(12) PATENT APPLICATION PUBLICATION (21) Application No. 202241066581 A

(19) INDIA

(22) Date of filing of Application :20/11/2022 (43) Publication Date : 23/12/2022

(54) Title of the invention : A NOVEL MODEL FOR METRICS TO ASSESS THE QUALITY OF SOFTWARE REQUIREMENTS SPECIFICATION

(51) International classification	:G06F0008100000, G06F0011360000, G06F0008770000, G06F0008360000, G06Q0010060000
(86) International Application No Filing Date	:PCT// / :01/01/1900
(87) International Publication No	: NA
(61) Patent of Addition to Application Number Filing Date	:NA :NA
(62) Divisional to Application Number Filing Date	:NA :NA

(71)**Name of Applicant :**
1)Andhra University
Address of Applicant :A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003 Visakhapatnam ----------- -----------
Name of Applicant : NA
Address of Applicant : NA
(72)**Name of Inventor :**
1)Prof. James Stephen Meka
Address of Applicant :Professor and Principal, Department of CSE, Wellfare Institute of Science Technology and Management (WISTM), EC Member, Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code:530003 Visakhapatnam ----------- -----------
2)Mr.Raja Ramesh Merugu
Address of Applicant :Research Scholar, Department of CS & SE, A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003 Visakhapatnam ----------- -----------
3)Prof. Prasad Reddy P.V.G.D.
Address of Applicant :Senior Professor, Department of CS & SE, A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003 Visakhapatnam ----------- -----------

(57) Abstract :
The requirements engineering phase will provide the software requirements specification. It will guide software development. Quality requirements specifications lead to project success. Poor software requirements specification will cause project failure. Thus, quality SRS (software requirements specification) is the most significant requirement engineering activity. The present invention disclosed herein is a novel model for metrics to assess the quality of software requirements specification comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS in the present invention. Each has pros and cons. Few metrics exist to evaluate SRS quality. These metrics can be applied to any SRS, regardless of project type. Three case studies use these measures, with satisfactory outcomes.

No. of Pages : 25 No. of Claims : 10

FORM 1

THE PATENTS ACT, 1970
(39 of 1970)
&
THE PATENTS RULES, 2003
APPLICATION FOR GRANT OF PATENT
[See sections 7, 54 & 135 and rule 20(1)]

(FOR OFFICE USE ONLY)

Application No.: ....
Filing Date: ....
Amount of Fee Paid: ....
CBR No.: ....
Signature: ....

1. APPLICANT(S):

Sr.No	Name	Nationality	Address	Country	State
1	Andhra University	India	A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003	India	Andhra Pradesh

2. INVENTOR(S):

Sr.No	Name	Nationality	Address	Country	State
1	Prof. James Stephen Meka	India	Professor and Principal, Department of CSE, Wellfare Institute of Science Technology and Management (WISTM), EC Member, Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code:530003	India	Andhra Pradesh
2	Mr.Raja Ramesh Merugu	India	Research Scholar, Department of CS & SE, A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003	India	Andhra Pradesh
3	Prof. Prasad Reddy P.V.G.D.	India	Senior Professor, Department of CS & SE, A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003	India	Andhra Pradesh

3. TITLE OF THE INVENTION: **A NOVEL MODEL FOR METRICS TO ASSESS THE QUALITY OF SOFTWARE REQUIREMENTS SPECIFICATION**

4. ADDRESS FOR CORRESPONDENCE OF APPLICANT / AUTHORISED PATENT AGENT IN INDIA:

Andhra University, Prof. James Stephen Meka, Professor and Principal, Department of CSE, Wellfare Institute of Science Technology and Management (WISTM), EC Member, Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code:530003

Telephone No.:
Fax No.:
Mobile No: ....
E-mail: jamesstephenmm@yahoo.com

5. PRIORITY PARTICULARS OF THE APPLICATION(S) FILED IN CONVENTION COUNTRY:

Sr.No	Country	Application Number	Filing Date	Name of the Applicant	Title of the Invention

6. PARTICULARS FOR FILING PATENT COOPERATION TREATY (PCT) NATIONAL PHASE APPLICATION:

International Application Number	International Filing Date as Allotted by the Receiving Office
PCT//	

7. PARTICULARS FOR FILING DIVISIONAL APPLICATION

Original (first) Application Number	Date of Filing of Original (first) Application

8. PARTICULARS FOR FILING PATENT OF ADDITION:

Main Application / Patent Number:	Date of Filing of Main Application

9. DECLARATIONS:

(i) Declaration by the inventor(s):

I/We, Prof. James Stephen Meka, Mr.Raja Ramesh Merugu, Prof. Prasad Reddy P.V.G.D., is/are the true & first inventor(s) for this invention and declare that the applicant(s) herein is/are my/our assignee or legal representative.

(a)Date: ----- Dated this 20th day of November, 2022

(b)Signature(s) of the inventor(s): ....

(c)Name(s):
Prof. James Stephen Meka

Mr.Raja Ramesh Merugu

Prof. Prasad Reddy P.V.G.D.

(ii) Declaration by the applicant(s) in the convention country:

I/We, the applicant(s) in the convention country declare that the applicant(s) herein is/are my/our assignee or legal representative.

(a)Date: ----- Dated this 20th day of November, 2022

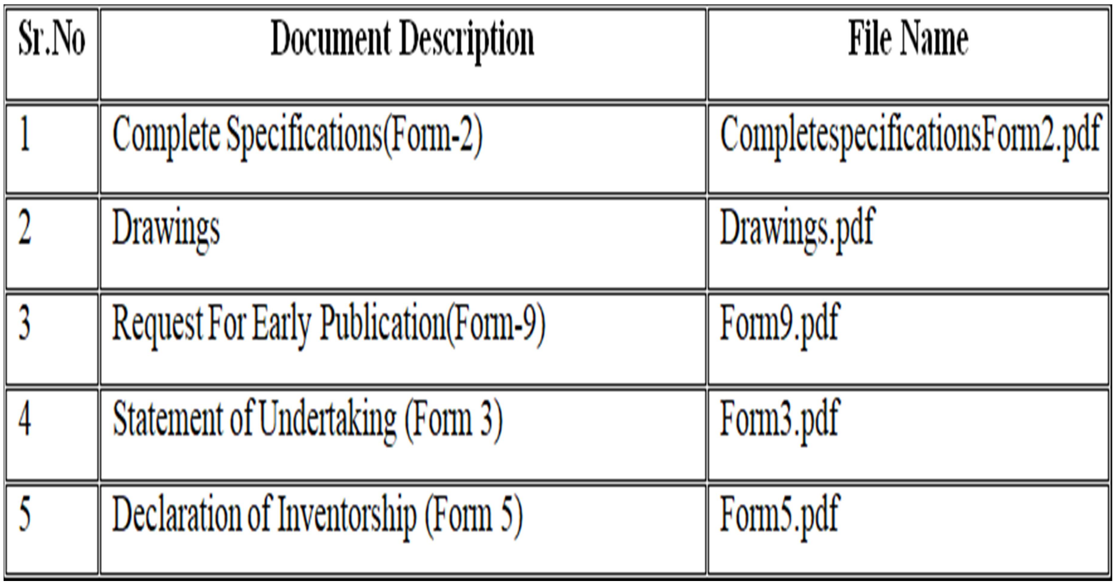
(b)Signature(s) of the inventor(s): ....

(c)Name(s): Andhra University

(iii) Declaration by the Applicant(s):

- **The Complete specification relating to the invention is filed with this application.**
- **I am/ We are, in the possession of the above mentioned invention.**
- **There is no lawful ground of objection to the grant of the Patent to me/us.**

10. FOLLOWING ARE THE ATTACHMENTS WITH THE APPLICATION:



Sr.No	Document Description	File Name
1	Complete Specifications(Form-2)	CompletespecificationsForm2.pdf
2	Drawings	Drawings.pdf
3	Request For Early Publication(Form-9)	Form9.pdf
4	Statement of Undertaking (Form 3)	Form3.pdf
5	Declaration of Inventorship (Form 5)	Form5.pdf

I/We hereby declare that to the best of my/our knowledge, information and belief the fact and matters stated herein are correct and I/We request that a patent may be granted to me/us for the said invention.

Dated this (Final Payment Date): -------- Signature: ....

Name(s):

Prof. James Stephen Meka

Mr.Raja Ramesh Merugu

Prof. Prasad Reddy P.V.G.D.

To

The Controller of Patents

The Patent office at CHENNAI

FORM 2

THE PATENTS ACT, 1970
(39 of 1970) &
THE PATENTS RULES, 2003

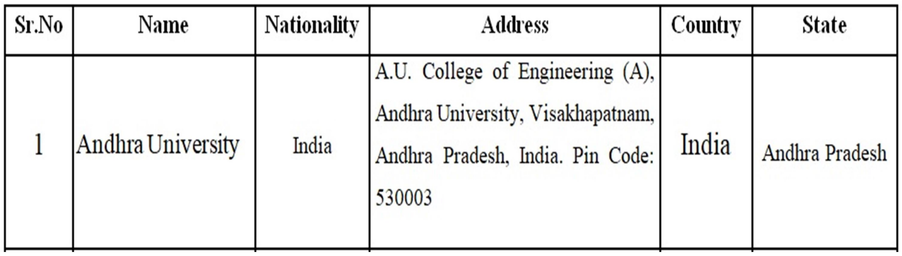
COMPLETE SPECIFICATION

(See section 10, rule 13)

1. TITLE OF THE INVENTION:

A NOVEL MODEL FOR METRICS TO ASSESS THE QUALITY OF SOFTWARE REQUIREMENTS SPECIFICATION

2. APPLICANT(S)



Sr.No	Name	Nationality	Address	Country	State
1	Andhra University	India	A.U. College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India. Pin Code: 530003	India	Andhra Pradesh

3. PREAMBLE TO THE DESCRIPTION:

COMPLETE SPECIFICATION

The following specification particularly describes the invention and the manner in which it is to be performed.

A NOVEL MODEL FOR METRICS TO ASSESS THE QUALITY OF SOFTWARE REQUIREMENTS SPECIFICATION

FIELD OF INVENTION

The present invention relates to the technical field of Computer Science.

Particularly, the present invention is related to a novel model for metrics to assess the quality of software requirements specification of the broader field of Software Engineering of Computer Science.

More particularly, the present invention is related to a novel model for metrics to assess the quality of software requirements specification provides various ways to assess the quality of SRS with metrics.

BACKGROUND & PRIOR ART

A software requirements specification (SRS) is document that is created when a detailed description of all aspects of the software to be built must be specified before the project is to commence. A software systems' requirements specification specifies all the necessary requirements for software system. This task explains the software system behaviour which is supposed to be developed. It contains use cases, which describes the interactions among the users and the software system. A Software Requirements Specification (SRS) is a document that describes all external observable behaviour and characteristics expected of a software system. This SRS document provides necessary information about all the requirements which are needed for the development of a software system. The software requirements specification is the most important document in software development. This document should be always a qualitative document. This document will drive the entire software development. If the SRS developed is quality SRS then it leads to the success of the project. If the SRS is not a quality SRS definitely it leads to failure of the project. Hence producing the quality SRS is most important thing in the software development. For this there are different approaches proposed by different authors. The methods/approaches are having their own disadvantages and limitations. Hence we propose a metrics in this paper to measure the quality of software requirements specification.

Different assessment models were proposed in the literature for measuring the quality of software requirements specification. Some of the models concentrate on individual measures for quality parameters of the software requirements specification. However several assessment models were proposed for comprehensive assessment of the quality of requirements specifications. Davis et al. proposed a set of 24 quality attributes and associate measures for 18 attributes. But how these measures were used to measure the quality of requirements specification is practically not presented in the paper.. For example, the authors proposed a measure for ambiguity as the ratio of unambiguous to all requirements without providing the further details. Kenett proposed an assessment model to measure the quality which is based on measurement of natural-language specifications. This model relies on identifying certain parts of the sentences, called attributes by the authors, such as initiators of an action, actions, objects, conditions, constraints, and so on. The proposed model includes base measures, for example., the ratio of missing conditions to all sentences requiring a condition, as well as compound measures, which are obtained as a weighted sum of base measures. Unfortunately, the authors have not specified the procedure for the data collection. Overhage et al. proposes the 3QM-framework to assesses the quality of business process models. Along with individual measures, weights are also associated with each measure to obtain the overall quality assessment. Monperrus et al. performed a study on requirements metrics in order to know what information is necessary to apply measures of requirements quality. But, the scope of the study was limited and particularly the study excludes the measures which collect data on a syntactic level or which are "natural-language based".

The present invention, referring to Figure 1, illustrates a novel model for measuring and improving the quality of software requirements specification (SRS) comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS.

There are some models for measuring and improving the quality of software requirements specification (SRS) where there exist some drawbacks in quality of software requirements specification (SRS). Some of the work listed in the prior art is as follows:

US9311218 - *Method and apparatus for the determination of a quality assessment of a software code with determination of the assessment coverage,* presents “In a method for determining a quality assessment of a software code, the coverage is concomitantly calculated when determining the assessment. In order to increase the coverage, additional measurement results and assessments may be taken into account. Following changes to the software base, it is determined which of the additional measurements and assessment results should be renewed in order to provide or ensure the defined coverage.”

US9967211 - *Metric for automatic assessment of conversational responses,* states*”* Examples are generally directed towards automatic assessment of machine generated conversational responses. Context-message-response n-tuples are extracted from at least one source of conversational data to generate a set of multi-reference responses. A response in the set of multi-reference responses includes it context-message data pair and rating. The rating indicates a quality of the response relative to the context-message data pair. A response assessment engine generates a metric score for a machine-generated response based on an assessment metric and the set of multi-reference responses. The metric score indicates a quality of the machine-generated conversational response relative to a user-generated message and a context of the user-generated message. A response generation system of a computing device, such as a digital assistant, is optimized and adjusted based on the metric score to improve the accuracy, quality, and relevance of responses output to the user.”

US7908583 - *Evidentiary enrichment of traceability links between software specification requirements,* states “Traceability links between software specification requirements are evidentially enriched. A traceability link indicates that a second specification requirement is dependent to some degree on a first specification requirement. A likelihood that the second specification requirement for software changes due to the first specification requirement for the software changing, and/or a degree of change of the second specification requirement due to the first specification requirement changing, are determined. A coupling value for the traceability link may be determined as directly proportional to the likelihood that the second specification requirement changes due to the first specification requirement changing, and to the degree of change of the second specification requirement due to the first specification requirement changing."

US8572574 - *Solving hybrid constraints to validate specification requirements of a software module,* states "In one embodiment, a software module is validated according to requirements associated with the software module. The software module has numeric and string variables, and is associated with first numeric constrains and first string constraints. Second numeric constraints applying to specific numeric variables and second string constraints applying to specific string variables are inferred. Each numeric constraint is represented with an equation, and each string constraint is represented with a finite state machine. Attempt to solve a solution for the numeric and string variables that satisfies all the first and second numeric constraints, all the first and second string constraints, and all the requirements associated with the software module by iteratively testing different possible values for the numeric and string variables."

Groupings of alternative elements or embodiments of the invention disclosed herein are not to be construed as limitations. Each group member can be referred to and claimed individually or in any combination with other members of the group or other elements found herein. One or more members of a group can be included in, or deleted from, a group for reasons of convenience and/or patentability. When any such inclusion or deletion occurs, the specification is herein deemed to contain the group as modified thus fulfilling the written description of all related groups used in the appended claims.

The above information disclosed in this background section is only for enhancement of understanding of the background of the invention and therefore it may contain information that does not form the prior art that is already known in this country to a person of ordinary skill in the art.

SUMMARY OF INVENTION

The present invention, referring to Figure 1, illustrates a novel model for measuring and improving the quality of software requirements specification (SRS) comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS. Software requirements specification is the most important document in software development which will be produced in the requirements engineering phase. This document will drive the entire software development process. If we can produce quality requirements specification it leads to success of the project. If it is not quality software requirements specification it will leads to failure of the project. Hence getting the quality SRS is most important activity in software development at requirements engineering phase. For this different approaches are proposed. Each approach has its own advantages and disadvantages. In view of this few metrics are developed to assess the quality of SRS. These metrics are generalized and they can be applied to any SRS irrespective of the type of the project. These metrics are applied on three case studies and the results are satisfactory.

The success of the software project always depends on the software requirements specification. The right software requirements specification always leads to success of software project. In this invention we have presented metrics to measure the quality of software requirements specification. These metrics are used to measure the quality of software requirements specification by using the inputs received by various stakeholders of the software system. The metrics are successfully applied on three case studies and the results are presented in this invention.

BRIEF DESCRIPTION OF SYSTEM

The Accompanying Drawings are included to provide further understanding of the invention disclosed here, and are incorporated in and constitute a part this specification. The drawing illustrates exemplary embodiments of the present disclosure and, together with the description, serves to explain the principles of the present disclosure. The Drawings are for illustration only, which thus not a limitation of the present disclosure.

The present invention, referring to Figure 1, illustrates a novel model for measuring and improving the quality of software requirements specification (SRS) comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS.

Referring to Figure 2, illustrates metric values of software requirements specification (LMS, OSS and ATM) visualizes the metrics.

Referring to Figure 3, illustrates software requirements specification quality of LMS, OSS and ATM.

Referring to Figure 1, illustrates a novel model for measuring and improving the quality of software requirements specification (SRS), in accordance with an exemplary main embodiment of the present disclosure.

Referring to Figure 2, illustrates metric values of software requirements specification (LMS, OSS and ATM), in accordance with another exemplary embodiment of the present disclosure.

Referring to Figure 3, illustrates software requirements specification quality of LMS, OSS and ATM, in accordance with another exemplary embodiment of the present disclosure.

DETAIL DESCRIPTION OF THE PRESENT INVENTION

The following detailed description will make the invention more well-known, and objects other than those indicated below will become apparent. This description is based on the appended drawings. The invention will become more well-known as a result of the following full description, and objects other than those described below will become evident. The drawings that come with the invention are described in this section. It's also important to note that extra or alternative measures should be adopted. A skilled person in the art will be able to completely understand the current disclosure when embodiments are provided. Several specifics relating to various components and processes are described in order to provide a thorough understanding of embodiments of the current disclosure. The information provided in the embodiments should not be viewed as restricting the scope of this disclosure, as those skilled in the art will understand. The order of steps revealed in this invention's process and method should not be taken to imply that the order defined or illustrated is required. Alternatives or additional steps should also be considered.

The present invention herein is a novel model for measuring and improving the quality of software requirements specification (SRS) is explored, a novel model for measuring and improving the quality of software requirements specification (SRS) is provided in the following layout that explains the entire view of the implementation of the invention that provides metrics for the assessment of the quality of software requirements specification.

The quality of Software requirements specification can be measured based on the characteristics of good SRS which are specified in IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998). The characteristics of good SRS are: Correct, Unambiguous, Complete, Consistent, and Ranked for importance and/or stability, Verifiable, Modifiable, and Traceable. Based on the characteristics of good SRS we have proposed metrics for each characteristic to measure the SRS.

<u>Correct</u>: SRS is said to be correct if and only if, every requirement stated in the software requirements specification contributes to the satisfaction of user needs. Correctness can be measured with respect to individual requirement of the specification and percentage of satisfaction of the need.

i. **Needs (**Cor_{needs}**):** A Requirement or if the SRS reflects the user need/actual need then it is correct.

ii. **Comparison (**Cor_{com}**):** A Requirement or if the SRS agrees with the standards and project documentation then it is correct.

iii. **Constraints (**Cor_{consr}**):** The SRS must accurately and precisely identify the conditions and limitations of all situations that the desired capability will encounter

$$\mathbf{M_1 = \forall_{i=1}^{N_r} [\textstyle\sum_{i=1}^{n} Cor_{needs} * n\text{-}1 + \sum_{i=1}^{n} Cor_{com} * n\text{-}1 + \sum_{i=1}^{n} Cor_{consr} * n\text{-}1] * [n_{sf} * N_r]^{-1}} \quad \mathbf{(1)}$$

Where, Nr is Total no. of Requirements and n_{sf} no. of sub factors.

<u>Unambiguous</u>: An SRS is said to be unambiguous if every requirement stated in the specification should have only one interpretation. So that it cannot lead to any confusion in understanding of requirements for both who create it and those who use it.

i. Word level ambiguity (Una_{wa})

ii. Sentence level ambiguity (Una_{sa})

iii. Domain ambiguity(Una_{da})

$$\mathbf{M_2 = \forall_{i=1}^{N_r} [\textstyle\sum_{i=1}^{n} Una_{wa} * n^{-1} + \sum_{i=1}^{n} Una_{sa} * n^{-1} + \sum_{i=1}^{n} Una_{da} * n^{-1}] * [n_{sf} * N_r]^{-1}} \quad (2)$$

Where, Nr is Total no. of Requirements and n_{sf} no. of sub factors.

<u>Complete:</u> An SRS is complete if, and only if, it includes all the requirements that is expected that software system supposed to do. The requirements specification should include all the functional requirements and non-functional requirements along with performance constraints, design constraints, interface requirements and other requirements if any.

i. **Requirements** (Com_{req})**:** An SRS is complete if all types of requirements are included i.e Functional, Non-functional, Interface and Data requirements.

ii. **Definition** (Com_{def}): A Requirement is complete if responses of the requirement to all realizable classes of input data in all realizable classes of situations and definitions and measures of all terms of that requirement stated explicitly.

iii. **Labeling/Referencing** (Com_{lab})**:** An SRS is complete if full labels and references to all figures, tables and diagrams are stated explicitly.

iv. **Undetermined** (Com_{und}): A requirement is complete if there is not "To Be Determined" section

v. **Constraints** (Com_{cons})**:** A requirement is complete if constraints on that requirement are stated explicitly

$$\mathbf{M_3}=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathbf{Com_{req}}*\mathbf{n}^{-1}+\sum_{i=1}^{n} \mathbf{Com_{def}}*\mathbf{n}^{-1}+\sum_{i=1}^{n} \mathbf{Com_{lab}}*\mathbf{n}^{-1}+$$

$$\sum_{i=1}^{n} \mathbf{Com_{und}}*\mathbf{n}^{-1}+\sum_{i=1}^{n} \mathbf{Com_{cons}}*\mathbf{n}^{-1}] * [\mathbf{n_{sf}} * \mathbf{N_r}]^{-1} \qquad (3)$$

Where, Nr is Total no. of Requirements and n_{sf} no. of Sub Factors

Consistent: Consistency means the requirements stated in the document should not have any contradictory definitions.

i. **Internal Consistency** (Cons_{ic})**:** Internal consistency refers to there should be no conflicts between the requirements within the SRS document.

ii. **External Consistency** (Cons_{ec})**:** External consistency refers to SRS document must be compatible with all its related other documents.

$$\mathbf{M_4} = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathbf{Cons_{ic}}*\mathbf{n}^{-1} +\sum_{i=1}^{n} \mathbf{Cons_{ec}}*\mathbf{n}^{-1}] * [\mathbf{n_{sf}} * \mathbf{N_r}]^{-1} \quad (4)$$

Where, Nr is Total no. of Requirements and n_{sf} no. of sub factors

Ranked for importance and/or stability: All the requirements may not be equally important related to a software product. Some requirements may be important and some may be desirable. So it is necessary to rank the requirements for importance or stability by providing an identifier to indicate its importance or stability.

i. **Classification** (Ran_{Clas})**:** Each requirement in the SRS has an identifier which indicates its importance or stability.

ii. **Constancy** (Ran_{cons})**:** Constancy can be expressed in terms of the number of expected changes to any requirement based on the experience or knowledge of forthcoming events that effects the organization, functions and people supported by the software system.

iii. **Inevitability** (Ran_{inev}): Based on the necessity, requirements are ranked by dividing them in to different classes such as essential, optional and conditional requirements.

$$\mathbf{M_5}{=}\forall_{\mathbf{i=1}}^{\mathbf{N_r}}[\textstyle\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ran_{class}}{*}\mathbf{n}^{-1}{+}\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ran_{cons}}{*}\mathbf{n}^{-1}{+}\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ran_{inev}}{*}\mathbf{n}^{-1}] * [\mathbf{n_{sf}} * \mathbf{N_r}]^{-1} \quad (5)$$

Where, Nr is Total no. of Requirements and n_{sf} no. of Sub Factors

Verifiable: Requirements specified in the document should always be testable. If all the requirements are testable then only we can know whether system is working as per needs of the user.

i. **Ambiguity** (Ver_{amb})**:** An ambiguous requirement is not verifiable.

ii. **Vagueness** (Ver_{vag})**:** Usage of concrete terms increases the verifiability.

iii. **Immeasurability** (Ver_{imm})**:** Usage of measurable quantities increases the verifiability.

iv. **Testability:** Testability of a requirement leads to verifiability.

$$\mathbf{M_6}{=}\forall_{\mathbf{i=1}}^{\mathbf{N_r}}[\textstyle\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ver_{amb}}{*}\mathbf{n}^{-1}{+}\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ver_{vag}}{*}\mathbf{n}^{-1}{+}\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ver_{imm}}{*}\mathbf{n}^{-1}{+}\sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{Ver_{test}}{*}\mathbf{n}^{-1}] * [\mathbf{n_{sf}} * \mathbf{N_r}]^{-1} \quad (6)$$

Where, Nr is Total no. of Requirements and n_{sf} no. of sub factors.

Modifiable: Every SRS document must be modifiable. In modern software development there is a every possibility for changing the requirements. Whenever change is required the SRS should allow the changes without effecting the other requirements, its structure and style.

i. **Organization** (Mod_{org}): If the SRS is coherent and easy-to- use organization with a table of contents, an index and explicit cross-referencing then it can be easily modifiable.

ii. **Redundancy** (Mod_{red})**:** If same requirement appears at different places then modifiability becomes tedious process.

iii. **Intermixing** (Mod_{inter})**:** If requirements are intermixed then change in one requirement may affect the other requirement. So to increase modifiability we need to decrease the percentage of intermixed requirements.

$$\mathbf{M_7 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} Mod_{org}*n\text{-}1+\sum_{i=1}^{n} Mod_{red}*n\text{-}1+\sum_{i=1}^{n} Mod_{inter}*n\text{-}1] * [n_{sf} * N_r]^{-1}} \quad \mathbf{(7)}$$

Where, Nr is Total no. of Requirements and n_{sf} no. of sub factors.

Traceable: Traceability is the tracking of requirements throughout the product development lifecycle. It is a documented thread that provides forward and backward visibility into all activity surrounding each requirement (including design, development, testing, and support). Requirements traceability helps minimize the risk of negative outcomes and maximize productivity. Its benefits include greater team efficiency, easier regulatory compliance, and higher-quality products.

i. **Backward traceability** (Tra_{bt}): If the origin of the each of its requirement is clear and if it facilitates the referencing of each requirement in its future development, then requirements can be easily traceable.

ii. **Forward traceability** (Tra_{ft})**:** If each requirement is identified either by name or number then traceability becomes easy.

$$\mathbf{M_8 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} Tra_{bt}*n\text{-}1 +\sum_{i=1}^{n} Tra_{ft}*n\text{-}1] * [n_{sf} * N_r]^{-1}} \quad \mathbf{(8)}$$

Where, Nr is Total no. of Requirements and n_{sf} no. of sub factors.

The present invention, referring to Figure 1, illustrates a novel model for measuring and improving the quality of software requirements specification (SRS) comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS.

Software requirements validation is done based on the check list supplied to stake holders. Each stakeholder will provide their opinion in the form answers to the questions in the check list for each quality factor and sub factors. The proposed metric value for each character is computed based on the answers provided by the stakeholders. The proposed metrics is applied on three different case studies like the library management system, online shopping system and Automatic Teller Machine (ATM) system. The information collected from different people was used in validation.

Stakeholders may include Customer/Client, End Users, Business Analyst, SRS Experts, Managers, Developers, Sponsors and Vendors and Suppliers.

Case Studies: The proposed approach was applied on three case studies. First one is library management system, second one is online shopping system and third one is ATM system. Different stakeholders may use these systems where each stakeholder will have different requirements from various perspectives. The software requirements specifications of three case studies have been represented in brief manner. The proposed metrics for measuring quality of SRS is applied on three case studies. In order to compute the metrics the necessary input is collected from various stakeholders of the respective applications. The results and comparisons are provided with disclosure.

Library Management System (LMS): For this library management system different stakeholders are there like borrower, librarian etc. Each stakeholder is defined with their own requirements. The brief software requirements specification is given in below table-1.

TABLE 1

Brief software requirements specification for Library Management System (LMS)

System	Library management system
All functional requirements	Login, Search book, View catalogue, Reserve book, Return book, Barrow book, Pay fine, Update database, Add item, Register member, Issue book , Check reports.
All Sub functional requirements	Employee login, Publisher login, Student login, View by subject, View by course, View by publisher, Search book by author , Search book by ISBN, Search book by title, Request for book, Add journal, Add book, Add Cd's, Register faculty, Register student, Register publisher, Update barrower information, Update book information, Get book ,Edit report ,View the report.
All Non-Functional Requirements	Login response time must be <2 sec, Search book within 5 sec, Response time for select is 6 sec, Get book within 10 sec, Provide details within 5 sec, Response time for return book <10 sec, Add 10 students per 2 min, Add 10 employees per 2min, Add 10 publishers per 3 min, Add 10 books per 2min, Add 10 journals per 2 min, Add 20 CD's per 2 min, Verification must be fast, Issue book must be < 1 min, Update account for every transaction, Details should verified within 2sec, Update account for every 10 min.

a) **Correct**:

$$M_1 =\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Cor}_{needs}*n^{-1}+\sum_{i=1}^{n} \mathrm{Cor}_{com}*n^{-1}+\sum_{i=1}^{n} \mathrm{Cor}_{consr}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_1 =[50+50+50]*[3*50]^{-1} = [150]*[150]^{-1}=1$$

b) **Unambiguous**:

$$M_2 =\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Una}_{wa}*n^{-1} +\sum_{i=1}^{n} \mathrm{Una}_{sa}*n^{-1}+\sum_{i=1}^{n} \mathrm{Una}_{da}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_2 = [48+44+46]*[50]^{-1} =0.92$$

c) **Complete:**

$$M_3 =\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Com}_{req}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{def}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{lab}*n^{-1}+$$
$$\sum_{i=1}^{n} \mathrm{Com}_{und}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{cons}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_3= [50+50+50+50+50]*[5*50]^{-1} = [250]*[150]^{-1}=1$$

d) **Consistent:**

$$M_4 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Cons}_{ic}*n^{-1} +\sum_{i=1}^{n} \mathrm{Cons}_{ec}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_4 = [45+41]*[2*50]^{-1} = [86]*[100]^{-1} = 0.86$$

e) **Ranked for importance and/or stability:**

$$M_5=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Ran}_{class}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ran}_{cons}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ran}_{inev}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_5 = [44+46+42]*[3*50]^{-1} =[132]*[150]-^{1}= 0.88$$

f) **Verifiable:**

$$M_6 =\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Ver}_{amb}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{vag}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{imm}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{test}*n^{-1}] *$$
$$[n_{sf} * N_r]^{-1}$$

$$M_6 = [46+48+50+44]*[4*50]^{-1} =[188]*[200]-^{1} =0.94$$

g) **Modifiable:**

$$M_7 =\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Mod}_{org}*n^{-1}+\sum_{i=1}^{n} \mathrm{Mod}_{red}*n^{-1}+\sum_{i=1}^{n} \mathrm{Mod}_{inter}*n^{-1}] *$$
$$[n_{sf} * N_r]^{-1}$$

$$M_7 = [42+40+44]*[3*50]^{-1} =[126]*[150]^{-1} =0.84$$

h) **Traceable:**

$$M_8 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Tra}_{bt}*n^{-1} +\sum_{i=1}^{n} \mathrm{Tra}_{ft}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_8 = [47+43]*[2*50]^{-1} =[90]*[100] =0.90$$

Final metric value $M=[M_1+M_2+M_3+M_4+M_5+M_6+M_7+M_8]*[8]^{-1}$

$$M= \frac{1+0.92+1+0.86+0.88+0.94+0.84+0.9}{8} = \frac{7.4}{8} = 0.9175$$

Online Shopping System (OSS): Online shopping is an application which provides various services to customers to purchase goods through internet directly from sellers. The main stakeholders in online shopping system are such as system and customer.

Each stakeholder is defined with their own requirements. The brief software requirements specification is given in below table-2.

TABLE 2

Brief software requirements specification for Online Shopping System

System	**Online shopping system**
All functional requirements	Login, Select product category , Select product, Add to cart, Check the shopping cart, Checkout, Confirm the order
All Sub functional requirements	User login, New user registration, Search product by name, Search product by type, Search product by category, View details, Select the item, Provide bill details, Provide delivery details, Provide payment details, Confirm the order, Receive input, Display products , Display product details, Receive details, Process the order.
All Non-Functional Requirements	Login response time must be < 2 sec, Search product should be completed within 5 sec, Time for add item is 6 sec, Checkout should process large amount data, Process order within 7 sec, Payment must be completed within 30 sec, Checkout within 5 sec.

a) **Correct**:

$$M_1=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} Cor_{needs}*n^{-1}+\sum_{i=1}^{n} Cor_{com}*n^{-1}+\sum_{i=1}^{n} Cor_{consr}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_1 =[30+30+30]*[3*30]^{-1} =[90]*[3*30]^{-1} = 1$$

b) **Unambiguous**:

$$M_2=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} Una_{wa}*n^{-1}+\sum_{i=1}^{n} Una_{sa}*n^{-1}+\sum_{i=1}^{n} Una_{da}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_2 = [28+25+25]*[3*30]^{-1} =[78]*[90]^{-1} =0.86$$

c) **Complete:**

$$M_3=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} Com_{req}*n^{-1}+\sum_{i=1}^{n} Com_{def}*n^{-1}+\sum_{i=1}^{n} Com_{lab}*n^{-1}+$$

$$\sum_{i=1}^{n} \mathrm{Com}_{und}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{cons}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_3= [30+30+30+30+30]*[5*30]^{-1}=[150]*[5*30]^{-1} = 1$$

d) Consistent:

$$M_4 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Cons}_{ic}*n^{-1} +\sum_{i=1}^{n} \mathrm{Cons}_{ec}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_4 = [24+22]*[2*30]^{-1} =[46]*[60]^{-1} =0.76$$

e) Ranked for importance and/or stability:

$$M_5=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Ran}_{class}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ran}_{cons}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ran}_{inev}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_5 = [24+22+26]*[3*30]^{-1} =[72]*[90]^{-1} =0.80$$

f) Verifiable:

$$M_6=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Ver}_{amb}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{vag}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{imm}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{test}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_6 = [26+27+28+27]*[4*30]^{-1}=[108]*[120]^{-1} =0.90$$

g) Modifiable:

$$M_7=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Mod}_{org}*n^{-1}+\sum_{i=1}^{n} \mathrm{Mod}_{red}*n^{-1}+\sum_{i=1}^{n} \mathrm{Mod}_{inter}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_7 = [27+23+25]*[3*30]^{-1} =[75]*[90]^{-1} =0.83$$

h) Traceable:

$$M_8 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Tra}_{bt}*n^{-1} +\sum_{i=1}^{n} \mathrm{Tra}_{ft}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_8 = [27+25]*[2*30]^{1}=[52]*[60]^{-1} =0.86$$

Final metric value $M= [M_1+M_2+M_3+M_4+M_5+M_6+M_7+M_8]*[8]^{-1}$

$$M=\frac{1+0.86+1+\ .76+0.8+0.9+0.83+0.86}{8} = \frac{7.01}{8} = 0.87625$$

<u>ATM System (ATM)</u>: ATM system is a device which provides various financial services to users, such as withdrawing, transferring, depositing the money and other payments of different banks and institutions. The brief software requirements specification is given below table-3.

TABLE 3

Brief software requirements specification for ATM System

System	ATM System
All functional requirements	Withdraw money, Check balance, Deposit money, Transfer money, Get mini statement, Display Information, Verify card, Process transaction, Update account, Update database, Print receipt, Change password.
All Sub functional requirements	Enter pin number, Send result, Receive pin number, Verify pin number, Select account type, Enter amount, Check account type, Process transaction, Dispatch cash, Receive cash, Select option, Takes envelope, Put cash, Place the envelope, Select account, Enter amount, Send money, Display account information, Receive details Update details.
All Non-Functional Requirements	Select account in 15 sec, Enter amount within 10 sec, Display information in 5sec, Customer should take money within 30 sec, Transfer money should be done 6sec, Transaction process must be completed within 5 sec, Update data base for every 2 min, Print receipt within two 5sec, Update system every week, Check details in 10 sec, Send money in 10 sec, Enter pin in 10 sec, Confirm pin in 5 sec.

a) **Correct**:

$$M_1=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Cor}_{needs}*n^{-1}+\sum_{i=1}^{n} \mathrm{Cor}_{com}*n^{-1}+\sum_{i=1}^{n} \mathrm{Cor}_{consr}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_1 =[45+45+45]*[3*45]^{-1}=[135]*[135]^{-1} = 1$$

b) **Unambiguous**:

$$M_2=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Una}_{wa}*n^{-1} +\sum_{i=1}^{n} \mathrm{Una}_{sa}*n^{-1}+\sum_{i=1}^{n} \mathrm{Una}_{da}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_2 = [41+40+39]*[3*45]^{-1} =[120]*[135]^{-1} =0.88$$

c) **Complete:**

$$M_3=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Com}_{req}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{def}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{lab}*n^{-1}+$$

$$\sum_{i=1}^{n} \mathrm{Com}_{und}*n^{-1}+\sum_{i=1}^{n} \mathrm{Com}_{cons}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_3= [45+45+45+45+45]*[4*45]^{-1} =[225]*[225]^{-1} = 1$$

d) Consistent:

$$M_4 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Cons}_{ic}*n^{-1} +\sum_{i=1}^{n} \mathrm{Cons}_{ec}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_4 = [40+36]*[2*45]^{-1} =[76]*[90]^{-1} =0.84$$

e) Ranked for importance and/or stability:

$$M_5=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Ran}_{class}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ran}_{cons}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ran}_{inev}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_5 = [37+39+35]*[3*45]^{-1}=[111]*[135]^{-1} =0.82$$

f) Verifiable:

$$M_6= \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Ver}_{amb}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{vag}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{imm}*n^{-1}+\sum_{i=1}^{n} \mathrm{Ver}_{test}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_6 = [40+41+43+40]*[4*45]^{-1} =[164]*[180]^{-1} =0.91$$

g) Modifiable:

$$M_7=\forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Mod}_{org}*n^{-1}+\sum_{i=1}^{n} \mathrm{Mod}_{red}*n^{-1}+\sum_{i=1}^{n} \mathrm{Mod}_{inter}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_7 = [38+35+32]*[3*45]^{-1}=[105]*[135]^{-1} =0.77$$

h) Traceable:

$$M_8 = \forall_{i=1}^{N_r}[\sum_{i=1}^{n} \mathrm{Tra}_{bt}*n^{-1} +\sum_{i=1}^{n} \mathrm{Tra}_{ft}*n^{-1}] * [n_{sf} * N_r]^{-1}$$

$$M_8 = [40+38]*[2*45]^{-1}=[78]*[90]^{-1} =0.86$$

Final metric value $M= [M_1+M_2+M_3+M_4+M_5+M_6+M_7+M_8]*[8]^{-1}$

$$M= \frac{1+0.88+1+0.84+0.82+\ .91+0.77+0.86}{8} = \frac{7.08}{8} = 0.885$$

Results:

We have systematically applied defined metrics on the three case studies and the results were so useful to decide and take decision to move forward for development or rewrite entire SRS or some portions of SRS. The ideal Quality metric value is 1.If metric value is0.9 and above we can proceed with development. If it is 0.8 and above, need to identify and verify the few portions based on the values of individual metrics. If the quality metric value is 0.7 and above, careful examination of entire document is required and if it is below 0.7, better to rewrite the entire document with the help of individual metric values. Metric values are served as indicators for accepting or rejecting the SRS or else it helps to identify the portions of SRS where deep insight is needed. Metric values and its indicators are given in the below table-4 and table-5.

TABLE 4

Quality Metric value and its indication

Quality Metric Value	Indication
1	Acceptable
<1 and >=0.9	Acceptable with Minor Changes
<0.9 and >=0.8	Conditional acceptance with few Portions to be rechecked
<0.8 and >=0.7	Many portions are to be rechecked based on individual Metric Values
<0.7	Reject with Constructive Suggestions

TABLE 5

Individual and Final Quality Metric values of LMS, OSS and ATM

Metric/Case Study	LMS	OSS	ATM
Correct	1	1	1
Unambiguous	0.92	0.86	0.88
Complete	1	1	1
Consistent	0.86	0.76	0.84
Ranked for importance and/or stability	0.88	0.8	0.88
Verifiable	0.94	0.9	0.91
Modifiable	0.84	0.83	0.77
Traceable	0.9	0.86	0.86
Final Quality Metric Value	0.9175	0.87625	0.885

Referring to Figure 2, illustrates metric values of software requirements specification (LMS, OSS and ATM) visualizes the metrics. This graph represents the individual Metric values of SRS documents of LSM, OSS and ATM. These values represent the strengths and weaknesses of different characteristics of SRS document. This data will provide the insight into the SRS document, in which characteristic our SRS is strong and in which characteristic our SRS is weak. This insight is more helpful for improving the particular portions of the SRS document. Referring to Figure 3, illustrates software requirements specification quality of LMS, OSS and ATM. This graph represents the final Quality of three SRS documents, in which the Quality of Library Management System (LMS) SRS is better than other two SRS documents of Online Shopping System (OSS) and ATM Systems. As per quality report, with the help of individual metric values OSS and ATM, portions of SRS need to be identified and improved.

It is to be understood that the above description is intended to be illustrative, and not restrictive. For example, the above-discussed embodiments may be used in combination with each other. Many other embodiments will be apparent to those of skill in the art upon reviewing the above description. The benefits and advantages which may be provided by the present invention have been described above with regard to specific embodiments. These benefits and advantages, and any elements or limitations that may cause them to occur or to become more pronounced are not to be construed as critical, required, or essential features of any or all of the embodiments.

MEKA JAMES STEPHEN Digitally signed by MEKA JAMES STEPHEN
Date: 2022.11.20 15:42:37 +05'30'

CLAIMS

We claim:

1. A novel model for metrics to assess the quality of software requirements specification comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS in the present invention.

2. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein few metrics exist to evaluate SRS quality. These metrics can be applied to any SRS, regardless of project type.

3. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein The characteristics of good SRS are correct, unambiguous, complete, consistent, ranked for importance and/or stability, verifiable, modifiable, and traceable.

4. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein the proposed metrics is applied on three different case studies like the library management system, online shopping system and Automatic Teller Machine (ATM) system.

5. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein the information collected from different people was used in validation. Stakeholders may include Customer/Client, End Users, Business Analyst, SRS Experts, Managers, Developers, Sponsors and Vendors and Suppliers.

6. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein the library management system has different stakeholders are there like borrower, librarian etc. Each stakeholder is defined with their own requirements.

7. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein the online shopping is an application

which provides various services to customers to purchase goods through internet directly from sellers. The main stakeholders in online shopping system are such as system and customer.

8. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein ATM system is a device which provides various financial services to users, such as withdrawing, transferring, depositing the money and other payments of different banks and institutions.

9. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein we have systematically applied defined metrics on the three case studies and the results were so useful to decide and take decision to move forward for development or rewrite entire SRS or some portions of SRS.

10. A novel model for metrics to assess the quality of software requirements specification as claimed in claim 1, wherein the ideal Quality metric value is 1.If metric value is0.9 and above we can proceed with development. If it is 0.8 and above, need to identify and verify the few portions based on the values of individual metrics. If the quality metric value is 0.7 and above, careful examination of entire document is required and if it is below 0.7, better to rewrite the entire document with the help of individual metric values. Metric values are served as indicators for accepting or rejecting the SRS or else it helps to identify the portions of SRS where deep insight is needed.

MEKA JAMES STEPHEN Digitally signed by MEKA JAMES STEPHEN
Date: 2022.11.20 15:43:11 +05'30'

Dated this 20th day of November, 2022

A NOVEL MODEL FOR METRICS TO ASSESS THE QUALITY OF SOFTWARE REQUIREMENTS SPECIFICATION

ABSTRACT

The requirements engineering phase will provide the software requirements specification. It will guide software development. Quality requirements specifications lead to project success. Poor software requirements specification will cause project failure. Thus, quality SRS (software requirements specification) is the most significant requirement engineering activity. The present invention disclosed herein is a novel model for metrics to assess the quality of software requirements specification comprising of: Re-Expert team (101); Instruction with stake holders (102); Quality factors (103); Identification of sub-factors (104); Formulation of metrics (105); SRS validation by the stake holders using a check list (106); Application of metrics and measurements (107); SRS quality report preparation (108); and directions for improving the quality of the SRS (109); provides various ways to assess the quality of SRS in the present invention. Each has pros and cons. Few metrics exist to evaluate SRS quality. These metrics can be applied to any SRS, regardless of project type. Three case studies use these measures, with satisfactory outcomes.

MEKA JAMES STEPHEN Digitally signed by MEKA JAMES STEPHEN
Date: 2022.11.20 15:43:26 +05'30'

Dated this 20th day of November, 2022

DRAWINGS

Total No of Sheets: 02

Sheet No.1

Applicant: Andhra University.

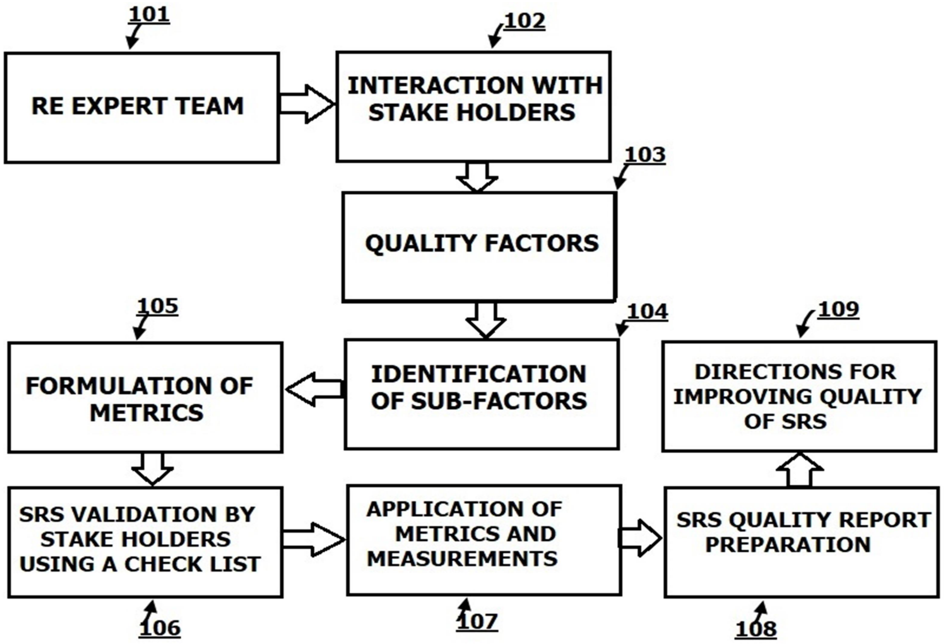

101
RE EXPERT TEAM
102
INTERACTION WITH STAKE HOLDERS
103
QUALITY FACTORS
104
IDENTIFICATION OF SUB-FACTORS
105
FORMULATION OF METRICS
106
SRS VALIDATION BY STAKE HOLDERS USING A CHECK LIST
107
APPLICATION OF METRICS AND MEASUREMENTS
108
SRS QUALITY REPORT PREPARATION
109
DIRECTIONS FOR IMPROVING QUALITY OF SRS

FIGURE 1: A Novel model for measuring and improving the Quality of Software Requirements Specification

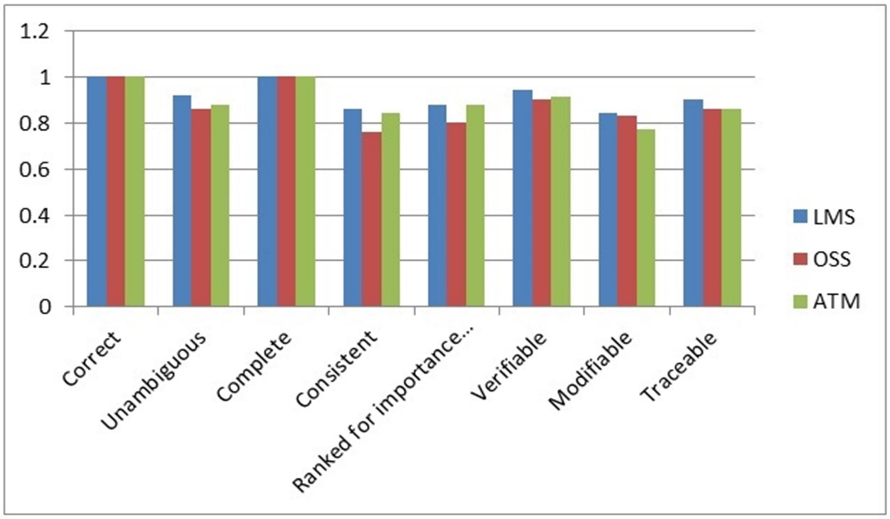


FIGURE 2: Metric Values of Software Requirements Specification.

MEKA JAMES STEPHEN Digitally signed by MEKA JAMES STEPHEN Date: 2022.11.20 15:43:57 +05'30'

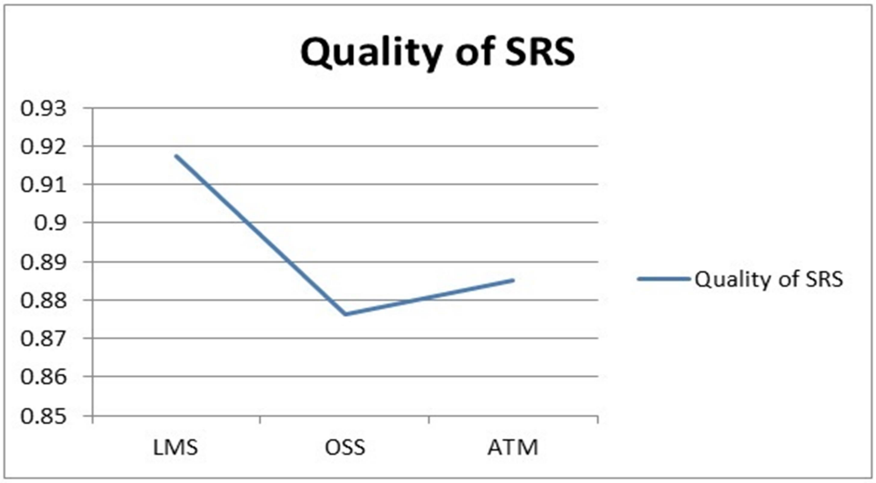


FIGURE 3: Software Requirements Specification Quality of LMS, OSS and ATM.

MEKA JAMES STEPHEN Digitally signed by MEKA JAMES STEPHEN
Date: 2022.11.20 15:44:14 +05'30'