Improving Reuse of Test Strategy based on ISO/IEC Standards[☆]

민 경 식¹   이 정 원²   이 병 정^{1*}

Kyeongsic Min   Jung-Won Lee   Byungjeong Lee

ABSTRACT

A test plan is a high level document detailing objectives, processes, schedules and so on for verifying a developed software. And a test strategy, a component of a test plan, is about how to test software products to guarantee its quality and find bugs in the software in advance. Therefore, establishing effective and suitable test strategies is important for elaborating test processes. However, these tasks are difficult for project managers who write a test plan if they were not trained well in software test processes. And mis-designed test strategies will also mislead entire testing behaviors that testers would do. As a result, there would be a low quality software product in the end. To solve this problem, we propose a new test strategy reuse technique in this paper. By utilizing test plans of already completed software development projects, we lead test planer to reuse suitable and effective test strategies which were used in previous projects. To do so, we evaluate existing test strategies by utilizing ISO/IEC 25010 quality model for evaluating the suitability of test strategies and also use effectiveness metrics for test strategies. And from these evaluations, we predict completeness of new test plan that is written by reused test strategies. It can help the project manager to write an appropriate test plan for the quality characteristics which are selected as objectives for testing and software product. We show the possibility of our approach by implementing a prototype into the existing framework in a case study.

☞ keyword : Test Strategy Reuse, Test Plan Reuse, ISO/IEC 25010, ISO/IEC 25023, ISO/IEC 29119

1. Introduction

As the complexity of software has been recently increased continually, developing high-quality software is getting more difficult[13]. In this aspect, A test process is important as a method to verifying the quality of software before deployment. In general, project managers usually write a test plan for testing developed software to make it to a better software product. However, building a sophisticated test plan is not an easy task for a beginner, and an inadequate test plan can make it difficult to test for the desired quality, or lead to unnecessary testing.

Test strategies, which are parts of the test plan, allow for consistent testing so that directional testing is possible. It is necessary to maintain an optimized organizational-level test strategy that reflects the experience gained through testing activities[1]. Based on the optimized test strategy, test plan of a new project would be established and the actual tests are performed, so testing completeness is continuously improved and testing becomes convincing.

The motivations of this study are as follows.
- Even though there are enough optimized test strategy, selecting suitable test strategies for software requirements is also a time-consuming task.
- ISO/IEC 25010[2] proposes quality models for software and describes that use of the quality models is beneficial in activities during product development; identifying software requirements, identifying software testing objectives and so on.

Therefore, in this paper, we propose a test strategy reuse technique through suitability and effectiveness prediction with test phases. And we implemented our approach in the web-based R&D software test support framework[10]. The project manager can reuse existing test strategies in the

repository and evaluate his new test strategies for maintaining optimized test strategies for future R&D software projects.

This study provides the following contributions.

● We utilize a suitability and effectiveness evaluation of a test strategy approach to maintain optimized test strategies, respectively.

● We define a new test planning processes based on ISO/IEC 29119-2. It is used for reducing efforts of designing new test strategies and improving the quality of the test plan by reusing test strategy.

● We support the project manager to reuse existing test strategies more easily. The reusing process is divided into test phases and it uses keyword-based search and recommendation score of the test strategy.

● We propose how to predict completeness of the test plan that is written with reused test strategies according to sub-characteristics in ISO/IEC 25010 quality model as software requirements and also objectives of the test processes.

● We also prototyped the above contributions as a part of our framework for showing those effectivenesses and possibilities.

The outline for the rest of the paper is as follows. We first describe the background knowledge in Section 2. In Section 3, we introduce related works. After that, We explain our methodology in Section 4. And in Section 5, we represent a case study about our methodology with the implementation of our methodology in the R&D software test support framework. We then discuss our research in Section 6. At last, in Section 8, we conclude our paper.

2. Background Knowledge

2.1 ISO/IEC 25010

ISO/IEC 25010[2] suggests a product quality model consist of 8 characteristics which are subdivided into 31 sub-characteristics and a *quality in use* model which is composed of 5 characteristics. The scope of application of the quality models includes supporting evaluation and specification of software from perspectives of quality

assurance, development and so on. Parts of benefits of using the quality models are as follows.

● A benefit during an activity that identifies software requirements.

● A benefit during an activity that identifies software testing objectives.

● A benefit during an activity that identifies quality control criteria for quality assurance.

● Configuring measurement methods of quality characteristics in supporting these activities.

Therefore, in this paper, we utilize sub-characteristics of the quality models based on these benefits to set objectives of software testing and its test plan.

2.2 ISO/IEC 25023

ISO/IEC 25023[3] defines quality measures in terms of characteristics and sub-characteristics defined in ISO/IEC 25010. Ultimately, It is for quantitatively evaluating system and software product quality. The standard includes the following contents.

● A set of quality measure metrics of characteristics and sub-characteristics.

● A description that is how to apply software product quality measures.

2.2 ISO/IEC 29119

ISO/IEC 29119 is set of software testing standards such as test processes, test documentation and test techniques[4, 14]. In our study, we use ISO/IEC 29119-2[5], part 2 of ISO/IEC 29119, which defines test process which uses risk-based approach. It explains a multi-layered model showing all the test processes in Figure 1.

In this paper, among all test processes, we focus on the *test planning process* of *test management processes*. In addition, the standard also describes *test planning processes* that are used for planning a test plan. In a project, a project manager develop project test plan, test plan for specific test phase or test plan for specific test type. For developing test plan, the processes as shown in Figure 2 should be done. In our approach, we deal with the *design test strategy* process and the *document test plan* process by utilizing identified

risks and risk treatment approaches.
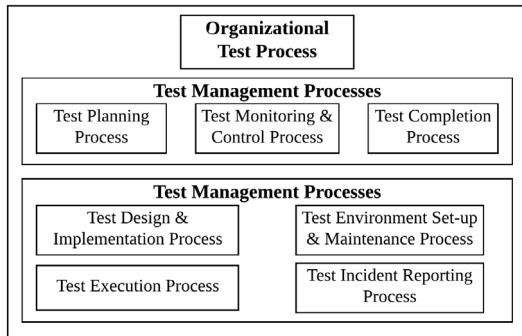
(Figure 1) A Multi-layered Model and Activities of ISO/IEC 29119-2
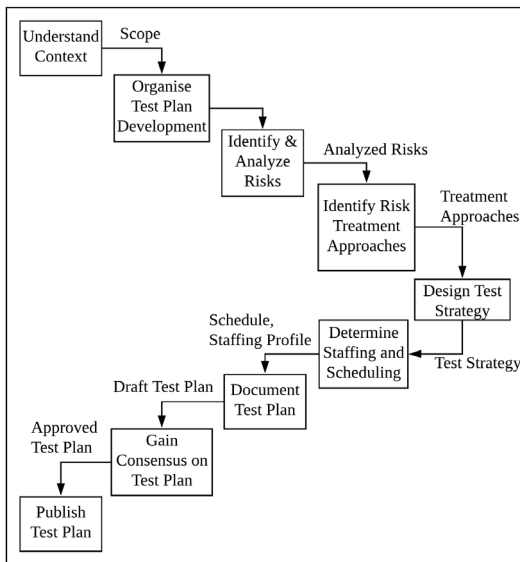
(Figure 2) Test Planning Processes

3. Related Works

There are many existing studies using a variety of methodologies for reducing test effort based on reuse approach[6].

In previous studies, K. Min et al.[7] introduced a test strategy reuse technique from existing software products. And it predicts completeness of test plan by test strategy

compatibility. This study is also based on ISO/IEC 25010 quality model. J. Al Dallal et al.[8-Reusing class-based] proposed generating specification-based reusable class-level test cases during the object-oriented framework development stage. The test cases are reused for the application development stage.

As far as we know, [7] is the first study that researches a test strategy reuse approach for reducing test effort. The study also represented the browsing method of test strategy recommendations to support test planning process. However, if there are too many existing projects and test strategies, simply browsing high suitable and effective test strategies is not effective to choose appropriate test strategies to the risks of the project. And [8] focused on the reuse test case for test execution stage to reduce the effort of developing test cases. Therefore, there is a difference between the reused test component of the study and our research.

And there is a previous study about measures of test strategy. Y. Chen et al.[9] proposed an effective metrics for test strategy evolution. For designing and evolving optimized cost-effective test strategies, the metrics are helpful. However, in the study, there are limitations that it only considered the effectiveness of test strategies. For test strategy reuse, identified risks and testing objectives are core elements to develop the appropriate test plan. Therefore, considering suitability of test strategies to the project is also as important as considering its effectiveness.

4. Test Strategy Reuse Technique

4.1 Overview

In this paper, we extend the test planning processes in Figure 2 and add the 'Deploy software product' process so that our proposed methods can be represented. An overview of the proposed method is shown in Figure 3. Some processes with little relation to this reuse study are omitted, and the element and processes to which the proposed technique is applied are indicated by bold lines.

First of all, the final goal of this technique is the project manager effectively develops a test plan that meets the objectives of testing. To achieve this goal, the project manager first identifies software testing requirements through
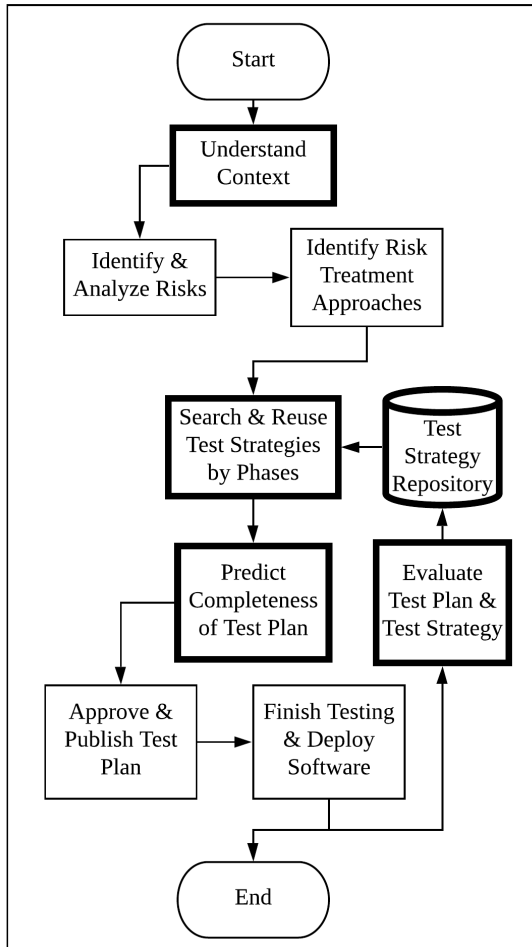
(Figure 3) An Overview of our Extended Test Planning Processes Including Evaluation Process

understanding context and the ISO/IEC 25010 quality model. Then the project manager identifies and analyzes risks and treatment approaches. Then, the test strategies are keyword-retrieved from the test strategy repository according to the test phases, and the appropriate test strategies are reused. The next step is to predict the completeness of the test plan based on reused test strategies. The tester will conduct the test according to the written and approved test plan, and will then deploy the software product. After software deployment, the final step is to evaluate the used test plan and test strategies in accordance with the metrics and these results are stored in the test strategy

repository for future projects. A detailed description of each process is given in the following section.

4.2 Understanding Context

Understanding context and software test requirements is required to support test planning. And the proposed technique uses ISO/IEC 25010 to identify test requirements [5].

In order to reuse suitable test strategies, evaluation criteria are needed to determine which test strategy is appropriate. Therefore, in this process, the project manager selects the ISO/IEC 25010 sub-characteristics and sets them as test objectives[2, 10] in the meaning of requirement of a final software product. For example, if the ongoing project is a financial project, security is very important for the project. So, the *confidentiality* and *integrity* of the *security* characteristic will be added as test objectives. In the next steps, understanding context and sub-characteristics is used to identify the risks and the corresponding treatment approaches, and to develop a test plan.
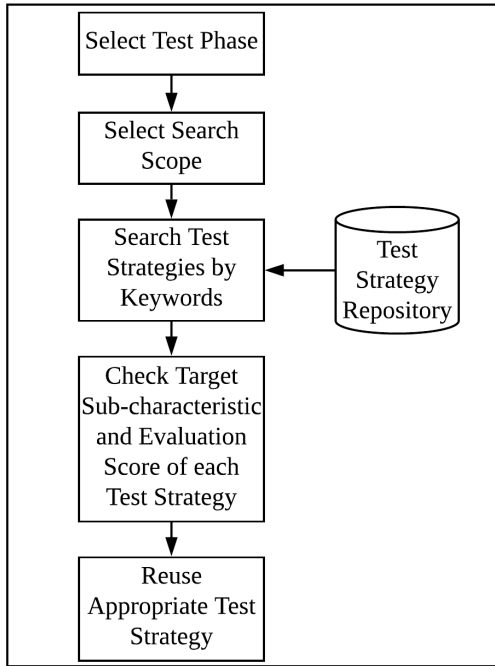
4.3 Searching & Reusing Strategies by Phases

In this process, the test manager searches the test strategies based on the keywords and reuses the test strategy suitable for the project.

Figure 4 shows the details of the process. The project manager first selects the phase (eg, unit level, integration level, and system level) of the test plan to be created. Then, he sets a search scope (e.g., analyzed risks, treatment approaches). After the test phase and search scope setting is completed, the test strategy which corresponds to the selected test phase and contains the keywords in the search scope is searched in the test strategy repository. The test manager checks the contents of the test strategy along with the recommendation score from the search results and reuses the appropriate test strategies for the project.

In this study, the method of calculating the recommendation score of test strategy is as follows.

$$RecScore(ts, tc) = Score_e \times Score_s$$
$$0 \le Score_e, Score_s \le 1$$

Select Test Phase

→ Select Search Scope

→ Search Test Strategies by Keywords ← Test Strategy Repository

→ Check Target Sub-characteristic and Evaluation Score of each Test Strategy

→ Reuse Appropriate Test Strategy

(Figure 4) Detailed Steps of Process of Search & Reuse Test Strategies by Phases

- ts is a test strategy.
- tc is a target sub-characteristic.
- $Score_e$ is an effectiveness score of a test strategy[9].
- $Score_s$ is a suitability score of a test strategy[3].

This process can be iterated according to the division of the test phase. For example, if you want to create a test plan by dividing test phases into unit levels, integration levels, and system levels in a project, the process will be repeated three times.

By reusing test strategies in this process, project managers can reduce the effort required to write new test strategies. In addition, high-quality test plans can be developed by reusing optimized test strategies and ensuring that test objectives are met through targeted sub-characteristics of the test strategy.

4.4 Predicting Completeness of Test Plan

This process predicts the completeness of the test plan established through reuse in the previous process. The criteria for predicting completeness are the test objectives set out in Section 4.2, which were selected as the sub-characteristics of ISO/IEC 25010. The method of calculating the completeness score of the test plan in this study is as follows.

$$ComScore(R,O) = \frac{\sum_{s=1}^{n(O)} \mathrm{Pre}(R,O_s)}{n(O)} \times 100\%$$

$$\mathrm{Pre}(R,O_s) = \sum_{t=1}^{n(R)} Score_s$$

- R is a set of reused test strategies.
- O is a set of objectives of a test plan, which consist of sub-characteristics.
- $\mathrm{Pre}(R,O_s)$ is prediction score which is the sum of suitability score of test strategies towards a sub-characteristic.
- $Score_s$ is a suitability score by the sub-characteristic metric.

By predicting the completeness of the test plan based on the test objectives, the project manager can guess how much of the draft test plan will meet the test objectives. A test plan that fully meets the test objectives will help you to carry out the *Test Execution Process* that meets the test objectives in the future.

4.5 Evaluating Test Plan & Test Strategy

This section describes the process of evaluating used test plans and test strategies. This evaluation process is an essential process for the processes in Section 4.3 and Section 4.4.

After the project is completed, the project manager will evaluate the test plan and test strategy so that to manage the optimized test strategy. We use the test effectiveness metric proposed in [9] and ISO/IEC 25023 metrics for evaluation. The test effectiveness metric is as follows.

$$Score_e = \frac{\mathrm{defect}_{TT}}{number\ of\ \mathrm{defects}}$$

- defect_{TT} is the number of defects found by the test team.
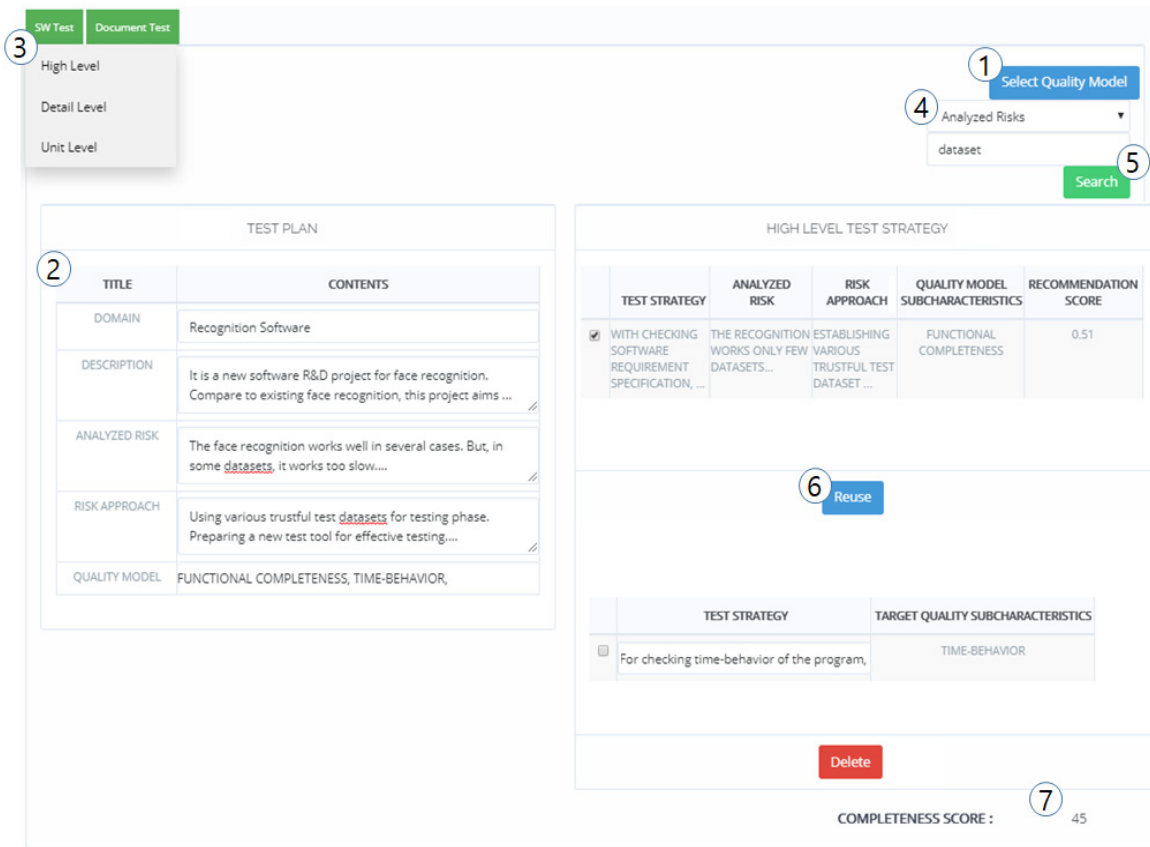
(Figure 5) A Page of the Prototype for Test Planning in Case Study

In the evaluation using the metrics of ISO / IEC 25023, only the metrics corresponding to the selected sub-characteristics as the test objectives in Section 4.2 are used and the results are stored as the suitability score of the test strategy aimed at each sub-characteristic. In the case of an evaluation using the Test Effectiveness metric, it is used as an effectiveness score of the entire test strategy used in the project being evaluated.

The process is a feedback process that reflects the completion result of the project to the used test plan and test strategies. It has a feature that the evaluation result of the final software product according to the ISO/IEC 25010 quality model can be utilized as it is.

5. Implementing the Reuse Technique into the Framework

In this section, we introduce a prototype of our proposed reuse technique which is implemented into the web-based R&D software test support framework[10, 11]. This framework is a software project test support framework for software R&D projects by using test technique, development artifacts verification, defect tracking and management technology based on test results. And because the framework is for software R&D projects, it is necessary to define software R&D test activities. We used the definition in [12] that divided into three phases: high-level test planning, detailed-level test planning, and unit-level test planning. It

will be used in test strategy reuse with phases. Note that, we used the software R&D project's test activity definition, but it can be replaced with another general definition and test phases.

Through a case study in this section, We represent the possibility of our approach. We assume that there is a recognition software R&D project like human face recognition software. In this case, Figure 5 shows a page of the prototype for test planning. The detailed processes that the project manager need to do for test planning are as follows.

① Understand Context: The project manager first understands the context by identifying the requirements of the software. And then, he selects sub-characteristics as objectives of a new test plan. Figure 6 shows selecting sub-characteristics step. In this case, the project manager simply chooses *Functional Completeness* and *Time-behavior sub-characteristics*.

② Identify Risks and Risk Treatment Approaches: The project manager identifies and writes analyzed risks and its treatment approaches. In the case, the analyzed risk is recognition function works too slow sometimes.

③ Select Test Phase: There is three test phase according to the definition of software R&D test activities.
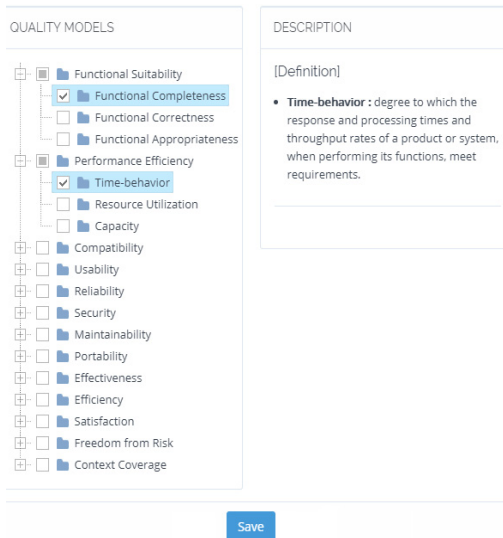
(Figure 6) A Page for Selecting Sub-characteristics

④ Select Search Scope: The project manager can select a scope(e.g. analyzed risks, treatment approaches, test strategy) for searching test strategies.

⑤ Search Test Strategies by Keywords: Keyword "dataset" is written in ⑤ of Figure 5. In this case, there had been two results with recommendation score, target sub-characteristic and the other information.

⑥ Reuse Appropriate Test Strategy: The project manager can select test strategies for reuse with considering recommendation score. In this case, we reused one of the searched test strategies.

⑦ Predict Completeness of Test Plan: Test plan completeness prediction score is shown at ⑦ of Figure 5. It is calculated based on the suitability score of the reused test strategy.

⑧ Evaluate Test Plan & Test Strategy: After deploying software product or finishing the project, the project manager evaluates test plan & test strategy as shown in Figure 7. Figure 7 only shows a page for test strategy suitability evaluation for example. In our case, two sub-characteristics that were chosen in step 1 are shown to be evaluated.

6. Discussion

The feature of the proposed technique and related works is to reduce test efforts by reusing components related to testing processes. Using these technique, it is possible to execute effective testing processes with minimized time and effort of the testers. In this chapter, we compare the test strategy reuse method of this study with the existing studies and then discuss the limitations.

6.1 Comparison with Existing Study

Table 1 shows the qualitative comparison of the features of this study and previous studies. Most of the previous studies, except [9], used reuse techniques to reduce test effort. Instead, [9] enabled the evolution of test strategies through the effectiveness evaluation of test strategies. On the other hand, [8] used the reuse technique, but the reused component is the test case, and the reuse method is selecting test strategies by the tester manually. [7] aims to reuse test strategies like this

Number ▲	Standard ⇕	Characteristics ⇕	Sub-characteristics ⇕	Metric ⇕	Params ⇕	Score ⇕
1	ISO/IEC 25010	Functional Suitability	Functional Completeness	1-A/B	A = Number of functional requirements, B = Number...	Result
2	ISO/IEC 25010	Performance Efficiency	Time-behavior	(B-A)/C	A = Time of entering a command, B = ...	Result

< Before 1 Next >

Save

(Figure 7) A Page for Evaluating Suitability of Test Strategy

study. However, [7] reuses the test strategy through the browsing technique, and there is insufficient consideration of the test phases. On the other hand, our proposed method classified the test strategy by test phases, and the keyword-based search enables the project manager to reuse the test strategy easily and accurately.

(Table 1) Qualitative Comparison of Our Study and Existing Studies

	(7)	(8)	(9)	This Study
Reuse Technique	O	O	X	O
Division by Test Phases	X	X	X	O
Methods for reuse	Browsing	Manual	X	Searching
Reusable Component	Test Strategy	Test Case	X	Test Strategy
Test Strategy Evolution	O	X	O	O

6.2 Limitation

In our consideration, there are limitations to our study. One limitation is that the project manager can maliciously modify the entire strategy, even if the strategy is reused, in the current prototype. The other one is that the current test strategy is classified into the test phases and the quality model. Therefore, if existing projects are not enough, it will be difficult to get test strategies that can achieve the desired test objectives.

7. Conclusion

For reducing test efforts, in this paper, we proposed a test strategy reuse technique with phases from existing projects. Also, test plan completeness prediction technique by its objectives was proposed. And we extended test planning processes in ISO/IEC 29119-2 to include the proposed techniques. We showed the possibility of our techniques by implementing a prototype in the existing framework with applying proposed test activities in the previous study.

Our future research will include distinguishing project-specific keywords(e.g. tester name, testing time, test completion criteria) through context awareness of test strategy. Then a project manager only can modify the project-specific keywords of existing test strategies. It will prevent the first limitation of Section 6.2. And we expect classifying the ISO/IEC 25010 quality model into phases would allow more accurate reuse technique.

References

[1] M. Grindal, J. Offutt, J. Mellin, "On the Testing Maturity of Software Producing Organizations," in Proc. of Testing: Academic & Industrial Conference - Practice And Research Techniques (TAIC PART'06), pp.171-180, 2006.
http://dx.doi.org/10.1109/TAIC-PART.2006.20

[2] ISO/IEC 25010, "Systems and Software Engineering: Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Model".

[3] ISO/IEC 25023, ″Systems and Software Engineering: Systems and Software Quality Requirements and Evaluation (SQuaRE) – Measurement of system and software product quality″.

[4] J. Kasurinen, P. Runeson, L. Riungu, K. Smolander, ″A self-assessment framework for finding improvement objectives with ISO/IEC 29119 test standard,″ in Proc. of Systems, Software and Service Process Improvement, Vol. 172, 2011. https://doi.org/10.1007/978-3-642-22206-1_3

[5] ISO/IEC 29119, ″Software and Systems Engineering - Software Testing - Part 2: Test Process″.

[6] R. Tiwari, N. Goel, ″Reuse: reducing test effort,″ ACM SIGSOFT Software Engineering Notes, Vol. 38, No. 2, pp.1-11, 2013. http://doi.org/10.1145/2439976.2439982

[7] K. Min, J. Lee, B. Lee, ″Test Strategy Reuse by Compatibility Prediction based on ISO/IEC 25010 Quality Model,″ in Proc. of KSII The 10th International Conference on Internet(ICONI), 2018.

[8] J. Al Dallal, P. Sorenson, ″Reusing class-based test cases for testing object-oriented framework interface classes: Research Articles,″ Journal of Software Maintenance and Evolution: Research and Practice, Vol. 17, No. 3, pp.169-196, 2005. http://dx.doi.org/10.1002/smr.308

[9] Y. Chen, R. Probert, K. Robeson, ″Effective Test Metrics for Test Strategy Evolution,″ in Proc. of the 2004 Conference of the Centre for Advanced Studies on Collaborative Research, pp. 111-123, 2004.

http://doi.org/10.1145/1034914.1034923

[10] S. Back, E. Lee, J. Lee, B. Lee, ″A Test Framework Verifying Artifacts in Software R&D Project,″ in Proc. of the Korea Computer Congress, pp. 588-590, 2015.

[11] S. Song, A. Dashbalbar, J. Lee, B. Lee, ″Test Framework Requirements to Verify Artifacts in Software R&D Project,″ International Journal of Software Engineering and Its Applications, Vol. 10, No. 11, pp.83-94, 2016. http://doi.org/10.14257/ijseia.2016.10.11.07

[12] A. Dashbalbar, E. Lee, J. Lee, B. Lee, ″Describing Activities to Verify Artifacts(Documents and Program) in Software R&D,″ Journal of Internet Computing and Services(JICS), Vol. 17 No. 2, pp.39-47, 2016. http://doi.org/10.7472/jksii.2016.17.2.39

[13] G. Yang, K. Min, J. Lee, B. Lee, ″Applying Topic Modeling and Similarity for Predicting Bug Severity in Cross Projects,″ KSII Transactions on Internet and Information Systems (TIIS), Vol. 13 No. 3, pp.1583-1598, 2019. http://doi.org/10.3837/tiis.2019.03.026

[14] A. Dashbalbar, S. Song, J. Lee, B. Lee, ″Towards Enacting a SPEM-based Test Process with Maturity Levels,″ KSII Transactions on Internet and Information Systems (TIIS), Vol. 11 No. 2, pp.1217-1233, 2017. http://doi.org/10.3837/tiis.2017.02.034

◐ 저 자 소 개 ◑

민 경 식(Kyeongsic Min)
2018년 서울시립대학교 컴퓨터과학부(학사)
2018년~현재 서울시립대학교 컴퓨터과학부 석사과정
관심분야 : 소프트웨어테스트, 소프트웨어공학, 블록체인, 스마트 컨트랙트
E-mail : ksmin1710@uos.ac.kr

이 정 원(Jung-Won Lee)
1993년 이화여자대학교 전자계산학과(학사)
1995년 이화여자대학교 전자계산학과(석사)
1995년~1997년 LG종합기술원 주임연구원
2003년 이화여자대학교 컴퓨터학과(박사)
2003년~2006년 이화여자대학교 컴퓨터학과 BK교수, 전임강사(대우)
2006년~현재 아주대학교 전자공학과 교수
관심분야 : Embedded Software, Automotive Software, Bio·Medical Data Modeling
E-mail : jungwony@ajou.ac.kr

이 병 정(Byungjeong Lee)
1990년 서울대학교 계산통계학과(학사)
1990년~1998년 (주)하이닉스반도체 연구원
1998년 서울대학교 전산과학과(석사)
2002년 서울대학교 전기·컴퓨터공학부(박사)
2002년~현재 서울시립대 컴퓨터과학부 교수
관심분야 : 소프트웨어테스트, 소프트웨어 진화, 소프트웨어공학
E-mail : bjlee@uos.ac.kr