Software Quality Prediction by using Structural Equation Model

Somchai Prakancharoen
Faculty of science, Rajabhat Chandrakasem University
Bangkok, Thailand
somchai.p@chandra.ac.th

Niyom Sutthaluang

Faculty of science, Rajabhat Chandrakasem University
Bangkok, Thailand
Niyom.s@chandra.ac.th

***Abstract* – Software quality is a topic that both software developer and customer have same direction needs. The objective of this research was to find out the factors that have direct effect with software development quality. Two hundred and fifty sample data were collected from twenty small size software development firms during 2018-2019. The software projects were considered collect from small software development firm. The software project, in this research, was chosen from software project that finish time of software development was less than six months. Factor analysis and Structural equation modelling were used to explore and confirm goodness of fit model. The result saturated model was cross validated by fifty testing finished software project. The accuracy of prediction of model was about 71.50 percent.**

***Keywords* - Software quality, prediction, factor analysis, structural equation model.**

I. INTRODUCTION

Software quality meaning in this research was covered about software reliability, performance, maintainability and rate of delivery of developed software.

Good software product should have characteristics that represent prefer qualities.

Reliability: software should have small number of errors or bugs, low mean time between errors: MTBE as much as possible.

Performance: software processing time should take a short time period.

Security: software should have less vulnerabilities.

Maintenance: software should easily maintain in order to preserve system availability.

Rate of delivery: customers want to launce procured information system on planned time.

The objective of this research was to analyze what characters were related to software quality.

Software developer should concern on those significant characters in order to increase developing software quality.

II. RELATED THEORY AND RESEARCH

A.Theory

- Factor analysis [1]

Factor analysis is a method that is used to construct a new latent variable from related attributes. There should be formed up more than one factor. Another reason of factor analysis doing is that factor analysis can reduce number of attribute and eliminate multi co-linearity problem between related attributes. Relate attributes were grouped to be a new latent variable so that multi co-linearity problem are reduced.

Cronbach's alpha test is used to check whether attribute grouping is good enough or not. Cronbach's alpha score should greater than or above 0.7.

Kaiser-Meyer-Olkin (KMO) test is a technique that is used to test whether data sampling is adequate or not. KMO value should be greater than 0.6.

- Structural equation model [2]

Structural equation model, SEM, is a method that is used to test whether proposed causal model is suitable or not.

User has to construct his casual model under his research hypothesis. Normally, causal model is constructed from relation between many factors.

There are many statistics test about model fitting. First, Chi square test, CMIN, is used to test overall fitting of purposed model. The "p" value of chi square test should be significant less than 0.05.

Root mean square error of approximation (RMSEA) is used to test error value of model this value should be less than 0.05.

Adjusted goodness of fit index (AGFI) is used to test goodness of model. AGFI value ought to greater 0.9.

- Cross validation [3]

Saturated structural equation that was gained from SEM method was calculated from training data observation. This model ought to be tested with other data observation, call as testing data. Less difference value between its real (actual) value and predicting value show that if a model has a good accurate prediction.

Mean magnitude of relative error (MMRE) is a technique that is used to present average amount of magnitude relative (absolute) error.

$$MMRE = \frac{1}{n} \cdot \frac{\sum_{i=1}^{n} |actual - estimated|}{actual} \quad (1)$$

Where "n" is a number of testing observations that are used in model accuracy testing.

B. Related research

Gilad [4] presents that there are five factors that related to software quality of software development.

Factor-1 – Reliability (RELY) constraint which are composed of failure rate (FAIE), mean time between failure (MTBF).

Factor-2-Performance constraint (PERE) which are composed of load testing (LOAG), stress testing (STRG), soak testing (SOAG).

Factor-3-Security requirement (SECY) which are composed of number of vulnerabilities (NUMS), time to resolution (TIMN), deployment of security update (DEPT).

Factor -4-Maintanability (MAIY) which are composed of software complexity (SOFY), software static code analysis (SOFS), Software size (SOFE).

Factor -5- Rate of delivery (DELY) which are composed of number of software release (NUME), delivered rate (DELE), user consumption of release (USEE).

Factor -7- Usability which are composed of satisfaction level (SATL), completion rate (COME).

Bijay Jayaswal & Peter Patton [5] suggest that program quality could be measured by many software metrics such as capability of developer team, software usability, performance, reliability, instability, maintainability, documentation and availability.

III. RESEARCH METHODOLOGY

A. Limitation

Data observations were collected from twenty small software development firms in Thailand.

Software project program were chosen just only small size of program which could be finished in about six months or less than six months.

This limitations should provide similarity environments of software development firm or project.

B. Sample

Two hundred finished software projects were preserved as training data set while fifty finished software project were set as testing data set. Data gathering were conducted during May 2018-May 2019.

C. Hypothesis

- Exploratory for factor

Factor-1-Reliability (RELY) constraint was composed of failure rate (FAIR), mean time between failure (MTBF), software availability (SOFA).

Factor-2-Performance (PERE) constraint was composed of load testing (LOAT), stress testing (STRT), soak testing (SOAT).

Factor-3-Maintainability (MAIY) which was composed of software complexity (SOFC), software size (SOFS).

Factor-4-Rate of delivery (DELY) which was composed of number of software release (NUME), delivered rate (DELE)

Factor-5-Usability (USAY) which was composed of satisfaction level (SATF), requirement completion rate (REQC)

Factor -6- Software quality (SOFQ) which was composed of Software quality class attribute (SQCA).

- Research Hypothesis-Confirmation

H1: Factor-Reliability (RELY) has direct effect to factor software quality (SOFQ).

H2: Factor-Performance constraint (PERE) has direct effect to factor software quality (SOFQ).

H3: Factor-Maintainability (MAIY) has direct effect to factor software quality (SOFQ).

H4: Factor-Rate of delivery (DELY) has direct effect to factor software quality (SOFQ).

H5: Factor-Usability (USAY) has direct effect to factor software quality (SOFQ).

D. Questionnaire

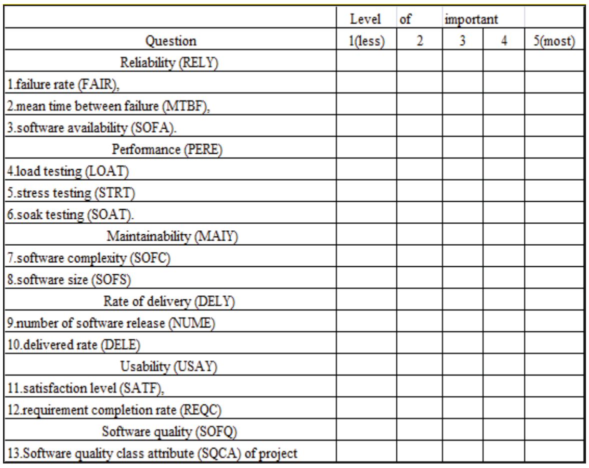
Questionnaire was composed of two parts.

Part-1: respondent give information of finished project about duration, amount of software project work months**.**

Part -2: respondent gave answer about thirteen attributes. Thirteenth attribute was respondent opinion in overall quality of answering in finished software project development.

Likert's scale 1-5 was used to rating each question. Level "1" means on low important while "5" is high important.

TABLE I. PART -2 QUESTIONNAIRE DETAIL



	Level	of	important		
Question	1(less)	2	3	4	5(most)
Reliability (RELY)					
1.failure rate (FAIR),					
2.mean time between failure (MTBF),					
3.software availability (SOFA).					
Performance (PERE)					
4.load testing (LOAT)					
5.stress testing (STRT)					
6.soak testing (SOAT).					
Maintainability (MAIY)					
7.software complexity (SOFC)					
8.software size (SOFS)					
Rate of delivery (DELY)					
9.number of software release (NUME)					
10.delivered rate (DELE)					
Usability (USAY)					
11.satisfaction level (SATF),					
12.requirement completion rate (REQC)					
Software quality (SOFQ)					
13.Software quality class attribute (SQCA) of project					

E. Descriptive statistics

Two hundred and fifty finished software development projects were collected from twenty software development firm. Data cleaning was performed in order to check observation if they were matched to research condition. Data outlier detection was used to detect bias data observation before sent them in to next statistical data processing tasks. Fifty four percent of software project were finished developing in about six months.

TABLE II. PERCENTAGE OF PROGRAMMER USING IN EACH SOFTWARE PROJECT

Amount of month	Percentage
1	-
2	-
3	2
4	14
5	30
6	54
Total	100

F. Factor analysis

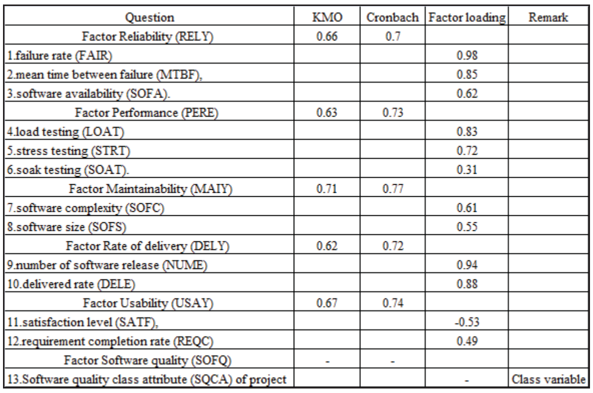
- Exploratory

Factor method was used to explore for its attribute (member) on five proposed factors.

"Principal component analysis" technique and "vari-max" rotational technique were chosen.

The result of exploratory factor analysis was shown in table 3. All attributes were grouped into proposed assigned factor with significant statistics, KMO and Cronbach alpha test.

TABLE III. FACTOR AND ITS RELATED ATTRIBUTES



Question	KMO	Cronbach	Factor loading	Remark
Factor Reliability (RELY)	0.66	0.7		
1.failure rate (FAIR)			0.98	
2.mean time between failure (MTBF),			0.85	
3.software availability (SOFA).			0.62	
Factor Performance (PERE)	0.63	0.73		
4.load testing (LOAT)			0.83	
5.stress testing (STRT)			0.72	
6.soak testing (SOAT).			0.31	
Factor Maintainability (MAIY)	0.71	0.77		
7.software complexity (SOFC)			0.61	
8.software size (SOFS)			0.55	
Factor Rate of delivery (DELY)	0.62	0.72		
9.number of software release (NUME)			0.94	
10.delivered rate (DELE)			0.88	
Factor Usability (USAY)	0.67	0.74		
11.satisfaction level (SATF),			-0.53	
12.requirement completion rate (REQC)			0.49	
Factor Software quality (SOFQ)	-	-		
13.Software quality class attribute (SQCA) of project			-	Class variable

G. Structural equation model

- Proposed structural equation model

Proposed structural equation model was illustrated in figure 1. Factors and their attributes were brought from result in table 3.

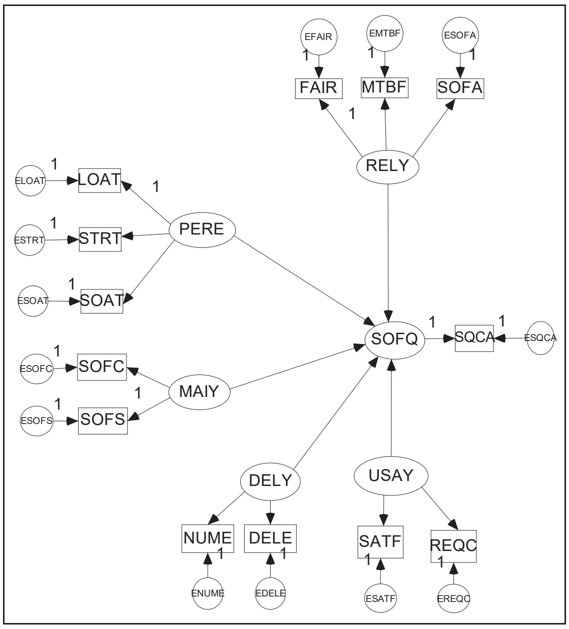
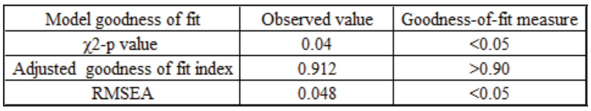


Fig 1. Propose structural equation model

- Saturated structural equation model

Proposed structural model was estimated its significant of all statistics by using maximum likelihood estimation technique. Standardized statistical coefficients were calculated and be tested on their significant. Model fit index were shown in table 4.

TABLE IV. STATISTICAL MODEL FIT INDEX



Model goodness of fit	Observed value	Goodness-of-fit measure
χ2-p value	0.04	<0.05
Adjusted goodness of fit index	0.912	>0.90
RMSEA	0.048	<0.05

Chi square test and its p value was significant value since its value is 0.04, less than criteria < 0.05.

Root mean square error of approximation (RMSEA) was significant since its value is less than 0.05.

Adjusted goodness of fit index (AGFI) value, 0.048, was less than 0.05.

Saturated structural equation model that fitted was illustrated in figure 2.

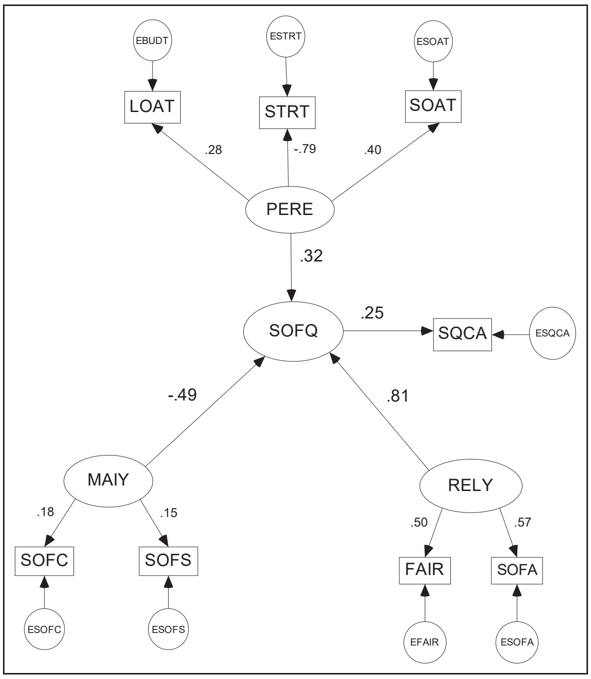


Fig 2. Saturated structural equation model (standardized form)

Hypothesis #4, #5 were not significant direct effect to factor software project development quality (SOFQ). There were only three hypothesis #1, #2 and #3 that passed criteria of model fitting significant testing.

Hypothesis #1, software reliability factor has a direct effect to software quality in positive amount (+0.81). Attributes failure rate (FAIR) and software availability (SOFA) were attributes of software reliability factor.

Hypothesis #2, factor performance (PERE) has a positive direct effect to software quality (+0.32). All attributes were grouped into factor PERE. Factor PERE was significant direct effect to factor SOFQ.

Hypothesis #3, factor maintainability (MAIY) has direct effect to SOFQ in negative standardized coefficient at "-0.49". Attribute "SOFC" and "SOFS" were component of factor "MAIY".

Dependent variable SQCA could be calculated from equation (2).

$$SQCA = 0.25SOFQ - 0.02; esqca = 0.02 \quad (2)$$

$$SOFQ = 0.32PERE - 0.49MAIY + 0.81RELY \quad (3)$$

$$PERE = 0.28LOAT - 0.79STRT + 0.40SOAT \quad (4)$$

$$MAIY = 0.18SOFC + 0.15SOFS \quad (5)$$

$$RELY = 0.50FAIR + 0.57SOFA \quad (6)$$

(Standard error: ELOAT=0.33, ESTRT=0.25, ESOAT=0.24, ESOFC=0.31, ESOFS=0.14, EFAIR=0.21, ESOFA=0.17)

H. Cross validation

SQCA class attribute of SOFQ factor was set to be as dependent variable.

Fifty testing observations were accuracy testing by MMRE technique. SQCA attribute was dependent variable predicting value that could be calculated from equation (2)-(6). These predicted SQCA were compared to their actual SQCA value from testing observations. Average magnitude error, MMRE, was calculated from equation (1).

Average of "Magnitude relative error: MRE", MMRE, value of cross validation testing was 0.285 hence percentage of estimation error was 28.50 percent.

Therefore, the percent of correctness prediction was about 71.50 percent, 100-28.50.

IV. CONCLUSIONS

There were only seven attributes or three factors that have direct effect on software quality.

Software reliability factor was most important since it was a positive large coefficient (0.81) to factor software quality.

Software performance (PERE) was secondary important direct effect to factor SOFQ with 0.32 significant coefficient.

Software maintainability (MAIY) was negative direct effect (-0.49) to factor SOFQ. Attribute software complexity (SOFC), software size (SOFS) were direct effect in negative direction. Thus, increase attention in software complexity and software size should reduce software quality since quality was difficult to reach.

Software developer should concern about software maintainability (MAIY), Developing software should not has a large software size. Object oriented software development should be applied in programming paradigm. Software complexity metrics should be used to detect software complexity if software has more complexity or not. Software design may be refactoring in order to reduce its complexity.

REFERENCE

[1] Olkin, A.R. Sampson, "Multivariate Analysis: Overview", International Encyclopedia of the Social& Behavioral Sciences, 2001.

[2] James L. Arbuckle, "Easy Structural Equation Modeling and Confirmatory Factor Analysis", 1st Edition, Psychology Press;1996.

[3] Devijver, Pierre A.; Kittler, Josef (1982). "Pattern Recognition: A Statistical Approach", London, GB: Prentice-Hall. ISBN 0-13-654236-0.

[4] Gilad Maayan, "What Software Quality (Really) Is and theMetrics". available at: https://www.altexsoft.com/blog/engineering/what-software-quality-really-is-and-the-metrics-you-can-use-to-measure-it/ on 8 august 2019.

[5] Bijay Jayaswal & Peter Patton, "Software Quality Metrics", Developer.com. Available on 8 august 2019 at:https://www.developer.com/tech/article.php/3644656/SoftwareQualityMetrics.html