iJET EXPLORE International Journal of Engineering & Technology
CENTRE OF PROFESSIONAL RESEARCH PUBLICATIONS
ISSN 2049-3444

International Journal of Engineering and Technology Volume 2 No. 7, July, 2012

A Literature Review and Classification of Selected Software Engineering Researches

Ahmed Saleem Abbas, W. Jeberson, V.V. Klinsega
SHIATS, University, Department of Computer Science & Information Technology, India

ABSTRACT

In this paper, a literature review and classification scheme for selected software engineering researches is presented. The study shows that an increasing volume of software engineering researches have been conducted in diverse range of areas. The articles are classified and results of these are presented based on classification scheme that consist of five main categories: software development process, software management, software engineering techniques, software re-engineering, and software applications. Analyses of the selected researches are carried out and gaps in the research are identified. A comprehensive list of references is presented. This review is intended to provide impetus in research and help simulate further interest.

Keywords: *Software engineering, Software Re-engineering, Review, Software development process, Software engineering techniques.*

1. INTRODUCTION

More than 250 papers, which were published between the time period 2008-2012 in conferences and international journals like Springer, IEEE, IJSEA, IJCSI, IJCSE, IJCA, IJEST, famoosR, COMATS, and Elsevier were collected, analyzed and classified into a number of categories and subcategories. The study led to the identification of gap in software engineering researches and enabled the authors to recommend the area where there is a lot of scope for future research.

What is software engineering?

Software engineering can be defined as the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software.[1] It is the application of engineering to software because it integrates significant mathematics, computer science and practices whose origins are in engineering.[2] It is also defined as a systematic approach to the analysis, design, assessment, implementation, testing, maintenance and reengineering of software, that is, the application of engineering to software.[3] The term software engineering first appeared in the 1968 NATO Software Engineering Conference, and was meant to provoke thought regarding the perceived "software crisis" at the time.[4][5]

2. IDENTIFIED SOFTWARE ENGINEERING RESEARCH AND EXTENT OF RESEARCH CARRIED OUT

The classification framework shown in figure (1) is based on the literature review and the nature of software engineering researches, which meant to give an understanding of how the subject has evolved and progressing. Since software engineering is an important area it is very essential to be clear of it's definition. There are several researchers working in this area and trying to define and specify the universally accepted list of software engineering principles and concepts and trying to apply all these principles and concepts on a real project but until now they could not do that completely and perfectly. The software engineering researches can be classified into five categories, which are; (I) software development process which account to 35 % (90 papers), (II) software management which counts to 30 % (75 papers), (III) software re-engineering which shares (36 papers) of the distribution 14 %, (IV) software engineering techniques the contribution in which is 13 % (34 papers) and (V) software applications which accounts to 8 % (19 papers). The proportion of research carried out in each category is shown in figure (2).

These main categories are further divided into a number of sub categories as in figure (1). The percentage distribution of research in the main categories is shown in figure (2) and according to the year of publishing these

Papers can be grouped as shown in figure (3). The papers are further grouped based on the journal in which they are published as shown in figure (4), and the number of papers published in each journal as shown in table (3). All the categories and their sub-categories will be discussed in detail in the later sections with graphs and tables.

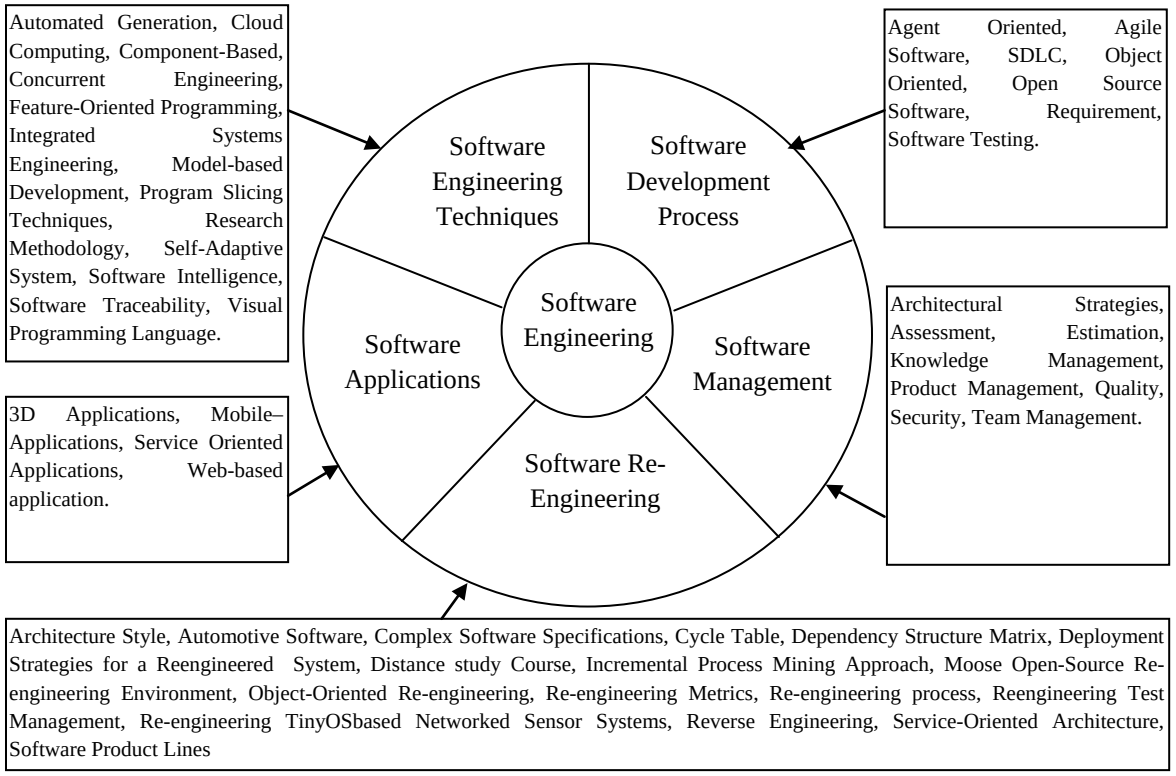

Figure (1): the literature reviews papers classification

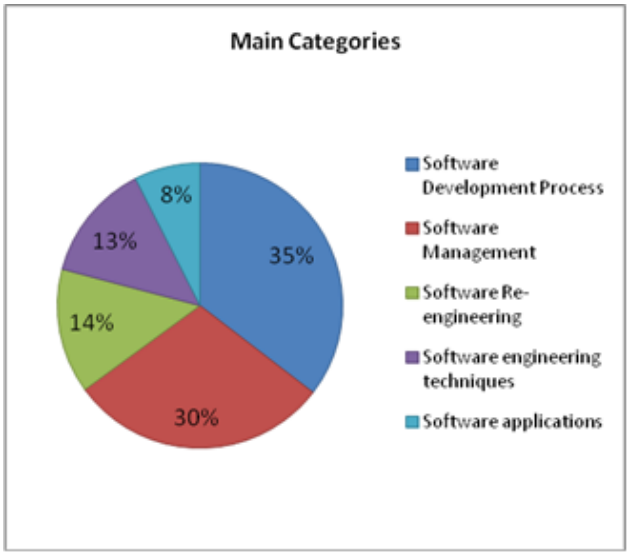

Figure (2): Percentage distribution of research in main categories

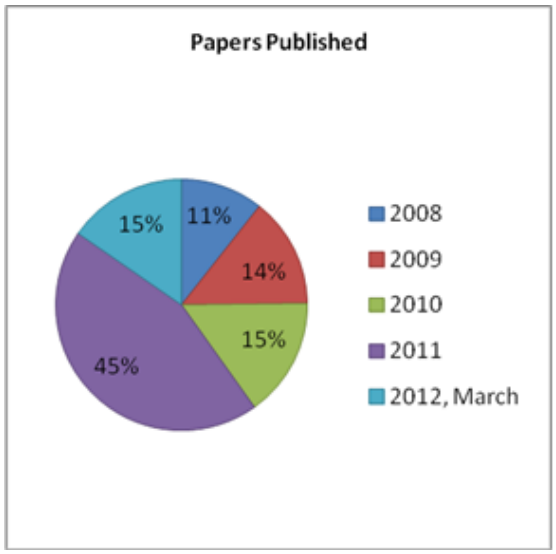

Figure (3): The papers published in a year

Table (1): Number of published papers in each category

Main Categories	No. of papers
Software Development Process	90
Software Management	75
Software Re-engineering	36
Software engineering techniques	34
Software applications	19
Total	254

Table (2): Number of published papers in each year

Year	No. of published papers

2008	27
2009	36
2010	39

2011	113
2012	39
Total	254

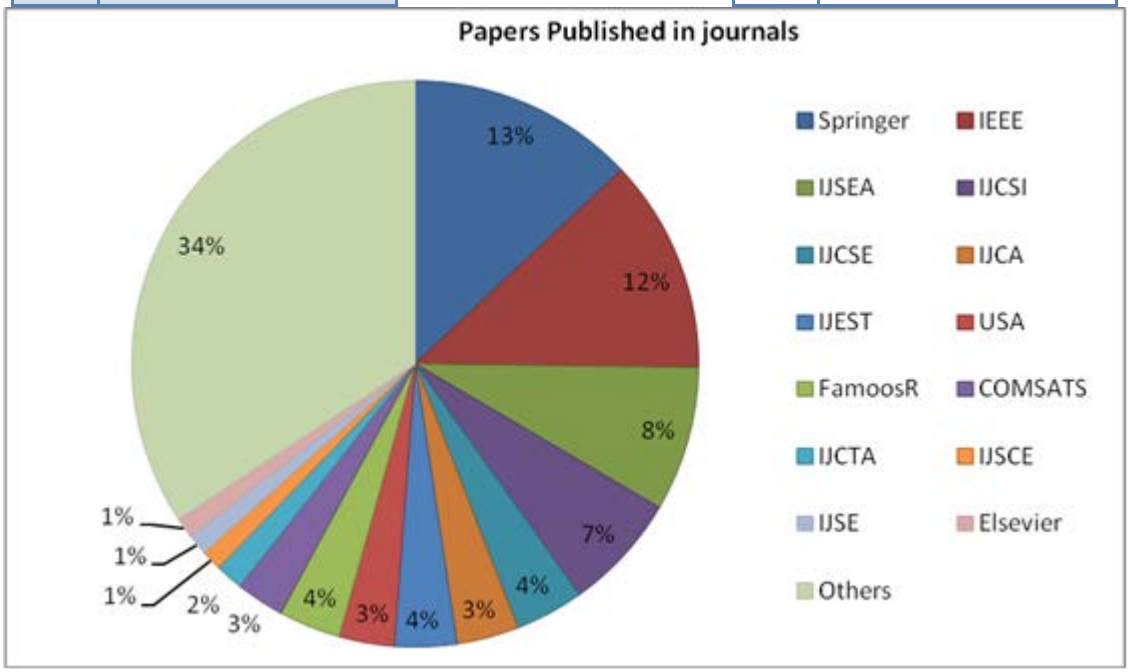

Figure (4): Papers published in various journals

Table (3): Number of papers published in various Journals

Journals / Institute	No. of published papers	Journals / Institute	No. of published papers
Springer	33	FamoosR	9
IEEE	31	COMSATS	7
IJSEA	21	IJCTA	4
IJCSI	17	IJSCE	3
IJCSE	10	IJSE	3
IJCA	9	Elsevier	3
IJEST	9	Others	87
USA	8	Total	254

Extent of research in subcategories of each category is described in detail as follows:

2.1 Software Development Process

A software development process, also known as a software development life cycle (SDLC), is a structure imposed on the development of a software product. Similar terms include software life cycle and software process. It is often considered a subset of systems development life cycle. There are several models for such processes, each describing approaches to a variety of tasks or activities that take place during the process. Some people consider a life-cycle model a more general term and a software development process a more specific term. For example, there are many specific software development processes that 'fit' the spiral life-cycle model. ISO/IEC 12207 is an international standard for software life-cycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software [7].

There are many activities involved in software development called as “Software development activities”, these activities is planning (requirements analysis), design, Implementation, testing and documenting, Deployment and maintenance.

In this research area there are 90 papers, which are analyzed and classified into number of sub-categories, which are described briefly as follows:

a. SDLC: It is a software development life cycle and there many models like Waterfall model, V model, spiral model, incremental model, and evolutionary model [7]. In this subcategory there are 32 papers and their percentage is 34%.

b. Requirement engineering: Requirement is a singular documented physical and functional need that a particular product or service must be or perform. It is most commonly used in a formal sense in systems engineering, software engineering, or enterprise engineering. It is a statement that identifies a necessary attribute, capability, characteristic, or quality of a system for it to have value and utility to a user [7]. In this subcategory there are 15 papers and their percentage is 16%.

c. Software testing: Software testing can be stated as the process of validating and verifying that a software program/application/product: meets the requirements that guided its design and development; works as expected; and can be implemented with the same characteristics [8]. In this subcategory there are 15 papers and their percentage is 16%.

d. Agile Software development process: Agile software development is a group of software development methods based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. It promotes adaptive planning, evolutionary development and delivery, a time-boxed iterative approach, and encourages rapid and flexible response to change. It is a conceptual framework that promotes foreseen interactions throughout the development cycle [9]. In this subcategory there are 12 papers and their percentage is 13%.

e. Open Source Software development process: Open-source software (OSS) is computer software that is available in source code form: the source code and certain other rights normally reserved for copyright holders are provided under an open-source license that permits users to study, change, improve and at times also to distribute the software. Open source software is very often developed in a public, collaborative manner. Open-source software is the most prominent example of open-source development and often compared to (technically defined) user-generated content or (legally defined) open content movements [10]. In this subcategory there are 7 papers and their percentage is 8%.

f. Object oriented software development process: It is a programming paradigm using "objects" – data structures consisting of data fields and methods together with their interactions – to design applications and computer programs. Programming techniques may include features such as data abstraction, encapsulation, messaging, modularity, polymorphism, and inheritance. Many modern programming languages now support OOP, at least as an option [11]. In this subcategory there are 6 papers and their percentage is 6%.

g. Agent oriented software development process: It is a programming paradigm where the construction of the software is centered around the concept of software agents. It has externally-specified agents (with interfaces and messaging capabilities) at its core. They can be thought of as abstractions of objects. Exchanged messages are interpreted by receiving "agents", in a way specific to its class of agents [12]. In this subcategory there are 3 papers and their percentage is 3%.

All these sub-categories are shown in figure (5) and the numbers of papers in each sub-category is shown in table (4). The number of published paper in this category with respect to the year of publishing is shown in figure (6).

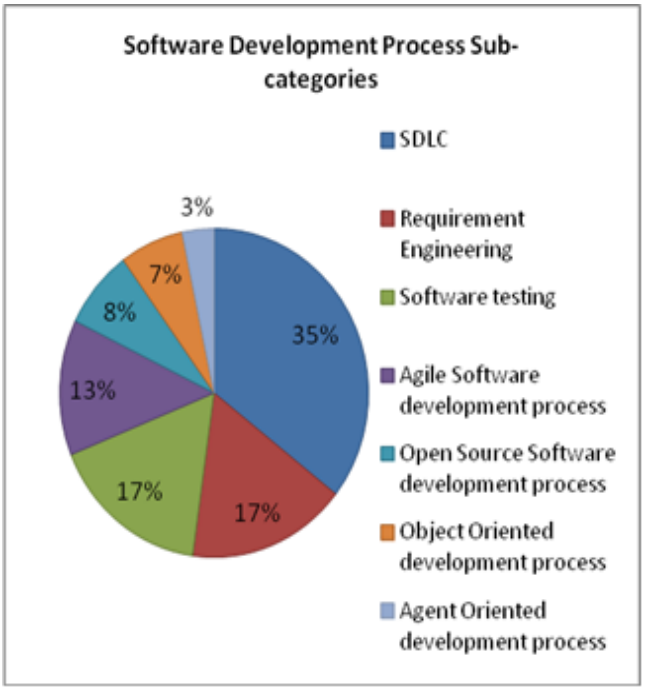

Figure (5): Software Development Process sub-categories

Table (4): Number of papers in each sub-category of Software Development Process

Software Development Process Sub-categories	No. of paper in each sub-category
SDLC	32
Requirement Engineering	15
Software testing	15
Agile Software Engineering	12
Open Source Software Engineering	7
Object Oriented Engineering	6
Agent Oriented Engineering	3

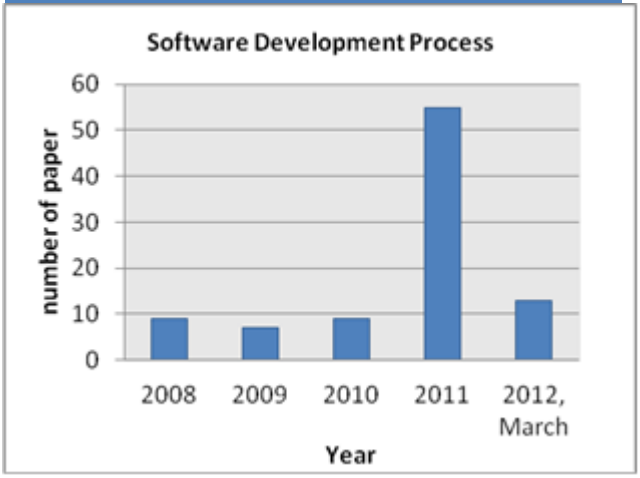

Figure (6): Number of papers published in the software development process based on the year of publishing.

2.2 Software Management

Software project management begins with a set of activities that are collectively called project planning. Before the project can begin, the manager and the software team must estimate the work to be done, the resources that will be required, and the time that will elapse from start to finish. Whenever estimates are made, we look into the future and accept some degree of uncertainty as a matter of course [13].

In this research area there are 75 papers, where are classified into number of sub-categories, brief description of each one is given below:

h. Assessment: Several methods have been created to define an assessment process for software. Some focus on some aspects like the maturity, the durability and the strategy of the organization around the project itself. Other methodologies add functional aspects to the assessment process [13]. In this subcategory there are 18 papers and their percentage is 24%.

i. Quality: In the context of software engineering, software quality refers to two related but distinct notions that exist wherever quality is defined in a business context: Software functional quality reflects how well it complies with or conforms to a given design, based on functional requirements or specifications. That attribute can also be described as the fitness for purpose of a piece of software or how it compares to competitors in the marketplace as a worthwhile product; Software structural quality refers to how it meets non-functional requirements that support the delivery of the functional requirements, such as robustness or maintainability, the degree to which the software was produced correctly [13]. In this subcategory there are 16 papers and their percentage is 21%.

j. Architectural Strategies: As organizations enhance their architecture competency, architects will come to play a more and more active role in the strategy setting process, in addition to the crucial the role they already play in business strategy implementation. However, strategy is not part of the curriculum of the average computer science program, nor is it a competency that organizations necessarily foster in the technical community. In this primer, our goal is to provide you with a strategy for the intelligent introduction to state-of-the-art business strategy thinking and models. Our aim is to help you develop the language, sensibilities, and tools that will enable you to quickly position yourself to make a useful contribution to the strategy process. There more attention to this competency because it is emerging as a critical avenue for competitive distinction. The trend has been for technology to become more essential to basic business capabilities as well as to creating differentiating value propositions that distinguish the business in the marketplace. At the same time, technology is complex and fast moving enough that it requires its own kind of attention, and it falls to the architects to bring this insight to the strategy table. With this foundation in the formulation of the business strategy, architects can also be a more effective bridge between business strategy and the technical strategy that implements it [14]. In this subcategory there are 12 papers and their percentage is 16%.

k. Estimation: Software Estimation is responsive to the widespread problems the software industry has experienced in creating meaningful cost and schedule estimates. Software estimation is not as hard or mysterious as many people think, but the knowledge

of how to create effective estimates has not been well publicized. Software Estimation provides a comprehensive set of tips and heuristics that software developers, technical leads, and project managers can apply to create more accurate estimates. It presents fundamental estimation techniques and addresses specific estimation challenges. It explains how to avoid common pitfalls. Software Estimation doesn't avoid hairy mathematical approaches, but the non-mathematical reader will find plenty of useful guidelines without getting bogged down in complex formulas [15]. In this subcategory there are 11 papers and their percentage is 15%.

l. Team Management: it is refers to techniques, processes and tools for organizing and coordinating a group of individuals working towards a common goal. The 'Team Development Model', identified by Bruce Tuckman, offers a foundational definition of the stages teams go through during their lifecycle. While the activities of team management are not new, many of the tools used by team managers are. The more Organizational Development-oriented practitioners often use interview-based analysis and provide reportage and insights that team leaders and their management may use to adapt team practices for higher performance. Teams can also be developed through team building activities - which can also be used simply to build relationships where team members lack cohesion due to organizational structure or physical distance. Project managers may approach team management with a focus on structure, communications and standardized practices [13]. In this subcategory there are 9 papers and their percentage is 12%.

m. Security: Security is the degree of protection against danger, damage, loss, and crime. Security as a form of protection is structures and processes that provide or improve security as a condition. Security has to be compared to related concepts: safety, continuity, reliability. The key difference between security and reliability is that security must take into account the actions of people attempting to cause destruction [16]. In this subcategory there are 7 papers and their percentage is 9%.

n. Knowledge management: comprises a range of strategies and practices used in an organization to identify, create, represent, distribute, and enable adoption of insights and experiences. Such insights and experiences comprise knowledge, either embodied in individuals or embedded in organizations as processes or practices. Knowledge management efforts typically focus on organizational objectives such as improved performance, competitive advantage, innovation, the sharing of lessons learned, integration and continuous improvement of the organization. KM efforts overlap with organizational learning, and may be distinguished from that by a greater focus on the management of knowledge as a strategic asset and a focus on encouraging the sharing of knowledge [17]. In this subcategory there are 1 papers and their percentage is 1%.

o. Product Management: it is an organizational lifecycle function within a company dealing with the planning or marketing of a product or products at all stages of the product lifecycle. Product management (inbound focused) and product marketing (outbound focused) are different yet complementary efforts with the objective of maximizing sales revenues, market share, and profit margins. The role of product management spans many activities from strategic to tactical and varies based on the organizational structure of the company [18]. In this subcategory there are 1 papers and their percentage is 1%.

p. All these sub-categories are shown in figure (7) and the numbers of papers published in each sub-category is shown in table (5). The number of papers published in this category with respect to the year of publishing is shown in figure (8).

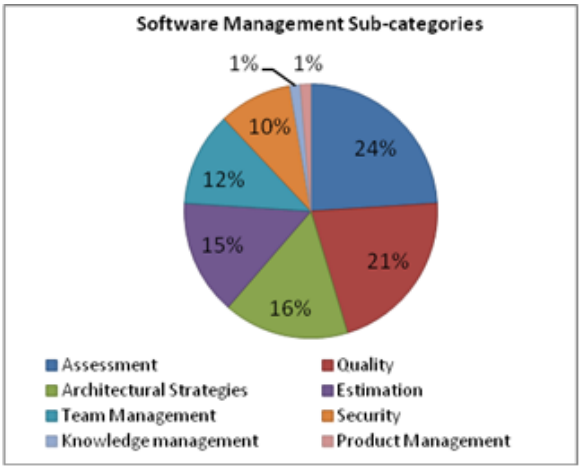

Figure (7): software management sub-categories

Table (5): Number of papers in each sub-category of software management

Software Management Sub-categories	No. of paper in each sub-category
Assessment	18
Quality	16
Architectural Strategies	12
Estimation	11

Team Management	9
Security	7
Knowledge management	1
Product Management	1

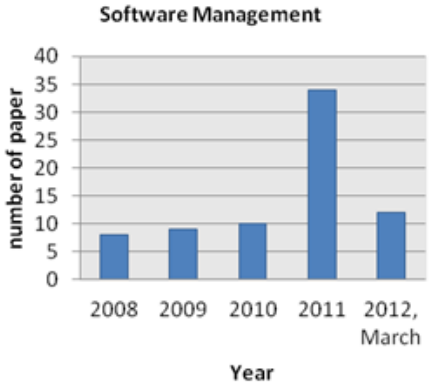

Figure (8): Number of papers published in the software management based on the year of publishing

2.3 Software Re-engineering

The reengineering of software was described by Chikofsky and Cross in their 1990 paper [19], as "The examination and alteration of a system to reconstitute it in a new form". Less formally, reengineering is the modification of a software system that takes place after it has been reverse engineered, generally to add new functionality, or to correct errors. This entire process is often erroneously referred to as reverse engineering; however, it is more accurate to say that reverse engineering is the initial examination of the system, and reengineering is the subsequent modification. Re-engineering is mostly used in the context where a legacy system is involved [20]. Software systems are evolving on high rate because there more research to make the better so therefore software system in most cases, legacy software needs to operate on a new computing platform. 'Re-engineering' is a set of activities that are carried out to re-structure a legacy system to a new system with better functionalities and conform to the hardware and software quality constraint.

In this research area there are many papers 36 papers, these papers are classified into number of sub-categories, these categories as following with brief description for each one of them:

a. Reverse Engineering: it is the process of discovering the technological principles of a device, object, or system through analysis of its structure, function, and operation. It often involves taking something (e.g., a mechanical device, electronic component, software program, or biological, chemical, or organic matter) apart and analyzing its workings in detail to be used in maintenance, or to try to make a new device or program that does the same thing without using or simply duplicating (without understanding) the original [21]. In this sub-category there are 12 papers and their percentage is 33%.

b. Object-Oriented Re-engineering: Object oriented development has proved its worth in today's system because its design and development is better, reliable and easier to access than the traditional methodologies. Due to updated requirements and lack of documentation in old systems has provided a motivation to revamp the systems. Rebuilding or redesigning the same system is highly expensive. To overcome this problem reverse engineering of the system is used as most suitable alternative [22]. In this subcategory there are 3 papers and their percentage is 8%.

c. Service-Oriented Architecture: In software engineering, a Service-Oriented Architecture (SOA) is a set of principles and methodologies for designing and developing software in the form of interoperable services. These services are well-defined business functionalities that are built as software components (discrete pieces of code and/or data structures) that can be reused for different purposes. SOA design principles are used during the phases of systems development and integration [23]. In this subcategory there are 3 papers and their percentage is 8%.

d. Automotive Software: it is different from the kind of software that is usually addressed by current reengineering research. These papers give an overview of the particular conditions and challenges that in the automotive domain [24]. In this subcategory there are 2 papers and their percentage is 6%.

e. Cycle Table: Identifying and understanding cycles in large applications is a challenging task. In these papers CycleTable which displays both direct and indirect references was presented. It shows how minimal cycles are intertwined through shared edges and allows the reengineer to devise simple strategies to remove them [25]. In this subcategory there are 2 papers and their percentage is 6%.

f. Re-engineering Metrics: A metric is nothing more than a standard measure to assess your performance in a particular area. Metrics are at the heart of a good, customer-focused process management system and any program directed at continuous improvement. The focus on customers and performance standards show up in the form of metrics that assess your ability to meet your customers' needs and business

objectives [26]. In this subcategory there are 2 papers and their percentage is 6%.

g. Re-engineering process: it is all the activities from the legacy software analyzing to the generation of new software. In this subcategory there are 2 papers and their percentage is 6%.

h. There are another research in the software re-engineering field, about one paper in each sub-categories, these sub-categories are: Complex Software Specifications, Dependency Structure Matrix, Deployment Strategies for a Reengineered System, Distance study Course, Incremental Process Mining Approach, Moose Open-Source Re-engineering Environment, Reengineering Test Management, Re-engineering TinyOSbased Networked Sensor Systems, Software Product Lines, Architecture Style.

All these sub-categories are shown in figure (9) and the numbers of papers in each sub-category is shown in table (6). The number of papers published in this category with respect to the year of publishing is shown in figure (10).

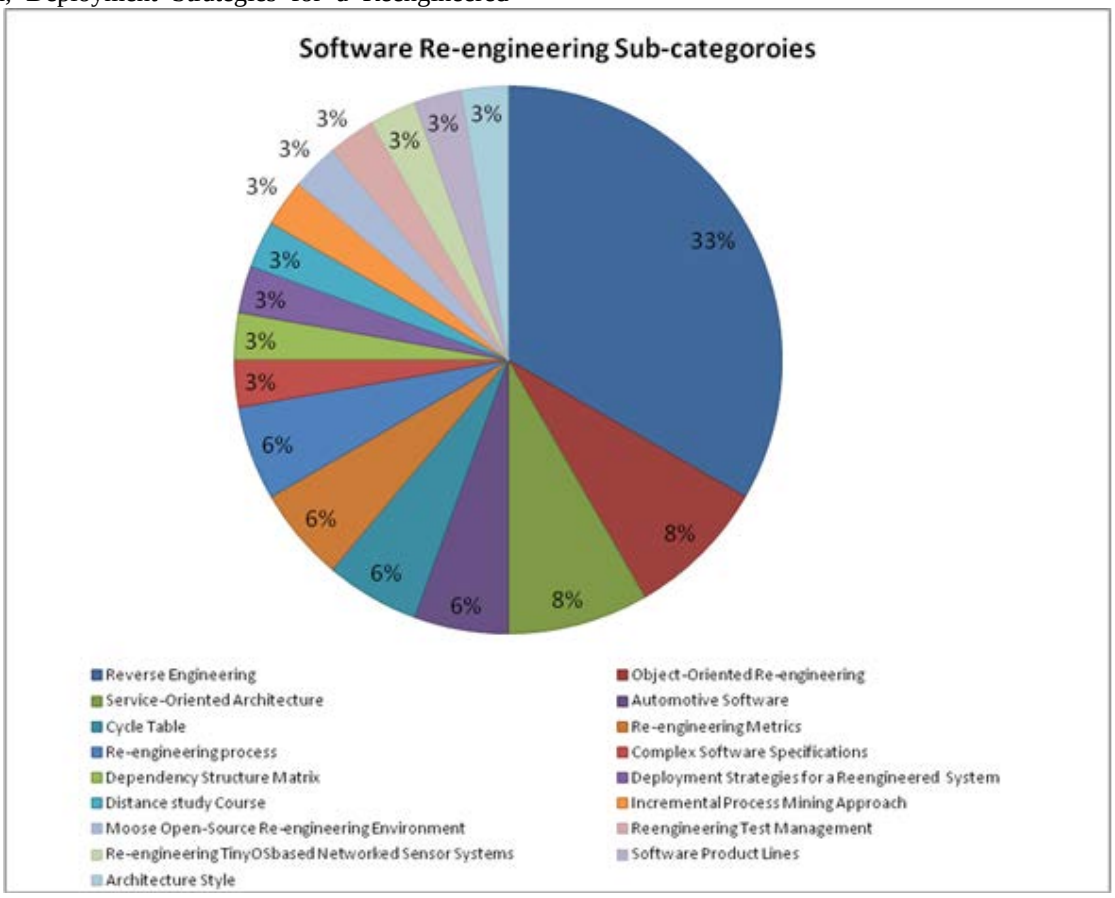

Figure (9): Software Re-engineering sub-categories

Table (6): Number of papers in each sub-category of Software Re-engineering

Software Re-engineering Sub-categories	No. of paper in each sub-category
Reverse Engineering	12
Object-Oriented Re-engineering	3
Service-Oriented Architecture	3
Automotive Software	2
Cycle Table	2
Re-engineering Metrics	2
Re-engineering process	2

Complex Software Specifications	1
Dependency Structure Matrix	1
Deployment Strategies for a Reengineered System	1
Distance study Course	1
Incremental Process Mining Approach	1
Moose Open-Source Re-engineering Environment	1
Reengineering Test Management	1
Re-engineering TinyOSbased Networked Sensor Systems	1
Software Product Lines	1
Architecture Style	1

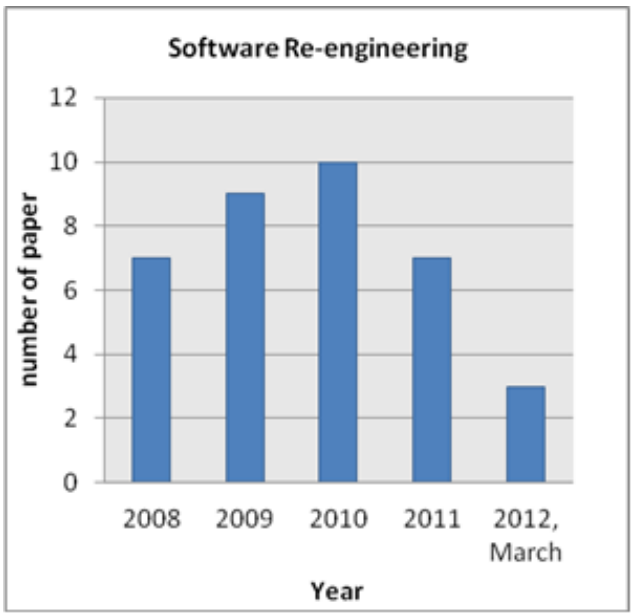

Figure (10): Papers published in the software Re-engineering.

2.4 Software Engineering Techniques

Today, software is becoming ever more complex, developed with a very short time to market, and required to cope with changing requirements. This scenario calls for increased levels of flexibility and agility, both in the technologies used and the processes followed for engineering software. So many techniques were provided that can meet these challenges and these techniques are known as Software engineering techniques.

In this research area there are 33 papers, which are classified into number of sub-categories as following with brief description for each:

a. Component-Based: Component-based software engineering (CBSE) (also known as component-based development (CBD)) is a branch of software engineering that emphasizes the separation of concerns in respect of the wide-ranging functionality available throughout a given software system. It is a reuse-based approach to defining, implementing and composing loosely coupled independent components into systems. This practice aims to bring about an equally wide-ranging degree of benefits in both the short-term and the long-term for the software itself and for organizations that sponsor such software [27]. In this sub-category there are 10 papers and their percentage is 29%.

b. Automated Generation: is a style of computer programming that uses automated source code creation through generic frames, classes, prototypes, templates, aspects, and code generators to improve programmer productivity. It is often related to code-reuse topics such as component-based software engineering and product family engineering [28]. In this sub-category there are 7 papers and their percentage is 20%.

c. Concurrent Engineering: is a work methodology based on the parallelization of tasks (i.e. performing tasks concurrently). It refers to an approach used in product development in which functions of design engineering, manufacturing engineering and other functions are integrated to reduce the elapsed time required to bring a new product to the market [29]. In this sub-category there are 3 papers and their percentage is 9%.

d. Research Methodology: is a way to systematically solve the research problem. lt may be understood as a science of studying how research is done scientifically. There are various steps that are generally adopted by a researcher in studying the research problem along with the logic behind them. lt is necessary for the researcher to know not only the research methods/techniques but also the methodology. Researchers not only need to know how to develop certain indices or tests, how to

calculate the mean. The mode, the median or the standard deviation or chi-square, how to apply particular research techniques, but they also need to know which of these methods or techniques are relevant and which are not, and what would they mean and indicate and why. Researchers also need to understand the assumptions underlying various techniques and they need to know the criteria by which they can decide that certain techniques and procedures will be applicable to certain problems and others will not. All this means that it is necessary for the researcher to design his methodology for his problem as the same may differ from problem to problem [30]. In this sub-category there are 3 papers and their percentage is 9%.

e. Cloud Computing: it is a metaphor used by Technology or IT Services companies for the delivery of computing requirements as a service to a heterogeneous community of end-recipients. The term cloud theoretically signifies abstraction of technology, resources and its location that are very vital in building integrated computing infrastructure (including networks, systems and applications). All Cloud computing models rely heavily on sharing of resources to achieve coherence and economies of scale similar to a utility (like the electricity grid) over a network (typically the Internet) [31]. In this sub-category there are 2 papers and their percentage is 6%.

f. Self-Adaptive System: the self-adaptability of systems has been studied in a wide range of disciplines, from biology to robotics, only recently has the software engineering community recognized its key role in enabling the development of future software systems that are able to self-adapt to changes that may occur in the system, its requirements, or the environment in which it is deployed [32].

g. There are another research in the field of software engineering techniques, about one paper in each sub-categories, these sub-categories are: Feature-Oriented Programming, Integrated Systems Engineering, Model-Based Development, Program Slicing Techniques, Software Intelligence, Software Traceability, and Visual Programming Language

All these sub-categories are shown in figure (11) and the number of papers in each sub-category is shown in table (7). The number of papers published in this category with respect to the year of publishing is shown in figure (12).

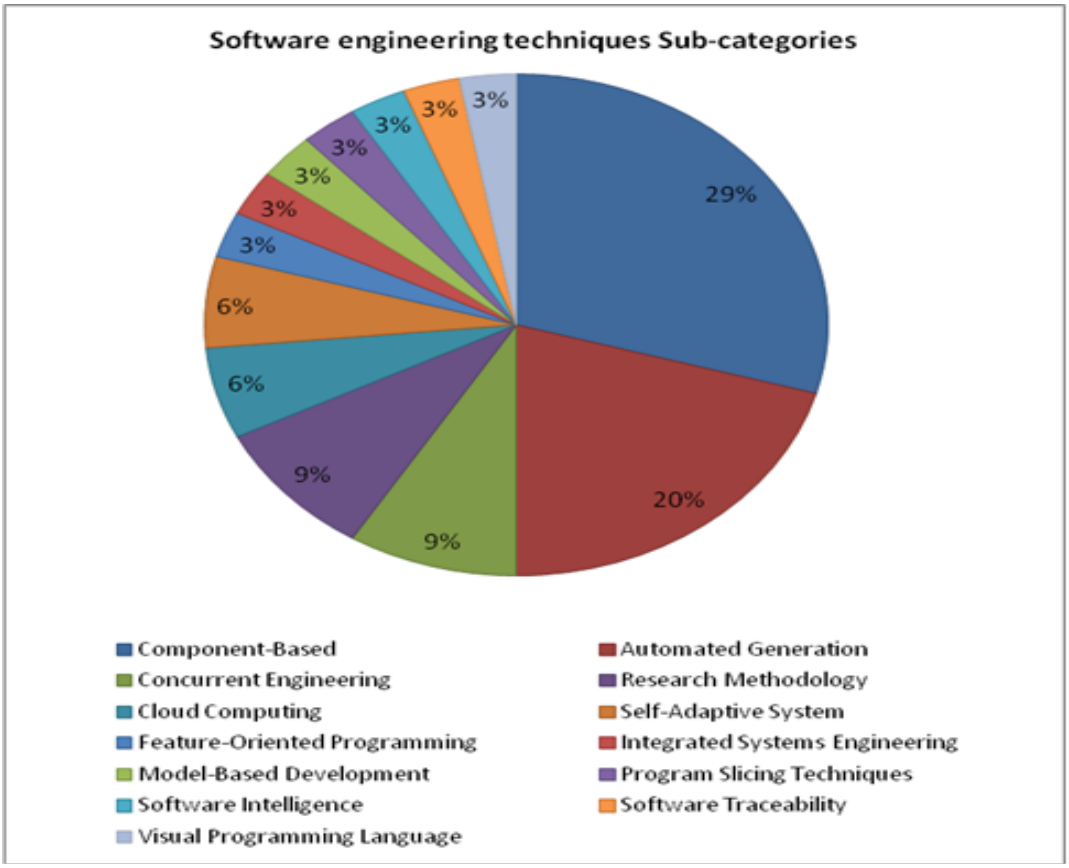

Figure (11): Software engineering techniques sub-categories

Table (7): Number of papers published in each sub-category of Software engineering techniques

Software engineering techniques Sub-categories	No. of paper in each sub-category
Component-Based	10
Automated Generation	7
Concurrent Engineering	3
Research Methodology	3
Cloud Computing	2
Self-Adaptive System	2
Feature-Oriented Programming	1
Integrated Systems Engineering	1
Model-Based Development	1
Program Slicing Techniques	1
Software Intelligence	1
Software Traceability	1
Visual Programming Language	1

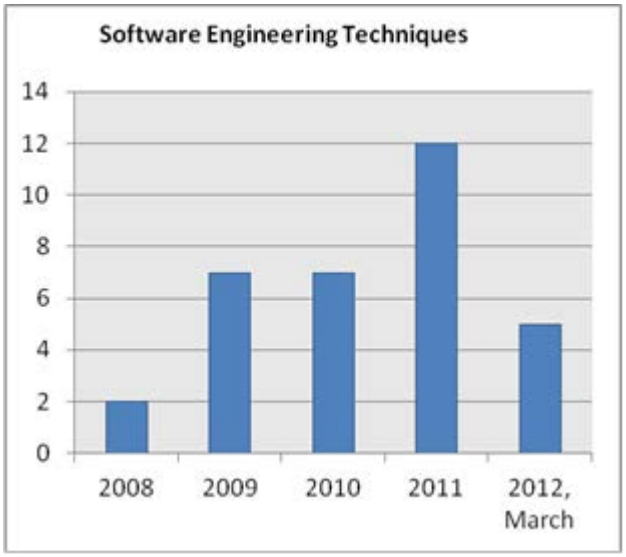

Figure (12): Papers published in the software engineering techniques.

2.5 Software Applications

Also known as an application or an "app", is computer software designed to help the user to perform specific tasks. Examples include enterprise software, accounting software, office suites, and graphics software and media players. Many application programs deal principally with documents. Apps may be bundled with the computer and its system software, or may be published separately. Some users are satisfied with the bundled apps and need never install one. Application software is contrasted with system software and middleware, which manage and integrate a computer's capabilities, but typically do not directly apply in the performance of tasks that benefit the user. The system software serves the application, which in turn serves the user [33].

In this research area there are many papers 19 papers, these papers are classified into number of sub-categories, these categories as following with brief description for each one of them:

a. Web-based application: A web application is an application that is accessed over a network such as the Internet or an intranet. The term may also mean a computer software application that is coded in a browser-supported language (such as JavaScript, combined with a browser-rendered markup language like HTML) and reliant on a common web browser to render the application executable. Web applications are popular due to the ubiquity of web browsers, and the convenience of using a web browser as a client, sometimes called a thin client. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity, as is the inherent support for cross-platform compatibility. Common web applications include webmail, online retail sales, online auctions, wikis and many other functions [34]. In this sub-category their are 12 papers and there percentage is 63%.

b. Mobile – Applications: It is application software that developed for small low-power handheld devices such as personal digital assistants, enterprise digital assistants or mobile phones. These applications are either pre-installed on phones during manufacture, downloaded by customers from various mobile software distribution platforms, or web applications delivered over HTTP which use server-side or client-side processing (e.g. JavaScript) to provide an "application-like" experience within a Web browser [35]. In this sub-category there are 3 papers and their percentage is 16%.

c. Service Oriented Applications: A service-oriented architecture is essentially a collection of services. These services communicate with each other. The communication can involve either simple data passing or it could involve two or more services coordinating some activity. Some means of connecting services to each other is needed. Service-oriented architectures are not a new thing. The first service-oriented architecture for many people in the

past was with the use DCOM or Object Request Brokers (ORBs) based on the CORBA specification [36]. In this sub-category there are 3 papers and their percentage is 16%.

d. 3D Applications: it is refers to programs used to create 3D computer-generated imagery, there are many software like 3D Studio MAX, 3D modeling application, Animation Master, and etc. [37]. In this sub-category there are 1 papers and their percentage is 5%.

These sub-categories are shown in figure (13) and the numbers of papers in each sub-category is shown in table (8). The number of papers published in this category with respect to the year of publishing is shown in figure (14).

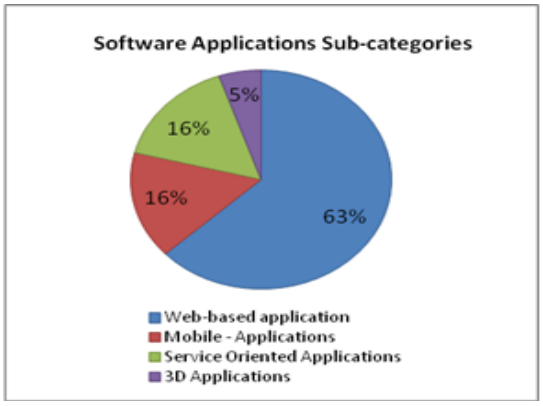

Figure (13): Software engineering techniques sub-categories

Table (8): Numbers of papers published in each sub-category of Software application

Software applications Sub-categories	No. of paper in each sub-category
Web-based application	12
Mobile - Applications	3
Service Oriented Applications	3
3D Applications	1

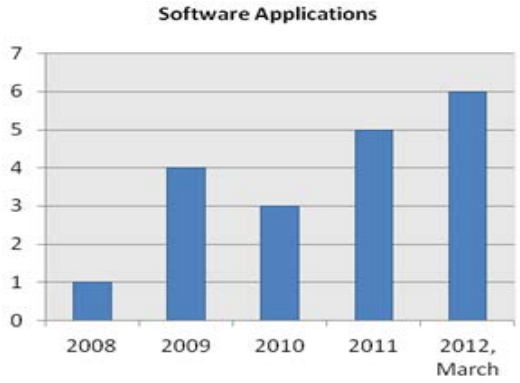

Figure (14): Papers published in the software applications category.

3. THE GAP IN SOFTWARE ENGINEERING

On classification of the researches published in the last five years and analysis of data, it gives a concrete idea that research conducted in software re-engineering, software engineering techniques and software applications is meager while a higher percent of research is carried out in rest of the two areas as shown in figure (2) and table (1). The comparison among the number of papers published in all five categories with respect to the year of publishing is shown in figure (15). From the figure, it is understood that very less research is conducted in the area of software re-engineering in the current year (2012, March), therefore, research gap exists in the area of software re-engineering which means more research is possible in the same.

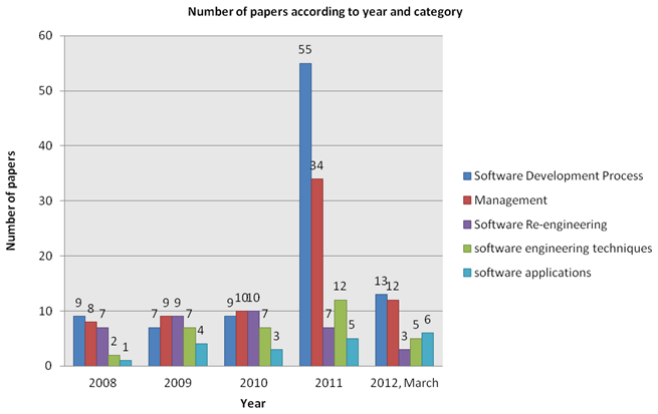

Figure (15): Papers published in each category from the year (2008 - 2012,March).

4. CONCLUSION

On reviewing more than 250 papers related to software engineering published in various reputed journals, the authors have realized that software engineering is a very important research area and it can be classified into various sub-areas like software development process, software management, software re-engineering, software engineering techniques and software applications. Research was carried out to a lesser in all five categories in the year 2008, 2009 and 2010, where as there is substantial rise in research conducted in the year 2011especitialy in the first two categories, which is again followed by a sudden drop in the year 2012 until March. Further, the study also provides an insight on the insufficient research carried out in software re-engineering and hence the authors recommend greater scope in this particular area and that extensive research may be conducted to improve and enhance the software re-engineering processes.

REFERENCES

[1] SWEBOK executive editors, Alain Abran, James W. Moore; editors, Pierre Bourque, Robert Dupuis. (2004). Pierre Bourque and Robert Dupuis. ed. Guide to the Software Engineering Body of Knowledge - 2004 Version. IEEE Computer Society. pp. 1–1. ISBN 0-7695-2330-7. http://www.swebok.org.

[2] ACM (2006). "Computing Degrees & Careers". ACM. http://computingcareers.acm.org/? page_id=12. Retrieved 2010-11-23.

[3] Laplante, Phillip (2007). What Every Engineer Should Know about Software Engineering. Boca Raton: CRC. ISBN 9780849372285. http://books.google.com/?id= pFHYk0KWAEgC&lpg=PP1&dq=What%20Every%20Engineer%20Should%20Know%20about%20Software%20Engineering.&pg=PA1#v=onepage&q&f=false. Retrieved 2011-01-21.

[4] Peter, Naur; Brian Randell (7–11 October 1968). "Software Engineering: Report of a conference sponsored by the NATO Science Committee" (PDF). Garmisch, Germany: Scientific Affairs Division, NATO. http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF. Retrieved 2008-12-26.

[5] Randell, Brian (10 August 2001). "The 1968/69 NATO Software Engineering Reports". Brian Randell's University Homepage. The School of the Computer Sciences, Newcastle University. http://homepages.cs.ncl.ac.uk/brian.randell/NATO/NATOReports/index.html. Retrieved 2008-10-11. "The idea for the first NATO Software Engineering Conference, and in particular that of adopting the then practically unknown term "software engineering" as its (deliberately provocative) title, I believe came originally from Professor Fritz Bauer."

[6] Ralph, P., and Wand, Y. A Proposal for a Formal Definition of the Design Concept. In, Lyytinen, K., Loucopoulos, P., Mylopoulos, J., and Robinson, W., (eds.), Design Requirements Engineering: A Ten-Year Perspective: Springer-Verlag, 2009, pp. 103-136

[7] Wiegers, Karl E. (2003). Software Requirements, Second Edition. Microsoft Press. ISBN 0-7356-1879-8.

[8] Software Testing by Jiantao Pan, Carnegie Mellon University.

[9] Beck, Kent; et al. (2001). "Manifesto for Agile Software Development". Agile Alliance. Retrieved 14 June 2010.

[10] Verts, William T. (2008-01-13). "Open source software". World Book Online Reference Center.

[11] McCarthy, J.; Brayton, R.; Edwards, D.; Fox, P.; Hodes, L.; Luckham, D.; Maling, K.; Park, D. et al (March 1960). LISP I Programmers Manual. Boston, Massachusetts: Artificial Intelligence Group, M.I.T. Computation Center and Research Laboratory. p. 88f. "In the local M.I.T. patois, association lists [of atomic symbols] are also referred to as "property lists", and atomic symbols are sometimes called "objects".

[12] Shoham, Y. (1990) (Technical Report STAN-CS-90-1335). Agent-Oriented Programming. Stanford University: Computer Science Department.

[13] Pressman, Scott (2005), Software Engineering: A Practitioner's Approach (Sixth, International ed.), McGraw-Hill Education Pressman, p. 388.

[14] Bredemeyer, Dana and Ruth Malan, "Architecture Strategy Choreographs the Dance of Change", slides from the presentation to be given at the Enterprise Architecture Conference in Phoenix, AZ in November, 2003.

[15] Steve McConnell's instantly indispensable Software Estimation: book, 2006.

[16] Bruce Schneier, Beyond Fear: Thinking Sensibly about Security in an Uncertain World, Copernicus Books, pages 26-27

[17] Addicott, Rachael; McGivern, Gerry; Ferlie, Ewan (2006). "Networks, Organizational Learning and Knowledge Management: NHS Cancer Networks". Public Money & Management 26 (2): 87–94. doi:10.1111/j.1467-9302.2006.00506.x.

[18] Rajeev Misra, 2009, “Software Product Management”, InfoPro Worldwide, Inc. 12 Perrine Road, Suite # 104 Monmouth Jn., NJ 08852, USA, Tel: 1 (609) 606 9020, Fax: 1 (609) 452 0335, Web site : www.infoproworldwide.com

[19] Chikofsky, E. and Cross, J., 1990. Reverse Engineering and Design Recovery: A Taxonomy. IEEE Software, 7(1):13-18.

[20] Asit Kumar Gahalaut et. al. / International Journal of Engineering Science and Technology Vol. 2(06), 2010, 2296-2303

[21] Eilam, Eldad & Chikofsky, Elliot J. (2007). Reversing: secrets of reverse engineering. John Wiley & Sons. p. 3. ISBN 9780764574818. http://books.google.com/books?id=K8vIf99UBYQC&pg=PA3.

[22] V. Carchiolo and et. Al, "Object-oriented re-engineering of manufacturing models: A case study", Springer Science, Business Media, LLC, paper, 2010

[23] Bell, Michael (2008). "Introduction to Service-Oriented Modeling". Service-Oriented Modeling: Service Analysis, Design, and Architecture. Wiley & Sons. pp. 3. ISBN 978-0-470-14111-3.

[24] V. S. Coerne and et al, "Automotive Software: Characteristics and Reengineering Challenges", Corporate Sector Research and Advance Engineering Software, P.O. Box 30 02 40, 70442 Stuttgart, Germany, paper, 2010.

[25] J. Laval and et. al, "Identifying cycle causes with CycleTable", FAMOOSr, paper, 11-Oct-09

[26] J. W. Wesner et al, Winning with Quality, Addison-Wesley, Reading, MA, 1995

[27] Wallace, Bruce (May 19, 2010). "A hole for every component, and every component in its hole". Existential Programming. "There is no such thing as a Component".

[28] James Wilcox, "Paying Too Much for Custom Application Development", March 2011.

[29] Ma, Y., Chen, G. & Thimm, G.; "Paradigm Shift: Unified and Associative Feature-based Concurrent Engineering and Collaborative Engineering", Journal of Intelligent Manufacturing, DOI 10.1007/s10845-008-0128-y

[30] C. R. Kothari, "Research Methodology: Methods and Techniques", book, second edition, 2004.

[31] "The NIST Definition of Cloud Computing". National Institute of Science and Technology.http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf. Retrieved 24 July 2011.

[32] Cheng, and et. al., "Software Engineering for Self-Adaptive Systems", book, ISBN 978-3-642-02160-2.

[33] Ceruzzi, Paul E. (2000). A History of Modern Computing. Cambridge, Mass.: MIT Press. ISBN 0-262-03255-4.

[34] What is a Web Application? by Daniel Nations, former About.com Guide.

[35] Mobile Application Development, http://www.webmapsolutions.com/mobile-arcgis-paper-gps-data-collection

[36] Thomas Erl, "Service-Oriented Architecture: Concepts, Technology, and Design", book, Kindle edition,

[37] "About Aerospace Coordinate Systems". Retrieved 2007-11-23.

[38] Eric PLATON' and Yuichi SEP, "Security software engineering in wireless sensor networks", National Institute of Informatics, The University of Tokyo, 2008.

[39] Simon Denier, and et. al, , "Reverse Engineering the Literature on Design Patterns and Reverse-Engineering", PTIDEJ Team DIRO, Universite de Montreal and DGIGL, Ecole Polytechnique de Montreal, 2008

[40] William M. Mongan, and et al, "Re-engineering a Reverse Engineering Portal to a Distributed SOA", Drexel University Department of Computer Science, 3141 Chestnut Street, Philadelphia, PA 19104, fwmm24, ms333, spirosg@drexel.edu, 2008.

[41] Samuel Mann Lesley Smith, "Biomimicry as a super systems metaphor for software engineering?", Otago Polytechnic Dunedin, NZ, smann, lsmith @tekotago.ac.nz, 2008.

[42] MEHRDAD NOJOUMIAN and TIMOTHY C. LETHBRIDGE, "Reengineering PDF-Based Documents Targeting Complex Software Specifications", University of Waterloo, Canada, University of Ottawa, Canada, 2008.

[43] Betty H.C. Cheng, and et. al., "Software Engineering for Self-Adaptive Systems", Dagstuhl Seminar, 2008.

[44] Alexandre Bergel, and et. al, "Enhanced Dependency Structure Matrix for Moose",Accepted to FamoosR, 2008

[45] Craig Niiyama, and et. al, "Service-Oriented Software Reengineering Methodology for Composite Services", Computing & Software Systems, Institute of Technology, University of Washington, Tacoma, {cniiyama, chungsa, dchinn}@u.washington.edu, Milgard School of Business, University of Washington, Tacoma, sergiod@u.washington.edu, 2008

[46] Brian R. von Konsky and Jim Ivins, "Assessing the Capability and Maturity of Capstone Software

Engineering Projects", Australian Computer Society, Tenth Australasian Computing Education Conference (ACE2008), Wollongong, Australia, January 2008. Conferences in Research and Practice in Information Technology, Vol. 78. Simon and Margaret Hamilton Digital Ecosystems for Business Intelligence Institute Curtin (1)University of Technology GPO Box U1987 Perth, Western Australia 6845 B.vonKonsky@acsmail.net.au, 2008.

[47] Steve M. Easterbrook and Timothy C. Johns, "Engineering the Software for Understanding Climate Change", Dept of Computer Science, U. of Toronto, Canada, sme@cs.toronto.edu and Met Office Hadley Centre, United Kingdom, tim.johns@metoffice.gov.uk, 2008

[48] Jurgen Ebert Volker Riediger. and Andreas Winter., "Graph Technology in Reverse Engineering, The TGraph Approach", University of Koblenz-Landau, Institute for Software Technology, Universit¨atsstr. 1. 56070 Koblenz, Germany ebert,riediger@uni-koblenz.de, and Johannes-Gutenberg-University Mainz, Institute for Computer Science, Staudingerweg 9. 55128 Mainz, Germany, winter@uni-mainz.de, 2008.

[49] Vigdis By Kampenes and et. al, "Flexibility in Research Designs in Empirical Software Engineering", Department of Software Engineering, Simula Research Laboratory, P.O. Box 134, NO-1325 Lysaker, Norway, Department of Informatics, University of Oslo, P.O. Box 1080 Blindern, NO-0316 Oslo, NorwaySINTEF ICT, NO-7465 Trondheim, Norway, 2008.

[50] Kai Petersen, and et. Al.,"Systematic Mapping Studies in Software Engineering", School of Engineering, Blekinge Institute of Technology, Box 520 SE-372 25 Ronneby. (kai.petersen | robert.feldt | shahid.mujtaba | michael.mattsson)@bth.se, and Ericsson AB, Box 518. SE-371 23 Karlskrona. (kai.petersen | shahid.mujtaba)@ericsson.com, 2008.

[51] Lina Tankeleviciene, "using reengineering techniques for distance study course improvement", International Book Series "Information Science and Computing, The paper is selected from Third International Conference "Modern (e-) Learning" MeL 2008, Varna, Bulgaria, June-July 2008.

[52] Daniel Halperin, and et. al, "Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses", IEEE Symposium on Security and Privacy, 2008.

[53] Niklaus Wirth, "A Brief History of Software Engineering", IEEE, 2008

[54] Taehyung (George) Wang, and et. al., "assessing student learning in software engineering", Consortium for Computing Sciences in Colleges CCSC, 2008.

[55] John Karn, "An Ethnographic Study of Conflict in Software Engineering Teams", Journal of Information, Information Technology, and Organizations, 2008.

[56] Jannik Laval, and et. al., "Assessing the Quality of your Software with MoQam", Accepted to FamoosR, 2008.

[57] Chris Jesshope and Alex Shafarenko,, "Concurrency Engineering", Institute for Informatics, University of Amsterdam and Department of Computer Science, University of Hertfordshire, 2008.

[58] WH Morkel Theunissen, "Software Engineering Professionalism", paper, 2008.

[59] Eelco Visser,"WebDSL: A Case Study in Domain-Specific Language Engineering", Software Engineering Research Group Delft University of Technology, 2008.

[60] Supannika Koolmanojwong,, nd et. al., "Incremental Commitment Model Process Guidelines for Software Engineering Class", Center of Systems and Software Engineering University of Souther California Los Angeles, CA 90089-0781 {koolmano, aroonvat, icharoen}@usc.edu, 2008.

[61] Karthik Viswanathan, "Right Engineering SaaS: Successfully deploying Software-as-a-Service Models", Aspire Systems - Transforming Product Development E-mail :info@aspiresys.com, Web: www.aspiresys.com, 2008.

[62] Ujjal Marjit and Sibsankar Jana, "Mashups: An Emerging Content Aggregation Web2.0 Paradigm", 7th International CALIBER-2009, Pondicherry University, Puducherry, February 25-27, 2009, © INFLIBNET Centre, Ahmedabad

[63] nurul naslia khairuddin and khairuddin hashim, "Application of Bloom's Taxonomy in Software Engineering Assessments", Proceedings of the 8th WSEAS International Conference on applied computer science (ACS'08), 2008.

[64] S.K.Mishra, and et. al., "Creating Reusable Software Component from Object-Oriented Legacy System

through Reverse Engineering", Journal of object technology, Online at http://www.jot.fm., 2009.

[65] Christian Bartelt,and et. al., "Orchestration of Global Software Engineering Projects", Technische Universität Clausthal, Software Systems Engineering, Julius-Albert-Str. 4, 38678 Clausthal-Zellerfeld, Germany {bartelt | rausch}@tu-clausthal.de, 2009.

[66] Alexandre Bergel, and et. al, "SQUALE – Software QUALity Enhancement",RMoD Team, INRIA, Lille, France, Qualixo, Paris, France, and Labo IA, University of Paris 8, France http://www.squale.org, 2009.

[67] Jesper Andersson, "Software Engineering for Self-Adaptive Systems", Springer-Verlag Berlin Heidelberg, 2009

[68] Christopher Poile,, and et. al., " Coordination in Large-Scale Software Development: Helpful and Unhelpful Behaviors", Dept. of Management Sciences University of Waterloo Waterloo, ON, Canada cpoile@uwaterloo.ca and Microsoft Research Redmond, WA, fandrew. begel, naching@microsoft.com. and Dept. of Computer Science, North Carolina State University, Raleigh, NC, lucas.layman@ncsu.edu, 2009.

[69] Massimo Cossentino and et. al., "ASPECS: an Agent-oriented Software Process for Engineering Complex Systems", Multiagent Systems Group, System and Transport Laboratory, University of Technology of Belfort Montb´eliard, 90010 Belfort cedex, France. e-mail: fmassimo.cossentino, nicolas.gaud, vincent.hilaire, stephane.galland, abder.koukamg@utbm.fr, http://set.utbm.fr. and ICAR Institute, National Research Council, Palermo, Italy. http://www.pa.icar.cnr.it/cossentino, 2009.

[70] Fernando Olivero,, and et. al., "Lumi`ere: An Infrastructure for Producing 3D Applications in Smalltalk", FAMOOSr 2009 3rd Workshop on FAMIX and MOOSE in Software Reengineering, 2009.

[71] Alberto Bacchelli, and et. al., "Miler – A Tool Infrastructure to Analyze Mailing Lists", FAMOOSr 2009 3rd Workshop on FAMIX and MOOSE in Software Reengineering, 2009.

[72] Daniel Ratiu, "Challenges for Domain Knowledge Driven Program Analysis", FAMOOSr 2009, 3rd Workshop on FAMIX and MOOSE in Software Reengineering, 2009.

[73] Fabrizio Perin, "Driving the refactoring of Java Enterprise Applications by evaluating the distance between application elements", FAMOOSr 2009, 3rd Workshop on FAMIX and MOOSE in Software Reengineering, 2009.

[74] David R¨othlisberger, "Why and How to Substantiate the Good of our Reverse Engineering Tools?", FAMOOSr 2009, 3rd Workshop on FAMIX and MOOSE in Software Reengineering, 2009.

[75] Gordana Dodig-Crnkovic, Robert Feldt, "Professional and Ethical Issues of Software Engineering Curricula", Experiences from a Swedish Academic Context, 2009.

[76] Mahmoud O. Elish , and et. al., "Application of TreeNet in Predicting Object-Oriented Software Maintainability: A Comparative Study", In Proceedings of the 13th IEEE European Conference on Software Maintenance and Reengineering (CSMR'09), pp. 69-77, Germany, Mar. 2009.

[77] Robert Feldt, and et. al., "Generic Skills in Software Engineering Master Thesis Projects: Towards Rubric-Based Evaluation", 978-0-7695-3539-5/09 $25.00 © 2009 IEEE
DOI 10.1109/CSEET.2009.54, 2009.

[78] Dag Sjoberg, "Guidelines for conducting and reporting case study research in software engineering", Empir Software Eng (2009) 14:131–164, DOI 10.1007/s10664-008-9102-8 this article is published with open access at Springerlink.com, 2009.

[79] Joseph Chao and Jennifer Brown, "Cross-Departmental Collaboration for the Community: Technical Communicators in a Service-Learning Software Engineering Course", Issues in Informing Science and Information Technology, Volume 6, 2009.

[80] Sita Ramakrishnan, "Innovation and Scaling up Agile Software Engineering Projects", Issues in Informing Science and Information Technology, Volume 6, 2009.

[81] Mamta Garg, and et. Al., "Reverse Engineering – Roadmap to Effective software Design", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, May 2009.

[82] Engr.Farah Naaz Raza, "Artificial Intelligence Techniques in Software Engineering (AITSE)", Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 Vol I, IMECS 2009, March 18 - 20, 2009, Hong Kong.

[83] Rajeev Misra, "Software Product Management", InfoPro Worldwide, Inc. 12 Perrine Road, Suite # 104, Monmouth Jn., NJ 08852, USA, Tel: 1 (609) 606 9020, Fax: 1 (609) 452 0335, Web site : www.infoproworldwide.com, 25th August, 2009

[84] Nenad Stankovic and Tammam Tillo, "Concurrent Software Engineering Project', Journal of Information Technology Education: Innovations in Practice, Volume 8, 2009

[85] Jannik Laval, and et. al., "Identifying cycle causes with Enriched Dependency Structural Matrix", Accepted to WCRE, 2009.

[86] Radu Calinescu and Marta, "Software Engineering Techniques for the Development of Systems of Systems", Computing Laboratory, University of Oxford Wolfson Building, Parks Road, Oxford OX1 3QD, UK, 2009.

[87] Nina Marwede, and et. al., "Automatic Failure Diagnosis Support in Distributed Large-Scale Software Systems based on Timing Behavior Anomaly Correlation ", 13th European, Conference on Software Maintenance and Reengineering (CSMR 2009), pages 47{57. IEEE}.

[88] TERESA F. LUNT, and et. al., "The SeaView Security Model", IEEE transactions on software engineering, VOL. 16, NO. 6, 2009.

[89] Stefan BIFFL and Alexander schatten, "A Platform for Service-Oriented Integration of Software Engineering Environments", Institute for Software Technology and Interactive Systems (ISIS) Technische Universität Wien, A-1040 Vienna, Austria, 2009.

[90] David A Barnhart, and et. al., "SPIDR: Integrated Systems Engineering Design-to-Simulation Software for Satellite Build", 7th Annual Conference on Systems Engineering Research 2009 (CSER 2009) Loughborough University – 20th - 23rd April 2009.

[91] Yajing Zhao, and et. al., "Ontology Classification for Semantic-Web-Based Software Engineering", IEEE transactions on services computing, VOL. 2, NO. 4, October - December 2009.

[92] Jasmine K.S and Dr. R. Vasantha, "Reengineering Test Management for Increasing Testing Effectiveness in Component Based Enterprise Application", International Journal of Computer Theory and Engineering, Vol. 1, No. 4, October2009, 1793-8201.

[93] Vincent Schulte-Coerne, and et. al., "Automotive Software: Characteristics and Reengineering Challenges", Corporate Sector Research and Advance Engineering Software P.O. Box 30 02 40, 70442 Stuttgart, Germany, 2010.

[94] Martin Wirsing and et. al., "SENSORIA Patterns: Augmenting Service Engineering with Formal Analysis, Transformation and Dynamicity", This work has been partially sponsored by the project SENSORIA, IST-2 005-016004, 2010.

[95] Nicholas T. Pilkington, "ESIDE: An Integrated Development Environment for Component-Based Embedded Systems", Department of Computer Science, Portland State University Portland, OR 97207, USA, 2010.

[96] asit kumar gahalaut and padmavati khandnor, " reverse engineering: an essence for software re-engineering and program analysis.", international journal of engineering science and technology, vol. 2(06), 2010, 2296-2303.

[97] Bharat Chhabra, et. al., "Software Engineering Issues from the Cloud Application Perspective", International Journal of Information Technology and Knowledge Management, July-December 2010, Volume 2, No. 2, pp. 669-673.

[98] J. P. McDermott, "Fine-Grained Inspection for Higher-Assurance Software Security in Open Source", Proceedings of the 43rd Hawaii International Conference on System Sciences - 2010.

[99] Satwinder Singh, et. al., "Re-engineering To Analyze and Measure Object Oriented Paradigms", IEEE, 2010.

[100] André Cristiano Kalsing, and et. al., "An Incremental Process Mining Approach to Extract Knowledge from, Legacy Systems ", 2010 14th IEEE International Enterprise Distributed Object Computing Conference.

[101] Nicolas Anquetil, and et. al., "Legacy Software Restructuring: Analyzing a Concrete Case", INRIA Lille-Nord-Europe, France, 2010.

[102] Grace Lewis, " Basics About Cloud Computing", Software Engineering Institute, Carnegie Mellon University, 4500 Fifth Avenue, Pittsburgh, PA 15213-2612, Phone: 412-268-5800, Toll-free: 1-888-201-4479, www.sei.cmu.edu, 2010.

[103] Todd W. Schiller and Michael D. Ernst, "Rethinking the Economics of Software Engineering", FoSER

2010, November 7–8, 2010, Santa Fe, New Mexico, USA.
Copyright 2010 ACM 978-1-4503-0427-6/10/11.

[104] Ahmed E. Hassan and Tao Xie, "Software Intelligence: The Future of Mining Software Engineering Data", School of Computing, Queen's University, Kingston, ON, Canada, Department of Computer Science, North Carolina State University, Raleigh, NC, USA, 2010.

[105] V. Carchiolo, D72and et. al., "Object-oriented re-engineering of manufacturing models: A case study", Springer Science + Business Media, LLC, 2010.

[106] Manar Abu Talib, and et. al., "Secure Software Engineering: A New Teaching Perspective Based on the SWEBOK", Interdisciplinary Journal of Information, Knowledge, and Management Volume 5, 2010.

[107] Philippe Dugerdil, and et. al., "Automatic Generation of Abstract Views for Legacy Software Comprehension", ISEC '10. February 25-27. 2010. Mysore. India.

[108] Claude Y. Laporte, and et. al., "A Software Engineering Lifecycle Standard for Very Small Enterprises", École de technologie supérieure, Montréal, Canada, Centre d'Excellence en Technologies de l'Information et de la Communication, Charleroi, Belgium, and School of Computing, Dublin City University, Dublin, Ireland, 2010.

[109] Jannik Laval, and et. al., "Supporting Simultaneous Versions for Software Evolution Assessment", Elsevier, 2010.

[110] Ira Monarch, and et. al., "The Role of Informatics in Software Engineering: Literature Reviews, Agenda and Software Informatics', Software Engineering Institute 4500 Fifth Avenue, 2010.

[111] Victor Pankratius, "Software Engineering in the Era of Parallelism", Multicore Software Engineering Young Investigator Group, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany, 2010.

[112] Mark Harman, "The Relationship between Search Based Software Engineering and Predictive Modeling", University College London, Department of Computer Science, CREST Centre, Malet Place, London, WC1E 6BT, UK., 2010.

[113] Bruce Powel Douglass and Mats Gothe, "IBM Rational Workbench for Systems and Software Engineering", Copyright IBM Corp. 2010. All rights reserved. ibm.com/redbooks

[114] A. Begel and N. Nagappan, "Pair Programming", Microsoft Research, published in ESEM 2008.

[115] Anthony I. Wasserman, and et. al., "Software Engineering Issues for Mobile Application Development", FoSER 2010, Santa Fe, New Mexico, USA. Copyright 2010 ACM 978-1-4503-0427-6/10/11.

[116] Matthias Galster, "Life-cycle Assessment in Software Engineering", mgalster@ucalgary.ca, 2010.

[117] N.Sasirekha and et. al., "program slicing techniques and its applications", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.3, 2011.

[118] Csaba Nagy, and et. al., "Solutions for Reverse Engineering 4GL Applications, Recovering the Design of a Logistical Wholesale System", 2011 15th European Conference on Software Maintenance and Reengineering, IEEE computer Society.

[119] Jörg Barthold and et. al., "Re-Engineering of a Hierarchical Product Line", 2011 15th International Software Product Line Conference, IEEE computer Society.

[120] Jorge J. Gomez-Sanz, and et. al., "Understanding Agent Oriented Software Engineering Methodologies", GRASIA Research Group, Universidad Complutense de Madrid, Avda. Complutense, 28040 Madrid, Spain, 2011.

[121] Nishant Dubey, "Paper Presentation on Software Development Automation by Computer Aided Software Engineering (CASE)", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 1, January 2011, ISSN (Online): 1694-0814, www.IJCSI.org

[122] Karine Mordal-Manet, and et. al., "An empirical model for continuous and weighted metric aggregation", LIASD, University of Paris 8, France, RMoD Team, INRIA, Lille, France, Qualixo, Paris, France, Air France-KLM, France, and McCabe Software, US

[123] Ashok Kumar Behera, and et. al., "Analysis and Design of Software Visualization Tool for the Behavior of Object Oriented Programming",

International Journal of Computer Applications (0975 – 8887) Volume 13– No.1, January 2011

[124] Jannik Laval and et. al., "Cycles Assessment with CycleTable", RMoD Team INRIA - Lille Nord Europe - USTL - CNRS UMR 8022 Lille, France, 2011.

[125] Umesh Banodha and Kanak Saxena, "A Software Architecture Style for Medical Process Re-engineering", Proceedings of the World Congress on Engineering and Computer Science 2011 Vol I, WCECS 2011, October 19-21, 2011, San Francisco, USA

[126] Fernando Almeida and et. al., "open standards and open source: enabling interoperability", international Journal of Software Engineering & Applications (IJSEA), Vol.2, No.1, January 2011.

[127] Zuhal Tanrikulu and Tuna Ozce, "standardization of information systems development processes and banking industry adaptations", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.2, April 2011

[128] Bradley Deken and Chris Cowen, "Development of Computer Aided Learning Software for Use in Electric Circuit Analysis", International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies, 2011.

[129] Rahul Thakurta, and Subhajit Dasgupta, "impact of software requirement volatility pattern on project dynamics: evidences from a case study", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.3, July 2011

[130] Hollie Boudreaux and et. al., "an algorithmic and software engineering based approach to robust video game design", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.3, July 2011, DOI : 10.5121/ijsea.2011.2303

[131] Kardile Vilas Vasantrao, "Understand Need "Uncertainty Analysis" in the System Development Modeling Process", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.3, July 2011, DOI: 10.5121/ijsea.2011.2307.

[132] Md. Anawarul Kabir and et. al., "unified software development model for free/open source software", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.3, July 2011, DOI :10.5121/ijsea.2011.2308.

[133] Sk.MD.Rafi and Shaheda Akthar, "Discrete Software Reliability Growth Models with Discrete Test Effort Functions", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011, DOI : 10.5121/ijsea.2011.2402.

[134] Pritha Guhaand, and et. al., "incorporating agile with mda case study: online polling system", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011, DOI : 10.5121/ijsea.2011.2408.

[135] S. Arun Kumar and T.Arun Kumar, "study the impact of requirements management characteristics in global software development projects: an ontology based approach", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011, DOI : 10.5121/ijsea.2011.2410.

[136] Mahmoud Khraiwesh, "Validation Measures in CMMI", World of Computer Science and Information Technology Journal (WCSIT), ISSN: 2221-0741, Vol. 1, No. 2, 26-33, 2011

[137] Bimal Aklesh Kumar, "software architecture for fiji national university campus information systems", Bimal Aklesh Kumar / Indian Journal of Computer Science and Engineering (IJCSE), ISSN : 0976-5166 Vol. 2 No. 2 Apr-May 2011.

[138] Praveen Ranjan Srivastava and Km Baby, "Automated Software Testing Using Metahurestic Technique Based on An Ant Colony Optimization", This paper appears in: Electronic System Design (ISED), 2010, International Symposium on, Issue Date: 20-22 Dec. 2010, On page(s): 235 - 240, Location: Bhubaneswar, Print ISBN: 978-1-4244-8979-4, INSPEC Accession Number: 11835766, Digital Object Identifier: 10.1109/ISED.2010.52.

[139] Ardhendu Mandal, "BRIDGE: A Model for Modern Software Development Process to Cater the Present Software Crisis", 2009 IEEE International Advance Computing Conference (IACC 2009), Patiala, India, 6–7March 2009.

[140] Ardhendu Mandal, "srs builder 1.0: An Upper Type CASE Tool for Requirement Specification.", Department Of Computer Science and Application, University of North Bengal (N.B.U.), INDIA, 2011.

[141] Kardile Vilas Vasantrao, "enhance accuracy in software cost and schedule estimation by using "uncertainty analysis and assessment" in the system modeling process", International Journal of Research & Innovation in Computer Engineering, Vol 1, Issue

1, (6-18), August' 2011, www. ijrice .com `© Innovation Science Publications.

[142] Badr Elmir, and et. al., "A Novel Approach for Periodic Assessment of Business Process Interoperability", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011, ISSN (Online): 1694-0814, www.IJCSI.org.

[143] Carlos Alberto Fernández and Martín José, "Towards the integration of formal specification in the Áncora methodology", Instituto de Computación, Universidad Tecnológica de la Mixteca, carretera a Acatlima Km. 2.5 Huajuapan de León, Oax., México C.P. 69000, and División de Estudios de Posgrado, Universidad Tecnológica de la Mixteca, carretera a Acatlima Km. 2.5 Huajuapan de León, Oax., México C.P. 69000,

[144] Asif Irshad Khan and et. al., "A Comprehensive Study of Commonly Practiced Heavy and Light Weight Software Methodologies", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 2, July 2011, ISSN (Online): 1694-0814, www.IJCSI.org.

[145] S.Malathi and Dr.S.Sridhar, "A Classical Fuzzy Approach for Software Effort Estimation on Machine Learning Technique", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 6, No 1, November 2011. ISSN (Online): 1694-0814, www.IJCSI.org, 2011.

[146] Phuc V. Nguyen, "the study and approach of software re-engineering", Department of Cipher & Information Technology, Ho Chi Minh City's Vietnamese Communist Party Committee, Ho Chi Minh City, Vietnam, 2011

[147] Aminat Showole, and et. al., "Layered Approach to Open Source Software Development Success", IBIMA Publishing Communications of the IBIMA, http://www.ibimapublishing.com/journals/CIBIMA/cibima.html, Vol. 2011 (2011), Article ID 160480, 22 pages, DOI: 10.5171/2011.160480

[148] Yasushi Sato,, "An Inconspicuous Giant: NTT's Role in the Development of Software Engineering in Japan", IEEE Annals of the History of Computing, Published by the IEEE Computer Society, 1058-6180/11/$26.00 @ 2011 IEEE

[149] Smriti Srivastava, and et. al., "A Laboratory Testbed for Embedded Fuzzy Control", IEEE TRANSACTIONS ON EDUCATION, VOL. 54, NO. 1, FEBRUARY 2011, 0018-9359/$26.00 © 2010 IEEE

[150] Xavier Amatriain, "Frameworks Generate Domain-Specific Languages: A Case Study in the Multimedia Domain", IEEE transactions on software engineering, VOL. 37, NO. 4, JULY/AUGUST 2011, 0098-5589/11/$26.00 @2011 IEEE, Published by the IEEE Computer Society

[151] Pratyusa K., and et. al., "An Attack Surface Metric", IEEE transactions on software engineering, VOL. 37, NO. 3, MAY/JUNE 2011, 0098-5589/11/$26.00@2011 IEEE, Published by the IEEE Computer Society

[152] Vladan Deved˘zic´ and Sa˘sa R. Milenkovic, "Teaching Agile Software Development: A Case Study", IEEE TRANSACTIONS ON EDUCATION, VOL. 54, NO. 2, MAY 2011.

[153] Yue Jia, and Mark Harman, "An Analysis and Survey of the Development of Mutation Testing", IEEE transactions on software engineering, VOL. 37, NO. 5, SEPTEMBER/OCTOBER 2011, 0098-5589/11/$26.00 2011 IEEE, Published by the IEEE Computer Society

[154] Moonzoo Kim and Hotae Kim, "A Comparative Study of Software Model Checkers as Unit Testing Tools: An Industrial Case Study", IEEE transactions on software engineering, VOL. 37, NO. 2, MARCH/APRIL 2011, 0098-5589/11/$26.00 @2011 IEEE, Published by the IEEE Computer Society

[155] Magne Jørgensen and Stein Grimstad, "The Impact of Irrelevant and Misleading Information on Software Development Effort Estimates: A Randomized Controlled Field Experiment", IEEE transactions on software engineering, VOL. 37, NO. 5, SEPTEMBER/OCTOBER 2011, 0098-5589/11/$26.00 @2011 IEEE, Published by the IEEE Computer Society

[156] Yonghee Shin and et. al., "Evaluating Complexity, Code Churn, and Developer Activity Metrics as Indicators of Software Vulnerabilities", IEEE transactions on software engineering, VOL. 37, NO. 6, NOVEMBER/DECEMBER 2011, 0098-5589/11/$26.00 @2011 IEEE, Published by the IEEE Computer Society

[157] Ivica Crnkovic and Michel R.V., "A Classification Framework for Software Component Models", IEEE transactions on software engineering, VOL. 37, NO. 5, SEPTEMBER/OCTOBER 2011, 0098-

5589/11/$26.00 @ 2011 IEEE, Published by the IEEE Computer Society

[158] Andrea De Lucia, and et. al., "Improving Source Code Lexicon via Traceability and Information Retrieval", IEEE transactions on software engineering, VOL. 37, NO. 2, MARCH/APRIL 2011, 0098-5589/11/$26.00 @ 2011 IEEE, Published by the IEEE Computer Society

[159] Radu Calinescu, and et. al., "Dynamic QoS Management and Optimization in Service-Based Systems", IEEE Transactions on software engineering, VOL. 37, NO. 3, MAY/JUNE 2011, 0098-5589/11/$26.00 @ 2011 IEEE, Published by the IEEE Computer Society

[160] W. Ke and P. Zhang, "Effects of Empowerment on Performance in Open-Source Software Projects", IEEE transactions on engineering management, VOL. 58, NO. 2, MAY 2011

[161] P. K. Kapur and et. al., "A Unified Approach for Developing Software Reliability Growth Models in the Presence of Imperfect Debugging and Error Generation", IEEE TRANSACTIONS ON RELIABILITY, VOL. 60, NO. 1, MARCH 2011

[162] Cleidson R.B. de Souza and David F. Redmiles, "The Awareness Network, To Whom Should I Display My Actions? And, Whose Actions Should I Monitor?", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 37, NO. 3, MAY/JUNE 2011

[163] Anandasivam Gopal, and et. al., "Coordination and Performance in Global Software Service Delivery: The Vendor's Perspective", IEEE transactions on engineering management, VOL. 58, NO. 4, NOVEMBER 2011.

[164] Kavita Choudhary, "Parametric Estimation of Software Systems", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-2, May 2011

[165] Mazni Omar and et. al., "The Impact of Agile Approach on Software Engineering Teams", American Journal of Economics and Business Administration 3 (1): 12-17, 2011, ISSN 1945-5488 , 2010 Science Publications.

[166] Martin Nordio and et. al., "Collaborative Software Development on the Web", ETH Zurich, Switzerland, 2011.

[167] Bruno Antunes and et. al., "Context Capture in Software Development", Knowledge and Intelligent Systems Laboratory Cognitive and Media Systems Group, Centre for Informatics and Systems of the University of Coimbra, Coimbra, Portugal, 2011.

[168] Damien Cassou Emilie Balland Charles Consel and et. al., "Leveraging Software Architectures to Guide and Verify the Development of Sense/Compute/Control Applications", ICSE '11, May 21–28, 2011, Waikiki, Honolulu, HI, USA.

[169] Eleni Constantinou and et. al., "Open Source Software: How Can Design Metrics Facilitate Architecture Recovery?", Computer Science Department, Aristotle University of Thessaloniki, Thessaloniki, Greece, econst@csd.auth.gr, stamelos@csd.auth.gr, and Computer Science Department, Aristotle University of Thessaloniki, Thessaloniki, Greece, and Computer Science and Telecom. Department T.E.I. of Larisa, Greece, gkakaron@teilar.g

[170] Maurice H. ter Beek and et. al., "Product Lines for Service Oriented Applications - PL for SOA", Istituto di Scienza e Tecnologie dell'Informazione, ISTI–CNR, Pisa, Italy, and IMT Institute for Advanced Studies Lucca, Lucca, Italy, 2011.

[171] Robert Da˛browski and et. al., "Software is a directed multigraph (and so is software process)", Institute of Informatics, Warsaw University, Banacha 2, 02-097 Warsaw, Poland, 2011.

[172] P. A. Judas and L. E. Prokop, "A historical compilation of software metrics with applicability to NASA's Orion spacecraft flight software sizing", NASA-Lyndon B. Johnson Space Center, 2101 NASA Parkway, 2011.

[173] Laurie McLeod and et. al., "Qualitative research on software development: a longitudinal case study methodology", Springer Science+Business Media, LLC 2011.

[174] Steve Adolph and et. al., "Using grounded theory to study the experience of software development", Springer Science+Business Media, LLC 2011.

[175] Jana Samalikova and et. al., "Toward objective software process information: experiences from a case study", The Author(s) 2010. This article is published with open access at Springerlink.com

[176] Faheem Ahmed and Luiz Fernando Capretz, "A business maturity model of software product line engineering:, Springer Science+Business Media, LLC 2010.

[177] T. A. W. Jarratt, and et. al., "Engineering change: an overview and perspective on the literature", Springer-Verlag London Limited 2010.

[178] Natalia Juristo and Sira Vegas, "The role of non-exact replications in software engineering experiments", Springer Science+Business Media, LLC 2010.

[179] Anne Martens, and et. al., "From monolithic to component-based performance evaluation of software architectures", Springer Science+Business Media, LLC 2010

[180] Tomaz Dogs˘a and et. al., "The effectiveness of test-driven development: an industrial case study", Springer Science+Business Media, LLC 2011.

[181] Marko Rosenmüller, and et. al., "Flexible feature binding in software product lines", Springer Science+Business Media, LLC 2011.

[182] Yao-Jen Chang and et. al., "Student Engineers as Agents of Change: Combining Social Inclusion in the Professional Development of Electrical and Computer Engineering Students", Springer Science+Business Media, LLC 2010

[183] Andy Zaidman and et. al., "Studying the co-evolution of production and test code in open source and industrial developer test processes through repository mining", This article is published with open access at Springerlink.com, 2010.

[184] Jeronimo Castrillon and et. al., "Component-based waveform development: the Nucleus tool flow for efficient and portable software defined radio", Springer Science+Business Media, LLC 2011

[185] Cuauhtemoc Lopez-Martin, "Applying a general regression neural network for predicting development effort of short-scale programs", Springer-Verlag London Limited 2010

[186] Stephen Gilmore and et. al., "Non-functional properties in the model-driven development of service-oriented systems", Springer-Verlag 2010.

[187] Zude Li and et. al., "Characteristics of multiple-component defects and architectural hotspots: a large system case study", Springer Science+Business Media, LLC 2011.

[188] Alistair Sutcliffe and et. al., "Experience with user-centred requirements engineering", Springer-Verlag London Limited 2011.

[189] Eran Rubin and Hillel Rubin, "Supporting agile software development through active documentation", Springer-Verlag London Limited 2010

[190] Kristina Pitula and Thiruvengadam Radhakrishnan, "On eliciting requirements from end-users in the ICT4D domain", Springer-Verlag London Limited 2011.

[191] Steve Bannerman and Andrew Martin, "A multiple comparative study of test-with development product changes and their effects on team speed and product quality", Springer Science+Business Media, LLC 2010.

[192] Faheem Ahmed and Luiz Fernando , "An architecture process maturity model of software product line engineering", Springer-Verlag London Limited 2011

[193] Jitendra S. Kushwah and Mahendra S. Yadav, "testing for object oriented software", Indian Journal of Computer Science and Engineering (IJCSE), 2011.

[194] B.Bharathi and G.Kulanthaivel, "towards developing a performance evaluator for component based software architectures", Indian Journal of Computer Science and Engineering (IJCSE), 2011.

[195] Bimal Aklesh Kumar, "software architecture for fiji national university campus information systems", Indian Journal of Computer Science and Engineering (IJCSE), 2011.

[196] Arti Chhikara and R.S.Chhillar, "Impact of Aspect Orientation on Object Oriented Software Metrics", Indian Journal of Computer Science and Engineering (IJCSE), 2011.

[197] Arashdeep Kaur and Sunil Gulati, "A Framework for Analyzing Software Quality using Hierarchical Clustering", International Journal on Computer Science and Engineering (IJCSE), 2011

[198] Ghazal Keshavarz and et. al., "Metric for Early Measurement of Software Complexity", International Journal on Computer Science and Engineering (IJCSE), 2011.

[199] Sanjana and Dr. P.K. Suri, "Simulator for Software Project Reliability Estimation", International Journal on Computer Science and Engineering (IJCSE), 2011.

[200] Nishant Dubey, "A Paper Presentation on Software Development Automation by Computer Aided Software Engineering (CASE)", IJCSI International

Journal of Computer Science Issues, Vol. 8, Issue 1, January 2011.

[201] Osama Shata, "Teaching Software Engineering: Problems and Suggestions", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 1, January 2011.

[202] Rashmi Yadav, and et. al., "Software Engineering for Practiced Software Enhancement", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March 2011.

[203] Dr. Satya Prasad Ravi, and et. al., "SPC for Software Reliability: Imperfect Software Debugging Model", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, May 2011

[204] Rajender Singh Chhillar and Parveen Kajla, "A New - Knot Model for Component Based Software Development", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, May 2011, ISSN (Online): 1694-0814, www.IJCSI.org

[205] Narender Singh and Nasib Singh Gill, "Aspect-Oriented Requirements Engineering for Advanced Separation of Concerns: A Review", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 2, September 2011, ISSN (Online): 1694-0814, www.IJCSI.org

[206] Nor Azliana Akmal Jamaludin and Shamsul Sahibuddin, "Development Strategy using Cognitive Domain in e-Requirement Engineering Learning System", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 2, September 2011, ISSN (Online): 1694-0814, www.IJCSI.org

[207] Mohammed Abdul Bari and Dr. Shahanawaj Ahamad, "Process of Reverse Engineering of Enterprise Information System Architecture", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, September 2011 ISSN (Online): 1694-0814 www.IJCSI.org, 2011.

[208] Ashish B. Sasankar and Dr Vinay Chavan, "SWOT Analysis of Software Development Process Models", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, September 2011, ISSN (Online): 1694-0814, www.IJCSI.org

[209] R. Jeyakarthik, "Requirements Engineering In Current Web Engineering Methodologies", R.Jeyakarthik, Int. J. Comp. Tech. Appl., Vol 2 (3), 490-497, 2011.

[210] Rituraj Jain, "Improvement in Software Development Process and Software Product through Knowledge Management", Rituraj Jain et al, Int. J. Comp. Tech. Appl., Vol 2 (5), 1557-1562, 2011.

[211] Sheikh Umar Farooq, and et. al., "software measurements and metrics: role in effective software testing", Sheikh Umar Farooq et al. / International Journal of Engineering Science and Technology (IJEST), 2011.

[212] Dr.S.S.Riaz Ahamed, "integrated approach and strategic case issues of project engineering: a comprehensive study", International Journal of Engineering Science and Technology (IJEST), 2011.

[213] Dr.S.S.Riaz Ahamed , "the influence of scope and timing of quality assurance in the planned and systematic pattern of software development process: a novel study", International Journal of Engineering Science and Technology (IJEST), 2011.

[214] OLGA Fedotova, and et. al., "software effort prediction: an empirical evaluation of methods to treat missing values with rapid miner", International Journal of Engineering Science and Technology (IJEST), 2011.

[215] R.Amuthakkannan, "Effective Software Assembly for the Real time systems using Multi-level Genetic Algorithm", International Journal of Engineering Science and Technology (IJEST), 2011.

[216] DR. NACHAMAI and et. al., "enacted software development process based on agile and agent methodologies", International Journal of Engineering Science and Technology (IJEST), 2011.

[217] Jyothi R and Dr. V. K. Agrawal, "process quality analysis of perfective maintainability for component-based software systems using aspect-oriented programming techniques", International Journal of Engineering Science and Technology (IJEST), 2011.

[218] Suchetha Vijay and Annapoorna Shetty, "A study on different approaches towards Aspect Oriented Requirements Engineering", Indian Journal of Computer Science and Engineering (IJCSE), Aug - Sep 2011.

[219] Ashok Kumar Behera and Rajesh Tiwari, "Analysis and Design of Software Visualization Tool for the Behavior of Object Oriented Programming", International Journal of Computer Applications (0975 – 8887), 2011.

[220] Dhirendra Pandey, and et. al., "Security Requirement Engineering Issues in Risk Management", International Journal of Computer Applications (0975 – 8887), 2011.

[221] R.Chinnaiyan, and et. al., "An Experimental Study on Reliability Estimation of GNU Compiler Components – A Review", International Journal of Computer Applications (0975 – 8887), 2011.

[222] M. Aqeel Iqbal, and et. al., "Critical Aspects of Project Management Causing Major Impacts on Software Development", International Journal of Computer Applications (0975 – 8887), 2011.

[223] Bandla Srinivasa Rao and et. al., "Monitoring Software Reliability using Statistical Process Control: An Ordered Statistics Approach", International Journal of Computer Applications (0975 – 8887), 2011.

[224] Niharika Aggarwal and Aditya Garg, "Using Re-Usable, Secure Software Engineering Principles for Designing User Focused Mixed Reality Systems", International Journal of Computer Applications (0975 – 8887), 2011.

[225] Aurangzeb Khan, and et. al., "Managing Requirement Risks in Global Software Development", International Journal of Computer Applications (0975 – 8887), 2011.

[226] Elizabeth Bjarnason, and et. al., "Requirements Are Slipping Through the Gaps - A Case Study on Causes & Effects of Communication Gaps in Large-Scale Software Development", 2011 IEEE 19th International Requirements Engineering Conference, 2011.

[227] Francesca Arcelli Fontana and Marco Zanoni, "A tool for design pattern detection and software architecture reconstruction", Elsevier, 2011.

[228] Krzysztof Wnuk and et. al., "Scaling Up Requirements Engineering –Exploring the Challenges of Increasing Size and Complexity in Market- Driven Software Development", Department of Computer Science, Lund University, Sweden, and Siemens Corporate Research, Inc, USA, 2011.

[229] Manik Sharma and et. al., "A Comparative Study of Static Object Oriented Metrics", International Journal of Advancements in Technology, http://ijict.org/ , ISSN 0976-4860, 2012.

[230] Jordan Vrtanoski and Toni Stojanovski, "Order Handling in Convergent Environments", 2011 2nd International Conference on e-Education, e-Business, e-Management and E-Learning (IC4E 2011)

[231] Toni Stojanovski PhD and Tomislav Dzekov PhD, "Rapid Application Development Using Software Factories", Faculty of Informatics, European University, Republic of Macedonia, 2012.

[232] Fatima-Zahra Belouadha and et. al., "A MDA approach for defining WS-Policy semantic non-functional properties", International Journal of Engineering Science and Technology, Vol. 2(6), 2010, 2164-2171

[233] Faudziah Ahmad and et. al., "investigating the awareness of applying the important web application development and measurement practices in small software firms", International Journal of Computer Science & Information Technology (IJCSIT) Vol 3, No 6, Dec 2011.

[234] Ch Ram Mohan Reddy and et. al., "General Methodology for developing UML models from UI", International Journal on Web Service Computing (IJWSC), Vol.2, No.4, December 2011.

[235] Chintan Amrit, and et. al., "Identifying Coordination Problems in Software Development: Finding Mismatches between Software and Project Team Structures", 2012.

[236] Rizwan Jameel Qureshi, "requirements and the baseline plan", Department of Computer Science, COMSATS Institute of Information Technology, Defence Road, Lahore, 2012.

[237] Ananda Rao and M.Gopichand, "Four Layered Approach to Non-Functional Requirements Analysis", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 6, No 2, November 2011, ISSN (Online): 1694-0814, www.IJCSI.org.

[238] Ravi S. Iyer, "Improve the Practice of Software Development by Having a Software Development Career Track in CS & IT Academia", Software Consultant, Puttaparthi, Andhra Pradesh, India, 2012.

[239] Murtaza Hussain Shaikh, "collaboration for enhancing the system development process in open source diligence.", Advanced Computing: An International Journal (ACIJ), Vol.3, No.1, January 2012.

[240] Youssef Bassil, "building sustainable ecosystem-oriented architectures", International Journal in Foundations of Computer Science & Technology,Vol. 2, No.1, January 2012.

[241] M. Rizwan Jameel Qureshi and M. E. Sandhu, "Survey-Based Analysis of the Proposed Component-Based Development Process", Dept. of Computer Science, COMSATS Institute of Information Technology, Lahore. National College of Business Administration & Economics 40 E/I Gulberg III Lahore, Pakistan, 2012.

[242] M. R. J. Qureshi, "An Evaluation of the Improved XP Software Development Process Model", Dept. of Computer Science, COMSATS Institute of Information Technology Lahore Pakistan Defence Road, Off Raiwind Road Lahore Pakistan

[243] M. R. J. Qureshi and S.A. Hussain, "An Improved XP Software Development Process Model", Dept. of Computer Science, COMSATS Institute of Information Technology Lahore Pakistan, 2008.

[244] Nayyar Iqbal and M. Rizwan Jameel Qureshi, "Improvement of Key Problems of Software Testing in Quality Assurance", Department of Computer Science, COMSATS Institute of Information Technology, Lahore Pakistan, 2009.

[245] M. Rizwan Jameel Qureshi and M. E. Sandhu, "A Validation of the Proposed Component-Based Development Process", Dept. of Computer Science, COMSATS Institute of Information Technology, Lahore. National College of Business Administration & Economics 40 E/I Gulberg III Lahore, Pakistan, 2009.

[246] M. Rizwan Jameel Qureshia and et. al., "A new teaching model for the subject of software project management", Department of Computer Science, COMSATS Institute of Information Technology M. A. Jinnah Campus, Defence Road off Raiwind Road, Lahore Pakistan, 2012.

[247] M. Nadeem, and et. al., "A step forward to component-based software cost estimation in object-oriented environment", Pakistan Journal of Science (Vol. 62 No. 4 December, 2010)

[248] M. Rizwan Jameel Quresh, "Empirical Evaluation of the Proposed eXSCRUM Model: Results of a Case Study", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, May 2011, ISSN (Online): 1694-0814, www.IJCSI.org.

[249] Asif Irshad Khan and et. al., "A Comprehensive Study of Commonly Practiced Heavy & Light Weight Software Methodologies", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011, ISSN (Online): 1694-0814, www.IJCSI.org.

[250] Ahmed Barnawi, and et. al., "A Framework for Next Generation Mobile and Wireless Networks Application Development using Hybrid Component Based Development Model", International Journal of Research and Reviews in Next Generation Networks (IJRRNGN), Vol. 1, No. 2, December 2011, ISSN: 2046-6897, © Science Academy Publisher, United Kingdom, www.sciacademypublisher.com

[251] Ahmed Barnawi, and et. al., "novel component-based development model for sip-based mobile application", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[252] Lorina Negreanu, and et. al., "A Formal Approach for the Development of Service-Oriented Applications", Computer Science Department, "Politehnica" University of Bucharest Bucharest, Romania, 2012.

[253] Rashmi Gupta and Shalini Raghav, "Risk Assessment Techniques and Survey Method for COTS Components", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[254] Toni Stojanovski, and et. al., "Scalability of Data Binding in ASP.NET Web Applications", Faculty of Informatics, European University, Bld. Kliment Ohridski 68, 1000 Skopje, MACEDONIA, 2012

[255] Isabella and Emi Retna, "study paper on test case generation for gui based testing", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[256] Dr. Reena Dadhich and Ujala Chauhan, "integrating cmmi maturity level-3 in traditional software development process", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[257] Sandhya Tarar and Dr. Ela Kumar, "Design Paradigm and Risk Assessment of Hybrid Re-engineering with an approach for development of Re-engineering Metrics ", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[258] V.Dattatreya and et. al., "agile programming and design patterns in web development -a case study",

International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[259] Vikash Lalsing, and et. al., "people factors in agile software development and project management", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012

[260] Isabella and Emi Retna, "study paper on test case generation for gui based testing", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012

[261] S. Babu and Dr.R.M.S. Parvathi., "Development of Dynamic Coupling Measurement of Distributed Object Oriented Software Based on Trace Events",

[262] B.Jalender, and et. al., "designing code level reusable software components", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012

[263] Mar´ıa-Cruz Valiente, and et. al., "Applying Ontology-Based Models for Supporting Integrated Software Development and IT Service Management Processes", IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, VOL. 42, NO. 1, January 2012.

[264] Benjamin Heitmann and et. al., "An Empirically Grounded Conceptual Architecture for Applications on the Web of Data", IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, VOL. 42, NO. 1, January 2012 51.

[265] Holger M. Kienle, "Open Data: Reverse Engineering and Maintenance Perspective", arXiv:1202.1656v1 [cs.SE] 8 Feb 2012.

[266] Drazen Lucanin and Ivan Fabek, "A Visual Programming Language for Drawing and Executing Flowcharts", arXiv:1202.2284v1 [cs.PL] 10 Feb 2012.

[267] Geeta Nagpal and et. al., "A Hybrid Technique using Grey Relational Analysis and Regression for Software Effort Estimation using Feature Selection", International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-1, Issue-6, January 2012.

[268] Brendan Murphy, "The difficulties of building generic reliability models for software", Springer Science+Business Media, LLC 2011.

[269] Jesús M. González-Barahona: and Gregorio Robles, "On the reproducibility of empirical software engineering studies based on data retrieved from development repositories", This article is published with open access at Springerlink.com, 2011.

[270] Ernst Sikora and et. al., "Industry needs and research directions in requirements engineering for embedded systems", Springer-Verlag London Limited 2011.

[271] Martin Ivarsson and Tony Gorschek, "Tool support for disseminating and improving development practices", Springer Science+Business Media, LLC 2011.

[272] Dirk Riehle, "The single-vendor commercial open course business model", Springer-Verlag 2010.

[273] Marsha Chechik and et. al., "A relationship-based approach to model integration", Springer-Verlag London Limited 2011.

[274] Irene Luque Ruiz and Miguel Ángel Gómez-Nieto, "Rolling: A new technique for the practical teaching in computer science university degree", Springer Science+Business Media, LLC 2010.

[275] Yosi Ben Asher and et. al., "Refactoring techniques for aggressive object inlining in Java applications", Springer Science+Business Media, LLC 2011.

[276] Raphael Mannadiar and Hans Vangheluwe: , "Modular artifact synthesis from domain-specific models", Springer-Verlag London Limited 2011.

[277] Veerapaneni Esther Jyothi and K. Nageswara Rao, "Effective Implementation of Agile Practices – Incoordination with Lean Kanban", International Journal on Computer Science and Engineering (IJCSE), 2012.

[278] Abhiraja Sharma, and et. al., "Defect Prevention Technique in Test Case of Software Process for Quality Improvement", Int.J.Comp.Tech.Appl,Vol 3 (1), 56-61, IJCTA | JAN-FEB 2012, Available online@www.ijcta.com, ISSN:2229-6093.

[279] Lata Nautiyal, and et. al., "Elite: A New Component-Based Software Development Model", Int.J.Computer Techology & Applications,Vol 3 (1),119-124, IJCTA | JAN-FEB 2012, Available online@www.ijcta.com, ISSN:2229-6093.

[280] Mrinal Singh Rawat and et. al., "Survey on Impact of Software Metrics on Software Quality", (IJACSA) International Journal of Advanced Computer Science

and Applications, Vol. 3, No. 1, 2012, www.ijacsa.thesai.org.

[281] M.R. Dube, S.K. Dixit: , "Modeling Theories and Model Transformation Scenario for Complex System Development", International Journal of Computer Applications (0975 – 8887) Volume 38– No.7, January 2012.

[282] Hongli Liu, and et. al., "An object-oriented serial implementation of a DSMC simulation package", Published by Elsevier Ltd., Computers & Fluids, journal homepage: www.elsevier.com/locate/compfluid, Computers & Fluids 57 (2012) 66–75

[283] Musawwer Khan, and et. al., "Deployment Strategies for a Reengineered Information System in Context of Legacy System", Research Journal of Applied Sciences, Engineering and Technology 4(3): 178-185, 2012, ISSN: 2040-7467, © Maxwell Scientific Organization, 2012.

[284] Sunil Kumar Jangir, and et. al., "evolution and melioration of software management processes", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012.

[285] A.Nirmal Kuma and Dr.B.G.Geetha, "Achieving Software Engineering Knowledge Items with an Unit Testing Approach", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

[286] Muthu Ramachandran, "Guidelines Based Software Engineering for Developing Software Components", Journal of Software Engineering and Applications, 2012, 5, 1-6 doi:10.4236/jsea.2012.51001 Published Online January 2012 (http://www.SciRP.org/journal/jsea).

[287] Saleh H. Al-Daajeh and et. al., "A Tactic-Based Framework to Evaluate the Relationships between the Software Product Quality Attributes", Int.J. of Software Engineering, IJSE Vol.5 No.1 January 2012.

[288] Devesh Kumar Srivastava, and et. al., "VRS Model: A Model for Estimation of Efforts and Time Duration in Development of IVR Software System", Int.J. of Software Engineering, IJSE Vol.5 No.1 January 2012.

[289] Hisham M. Haddad, and et. al., "Framework for Instituting Software Metrics in Small Software Organizations", Int.J. of Software Engineering, IJSE Vol.5 No.1 January 2012.