A WEB-BASED MODEL FOR INSPECTION INCONSISTENCIES RESOLUTION: A NEW APPROACH WITH TWO CASE STUDIES

Navid Hashemi Taba[1], Siew Hock Ow[2]

[1, 2] Department of Software Engineering, Faculty of Computer Science and Information Technology
University of Malaya, 50603, Kuala Lumpur, Malaysia

Email: nhtaba@siswa.um.edu.my[1], show@um.edu.my[2]

ABSTRACT

*This research is aimed at introducing an enhanced inspection model known as **Improved Software Inspection Model (ISIM)** by making some enhancements to overcome or reduce the weaknesses identified in Formal model. ISIM includes a method of selecting suitable inspectors based on their expertise and work experience; provide a database that stores information on potentials; has a database that stores the possible causes of each of the defects stored in the defect database; and incorporates a few enhancements in the inspection process. To facilitate the selection of suitable inspectors and the inspection process, an inspection tool was developed using PHP with MySQL as database management system. To determine whether the use of ISIM can improve the inspection process, the quality of the process was evaluated using paired T-test which compares the differences between Formal and ISIM models, in terms of the number of defects detected, inspection time, and the productivity of the inspectors. Two case studies were conducted in two companies and inspections were carried out on two software development projects to collect the data needed for the statistical test. The statistical test results show that at α = 0.05, the mean values of the number of defects are detected; the inspection time; and the productivity of the inspectors when using Formal and ISIM are 19.78 and 28.13; 220.97 minutes and 213.69 minutes; and 0.0891 and 0.1328, respectively. This shows that there are significant improvements in the number of defects detected (i.e. difference = 8.35; 42.21%), and productivity of the inspectors (i.e. difference = 0.0437; 49.05%). However, there is only a slight improvement in the inspection time. It can be concluded that ISIM together with its tool improve the quality of inspection process in the two case studies.*

Keywords: Software Quality, Software Inspection, Software Test, Inspection Model, Inconsistency Elimination

1.0 INTRODUCTION AND BACKGROUND

Software inspection is a technical review that was developed by Michael Fagan (1976) to enhance the quality of software as well as its efficiency and effectiveness [1]. According to him, there are five mandatory phases and one optional operation (acceptance) in a formal software inspection process. As shown in Fig. 1, the five phases are *Overview, Preparation, Inspection, Rework*, and *Follow-up*. Fagan defined four roles in the inspection process, which are *Moderator, designer, coder* and *tester. Moderator* is the coach of the inspection team. The duties of a moderator are scheduling the inspection meeting and reporting the results of inspection. Moreover, follow-up of the reworks after inspection is another responsibility of the moderator. In *overview* operation, the designer will deliver the artifact to moderator and moderator will arrange the inspection process. Moderator also calls the inspector and decides about the resources needed for inspection process. In preparation phase, all inspectors obtain the artifact as well as necessary documents provided by moderators. In meeting phase, all inspectors are involved in meeting session to negotiate the potential defects and specify them. In meeting session, all inspectors participate to recognize the defects. After specifying the defects, the rework phase will start. In this phase, the author (designer) as the owner of artifact is responsible to work on the artifact and makes it free of defects addressed in the meeting. The last mandatory phase of inspection process is the follow-up and the main responsibility is on the moderator to check if all defects are fixed and ask the author for clarifications (if any).
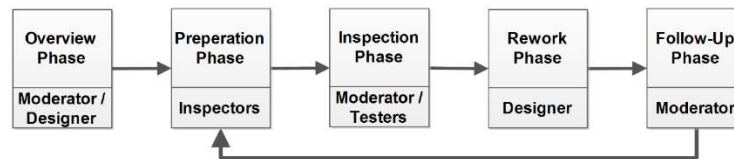
1

Fig. 1. The five phases of Fagan's formal inspection process

The studies show that there is no justification to ensure the companies managers for cost deduction and reduce the defects by inspection rather than using software regular tests [2]. In a comprehensive study, the researchers concludes [3] : " most of the organizations are not able to use inspections in software development process as these are too rigorous to follow, even with computer tool support it is complex for organizations to implement it."

The recent researches on inspection evolution show that the efforts to eliminate meetings for improving efficiency due to disadvantages such as increase the false positives. Although many researches have shown that no-meeting inspection models are more efficient compared with meeting based inspections, no solution is provided for solving the false positive problem [4]. In false positive situation inspectors indicate a defect when it is not [5]. The tools developed for the new inspection models have weakness in coordination support [6].

An inherited and transmitted software inspection problem that is remained after three decades is choosing, evaluation and appraisal of inspectors for an inspection process [7], [8]. None of the existing models focus on the selection procedures, qualification process, or even recommend a placement test of their abilities. From another point of view most of studies till now are done based on some experiments and surveys. In a related comprehensive study [9], researchers stated: "additional research is needed to validate these findings both inside and outside of a laboratory environment."

2.0 PROBLEM STATEMENT AND RESEARCH OBJECTIVE

Based on the study done in the first activity of this research, the main problem of this research stands out. Reliable sources indicate that existing models and tools for software inspection are more focused on inspection of code and program [10], [11]. In other words, less attention has been focused in the models on inspection of artifacts related to the software analysis and design artifacts. Fagan in 2002, three decades after introducing his own formal inspection model, have provided explanations to justify the shortcomings of models in the early stages of software development [12]. He states that inspection of artifacts in the analysis and design stage is an endeavor that still requires highly-skilled human resources, and models can only provide a tool for this matter, an undertaking which has not yet been attempted frequently. Moreover, review of reliable sources show that a comprehensive categorization of potential shortcomings or defects in analysis and design stages has not been done neither has it been recommended for software inspection [13]. Therefore, preparing a model which supports software inspection in analysis and design stages, accompanied by provision of proper categorization of potential defects and also a tool for gathering model data, is the main objective of this research [14].

The objective of this research is to propose a new inspection model to improve the software inspection process, and to evaluate the efficiency of the proposed model. The evaluation was made based on the data collected from two case studies. Hence, it is an applied and experimental research. A variety of spectrum of researches in fields of inspection models show that only 35% of scientific researches focused on code inspection; however, the new era of programming environment aims using the automated tools for code inspection and focusing on web-inspection. In the other words, inspecting the analysis and design of deliverable artifacts and inspection efficiency are remained as the important issues during the past three decades [15].

3.0 RESEARCH METHOD USED

This study aims to evaluate the efficiency of the proposed inspection model. Before conducting the case study, the metrics used to evaluate the proposed model were defined. The following sections explain the research activities in detail.

2

3.1 Step 1: Proposing a New Inspection Model

In light of shortcomings of the current model in software inspection a new inspection model was designed and developed. This model is based on a comprehensive categorization of the defects in the software analysis and design stage. This model provides remote inspection and can support virtual meetings. The factors behind the shortcomings in the analysis and design stage have also been a focus of this model. This model creates a learning process, through discovery and elimination of causal relationship between shortcomings and their causing factors, which will result in constant improvement of the inspection throughout various projects. The proposed model provides better support to software inspectors because of the above-mentioned unique features. This model has been thoroughly elaborated in following sections.

3.2 Step 2: Develop a Web-Based Tool to Support the Proposed Model

To facilitate the implementation of the proposed inspection model, a web-based system tool was developed using agile development techniques under the .NET environment. It has a database that stores the various types of defects which are commonly detected in the requirements analysis and design phases as well as the various potential causes of each defect. This would enable the author(s) of the artifacts to remove the detected defects quickly.

3.3 Step 3: Conduct Case Studies to Appraisal the Proposed Model

To appraise the efficiency and effectiveness of the proposed model, two case studies were conducted in two different computers and industrial enterprises. Three software projects were inspected in the two companies, respectively. The formal structured inspection process was first used by three inspectors (Team A) to find the defects in the artifacts. The defects detected by the inspectors and the inspection time were recorded using Microsoft Excel. Then, the inspection process was repeated by the inspection from team A using the proposed inspection model. The inspection data were collected and recorded using the web system. These two sets of data were compared to determine if the proposed model helps to find more defects, reduce the inspection time, and increase the productivity of the inspectors.

To avoid bias and to ensure validity and reliability of the research [16], both inspections were performed by the same team of inspectors (3 members) under similar environment, and on the same artifacts. These inspection processes were repeated by another team of inspectors (Team B) comprising three inspectors. This approach aims to reduce the bias due to the different knowledge and work experiences in software inspection of the software inspectors.

4.0 IMPROVED SOFTWARE INSPECTION MODEL (ISIM)

In this section, the proposed model to overcome the shortcomings of the existing inspection models is elaborated. The existing approaches in software inspection were mostly based on holding meetings composed of system developers and inspectors in a room to have a formal and step by step inspection of the system documents [17]. Advances in distributed systems technology and web based applications enabled us to hold some virtual sessions [18]. The aforementioned technology facilitates having remote inspections on artifacts and documents, also getting access to distributed databases to save and retrieve the related information and removing the defects. The proposed model is based on an integrated approach to identify the defects in early stages of software development process.

To implement the model proposed in this research, a virtual environment is designed for inspectors to have relations and it is compatible with model attributes. The web-based application enables collaborations between inspectors in inspecting an artifact of software development process. In a same way the collaborative system enables inspectors in each phase to record and register the defects in each inspection case and show the defects causes.

4.1 ISIM Workflow

The work flow in ISIM starts with defining a project. Registering the specifications of a project can be a guide for the artifact inspection process (See Fig. 2). Then, project phases are defined, specifically for this research model the analysis and design phases. The methodology for project implementation, and the models specified for each phase of the project can act as guidelines for the consolidator and the inspectors as well.
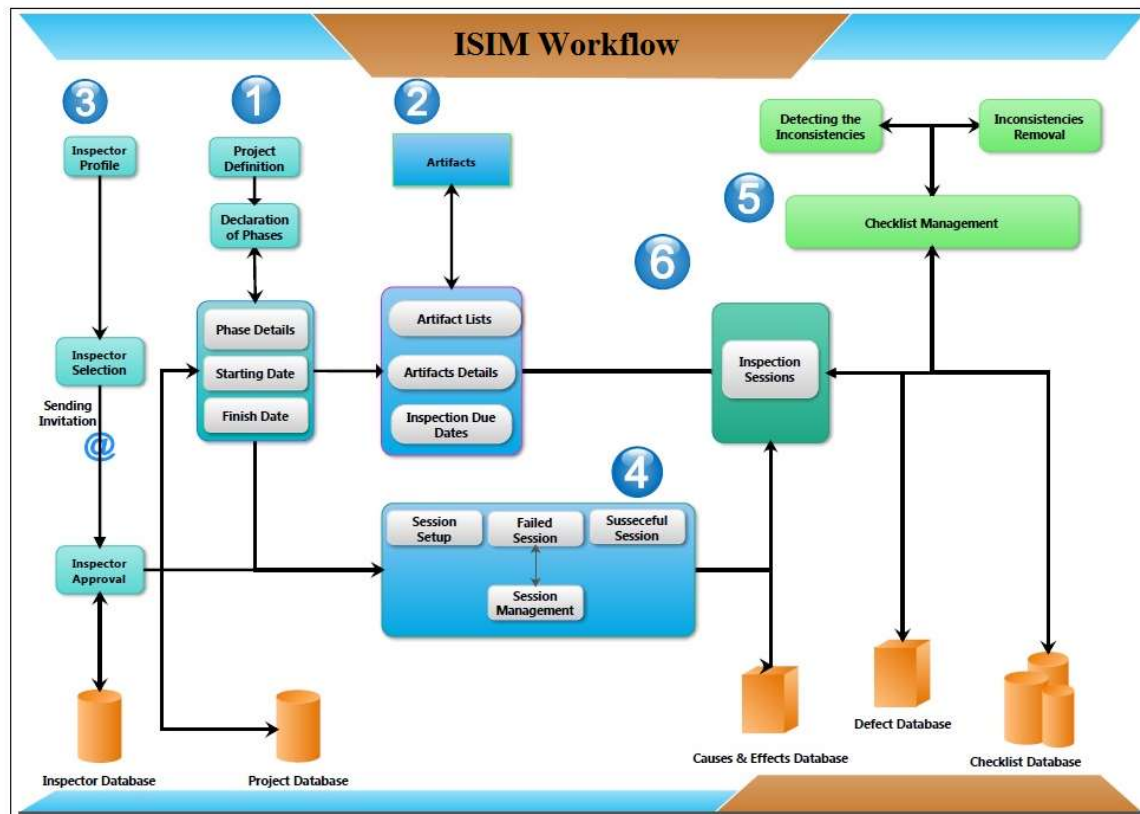
Fig. 2. Workflow of ISIM

In the second stage, the timetable for project phases is determined, where a list of artifacts related to each phase are defined. Documentation related to artifacts are collected, categorized and prepared. Due dates for each artifact are also set. These are the major and distinctive characteristics of the proposed model compared with existing inspection models. That is to say, considering the due date and limitations and other restrictions, artifacts are categorized and as the number of inspectors is limited, those which are priorities are moved to the top of the inspection list. A training package is used to provide the inspectors useful information on the artifact as well as the inspection process.

This stage is the definition of inspectors. Specifically, 3 or 5 inspectors are defined. The number of inspectors must be odd number so that in case of difference of opinion in the inspection process a decisive vote can be cast to avoid a tie. Profiles of the inspectors are gradually completed by this system. The projects and artifacts they have inspected, the number of defects they have detected are registered in their profiles and are updated. Their productivity rate is also determined for each inspection activity. An inspector's productivity is the rate of detected defects in a unit of time. In order to ensure the accuracy of this rate, the number of inspected artifacts and the number of detected defects and the phase of the project they belong to are also registered and announced in conjunction with the defect detection rate. The availability of the inspector for cooperation; is shown in the system, increasing speed of finding candidates for inspection increases. After inspection, candidates are selected, where an invitation is sent to them together with a brief description of the artifact and the deadline. Meanwhile, the inspectors have the opportunity to confirm their willingness to cooperate.

The fourth stage is the submission of a prepared training package to the inspectors. Possible questions are answered and an examination of preparedness and capabilities is conducted to verify that they are ready for inspection and to eliminate all ambiguities. This process can be repeated and unprofessional inspectors could be replaced. All inspectors can inspect a specific artifact simultaneously in a coordinated fashion or if there are time differences or they are otherwise occupied they can inspect the artifact they have received within the specified timeframe any time they can spare. Both options include positive and negative issue. If the inspection activity is conducted simultaneously the inspectors can have a dialogue and the Author will spend less time. Still, if it cannot be arranged and the sessions are

4

held consecutively the Author's experience will be taken from an inspection session to another and this will increase productivity. What more, there would be no repetitions as the Author has addressed the issue in the previous session.

The fifth stage is preparation of an appropriate checklist which can start after the second stage. At this stage the inventory of ready checklists and the inventory of potential defects are used and appropriate checklists for inspection of specific artifacts are designed and prepared. ISIM is supported with a web based system that offers a quick designing of checklists.

In the following stage, the artifacts with all the necessary information, data, guidelines, forms and documentations and the relevant checklists are sent to the inspectors. The producer (author) of the artifact will be available online so the inspectors can ask questions and consult with him/her. Authors were accessible by online tools to larify the ambiguities (if any). In spite of the fact that absence of simultaneity increases the presence of the Author, due to the online nature of the probable conversation, and the inspectors conduct the inspection. The length of time available for an inspection is two hours but can be extended. Ten minutes to the end of the session a warning is sent to the inspectors. At this stage, the consolidator integrates the submitted checklists and reviews them for inconsistencies. The inspectors might disagree on the category of defect or whether the item at hand is indeed a defect. In these circumstances, to eliminate any inconsistencies, the consolidator calls for a vote. The system registers the detected inconsistencies, when two persons agree on a defect that is decided and in other occasions a list of inconsistencies and options are sent to the inspectors and they are asked to vote. Eventually, the finalized list of defects is sent to the Author's for rectification. In addition, a change matrix is also sent so that the Author registers the resolved issues with their cause and the chosen solution. This information is saved in the cause and effect database to be provided to the inspectors for future inspection to accelerate inspection.

4.2 Defects and their Cause's Databases

A defect may be the result of so many causes, sequential or parallel to make the defect [19]. In the cause database, there are some categories to classify the causes and effects searched by keywords. We must pay attention that a cause may result in different causes or defects. For example using some inexperienced system analysts in one hand results in developing low quality analysis reports and may cause insufficient documents on defects. In the previous example, using inexperienced analysts resulted in programmers' inappropriate selection process.

4.3 Eliminating Defect Detection Inconsistencies in ISIM

The web based system refers to codes used by inspectors and flags defects detected separately by each inspector on common defects. Consolidator in the suggested model plays the role of the Mentor in the formal model. Consolidator has the duty to determine the common detected defects. If the cause or explanation provided by inspectors for a single common defect varies with others, the inconsistency must be eliminated. The disputed items will be sent to inspectors by the consolidator to eliminate the inconsistencies. The category of the defect and/or their cause is put to a vote.

4.4 Avoiding Defect Transition

ISIM focused on defects on analysis and design phases. Defects that are removed in early phases of software development process could be assumed as a real assistance to avoid the same failures in later development phases and avoid amplifying the latent defects [20], [21].

5.0 CARRYING OUT THE CASE STUDY

To evaluate the efficiency and effectiveness of the proposed model, comparison made with the Formal model as it is the most popular inspection model [22], two large projects have been selected from two companies. Investigating artifacts of analysis and design phases were performed by two groups of inspectors. An efficiency test done prior to the inspection process shows the capabilities of the inspectors. Moreover, some artifacts will be inspected by the inspectors of the first group formally, and others using the proposed model. The same procedure will be applied to the second group to neutralize the effect of inspectors' weaknesses and strengths resulting from the model efficiency.

5.1 Preparation Stage

For the preparation stage, the checklists in the format of Microsoft Excel forms were provided to the formal model inspectors. Concurrently the proposed model inspectors have used the same checklists through the online tool. The maximum length for the sessions was four hours; two hours for preparation and two hours for formal sessions or consolidator of the new model. If the inspectors decided that all the defects were detected and their work with the artifact was over they could end the session. In the second model a warning was sent to the inspectors 110 minutes inside the inspection and following that their access to the software was cut off exactly after two hours. If an activity was cut short, consolidator could provide the inspector with a short few-minutes-long additional time.

In both models the preparation time for inspection was set equally at two hours. In the second models the inspectors had to start inspection immediately after the artifact and the descriptions was sent to them. In the first model, one person had to introduce the artifact and provided relevant description and explanations. The checklists prepared for both models were identical; however, those who worked with the web system had access to the cause and effect rolls of the defects as well. Additionally, the defects detected in the web based system were kept and the inspectors could benefit the information related to them in future artifacts. This is in fact the learning characteristic of the system and the proposed model.

5.2 Inspection Procedure

Team A in both requirements analysis and design phases of each project, inspected 4 artifacts with the formal model and 4 artifacts with the proposed model using ISIM model. Team B also in each of the analysis and design phases inspected 4 artifacts with the formal model and 4 artifacts using ISIM and according to the proposed model. Therefore, all 32 artifacts of the project were inspected utilizing both models. Team A inspected 16 artifacts and 16 artifacts were inspected by Team B. Alternating the inspection models eliminated the possibility of bias and both teams worked using both models. The order of artifact inspection was altered in each phase of the project in the first phase four artifacts were inspected by team A with the formal model, in the second phase the first four artifacts were inspected by the same team using the ISIM model.

5.3 Validity and Reliability

Reliability or longevity of a research required results to be consistent [23]. The metrics provided in this section have been utilized for assessment of the introduced model. These measurements have provided an accurate and precise assessment of the model and have the internal validity of the research. Bias must reach the minimum possible levels for a research to be expandable and its external credibility to be confirmed. Two principals have been observed to minimize the bias in this research. Firstly, the Projects Artifacts have been selected randomly. Secondly, inspectors with similar experiences and levels of expertise have been recruited. A preliminary training on the utilization of the model and its tool, in addition to a preparedness placement test were organized and held to minimize bias.

5.4 Inspection Time Calculation

The inspection time in the proposed model is the sum of times consumed by individual inspectors to provide the consolidated list to the inspection leader and the time s/he consumes for correcting and returning to inspectors for approval, voting and elimination of inconsistencies. The inspectors leave approval or a comment. To make With the aim of making calculations easier, time has been calculated on a minute-basis, even though web system has the capacity to register seconds and even more details (in order to compare the two models in equal conditions and to avoid numerous details durations are rounded to the minute). Equation 1 illustrates the time calculation for the proposed model [24]. Here the inspection time is the sum of the three times, namely, preparation (by inspectors separately), consolidation (by the consolidator) and time for elimination of probable inconsistencies through resubmission of the form and the checklist to inspectors for voting.

Equation 1. Time calculation

IS= PT+CI+IRT

TIS: Total Inspection Time, PT: Preparation Time, CT: Consolidated Time,

IRT: Incontinency Removal Time

6.0 PILOT IMPLEMENTATION

For the pilot run the error insertion method was used. Specifically, 8 artifacts with a known inserted defects were given to both selected teams. Artifacts 1 to 4 were inspected by both teams using the formal model. Artifacts 5 to 8 were inspected by both teams utilizing ISIM. The number of inserted defects was equal for both groups. However, the artifacts were organized in a different sequence so as not to give the inspectors any awareness about the number of defects inserted. For instance, the first artifact that was inspected using the formal model had 22 defects and this was the number of defects detected in the second artifact by another team that was inspected using the proposed model. Similarly, the second artifact of the first group had the same number of defects with the last artifact of the second group.

Table 1. Pilot data for the process validation

Inserted defects		Formal Model				Proposed Model			
		Art-1	Art- 2	Art-3	Art-4	Art-5	Art-6	Art-7	Art-8
		22	31	35	19	19	22	35	31
Team A	Spotted defects	14	22	27	15	16	19	25	26
	False positive	0	0	3	1	1	0	0	1
	Defects* Remained	8	9	11	5	4	3	10	6
Team B	Spotted defects	16	21	29	14	17	20	23	28
	False positive	2	1	0	0	0	0	0	1
	Defects* Remained	10	11	6	5	2	2	12	4

Defects Remained = No. of inserted defects – No. of detected defects + No. of false positives

Table 1 contains detailed information of the inserted defects and results of the pilot implementation. Team A using the formal model have detected 73 defects out of a total of 107 inserted defects and team B have detected 70 defects out of a total of 107 inserted defects. The level of efficiency for both team stands at approximately 70 % and the difference between the ability of the two teams in detecting defects is less than 3 %, which are acceptable. Using the proposed model, team A detected 86 defects out of a total of 107, which brings their capability to detect defects to 80 %. Team B detected 88 defects and stands equal to team A as far as the capability for defect detection is concerned. The difference in the two teams' performance using the proposed model and the designed tool is slightly over 2 %, which is also acceptable.

7.0 ANALYTICAL TABLES

Table 2 to 5 present the analysis that include significant information about the case study. Artifact information by their codes, document page number, and the measured function point for each of them is included in the tables. The first section is the case study of the first eight artifacts that is related to the analysis phase of the first project. Likewise, the next table is the case study of eight artifacts of the design phase of the first project that have been inspected in the first study. The third and fourth tables respectively include information on artifacts of the analysis and design phases of the second project.

8.0 ANALYTICAL DISCUSSION

The inspection time of each artifact using both models and the number of detected defects are mentioned in the analytical tables. The items which were wrongly flagged as defects are marked as false positives and took out from the table. The number of correctly detected defects has been included in the table after deduction of the false positives.

7

Defects that have been marked by both teams and defects which have been detected using the proposed model occupied the next row in the table. This was it is clarified which model had more success in detecting defects. The last three sections of the table are allocated to the values of the three types of metrics, namely, efficiency, effectiveness and correctness.

Table 2. First case study at analysis phase

	Project 1 Analysis Phase	Artifact							
		E1_A1	E1_A2	E1_A3	E1_A	E1_A5	E1_A6	E1_A7	E1_A8
1	*Function Point*	117	66	70	62	41	40	59	78
2	*Number of Pages*	27	23	23	22	17	18	19	24
3	***Inspection by ISIM***	**Team A**				**Team B**			
3.1	*Inspection time using ISIM*	221	218	214	212	206	201	210	218
3.3	*Correct defects spotted by ISIM*	34	26	28	26	17	23	26	28
4	***Inspection by Formal***	**Team B**				**Team A**			
4.1	*Inspection time by Formal*	229	218	220	217	216	216	217	223
4.3	*Correct defects spotted by Formal*	26	15	18	15	12	21	13	21
5	***Statistic***	**Team A and Team B**				**Team A and Team B**			
5.1	*Total spotted correct defects*	36	26	29	27	17	23	27	28
5.2	*Common spotted correct defects*	24	15	17	14	12	21	12	21
5.3	*Correct defects only spotted by ISIM*	10	11	11	12	5	2	14	7
5.4	*Correct defects only spotted by Formal*	2	0	1	1	0	0	1	0
6	***Correctness Metric***	**Team A and Team B**				**Team A and Team B**			
6.1	*% Correct defects only by ISIM*	27	42	37	44	29	8	51	25
6.2	*% Correct defects only by Formal*	5	0	3	3	0	0	3	0
7	***Efficiency Metric***	Team A and Team B				Team A and Team B			
7.1	*Time per correct defect using ISIM*	6.13	8.38	7.37	7.85	12.11	8.73	7.77	7.78
7.2	*Time per correct defect using Formal*	6.36	8.38	7.58	8.03	12.7	9.39	8.03	7.96
8	***Effectiveness Metrics***	**Team A and Team B**				**Team A and Team B**			
8.1	*Correct defect per FP using ISIM*	3.4	2.5	2.5	2.3	2.4	1.7	2.2	2.7
8.2	*Correct defect per FP using Formal*	4.5	4.4	3.8	4.1	3.4	1.9	4.5	3.7
8.3	*Correct defect per Page using ISIM*	1.2	1.1	1.2	1.1	1	1.2	1.3	1.1
8.4	*Correct defect per Page using Formal*	1.03	1.53	1.27	1.46	1.41	0.85	1.46	1.14

Table 3. First case study at design phase

	Project 1 Design Phase	Artifact							
		E1_D9	E1_D10	E1_D11	E1_D12	E1_D13	E1_D14	E1_D15	E1_D16
1	*Function Point*	114	109	108	26	68	81	34	87
2	*Number of Pages*	29	25	25	13	23	19	16	23
3	***Inspection by ISIM***	**Team A**				**Team B**			
3.1	*Inspection time using ISIM*	221	220	220	199	214	219	205	224
3.3	*Correct defects spotted by ISIM*	32	36	35	11	25	31	21	34
4	***Inspection by Formal***	**Team B**				**Team A**			
4.1	*Inspection time by Formal*	229	228	227	210	220	229	218	220
4.3	*Correct defects spotted by Formal*	33	24	27	8	16	21	13	22
5	***Statistic***	**Team A and Team B**				**Team A and Team B**			
5.1	*Total spotted correct defects*	34	37	35	11	25	32	22	34
5.2	*Common spotted correct defects*	31	23	27	8	14	20	12	22
5.3	*Correct defects only spotted by ISIM*	1	13	8	3	11	11	9	12
5.4	*Correct defects only spotted by Formal*	2	1	0	0	2	1	1	0
6	***Correctness Metric***	**Team A and Team B**				**Team A and Team B**			
6.1	*% Correct defects only by ISIM*	2	35	22	27	44	34	40	35
6.2	*% Correct defects only by Formal*	5	2	0	0	8	3	4	0
7	***Efficiency Metric***	**Team A and Team B**				**Team A and Team B**			
7.1	*Time per correct defect using ISIM*	6.5	5.94	6.28	18.09	8.56	6.84	9.31	6.58
7.2	*Time per correct defect using Formal*	6.73	6.16	6.48	19.09	8.8	7.15	9.9	6.47

8

8	Effectiveness Metrics	Team A and Team B				Team A and Team B			
8.1	Correct defect per FP using ISIM	3.5	3	3	2.3	2.7	2.6	1.6	2.5
8.2	Correct defect per FP using Formal	3.4	4.5	4	3.2	4.2	3.8	2.6	3.9
8.3	Correct defect per Page using ISIM	1.1	1.4	1.4	0.8	1	1.6	1.3	1.4
8.4	Correct defect per Page using Formal	0.87	1.04	0.92	1.62	1.43	0.9	1.23	1.04

Table 4. Second case study at analysis phase

	Project 2 Analysis Phase	Artifact							
		E2_A17	E2_A18	E2_A19	E2_A20	E2_A21	E2_A22	E2_A23	E2_A24
1	Function Point	96	93	30	26	133	75	75	170
2	Number of Pages	35	32	12	10	30	25	23	35
3	Inspection by ISIM	Team A				Team B			
3.1	Inspection time using ISIM	219	219	206	193	223	213	218	227
3.3	Correct defects spotted by ISIM	35	33	19	13	36	27	21	39
4	Inspection by Formal	Team B				Team A			
4.1	Inspection time by Formal	225	226	212	209	230	223	222	234
4.3	Correct defects spotted by Formal	24	22	16	7	30	18	14	31
5	Statistic	Team A and Team B				Team A and Team B			
5.1	Total spotted correct defects	36	35	19	13	39	27	24	41
5.2	Common spotted correct defects	23	20	16	7	27	18	11	29
5.3	Correct defects only spotted by ISIM	12	13	3	6	9	9	10	10
5.4	Correct defects only spotted by Formal	1	2	0	0	3	0	3	2
6	Correctness Metric	Team A and Team B				Team A and Team B			
6.1	% Correct defects only by ISIM	33	37	15	46	23	33	41	24
6.2	% Correct defects only by Formal	2	5	0	0	7	0	12	4
7	Efficiency Metric	Team A and Team B				Team A and Team B			
7.1	Time per correct defect using ISIM	6.08	6.25	10.84	14.84	5.71	7.88	9.08	5.53
7.2	Time per correct defect using Formal	6.25	6.45	11.15	16.07	5.89	8.25	9.25	5.7
8	Effectiveness Metrics	Team A and Team B				Team A and Team B			
8.1	Correct defect per FP using ISIM	2.8	3.5	1	1	7.8	3.2	2.8	6
8.2	Correct defect per FP using Formal	4	4.2	1.8	3.7	4.4	4.1	5.3	5.4
8.3	Correct defect per Page using ISIM	0.9	0.8	2.3	2.6	0.5	0.9	1.1	0.8
8.4	Correct defect per Page using Formal	1.45	1.45	0.75	1.42	1	1.38	1.64	1.12

Table 5. Second case study at design phase

	Project 2 Design Phase	Artifact							
		E2_D25	E2_D26	E2_D27	E2_D28	E2_D29	E2_D30	E2_D31	E2_D32
1	Function Point	44	80	49	74	70	58	67	120
2	Number of Pages	18	25	19	23	23	21	16	31
3	Inspection by ISIM	Team A				Team B			
3.1	Inspection time using ISIM	200	218	207	223	214	208	213	215
3.3	Correct defects spotted by ISIM	22	28	26	**42**	**37**	24	28	37
4	Inspection by Formal	Team B				Team A			
4.1	Inspection time by Formal	216	224	216	221	224	214	214	224
4.3	Correct defects spotted by **Formal**	24	22	14	**27**	**24**	14	14	27
5	Statistic	Team A and Team B				Team A and Team B			
5.1	Total spotted correct defects	24	28	27	**47**	**40**	25	28	38
5.2	Common spotted correct defects	22	22	13	**22**	**21**	13	14	26
5.3	Correct defects only spotted by ISIM	0	6	13	**20**	**16**	11	14	11
5.4	Correct defects only spotted by Formal	2	0	1	**5**	**3**	1	0	1
6	Correctness Metric	Team A and Team B				Team A and Team B			
6.1	% Correct defects only by ISIM	0	21	48	38	40	44	50	28

6.2	% Correct defects only by Formal	8	0	3	10	7	4	0	2
7	**Efficiency Metric**	**Team A and Team B**				**Team A and Team B**			
7.1	Time per correct defect using ISIM	8.33	7.78	7.66	4.74	5.35	8.32	7.6	5.65
7.2	Time per correct defect using Formal	9	8	8	4.7	5.6	8.56	7.64	5.89
8	**Effectiveness Metrics**	**Team A and Team B**				**Team A and Team B**			
8.1	Correct defect per FP using ISIM	2	2.8	1.8	1.8	1.8	2.4	2.3	3.2
8.2	Correct defect per FP using Formal	1.8	3.6	3.5	2.7	2.9	4.1	4.7	4.4
8.3	Correct defect per Page using ISIM	1.2	1.1	1.3	1.7	1.6	1.1	1.7	1.1
8.4	Correct defect per Page using Formal	0.75	1.13	1.35	0.85	0.95	1.5	1.14	1.14

The most important criteria are the number of correctly detected defects to the function point as a criterion for complexity and the page number as a standard for size. The results of calculations carried out on the metrics indicate that on average for each 2.7 function points one defect was detected using the proposed model, and for each 3.8 function points one defect was detected using the formal model. This value is 40 % higher in the design phase and shows that utilization of the proposed model in the design phase yields a 10% increase in effectiveness compared to the analysis phase. The efficiency metric related to the number of pages also indicates that at least 4 defects were detected on each 3 pages using the proposed model. Use of the formal model shows detection of 2 defects in three pages which translates into more than 40 percent less efficiency compared to the proposed model.

9.0 THE EFFICIENCY AND PRODUCTIVITY OF BOTH MODELS

Fig. 3 and Fig. 4 depicts the efficiency of the Formal and proposed model (ISIM). The time consumed for detecting defects of each artifact in each phase of the project along with the number of defects correctly detected by each inspector. Details of the artifact inspection times by both teams using both the formal and the proposed model are also included separately.
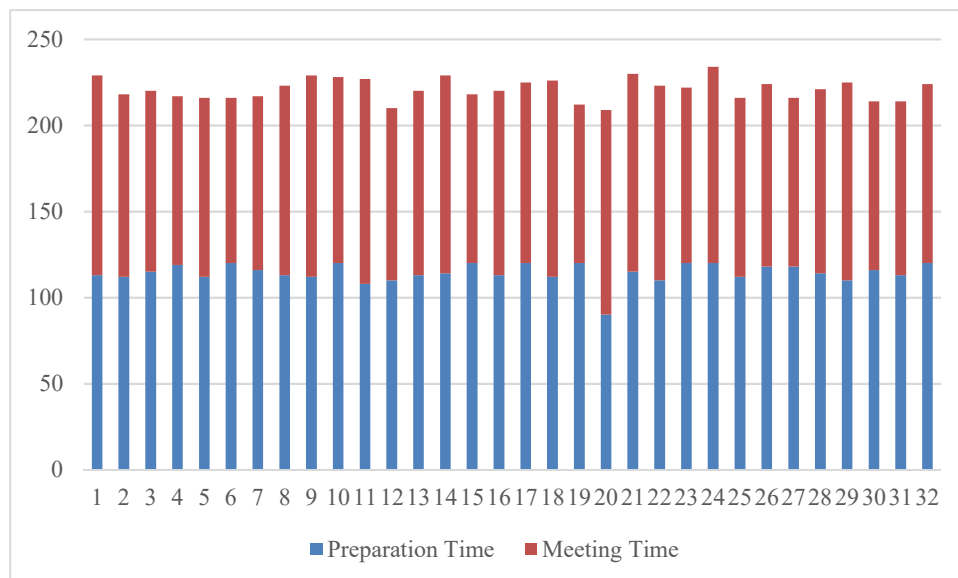

Fig. 3. Inspection time breakdown of Formal model

Productivity is established by the number of correctly detected defects and efficiency is obtained by dividing the time consumed with the correctly detected defects [25]. For instance first inspector of team B correctly detected 18 defects in 109 minutes which means s/he had an efficiency of detecting a defect every six minutes. For the proposed model,

10

the lowest level of efficiency was manifested by the first inspector of team A during inspection of artifact number 12 where s/he nearly spent 15 minutes for each defect. The highest level of efficiency was by the third inspector of team A for detecting a defect in 3 minutes. For inspections using the formal model, the third inspector of team B was the most efficient for detecting a defect every 4 minutes and 12 seconds while inspecting artifact 9. The lowest efficiency was during inspection of artifact number 7 by the third inspector of team A, who spent over 23 minutes for finding each defect.
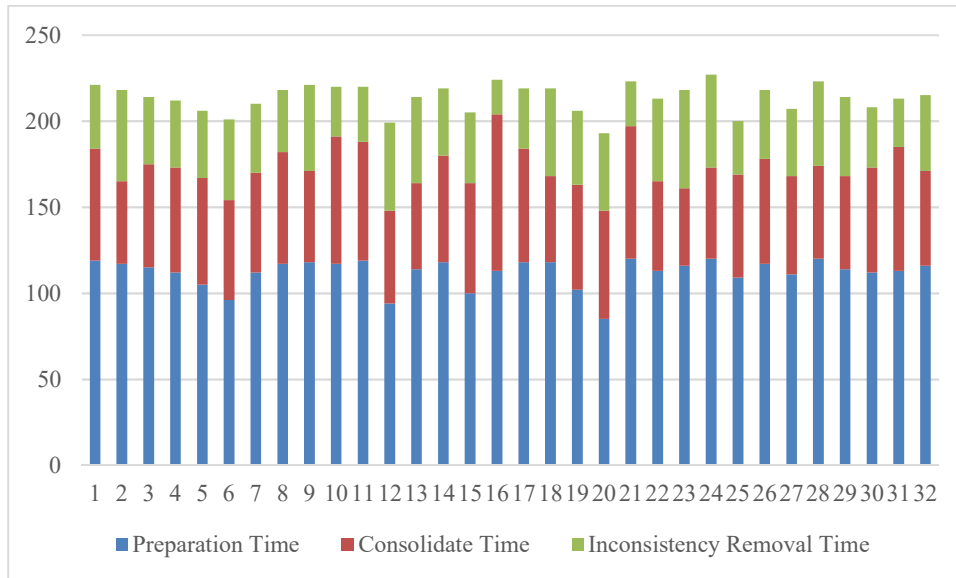
Fig. 4. Inspection time breakdown of ISIM

10.0 THE EFFECTIVENESS OF INSPECTION EFFORTS

The number of defects detected by each two team members using Formal and proposed model are shown in Table 6.Fig. 5 shows the effectiveness of the Formal and proposed model for each team members. There is a direct and positive relationship between preparation time and the number of detected defects [26]. He has also demonstrated that the effectiveness of preparation time and individual work of the inspectors is much higher than that of the number of artifacts and also much higher than that of the length of the session or the meeting. Based on this research the proposed model and the designed tool pave the way for preparation and individual work of the inspectors. The statistical analysis on the gathered data in this study have shown same result. It means, the number of detected defects indicate a positive linear relationship with the length of the inspection time (see Fig. 6).

Table 6.   Team effectiveness using two model

Team member	Number of defects detected using ISIM	Number of defects detected using Formal
I1A	69	53
I2A	52	47
I3A	79	46
I1B	71	49

11

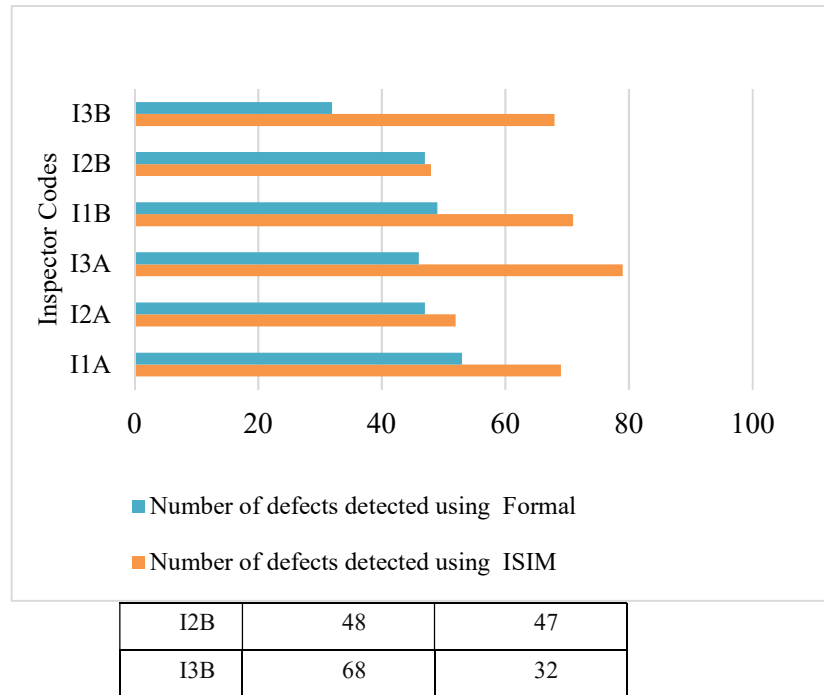|		
I2B	48	47
I3B	68	32

Fig. 5. Effectiveness of two models for each inspectors

Fig. 7 and Fig. 8 indicate that in the proposed model the higher the number of inspections, the faster the defects are detected. The diagrams show that in each inspection, the defects in the last artifacts are detected sooner than those of the first artifacts. There are no meaningful differences in this regard using the formal model, although the above-mentioned diagrams indicate that the detection rate of the defects relatively improved. However, no improvement can be seen in the later inspections from phase to phase, and specifically the speed of detection actually decreased in the last inspection (artifact 30 and artifact 31) and efficiency of the inspectors deteriorated from detecting each defect in 7 minutes to 15 minutes per defect in formal model and 6 minutes to 9 minutes per defect using ISIM.
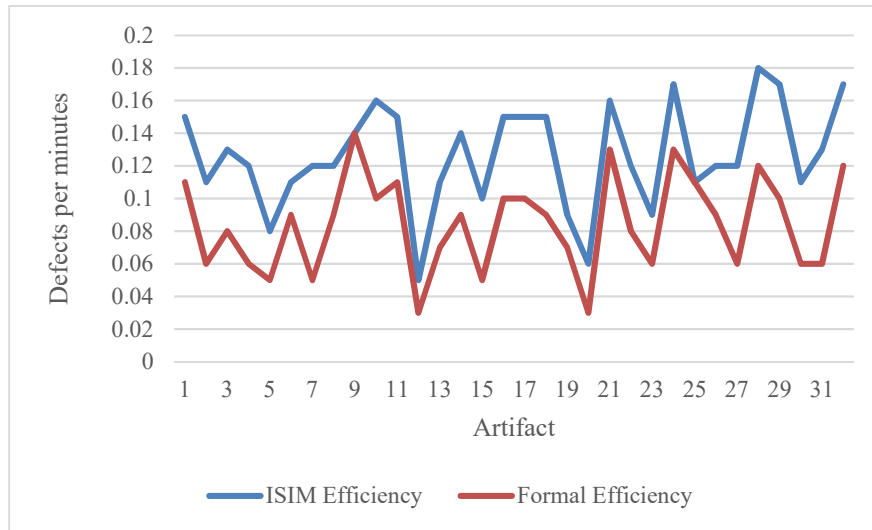
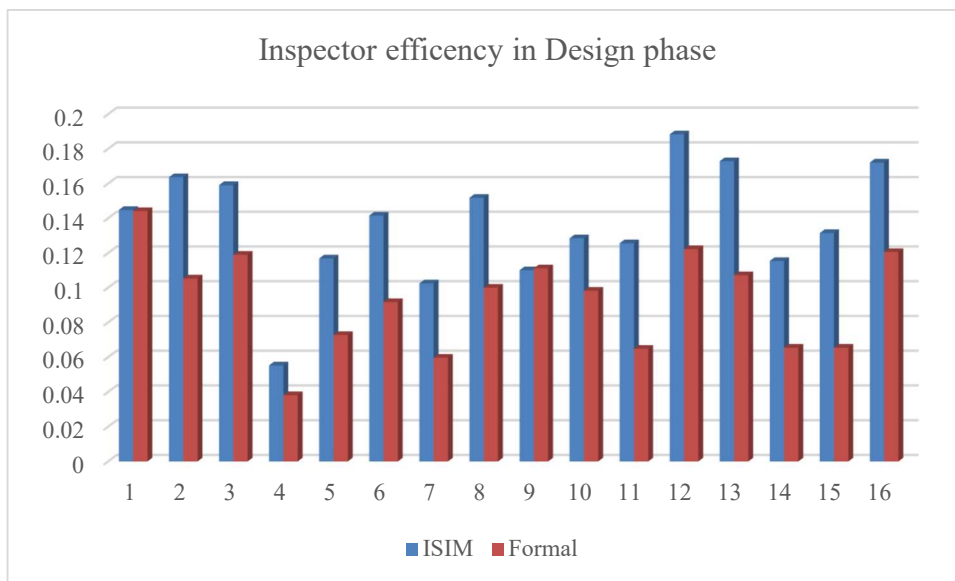Fig. 6. Comparing all defects detected in two projects using both models

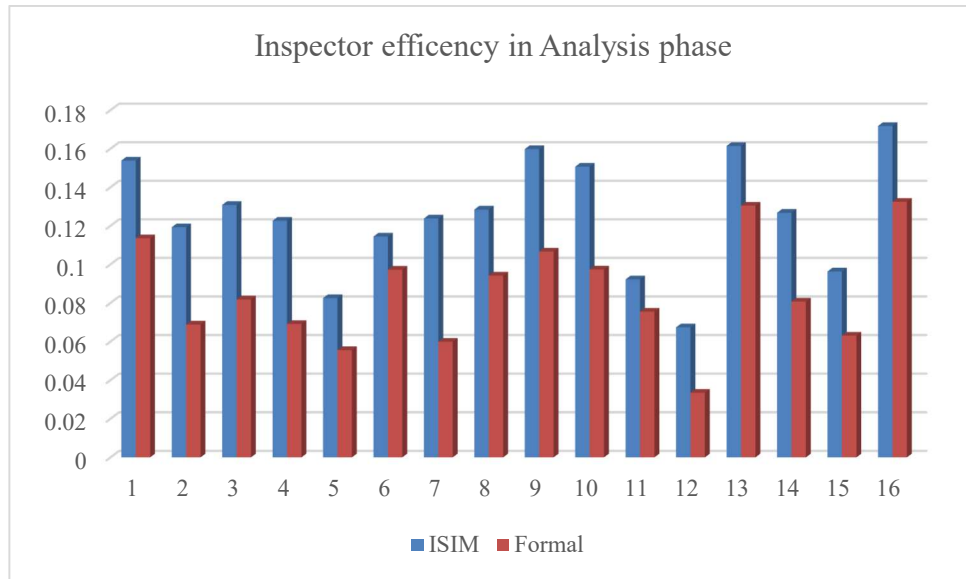Fig. 7. Comparing all defects detected in design phase using both modes

Fig. 8. Comparing all defects detected in analysis phase for two projects

For the proposed model, the average number of detected defects in the analysis phase for the first project was 26 and for the second project was 28. The average detected defects in the analysis phase of both projects for each artifact were 27 and for each unit of complexity was 3. Total number of correctly detected defects in this phase for all artifacts using the proposed model was 431.

For the formal model the average number of detected defects in the analysis phase for the first project was 18 and the number for the second project was 20. This shows a 40% decrease in comparison with the proposed model. The average detected defects in the analysis phase of the formal model was 19 and average number of defects by time was three defects per minute and for each unit of complexity for the analysis phase in both projects it was 0.25. Total number of correctly detected defects in this phase for all artifacts using the formal model was 303.

Generally, for both analysis and design phases, the total number of detected defects through the proposed model was 900 and the formal model the number is 633. The average time for detecting any defect was 4 minutes for the proposed model and 5 minute and 50 seconds for the formal model. The average defects detected for each unit of complexity was 0.37 in the proposed model and 0.26 for the formal model.

Table 7 and Table 8 conclude the statistical test results. At $\alpha = 0.05$, the paired t-test shows that there is a difference in using ISIM to improve the inspection process – in terms of No. of defects detected ($p < 0.05$ at df = 31). The inspection time of Method 1 has normal distribution but NOT normal distribution for Method 2. Hence, Wilconson Sign Rank test is used to test the improvement of inspection time. At $\alpha = 0.05$, the Wilcoxon Signed Rank test shows there is a difference in using ISIM to improve the inspection process – in terms of inspection time ($p < 0.05$)

Table 7.   Paired samples statistics

	Defect No.	Mean	N	Std. Deviation	Std. Error Mean
Pair 1	Total No. of defects detected using Method 1 (Formal Inspection)	19.78	32	6.622	1.171
	Total No. of defects detected using Method 2 (ISIM)	28.13	32	7.499	1.326

14

Table 8. Tests of normality

Time	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
Total inspection time using Method 1 (Formal Inspection)	.090	32	.200*	.981	32	.826
Total inspection time using Method 2 (ISIM)	.169	32	.020	.946	32	.113

11.0 RESULTS AND CONCLUSION

The application of an inspection model in the preliminary development stages of software has been reviewed in the present research. The two case studies on artifacts of two bit software projects by two teams comprised of experienced inspectors proved the efficiency and effectiveness of the proposed model. Validity and credibility of the model have been assessed proved by taking executive measures to avoid bias. Findings show a 33% to 40% increase in the productivity of the inspectors of both teams in inspecting artifacts of the two projects in both analysis and design phases.

If fixing each defect takes 20 person/minutes [27], considering the 40 percent improvement in defect detection through the use of the proposed model then, there would be at least a 40 percent decrease in the defect fixing costs. The researchers emphasized that the person/hour basis is appropriates for calculation of defect fixing costs for measuring the difference in costs that would results from utilizing a better model. Findings of their study indicate that a defect that can be fixed using 20 person/hours in the early stages, requires 4 times the resources that is 80 person/hours in the later stages. Therefore, if the proposed model is not applied the costs of fixing unseen defects in the later stages can increase by 80% which is a meaningful difference. The possibility of registering information on the cause of the defects, their types and their impact level has made learning possible as a major feature of the software. This way the detection of candidate defects through repetition of inspection becomes more probable.

REFERENCES

[1]   M. E. Fagan, "Design and code inspections to reduce errors in program development", *IBM Systems Journal*, Vol. 15, No. 3, pp. 182–211, 1976.

[2]   R. Klimek, "A system for deduction-based formal verification of workflow-oriented software models", *International Journal of Applied Mathematics and Computer Science*, Vol. 24, No. 4, Jan. 2014.

[3]   D. Mishra and A. Mishra, "Simplified software inspection process in compliance with international standards", *Computer Standards & Interfaces*, Vol. 31, No. 4, pp. 763–771, Jun. 2009.

[4]   S. Misra, L. Fernández, and R. Colomo-Palacios, "A simplified model for software inspection", *Journal of Software: Evolution and Process*, Vol. 26, No. 12, pp. 1297–1315, Oct. 2014.

[5]   H. Srikanth and S. Banerjee, "Improving test efficiency through system test prioritization", *Journal of Systems and Software*, Vol. 85, No. 5, pp. 1176–1187, May 2012.

[6]   H. Zhang and M. Ali Babar, "Systematic reviews in software engineering: An empirical investigation", *Information and Software Technology*, Vol. 55, No. 7, pp. 1341–1354, Jul. 2013.

[7]   N. M. C. Valentim and T. Conte, "Improving a Usability Inspection Technique Based on Quantitative and Qualitative Analysis", *Brazilian Symposium on Software Engineering*, Sep. 2014.

[8]   D. Winkler, B. Thurnher, and S. Biffl, "Early Software Product Improvement with Sequential Inspection Sessions: An Empirical Investigation of Inspector Capability and Learning Effects", *33rd EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO 2007)*, Aug. 2007.

[9]   J. W. Wilkerson, J. F. Nunamaker, and R. Mercer, "Comparing the Defect Reduction Benefits of Code Inspection and Test-Driven Development", *IEEE Transactions on Software Engineering*, Vol. 38, No. 3, pp. 547–560, May 2012.

[10]   Anon. "Improving manual analysis of automated code inspection results: Need and effectiveness", *IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, Nov. 2013.

[11]   V. Balachandran, "Reducing human effort and improving quality in peer code reviews using automatic static analysis and reviewer recommendation", *35th International Conference on Software Engineering (ICSE)*, May 2013.

[12]   M. E. Fagan, "Advances in Software Inspections," *Software Pioneers*, pp. 609–630, 2002.

[13]   R. Malhotra, N. Kapoor, R. Jain, and S. Biyani, "Severity Assessment of Software Defect Reports using Text Classification", *IJCA*, Vol. 83, No. 11, pp. 13–16, Dec. 2013.

[14]   B. Marculescu, R. Feldt, and R. Torkar, "Practitioner-Oriented Visualization in an Interactive Search-Based Software Test Creation Tool", *20th Asia-Pacific Software Engineering Conference (APSEC)*, Dec. 2013.

[15]   H. Rajput and L. K. Singh, "Improvement of Software Quality Attributes in Object Oriented Analysis and Design Phase Using Goal-Question-Metric Paradigm", Journal of Software Engineering and Applications, Vol. 4, No. 6, pp. 345–349, 2011.

[16]   J. W. Creswell. *Educational research planning, conducting, and evaluating quantitative and qualitative research.* ed. 3. Upper Saddle River, New Jersey: Pearson Prentice Hall, 2008.

[17]   J. Albiero, "Formal software inspection as an option for software quality improvement",   SAE Technical Paper Series, 22 Nov. 2005.

[18]   P. Fernandes, T. Conte, and B. Bonif'cio, "WE-QT: A Web Usability Inspection Technique to Support Novice Inspectors", *26th Brazilian Symposium on Software Engineering*, Sep. 2012.

[19]   S. Biffl, "Using inspection data for defect estimation", *IEEE Software*, Vol. 17, No. 6, pp. 36–43, 2000.

[20]   H. He and D. Liu, "An Automatic Software Requirement Analysis Approach based on Intelligent Planning Technology", *JSW*, Vol. 9, No. 11, Nov. 2014.

16

[21]   S. Basak and M. Shazzad Hosain, "Software Testing Process Model from Requirement Analysis to Maintenance", *IJCA*, Vol. 107, No. 11, pp. 14–22, Dec. 2014.

[22]   C. Jones, "Software defect-removal efficiency", *Computer*, Vol. 29, No. 4, pp. 94–95, Apr. 1996.

[23]   R. Tantri and N. N. Murulidhar, "An Efficient Estimator of Reliability for Exponential Class Software Reliability Models", *Lecture Notes on Software Engineering*, Vol. 2, No. 3, pp. 201–204, 2014.

[24]   T. R. Gopalakrishnan Nair, V. Suma, and P. Kumar Tiwari, "Significance of depth of inspection and inspection performance metrics for consistent defect management in software industry", *IET Software*, Vol. 6, No. 6, p. 524, 2012.

[25]   Y. Rafique and V. B. Misic, "The Effects of Test-Driven Development on External Quality and Productivity: A Meta-Analysis", *IEEE Transactions on Software Engineering*, Vol. 39, No. 6, pp. 835–856, Jun. 2013.

[26]   M. Ciolkowski, O. Laitenberger, D. Rombach, F. Shull, and D. Perry, "Software inspections, reviews and walkthroughs", *Proceedings of the 24th International Conference on Software Engineering*. ICSE 2002, 2002.

[27]   T. Berling and T. Thelin, "An industrial case study of the verification and validation activities", *Proceedings. 5th International Workshop on Enterprise Networking and Computing in Healthcare Industry* (IEEE Cat. No.03EX717), 2003.