CHAPTER - 4

HUMAN RESOURCE ALLOCATION FOR SECURE SOFTWARE DEVELOPMENT

4.1 INTRODUCTION

Nowadays, as the modern society becomes increasingly dependent on software, the cost and effect of software failures could be destructive. Demand of software capabilities in both private and public organizations insists there is a need to provide trustworthy software. Developing software systems have enormous significance that can be exactly trusted. In software development processes, risk management is an important one. The management of any organization aims to achieve its objective by monitoring and reducing risks, whether working in the public sector or private sector.

Risk could significantly increase or impede the ability of an entity to achieve its present or future business objectives. Risk management is a main concept associated with the safety and financial integrity of an organization, its strategic development and risk assessment. The risks must be identified, analysed, monitored and managed so that they are maintained in particular limit, accepted by the entity’s management. Risk management process is still in progress and the results are generated in the aspect of accepting, eliminating or reducing risks that affect the achievement of objectives. The aim is to optimize the entity’s exposure to risk in order to avoid threats, prevent losses and utilize opportunities.

Integrated Risk Management (IRM) deals with the risks associated with different kind of levels in the organization such as strategy, tactics and covering both threat and opportunity. The effective implementation of the IRM can produce a number of advantages to the organization which are not present in the typical limited scope risk process. Nowadays risk management process is increasingly appreciated and recognized in the aspect of theory and practice which means, increasing the number of specialists in the field, and increasing the interest of managers within organizations to design and implement effective risk management systems to meet the objectives. IRM system
operates with wide categories of risk, such as personnel risk, financial risk, legal risk, etc., with different risk connected to various activities, risk related to various operations or transaction. Furthermore, external risks may affect the development of the overall entity or creating one or more activity executed within the organization.

The Capability Maturity Model Integration (CMMI) is defined as the process model which concludes a clear explanation of what an organization should do to enhance the performance. It consists of five “Maturity Levels” or three “Capability Levels,” the CMMI enumerates the most significant key factors that are required to produce good products, or provide proper services, and wraps them all up in a comprehensive model. The CMMI also helps us to detect and achieve measurable business goals, build better products, keep customers happier, and ensure that workers are working as efficiently as possible.

The purpose of this work is to implement risk recognition and risk reduction strategies in global software development (GSD) [86]. GSD systematic literature reviews (SLR) research focuses on finding what research has been conducted in the area and to deduce if the SLRs furnish appropriate risk and risk reduction advice to give guidance to organizations involved with GSD. A trustworthy software security requirements behaviour model can be used to make web services and applications secure [87]. Authors of [88], [89], [90] defined trustworthiness in the aspect of software industry, and also summarized some suggestions about trustworthy software research methods. Developing trustworthy software is a multifaceted problem of security engineering, software engineering and risk management. Thus, making secure software starts with predominant software engineering practices, increased with sound technical practices, and supported by risk management practices that boost trustworthy software development [91]. In order to develop trustworthy software, a structured process risk control is required. This study demonstrated that a significant relationship existed between risk and project success [92]. It also investigates an adaptive system of integration technical risk assessment model, which is based on Bayesian belief networks coupled with parametric models [93], [94]. This model gives statistical information for decision makers, enhancing risk management of complex projects.
4.2 HUMAN RISK IN SOFTWARE DEVELOPMENT

Risk in software development project is defined as the product of the risk factor surrounding a software development project and the impact associated with failure. The factors which threaten the success of software development are the cause of emerging uncertainty surrounding a software development project. These factors are known as “risk factors” which affects the effective completion of a development project.

Various aspects of human being involvement and its impact in a software development process are termed as human factors. The human factor is the element which is responsible for the effective team formation in the development process. People have many choices to set targets in life by considering different options what they have. These choices will be influenced by the psychological factors. This will be useful for modelling the human factors in risk management.

4.3 ESSENTIAL OF HUMAN RISK MANAGEMENT

Minimizing project risks is important in software development process. Investigation on risk management in software development has especially concentrated on forming instructions for different tasks. It helps to avoid project failure due to improper schedule, budget constraints and dissatisfaction of customer expectations. Risk management handles the project in the aspect of risk identification and rectification in order to develop control measures. Rectification may not mean abandoning the task that involves risk. The tasks which are rated as high risk tasks are essential to register the uniqueness of the product over other competitive products in the market [95]. The main motivation of risk management is to identify all possible risks associated with that project, evaluate their severity, and impacts, and then to decide control actions. Minimizing unpredicted issues which are arising during the course of the project is the main idea behind risk management. Improper planning in risk control may lead to tempestuous completion of the project. Software development risk management involves in the process which is used to derive the preventative strategy in order to complete the project within the specified limit of time and money. It may lead to reduce project cost and time required to complete the project and to increase the quality of the product. Improper risk management may lead to loss of revenue and customer satisfaction which causes negative impact on the product. Once the risk factors are successfully recognized and evaluated, the next step is to control the risk.
4.4 RESOURCE FACTORS OF HUMAN RISK MANAGEMENT

Every industry is involved with software area which provides automation in real time applications in an efficient manner. So there is a need to develop trustworthy software for various fields. Software development is the process which is based on human based intellectual activity. During implementation, various issues especially issues related to appropriate representation of developers participating in the development processes may be occur.

As well-known software development is a human-centered process, the process and the performance is influenced by human factors. Effect of human factors in the development process may be customer oriented (influencing the software development market) or developer oriented i.e. (Influencing the development process) and finally having a distinguishable impact on success by being a manager. Human factors associated with the development process can be viewed from different aspects such as psychological, cognitive, management and technical. Different human factors may influence different levels of impact in the process and it may vary from organizational to individual. Even though human factors have specific importance in software development, it should not be overlooked in risk management process which may lead to improper assessment. So there is a need to identify and describe the human factors in an efficient manner to avoid the potential losses associated with the risk. In the following sections we are going to see the factors which are noticeable with respect to software development. In the following we discuss those factors which appear to be the most important with respect to software development.

A resource is any item or person required for the execution of the projects. This covers many things- from paper clips to key personnel – and it is unlikely that we would wish to itemize every resource required. Let alone draw up a schedule for their use stationery and other standard office supplies, need not normally be the concern of the project manager ensuring there is always an adequate supply is the role of the office manager. The project manager must concentrate on those resources where there is a possibility that, without planning, they might not be sufficiently available when required.

Some resources, such as a project manager, will be required for the duration of the project whereas others, such as a specific software developer, might be required for
a single activity. The former, while vital to the success of the project, does not require the same level of scheduling as the latter. Individual programmers might be committed to working on a number of projects and it will be important resources will fall into one of seven categories.

Labour, the main items in this category will be members of the development project team such as the project manager, systems analysts and software developers. Equally important will be the quality assurance team and other support staff and any employees of the client organization who might be required to undertake or participate in specific activities.

Equipment, obvious items will include workstations and other computing and office equipment. We must not forget that staff also need basic equipment such as desk and chairs.

Materials are items that are consumed, rather than equipment that is used. They are of little consequence in most software projects but can be important for some – software that is to be widely distributed might, for require supplies of disks to be specially obtained.

Space for projects that are undertaken with existing staff, space is normally readily available. If any additional staff should be needed then office space will need to be found.

Services some projects will require procurement of specialist services – development of a wide area distributed system, It requires scheduling of telecommunication service

Time is the resource that is being offset against the other primary resources – project time scales can sometimes be reduced by increasing other resources and will almost certainly be extended if they are unexpectedly reduced.

Money is a secondary resource – it is used to buy other resources and will be consumed as other resources are used. It is similar to other resources in that it is available at a cost in this case interest charges, changing the level of resources on a project overtime, particularly personnel, generally adds to the cost of a project. Recruiting staff has costs and, even where staff are transferred internally, time will be
needed for familiarization with the new project. In practice we are normally concerned with departures from the plan in four dimensions—delays in meeting target dates, shortfalls in quality, inadequate functionality, and cost going over target.

4.4.1 Assessment of Productivity

Persons involved in the development process can be represented in terms of their productivity. Productivity can be explained as the amount of output per unit of input. Time required to complete the process can be derived from the relationship between person’s productivity and time taken by the person.

Personal productivity of a person will vary according to the time. In order to fix the productivity of a person, productivity of a person over a long time period should be taken into consideration. Otherwise it leads improper assessment. The overall responsibility for ensuring satisfactory progress on a project is often the role of the project steering committee or Project Board. Day-to-day responsibility will rest with the project manager and, in all but the smallest of projects, aspects of this can be delegated to team leaders.

Individual team members usually report directly to the project manager, but in most cases team leaders will collate reports on their section’s progress and forward summaries to the project manager. These, in turn, will be incorporated into project-level reports for the steering committee and, via them or directly, progress reports for the client.

Reporting may be oral or written, formal or informal, or regular or ad hoc while effective team leader or project manager will be in touch with team members and available to discuss problem, any such informal reporting of project progress must be complemented by formal reporting procedures. The frequency with which a manager needs to receive information about progress will depend upon the size and degree of risk of the project or that part of the project under their control. Team leaders need to assess progress daily where as project managers may find weekly or monthly reporting appropriate. In general, the higher the level, the less frequent and less detailed the reporting needs to be. There are, however, strong arguments in favour of formal weekly collection of information from staff carrying out activities. Collecting data at the end of
each week ensures that information is provided while memories are still relatively fresh and provides a mechanism for individuals to review and reflect upon their progress.

**4.4.2 Human Skills, Experiences and Learning in Software Development**

Individual experiences and skills have strong influence on the personal productivity. Skill can be defined as the ability of a person to a task. Skills can be categorized based on domain and the phase of its production. Phase of production deals with the stages of development process such as designing, coding, testing. And also experience plays an important role in productivity. Experience will help to handle uncertainty arising during the development process. So experience has a significant influence in the process of solving the problems.

Communication is a vital issue in software development projects. The three major stakeholders – the customer who financially sponsors the project, the contractor who performs the system development and the eventual user of the system - have to communicate regularly during the project. Ineffective communication was the single major factor which caused most number of software failures. Communication may breakdown because of different reasons including disagreement over goals and objectives, preconceived notions and opinions often leading to prejudice, semantic differences and even misunderstandings in non verbal communications. Communication problems are more likely to occur if there is a geographic separation among members. They view that project involving multiple cities or countries are more likely to have problems. There also exist a number of management problems between the off-shore and on-site teams.

Lack of English communication skills is another major factor which affects the performance of a project team. Research conducted by India’s National Index of communication skills shows that only 10 percent of IT applicants have adequate language skills (www.itbusinessedge.com). Good communication skills, especially spoken English skills, are mandatory and important in today’s global business environment.

Organizations need to make provisions for effective intra organizational communications. There are a number of interfaces which enable the same. Documentation is an integral part of a software system. It contains the information that
is necessary to effectively and successfully develop, use and maintain a system. However in practice, the creation of appropriate documentation is largely neglected. Upon completion of the project, members are spread all over the company without adequate documentation of the essential components. This leads to loss of knowledge and experiences. Project documentation should support communication during the project and address the information needs of various people- project members, project management, project supervision from the user’s perspective, the completeness of documentation with respect to the user’s tasks is an important indicator of the quality of software development.

Teleconferencing and video conferencing systems and services are the main set of technologies developed to support group work. Unreliable telecommunication facilities such as slow internet connections or telephone lines, slow computer networks etc may seriously hamper the progress of work.

From experience, one can develop his/her skills. So experience is the basement to develop skills. This approach is also known as learning by doing. Doubling the experience productivity can be increased in an efficient manner. Learning plays an important role in developing the skill. Applying the learned tricks in real time application turns into experience. Applying the learned tricks in an efficient manner will lead to a way to develop our skills. Learning capacity is the key factor to introduce changes in the product. By learning new technologies, one can incorporate the advancement in the existing product. Integrating new technologies in the product will used to enhance the uniqueness of a product. Customer satisfaction can be provided by doing this. Learning capability of a person is directly proportional to the productivity of that person.

Progress assessment will normally be made on the basis of information collected and collated at regular intervals or when specific events occur. Wherever possible, this information will be objective and tangible – whether or not a particular report has been delivered, However, such end of activity deliverables might not occur sufficiently frequently throughout the life of the project. Here progress assessment will have to rely on the judgement of the team members who are carrying out the project activities.

“Team member skills” refer to the level of experience, knowledge and skills that software development team members have. The literature has identified a range of
skills that team members may not possess, but that can impact the development project. They include lack of experience and lack of software development knowledge which are critical for project success.

Team member experience is a crucial factor for the success of a project. It has been observed that those with a higher level of design skills and experience produce better design. The standard of work may turn out to be below the expected levels owing to lack of ability and experience of staff. This lack of know – how can extend to hardware, operating systems and other software observes that the presence of inexperienced personnel can lead to troubled software development projects.

The level of application experience possessed by the development team has also been identified as a critical risk factor in the literature. There are instances where software professionals do not possess the right skills or background to understand the business requirements or apply the right tools to model and produce the corresponding systems identifies lack of application knowledge among project team members as a major cause for project failure. When project team members do not understand the application they are developing, the likelihood of software failures increases. The quality of people in software teams is one of the most important factors in improving productivity and quality in software projects. Lack of adequate and frequent training adversely affects capabilities.

Conflicts among team members hamper the development of a software project. The reasons for conflicts may be task based, process based and relationship based. The most frequent source of conflict is usually disagreement over goals and work processes, which is often a result of communication gaps between team members. Such conflicts cause severe damage to team performance. Collaborative problem solving frequently leads to disagreements among individuals or groups which can lead to conflicts within an organization, open confrontations among software development team members in a troubled project.

The motivation level of the project team has a direct impact on the project. There are many studies linking motivation to productivity increase. A number of factors including reward structures and performance measures affect the motivational levels, staff motivation in determining whether project goals tend to be achieved satisfactory. A team where members have a less satisfactory salary package, where promotions are
conspicuous by their absence, staff motivation tends to be low. Setting deadlines and consequences for not meeting the same, is another way of motivating employee, level of motivation of staff members determines the pace of their work. Reward and recognition are considered by organizations and managers as an important element in motivating individual employees. High employee turnover, which is considered to be detrimental to organizational performance, is often the result of lack of reward for good performance and lack of opportunities for career advancement.

Another factor that has an impact on software development risk lies in the manner in which responsibility is assigned to each team member. Several sources have mentioned that if the responsibility is assigned to each team member. Several sources have mentioned that if the responsibility of each team member is not clear, the project may have problems. Projects are likely to fail if each member is unclear about her/his responsibility. One of the biggest problems in software project management has been to establish a proper accountability structure. In other words, team members are often disorganized and managed with a very vague delineation of responsibilities. Uncertainty surrounding the responsibility for various project functions led to a negative impact on the success of the project.

4.4.3 Characteristic Features of Project Manager role in Risk Management

Experience deals with long term positive influence on the process associated with the development. In the process of development there may be short term fluctuations which will lead to a serious problem. Negative emotions are the key factors which are indirectly proportional to the productivity. Variety emotions related with the human are enjoyment, hope, relief, pride, gratitude, boredom, anxiety, or disappointment. Negative emotions will lead to a negative impact on the productivity.

Emotions can be influenced by the task, Possibility of doing and other characteristics according to the situation. With respect to boredom, it may lead to reduce self-motivation, involving in various activities which in turn reduce the quality and performance of the product. Internal motivation is associated with the thinking of person about the work or project. It cannot be compensated with the extra hours working. It may lead to failure of the project in unpredicted manner. The below figure 4.1 shows the structural risk management control loop.
Figure 4.1: Risk Management Control Loop

There is a contradictory mediator in emotions. For example, anxiety may increase the exterior motivation although there is a chance of blocking the activities of a person. Contradictory mediator will produce significant impacts on the product according to the situation. In order to avoid failure due to lack of intrinsic motivation, leadership quality of the team head should be high. Leadership quality should deal with decentralizing the work among team members, increasing intrinsic motivation among team members, perfect scheduling of tasks, punctuality, appraising the activities of team members, proper communication with the members. Partiality influences the leadership style in a negative manner. Stress can be defined as the factor related to disturbing perception. The causes of stress can be workload, misunderstanding, etc. If the person is not recognized for his/her work, it will lead a person in a negative manner which reduces his/her intrinsic motivation and thus in turn become the key factor for stress.
Making informed decisions by consciously assessing what can go wrong as well as the likelihood and severity of the impact is at the heart of risk management. The need to manage risk increases with system complexity. Communication lines, the structure of authority, and lines of accountability are significant in organizing the risk-based management process. It is important to effectively communicate risks to everyone involved and to reward reporting of omissions and errors. The management structure is also important to create a sense and discipline of accountability. Obtaining the top management commitment is also a crucial factor. This can be achieved through increasing payoffs associated with successfully completing the project, creating opportunities for senior managers to publicly their support for the project, and aligning the project with other goals that are viewed as central to the organization. Management should pay attention to the organizational behavioural aspects of the project and build a supporting base for the project within the organization.

The technical background of the project manager plays a very important role in risk minimization. The project manager is to be backed by an leader, who works closely with the project development. Findings skilful, open-minded software managers is instrumental in improving the risk management capacity. In order to reduce the need for self people who initiated the project are replaced with people who will naturally have greater objectivity. Failure of the project could be avoided if the project managers adopt a broader view of project management spanning both the rational approach and the psychological approach.

4.4.4 HR policies and Training

There is extensive empirical literature investigating the relationship between HR policies and organizational performance. The relationship between HR and business results is built on a simple premise that better deployment and use of HR should correlate with higher business performance. Those organizations where the HR department is supportive and helpful, encounter fewer incidents of software failure. Job matching is an essential feature of risk management. Matching an individual to a particular job is based on the idea that personnel differ to a considerable extent in abilities relevant for the successful performance of a task. Tools which are used to match computer professionals to jobs include assessment centres and simulations. Personnel systems designed around ability profiling have been enhance organizational
productivity and translate into economic benefits for employers. When individuals are considered for a position they should be consulted before hand, lest it results in a problem. It will be easier to trace and correct errors if accountability is established for a particular task.

Attendance systems are designed to tackle employee absenteeism and track the number of hours employees work. Flexibility in working time includes a variety of arrangements for employees such as part-time work; job sharing, flexitime, fixed term contracts, subcontracting and career/employment break schemes. These arrangements have been introduced for a variety of reasons which include economic factors, improvement in productivity and competitiveness, timely completion of work etc. Flexible hours of work and employment schedules affect family and employee satisfaction. Dissatisfaction with the job, stress and poor health have been found to be outcomes of heavy work schedules. Bonuses and commissions do have a motivational effect on employees. Targets appear all the more achievable if there are a series of rewarded steps in the form of promotions and incentives.

The growing demand for software development requires increasingly productive people. For effective project management, team members need global expensive and adequate training. To prevent large variations in employee performance, effective methods of training should be adopted. Training is gaining importance owing to the magnitude of the problem of producing a sufficient number of well trained software engineers. Training of employees is regarded as one of the most important functions of efficient resources management. Adequate and frequent training increases individual capabilities. Effective training in technical aspects, project management, communication and other relevant areas can help newcomers become more effective.

Training on risk management is essential for project managers, that can be improved by increasing their awareness of risk management methods.

4.4.5 Project coordination and User coordination

The project requires a series of coordination measures. Internal integration focuses on coordinating the project team members while external integration focuses on the external agencies. Project involving new technology should rely more on internal integration tools that are designed to enhance the team’s technical competence and
operation as an integrated unit. Software project management requires different types of coordination at different stages and a major portion of the organizational design problem is choosing the particular type of coordination that matches the given uncertainty. Given the temporary nature of software projects minor slippages in the control process can have a greater adverse impact than the same slippage in a more permanent organization, that suggests an impersonal mode of project coordination for low risk projects and group mode of coordination for high risk projects, higher levels of formal planning in high risk projects when cost control is taken as a measure of performance. But when system quality is studied as the performance objective, it is seen to have positive correlation with user planning.

The risk based perspective of software project coordination also takes into account the temporary nature of software projects and teams, coordinating with suppliers and sub-contractors is also an important risk management task. In order to have the wholehearted cooperation of sub-contractors and suppliers, companies need to maintain excellent relationships and operations with them. The open-minded attitude in the interactions with suppliers will enable companies to understand their needs better and handle conflicts better.

Proper communication systems should be designed to integrate users into the development environment. Some of the strategies recommended include selection of the user as the project manager, creation of a user steering committee, frequent and in-depth meetings of this committee, a user-managed change control process, training of the user, frequency and detail of distribution of the minutes of the project meeting to key users, selection of users as team members, formal user specification approval process, progress reports prepared for corporate steering committee, giving users the responsibility for the installation of the system and letting users manage decision on key action.

4.4.6 Team Cohesion

Interpersonal relationships can be categorized into two categories. First category deals with the social relationship such as coordination between the team members, communication between the members, leadership style, and horizontal coordination [96]. This type of interpersonal relationship will be considered as official one. Second category is deals with personal relationship between the team members and this will be
considered as informal. Team cohesion will be achieved if there is job satisfaction and well group atmosphere. Good team cohesion will lead to improved productivity. Team cohesion can achieved with the help of good leadership quality. Coming to the size of the team, separation of team members should be done in a conditional manner. Because smaller team size will lead to reduction in self-motivation among team members and larger team size will lead to reduce proper communication and allocation of work. Proper communication between the team members is an important key factor to complete the project successfully.

Improper communication will lead to reduction in quality of the product. Culture is also a significant factor in risk management which influences the productivity. Persons from different countries, states, and districts may be present in a team. Cultural diversity leads to the influence of different languages in the process. Different languages may affect the system performance because of improper communication.

Team related risk items are very common in software development projects. These issues include frequent shuffling of team members, a high diversified team, employee attrition, lack of skills among members, conflicts in the team, level of staff motivation and improper definition of responsibilities.

Employee turnover is a much studied phenomenon. Team member turnover results in huge costs for the organization, in terms of both direct and indirect costs. Direct costs are incurred on account of replacement and recruitment. Indirect costs arise out of pressure on remaining staff, cost of learning, product or service quality, organizational memory and the loss of social capital. Considering the high costs associated with replacing IT staff and the value of their experience, companies need to devise mechanisms designed to keep IT staff longer. Studies in an engineering services organization revealed that employee turnover was noticed more among professionals who faced stress due to changing technologies and organization requirements. Many projects also suffer from overstaffing where project teams are staffed with large number of unnecessary people waste resources and add coordination problems. However diversity provides an environment that encourages every individual to contribute her/his own ideas. This in turn leads to higher motivation and consequently higher quality team output.
Software development relies heavily on teamwork; how to streamline this collaborative development is an essential training subject in computer and software engineering education. A team process known as the meeting flow approach, the results revealed that meeting flow significantly enhances a team’s communication and coordination and balances members’ contributions by giving mutual support and effort.

4.4.7 Requirement Management

The risk that arises out of a lack of proper user commitment and support can be minimized by establishing the right contacts and creating a “home base” in the user organization. User involvement during the definition stage of the project will also help to make users more realistic in their expectations of the outcome. The key to minimizing user requirements teaming with customers helps to reduce the overall risk to both sides. Formation of a risk team comprising of the developer, customer, systems architecture group and contract management create a forum for open discussion of risks that crossed organizational lines. Projects with relatively little structure can benefit from external integration tools that create effective links between the project team and the client’s organization. To integrate the technical and end-user perspectives, first during the requirements and the environment in which the software will operate. Second during the implementation phase, all parts of the specifications should be implemented correctly. Third, the implemented system must be validated, i.e., software developers must ensure that the implemented system represents a correct mapping of the specifications.

Another useful practice is to have the client’s representatives participate in technical reviews to assure a common understanding of client needs and avoid future surprises. Project managers should be able to create and maintain long-term relationships with users and promote user commitment to the project that the primary job of the software project manager is to structure the project to meet the ‘win’ conditions of various stakeholders.

User manuals need to be prepared with a lot of care, observes that documentation involves the integration of three basic decisions – decisions about the content of the manual, decisions about the presentation of the information in the manual, and decisions about how the effectiveness of the manual should be evaluated. Good documentation can minimize the effects of a flawed interface.
Requirements management has been identified as one of the most critical aspects in controlling technology related risks. Organizational analysis, user surveys, information hiding, task analysis, user characterization, requirement scrubbing have been discussed in this regard. Experimentation, synthesis from characteristics of the utilizing system, paying early attention to poorly defined parts and system functionality, allowing the project to be driven by the user community and not by the developers are measures suggested to bring requirement changes under control. Poor execution of requirement elicitation will almost guarantee that the final project is a complete failure. Requirement issues need to be resolved as the project progresses. Unnecessary changes or requirements should not be entertained.

The risk of unrealistic estimation of budget and cost can be minimized through software reuse and requirement scrubbing. The system should be kept as simple as possible and should be designed to cost. The project success rate increases with standardized processes and design methodologies. Multi context analysis of the features required will also reduce such risk.

4.4.8 Appropriate Methodology

Project managers need to make a series of decisions at the initial stage and during the course of a software project to ensure a high quality software product. Suitable design approaches like prototyping, evolutionary approach, modular approach, simulation modelling etc reduce the risk of lack of familiarity of the designers.

Simulation has been used to address a variety of issues in software development projects including strategic management, project planning and control and process improvement. Simulation support also reduces the development time using simulation models of the designs, we can increase the accuracy of early estimates of performance.

Practitioners view prototyping as an ideal approach for communication with users, as it provides more flexible designs, and is better for the early detection of problems. Software re-use, the use of software artefacts across multiple projects, is an important strategy for improving software development efficiency and increasing the quality of software systems. When re-use is attempted, developers usually have access to the implementation code which can be modified to match a new project’s
requirements. The major savings are realized in the detailed design, coding and testing phases. For these phases, software can be used or adapted rather than uniquely developed.

Benchmarking has been defined as “the search for and the implementation of best practice”. In the area of software development it is perceived as an assessment method. Which is concerned with the collection of quantitative data on topics such as size, effort, defects, schedules and costs. It also helps managers to identify the quantum of improvement required to be the best.

4.4.9 Project control

Project control includes comparing actual progress to planned progress and taking corrective action when performance deviates significantly from the plan. It involves collecting information about costs, schedules, and technical output such as, code, design, documentation, test plans, training materials, and procedures. Information can be gathered via meetings, interviews, walk-through, and formal technical reviews.

Quality Assurance (QA) makes sure that the product meets user requirements and that it provides the desired functionality and quality. While the whole project team should be committed to building quality into the product, it is a general practice to have a separate individual or group whose primary responsibility is quality assurance, asserted that there should be a mechanism for comparing activities performed with a standard of what should be carried out, a procedure for changing behaviour if there is a need and a feedback method mechanism.

Review meetings play a major role in project control. Their purpose is to assess progress and identify areas of deviations from the plan so that corrective action can be taken. Project review meetings provide visibility to plans and create opportunities for obtaining and enforcing commitments from the participants. Project review meetings are most effective when they are scheduled at regular time intervals and follow an established agenda. These meetings are used to resolve interpersonal conflicts as well as technical issues. Team members discover that, as they interact, they spend less time dealing with interpersonal issues at these meetings and more time on solving important technical problems.
Any significant deviation or variances from the plan require prompt attention from the project manager so that timely corrective action can be taken. Project managers should inform users of the impact of scope changes about details of project cost and schedule. Project managers should be able to distinguish between desirable and absolutely necessary functionality. The project manager must be able to identify the source of the problem. If there is a major deviation from the plan, the project manager must decide whether re-planning future activities is warranted.

Technical performance control, the process of assuring that all technical requirements are met, is normally exercised through a variety of design reviews. These reviews are usually held at major milestones (e.g. completion of requirements definition phase, design phase, or coding) but it can be held at other times during the project also. The progress towards important technical goals should be tracked through appropriate metrics during the project. The metrics provide project managers visibility of what has been achieved, and their trends offer predictions of what can be expected in the future. Software engineers use different types of software development technical review for the purpose of detecting defects in software products.

4.4.10 Being responsible for risk

One of the most common methods for identifying the presence of risk factors and risk management strategies in a particular project has been the use of checklists. These checklists present a list of all potential risks and risk management factors that might be applicable in a software development project. Identifying the risk factors in software development risk framework was a general framework applicable to any software projects. Project managers to consider the aggregate risk of the portfolio of projects. Also, they did not recognize the fact that those different project require different managerial approaches. He categorized projects based on project size (size in cost, time, staffing level or number of affected parties), Experience with technology.

The Software Engineering Institute (SEI) has contributed considerably over time to the study of project risk management and has produced a significant amount of literature with a comprehensive inventory of variables related to the assessment of software development project risk.
Boehm’s model [10] suggests a comprehensive set of steps and guidelines to manage software development risks. For each of the top 10 top risk items, Boehm developed a set of risk management techniques that “have been most successful to data in avoiding and resolving the source of risk”. The idea is that after detecting the most important risk items risk-managers can compile the associated set of risk management measures and plans. These risk management practices include Award-fee contracts, Bench marking, Contract, Design to cost, Early user’s manuals, Incremental development, Information hiding, Instrumentation, Mission analysis, Morale building, Pre-award audits, Pre-scheduling, Prototyping, Requirements scrubbing, Software Re-use, Team building, Tuning, User Surveys, Compatibility analysis, Competitive design, Cost-benefit analysis, Cross-training, Detailed multi source cost and schedule estimation, High change threshold incremental development, Inspections, Job-matching modelling, Organizational analysis, Reference checking Scenarios, Simulation, Staffing with top talent, Task analysis, Technical analysis and user characterization.

The researcher have taken a brief look at some tools which can help us to define and analyse problems, and identify and evaluate possible solutions. However, the implementation of a solution which is based on plans containing estimates, approximations and uncertainties, involves risk. The idea of risk is a particularly important one, and deserves more extended treatment. Our basic goal, and therefore our measure of success, is to develop systems which have the user-value-to-cost ratio within acceptable limits. This implies that user-value of systems must be planned and controlled at all times, and project and operation system cost must be planned and controlled at all times. The organization shall be able to succeed in spite of future shock change in any part of the environment – e.g. inflation, people turnover, new organizational forms, organizational growth, technological changes, etc.

Each of our professionals (managers and technicians) shall be selected, managed, and trained to be competent to help us reach planned goals. The goal type and planned level shall only be set by group top management, and it is our responsibility to be able to respond to these goals. All risk elements shall be explicitly identified and controlled by us at all stages of planning, design and management that is written documentation for risk control must be present in all planning. All activity shall be controlled in all critical attributes, at all stages of development that is what we mean by ‘management by objectives’.
The researcher can draw conclusions for action from what we see, at a fairly early stage, if we are willing to accept a certain risk that such conclusions may be incorrect, partially misleading, or have a calculable degree of error in them. If we make the mistake of insisting on 100% safe conclusions, then we may be committed to continuing our investigations forever, and no practical change will be implemented. We should rather concentrate on knowing and controlling the degree of risk involved. We should let other people know that we are intentionally doing so. And we should take it upon ourselves to inform others of the degree of risk they run of being wrong when they accept any our statements as a basis for action.

Knowing that there is a risk, and knowing even approximately what the level of that risk is, is a sign of professional competence. There are too many professionals who mistakenly believe that to raise issues of risk is a sign of incompetence – because it shows that they do not have full control. This is wrong. Knowing that we must share this understanding of risk with others on our own initiative is a sign of sound professional ethics.

The real professional is one who knows the risks, their degree, their causes, and the action necessary to counter them, and shares this knowledge with his colleagues and clients. If we attempt to carry out large projects with unclear goal statements, inadequate planning, less than wholly competent people, and weak management methods, in areas which are so uncharted as to justify the use of research funds, we should expect overruns of resource expenditure. We should remember that a professional engineer is expected to make a clear distinction between predictable costs and highly unpredictable cost factors. The question is how can we, as managers, avoid the worst effects of systems full of unknown when they go wrong on us. Risk prevention is more cost-effective than risk reduction. When something happens during the project that you did not foresee, which increases deviation from planned risk, immediately raise the issue, in writing, with your constructive suggestion as to how to deal with it.

Management involves constantly making judgments, with varied degrees of risk that these judgments lead to incorrect or inadequate decisions. We are referring here to perfectly ordinary deviations for factors such as human resource requirements, costs, implementation time, effective promised results.
4.5 BAYESIAN NETWORK BASED RISK ASSESSMENT

Trustworthiness of software is an important key factor to satisfy the customer expectations. Trustworthiness of a product depends upon the risk management involved in the various stages of process. Risk associated with the development may lead to decline in trustworthiness. Software process network model can be constructed based on the following processes: Planning, Requirement, Design, Development, Testing and Implementation. Risk can be occurred during any of these processes. Risk involved in these stages may have different types of impact on the product. According to that trustworthiness of the product will vary. Risk is nothing but the product of probability of risk factor and the impact of the risk.

\[
\text{Risk} = \text{Probability of risk factor} \times \text{Impact of the risk.} \quad (4.1)
\]

This risk factor deals with the uncertainty associated with the human factors which involved in the software development process. There are many human factors involved in the software development process such as psychological factors, skills, learning capacity of a person, productivity of a person, leadership quality of a person, Experience of a person etc. These factors have different levels of impact on the outcome of a project. So there is a need weigh the factors according to its impact on the product.

If the entire factor’s impact is considered equally while evaluating the risk, it may lead to improper risk assessment. Bayesian network model based risk assessment is introduced in this work [97]. Different types of human factors have different probability of risk on the product. So in order to estimate trustworthiness of a product there is a need to calculate joint probability of all human factors on the product. Bayesian network model is used here to evaluate the joint probability of risk factors on the product.
Bayesian network model is the combination of graphical theory model and probability model. Graphical model involves the process in which complicated system will be decomposed into number of smaller nodes to make the evaluation easy [98]. After analysing the smaller nodes in the system, they should be combined in order to make a decision about the system. For this purpose probability model is integrated with the graphical model in Bayesian network model. Join probability of 2 variables can be defined as,

$$P(A/B) = P(B/A)P(A)/P(B)$$  \hspace{5cm} (4.2)

The overall architecture of risk assessment in software development process is shown in figure 4.1.

Risk is defined as the product of probability of risk occurrence and impact of risk on the product. Probability of risk occurrence can be calculated using Bayesian network model [99]. Probability of risk occurrence is the product of experimental impact value calculated from Bayesian network and risk identification weighing function in the aspect of cost, time, CMMI. It can be expressed as,

$$P_{ij} = P_{ij}^*, R_l = P_{ij}^* f^l(Cost_j, Time_j, CMMI)$$ \hspace{5cm} (4.3)
Where $P_{ij}$ represents the probability of risk occurrence, $R_{ij}$ is the risk identification effectiveness, $P_{ij}^*$ is the experimental value calculated from Bayesian network. Impact of risk factor can be defined in terms of impact of sub deliverables in sub process which is defined as the product of experimental impact value calculated from Bayesian network and risk control weighing function based on time, cost, CMMI. Impact of risk factor can be expressed as,

$$I_{ij} = l_{ij}^* R_{C_j} = l_{ij}^* f^2 (\text{Cost}_j, \text{Time}_j, \text{CMMI})$$  \hspace{1cm} (4.4)

Where $l_{ij}$ represents the impact of risk, $R_{C_j}$ is the risk control effectiveness, $l_{ij}^*$ is the experimental value calculated from Bayesian network [100]. The occurrence of risk $R_{ij}$ can be expressed as,

$$R_{ij} \sim B(1, P_{ij})$$  \hspace{1cm} (4.5)

Total risk can estimated as follows:

$$E_{tot,j} = \sum_{i=1}^{\#} (R_{ij} \cdot I_{ij}) = \sum_{i=1}^{\#} (R_{ij} \cdot R_{ij}(\text{Cost}, \text{Time}, \text{CMMI}), l_{ij}^*, R_{C_j} (\text{Cost}, \text{Time}, \text{CMMI})$$  \hspace{1cm} (4.6)

Trustworthiness affected by the process risk can be expressed as

$$T_{Risk} = (T_1, ..., T_5)_{1\times 5}, W_{5 \times 1} = (E_{tot,1}, ..., E_{tot,5})_{1\times 6}.$$  

$$\begin{pmatrix}
IM_{1,1} & \cdots & IM_{1,5} \\
\vdots & \ddots & \vdots \\
IM_{5,1} & \cdots & IM_{5,5}
\end{pmatrix} \cdot \begin{pmatrix}
\omega_1 \\
\vdots \\
\omega_5
\end{pmatrix}$$  \hspace{1cm} (4.7)

Where, impact element $IM_{m,n}$ indicates the level of the $m^{th}$ sub-deliverable’s impact on the nth trustworthiness attribute.

Trustworthiness can be affected by process quality and risk factors associated with the development process [101]. Trustworthiness should be evaluated by considering both process quality and risk factors. Process quality can be expressed as,

$$T_{PQ} = f^P (\text{Cost}, \text{Time}, \text{CMMI})$$  \hspace{1cm} (4.8)
Trustworthiness can be evaluated from the difference between trustworthiness affected by process quality and risk reduction.

\[ T = T_{PC} - T_{Risk} \]  \hspace{1cm} (4.9)

Thus the constraints considered to calculate trustworthiness of a product leads to provide effective risk management process. In our study Human resource based risk is the key factor behind the risk management process. Better risk management process involved in the development process helps to avoid complete cancellation of a product and will lead to register the uniqueness in the market.

4.6 MANAGING THE ALLOCATION OF RESOURCES WITHIN PROGRAMMES

We are now going to examine in more detail programmes where resources have to be shared between concurrent projects. Typically, an information technology department has pools of particular types of expertise, such as software developers, database designers and network support staff, and these might be called upon to participate in a number of projects which can be going on at the same time.

In these circumstances, programme managers will have concerns about the optimal use of specialist staff. These concerns can be contrasted with those of project managers. The project managers are said to have an ‘impersonal relationship’ with resource types because, essentially, they require, for a competent systems analyst and who fills that role does not matter. The programme manager has a number of individual systems analysts under his or her control whose deployment has to be planned.

When a project is planned, at the stage of allocating resources, programme management will be involved some activities in the project might have to be delayed until the requisite technical staff are employed full-time, work on other projects. Where expensive technical staff are employed full-time, then you would want to avoid them having short periods of intense activity.

4.6.1 Research and Development programmes

Truly innovative organizations, especially those that are trying to develop new products for the market, are well aware that projects will vary in terms of their risk of failure and
the potential returns that they might eventually reap. Some development projects will be relatively safe, and result in the final planned product, but that product might not be radically different from existing ones on the market. Other projects might be extremely risky, but the end result, if successful could be a revolutionary technological breakthrough that meets some pressing but previously unsatisfied need.

The risks associated with an innovative project will fluctuate. Research work may lead to technological breakthroughs or to the discovery of insurmountable problems, so the portfolio of projects needs to be reviewed on a regular basis. Some technological developments, if handled properly, benefit whole industries. The development of the internet and the world wide web is an example. Often, technological products can only be exploited if they inter-operate with other products. Companies therefore often come together to work collaboratively on new technologies in a ‘pre-competitive’ phase. Separate projects in different organizations need to be coordinated and this might be done as a programme.

Where necessary, assess the efficiency of changing the plan to fit the resources. In general, the allocation of resources to activities will lead us to review and modify the ideal activity plan. It may cause us to revise stage or project completion dates. In any event, it is likely to lead to a narrowing of the time spans within which activities may be scheduled.

The final result of resource allocation will normally, be a number of schedules, including:

- Activity Schedule indicating the planned start and completion dates for each activity.
- Resource schedule showing the dates on which each resource will be required and the level of the requirement.
- Cost Schedule showing the planned cumulative expenditure incurred by the use of resource over time.

**4.7 RESULTS AND DISCUSSION**

From the analysis of the several of articles of distributed software systems, we included four new risk areas to the existing eight risk areas. The risk possibilities of the risk areas are ranges in to Low (0-10), Medium (11-20), High (21-30) and Extreme High (31-40) as shown in figure 4.3. On the basis of the impact of risk possibilities, they are categorized. The risk areas from 1 to 18 denotes possibilities of various risk areas.
The first added risk area is loss of team coherence as in figure 4.3. Team members must coordinate with themselves for the success of the system. The project manager should try to reduce negative thoughts on the project among team members.

The next proposed risk area is Coordination breakdowns. The customer and the organizations break downs can be minimized by the communications between the units of the organization and end users. Vertical and horizontal coordination breakdowns are controlled by the proper communication between the top, middle, low levels of management. To promote the coordination between the team members, the work charts are used to store ongoing task of the members. The work charts are controlled by the project manager and it can be easily access by all the members of teams. The procedural values of the task units are stored in database for the future use.
The above figure 4.4 shows the team coherence of team action, decision making and involvement. The Time zone difference is the main proposed risk area, which is having the greatest impact on the coordination of the distributed software system [93]. Face to Face meetings in a common work place is not possible in the distributed software development. The virtual face to face meeting can be easily done with the help of teleconferencing. Each of the several difference zones makes the difference in timings. Coordinating the different zone team members is a difficult process. Knowing of the time zone difference around the world is needed. So we propose a web tools for the successful meeting in distributed environments. The time charts is prepared by selecting the date, month, year of the meeting.

4.7.1 Risk Managerial Techniques

Risk manager should try to understand the personal variables risk such as application knowledge, Educational qualifications and technical variables risk such as software knowledge, hardware knowledge.
The ability of the staffs on the distributed work environment should be identified by the risk manager and appropriate procedural activities should be implemented to minimize the effect of the risk. After finding the risks, he should try to minimize its impact on productivity.

4.7.2 Configuration Management Techniques

Configuration management (CM) is the process of complete recording and updating of information that describes an organization’s product. The purpose of Software Configuration management is to increase and maintain the reliability of the product. Software configuration management involves identifying configuration settings, controlling, recording and reporting status about the changes in configuration. The changes in the product are introduced by the developer in order to satisfy the user requirements. End user stratifications increase the organizations productivity. Requirements of end users will be changed continuously. Requirements are modified because of changing circumstances or the users get a clearer idea of what is really needed.

Other, internal, changes will crop up, careful control of these changes is needed because an alteration in one document often implies changes to other documents and the system products based on that document. Control of changes and documentation ought to be the responsibility of someone who may variously be named the Configuration Manager or Configuration Librarian. The responsibilities of configuration manager is given below:

- The identification of all items that are subject to change control.
- The establishment and maintenance of a central repository of the master copies of all project documentation and software products;
- The setting up and running of a formal set of procedures to deal with changes;
- The maintenance of records of who has access to which library items and the status of each library item.
4.7.3 Change control procedures

A simple change control procedure for operational systems might have the following steps

- One or more users might perceive a need for a modification to a system and ask for a change request to be passed to the development staff.

- The user management consider the change request and, if they approve it, pass it to the development management.

- The development management delegate a member of staff to look at the request and to report on the practicality and cost of carrying out the change. They would, as part of this, assess the products that would be affected by the change.

- The development management report back to the user management on the findings and the user management decide whether, in view of the cost quoted, they wish to go ahead.

- One or more developers are authorized to take copies of the master products that are to be modified.

- The copies are modified. In the case of software components this would involve modifying the code and recompiling and testing it.

- When the development of new versions of the product has been completed the user management will be notified and copies of the software will be released for user acceptance testing.

- When the user is satisfied that the products are adequate they will authorize their operational release. The master copies of configuration items will be replaced.

4.8 CONCLUSION

In this analysis, we have proposed Bayesian based risk assessment in software development process in which additional risk factors associated with human resource are included. The proposed system first identifies the risk factors associated with human resource. There are many risk factors associated with human resource. Each of these
risk factors are weighed and taken in account to evaluate the risk associated with the development process. Bayesian based risk assessment process model is introduced in this paper. Risk control techniques are applied to the problem identified in the evaluation stage. Risk managerial techniques, Configuration management techniques are the important resolution techniques in the distributed software system. The proposed risk assessment technique performs better because the factors considered to evaluate the risk have great influence on the productivity of the system.