of a team's cultural values on declarative and procedural memory. Our findings showed that when there is a focus on customer orientation and innovativeness orientation the project team will have well-defined procedures, a standard approach, strong skills, expertise and knowledge of people, in its software development efforts. However, an interesting finding was that systematic management control was positively related to procedural memory and negatively related to declarative memory. Finally, our results indicated that social responsibility only positively impacts declarative memory.

Our work suggests that managers should enhance team culture values to leverage the memory during the project.

3. Limitations

There are some methodological limitations to this study. Specifically, our research was prone to common method bias, since, in the questionnaire, the same respondents answered the dependent variable that answered the independent variable. We checked this potential problem with the Harman one-factor test. The results of an unrotated principal component analyses indicated that common method variance was not a problem as several factors with eigenvalue greater than 1 were identified—

Table 3
The results.

Hypothesis	Relationship	Path coefficient (β)	Results
H1a	Employee orientation → declarative memory	−0.10	Not supported
H1b	Employee orientation → procedural memory	−0.13	Not supported
H2a	Customer orientation → declarative memory	0.40**	*Supported*
H2b	Customer orientation → procedural memory	0.25**	*Supported*
H3a	Innovativeness orient → declarative memory	0.38***	*Supported*
H3b	Innovativeness orient → procedural memory	0.31*	*Supported*
H4a	Systematic management control → declarative memory	−0.28**	*Supported*
H4b	Systematic management control → procedural memory	0.31***	*Supported*
H5a	Social responsibility → declarative memory	0.32**	*Supported*
H5b	Social responsibility → procedural memory	0.20	Not supported
H6a	Declarative memory → development cost	−0.03	Not supported
H6b	Declarative memory → speed-to-market	0.09	Not supported
H6c	Declarative memory → market success	0.29**	*Supported*
H7a	Procedural memory → development cost	−0.05	Not supported
H7b	Procedural memory → speed-to-market	0.10	Not supported
H7c	Procedural memory → market success	0.01	Not supported
H8a	Memory dispersion × declarative memory → development cost	−0.19	Not supported
H8b	Memory dispersion × declarative memory → speed-to-market	−0.15	Not supported
H8c	Memory dispersion × declarative memory → market success	−0.9	Not supported
H9a	Memory dispersion × procedural memory → development cost	0.11	Not supported
H9b	Memory dispersion × procedural memory → speed-to-market	0.38*	*Supported*
H9c	Memory dispersion × procedural memory → market success	0.10	Not supported
	Memory dispersion → market success	0.23	
	Memory dispersion → speed-to-market	0.25	
	Memory dispersion → development cost	0.32*	
	Technological turbulence → market success	0.05	
	Technological turbulence → speed-to-market	−0.08	
	Technological turbulence → development cost	−0.06	
Control variables	Market turbulence → market success	0.20	
	Market turbulence → speed-to-market	0.07	
	Market turbulence → development cost	0.20	
	Team size → declarative memory	0.11	
	Team size → procedural memory	−0.23*	

Fit measures	Endogenous construct	Main effect model	Final model
R^2	Declarative memory	0.45	0.45
	Procedural memory	0.65	0.65
	Market success	0.39	0.43
	Speed-to-market	0.26	0.42
	Development cost	0.17	0.25

* $p < 0.1$.
** $p < 0.05$.
*** $p < 0.01$.

explaining 76.4%. Self-report data is most problematic for topics that generate strong sentiments. Market success, speed-to-market and development cost are not emotionally laden subject and, thus, less likely to be distorted by self-reporting.

Utilizing a cross-sectional design with questionnaires was another limitation of our study. However, as a cross-sectional field study, this research provided evidence of associations. In addition to the nature of data, the generalizability of sampling is another limitation. The study was conducted in a specific national context, i.e., Turkish firms. It is important to note that readers should be cautious when generalizing the results to different cultural contexts. Further, the sample size was relatively small. A larger sample may aid in demonstrating greater discrimination of the constructs.

4. Conclusion

We know little about the contents of TM in software development projects. In this study, we investigated the determinants and consequences of two of the TM components: declarative and procedural memory. Our findings demonstrated that team culture had a positive and negative influence on the two memories. Declarative memory positively impacts the market success of software, and procedural memory influences the speed-to-market when it is dispersed throughout the team.

Appendix A. Measures

Standardized loadings are in parentheses.

CR: composite reliability; AVE: average variance extracted; r_{wg}: inter-rater agreement.

*Denotes the dropped item, either they reduce the AVE less then 0.50, or they have low loading weights.

[r]Shows the reversed question items.

Employee orientation (*Adapted from* [34])

The followings were emphasized in our project team:

- promoting feeling-sharing among employees (0.71);

- emphasizing team building (0.79);
- encouraging cooperation (0.81);
- trusting in employees (0.77);
- fertilizing cooperative spirit (0.72);
- concerning for the individual development of employees (0.68);
- consideration among employees (0.67);
- caring about opinions from employees (0.78).

CR = 0.91

AVE = 0.56

$r_{wg} = 0.89$

Customer orientation (*Adapted from* [34])

Satisfying the need of customers at the largest scale was emphasized in our project team (0.75):

- Sincere customer service was emphasized in our project team (0.85).
- The idea of customer is number one was emphasized in our project team (0.78).
- Providing first class service to customers was emphasized in our project team (0.86).
- The profit of customer was emphasized extremely in our project team (0.69).

CR = 0.89

AVE = 0.62

$r_{wg} = 0.72$

Innovativeness orientation (*Adapted from* [34])

Our project team was:

- developing new products and services continuously (0.86);
- ready to accept new changes (0.90);
- able to adopt high-tech bravely (0.88);
- encouraging new ideas (0.86).

CR = 0.93

AVE = 0.77

$r_{wg} = 0.74$

Systematic management control (*Adapted from* [34])

Management in our project team was:

- keeping strictly working disciplines (0.79);
- having a clear standard on praise and punishment (0.65);
- possessing a comprehensive system and regulations (0.80);
- setting a clarity goals for employees (0.79);

CR = 0.85

AVE = 0.58

$r_{wg} = 0.81$

Social responsibility (*Adapted from* [34])

Our project team showed social responsibility (0.90):

- The mission of our project team was to serve benefits for people (0.75).
- Our project team was emphasizing on economic as well as social profits (0.90).

CR = 0.89

AVE = 0.73

$r_{wg} = 0.76$

Speed-to-market (*Adapted from* [17])

- This product (software) was developed and launched (fielded) faster than the major competitor for a similar product (0.77).
- This product (software) was completed in less time than what was considered normal and customary for our industry (0.90).
- This product (software) was launched on or ahead of the original schedule developed at initial project go-ahead (0.86).
- Top management was pleased with the time it took us from specs to full commercialization (0.80).

CR = 0.90

AVE = 0.70

$r_{wg} = 0.84$

Market success of software products (*Adapted from* [5])

Our product (software):

- met or exceeded volume expectations (0.88);
- met or exceeded the first year number expected to be produced and commercialized (0.85);
- met or exceeded overall sales expectations (0.88);
- met or exceeded profit expectations (0.89);
- met or exceeded return on investment expectations (0.88);
- met or exceeded senior management expectations (0.78);
- met or exceeded market share expectations (0.76);
- met or exceeded customer expectations (0.66).

CR = 0.95

AVE = 0.69

$r_{wg} = 0.87$

Development cost (*Adapted from* [10])

Our product (software):

- Was launched within or under the original budget[r] (0.83).
- Came in at or below cost estimation for development[r] (0.91).
- Came in at or below cost estimation for production[r] (0.93).

CR = 0.92

AVE = 0.79

$r_{wg} = 0.86$

Team memory contents (*Adapted from* [19])

Procedural memory

In our project team, we had:

- well-defined procedures (0.88);
- a standard approach (0.94);
- strong skills and abilities (0.89).

CR = 0.93

AVE = 0.82

$r_{wg} = 0.90$

Declarative memory
Our team had:

- a great deal of knowledge about project activities (0.78);
- strong expertise (0.86);
- knowledgeable people who has technical and project experiences before (0.77).

CR = 0.85

AVE = 0.65

$r_{wg} = 0.89$

Team memory characteristics (*Adapted from* [25])
Team memory dispersion
Our project team is successful in:

- product design (0.74);
- brand name (0.88);
- packaging (0.88);
- promotional content (0.80);
- product quality level (0.76).

CR = 0.91

AVE = 0.66

$r_{wg} = 0.85$

Technological turbulence (*Adapted from* [14])

- The technology in this product area is changing rapidly (0.60).
- Technological changes provide big opportunities in this product area (0.88).
- It is very difficult to forecast where the technology in this product area will be in the next 5 years*.
- A large number of new product ideas in this area have been made possible through technological breakthroughs (0.90).
- Technological developments in this product area are rather minor*.

CR = 0.83

AVE = 0.64

$r_{wg} = 0.76$

Market turbulence (*Adapted from* [14])

- In our kind of business, customer's product preferences change quite a bit over time*.
- Our customers tend to look for new products all the time*.
- We are witnessing demand for our products and services from customers who never bought them before (0.85).
- New customers tend to have product-related needs that are different from those of our existing customers*.
- We cater to much the same customers that we used to in the past[r] (0.81).

CR = 0.83

AVE = 0.70

$r_{wg} = 0.77$

References

[1] A.E. Akgün, G.S. Lynn, J. Byrne, Antecedents and consequences of unlearning in new product development teams, Journal of Product Innovation Management 23, 2006, pp. 73–88.
[2] P. Berthon, L.F. Pitt, M.T. Ewing, Corollaries of the collective: the influence of organizational culture and memory development on perceived decision-making context, The Journal of Academy of Marketing Science 29, 2001, pp. 135–150.
[3] D.R. Chang, H. Cho, Organizational memory influences new product success, Journal of Business Research 61, 2008, pp. 13–23.
[4] W.W. Chin, B.K. Marcolin, P.R. Newsted, A partial least squares latent variable approach for measuring interaction effects: results from a Monte Carlo simulation study and an electronic mail emotion/adoption study, Information Systems Research 14, 2003, pp. 189–217.
[5] R.G. Cooper, E.J. Kleinschmidt, Success factors in product innovation, Industrial Marketing Management 16, 1987, pp. 215–223.
[6] L. D'Adderio, Configuring software, reconfiguring memories: the influence of integrated systems on the reproduction of knowledge and routines, Industrial and Corporate Change 12, 2003, pp. 321–350.
[7] D. De Long, L. Fahey, Diagnosing cultural barriers to knowledge management, The Academy of Management Executive 14, 2000, pp. 113–128.
[8] N. Dew, S.R. Velamuri, S. Venkataraman, Dispersed knowledge and an entrepreneurial theory of the firm, Journal of Business Venturing 19, 2004, pp. 659–679.
[9] C.B. Gibson, M.E. Zellmer-Bruhn, Minding your metaphors: applying the concept of teamwork metaphors to the management of teams in multicultural contexts, Organizational Dynamic 31, 2002, pp. 101–116.
[10] A. Griffin, A. Page, An interim report on measuring product development success and failure, Journal of Product Innovation Management 10, 1993, pp. 291–308.
[11] W.D. Haseman, D.L. Nazareth, S. Paul, Implementation of a group decision support system utilizing collective memory, Information & Management 42, 2005, pp. 591–605.
[12] W. He, Q. Qiao, K. Wei, Social relationship and its role in knowledge management systems usage, Information & Management 46, 2009, pp. 175–180.
[13] B. Hooff, M. Huysman, Managing knowledge sharing: emergent and engineering approaches, Information & Management 46, 2009, pp. 1–8.
[14] B.J. Jaworski, A.K. Kohli, Market orientation: antecedents and consequences, Journal of Marketing 57, 1993, pp. 53–70.
[15] J.J. Jiang, G. Klein, H. Hwang, J. Huang, S. Hung, An exploration of the relationship between software development process maturity and project performance, Information & Management 41, 2004, pp. 279–288.
[16] R. Kantsperger, W.H. Kunz, Managing overall service quality in customer care centers: empirical findings of a multi-perspective approach, International Journal of Service Industry Management 16, 2005, pp. 135–152.
[17] E.H. Kessler, A.K. Chakrabarti, Speeding up the phase of new product development, Journal of Product Innovation Management 16, 1999, pp. 231–247.
[18] M. Kleijnen, K. Ruyter, M. Wetzels, An assessment of value creation in mobile service delivery and the moderating role of time consciousness, Journal of Retailing 83, 2007, pp. 33–46.
[19] K. Kyriakopoulos, K. Ruyter, Knowledge stocks information flows in new product development, Journal of Management Studies 41, 2004, pp. 1469–1498.
[20] M.J. Lankau, A. Ward, A. Amason, T. Ng, et al., Examining the impact of organizational value dissimilarity in top management teams, Journal of Managerial Issues 19, 2007, pp. 11–36.
[21] G.S. Lynn, A.E. Akgün, A new product development learning model: antecedent and consequences of declarative and procedural knowledge, International Journal of Technology Management 20, 2000, pp. 490–510.
[22] S.D. Miller, R.A. DeCarlo, A.P. Mathur, J.W. Cangussu, A control-theoretic approach to the management of the software system test phase, Journal of Systems and Software 79, 2006, pp. 1486–1503.
[23] D.Z. Milosevic, Selecting a culturally responsive project management strategy, Technovation 22, 2002, pp. 493–508.
[24] R. Mirani, Procedural coordination and offshored software tasks: lessons from two case studies, Information & Management 44, 2007, pp. 216–230.
[25] C. Moorman, A.S. Miner, The impact of organizational memory on new product performance and creativity, Journal of Marketing Research 34, 1997, pp. 91–106.
[26] J. Morrison, Team memory: information management for business teams, in: Proceedings of the 26th Hawaii International Conference in Systems Sciences, vol. 4, 1993, pp. 122–131.
[27] C.A. O'Reilly, J. Chatman, D.F. Caldwell, People and organizational culture: a profile comparison approach to assessing person–organization fit, Academy of Management Journal 34, 1991, pp. 487–516.
[28] L. Ozdamar, E. Alanya, Uncertainty modelling in software development projects (with case study), Annals of Operations Research 102, 2001, pp. 157–178.
[29] M.H. Raymond, G. McCray, R. Purvis, T. Roberts, Exploiting organizational knowledge in developing IS project cost and schedule estimates: an empirical study, Information & Management 44, 2007, pp. 598–612.
[30] B.H. Reich, Managing knowledge and learning in IT projects—a conceptual framework and guidelines for practice, Project Management Journal 38, 2007, pp. 5–19.

[31] O. Rexfelt, E. Rosenblad, The progress of user requirements through a software development project, International Journal of Industrial Ergonomics 36, 2006, pp. 73–81.

[32] J. Smith, A. Adhikari, R.H. Tondkar, Exploring differences in social disclosures internationally: a stakeholder perspective, Journal of Accounting and Public Policy 24, 2005, pp. 123–151.

[33] A.M. Stewart, G.W. Mullarkey, J. Craig, Innovation or multiple copies of the same lottery ticket: the effect of widely shared knowledge on organizational adaptability, Journal of Marketing Theory and Practice 11, 2003, pp. 25–46.

[34] A.S. Tsui, Z. Zhang, H. Wang, R.K. Xin, J.B. Wu, Unpacking the relationship between CEO leadership behavior and organizational culture, The Leadership Quarterly 17, 2006, pp. 113–137.

[35] Y. Wang, H. Lo, Customer-focused performance and the dynamic model for competence building and leveraging: a resource-based view, The Journal of Management Development 22, 2003, pp. 483–527.

[36] E.T.G. Wang, S. Shih, J.J. Jiang, G. Klein, The relative influence of management control and user–IS personnel interaction on project performance, Information and Software Technology 48, 2006, pp. 214–220.

[37] M.H. Zack, The role of decision support systems in an indeterminate world, Decision Support Systems 43, 2007, pp. 1664–1674.

Halit Keskin is an associate professor of science and technology studies in the School of Business Administration at Gebze Institute of Technology, Turkey. He received his Ph.D. in management and organization from Gebze Institute of Technology. His research interests include technology and innovation management, knowledge management and human resource management in high-tech firms.