INVESTIGATION OF QUALITY AND FUNCTIONAL RISK ISSUES IN REENGINEERING PROCESS OF LEGACY SOFTWARE SYSTEM

Anand Rajavat \(^1\) and Vrinda Tokekar \(^2\)

\(^1\) Department of CSE, Shri Vaishnav Institute of Technology and Science, Indore, M. P., India
\(^2\) Department of IT, Institute of Engineering and Technology (DAVV), Indore, M. P., India

ABSTRACT

The modern business environment requires organizations to be flexible and open to change if they are to gain and retain their competitive age. Competitive business environment needs to modernize existing legacy system in to self-adaptive ones. Reengineering presents an approach to transfer a legacy system towards an evolvable system. Software reengineering is a leading system evolution technique which helps in effective cost control, quality improvements and time and risk reduction. However successful improvement of legacy system through reengineering requires portfolio analysis of legacy application around various quality and functional parameters some of which includes reliability and modularity of the functions, level of usability and maintainability as well as policy and standards of software architecture and availability of required documents. Portfolio analysis around these parameters will help to examine the legacy application on quality and functional gaps within the application [1].

In this paper we identify and measure risk factors related to quality and functional aspect of legacy system. Paper analyzes a variety of risk components related to quality and functional dimensions of legacy application. Identification and development of various metrics and important measures to compute impact value of individual risk component is also addressed.

Existing quality and functional level of legacy system are considered to identify and categories risk components. The impact of various technical issues such as inconsistency between architecture of legacy and target system, unmodularised legacy system architecture, unavailability of required design documents, high degree of coupling between modules of legacy system has been covered in the paper.

KEYWORDS

Reengineering, Quality, Risk Measurement

1. INTRODUCTION

Technical reasons to reengineer existing legacy system includes mission requirements of target system, modification for new application, quality improvement of existing applications, rehosting to a new system, translation to a new language etc. Reengineering usually requires significant start-up resources such as personnel, organizational infrastructure, managerial & financial support and a considerable legacy system state to support further modification through reengineering.
In this paper we cover risk components related to the various quality attributes and functional capability of the legacy system. A legacy system having basic functional capability and fundamental quality level is a good candidate for reengineering. Reengineering can improve quality and functional capability of such legacy system in a cost effective manner. On the other side if the legacy system having very poor functional and quality capability the risk of reengineering increases. The redevelopment option is generally used for such type of legacy systems.

We identify and measure risk impact due to the functional competence and quality factors which includes reliability, availability, usability, modularity and performance of the legacy system. Probability of reengineering success increases with the legacy system having considerable quality levels. Impact of inconsistency among legacy and target system architecture, high degree of coupling, complex user interface as well as level of reliability and availability are measured within the context of technical domain. Identified risk components and relative risk factors helps to analyze quality and functional capability of existing legacy system [2].

2. LITERATURE REVIEW

Most of the legacy systems are too expensive to maintain and also confine development growth of organizations which operate them. These legacy systems are normally mission-critical: if one of these systems stops working the overall business process stops. To maintain the value of such legacy system effectively it is essential to develop a systematic modernization strategy which satisfies; modern business mission, technology and user needs. The appropriate selection of modernization strategy requires analyzing various quality and functional parameters of legacy software system in concern with requirement of target system. Increase failure rate of reengineering project require evaluating risks in the process of reengineering. Yonggui in [3] proposed a risk assessment model based on study findings and the actual survey of enterprises. In addition analysis helps enterprises to improve utilization and optimization of the limited resources, so as to improve the performance of reengineering process. Proposed model develop a fuzzy AHP method to select BPR appropriately for an organization from available alternatives.

Bennett et al. [4] propose a two-phase model to assist organizations in making decisions about legacy system. This approach is very close to business process reengineering [5] that aims to change the processes and information systems of an organization according to new business directions [6]. A cost-efficient systematic approach proposed by Mehdi Amoui in [7] for the modification of software system concentrate on the requirements of target system. Proposed approach use novel concepts of self-adaptive software, co-evolutionary model, and primitive effecting operations.

3. QUALITY ISSUES IN LEGACY SYSTEM REENGINEERING

Quantifiable quality measures are useful to achieve success in reengineering process. Understanding the factors that influence software quality and various quality drivers will help software engineers and managers to take more informed decision in improving the software product through reengineering [8]. We identify and develop risk components and impact measurement mechanism for quality segment as follows:
i. Reliability Risk Component

Reliability is a complex concept that should always be considered at the system rather than the individual component level, because the components in the system are interdependent [9]. The level of reliability risk affects the overall cost and schedule for the evolution of legacy system through reengineering.

- Impact Measurement Mechanism

Reliability implies the extent to which a legacy system performs its intended functions with required precision. The reliability level of existing legacy system will affect overall impact of reliability risk in reengineering. High failure rate for different functions of legacy system will affect overall cost, schedule and risk of reengineering. The impact of reliability risk component for an existing legacy system is measured using equation 1.

\[
(TIRR) = (POSF) + ROFO
\]

Where TIRR represents Total impact of reliability risk, POSF represents probability of system failure, ROFO represents rate of failure occurrence. The value of probability of system failure POSF and rate of failure occurrence ROFD are computed using equation 2 and 3.

\[
Probability\ of\ system\ failure\ (POSF) = \frac{Number\ of\ failures}{Total\ Number\ of\ services}
\]

\[
Rate\ of\ failure\ occurrence\ (ROFD) = \frac{Number\ of\ unexpected\ behavior}{Total\ Number\ of\ operational\ time\ units}
\]

ii. Usability Risk Component

The usability risk component represents the loss associated with dissatisfaction of user due to inefficient and complex system support and user interface. It defines inconvenience and impracticality of use [10].

- Impact Measurement Mechanism

User friendliness has become ubiquitous in the success of any software product. From the customers’ standpoint, all problems they encounter while using the legacy system, not just the valid defects, are problems with the software. Problems that are not valid defects are the usability problems. These non-defect-oriented problems, together with the defect problems, constitute the total problem space of the legacy system from the customers’ perspective. The equation 4 is used to compute impact of usability risk component.

\[
(TIUR) = \frac{Total\ No.\ of\ Non\ Defect\ Oriented\ Problems}{Total\ Problem\ Space}
\]

Where TIUR represents total impact of usability risk component, total problem space is computed using equation 5

\[
Total\ problem\ space = Valid\ defect\ problem\ +\ Non\ defect\ problems
\]
iii. Performance Risk Component

Performance of a legacy system is defined as the fulfillment of a system purpose without waste of resources, such as memory space and processor utilization, network bandwidth, time etc [11]. Performance risk component defines the loss associated with improper resource utilization and massive response time. Lower degree of processing speed, throughput and efficiency also affect impact of performance risk component.

System performance is the critical issue to the success of any software system. However many legacy software systems fail to meet their performance objectives when they are initially constructed.

- Impact Measurement Mechanism

Measurement of performance risk component in reengineering process of legacy system involves quantification of various performance measures which will affects overall impact of performance risk component. We investigate several measures and corresponding scale values shown in table 1. Once we have identified the key measures and assign scale value to each measure a mean opinion score board as shown in table 2 is used to check performance level and Impact value of performance risk component.

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Measures</th>
<th>Low</th>
<th>Average</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Memory space utilization</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Processor utilization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Network bandwidth utilization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Response Time</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Processing speed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Throughput</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Defect Removal Efficiency</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Dependencies on external interface</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Level of user interface</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Capacity to handle predefined user load</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total wattage value</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: 2 Mean Opinion Score for Performance Risk Component

<table>
<thead>
<tr>
<th>Score</th>
<th>Comment</th>
<th>Total Impact of Performance Risk (TIPER)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20-30</td>
<td>Good system performance</td>
<td>0</td>
</tr>
<tr>
<td>10-20</td>
<td>System performance is patchy. Software is good at some things, but there's area for improvement.</td>
<td>0.5</td>
</tr>
<tr>
<td>0-10</td>
<td>Disgraceful system performance</td>
<td>1</td>
</tr>
</tbody>
</table>
The risk measurement mechanism consists of integrating individual technical performance measures in a way that produce an overall impact of performance risk in reengineering. The computed index shows the impact of performance risk presently in the legacy system.

iv. Modularity Risk Component

A module represents a set of related concerns. Coupling is a measure of interconnection among modules in a program structure. Coupling depends on the interface complexity between modules, the point at which entry or reference is made to a module, and what data pass across the interface. [9].

Modularity risk component represents the risk of loss associated with unmodularised legacy system having large degree of coupling between its different modules.

• Impact Measurement Mechanism

Effectiveness of reengineering strategy requires modules of legacy system are independent of one another but can communicate with each other in a loosely coupled fashion. To quantify impact value of modularity risk component we use a metric for module coupling that encompasses data and control flow coupling, global data, and environmental coupling. The value of overall module coupling represents the impact of modularity risk component. The measures required to compute module coupling are defined in terms of each of the three coupling as

For data and control coupling

1. $D_i$: number of input data parameters
2. $C_i$: number of input control parameters
3. $D_o$: number of output data parameters
4. $C_o$: number of output control parameters

For global coupling

1. $G_d$: number of global variables used as data
2. $G_c$: number of global variables used as control

For environmental coupling

1. $W$: number of modules called (fan-out)
2. $R$: number of modules calling the module under consideration (fan-in)

Using these measures a module coupling indicator, $M_c$, is defined using equation 6 & 7.

$$M_c = K/M \quad (6)$$

$$M = D_i + a.C_i + D_o + b.C_o + G_d + c.G_c + W + R \quad (7)$$

Where $k = 1$ represents a proportionality constant and $a$, $b$ and $c$ are the coefficient, that may be adjusted by the analyst. Higher value of the $M_c$ represents the lower degree of overall module coupling which is stand for low impact of modularity risk in reengineering process.
v. **Availability Risk Component**

To reengineer a legacy system effectively, a system baseline under configuration management should be in place to aid in disciplined reengineering. In addition, the following items need to be in place: analysis and design documents, data on the costs of maintaining the system, adequate configuration management, and project management planning capabilities. If these capabilities are not available, the reengineering effort becomes crippled and chaotic. Availability risk component represents the loss associated with unavailability of required design and change management document of legacy system [12].

- **Impact Measurement Mechanism**

Measurement of availability risk component requires thoroughly understanding of system’s configuration to accurately measure availability of legacy system as experienced by end users. This includes components and resources used by the application and the hardware and software components required to access those resources. The next step is to monitor all these components for outages, then calculate end-to-end availability. The impact of availability risk component is calculated as follows:

Table: 3 represents measures and relative scale used to compute total scale value of availability risk component.

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Measure</th>
<th>Scale value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes = 1</td>
<td>No = 0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Unavailability of required analysis documents of legacy system</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Unavailability of required design documents of legacy system</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Unavailability of required resources on time</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Unavailability of user requirements and feedback</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Insufficient documentation of development environment</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Inadequate quality &amp; security of available information</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Inadequate accessibility and presentation of required information.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Required information loss or theft</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Low degree of availability for different functions of legacy system</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total scale value</td>
<td></td>
</tr>
</tbody>
</table>

Once we have identified the key measures, a measurement metrics represented by equation 8 is developed to compute Impact value of availability risk. Value of each measure is calculated by using a scale that represents to what extent users of legacy system and developers of target system agree or disagree for respective measure. If we use a scale 1 for yes and 0 for no option then we get the value for each measure by looking at the answers given by users of legacy system and developers of target system. The average value is used to assign scale for each measure for more than one answer.
Where TIAR represents total impact of availability risk, $X$ represents scale value of each measure and $N$ represent number of measures.

4. **FUNCTIONAL ISSUES IN LEGACY SYSTEM REENGINEERING**

A high level of strategic risk impact measurement has substantial impact on the success or failure of a reengineering approach. However these strategies may be seriously flawed or incomplete due to lack of attention to functional risk attributes of legacy and target system. In this paper we analyze viability of existing legacy system architecture to satisfy requirements of target system. We also analyze interface dependency between various functions of the legacy system [13].

We quantify risk impact due to the different functional issues like level of training for existing workforce, intensity of various functional parameters including backlog, defect rate response time and maintenance time. To consider these issues we identify five different functional risk components as follows.

i. **Software Architecture Risk Component**

The software architecture of a computing system is the structure or structures of the system, which comprises software components, the externally visible properties of those components, and the relationship between them. Failure to evaluating the existing architecture and inconsistency between present and target architecture will increases the impact of software architecture risk component. Software architecture risk component represents the risk of loss associated with inconsistency between existing architecture of legacy system and desired architecture of target system [14, 15].

- **Impact Measurement Mechanism**

Software architecture is defined as the hierarchical structure of system components, the manner in which these components interact, and the structure of data that are used by the components. Understand ability and viability of existing legacy system architecture plays an important role in the success of reengineering. If the old architecture is well understood and documented, it becomes possible to use the existing interface and documentation style in the new architecture to increase user friendliness. Software architecture impact measurement mechanism analyzes existing architecture of legacy system, comprising software elements, the externally visible properties of those elements, and the relationships between them in accordance with desired architecture of target system. Impacts of software architectural risk often lead to project inefficiencies, poor communication, and inaccurate decision making. Equation 9 is used to compute impact of software architecture risk component.

\[
\text{TIAR} = \frac{\sum_{i=1}^{N} (X_i)}{N}
\]

Where, TIAR represents total impact value of software architecture risk component, ANAI represents architecture nonadaptability index for existing legacy system. We define a nonadaptable element of architecture which cannot be reused or adaptable in the development of
new architecture for target system. When a large legacy system contains more than one architecture layout to represents different components of system we define nonadaptable architecture that cannot be reused in the development of target system architecture. In this case we also use software nonadaptable index (SNAI) to compute impact of software architecture risk component using equation 10.

\[
T[AR] = [SNAI] \quad (10)
\]

To compute architecture nonadaptable index (ANAI) and software nonadaptable index (SNAI) we utilize equation 11 & 12.

\[
\text{Architecture nonadaptability index (ANAI)} = \frac{\text{No. of nonadaptable elements in the existing architecture}}{\text{Total number of elements in existing architecture}} \quad (11)
\]

\[
\text{Software nonadaptability index (SNAI)} = \frac{\text{No. of nonadaptable architecture of the software}}{\text{Total number of architectures for that software}} \quad (12)
\]

ii. Complexity Risk Component

Complexity risk component is the risk of loss associated with complex legacy system architecture with large numbers of depended links within the system and with external entities. There are several attributes of a system that can indicate how complex it is such as number of organizational units involved, number of stakeholders involved, number of hierarchical levels occupied by the users, uncertain requirements, changing scope, number of complex application with complex relationship.

- Impact Measurement Mechanism

Measurement mechanism of complexity risk component analyze and measure several system attributes such as number of organizational units involved, number of stakeholders involved, number of hierarchical levels occupied by users, uncertain requirements, changing scope, number of complex application with complex relationship to compute total risk impact of project complexity risk component in the reengineering process of legacy system. We assign corresponding scale value to each of the identified complexity attribute by analyzing existing state of legacy system. The average scale value of all the attributes represents impact value of complexity risk component. Table 4 summarizes various complexity attributes and instruction to assign relative scale value.

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Complexity Attributes</th>
<th>Scale value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Yes = 1</td>
</tr>
<tr>
<td>1</td>
<td>Large number of organizational units involved</td>
<td>No = 0</td>
</tr>
<tr>
<td>2</td>
<td>Large number of stakeholders involved</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Number of hierarchical levels occupied by users is high</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Probability of uncertain requirements</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Possibility of changing scope</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Large number of complex application with complex relationship</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Number of external interface is large</td>
<td></td>
</tr>
</tbody>
</table>
8 Large number of user inquiries
9 Large numbers of user input and outputs

| Total Impact of complexity risk (TICR) | Average scale value of all the complexity attributes |

### iii. Maintainability Risk Component

Maintainability risk component represents the impact of flatten legacy system or legacy system having improper maintenance activities which will increases complexity of legacy system structure. Maintainability impact measurement mechanism analyzes legacy system architecture to identify number of modules added, deleted and changed after the first release of the legacy system.

- **Impact Measurement Mechanism**

  The improper maintenance of legacy system has a large influence on reengineering failure. Many legacy systems are not under adequate control due to the ad hoc maintenance activities during the operational life of the legacy system. These legacy systems also have high degree of coupling between different sections of legacy system, inadequate historical measurement and inadequate change control processes. As a result, it is difficult to understand the current system and improve quality and functionality through reengineering. Maintainability risk measurement mechanism analyzes structure of legacy system to identify different functional and operational changes made after the first release of the legacy system. Impact of maintainability risk component is measured by using equation 13.

\[
TIMTR = \frac{Mc + Ma + Md}{MT}
\]

Where
- \(TIMTR\) represents total impact value of maintainability risk component
- \(MT\) represents the total number of modules in the existing legacy system
- \(Mc\) represents the number of modules in the existing legacy system that have been changed after first release
- \(Ma\) represents the number of modules in the existing legacy system that have been added after first release
- \(Md\) represents the number of modules in the existing legacy system that have been deleted after first release

### iv. Training Risk Component

Training risk component defines the risk of loss associated with unfamiliarity of the existing work force on advanced tools and technology used to achieve target system goals. Training risk measurement mechanism analyzes requirement of customized and specialized training programs and special consulting services for existing workforce [16].

29
• **Impact Measurement Mechanism**

Training risk measurement mechanism measures requirements of customized and specialized training programs and special consulting services for present user of the legacy system so that they are comfortable with functionality of target system. The training risk recognizes the elements and steps necessary for training the various staff to use of the relevant functionality of the target system. The lack of training for the work force on advanced tools & technology used in target system can causes reengineering efforts to fail. If the existing work force is not capable to grasp advanced tools and technology used in target system, organizations should invest large effort to train existing staff. Training impact measurement mechanism identifies and quantify training requirement for existing work force. The impact of training risk component depends on the number of untrained workforce in the organization to support operational plan of target system. Equation 14 is used to measure impact value of training risk component.

\[
\text{TITRR} = \frac{\text{No of workforce required training}}{\text{Total no of workforce in organization}}
\]  

Where TITRR represents total impact value of training risk component

v. **Technology Risk Component**

Reengineering involve systematic transformation of a legacy system in to a new form to realize quality improvement in operation, system capability, functionality, performance, or evolvability at a lower cost, schedule, or risk to the customer. Technology risk component correspond to the possibility that the present state of legacy system and organization will fail to support advance technology and tools used to fulfil the requirements of target system. The level of deterioration and obsolescence in legacy system affect the overall impact value of technology risk component.

• **Impact Measurement Mechanism**

Many organizations are planning to “migrate “ their legacy system to respond to new enterprise standards, incorporate new product and system features, improve performance, and cope with endless new software release and staff off hardware and software obsolescence. However many of these effort are often less than successful due to incompetent legacy system architecture and inadequate organizational support. Technology risk measurement mechanism defines several technical attributes to analyze existing capability of legacy system state. If existing hardware and software contains serious bugs it could degrade efficiency or effectivenes of technology adopted for achieving target system goals. Incompatibility of technology with other technology used in legacy system also affects impact of technology risk component. Impact measurement mechanism allocates discrete scale value to each identified technical attributes to compute impact value of technology risk component. Table 5 summarizes attributes and relative scale values used in the measurement of technology risk component.
Table: 5 Attributes of Technology Risk Component

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Measure</th>
<th>Scale value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Existing hardware and software contain serious bugs</td>
<td>Yes = 1</td>
</tr>
<tr>
<td>2</td>
<td>Incompatibility of technology with other technology used in legacy system</td>
<td>No = 0</td>
</tr>
<tr>
<td>3</td>
<td>Inadequate business process to support new technology</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Increased backlog rate</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Increased defect rate of legacy system operations</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Obsolescence operating system version</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Obsolescence hardware version, Increased maintenance time.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Unbearable cost and schedule of applying new technology</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Adequate training required for existing users on new technology</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Inconsistency between technology and organizations strategic plan</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>New technology demands the use of new analysis, design and testing methods.</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>New technology put excessive performance constraints</td>
<td></td>
</tr>
</tbody>
</table>

Identified technical attributes and relative scale value assign by developers of target system is used by metrics shown in equation 15 to compute impact value of technology risk. We use a scale 1 for yes and 0 for no option for each attribute assign by developers of target system. The average value is used to assign scale for each measure for more than one answer.

\[
TITER = \frac{\sum_{i=1}^{N} (X_i)}{N}
\]

Where TITER represents total impact of technology risk, X represents scale value of each measure and N represent number of measures.

5. Conclusion

Now a day’s many software industries adopt reengineering to improve their legacy systems. Legacy system reengineering supports functional and quality improvements with time, cost and risk reduction. The cost and risk of reengineering depends on functional and quality competence of existing legacy system. A legacy system with basic functional capability and fundamental quality level is a good candidate for reengineering. Reengineering can improve quality and functional capability of such legacy system in a cost effective manner.

In this paper we analyze quality and functional issues in reengineering process of legacy software systems. Paper focuses on quality and functional segment of legacy system. In quality
segment we analyze and measure several quality parameters including reliability, usability, performance, modularity and availability for existing legacy system. Functional view covers issues related to the maintainability and architecture complexity of legacy system as well as technology and training concerns associated with requirements of target system. We first categories various risk components to analyze functional capability and quality level of legacy system. A variety of measurement metrics are developed to measure Impact value of each risk component related to quality and functional issues involve in reengineering process of legacy software system.

REFERENCES


[2] Anand Rajavat, Dr. (Mrs.) Vrinda Tokekar, “ReMm- a Measurement Model to Quantify the Effect of Reengineering Risk in Quality Perspective of Legacy System” published in Springer International Conference on Advances in Information Technology and Mobile Communication – AIM 2012


Authors

Anand Rajavat
Associate professor
Department of Computer Science & Engineering
Shri Vaishnav Institute of Technology and Science Indore, M. P., India
B.E. (Computer Science and Engineering)
M.E. (Software Engineering)
Ph.D (Pursuing) (Computer Engineering)
Areas of Interest:
Software Engineering, Object Oriented Analysis and Design, Computer Architecture

Dr. (Mrs.) Vrinda Tokekar
Professor & Head,
Department of Information & Technology,
Institute of Engineering & Technology,
Devi Ahilya University, Indore (M.P.) India
Ph. D. (Computer Engg.) in 2007 from DAVV, Indore
M.E. (Computer Engg.) in 1992 from DAVV, Indore
B.E. (Hons.) EEE, BITS Pilani in 1984,
Areas of Interest:
Computer Networks, Distributed Computing, Security in Wireless Networks, e-Governance,
Multimedia Communication, Software Engineering