PAPERS

Exploring Risk Dimensions in the Indian Software Industry

Arpita Sharma, Professional Development Department, Jaypee Institute of Information Technology, Noida, India
Santoshi Sengupta, Professional Development Department, Jaypee Institute of Information Technology, Noida, India
Aayushi Gupta, Professional Development Department, Jaypee Institute of Information Technology, Noida, India

ABSTRACT

The success of software projects is quite subjective in nature and is fettered by many risks, the perception of which varies from individual to individual and largely depends on the demographic characteristics of the executives and even the characteristics of the project. This study aims to identify and gauge the software risk dimensions and analyze the differences of perception among executives toward software risks. The contributions of this study untangle the issues underlying risks in the software industry and associates these issues with the perception of the "human" factor present in the industry.

KEYWORDS: software projects; risks; demographics

Project Management Journal, Vol. 42, No. 5, 78–91

Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/pmj.20258

INTRODUCTION

The Indian software industry is characterized by the diversity of the projects it engenders. From size to complexity and from scope to criticality, the attributes of a software project can be varied. Some projects are big, encompassing numerous resources with complex integration and interorganizational developmental processes, whereas others are relatively small and simple. Some have a worldwide reach, whereas some may affect only a small division or client group (Keil, Tiwana, & Bush, 2002). Some software projects are developed "in-house," whereas others may be outsourced. Regardless of the size, complexity, criticality, or scope, the success of software projects is not guaranteed. There is strong evidence to suggest an unacceptably high rate of software project failure in the software industry. A large number of projects are abandoned before completion or are completed after comprising on quality, cost, budget, or all three (Standish Group, 2009).

There has been a wide range of studies conducted, which have addressed issues pertaining to software projects: their success, their failure, the efficiency of the team handling them, and the organizational climate in which they are developed. Various studies (Anudhe & Mathew, 2009; Bannerman, 2008; Boehm, 1989; Costa, Barros, & Travassos, 2007; Dash & Dash, 2010; Dey, Kinch, & Ogunlana, 2007; Keil, Smith, Pawlowski, & Jin, 2004; Oz & Sosik, 2000; Ropponen & Lyytinen, 2000) have been conducted in the past to unveil the reasons for the high rates of delays and failures in the software industry. Miscommunication of requirements, inaccurate estimations, low team member morale, lack of client ownership, and lack of top management support are some of the risk factors that have been mentioned by most of the researchers, thus emphasizing the technical, managerial, and people aspects. Nonetheless, it cannot be said with the utmost confidence as to what extent and magnitude these factors affect the success or failure of the software project, nor can it be stated how many of these factors genuinely affect the success or failure of the software project. The reason for this vagueness is simple: these factors are merely products of the wide variety of perceptions of the software practitioners who actually handle these factors during the software development life cycle. A different hierarchical level will produce a different perception; a different span of experience will lead to a different perception; and a different project attribute will create a different perception. Thus, it is a fact that the perception toward these factors varies widely in the experience and designation of the software practitioners, as well as the duration, team size, and value of the project. The question now is:

How does the perception toward these factors vary with various groups of the demographic and project attributes?

This study aims to evaluate the various risk dimensions across a number of demographic characteristics of software practitioners and also aims to study the variance in risk dimensions among the various project characteristics handled by software practitioners. This study is important because it helps us understand the perception of the team vis-à-vis middle management and top management.

Literature Review

Software Risk Dimensions

Numerous studies have been conducted on the identification and management of software risk factors. These studies date back to 1975, when Brooks cited the causes of project failure on the basis of his experience at IBM. Thereafter, many researchers (Anudhe & Mathew, 2009; Bannerman, 2008; Boehm, 1989; Costa et al., 2007; Dash & Dash, 2010; Dey et al., 2007; Keil et al.; 2004; Oz & Sosik, 2000; Ropponen & Lyytinen, 2000) have been working incessantly on the identification and management of software risks, some of which are discussed in this study. Dey et al. (2007), through a case study on The Town and Country Planning Office (TCPO) in Barbados, identified the unavailability of key personnel, employee turnover, and incorrect/incomplete requirement as the risks affecting software development. The study also developed an integrated framework for managing risk in software development with the involvement of the stakeholders in the TCPO. Verner, Evanco, and Cerpa (2007) conducted exploratory statistical analyses on both the software developers and top management to identify their perceptions about the determinants of project success and used logistical regression to predict project success. Accordingly, the developers' perspectives suggest that success is more likely to happen if the project manager is involved in schedule negotiations; adequate requirements information is available when the estimates are made; initial effort estimates are good; staff leave is taken into account; and more staff is not added late in the project in order to meet an aggressive schedule.

Zhou, Vasconcelos, and Nunes (2008) analyzed ten case studies in the United Kingdom, the United States, and New Zealand to identify critical risk factors (scope creep, unwillingness of the customer to accept final systems, poor project management, etc.) at the pre-implementation and implementation stages of the software project. Bannerman (2008) conducted a study in government agencies in Australia to investigate the practices of a state government when dealing with software projects. Analysis of the study uncovered ten categories of risk factors—namely, project governance, project setup, partner engagement, business proprietorship, project management, change management, management of projects, recognition of red flags, management of risk, and benefit realization. The study also suggested the various risk management techniques used in these projects. Iacovou and Nakatsu (2008) used the Delhi survey method on 57 senior IT professionals and identified lack of top management commitment, the original set of requirements miscommunicated, language barrier in project communications, lack of required technical know-how by the offshore team, and failure to consider all costs as the core risk factors. The study aimed at providing insight into the risks affecting offshore-outsourced development projects. Anudhe and Mathew (2009), using case-based methodology and structured and semistructured interviews with senior management of various Indian software companies, described various risk factors affecting the software projects. Schedule and budget management (developing a collaborative work culture with clients), client expectations (educating the client to involve a deep level of involvement with the customer), requirements capture (elaborate data collection and proactive analysis), staffing (maintaining buffer resources, involving the client in resource recruitment), and changes in the client's corporate structure (transparency and adequate communication) are some of the risk factors, and their mitigation is mentioned in their study.

Perspectives of Software Professionals

The concept that different stakeholders can perceive software projects in different ways has also been well established in the literature (Lyytinen, Mathiassen, & Ropponen, 1998). Keil et al. (2002) have demonstrated that users and project managers differ in terms of their project risk perceptions, while Warkentin, Bekkering, and Moore (2007); Warkentin, Bekkering, Johnston, and Moore (2009); and Stephen, Keil, Mathiassen, Shen, and Tiwana (2007) have exhaustively studied the perception of risk among various demographic characteristics of software professionals and have suggested that professionals with more experience in project leadership are more likely to view projects and their associated risks more holistically and assign and resolve risk as if they were organizational in nature. Agarwal and Rathod (2006), Linberg (1999), Wateridge (1995), Verner et al. (2007), and Verner, Beecham, and Cerpa (2010) have explored the success indicators as perceived by software professionals.

All of these studies provide great insight into the practitioner's perception about the software risks dimensions; however, a study about the perception among various demographic characteristics of software practitioners on these dimensions is largely anecdotal and lacking in the Indian context. Furthermore, not much research has been done in understanding the dynamics of risk among various project characteristics handled by the software practitioners.

Objectives of the Study

Based on the literature review and the gaps identified in it, the objectives of the study are as follows:

1. To identify and rank the software risk dimensions affecting software projects in India.
2. To evaluate the difference of perception toward risk dimensions on the basis of the designation and experience of software professionals.
3. To evaluate the difference of perception toward risk dimensions on the basis of project characteristics—namely, team size, duration, and total value of the project handled by Indian software professionals.

Research Methodology

A systematic and coherent approach was adopted for the research study. First and foremost, a pilot study was done on 40 software project managers working in software companies located in the National Capital Region (India) to gauge the risk factors that affected the success of their last executed projects. The project managers in the pilot study were specifically asked to identify the critical risk factors affecting the software development life cycle. Based on the perceptions of the project managers in the pilot study and in-depth secondary data analysis, an exhaustive list of 23 risk items affecting the software development life cycle was identified. These items were meticulously drawn out after thorough interviews and a comprehensive literature review; thus, the list is all-inclusive. A questionnaire was prepared using these risk items and was administered to software professionals with a minimum of four years' experience in handling software projects in India.

To carry out extensive research, a survey design was used. Four major IT hubs in India—namely, NCR (Gurgaon, Noida, Delhi, Faridabad), Hyderabad, Bangalore, and Chennai—were selected. From each IT hub, eight companies were randomly selected, thus bringing the total to 32 companies. A total of 900 questionnaires were sent out, of which 340 filled-in questionnaires were returned, with a response rate of 37.7 percent; of these, only 300 were found to be completely filled out and 40 were discarded. A random sampling technique was used to gather data from IT professionals with more than four years of experience in handling software projects. The questionnaire was intricately designed to elicit information about (1) the personal characteristics of the respondents (namely, designation and total experience); (2) project characteristics handled by the respondents (namely, team size, duration of the project, and total value of the project); and (3) the risk factors impacting the success of the last executed project. In addition to gathering data through the questionnaires, in-person interviews were also conducted with the project managers and senior management to accentuate the data collected. Statistical Package for the Social Sciences (SPSS) version 17.0 was consistently used for the statistical analyses.

	Number of Items	Cronbach's Alpha	Kaiser-Meyer-Olkin Measure of Sampling Adequacy	Bartlett's Test of Sphericity	
Software risks	23	0.956	0.916	Approx. Chi-Square	5,309.252
				df	253.00
				Sig.	0.000

Table 1: Reliability and validity of the software risk factors and organizational climate factors.

Analysis and Findings

The analysis of the data set obtained was conducted in several steps. First, the major software risk dimensions were identified using principal component analysis (PCA). PCA is widely used to examine the underlying patterns for a large number of variables and to determine if the information can be summarized in a small set of factors or components for subsequent correlation or regression. Second, ANOVA-Duncan post-hoc was conducted to test the variance in these dimensions among various demographics and project characteristics. The reliability of the survey instrument was also tested.

To test the validity of the instrument, Cronbach's alpha and Kaiser-Meyer-Olkin (KMO) tests were conducted. Cronbach's alpha was calculated to measure the internal consistency and reliability of the instrument. The results of the Cronbach's alpha, Bartlett's test of sphericity, and KMO tests are shown in Table 1.

Personal Profiles of the Respondents

The first section of the instrument gathered information about the personal profiles of the respondents, which included designation and total experience; each of these background variables is described in the sections that follow. It must be noted here that the open-ended questions were asked from the respondents regarding the demographic and project characteristics. Because these data could not be used in raw form, the data were tabulated and grouped together according to the cumulative frequencies in such a way that each group has almost equal distribution. Table 2 provides the demographic characteristics.

Designation

Because the questionnaire was deliberately administered to IT professionals with more than four years of experience handling software projects, the respondents were primarily project leads and higher-level employees. As shown in Table 2, out of 300 respondents, 141 (47 percent) were primarily project managers, senior managers, and account managers, who were specified as Level 2, whereas 43 respondents (14 percent)

Characteristics	Number	Percentage
Designation Level 1 (project leads, technical leads, consultants, senior software engineers, lead consultants)	116	38.70%
Level 2 (project managers, senior managers, account managers)	141	47%
Level 3 (chief operating officer, head of IT, director, chief executive officer)	43	14.30%
Total experience (in years) 4–9 years 10–14 years More than 14 years	 112 123 65	 37.30% 41% 21.70%

Table 2: Demographic characteristics of the respondents.

were from the top management team (chief operating officer, head of IT, director, chief executive officer), who were specified as Level 3. Such a wide scale of distribution was necessary to enable a better analysis and interpretation of the data.

Total Experience

As shown in Table 2, the respondents were classified into three categories, depending on their total experience. The second category (123 respondents [41 percent]) was dominated by project managers and senior project managers, with total experience ranging from 10 to 14 years. A few directors and vice presidents were also in this category. In the last category, with more than 14 years of experience, there were 65 (21.7 percent) respondents, mainly from the senior management team. A few senior managers and account managers fell under this category. The main reason for this blend is that the software industry is a modern industry, and consists of 25- to 30-year-old individuals who can start their own businesses and hire their own employees; therefore, it is easier to reach higher levels at an early age compared with the traditional industries (e.g., iron and steel).

Thus, as seen from the demographics, the sample was dominated by project managers and senior project managers. Furthermore, an in-depth analysis has been conducted to gauge the profiles of projects handled by the respondents.

Profile of the Last Executed Project Handled by the Respondents

The respondents were asked to provide details of the last executed project they handled. The instrument contained questions on the team size of the project, total duration of the project, and, finally, the approximate value of the project in US dollars. These details are provided in Table 3.

Team Size

The total team size was divided into three categories. As is clear from Table 3, 100 projects were handled by a team size of three to ten members, whereas 89 projects were handled by a team size of 11 to 20 members; thus, 189 (63 percent) projects handled by the respondents had a team of 20 members or less. On the higher side, only 25 projects (8.4 percent) had a team ranging from 50 to almost 500. Only one project had a total value of US $145 million, with a team size of 500 team members.

Duration

As shown in Table 3, it was found that 113 projects (37.7 percent) were completed in less than one year. After detailed conversations with several IT professionals, it was found that projects cannot really be classified as short-term, medium-term, or long-term projects because they are organization-specific. For companies with thousands of employees, a project with a one-year duration would be considered a short-term project, but for a company with only 10 employees, the same project would be considered a long-term project.

Project Details	Number	Percentage
Number of team members in the project 3–10 11–20 More than 20	 100 89 111	 33.30% 29.70% 37%
Time taken to complete the project (in months) 1–9 months 10–19 months More than 20 months	 113 96 111	 37.70% 32% 37%
Total value of the project (in million dollars) 0.02–0.70 0.71–2.00 Greater than 2.00	 102 89 109	 34% 29.70% 36.30%

Table 3: Characteristics of the projects handled by the respondents.

Value

As shown in Table 3, the projects were almost evenly distributed among the three groups, with most of the projects falling into either the first group of project value (up to US $70,000) or the third group (value above US $2 million). However, after a detailed analysis, it was found that most of the projects outsourced to Indian software companies ranged between US $1 and 10 million, with 151 projects (50.3%) falling into this category.

Thus, it can be seen from the profile of the project that the sample is a homogenous mix of team, time, and total value of the project. Further study was done on extracting the risk factors as rated by these respondents.

Identification of the Risk Dimensions

As already pointed out, the 23 risk items used for the study were identified after an all-encompassing and exhaustive secondary data analysis and pilot study. Table 4 describes the 23 risk items used for the study.

Factor analysis was done to extract the software risk factors affecting the success of the projects; principal component analysis was the method of extraction and Varimax was the rotation method. As per the Kaiser criterion, only factors with eigenvalues greater than 1 were retained. Four factors in the initial solution have eigenvalues greater than 1; together, they account for almost 68% of the variability in the original variables. The items falling under each of these factors were then dealt with quite prudently. The factors extracted for the study are shown in Table 5.

SRS Variability Risk

The software requirement specification (SRS) variability risk is the name given to the first risk dimension identified through factor analysis. The items included in this are shown in Table 5. All of these items have one commonality: lack of proper flow of information leading to requirement variability. The problem of requirement variability initiates at the stage in which the project manager does not have any prior experience in handling the similar type of project. This problem is further heightened when the manager fails to understand the client because of regional or language differences. Together, these two issues create a ripple effect and open up the sea of miscommunication of requirements, gross estimation errors, conflicting and continuous requirement changes, and inaccurate requirement analysis, thus delaying the recruitment of the team, which results in a gross miscalculation of the cost of the project. This is the initiating stage of the project, and such issues are bound to create many problems in the project, even resulting in cancellation. The same has been reiterated by Boehm (1991), Baccarini (1996), King (1994), Clancy (1995), and Smite (2006). Besides this, the lack of client ownership and lack of drive to specify requirements are major contributors to vague requirements, and not enough clarification is done to solve them in time. Researchers even state that project managers fail to make correct estimations in the initial stages of the software development and sometimes distort the facts or become too optimistic, thus creating gross estimation errors (Keil, Im, & Mahring, 2007; Snow, Keil, & Wallace, 2007). Thus, it is obvious how crucial it is to understand the requirements correctly for the success of the project.

Team Composition Risk

As is clear from Table 5, the second factor has seven items loaded on it, and all are related to team composition risk. This factor deals with the risks related to the team members responsible for the development and execution of the project. The major contributors to these risks are the lack of top management support and unavailability of a competent project manager in handling the team. Any lack of interest exhibited by top management will result in hiring of an inexperienced or highly diverse team. In addition, if top management is not keen on investing in training or hiring a subject matter expert, there will be a risk of unavailability of a domain expert, which will create problems for the project (Barki & Hartwick, 1989; Gioia, 1996; Keil et al., 1998; Leitheiser & Wetherbe, 1986; Nah et al., 2001). In addition to senior management's lack of interest, the project manager is also responsible for contributing to the team's composition risk. The project manager acts as a liaison between top management and the team. All issues related to promotion and performance appraisals are handled by the project manager. If the manager is inept at handling the team issues, there is bound to be dissatisfaction among team members, resulting in low morale, lack of commitment, and, ultimately, high turnover (Bosco, 2004; Dey et al., 2007; Faraj & Sambamurthy, 2006; Standish Group, 2009). This is a very important risk affecting the project.

Control Processes Risk

As is clear from Table 5, control processes risk arises when the project does not have a good and sound configuration management. Configuration management is the detailed recording and updating of information that describes the project's hardware and software. Such information typically includes the versions and updates that have been applied to the software product and the locations and network addresses of hardware devices (Babich, 1986; Bersoff, 1984). If the project does not have a good and systematic configuration plan, it will lead to poor documentation, because the version will not be saved, updated, or made available when needed. The problem will be further aggravated due to insufficient testing. When a team member makes changes in the code and fails to save it in the central repository, it results in a loss to the project. The reason for this is very simple, because when the changes are not communicated to the tester,

23 Risk Items	Prior Literature and Pilot Study
Working with inexperienced team	Addison and Vallabh (2002); Hoodat and Rashidi (2009); Jiang and Klein (2001); Keil, Cule, Lyytinen, and Schmidt (1998); Schmidt, Lyytinen, Keil, and Cule (2001); Reel (1999); Smith et al. (2006); pilot study
Delay in recruitment and resourcing	Pilot study
Less or no experience in similar projects	Keil et al. (1998); McFarlan (1982); McLeod and Smith (1996); pilot study
Insufficient testing	Pilot study
Team diversity	Pilot study
Lack of availability of domain expert	Barki and Hartwick (1989); Gioia (1996); Keil et al. (1998); Leitheiser and Wetherbe (1986); Nah, Lau, and Kuang (2001); pilot study
Lack of commitment from the project team	Pilot study
High level of attrition	Arora, Arunachalam, Asundi, and Fernandes (2001); Dey et al. (2007); Reel (1999); pilot study
Estimation errors	Hoodat and Rashidi (2009); Smith et al. (2006); Zhou et al. (2008)
Inaccurate requirement analysis	Addison and Vallabh (2002); Field (1997); Keil et al. (1998); Mursu, Soriyan, and Olufokunbi (1996); Schmidt et al. (2001); pilot study
Lack of top management support	Barki and Hartwick (1989); Gioia (1996); Keil et al. (1998); Leitheiser and Wetherbe (1986); Nah et al. (2001)
Low morale of the team	Pilot study
Miscommunication of requirements	Iacovou and Nakatsu (2008); Shull, Rus, and Basili (2000); Zhou et al. (2008)
Conflicting and continuous requirement changes	Addison and Vallabh (2002); Boehm (1989); Clancy (1995); Hoodat and Rashidi (2009); Jones (1993); King (1994)
Language and regional differences with client	Anudhe and Mathew (2009); Arora et al. (2001)
Lack of client ownership and responsibility	Keil et al. (1998); Mursu et al. (1996)
Inadequate measurement tools for reliability	Pan (1999)
Third-party dependencies	Krasner (1998)
Inability to meet specifications	Pilot study
Inaccurate cost measurement	Galorath and Evans (2006); Masticola (2007)
Poor code and maintenance procedures	Pilot study
Poor documentation	de Souza, Anquetil, and de Oliveira (2007); Huang and Trauth (2007); Mattsson (2008); Oza et al. (2004)
Poor configuration control	Aharon and Ilana (2010); Sabherwal, Sein, and Marakas (2003)

Table 4: Risk items used for the study.

they will not be able to test the code, thus resulting in insufficient testing. A poor and mediocre code with inadequate maintenance procedures is also caused by improper configuration control. The importance of configuration control and documentation has also been mentioned by Huang and Trauth (2007); Oza, Hall, Rainer, and Grey (2004); Sabherwal, Sein, and Marakas (2003); and Jannson (2007).

Dependability Risk

Dependability risk is the name given to the fourth and last factor; the items falling under this risk are shown in Table 5. It is extremely vital for a software project to be dependable and reliable. Software dependability is defined as the ability to avoid service failures that are more frequent and more severe than acceptable (Avizienis, Laprie, Randell, & Landwehr, 2004). Dependability is a broad term, which includes

Factor	Item	Factor Loading	Factor Name
1	Conflicting and continuous requirement changes	0.789	SRS Variability Risk
	Inaccurate requirement analysis	0.786	
	Miscommunication of requirements	0.765	
	Estimation errors	0.687	
	Less or no experience in similar projects	0.646	
	Inaccurate cost measurement	0.635	
	Language and regional differences with client	0.612	
	Delays in recruitment and resourcing	0.604	
	Lack of client ownership and responsibility	0.533	
2	Lack of availability of domain expert	0.761	Team Composition Risk
	Working with inexperienced team	0.651	
	Team diversity	0.650	
	Lack of commitment from the project team	0.628	
	Low morale of the team	0.613	
	High level of attrition	0.596	
	Lack of top management support	0.561	
3	Poor documentation	0.733	Control Processes Risk
	Poor code and maintenance procedures	0.716	
	Insufficient testing	0.556	
	Poor configuration control	0.529	
4	Third-party dependencies	0.803	Dependability Risk
	Inability to meet specifications	0.598	
	Inadequate measurement tools for reliability	0.578	

Table 5: Factor analysis of the risk dimensions.

availability, reliability, safety, integrity, and maintainability of the software (Avizienis et al., 2004). Dependability of the software is therefore very crucial for the success of a project. For the successful completion of a project, all components (hardware and software) must be available at the right time and in the right place. To make things easier, sometimes part of the project is outsourced to a third-party vendor, which can be a great advantage if the right vendor is chosen. On the other hand, if an unreliable vendor is chosen, there can be many problems, such as poor and undependable components, late delivery of critical components, and so forth. This will create a disastrous effect on the project and prevent the project team from meeting the customer or client's specifications. Furthermore, any component from the third-party vendor cannot really be tested in isolation and, if proved unreliable, can completely thwart the entire project. This finding about dependability and reliability is supported by numerous studies and also conforms to studies by Sherer (1995), Pan (1999), Krasner (1998), and Ropponen and Lyytinen (2000).

Ranking of Software Risk Dimensions

For determining the most important risk affecting the software projects in India, means and standard deviations of the risk dimensions were calculated. The respondents were asked to rate the effect of each risk based on the success of their last executed project, on a scale of 1 to 5, in which 5 was too much effect and 1 was no effect at all. After the factor analysis, when four factors emerged, the score of each of the factors was computed by taking out the mean of the items falling under each factor. For example, in order to calculate the mean of dependability, the scores of all the items (i.e., third-party dependencies, inability to meet specifications, and inadequate measurement tools for reliability) were first added, and then the mean was calculated. Similarly, means and standard deviations were calculated for all the factors. The ranking of the dimensions, based on the means and standard deviations, is shown in Table 6.

It is clear from Table 6 that SRS variability has the highest mean of 3.06, stating that most of the respondents consider software requirement specification variability as the most important risk affecting software projects. The standard deviation for SRS variability risk is 1.06. The SRS variability risk is closely followed by dependability risk, with a mean of 2.84; team composition risk, with a mean of 2.75; and, finally, control processes risk, with a mean of 2.52, thus, clearly proving that SRS variability risk is the most important risk affecting Indian software projects, which is also in line with the findings of Sharma, Gupta, and Khilnani (2009). Although SRS variability risk is perceived to be the most crucial risk, is the perception the same across all the groups of designation and experience? This is explored in the following section.

S. No.	Factor Name	Mean	Standard Deviation
1	SRS Variability Risk	3.06	1.06
2	Dependability Risk	2.84	1.13
3	Team Composition Risk	2.75	1.03
4	Control Processes Risk	2.52	1.05

Table 6: Means and standard deviations of the risk factors.

Comparison of Risk Factors Across Various Personal and Project Characteristics

The dimensions of software risk so formulated after the factor analysis were then compared with the various personal characteristics of the respondents and the project characteristics handled by the respondents chosen for the study. The personal characteristics included experience and designation, whereas project characteristics included total team size, total time taken to complete the project, and the total value of the project. The comparisons are discussed in the following section.

Designation

Duncan's Mean Test was applied to compare the dimensions of software risk among the three designation groups of the respondents. All risk dimensions—namely, SRS variability, team composition, control processes, and dependability—showed significant differences in mean and standard deviation values. Table 7 shows all the values of mean and standard deviations of the dimensions of software risk across the various designation groups.

As shown in Table 7, the F-value is highest in cases of SRS variability. This factor has been ranked the highest by respondents of level 1 (project leads, technical leads, consultants, and analysts), with a mean of 3.58 and a standard deviation of 0.87, which implies that level 1 respondents perceive these risks as having a high effect on the success of the project. Dependability and team composition risks, with means of 3.26 and 3.19, respectively, are again considered more significant risks by level 1 respondents than the other two groups, which are dominated by project managers (level 2) and directors (level 3). This is due to the lack of sufficient experience, expertise, and authority in handling and mitigating risks. Another interesting fact that emerged out of the analysis was that the difference was significant only with level 1. It should be noted here that there were no significant differences between level 2 and level 3 respondents, thus testifying that these two levels have an almost similar opinion. When compared with level 1 employees, both these groups showed significant difference. Hence, it can be said with statistical confidence that there exists a difference in perception of these risks among the various designation groups, which conforms to many other previous studies. Stephen et al. (2007) testified that IT project managers with more experience have risk perceptions that differ from those of more junior managers. Warkentin et al. (2009) has also concluded that, instead of viewing risks as separate or discrete categories, managers at higher levels, owing to their more comprehensive organizational perspectives, are more likely to consider risks essentially organizational in nature compared with their junior managers. The same has been reiterated by Smith, Eastcroft, Mahmood, and Rode (2006).

Total Experience

Duncan's Mean Test was applied to compare the dimensions of software risk among three groups and formed on the basis of total experience. Significant differences were found in the mean values of all of the dimensions of risk. Table 8 shows all the values of mean and standard deviations of the dimensions of risk across the various experience groups. As shown, the F-value was highest in the case of SRS variability, followed by team composition, control processes, and dependability. It should

	D1 ($N = 116$)		D2 ($N = 141$)		D3 ($N = 43$)					
Risk Factors	Mean	*SD*	Mean	*SD*	Mean	*SD*	D1 v/s D2	D1 v/s D3	D2 v/s D3	*F*-value
SRS Variability	3.58	0.87	2.72	1.04	2.74	1.01	*	*	–	27.05**
Team Composition	3.19	0.98	2.41	0.94	2.67	1.05	*	*	–	20.89**
Control Processes	2.97	0.95	2.19	0.98	2.32	1.12	*	*	–	20.47**
Dependability	3.26	1.18	2.62	1.08	2.42	0.82	*	*	–	14.60**

Note. D1 = level 1, D2 = level 2, D3 = level 3; Duncan's Mean Test.
*Significant at 0.05 level. ** Significant at 0.01 level.

Table 7: Comparisons of risk factors among three designation groups.

Risk Factors	E1 (N = 112) Mean	SD	E2 (N = 123) Mean	SD	E3 (N = 65) Mean	SD	E1 v/s E2	E1 v/s E3	E2 v/s E3	F-value
SRS Variability	3.51	0.93	2.82	1.07	2.74	1.01	*	*	–	18.17**
Team Composition	3.14	0.99	2.52	0.97	2.51	1.02	*	*	–	13.61**
Control Processes	2.91	1.02	2.31	0.89	2.23	1.09	*	*	–	13.91**
Dependability	3.19	1.17	2.74	1.09	2.44	0.98	*	*	–	10.42**

Note. E1 = up to 9 years, E2 = 10–14, E3 = more than 14; Duncan's Mean Test.
*Significant at 0.05 level. ** Significant at 0.01 level.

Table 8: Comparisons of risk factors among three experience groups.

be noted again that the difference was significant only between two groups (i.e., between E1 [up to 9 years of experience] and E2 [10 to 14 years of experience] and E1 and E3 [more than 14 years of experience]). E2 and E3 had no significant differences between them as far as the perceptions of these four risk factors were concerned.

All four risks were ranked highest by E1 respondents, followed by E2, and then E3. This is not surprising, because employees with less experience are not as well versed in managing such issues or even mitigating them, compared with veterans in the industry for whom such issues do not emerge as risks but as minor challenges that need to be faced and dealt with. Respondents in the E2 and E3 categories, therefore, have similar opinions about such risks, and so there is no significant difference between the two. This finding also has congruence with earlier studies, such as those by Bourne and Walker (2004), Smith et al. (2006), and Warkentin et al. (2009), in which it was concluded that employees with more experience in project leadership were more likely to view projects and their associated risks more holistically and assign and resolve risks as if they were organizational in nature.

Project Characteristics

After comparing the dimensions of software risks with the various personal characteristics, the same dimensions were compared with the various project characteristics. The project characteristics included total team size, total time taken to complete the project, and the total value of the project. The comparisons are discussed in the sections that follow.

Total Team Size

Size refers to the magnitude of the resources needed to complete the project (Thompson, Iacovou, & Smith, 2009). According to this definition, human resources engaged in a project make up the team size. Past research also illustrates that the level of resources correlates with the complexity of the development (i.e., project-related risks [Baccarini, 1996; Francalanci, 2001; Otto & Magee, 2006]). The team size of the projects is an important variable and is associated with the risk dimensions. In this study, team size has been divided into three categories—namely, T1 (up to 10 members), T2 (11 to 20 members), and T3 (more than 20 members). As shown in Table 9, Duncan's Mean Test was conducted to find the significant differences among the means of these three categories. The finding shows that none of the *F*-values was significant; thus, it cannot be said with statistical confidence that the risk dimensions vary with the team size.

Total Duration

The total time taken for the completion of a project is an important attribute and is associated with risks, because it is an extensive resource for any project (Warkentin et al., 2009). The total duration of a project was categorized under three heads—namely, TT1 (up to 9 months), TT2 (10 to 19 months), and TT3 (more than 19 months). The risk factors were, thus, compared with these three categories using Duncan's Mean Test. Only team composition had significant differences among the three categories, with an F-value of 3.1201 (Table 10). None of the other risks had any significant differences among the three groups.

Team composition had a significant difference only between the TT2 and TT3 categories (i.e., between projects with durations of 10 to 19 months and projects with durations of more than 19 months). Duncan's Mean Test shows that there is a difference in mean values of risk between these two categories. Projects with longer durations have a higher mean compared with projects with shorter durations and this is because, as the length of time increases, the levels of employee morale and motivation tend to diminish, because such projects are generally viewed as software maintenance projects. With low or almost no challenge in the type of work, along with high attrition, employees lack the commitment necessary for the project, and thus the team composition emerges as a significant risk for projects with longer durations (Morris, 2004; Schmidt et al., 2001). Ropponen and Lyytinen (2000)

Risk Factors	T1 ($N = 100$) Mean	T1 SD	T2 ($N = 89$) Mean	T2 SD	T3 ($N = 111$) Mean	T3 SD	T1 v/s T2	T1 v/s T3	T2 v/s T3	F-value
SRS Variability	2.99	1.11	3.11	1.03	3.08	1.03	–	–	–	0.3304
Team Composition	2.71	1.13	2.74	1.01	2.79	0.96	–	–	–	0.1772
Control Processes	2.64	1.13	2.43	0.84	2.47	1.13	–	–	–	1.1641
Dependability	2.77	1.22	2.89	1.06	2.85	1.10	–	–	–	0.2921

Note. T1 = up to 10, T2 = 11–20, T3 = more than 20; Duncan's Mean Test.

Table 9: Comparisons of risk factors among three team size groups.

Risk Factors	TT1 ($N = 113$) Mean	TT1 SD	TT2 ($N = 96$) Mean	TT2 SD	TT3 ($N = 111$) Mean	TT3 SD	TT1 v/s TT2	TT1 v/s TT3	TT2 v/s TT3	F-value
SRS Variability	2.99	1.13	2.99	1.06	3.18	0.98	-	-	-	1.1998
Team Composition	2.70	1.12	2.56	1.00	2.93	0.94	-	-	*	3.1201*
Control Processes	2.47	1.10	2.46	1.06	2.60	0.99	-	-	-	0.5424
Dependability	2.76	1.15	2.75	1.19	2.98	1.07	-	-	-	1.4238

Note. TT1 = up to 9 months, TT2 = 10–19 months, TT3 = more than 19; Duncan's Mean Test.
*Significant at 0.05 level.

Table 10: Comparisons of risk factors among three total time groups.

and Warkentin et al. (2009) have pointed out that considering the time issue of a project, the team relationships have to be managed. As quoted in Warkentin et al. (2009), "Ultimately you need effective communication channels with your vendors and technology partners. Mutual respect and understanding play a large role in the relationship" (p. 14). This clearly states that team composition is associated with the duration of a project and that it has a larger impact on projects with longer durations as compared with shorter ones.

Total Monetary Value

Money is a critical resource that should be allocated and monitored for the success of software and information systems development projects (Laudise & Nuara, 2002; Warkentin et al., 2009). The total dollar value thus becomes an important attribute for any project, and it has been selected for comparing the risk factors. The total dollar values of projects in which the respondents were involved are divided into three categories—namely, V1 (up to US $0.70 million), V2 (US $0.71–2.00 million), and V3 (more than $2.00 million). As shown in Table 11, Duncan's Mean Test was applied to see if there were any differences in the mean values of the risk factors among the three categories of dollar values associated with the last executed projects; it was determined that none of the differences was significant. Thus, it cannot be said with statistical confidence that there exists a difference in the mean value of the risk factors across the three categories of project value.

Thus, it is clear how each of the software risk dimensions is viewed by the various stakeholders within the software development organization. Rather than viewing many software risks as separate or discrete categories, senior project managers, with their more comprehensive organizational perspectives, view all risks as essentially organizational in nature, compared with the perspectives of the project leads and technical leads. As far as the variance in risk dimensions among the project characteristics is concerned, the analysis did not show significant variance among most of the project characteristics. The team composition risk, however, did show a significant variance among the large-size projects, which is also in line with the findings of Ropponen and Lyytinen (2000).

Conclusion, Limitations, and Implications

To conclude, it can be said that based on the perceptions of software executives—ranging from middle to top level—primarily four kinds of risk dimensions prevail in the Indian software industry

Risk Factors	V1 (N = 102) Mean	V1 SD	V2 (N = 89) Mean	V2 SD	V3 (N = 109)\ Mean	V3 SD	V1 v/s V2	V1 v/s V3	V2 v/s V3	F-value
SRS Variability	2.93	1.17	3.12	1.06	3.13	0.94	-	-	-	1.1468
Team Composition	2.66	1.14	2.79	1.01	2.80	0.94	-	-	-	0.6530
Control Processes	2.49	1.09	2.54	1.07	2.51	1.00	-	-	-	0.0640
Dependability	2.72	1.23	2.87	1.16	2.93	1.00	-	-	-	0.9201

Note. V1 = up to US $0.70 million, V2 = US $0.71–2.00 million, V3 = more than US $2.00 million; Duncan's Mean Test.

Table 11: Comparisons of risk factors among three value groups.

that have an impact on the success of software projects. These four categories of software risks do affect the success of software projects in India. These risk dimensions are SRS variability risk, team composition risk, control processes risk, and dependability risk. These risks have also been ranked on the basis of their mean values, and it has been determined that SRS variability risk has the maximum effect on the success of the software projects. On further exploration, it has been found that there is a difference in the perceptions of software executives (based on their personal attributes and project attributes) toward these risk dimensions, wherein senior managers perceive these risks to be more controllable and having less impact on the success of the project compared with the perspectives of junior managers.

The analysis of risk dimensions and perceptions of individuals regarding these dimensions is extremely helpful for both practitioners and academicians because it helps in disentangling the perceptions about risk factors at various levels of demographic characteristics and also explains the variances among various groups of project characteristics. This will enable organizations to accurately gauge the software risk dimensions based on the designation and experience of employees and the duration, team size, and value of projects.

This study illuminates the existing theoretical foundations regarding software risks by exploring the detailed and ample responses provided by the various groups of experienced system developers and IT professionals. Although the respondents provided rich and fertile data for the examination of software risks, future studies should increase the sample size to enable quantitative assessments, in addition to the qualitative assessments provided here. In addition to the limitations posed by the number of study participants, future research should pursue the involvement of a greater number of organizations. Also, only software risks have been taken into consideration; these software risks include mainly technical and team-related issues. There are many more risks that are external to projects but have a deep impact on project success. Such risks are environmental, cultural, political, and even organization-specific. This study was based on a generic pattern of the software industry in India. More authentic results can emerge if organization-specific factors are chosen for the study. These constraints serve as limitations to the overall generalization of this study. Further research should be undertaken to find out ways to amalgamate the perceptions of less experienced, lower-level personnel with those of senior, upper-level IT professionals, and to study how these different views are managed to promote a more unified and consistent approach to risk management. The findings of the exploratory studies provide normative guidance to both junior and senior developers alike. These findings provide rich and robust evidence in support of the theories developed and advanced by other recent studies; that is, the identification of risks and intensity of risks as governed by the demographic and project characteristics. Now that it has been determined that magnitude and intensity of the impact of risks on software projects are largely matters of perception, further investigations can be conducted to determine the correlation of demographic characteristics and project characteristics with the intensity of the risk dimensions.

References

Addison, T., & Vallabh, S. (2002). Controlling software project risks: An empirical study of methods used by experienced project managers. *Proceedings of SAICSIT 2002*, Port Elizabeth (pp. 128–140).

Agarwal, N., & Rathod, U. (2006). Defining success for software projects: An exploratory revelation. *Journal of Systems and Software, 24,* 358–370.

Aharon, Y., & Ilana, L. (2010). Changing students perception regarding software documentation. *MCIS 2010 Proceedings.*

Anudhe, M. D., & Mathew, S. K. (2009). Risks in offshore IT outsourcing: A service provider perspective. *European Management Journal, 27,* 418–428.

Arora, A., Arunachalam, V.S., Asundi, J., & Fernandes, R. (2001). The Indian software services industry. *Research Policy, 30*(8), 1267–1287.

Avizienis, A., Laprie, J. C., Randell, B., & Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing, 1,* 11–33.

Babich, W. (1986). *Software configuration management.* Reading, MA: Addison-Wesley.

Baccarini, D. (1996). The concept of project complexity: A review. *International Journal of Project Management, 14*(4), 201–204.

Bannerman, P. L. (2008). Risk and risk management in software projects: A reassessment. *Journal of Systems and Software, 81,* 2118–2133.

Barki, H., & Hartwick, J. (1989). Rethinking the concept of user involvement. *MIS Quarterly, 13*(1), 53–63.

Bersoff, E. (1984). Elements of software configuration management. *IEEE Transactions of Software Engineering, 10,* 77–87.

Boehm, B. W. (Ed.). (1989). *Organizational climate and culture.* San Francisco, CA: Jossey-Bass.

Boehm, B. W. (1991). Software risk management principles and practices. *IEEE Software, 8*(1), 32–41.

Bosco, G. (2004). *Implicit theories of good leadership in the open-source community.* Retrieved from http://opensource.mit.edu/papers/bosco.pdf

Bourne, L., & Walker, D. H. T. (2004). Advancing project management in learning organizations. *The Learning Organization, 11*(1), 226–243.

Clancy, T. (1995). *Chaos: IT development projects.* Retrieved from http://www.standishgroup.com/msie.htm

Costa, H. R., Barros, M. D. O., & Travassos, G. H. (2007). Evaluating software project portfolio risks. *Journal of Systems and Software, 80,* 16–31.

Dash, R., & Dash, R. (2010). Risk assessment techniques for software development. *European Journal of Scientific Research, 42*(4), 615–622.

de Souza, S. C., Anquetil, N., & de Oliveira, K. M. (2007). Which documentation for software maintenance? *Journal of the Brazilian Computer Society, 13*(2), 31–44.

Dey, P. K., Kinch, J., & Ogunlana, S. O. (2007). Managing risk in software development projects: A case study. *Industrial Management and Data Systems, 107*(2), 284–303.

Faraj, S., & Sambamurthy, V. (2006). Leadership of information systems development projects. *IEEE Transactions of Engineering Management, 53*(2), 238–249.

Field, T. (1997). When BAD things happen to GOOD projects. *CIO,* pp. 55–62.

Francalanci, C. (2001). Predicting the implementation effort of ERP projects: Empirical evidence on SAP/R3. *Journal of Information Technology, 16,* 33–48.

Galorath, D. D., & Evans, M. W. (2006). *Software sizing estimation and risk management.* Auerbach Publications.

Gioia, J. (1996, November). Twelve reasons why programs fail. *PM Network,* pp. 16–20.

Hoodat, H., & Rashidi, H. (2009). Classification and analysis of risks in software engineering. *World Academy of Science, Engineering and Technology, 56,* 446–452.

Huang, H., & Trauth, E. (2007, April). Cultural influences and globally distributed information systems development: Experiences from Chinese IT professionals. *Proceedings of ACM SIGMIS-CPR Conference, St. Louis, Missouri, USA,* pp. 36–45.

Iacovou, C. L., & Nakatsu, R. (2008). A risk profile of offshore-outsourced development projects. *Communications of the ACM, 51*(6), 89–94.

Jannson, A. S. (2007, November/December). *Software maintenance and process improvement by CMMI.* Retrieved from http://www.utn.uu.se/sts/cms/filarea/0801_ann-sofie_jansson.pdf

Jiang, J., & Klein, G. (2001). Software project risks and development focus. *Project Management Journal, 32*(1) 20–26.

Jones, C. (1993). *Assessment and control of software risks.* Englewood Cliffs, NJ: Prentice-Hall.

Keil, M., Cule, P., Lyytinen, K., & Schmidt, R. (1998). A framework for identifying software project risks. *Communications of the ACM, 41*(11), 76–83.

Keil, M., Im, G. P., & Mahring, M. (2007). Reporting bad news on software projects: The effects of culturally constituted views of face-saving. *Information Systems Journal, 1*(791), 59–87.

Keil, M., Smith, H. J., Pawlowski, S., & Jin, L. (2004). Why didn't somebody tell me? Climate, information asymmetry and bad news about troubled projects. *The DATA BASE for Advances in Information Systems, 35*(2), 65–84.

Keil, M., Tiwana, A., & Bush, A. (2002). Reconciling user and project manager perceptions of IT project risk: A Delphi study. *Information Systems Journal, 12*(2), 103–119.

King, J. (1994). Sketchy plans, politics stall software development. *Computer World, 29*(24), 81.

Krasner, H. (1998). Looking over the legal edge of unsuccessful software projects. *Cutter IT Journal, 11*(3), 11–22.

Laudise, T. M., & Nuara, L. T. (2002). How to contract for a successful e-commerce development project: Beating the odds. *The Business Lawyer, 58,* 299–316.

Leitheiser, R. L., & Wetherbe, J. C. (1986). Service support levels: An organized approach to end-user computing. *MIS Quarterly, 10,* 336–350.

Linberg, K. R. (1999). Software developer perceptions about software project failure: A case study. *Journal of Systems and Software, 49,* 177–192.

Lyytinen, K., Mathiassen, L., & Ropponen, J. (1998). Attention shaping and software risk: A categorical analysis of four classical risk management approaches. *Information Systems Research, 9*(3), 233–255.

Masticola S. P. (2007). *A simple estimate of the cost of software project failures and the breakeven effectiveness of project risk management.* Presented at the First International Workshop on the Economics of Software and Computation, IEEE.

Mattsson, K. M. (2008, October). Problems in agile trenches. *Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, Kaiserslautern, Germany.

McFarlan, W. (1982). Portfolio approach to information systems. *Journal of Systems Management, 33*(1), 12–19.

McLeod, G., & Smith, D. (1996). *Managing IT projects.* Boyd and Fraser Publishing.

Morris, P. (2004, November). *Metrics based project governance.* Key note address from the IWSM Software Measurement Conference, Berlin, Germany. Retrieved from http://compaid.com/caiinternet/ezine/morris-metrics.pdf

Mursu, A., Soriyan, H.A., & Olufokunbi, K.C., (1996). *Toward successful ISD in developing countries: First results from a Nigerian risk study using the delphi method.* Retrieved from http://iris22.it.jyu.fi/iris22/pub

Nah, F., Lau, J., & Kuang, J. (2001). Critical factors for successful implementation of enterprise systems. *Business Process Management Journal, 7*(3), 285–296.

Otto, H. K., & Magee, C. L. (2006). Estimating factors affecting project task size in product development: An empirical study. *IEEE Transactions on Engineering Management, 53*(1), 86–94.

Oz, E., & Sosik, J. (2000). Why information systems projects are abandoned: A leadership and communication theory and exploratory study. *Journal of Computer Information Systems, 41*(1), 66–78.

Oza, N., Hall, T., Rainer, A., & Grey, S. (2004). Critical factors in software outsourcing: A pilot study. *Proceedings of the 2004 ACM Workshop on Interdisciplinary Software Engineering Research* (pp. 67–71).

Pan, J. (1999). *Software reliability.* Carnegie Mellon University. Retrieved from http://www.ece.cmu.edu/~koopman/des_s99/sw_reliability

Reel, J. S. (1999, May–June). Critical success factors in Software Projects. *IEEE Software*, pp. 18–23.

Ropponen, J., & Lyytinen, K. (2000). Components of software development risk: What influences it—A project manager survey. *IEEE Transactions on Software Engineering, 26*(2), 98–112.

Sabherwal, R., Sein, M., & Marakas, G. (2003). Escalating commitment to information systems projects: Findings from two simulated experiments. *Information & Management, 40*(8), 781–798.

Schmidt, R., Lyytinen, K., Keil, M., & Cule, P. (2001). Identifying software project risks: An international Delphi study. *Journal of Management Information Systems, 17*(4), 5–36.

Sharma, A., Gupta, A., & Khilnani, D. (2009). Identification and correlation of software project risks for excelling in IT industry. *FOCUS—The International Journal of Management Digest, 5*(1&2), 42–50.

Sherer, H. (1995). *The three dimensions of software risk: Technical, organisational, and environmental.* Presented at the 28th Annual Hawaii International Conference on System Sciences.

Shull, F., Rus, I., & Basili, V. (2000). How perspective-based reading can improve requirements inspections. *Computer, 33*(7), 73–79.

Smite, D. (2006). Requirements management in distributed projects. *Journal of Universal Knowledge Management, 1*(2), 69–76.

Smith, D., Eastcroft, M., Mahmood, N., & Rode, H. (2006). Risk factors affecting software projects in South Africa. *South African Journal of Business Management, 37*(2), 55–65.

Snow, A. P., Keil, M., & Wallace, L. (2007). The effects of optimistic and pessimistic biasing on software project status reporting. *Information & Management, 44*(2), 130–141.

Standish Group. (2009). *CHAOS summary.* Boston, MA: Author.

Stephen, D., Keil, M., Mathiassen, M., Shen, Y., & Tiwana, A. (2007). Attention-shaping tools, expertise, and perceived control in IT project risk assessment. *Decision Support Systems, 43*(1), 269–283.

Thompson, R., Iacovou, C., & Smith, H. J. (2009). Biased project status reports: A survey of IS professionals. *Proceedings of the 17th European Conference on Information Systems.*

Verner, J. M., Beecham, S., & Cerpa, N. (2010). Stakeholder dissonance: Disagreements on project outcome and its impact on team motivation across three countries. *Proceedings of the 2010 Special Interest Group on Management Information Systems's 48th Annual Conference on Computer Personnel Research* (pp. 25–33).

Verner, J. M., Evanco, W. M., & Cerpa, N. (2007). State of the practice: An exploratory analysis of schedule estimation and software project success prediction. *Information and Software Technology, 49*(2), 181–193.

Warkentin, M., Bekkering, E., Johnston, A. C., & Moore, R. S. (2009). Analysis of systems development project risks: An integrative framework. *The DATA BASE for Advances* in *Information Systems, 40(2),* 8–26.

Warkentin, M., Bekkering, E., & Moore, J. R. (2007). Risk perspectives in systems development. *Proceedings of the Americas Conference on Information Systems (AMCIS 2007).*

Wateridge, J. (1995). IT projects: A basis for success. *International Journal of Project Management, 13*(3), 169–172.

Zhou, L., Vasconcelos, A., & Nunes, M. (2008). Sorting decision making in risk management through an evidence-based information systems project risk checklist. *Information Management and Computer Security, 16*(2), 166–186.

Arpita Sharma holds a PhD in management and is a committed academician and devoted researcher. She has extensive experience in teaching undergraduate and postgraduate students. She started her career as a faculty member in the Professional Development Department with the Jaypee Institute of Information Technology (JIIT) in Noida, India, and then also spent some time with the Jaipuria Institute of Management in Noida. She has published around five research papers in international and national journals of repute. She has presented a number of papers in various conferences. Her areas of interest are project management, financial management, and presentation and communication skills. She is a highly motivated, innovative, and results-oriented educator, seeking to combine academic training in a learning-centric approach to make a solid contribution to organizational goals in higher education.

Santoshi Sengupta holds a PhD in management and has been working with the Jaypee Institute of Information Technology in Noida, for the past 6 years. She started her career in January 2005 in the faculty of the professional development department at JIIT. She has to her credit around 10 research publications in international and national journals of repute and also as chapters in books. She has presented a number of papers in international conferences both in India and abroad. She teaches organization behavior, total quality management, presentation and communication skills, English, and group and cooperative processes successfully to B.Tech students. She endeavors to augment herself as a committed researcher and proficient teacher to impart the best to her students. She is a highly motivated, inventive, and results-oriented academician who seeks to provide academic training using a learning-centric approach.

Aayushi Gupta, associate professor and head of the department of professional development, holds a PhD in management. She has more than 10 years of teaching experience. She has guided a number of PhD scholars under her and has nine research publications in international and national journals of repute and also has authored chapters in various books. She has presented a number of papers in international conferences both in India and abroad. She is a committed researcher and believes in quality work.