Usability Evaluation Framework for Software Engineering Methodologies

Zafar Masood, Shang Xuequn, and Jamal Yousaf

Abstract—**According to software quality models published usability is an important quality attribute to measure how usable and effective/useful is the software system. Concept of usability has been applied to improve the quality of software systems, human computer interaction, software process improvement, and software process descriptions. Over the years different usability models have been proposed listing the factors, criteria and metrics to measure the usability of software systems, software processes, and software process descriptions. Since usability evaluation is a difficult task varioususability evaluation methods (UEM) have been proposed to describe different processes for systematic and effective way for usability evaluation. Concept of usability can be applied to software engineering methodologies to find how usable and useful a methodology is for software engineers or software development organizations, and it can also be used by method engineers or meta-model developers for method engineering to improve the usability of software engineering methods. Usability models and Usability evaluation methods proposed for software systems cannot be applied directly to software engineering methodologies due to difference of domains, concepts, and end users. In this work a usability evaluation framework is proposed which comprises usability factors and their description, criteria and their description to measure factors, importance and grading of criteria fulfillment, guidelines to apply the framework and overall evaluation process.**

Index Terms—**Software engineering, usability evaluation, software engineering methodologies, software quality.**

I. Introduction

Quality in use is also called usability and is an important quality attribute. Usability is defined in different ways by ISO [1], shackle (1991), Nielson (1993), Preece (1994), Constantine and Lockwood (1999), Seffah [2], and Alonso-Rios[3]. Different usability models according to definition of usability are published as described in detail in section IV.

These models are proposed for software systems, human computer interaction, and software processdescription etc which cannot be directly applied to software engineering methodologies. A critical analysis and taxonomy [3] is presented by D. Alonso-Rios, A. Vázquez-García, E. Mosqueira-Rey, and V. Moret-Bonillool in which different usability models are compared and Victor Nassar[4] tried to develop common criteria for usability review incorporating all other models. Usability evaluation has been applied on domain specific languages [5], software modeling systems [6], software process improvement [7], and software development practices [8].

Mahvishkhurrum, tony gorschek [9] have done a systematic review for usability and usefulness assessment of domain analysis solutions for product lines and mentioned that findings can be used as the checklist for the assessment of solutions. MohdNaz’riMahrin,D. Carrington, and P. Strooper [10] mentioned the usability factors that can effect usability of software process descriptions but he limited his study only to the description of software process contents. MohdNaz’riMahrin also work on the selection of usability evaluation methods for software process description and developed a factor based framework for the usability evaluation of software process descriptions.

Different evaluation frameworks for software engineering methodologies [11]-[13] have been developed and evaluation of development methodologies [14], [15] has been done for different domains. ShokoofehHesari, Hoda Mashayekhi, and Raman Ramsin [13] tried to generalize the framework for the evaluation of software engineering methodologies. Despite of all that concept of usability has not been applied to software engineering methodologies to evaluate how usable and useful software engineering methodologies can be for software engineers or organizations to develop solutions and it can also help method engineers or meta-modeling for method engineering to develop methods which are more usable and useful for the end users. Taking the motivation from application of usability to software systems, human computer interaction, process improvement, process descriptions, and domain specific modeling languages etc in this work usability evaluation framework for software engineering methodologies is proposed that can be used to evaluate how usable or useful is the methodology for projects depending upon the usage context.

Remaining sections of work are organized as follows:

1) Software engineering methodologies
2) Evaluation of software engineering methodologies
3) Survey of usability and usability evaluation models
4) Usability evaluation framework for SEM
    - Usability factors for Software engineering methodology
    - Importance level of criteria fulfillment
    - Criteria to evaluate usability factors for SEM
    - Grading of criteria fulfillment by methodology
5) Guidance on applying the framework
    - Evaluation skills required
    - Organization orways to conduct evaluation.
    - Usability level of factor or methodology
    - Tailoring of framework
    - Evaluation of criteria
    - Evaluation of factor and methodology
6) Guidelines for evaluation process

Manuscript received September 10, 2013; revised December 31, 2013.

Zafar Masood and Shang Xuequn are with Nwpuxian, China (e-mail: zafar_iiui@hotmail.com, shang@nwpu.edu.cn).

Jamal Yousaf is with Tsinghua Beijing, China (e-mail: jamal_qau@hotmail.com).

DOI: 10.7763/LNSE.2014.V2.127

7) Future work
8) Summary and conclusions

II. Software Engineering Methodologies (SEM)

According to [16] software engineering methodology as a noun is a development process framework i.e. Rational unified process (RUP), Agile process (AP) etc while as a verb it is an approach for applying the development processframework to achieve the desired milestones by completing tasks and activities i.e. structured, object oriented, feature oriented etc as shown in the Fig. 1 below.

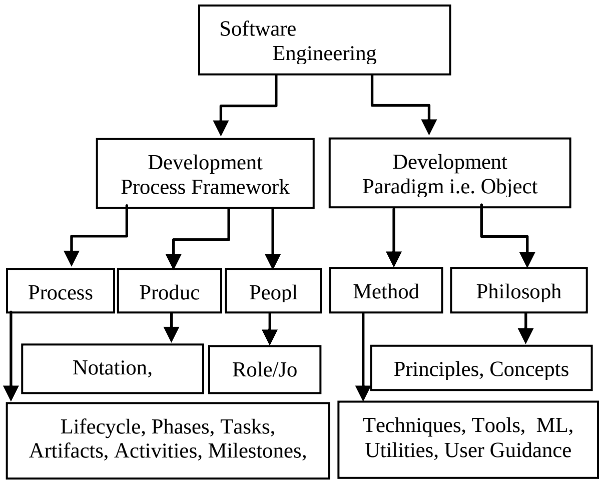
Software Engineering
Development Process Framework
Development Paradigm i.e. Object
Process
Produc
Peopl
Method
Philosoph
Notation,
Role/Jo
Principles, Concepts
Lifecycle, Phases, Tasks, Artifacts, Activities, Milestones,
Techniques, Tools, ML, Utilities, User Guidance

Fig. 1. Software engineering methodology.

According to [12] software engineering method provides an organized and systematic way of defining, analyzing, and specifying the problem while designing, implementing, and testing the solution against the specifications.

Software engineering methodology as shown in Fig. 1 can be described by process, product, people, and method description. According to SPEM [17] the concepts for software development method content are tasks, work products, role, and tool while concepts for the development process are activity, milestones, task use, work product use, and role use. According to RUP [18] software development process describes the structure built from workflows, activities, artifacts, and workers and when it is enacted than it describes the lifecycle, phase, iteration, and milestone.

Software Process description contains the process structure and enactment. Process is a composite structure containing phases, tasks, activities, and milestones while process enactment describes process workflow, phases, iterations, and milestones achievement criteria. Product description specifies the artifacts consumed or produced during the workflow execution after achieving a milestone. It describes the artifact content, structure or layout, and validity criteria. Artifact can be source code, document, model, or test case etc. People description is about the roles, jobs and skills required for successful fulfillment of tasks or activities in order to accomplish the milestones. Method describes a domain or discipline specific approach based on various concepts, philosophies, techniques, modeling language, and tools to solve real world complex problems. Method contains techniques, tools, utilities and user guidance to describe how to accomplish the tasks or activities in a systematic and effective way to produce quality artifacts.

According to [19] software engineering methodology defines the following aspects

1) Workflow of development process
2) Activities that are to be executed
3) Definition of work products with respect to content and structure/layout
4) Completion criteria for work products
5) Required skills for performing tasks
6) Responsibilities and capabilities of workers
7) Standards, guidelines, techniques and tools that are to be employed

III. Evaluation of Software Engineering Methodologies

Over the years numbers of software engineering methodology evaluation frameworks are published. DESMET [11], [12] is a methodology for evaluating software engineering methods/tools by Barbara. DESMET can be used to compare generic method or method which is a specific approach within a generic method or tool.

According to DESMET there are two types of evaluations:

1) Evaluation of measureable effects of using a method or tool.
2) Evaluation of method or tool appropriateness i.e. how much method is usable or useful.

DESMET refers measureable effects of using a method as quantitative or objective while method appropriateness as qualitative, feature analysis or subjective. Method appropriateness is accessed usually in terms of features provided by method/tool or training requirements. Another important consideration is how to organize the evaluation process. According to DESMET for qualitative evaluation it can be organized as survey, case study, or formal experiment. In qualitative screening it can be organized as feature screening mode, survey, formal experiment, or case study. According to DESMET qualitative feature screening mode can be done by a single person for number of methods where evaluator not only determines the features to be accessed and their rating scale but also done the assessment. In qualitative screening mode the evaluations are usually based on the literature describing the software method rather than actually using the method.

One way to evaluate software engineering methodology is to consider it as a software system while applying standard analyses, modeling techniques to prepare a model based on the required process, tasks, activities, roles etc. This reference model can serve as evaluation criteria against which any methodology can be evaluated [12]. According to Shokoofehhesari [13] a software methodology has generic and type specific features that can be evaluated. Generic features are modeling language, process, and application features while there can be number of type specific features. Proposed Usability evaluation framework for software engineering methodologies can be used to find out the methodology appropriateness i.e. how much the methodology is usable or useful and it lays down the factors, sub factors, and criteria to measure the factors using screening mode done by a single person or organization by analyzing the literature describing the software engineering methodology.

IV. Survey of Usability and Usability Evaluation Models

According to Hoyer, R. W. and Hoyer, B. B. Y there are two ways to define quality

1) Quality is the measurable characteristics of services and products satisfying the fixed specifications.
2) Quality is considered as the services and product capability to meet customer requirements.

According to ISO 9126:1 there are there are three ways to define software quality

1) Internal quality that is quality of code
2) External quality that is Quality of software execution
3) Quality in use that is Extent the user needs are met in the user's working environment

Quality in use is also called usability. According to ISO-9126 usability is the set of attributes that bear on the effort needed for use and on the individual assessment of such use, by as stated or implied set of users. According to ISO-9241-11 [1] usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. According to IEEE STD 9241-12-1990 usability is the ease with which user can learn to operate, prepare inputs, and interpret outputs of system.

According to shackle (1991) usability of the system is the capability in human function terms to be used easily and effectively by specified range of users, given specified training and user support to fulfill the specified range of tasks, within the specified range of scenarios. Nielson (1993) although not give formal definition of usability but presents the operational criteria which defines the concepts i.e. Learnability, Effectiveness, Memorability, Erros, and Satisfaction. According to Preece (1994) usability is the measure of throughput, learnability, throughput in terms of effectiveness, and attitude of users. According to Constantine and Lockwood (1999) usability can be evaluated by measuring efficiency in use, learnability, rememberability, reliability in use, and user satisfaction.

Ahmed Seffah, Mohammad Donyaee, Rex B. Kline, and Harkirat K. Padda[2] proposes a usability quality model QUIM (quality in use integrated measurement). QUIM lists 10 usability factors efficiency, effectiveness,productivity, satisfaction, learnability, safety, truthfulness accessibility, and usefulness. QUIM lists down also 26 criteria and metrics named time behavior, resource utilization, attractiveness, likeability, flexibility, minimal action, minimal memory load, operability, user guidance, consistency, self descriptiveness, feedback, accuracy, completeness, fault tolerance, resource safety, readability, controllability, navigability, simplicity, privacy, security, insurance, familiarity, load time, and appropriateness to directly measure usability of software systems. Alonso-Rios [3] presented the taxonomy of usability according to him usability can be described with Knowability, Operability, Efficiency, Robustness, Safety, and Subjective satisfaction as attributes.

According to [20] very limited research work has been done to develop usability factors, criteria and evaluation methods for software engineering methodologies as a whole. Mehrin [10] proposed thirteen usability factors for software process description named understandable, easy to learn, well structured, effective presented, effectively supported, tool supported, complete, concise, unambiguous, up-to-date, operable, navigable, and tailorable.

V. Usability Evaluation Framework for Software Engineering Methodologies

As in the last two sections we have given the introduction of software engineering methodologies, and usability evaluation models which is perquisite for defining and understanding the usability evaluation framework presented in this and next section.Overall framework contents and process are shown in Fig. 2 below.

In this section usability factors, criteria for factors, and grading for criteria fulfillment is defined and explained in detail. In Section VI guidance to apply framework which is integral part of framework is described.

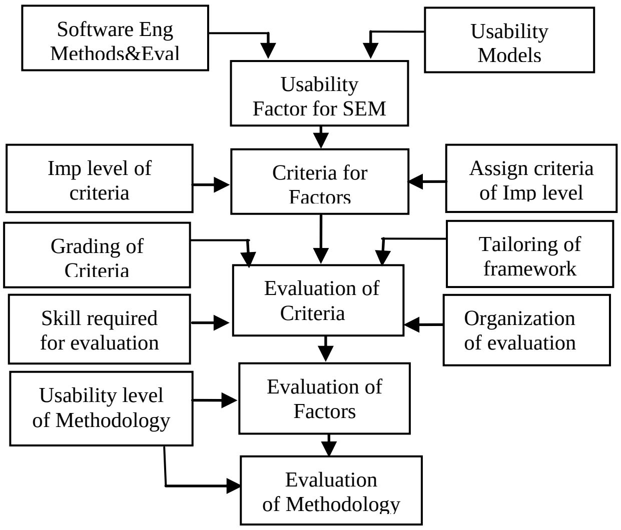

Fig. 2. Usability evaluation framework.

A. UsabilityFactors for Software Engineering Methodology

From the usability point of view when end users i.e. project managers, software engineers, or organizational executives chooses or adopts a software engineering methodology, successful adoption process occurs in following steps which are described in Table I:

1) Understanding
2) Learning
3) Applying
4) Effectiveness/Usefulness for future projects.
5) Satisfaction of End Users.

First step is to understand the methodology to answers the question what to do. Understanding is required for the clarity of concepts and to know about the philosophy and process of the methodology to solve the problems. Criterion for evaluation of understanding is listed in Table IV. After understanding next step is to learn methods, techniques, modeling language, implementation language, and tools to answer the question how to do. Learning is required to develop the essential skills required to complete the tasks and activities in order to achieve desired milestones. Criterion for evaluation of learning is listed in Table V.

After learning the next question is how to apply the methodology on organizational small, medium and large projects. Applicability of methodology is concerned with acquiring required resources, establishing the software development environment, and organizational culture. Criterion for evaluation of applying is listed in Table VI. After successfully applying the methodology next question is how useful the methodology is to solve real problems and to develop future projectsfor the organization. Effectiveness or Usefulness is the answer and can be evaluated by methodology completeness, coverage of phases, strength capability and expressiveness of modeling language and implementation technology, efficient utilization of resources i.e. time, people, money etc. Criterion for Effectiveness and Usefulness is listed in Table VII. Last thing that can be evaluated for the usability or usefulness of methodology is end user satisfaction like analysts, managers, software engineers etc. Criterion for evaluation of end user satisfaction is listed in Table VIII.

TABLE I: USABILITY FACTOR FOR SEM

Factor	Description
Understandability	Understanding means what to do, it is quality of description and elaboration of concepts about methodology and its elements i.e. process, product, people, method(s) etc so that user can understand easily.
Learnability	Learning means how to do, it is effort required to acquire technical skills by using methods, techniques, tools, utilities, developer network etc to solve examples, develop pilot projects or prototypes.
Applicability	How much convenient it is to establish development environment, organizational infrastructure, and to apply the methodology on organizational projects.
Effectiveness/ Usefulness	How much effective or useful the methodology is for the organization to develop current and future projects.
User satisfaction	After understanding, learning, and applying the methodology how much end user is satisfied during and after applying the methodology and his willingness to apply the methodology to future projects.

B. Importance Level of Criteria Fulfillment

Importance level of criteria fulfillment for achieving excellent evaluation score is described in Table II and is very important for right evalaution of usability factors.

TABLE II: IMPORTANCE LEVEL OF CRITERIA

Importance	Description
Mandatory	It is very important, obligatory and methodology must support it to be usable or useful.
Essential	It is critical and methodology should support it.
Expected	It is desirable and methodology is likely to support it.
Optional	It is optional and is better that methodology support it.

In order to assign importance level described above to criteria to be fulfilled we can generally classify criterions into groups and generalize the assignment criteria of importance level process. It can help us as a general rule or possible criteria but exceptions are there according to interpretation and priority of end user about the criteria. Importance level and general or possible assignment criteria is described in the Table III.

TABLE III: IMPORTANCE LEVEL ASSIGNMENT CRITERIA

Importance	General or Possible Assignment criteria
Mandatory	Technical strength and completeness. It is the most critical aspect of methodology required for it to be usable and useful because user cannot complete its tasks or achieve milestones and complete project at time without it. Criterions like completeness, method capability, tool support, scalability etc fall into this category.
Essential	Resources required or available to apply the methodology to organizational projects i.e. time, money, people, team skills etc. It plays an essential role for successful adaptation of methodology and completeness of projects. Some resources are only required at time of adoption of methodology while others are required on project to project basis.
Expected	Quality and availability of documentation which is very important for understanding and learning of methodology. It is expected by the end user of methodology to have it.
Optional	Community support i.e. consultancy, Online help, Familiarity etc for a successful adoption and application of methodology to pilot or real organizational projects.

C. Criteria to Evaluate Usability Factors for SEM

TABLE IV: CRITERIA FOR UNDERSTANDABILITY

Criteria/ (Importance)	Description and Guidance how to evaluate
Clarity (Expected)	Description of methodology and its elements process, product, people, and method in simple, comprehensive, explicit, and straightforward manner i.e. definitions with descriptions and examples in different scenarios using text, tables, diagrams, screen shots, source code etc.
Consistency (Essential)	Description of information in harmonious, uniform, and orderly way about methodology and its elements. Consistency means agreement, harmony, compatibility, or uniformity within and amongst the methodology elements i.e. tasks or activities should have corresponding roles to achieve them by producing specified products using corresponding techniques, tools and utilities.
Conciseness (Expected)	Description of information about methodology and its elements that is free from extra, irrelevant, and repeated details.
Unambiguous (Expected)	Description of information about methodology and its elements where there is only one interpretation of terms, definitions, or information throughout the description.
Well structured (Essential)	Presentation of information in a deliberate, disciplined, logical and proper way. Description of information using standard documentation guidelines i.e. headings, sub-headings, references, text font and size, tables, diagrams, lists etc.
Self descriptiveness (Mandatory)	Detailed description of methodology and its elements. Description of process structure i.e. lifecycle, phases, tasks, activities, and milestones etc; and enactment i.e. workflow, iterations, milestones achievement criteria etc.Description of product content, structure, layout etc.Description of roles, jobs, and skills required to complete tasks and accomplish milestones. Description of Method philosophy, modeling language, techniques, tools, implementation technology, and utilities etc.
Familiarity/ Universality (Optional)	Widespread usage of methodology, method, modeling language, tools, implementation technology and language for different domains. Established practice of methodology in different organizations developing solutions for different domains in all over world.

TABLE V: CRITERIA FOR LEARNABILITY

Criteria/ (Importance)	Description and Guidance how to evaluate
Training Availability (Expected)	Availability of training opportunities, workshops, and certification venues in different domains and areas of world for acquiring practical skills and learning of method.
User Guidance Availability (Essential)	Availability of technical documentation or developer network with examples, templates, checklist for description of tasks, activities, and action steps to guide the software engineers to learn how to use the modeling language, tools, techniques etc on practical examples to complete tasks.
User Friendliness of Tools/Tech (Expected)	Ease of use of tools, development environment, modeling language, implementation technology frameworks, or implementation language etc available so that user can easily learn the methodology.
Online help availability (Optional)	Availability of online help opportunities for different methodology elements i.e. process framework help, online developer network, developers forums, groups, sample projects or code websites etc.
Familiarity/ Universality (Expected)	Well established or generic use of methodology elements i.e. process framework as RUP, process lifecycle as spiral, modeling language as UML, implementation technology as .net, implementation language as c++ etc.
Learning Curve/ Method Complexity (Essential)	Effort required for developing skills for successful application of method to practical problems during learning or training. Effort depends on method complexity or difficulty level i.e. formal methods, non formal methods, no of tasks and activities to be completed, no of artifacts to be produced etc. method complexity depends on paradigm or philosophy i.e. structured, object oriented, component oriented, aspect oriented etc, modeling language complexity i.e. UML, AADL, VML, SYSML, Petri-nets etc, programming technology and language level low level, mid level, high level, 4th generation languages etc.

TABLE VI: CRITERIA FOR APPLICABILITY

Criteria/ (Importance)	Description and Guidance how to evaluate
Tool support (Mandatory)	CASE Tools availability and deployment requirements to automate the development process. Availability of integrated development environment and its Integration with other tools and technologies.
Consultancy availability (Optional)	Availability of consultancy support for the adoption of methodology i.e. establishment of development environment, help in application of methodology on pilot projects, real projects in organizational environment by methodology experts.
Resources required (Essential)	Resources required for establishment of organization infrastructure i.e. money, people, time etc. financial resources required to acquire technology, tools, skilled professional, hardware resources, and training or certification of human resource etc. requirement of skilled professionals and time required to adopt the method.
Team skills required (Essential)	Requirement of team skills for accomplishment of tasks and availability of skilled professional for team formation. it includes project managers, team leads, analysts, architects, designers, programmers, testers etc.
Technical/Legal Constraints (Optional)	Technical constraints for applying methodology i.e. specific modeling language, technology, programming language, operating system etc. legal constraints means licensing or any other regulatory constraints.
Flexibility (Expected)	Provision of tailoring of methodology elements i.e. process workflow configurability. Flexibility to adopt available methods or techniques, implementation technology, or programming language. Support for different platforms etc.
Scalability (Mandatory)	Applicability of methodology to different type of projects i.e. development or enhancement of existing software systems, new software systems etc. Applicability to different size of projects i.e. small, medium, and large etc. Applicability to different application domain i.e. desktop based, distributed computing, web development, aerospace etc.

TABLE VII: CRITERIA FOR EFFECTIVENESS/USEFULNESS

Criteria/ (Importance)	Description and Guidance how to evaluate
Scalability (Mandatory)	As described above in Table IV.
Flexibility (Expected)	As described above in Table IV.
Completeness (Mandatory)	Support of methodology for full life cycle coverage i.e. planning, analysis, architecture, design, construction, testing, deployment, maintenance etc.
Method capability (Mandatory)	Capability of method to solve complex problems. Capability of paradigm as level of abstraction, modularity, coupling, and cohesion etc. Capability and power of modeling language i.e. support for multiple paradigms, modularity, handling model inconsistencies, coverage of both static and dynamic aspects, supported views, support for model transformation logic etc. Capability and power of implementation technology or language i.e. availability of frameworks, libraries, toolkits for integration; compilation and deployment support for multiple platforms, capability to program as low, mid and high level language. Capability of integrated development environment by providing designing, programming, debugging, compilation, packaging, deployment, virtual execution support within single GUI.
Efficiency (Mandatory)	Efficient utilization of resources in terms of people, money and time to complete the projects.
Productivity (Essential)	Productivity achieved in the completion of projects using the methodology. Productivity focus on the output produced with respect to input resources consumed.

D. Grading of Criteria Fulfillment by Methodology

For usability evaluation evaluator will have to evaluate usability factors described in Table I. For usability factor evaluation evaluator will have to grade the criteria fulfilled by the methodology. It is the most time consuming and critical task for evaluator and whole evaluation result depends on the output of this task. Different grading or criteria fulfillment levels are described in the table below which evaluator can assign to the criterions listed in the Table IV to Table VIII.

TABLE VIII: CRITERIA FOR USER SATISFACTION

Tool Support, User friendliness of tools/ technologies, Online help availability, Efficiency, Productivity, Method Complexity, Method Capability, User guidance, Familiarity/Universality are the criteria to evaluate end user satisfaction. Criteria description is same as described above in Table IV- Table VIII.

TABLE IX: GRADING OF CRITERIA FULFILLMENT

Grading Level	Description	Score
Full Support	Feature is available and fullysupported by the methodology.	15
Partial Support	Feature is available but partially supported by the methodology.	10
Little support	Feature is mentioned but very little or ineffective support by the methodology.	5
No Support	Feature is neither mentioned nor supported by the methodology.	0

VI. GUIDANCE ON APPLYING THE FRAMEWORK

A. Evaluation Skills Required

Different criterions and grading of criteria are described above in the Table IV-Table VIII and Table IX respectively for evaluating factors. Certain skills are required by evaluator to grade the criteria fulfillment in light of description and guidance presented above in the tables. There are different types of criterion that requires either technical knowledge of software engineering methodologies, project management, analytical, technical writing/reading. Some criterions require evaluator to have technical skills to rightly grade the criteria fulfillment while others require managerial, analytical, or literary skills.

Criterion like clarity, consistency, unambiguous, well-structured, conciseness, and user guidance availability can be evaluated well by technical documentation writers who are part of software organization.Criterion like method capability, method complexity, tool support, completeness, scalability, flexibility, self descriptiveness, online help availability, user friendliness of tools, familiarity or universality, productivity, and efficiency requires technical skills.

Criterion like technical/legal constraints, team skills required, resources required, consultancy available, training availability requires managerial or corporate skills. Criterions like familiarity, training availability, legal constraints, consultancy availability, online help availability can be evaluated effectively without having sound technical or managerial knowledge.

If the evaluator has development and managerial skills then he can evaluate the methodology using the framework very effectively. Technical skills are the most required one because these are required for the evaluation of mandatory factors/sub-factors/criteria.

B. Organization or Ways to Conduct Evaluation

It is feature analysis based framework which can be applied for subjective evaluation [11] which can be organized in different ways like screening mode, surveys, case studies, and formal experiment.

If there is a single evaluator having technical, management and documentation skills then he can use screening mode [11] to evaluate the method. In screening mode the evaluator review the literature available i.e. methodology guide book, user guide, technical documentation, experience or project papers, review papers, white papers, training material, and groups/forums discussions to evaluate the factors. If there are large number of methods to be evaluated than screening mode is appropriate to evaluate the methods initially then selected or short listed methods can be evaluated further by other methods described below for achieving desired quality of evaluation results.

If the evaluator does not possess essential skills or expertise in the field for evaluation or expert opinion is required for evaluation of methodology then survey is an effective method for evaluation. For survey multiple questionnaires can be prepared by the evaluator for different type of end users according to the factors or criterion described above. Questioners can be used to acquire expert opinion from different type of end users or researchers. After receiving the response from experts and researcher evaluator can integrate and summarize the evaluation results.

If a team of software engineers is available then they can understand, learn and apply the methodology to a case study, pilot project or prototype and can observer the effectiveness/usefulness, user satisfaction.Developers should keep in mind the factors or criterion described in tables above during the adoption and development process to evaluate and grade the methodology according to the findings. Selection of case study is very important in a sense that after adoption and development all aspects of methodology can be evaluated.

If there are number of professional teams available to develop a real organizational project separately then formal experiment is suitable for evaluation in which each team develops the project using the methodology in isolation and evaluate the methodology as in case study mode. After each team evaluation exercise the evaluation results are crossed compared, integrated, and validated. Formal experiment is the most expensive way of evaluation but the quality of evaluation result is very high.

C. Usability Level of Factor or Methodology

TABLE X: USABILITY LEVEL FOR SEM

Usability level	Achievement criteria
Excellent	If methodology fully support mandatory, essential, expected and optional criteria.
Very High	If methodology fully support mandatory, essential, expected criterion and mix support for the optional criteria.
High	If methodology fully support mandatory, essential criteria and full or partial support for expected, while mix support for expected or optional criteria.
Very Good	If methodology fully support mandatory, essential criteria and mix support for expected and optional criteria.
Good	If methodology fully support mandatory criteria and full or partial support for essential while mix support for expected and optional criteria.
Average	If methodology fully support mandatory criteria but mix support for essential, expected or optional criteria.
Fair	If methodology has full or partial support for mandatory criteria and mix support for essential, expected and optional criteria.
Poor	If methodology has mix support for mandatory, essential, expected, or optional criteria.

After evaluation of methodology the evaluator should assign a usability level achieved by the methodology. Usability level shows how much a methodology is usable or useful for adoption and development of organizational projects. Possible usability levels and their achievement criteria is described in the Table X.

D. Tailoring of Framework

If required evaluator can tailor the framework according to requirements by adding new criteria for usability factor evaluation but the new criteria assignment should be in accordance with the philosophy or definition of usability factor described in Table I i.e. understandability means study of methodology elements, concepts, philosophy for overall understanding of methodology, learning means how to do or acquiring technical skills to fulfill tasks etc. New criteria and its description should be in accordance with description of usability factor. Evaluator can add or change grading, importance, or usability level for evaluation but overall process of evaluation should remain the same. According to the requirements framework can also be used for the individual evaluation of usability factors. Inclusion or exclusion of usability factors is not expected until there is change or extension in the definition or description of usability in published research.

E. Evaluation of Criteria

Evaluator can assign the grade to criteria fulfilled by the methodology after evaluation in a tabular form as shown in Table XI to all the criterions mentioned from Table IV-Table VIII. Evaluator will give brief remarks in the table and can describe details if required after the table in form of paragraphs by using text footnotes.

TABLE XI: CRITERIA EVALUATION FOR LEARNABILITY

Criteria/ (Importance)	Methodology support	Remarks
Training Availability (Expected)	Full support	Trainings are available in different parts of world at different time with material.
User Guidance Availability (Essential)	Partial support	User guidance documentation inform of text files is available but developer network is not available.
……..	……..	....

F. Evaluation of Factors and Methodology

After evaluating criteria in next step evaluator assign usability level achieved to usability factors and methodology in light of achievement criteria described in Table X. After assigning the usability levels achieved evaluator can give a summary report about usability of factors i.e. understandability, learn ability etc and the methodology. When concluding about the usability level of methodology, importance level of criteria fulfillment is very important for overall assessment.

Different methodologies can evaluated separately and then can be crossed compared on the basis of grading and usability level achieved at the methodology or individual usability factor level.

If evaluator wants to quantify the evaluation result or cross compare different methodologies on basis of numerical score than he can assign the score to criteria fulfillment evaluation grade i.e. full support 20, partial support 15 etc and compute score for mandatory, essential, expected, and optional criterions fulfilled which can be used tosort or compare methodologies.

TABLE XII: CRITERIA EVALUATION FOR LEARNABILITY

Methodology/ Factor	Usability level	Remarks
Methodology name	High	According to Table XI Methodology fully support mandatory, essential criteria and full or partial support for expected, while mix support for expected or optional criteria.
Understandability	Very High	Methodology fully support mandatory, essential, expected criterion and mix support for the optional criteria.
-----------	----------	----------------
-----------	----------	----------------

VII. GUIDELINES FOR EVALUATION PROCESS

Evaluation process contains sub-processes of planning, and performing the usability evaluation exercise. Usability evaluation framework is presented in this work and evaluator has to understand it by completing the tasks listed below while he startsplanning sub-process.

- Understand Evaluation Framework
- Software engineering methods concepts
- Usability evaluation and evaluation models
- Usability factors, criteria, importance level, grading, usability level etc
- Applying the framework

Plan evaluation sub-process is very much dependent upon the evaluation context or purpose or scope of evaluation and is very important for achieving the desired results. Performing and documenting evaluation results is the last sub-process and it ends with the evaluation results. Tasks or activities of sub-processes plan perform and document evaluations are listed below.

1) Plan evaluation
   - Understand evaluation framework as described in I
   - Decide or prepare list of methods to beevaluated
   - Perform survey of method(s) to be evaluated
   - Decide organization of evaluation i.e. screening mode etc based on the resources required or available
   - Acquire resources required for performing the evaluation i.e. documentation, hardware, software, people etc
2) Perform and document evaluation
   - Apply evaluation framework according to decided evaluation method and guidance presented in the Section VI
   - Document the evaluation results
   - Integrate the evaluation results if required i.e. formal experiment etc

VIII. FUTURE WORK

In future usability evaluation framework for software engineering methodologies will be applied to domain engineering methodologies available for software product family engineering by a single user using DESMET feature screening mode for usability evaluation and results will be published. If required necessary changes will be accommodated in the framework.

IX. Summary and Conclusions

Software engineering methodology is defined both as verb i.e. development process framework and as noun i.e. development paradigm or Software development method i.e. object oriented methods etc. In this work evaluation of software methodology is done both as development process framework and Software development method. First the concepts about software engineering methodologies and its elements are presented, and then a review for definition of usability and its evaluation methods are presented which can be applied to software systems, human computer interaction, and software process descriptions etc. These usability evaluation methods are not directly applicable to software engineering methodologies due to difference of domain, concepts and end users etc. Then we reviewed the work about the evaluation of software engineering methodologies published in renowned DESMET and other methodologies. DESMET mentioned the features or factors based evaluation which can be done quantitatively and qualitatively using feature screening mode, formal experiment, survey, or case study.

In this work we first describe the concepts about software engineering methods and usability evaluation models. Then we presented a usability evaluation framework by describing usability factors, criteria to evaluate usability factors,importance level of criteria fulfillment, grading of features or factors, and guidance on applying usability evaluation framework. In last section the evaluation process is discussed in very brief and concise wayby just defining the main tasks and workflow.

Concepts of usability can be applied to software engineering methodologies to find out how much a software development methodology is usable or useful for a software engineer or software development organization to adopt the methodology for the development of projects. It can also be used by method engineers or meta-model developers for method engineering to improve the usability of software engineering methods.

References

[1] *Guidance on Usability,* ISO Standard 9241-11-1998(E).

[2] A. Seffah, M. Donyaee, R. B. Kline, and H. K. Padda, "Usability measurement and metrics: a consolidated model," *Software Quality Journal,* vol. 14, no. 2, pp. 159-178, 2006.

[3] D. A. R ós, A. V. Garc á, E. M. Rey, and V. M. Bonillool, "Usability: a critical analysis and a taxonomy," *International Journal of Human Computer Interaction,* vol. 26, pp. 53–74, 2010.

[4] V. Nassar, "Common criteria for usability review," *Work: A Journal of Prevention, Assessment and Rehabilitation,* vol. 44, pp. 1053-1057, 2011.

[5] A. Barisic, V. Amaral, and M. Goulao, "Usability evaluation of domain specific languages," in *Proc. 8th International Conference on the Quality of Information and Communications Technology,* Lisbon Portugal, 2012, pp. 342 – 347.

[6] N. Kerzazi and M. Lavallee, "Inquiry on usability of two software process modeling systems using ISO/IEC 9241," in *Proc. 24th Canadian Conference onElectrical and Computer Engineering,* Niagara Falls, 2011, pp. 000773-000776.

[7] P. B. Polgar and M. Biro, "The usability approach in software process improvement," *Systems, Software and Service Process Improvement,* vol. 172, Berlin, Heidelberg:Springer-Verlag, 2011, pp. 133-142.

[8] M. K. Larusdottir, "Usability evaluation in software development practice," *Human-Computer Interaction–INTERACT,* vol. 6949, Berlin, Heidelberg: Springer, 2011, pp. 430-433.

[9] M. Khurrum and T Gorschek, "A systematic review of domain analysis solutions for product lines," *The journal of systems and software,* vol. 82, pp. 1982-2003, 2009.

[10] M. N. Mahrin, D. Carrington, and P. Strooper, "Investigating factors affecting the usability of software process descriptions," *Making Globally Distributed Software Development a Success Story*, vol. 5007, Berlin, Heidelberg: Springer, 2008, pp. 222-233.

[11] B. A. Kitchenham, "Evaluating software engineering methods and tool part 1: the evaluation context and evaluation methods," *ACM SIGSOFT Software Engineering Notes*, vol. 21, pp. 11-14, January 1996.

[12] B. A. Kitchenham, "Evaluating software engineering methods and tool part 6: identifying and scoring features," *ACM SIGSOFT Software Engineering Notes*, vol. 22, no. 2, pp. 16-18, March 1997.

[13] S. Hesari, H. Mashayekhi, and R. Ramsin, "Towards a general framework for evaluating software development methodologies," in *Proc. 2010 IEEE 34th Annual Computer Software and Applications Conference*, Seoul, July 2010, pp. 208-217.

[14] C. Lin, K. M. Kavi, F. T. Sheldon, K. M. Daley, and R. K. Abercrombie, "A methodology to evaluate agent oriented software engineering Techniques," in *Proc. the 40th Hawaii International Conference on System Sciences,* Hawaii, 2007, pp. 60-70.

[15] K. Sousa, J. Vanderdonckt, B. Sellers, and C. G. Perez, "Evaluating a graphical notation for modelling software development methodologies," *Journal of Visual Languages and Computing*, vol. 23, issue 4, pp. 195-212, 2012.

[16] Software development methodology. [Online]. Available: http://en.wikipedia.org/wiki/Software_development_methodology.

[17] Software and Systems Process Engineering Meta-Model Specification Version 2.0. (2008). [Online]. Available: http://www.omg.org/spec/SPEM/

[18] Rational unified process (RUP) Best Practices for Software Development Teams. [online]. Available: http://www01.ibm.com/software/rational/rup/

[19] G. Engels and S. Sauer, "A Meta-Method for Defining Software Engineering Methods," in Graph Transformations and Model-Driven Engineering, vol. 5765, Berlin, Heidelberg: Springer, 2010, pp. 411–440.

[20] M. N. Mahrin, "Selecting usability evaluation methods for software process descriptions," in *Proc. 2009 16th asia-pacific software engineering conference,* Penang, Dec. 2009, pp. 523-529.

Zafar Masood is a Pakistani national and done his master in computer science from international Islamic university Islamabad Pakistan in 2004. Currently he is a student of Master in software engineering at Northwestern Polytechnical university Xian, China in School of Computer Science and Technology.

He has worked as one year as software engineer in US based software development organization Elixir technologies Islamabad Pakistan and 7 years of software development experience in research based public sector organization in Islamabad pakistan. His research interests are software engineering, software architecture, and distributed computing.

Jamal Yousaf is a Pakistani national and done his master in mathematics from Quaid-e-Azam university Islamabad and master in computer engineering from University of Engineering and Technology Taxila. Currently he is PhD student in University of Tsinghua Beijing, China.

He has 12 years of software development experience in research based public sector organization in Islamabad Pakistan. His research interests are software engineering, social networking and path planning.