



Factors Influence Novice Programmers toward Test First Approach

Norzariyah Yahya¹, Siti Sarah Maidin², Arjumand Bano Soomro³

¹ Department of Computer Science, Kuliyyah of Information and Communication Technology, International Islamic University Malaysia, Malaysia, norzariyah@iium.edu.my

² School of Computing, Faculty of Computing, Engineering and Technology Asia Pacific University of Technology & Innovation Kuala Lumpur, Malaysia, dr.sitisarah@apu.edu.my

³ Institute of Information and Communication Technology University of Sindh Jamshoro, Pakistan, arjumand@usindh.edu.pk

ABSTRACT

Test First is one of the Agile development approaches. In practice, *Test First* needs a developer to design test cases followed by the development of actual codes. The previous study on *Test First* has been covering the quality of the codes, either internal quality of codes, external quality of codes, or productivity of codes. Also, research on the behavior of the developers toward *Test First* based on the developers experiences implementing the *Test First* approach. This research is looking into the behavior of developers, which focus on finding the factors that influence novice programmers' to execute *Test First* by using the Theory of Planned Behavior as the theoretical framework. The Theory of Planned Behavior framework is used to identify the factors that contribute to the Intention of novice programmers' to implement *Test First*. The factors were identified quantitatively using a set of questionnaire. The results indicated that *Behavioral Beliefs*, *Attitude towards Behavior*, *Normative Beliefs*, and *Subjective Norms* are the factors that influenced novice programmers to implement *Test First*.

Key words: *Test First*, Theory of Plan Behavior.

1. INTRODUCTION

Bissi, et al. [1] reported that research on *Test First* is group into four: 1) observation of effect in the primary study in which the research was comparing between *Test First* and Test Last Development, 2) Internal quality, 3) External quality and 4) Productivity. While Munir, et al. [2] stated that research conducted on *Test First* are using three methods and the methods are experiment, case study, and questionnaire, while in later finding, similarly Bissi, et al. [1] reported that the research findings gathered from the experiment, case

study, questionnaire, and simulations. Based on Bissi, et al. [1] and Munir, et al. [2], it can be seen that the studies of the experiments, case studies or questionnaire on *Test First* involved participants either from industry or academia. While this research is focusing on the participants from a series of experiments on *Test First* implementation, aiming at the behavior of the participants on their intention to implement *Test First*. Essentially, this research is a post-experiment questionnaire which is a sub-component of a series of experiments. The post-experiment questionnaire was developed based on the Theory of Planned Behavior model, and it was distributed to groups of participants consist of novice programmers who have been experiencing *Test First* development approach.

2. RELATED STUDY

This section is divided into two parts: 1) discussion on the definition of *Test First* and Test Driven Development, also explaining the difference between *Test First* and Test Driven Development, and 2) description of Theory of Planned Behavior.

2.1 Test First and Test Driven Development

. Generating test cases at an early stage is efficient than having it after the development phase [3]. One of the methods uses test cases is *Test First*. *Test First* is an Agile development approach [4] and classified as one of the eXtreme Programming (XP) development method. Beck [5] called this approach as Test Driven Development, while Fowler, et al. [6] preferred *Test First*. Test Driven Development or *Test First* is the development process, which starts with the design of unit test cases followed by the actual production code [7]. Dalton [8] defined Test Driven Development as an agile technique where a developer will write a basic test case to verify the desired functionality, knowing that it will fail, and then writes the minimum

amount of code to pass the test. While, Astels [9] defines Test Driven Development as the incorporation of unit test cases (test first), and the actual production codes, wherein the actual production codes need to pass those written test cases. Unit test is used to each ensure that each components of software performed as it is designed. A unit test is one of the most effective techniques to mitigate the impact of unexpected behavior of software systems [10].

In general, *Test First* is a practice which needs a developer to write the unit test cases followed by the development of the production code, and the written actual production codes were meant to pass the test cases. Also, all coding processes are implemented in a small chunk of tasks iteratively. Moreover, *Test First* is a subset of Test Driven Development. Specifically, in the *Test First*, the developers do not execute refactoring, and may not keep test cases up to date following the progression of the production code [11]. In addition, basis of Test Driven Development is composed of three iterative phases, Red (write a unit test and the unit test will automatically fail), Green (write production code and the actual code is tested by the unit test and it must pass the test cases and Blue (refactor the code) [12].

Interestingly, by implementing the iterative phases of Test Driven Development, Romano, et al. [13] found that developers who participated in their research write quick-and-dirty production code to pass the tests, do not update their tests often, and ignore refactoring. The findings reported by Romano, et al. [13] are contradicted with Agile Manifesto [14] and as reported by Borle, et al. [15], who stated that Test Driven Development key practice is reducing costs and improving code quality. Thus, what are the factors that might draw the developers' intention to write and update test cases of a software project?

2.2 Theory of Planned Behavior

The Theory of Planned Behavior (TPB) was introduced by Ajzen [16], which has emerged since 1998. According to Ajzen and Fishbein [17], the Theory of Planned Behavior is based initially on the Theory of Reasoned Action (TRA) by added the behavioral control as part of the model. The Theory of Planned Behavior Model is a model that has been used to predict and to understand the behavior in various fields [18]. The Theory of Planned Behavior model categorized its model into two determinants: 1) the indirect determinants which consist of *Behavioral Beliefs*, *Normative Beliefs* and *Control Beliefs*, and 2) the direct determinants with *the Attitude towards Behavior*, *Subjective Norms* and *Perceived Behavioral Control*.

The relationship between direct determinants and indirect determinants can be concluded as follows:

- i. *Behavioral Beliefs* correspond to *Attitude towards Behavior*
- ii. *Normative Beliefs* is assumed to influence *Subjective Norms*
- iii. *Control Beliefs* has some impact on *Perceived Behavioral Control*.

All three direct determinants; *Attitude towards Behavior*, *Subjective Norms*, and *Perceived Behavioral Control* correspond as the predictor for novice programmers' intention implementing *Test First*, and intention is a predictor to the behaviors.

The first direct determinant, *Attitude towards Behavior* describes by Ajzen [18] as the desire of a person to appraise or evaluate the behavior by to approve or to disapprove the questions on the behavior. Thus, to study the novice programmers' behavior towards *Test First*, the agreement for the novice programmers either to write test cases followed by the actual code or to refuse is the question this research will answer. The *Subjective Norms* refers to social pressure to perform or not to perform the behavior. In this research, two factors identified are instructor and classmates that might encourage or discourage the novice programmers' to execute the *Test First* approach development phases. Finally, the *Perceived Behavioral Control* denotes the perceived ease or difficulty of performing the behavior, and it is assumed to reflect past experience as well as anticipated impediments and obstacles. Ajzen [18] stated that intention is the assumption to capture the motivational factors that influence the behavior.

3. METHOD

This research is explored quantitatively to identify the factors that contribute to the intention of novice programmers to implement *Test First* approach. This research adopted the Theory of Planned Behavior model, as illustrated in Figure 1 as the research framework. The questionnaire for this research was constructed based on Ajzen [19], Ajzen [16], and Ajzen [20]. The questionnaires were pilot tested, and the feedback gathered from the pilot test were analyzed using the thematic analysis. Thematic analysis is a process of data reduction and is one of the significant data analytics options in qualitative studies [21]. Thematic analysis is also known as classic content analysis [22].

Conversely, in content analysis, the codes' categories are not pre-determined, and the codes are derived from the data. The data from the open-ended questions are analyzed using segmentation by finding the keywords or phrases from the responses, as suggested by Grbich [21]. Amendments are done based on the outcome of the pilot questionnaire before it was distributed for the actual data collection.

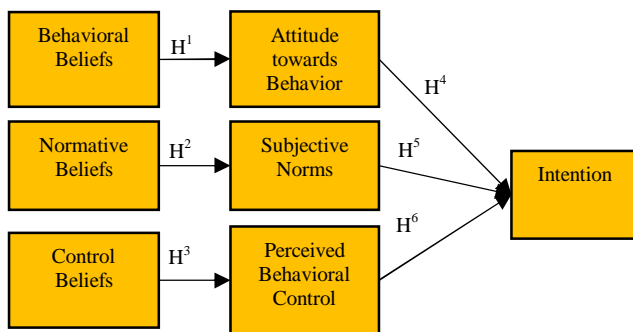


Figure 1: Hypotheses based on the Theory of Planned Behavior model.

3.1 Hypotheses

Six hypotheses were developed to identify the factors that might influence novice programming to implement *Test First*. The hypotheses are as follows:

- H¹: Favorable on *Test First* influences the *Attitude towards Behavior* to implement *Test First*.
- H²: Perception of others has a perceived social pressure on novice programmers’ on implementing *Test First*.
- H³: Novice programmers’ belief influences their self-efficacy towards *Test First*.
- H⁴: Novice programmers’ desire to implement *Test First* has a positive effect on Intention.
- H⁵: Social pressure influence novice programmers’ Intention towards *Test First*.
- H⁶: *Perceived Behavioral Control* has a positive influence on Intention.

4. ANALYSES AND RESULTS

Generally, 274 questionnaires were distributed to groups of novice programmers. However, 213 questionnaires were returned, and only 198 questionnaires were analyzed. Fifteen questionnaires were not analyzed due to incomplete form or missing values. The analyses started with the descriptive analyses, followed by reliability test, normality test, and lastly are the Spearman correlation.

4.1 Descriptive analyses

The descriptive analyses in Table 1 show the minimum distribution for each of the determinants are one, in which it indicated that the respondents are “Strongly agree” with the questions in the questionnaire. In contrast, the distributions of five to seven indicated that five is the respondents are “Slightly disagree,” six is “Disagree,” and seven is “Strongly disagree.” However, the mean for each determinant is two, wherein it indicated that the respondents agree that *Behavioral Beliefs*, *Normative Beliefs*, *Control Beliefs*,

Attitude towards Behavior and *Subjective Norms* are positive towards intention on *Test First*.

Table 1: Descriptive analyses

Determinants	Min	Max	Mean	Std. Deviation
Behavioral Beliefs	1.00	5.00	1.9327	.8322
Normative Beliefs	1.00	4.50	2.0884	.8207
Control Beliefs	1.00	6.00	2.0960	.8532
Attitude Towards Behavior	1.00	5.00	2.0328	.9318
Subjective Norms	1.00	5.00	2.2652	.9244
Perceived Behavioral Control	1.00	7.00	2.6237	1.2639
Intention	1.00	7.00	2.7727	1.1260

4.2 Reliability test

The reliability is used to measure the internal consistency of the data collected. All of the determinants are tested based on Cronbach’s Alpha (CA) >0.69 [23]. The result is presented in Table 2 shows that all determinants are reliable data except for *Normative Beliefs* (CA=.477) and *Perceived Behavioral Control* with CA = .632, which indicated low reliability. However, since all the items from each determinant were developed based on guidelines given by Ajzen, and the questionnaire was pilot tested, and the result of reliability test from the pilot test was observed that *Normative Beliefs* were having CA=.787 and *Perceived Behavioral Control* CA=.747 thus both factors were analyzed for its correlation.

Table 2: Reliability Test

Determinant	Cronbach’s Alpha (CA)	n of items
Behavioral Beliefs	.805	3
Normative Beliefs	.477	2
Control Beliefs	.795	3
Attitude Towards Behavior	.854	2
Subjective Norms	.814	2
Perceived Behavioral Control	.632	2
Intention	808	3

4.3 Normality test

Table 3 shows the outcome of normality test result from both Kolmogorov-Smirnov test and Shapiro-Wilk test. Based on Field [24], the Kolmogorov-Smirnov and Shapiro-Wilk Sig <.05 shows that the data significantly deviates from the normal distribution. However, the result of the distribution of data in this research is reported using the Shapiro-Wilk test. The distributions indicated that all determinants are not

fitting the normal distribution (Sig=.000) thus to continue with the correlation test, Spearman-Rho correlation is executed.

Table 3: Normality test

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
Behavioral Beliefs	.185	198	.000	.892	198	.000
Normative Beliefs	.174	198	.000	.923	198	.000
Control Beliefs	.196	198	.000	.909	198	.000
Attitude Towards Behavior	.226	198	.000	.876	198	.000
Subjective Norms	.173	198	.000	.930	198	.000
Perceived Behavioral Control	.173	198	.000	.930	198	.000

a.Lilliefors Significance Correction

4.4 Correlation

The results in Table 4 are generated based on Spearman-Rho correlation. Spearman correlation coefficient is used to measure the strength of the relationship between dimensions illustrated in Figure 1. The results show that *Behavioral Beliefs* is correlated with *Attitude towards*

Behavior ($r=.479$ and $p\text{-value}=.000$), while *Normative Beliefs* is correlated with *Subjective Norms* ($r=.382$ and $p\text{-value}=.000$) and *Control Beliefs* is correlated with *Perceived Behavioral Control* ($r=.264$ and $p\text{-value}=.000$). Positively, *Attitude towards Behavior* and *Subjective Norms* are correlated with *Intention*; $r = .405$ and $p\text{-value}=.000$, $r=.358$ and $p\text{-value}=.000$ respectively. The strength of the relationship between each determinant can be concluded as weak, wherein the strongest among the dimensions is *Behavioral Beliefs* with *Attitude towards Behavior*, $r = .479$.

Thus, based on Table 4, the results derived from the Spearman’s rho correlation suggested accepting the following hypotheses:

- H¹: Favorable on *Test First* influences the *Attitude towards Behavior* to implement *Test First*.
- H²: Perception of others has a perceived social pressure on novice programmers’ on implementing *Test First*.
- H³: Novice programmers’ belief influences their self-efficacy towards *Test First*.
- H⁴: Novice programmers’ desire to implement *Test First* has a positive effect on *Intention*.
- H⁵: Social pressure influence novice programmers’ *Intention towards Test First*.

And rejecting H⁶: *Perceived Behavioral Control* has a positive influence on *Intention*.

Table 4: Spearman’s rho correlation for intention towards *Test First*

		Behavioral Beliefs	Normative Beliefs	Control Beliefs	Attitude Towards Behavior	Subjective Norms	Perceived Behavioral Control	Intention
Behavioral Beliefs	Corr. Coe.	1.000	.458**	.512**	.479**	.281**	.290**	.343**
	Sig. (2-tailed)	.	.000	.000	.000	.000	.000	.000
Normative Beliefs	Corr. Coe.		1.000	.451**	.424**	.382**	.196**	.450**
	Sig. (2-tailed)		.	.000	.000	.000	.006	.000
Control Beliefs	Corr. Coe.			1.000	.556**	.237**	.264**	.290**
	Sig. (2-tailed)			.	.000	.001	.000	.000
Attitude Towards Behavior	Corr. Coe.				1.000	.346**	.190**	.405**
	Sig. (2-tailed)				.	.000	.007	.000
Subjective Norms	Corr. Coe.					1.000	.089	.358**
	Sig. (2-tailed)					.	.212	.000
Perceived Behavioral Control	Corr. Coe.						1.000	.118
	Sig. (2-tailed)						.	.098
Intention	Corr. Coe.							1.000
	Sig. (2-tailed)							.

** . Correlation is significant at the 0.01 level (2-tailed).

4.5 Regression Analysis

Regression analysis is a way to predict an outcome variable from one predictor variable (simple regression) or several predictor variables (multiple regressions) [24]. Field [24] also suggested that the statistical power estimation is executed in order to check for the probability that researchers correctly reject any null hypothesis.

In this research, regression is used to predict the weight of the dimensions studied. Figure 2 indicated that the *Behavioral Beliefs* is observed to be the moderate factor for *Attitude towards Behavior* at 46% and *Attitude towards Behavior* is a moderate factor (40%) for intention to write *Test First*. Consistently, the result indicated that *Normative Beliefs* correlates with the *Subjective Norms*; both dimensions are a predictor for intention with the strength of 39% and 38% consecutively. However, *Perceived Behavioral Control* was found not to be significant ($p\text{-value}=.299$); thus, it is consistently not a predictor for intention.

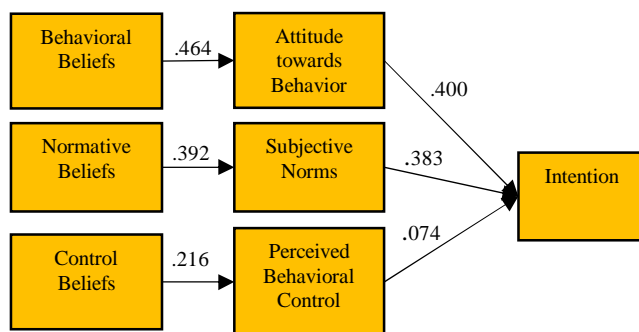


Figure 2: Path diagram of the Theory of Planned Behavior

5. DISCUSSION

Donald, et al. [25] stated that the central principle of the Theory of Planned Behavior is the model indicated a process from beliefs to behavior involved an individuals to believe and to perform, to process the idea of performing the behavior and to utilize the information available which lead them to decide either to positively accept the behavior or to reject the behavior. In which, this research is to explore the factor that might be accepted by the novice programmer to perform *Test First* as a development approach for their project.

The above results suggested that *Behavioral Beliefs*, *Attitude towards Behavior*, *Normative Beliefs*, and *Subjective Norms* are the factors that influenced the *Intention* to implemented *Test First*. By definition, *Behavioral Beliefs* is focusing on novice programmers' beliefs on the classes, training and coding experience helps them to understand *Test First* and to implement *Test First* while *Attitude towards*

Behavior refers to the degree to the novice programmer favorable and unfavorable evaluation or appraisal of novice programmer's to develop their project using *Test First* approach. Moving towards the next factor that given positive influence towards *Test First* is *Normative Beliefs* and *Subjective Norms*. *Subjective Norms* refers to the perceived social pressure to perform or not to perform the behavior [26]. Two entities, the instructor and classmates, are identified as a factor for the *Normative Beliefs* and *Subjective Norms*. Thus, in this research, novice programmers' beliefs in expectation from others such as their instructor, classmates, and also on what other people think will influence them towards *Test First*. The findings suggested that besides lecturing and training that assist or hinder novice programmers from testing, enforcement in a class by the lecturer and pressure from classmates or groupmates will increase the positive value of *Intention* towards *Test First*.

Control Beliefs refers to an individual's beliefs about the presence of factors that may facilitate or impede the performance of behavior [26], while *Perceived Behavioral Control* refers to people's perception of the ease or difficulty in performing the behavior of interest [27]. In this research, *Control Beliefs* and *Perceived Behavioral Controls* emphasize on novice programmers' expectation on their knowledge on writing codes, their programming skills and time that they have to develop will influence them to do *Test First*. However, in this research, both *Control Beliefs* and *Perceived Behavioral Controls* are found as a weak factor that leads novice programmer to do *Test First*.

In conclusion, the *Intention* towards *Test First* among novice programmers in this higher education institute is contributed by the view of the people that encourage them to test such as lecturers/instructors and classmates. Training, classes, and lectures on test cases are also identified as the factors that will help them in implementing the development approach. The stronger the intention to engage in a behavior, the more likely the novice programmers to write test cases in their project, however, the behavior still depends on the willingness to perform or not to perform the behavior with the non-motivational factors such as skill, money or cooperation from others

REFERENCES

1. W. Bissi, A. G. S. S. Neto, and M. C. F. P. Emer, **The effects of test driven development on internal quality, external quality and productivity: A systematic review**, *Information and Software Technology*, vol. 74, pp. 45-54, 2016. <https://doi.org/10.1016/j.infsof.2016.02.004>

2. H. Munir, M. Moayed, and K. Petersen, **Considering rigor and relevance when evaluating test driven development: A systematic review**, *Information and Software Technology*, vol. 56, pp. 375-394, 2014.
<https://doi.org/10.1016/j.infsof.2014.01.002>
3. L. Rajamanickam, N. A. Mat Saat, and S. N. Daud, **Software Testing: The Generation Tools**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, pp. 231-234, 2018.
<https://doi.org/10.30534/ijatcse/2019/20822019>
4. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, *et al.*, **The agile manifesto**, 2001a.
5. Beck, **Test-driven Development: By Example**, Addison-Wesley, 2003.
6. M. Fowler, K. Beck, J. Brant, W. Opdyke, and D. Roberts, **Refactoring: Improving the Design of Existing Code**, Pearson Education, 2012.
7. B. Rumpe, **Agile test-based modeling**, *arXiv preprint arXiv:1409.6616*, 2014.
8. J. Dalton, **Test-Driven Development**, *Great Big Agile*, Springer, 2019, pp. 263-264.
https://doi.org/10.1007/978-1-4842-4206-3_67
9. D. Astels, **Test-driven Development: A Practical Guide**, Prentice Hall PTR, 2003.
10. J. P. Pires and F. B. e Abreu, **Knowledge Discovery Metamodel-Based Unit Test Cases Generation**, in *11th International Conference on Software Testing, Verification and Validation (ICST), IEEE*, 2018, pp. 432-433.
<https://doi.org/10.1109/ICST.2018.00056>
11. I. Karac and B. Turhan, **What Do We (Really) Know about Test-Driven Development?**, *IEEE Software*, vol. 35, pp. 81-85, 2018.
<https://doi.org/10.1109/MS.2018.2801554>
12. D. Fucci, G. Scanniello, S. Romano, M. Shepperd, B. Sigweni, F. Uyaguari, B. Turhan, N. Juristo and M. Oivo, **An External Replication on the Effects of Test-driven Development Using a Multi-site Blind Analysis Approach**, in *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 2016, p. 3.
<https://doi.org/10.1145/2961111.2962592>
13. S. Romano, D. Fucci, G. Scanniello, B. Turhan, and N. Juristo, **Findings from a multi-method study on test-driven development**, *Information and Software Technology*, vol. 89, pp. 64-77, 2017.
<https://doi.org/10.1016/j.infsof.2017.03.010>
14. A. Alliance, **Manifesto for Agile Software Development**, ed, 2001.
15. N. C. Borle, M. Fegghi, E. Stroulia, R. Greiner, and A. Hindle, **Analyzing the effects of test driven development in GitHub**, *Empirical Software Engineering*, vol. 23, pp. 1931-1958, 2018.
<https://doi.org/10.1007/s10664-017-9576-3>
16. I. Ajzen, **Perceived Behavioral Control, Self-Efficacy, Locus of Control, and the Theory of Planned Behavior**, vol. 32, pp. 665-683, 2002.
<https://doi.org/10.1111/j.1559-1816.2002.tb00236.x>
17. I. Ajzen and M. Fishbein, **The influence of attitudes on behavior**, *The handbook of attitudes*, vol. 173, 2005.
18. I. Ajzen, **From intentions to actions: A theory of planned behavior**, in *Action control*, ed: Springer, 1985, pp. 11-39.
https://doi.org/10.1007/978-3-642-69746-3_2
19. I. Ajzen, **Theory of Planned Behaviour Questionnaire. Measurement Instrument Database for the Social Science**, 2013.
20. I. Ajzen, **Constructing a TpB Questionnaire: Conceptual and Methodological Considerations**, 2002 revised 2006.
21. C. Grbich, **Qualitative data analysis: An introduction**, Sage, 2013.
22. G. Guest, K. M. MacQueen, and E. E. Namey, **Applied thematic analysis**, Sage, 2011.
<https://doi.org/10.4135/9781483384436>
23. A. Field, **Discovering statistics using IBM SPSS statistics**, Sage, 2013.
24. A. Field, **Discovering statistics using SPSS:(and sex and drugs and rock 'n'roll). Introducing statistical methods**, ed: London: Sage, 2009.
25. I. Donald, S. Cooper, and S. Conchie, **An extended theory of planned behaviour model of the psychological factors affecting commuters' transport mode use**, *Journal of Environmental Psychology*, vol. 40, pp. 39-48, 2014.
<https://doi.org/10.1016/j.jenvp.2014.03.003>
26. I. Ajzen, **The theory of planned behaviour**, *Organizational Behaviour and Human Decision Processes*, vol. 2, 1991.
[https://doi.org/10.1016/0749-5978\(91\)90020-T](https://doi.org/10.1016/0749-5978(91)90020-T)
27. I. Ajzen, **Nature and operation of attitudes**, *Annual review of psychology*, vol. 52, pp. 27-58, 2001
<https://doi.org/10.1146/annurev.psych.52.1.27>