



msceis
2016

Proceedings

International Seminar on
Mathematics, Science, and Computer Science Education

**“Harnessing Local Wisdom to Build Competencies
of Excellence in Research and Collaboration
in The New Era of The ASEAN Economic Community”**



Published by :
Faculty of Mathematics and Science Education
Universitas Pendidikan Indonesia
Jl. Dr. Setiabudhi No. 226 Bandung, 40154, West Java Indonesia
Official website: <http://fpmipa.upi.edu>

ISBN : 978-602-95549-4-6

Proceeding Proceedings International Seminar on Mathematics, Science, and Computer Science Education 2016

Tema :

***“Harnessing Local Wisdom to Build Competencies of Excellence in
Research and Collaboration in The New Era of The ASEAN Economic
Community”***

Editor :

Dr. Topik Hidayat, M.Si.
Dr. Eng.Asep Bayu Dani Nandiyanto, M.Eng.
Al Jupri, Ph.D.
Irma Rahma Suwarna, Ph.D.
Lala Septem Riza, Ph.D.
Dr. Diah Kusumawati, M.Si.
Dr. Endi Suhendi, M.Si.
Dr. Heli Siti Halimatul Munawaroh, M.Si.
Dr. Siti Aisyah, M.Si.
Marthalina Iriani, M.T.
Reza Setiawan, M.T.
Ika Mustika Sari, M.PFis
Duden Saepuzaman, M.Pd., M.Si.

Layout and Design by :
Umar, S.Pd

Published by :
**Faculty of Mathematics and Science Education
Universitas Pendidikan Indonesia
Jl. Dr. Setiabudhi No. 226 Bandung, 40154, West Java, Indonesia
Official website: <http://fpmipa.upi.edu>**

PREFACE

We are pleased to welcome all of the participants to Second International Seminar on Mathematics, Science and Computer Science Education (MSCEIS 2016). MSCEIS has been started since 2013 as an International Seminar of Mathematics, Science and Computer Science Education. This seminar is motivated by improving the quality of mathematics, science and computer science education. The aims of the seminar are: (1) To bring together the scientists, education experts and practitioners, students, and civil society organization representatives in the scientific forum. (2) To share and to discuss theoretical and practical knowledge about innovation in mathematics, science and education.

MSCEIS 2016 is the 3rd seminar and it will be held every year to provide forum for researchers in Mathematics, Science and Computer Science Education to share new ideas or research result in their field. The theme for this seminar is ***“Harnessing Local Wisdom to Build Competencies of Excellence in Research and Collaboration in The New Era of The ASEAN Economic Community”***. This seminar is organized by FPMIPA UPI and SEAMEO QITEP in Science and sponsored by Universitas Pendidikan Indonesia (UPI).

MSCEIS will be held every year to provide forum for researchers in Mathematics, Science and Computer Science Education to share new ideas or research result in their field. The theme for this seminar is ***“Improving the Quality of Mathematics, Science and Computer Science Education through Research”***. This seminar is sponsored by FPMIPA UPI.

The scope of research results to be presented and discussed in this seminar covers Pure and Applied Mathematics, Science and Technology, Information and Technology, Mathematics, Science and Computer Science Education.

The MSCEIS 2016 Program features 13 invited speakers and 398 contributed oral presentations, which come from different countries: Jerman, Australia, Hongkong, Jepang, USA and Indonesia. All papers reviewed before and after they are presented in this event. Selected articles will be published by AIP Proceedings (indexed by scopus and ISI Thomson Reuters) and IOP Journal of Physics: Conference Series (indexed by scopus and ISI Thomson Reuters). Other papers will be published either in Jurnal Pengajaran MIPA (JPMIPA) or in an ISBN proceedings indexed by Google Scholar and IPI.

To all participants, we hope that you will learn new subjects, make new contacts, and have fruitful discussions with others. To overseas participants, we wish you a pleasant stay in Bandung.

Finally, we wish to express our sincere appreciation to all of the presenters for their valuable contributions and also to the members of the program committee for their excellent works in selecting abstracts and organizing the program.

Desember , 2016

MSCEIS 2016 Committee

THE COMMITTEE

Steering Committee

Chairperson

: Siti Fatimah, S.Pd., M.Si., Ph.D.

Members

1. Dr. Nahadi, M.Pd.
2. Dr. Indarjani (SEAMEO QITEP in Science)
3. Prof. Yaya Sukjaya S Kusumah, M.Sc., Ph.D.
4. Prof. Dr. Munir, M.IT.
5. Dr. Andhy Setiawan, M.Si.
6. Dr. Bambang Avip Priatna
7. Dr. Bambang Supriatno
8. Dr. Dadi Rusdiana, M.Si.
9. Dr. Diana Rochintaniawati, M.Ed.
10. Dr. Didik Priandoko
11. Dr. Hendrawan
12. Dr. Mimin Nurjhani Kusumastuti
13. Dr. Mudzakir
14. Dr. Muslim, M.Pd.
15. Dr. Sjaeful Anwar
16. Dr. Sufyani Prabawanto, M.Ed.
17. Entit Puspita, M.Si.
18. Eddy Prasetyo Nugroho, M.T.
19. Jajang Kusnendar, M.T.

International Advisory Board

1. Prof. Dr. Triyanta
(Director of SEAMEO QITEP in Science)
2. Prof. Khairurrijal
(Institut Teknologi Bandung, Indonesia)
3. Prof. Dr. H. Rr. Hertien K. Surtikanti
(Universitas Pendidikan Indonesia,
Indonesia)
4. Prof. Dr. H. R. Asep Kadarohman,
M.Si. (Universitas Pendidikan
Indonesia, Indonesia)
5. Prof. Dr. H. Moejiono, M.Sc.
(Universitas Budi Luhur, Indonesia)
6. Prof. Dato' Dr. Aziz bin Dreman
(Universiti Malaysia Terengganu,
Malaysia)
7. Prof. Jhon Wiliamson
(University of Tasmania, Australia)
8. Prof. Ilin Kai Wu
(National Taiwan Normal University,
Taiwan)

9. Prof. Dr. Munir, M.IT.
(Universitas Pendidikan Indonesia, Indonesia)
10. Prof. Bruce Waldrip
(University of Tasmania Australia)
11. Prof. Peter Aubusson
(University of Technology Sydney, Australia)
12. Prof. Lim Cher Ping
(The Hongkong Institute of Education)
13. Assoc. Prof. Dr. Agus Setiabudhi, M.Si.
(Universitas Pendidikan Indonesia, Indonesia)
14. Assoc. Prof. Dr. Turmudi, M.Ed.
(Universitas Pendidikan Indonesia)

Scientific Committee

1. Prof. Dr. Didi Suryadi, M.Ed.
(Universitas Pendidikan Indonesia, Indonesia)
2. Assoc. Prof. Rizky Rosjanuardi
(Universitas Pendidikan Indonesia, Indonesia)
3. Prof. Liliarsari, M.Pd.
(Universitas Pendidikan Indonesia, Indonesia)
4. Prof. Dr. Hj. Nuryani Y. Rustaman, M.Pd.
(Universitas Pendidikan Indonesia, Indonesia)
5. Prof. Dr. Hj. Anna Permanasari, M.Si.
(Universitas Pendidikan Indonesia, Indonesia)
6. Assoc. Prof. Dr. Harry Finnan, M.Pd.
(Universitas Pendidikan Indonesia, Indonesia)
7. Prof. Tatang Herman; M.Ed.
(Universitas Pendidikan Indonesia, Indonesia)
8. Assoc. Prof. Dr. Parlindungan Sinaga, M.Si.
(Universitas Pendidikan Indonesia, Indonesia)
9. Assoc. Prof. Dr. John Jenny
(University of Tasmania, Australia)
10. Assoc. Prof. Dr. Zakiyah M. Noor
(Institut Pendidikan Guru Kampus Tuanku Bainun,
Malaysia)
11. Assoc. Prof. Dr. Lim Yang Teck Kenneth
(National Institute of Education, Singapore)

Organizing Committee Chairpersons

1. Dr.Phil. Ari Widodo, M.Ed.
2. Lili Indarti, M.Hum. (SEAMEO QITEP in Science)
3. Dr. Fitri Khoerunnisa
4. Dr. Lili Hasanah, M.Si.
5. Dr. Rizky Rosjanuardi, M.Si.
6. Dr. Wawan Setiawan, M.Kom.
7. Dr. Siti Sriyati

Secretary

Ika Mustika Sari, S.Pd., M.PFis.

Secretariat Members

1. Eliyawati, M.Pd.
2. Rizwan Darmawan, S.S. (SEAMEO QITEP in Science)

Treasurer

1. Lilit Rusyati, M.Pd.
2. Gumyar Gustiana, S.E. (SEAMED QITEP in

Sections

Program

1. Ikmanda Nugraha, M.Pd.
2. Amalia Yuliana, S.Si. (SEAMEO QITEP in
3. Galuh Yulianti, Ph.D.
4. Rini Solihat, M.Si.
5. Ria Anggraeni, M.T.
6. Dr. Kartika Yulianti, M.Si.

Papers, Publication

- : 1. Dr. Topik Hidayat, M.Si.
2. Dr. Eng. Asep Bayu Dani Nandiyanto, M.Eng.
 3. Al Jupri, Ph.D.
 4. Irma Rahma Suwarma, Ph.D.
 5. Lala Septem Riza, Ph.D.
 6. Dr. Diah Kusumawati. M.Si.
 7. Dr. Endi Suherdi, M.Si.
 8. Dr. Heli Siti Halimatul Munawaroh, M.Si.
 9. Dr. Siti Aisyah, M.Si.
 10. Marthalina Iriani, M.T.
 11. Yudi Wibisono, M.T.
 12. Reza Setiawan, M.T. (SEAMEO QITEP in Science)
 13. Duden Saepuzaman M.Pd., M.Si.

Publik Relation and Documentation

1. Yaya Wihardi, M.Kom.
2. Haidar Helmy, S.P. (SEAMEO Q1TEP in Science)
3. Iyan Sopyan
4. Irfan Taufik, S.Kom.
5. Agus Prihatna

**Accommodation, Equipment,
Decaration and Transportation:**

1. Dian Hendnyana M.Pd.
2. Eki Nugraha, M.Kom.
3. Saepul Anwar, A.Md.
4. Wawan Waryana
5. Yudi ISyuntika
6. Rudi Suyadi, S.Pd.

Food

:

1. Wiwi Siswaningsih, M.Si.
2. Gebi Dwiyantri, M.Si.
3. Tini Kartini, S.Pd.

Supporting

1. Drs. Jajang Saefulloh
2. Nandang , S.SP., M.A.P
3. Drs. Rindang Supriyanto
4. Agus Mawan Giarto, S.Pd.
5. Fitri Ernawati, A.Md

Competencies in Software Engineering of Informatics Graduates at Coordinator of Higher Education Region (Kopertis III) Based on Final Assignment Evaluation

Kursehi Falgenti^{a)} and Danang Sutrisno^{b)}

University of Indraprasta PGRI, Jl. Raya Tengah Kelurahan Gedong, Pasar Rebo, Jakarta Timur, Indonesia.

^{a)}Corresponding author: kursehi_falgenti@unindra.ac.id

^{b)}dnsgs3674@gmail.com

Abstract. The aim of this research is to analyze texts of undergraduate thesis by graduate students of Technical Informatics in regards to know the duration of study, technical skills in software engineering, complexities of application. 200 samples of final assignments as research subjects were evaluated taken from 10 private university libraries at Coordinator of Private Universities (Kopertis) Level Region III. The topic of final assignment analyzed is about application software design. Sample analysis of this research uses open coding method. The first phase of open coding aims at knowing the duration of study and developing topical categories of undergraduate thesis. The second step of open coding is to classify technical skills in process of software engineering; the classification itself is based on Life Cycle Skill Areas in Software Engineering Competency Model (SWECOM) designed by IEEE Computer Society. The third step is to look at complexities of the application. Further, selective coding is done to determine the competency software engineering of undergraduate students at Coordinator of Private Universities Level Region III. Competencies in software engineering can be known through analyzing the duration of the study, technical skills and complexity of application. Result of this study has shown that many graduates of Technical Informatics Program do not complete their study on time; the duration they accomplish their study is beyond than 7-8 semesters. In software engineering, the graduates have good skills in designing process, however the skills to analyze and skills to application testing need more improvement. The applications produced by undergraduate students have low and medium complexities. A small number of applications which have high level of complexities. Based on this research, to improve students' competency in software engineering need to do some improvements: 1) to put big effort of students in software engineering, 2) to improve technical skills of students in the stages of System Development Life Cycle (SDLC). The result of this research can be used as basic information for Study Program of Technical Informatics of private universities, more specifically at Coordinator of Private Higher Education (Kopertis) Level Region III to develop curriculum in the field of software engineering based National Qualification Framework (KKNI) and National Standards for Higher Education (SN-DIKTI).

Keywords: Competency, Software Engineering, final assignment, SWECOM, open coding

INTRODUCTION

The undergraduate thesis as final assignment has to be written in form of a research writing. For undergraduate students, this undergraduate thesis is usually called *skripsi*. Nearly all of universities in Indonesia utilize undergraduate thesis is one of requirements to complete undergraduate degree. Thesis topics taken by students have to be relevant with their areas of study. For students of technical informatics, application software design is one of the frequent topics chosen by students. Students who pick up this topic are required to design and develop an application to overcome a particular problem utilizing certain tools and available methods. Planning steps, analysis steps, design steps and testing software steps are clearly described in the thesis. Students require to defense their research result in front of thesis examination committee members determined by university. The students declared pass through this examination are entitled an undergraduate degree and have owned qualification and competency in the field of informatics engineering. An qualified research is a research that produces results which have valuable benefits for solving people's problems and also this research comprises creative and innovative aspects. Thus, logically, a well written undergraduate thesis come from students who have competency in designing software applications.

Most of private universities in Special Capital Region (DKI) of Jakarta and surrounding areas Jakarta have offered program study of technical informatics. Universities in Jakarta and surrounding areas Jakarta are under coordination of Coordinator of Private Universities (Kopertis) Level Region III. Those universities have graduated a lot of undergraduates degree students in technical informatics. Many of whom have chosen research topic in the areas of application software design. Along with the increasing number of technical informatics students, automatically the quantity of undergraduate thesis written by students also goes up. From the undergraduate thesis, it can be explained the information on how long students spend to complete their study. Further, it can be obtained information on what application created by students and how the process of creating the software.

Through answering the research questions above it can be known technical skills in software engineer of students who have completed technical informatics at Coordinator of Private Universities (Kopertis) Level Region III. Technical skills have important role to understand the competencies in the field software engineer based on Software Engineering Competency Model (SWECOM) created by Computer Society IEEE (2014). Competencies of undergraduate students in software engineering can be seen through the text of their undergraduate thesis. of Text of undergraduate thesis is instrument In general, textual analysis of undergraduate thesis can be essential to know software engineering competencies in three components: a) the duration of study and characteristics of application developed as final assignment, b) process to develop application started from analysis, planning, developing and testing step, c) complexity of application.

LITERATURE STUDY

It is widely known that information technology (IT) disciplines in educational context is distinguished from other fields. One of the noticeable differences is that teaching IT is mostly related to practical skills. Like in software engineering, computer programming is part of curriculum in many universities in Indonesia. Learning computer programming languages cannot be taught through theoretical approach, it takes many practical tasks to develop technical expertise on programming concept and also using algorithms concept in mapping real problems into structure of programming languages (Grossdan Pinkwart, 2015). Some research on that issue have been conducted, Ihantola et al., (2010) and Jenkins et al., (2010) show that many students find difficulties in learning basic concepts of programming. Apiola and Tedre (2012) state that learning programming for students of developing countries has poor ability in mastering programming languages. Therefore, the impact is that they do not have sufficient understanding about complex computer programming. In examination, students usually cheat in doing their examination and plagiarism in completing final assignments is something common to do.

Study Program of Software Engineering

Many universities offer program study of software engineering to improve their reputation. This view is considered as a myth and baseless (Saedian et. al., 2002). Having program study of software engineering can improve the image, however, it is important to figure out the financial aspects spent in the long term cost; a huge amount of budget is required to build and operate computer laboratory.

APTİKÖM (2015) has mapped out the competencies of undergraduate students through categorizing into five different categories of computing disciplines, they are computer engineering, computer science (informatics /CS), software engineering (SE), information system, and information technology (IT). DIKTI also has standardized nomenclature study program under computer science group as follows: (1) computer system (Sistem Komputer), (2) Computer Science/Informatics (Ilmu Komputer/Informatika), (3) Information system (Sistem Informasi), (4) Information Technology (Teknologi Informasi), and (5) Software Engineering (Rekayasa Perangkat Lunak). Information Technology and Software engineering are relatively new disciplines. Software Engineering is developed from Technical Informatics while Information Technology is established from the development of Management Informatics. The name of software engineering study (*Studi Rekayasa Perangkat Lunak*) has not been commonly used at Computer Science for undergraduate level in Indonesia. In fact, there has not been many universities that open those both majors for undergraduate level. Software Engineering Program is still under the domain of Technical Informatics while Information Technology is under Management Informatics.

There are 58 Study Programs of Technical Informatics at Coordinator of Private Universities Region Level III. 30 of the study programs have lecturer ratio which is 1:30. 12 of the universities have the population of Technical Informatics students below or equal to 100 students. 6 universities have the population of Technical Informatics students beyond than 1000 students. While data from BAN-PT there are 53 private universities which have been accredited at Coordinator of Private Universities (Kopertis) Region Level III. From 53 universities, there are four study programs of Technical Informatics which have been accredited A, 15 of Study Program of Technical Informatics which are accredited B and the rest 34 of study program are got C. There four study programs which have not been accredited.

Final Assignment/ Undergraduate Thesis/Skripsi

Completing final assignment which is called undergraduate thesis is evidence of students' academic performance; the final assignment is supposed to be relevant to the field of study taken by students. In Indonesian context, undergraduate thesis is usually called with *skripsi*. Writing undergraduate thesis and presenting it in front of committee members is a requirement to complete undergraduate degree (Wiratha, 2006). The text of the undergraduate thesis written by students is kind of knowledge extraction after students take the course for almost four years. The graduate thesis should be presented in front of the undergraduate thesis committee and students should be able to defend their argumentation. The minimum grade to be declared a pass on computer science is C. Presenting undergraduate thesis is final assessment for students before they complete their study.

Learning Evaluation

The major focus of evaluation is to provide information that can be use as consideration to take a decision (Widoyoko 2009). Evaluation process is extremely essential in learning and teaching process. The process of evaluation or assessment started from daily learning activities in the class until final examination at the end of one semester. This evaluation determine the pass or fail of students on certain course. At the end of academic year universities perform final evaluation to decide students who can be declared a pass and entitled a bachelor's degree. The final evaluation at the end of academic year is through presenting students' undergraduate thesis, however, there are some universities do not require their students to write undergraduate thesis, instead they asked their students to design certain task.

For study program of technical informatics, students are required to choose topic about application software design. In their undergraduate thesis, students have to create particular application to solve a problem in the certain circumstance. In developing their application, students have to use certain tool and method. Following are general information need to be understood in choosing topics in the areas of software engineering:

1. Software platform

Software platforms vary; there are desktop platform for single user. There is a web platform that can serve multiple users. The web platform is run through a web server which processes ordering from clients. Information resulting from the processing step displayed in certain browser. A web-based application can be run as script by processor like ASP or PHP. Script and program will give information in HTML format to webserver and then passed through browser and also to process information from web server, for example saving data to data base. Beside web-based platform, presently mobile platform is growing very fast. The growing of cellular technology has produced smartphone. In smartphone, many applications can be run on the mobile device to enhance productivity or to provide many choices entertainment applications. Cloud platform run on cloud computing. Cloud platform provides services. The existence of cloud platform has changed the paradigm of application program; previously an application program is a traded product, today the program application running on platform cloud can be rent virtually.

2. Software Domain

Software created to a wide range of domains from business application like accounting application, inventory application. Educational applications like information system in enrolling new students, information system for academic purposes. Game applications for educational purposes or entertainment applications.

3. Software Architecture

Initially, software architecture is a stand alone application. The application created for one single user. The growing of networking technology has made architecture application keep abreast of new hardware development.

Through the support of programming languages for distributed system client-server architecture can be developed. The fast growing of the Internet technology enables application developed on internet base. Service Oriented Architecture (SOA) is an architecture standard to develop service-based application run on web service.

COMPETENCIES

Learning evaluation or assessment on the basis of competencies has long been used. In 1970s, the concept of mastery learning and competency-based teacher education were known. In mastery learning, a student declared a pass if he master 8% of learning materials (Yusuf, 2015). Yusuf (2015) defines competencies as integrated skills, manners and knowledge including basic values that should be demonstrated or reflected in daily habit and the ability to think critically. APTIKOM defines competency as accumulated performance of students in doing a number of task description. The task description have certain measurable and structural parameters consisting of independence aspect and perform individual responsibility.

Competencies in Software Engineering

The term programming emerged in 1960s as a job to make computer codes. Along with the growing of computer science, the term programming then has been developed into software engineering. Definition of software engineering is getting broader, it is not limited creating programming codes on the computer, but it is a part of computer science with systematic approach in building and maintain the software (Wirth, 2008).

Software Engineering Competency Model (SWECOM) is a project of IEEE Computer society. SWECOM Ver.1 consists of five components (figure 1). From five components of SWECOM, technical skills is the most important component. Technical skills are divided into two group which are software engineering life cycle skill area and crosscutting skill area. Software engineering life cycle skill area cover skills required to overcome some working activities within stages of developing software. In each of software development step, there are a number of skills that is supposed to be mastered (Table 1).

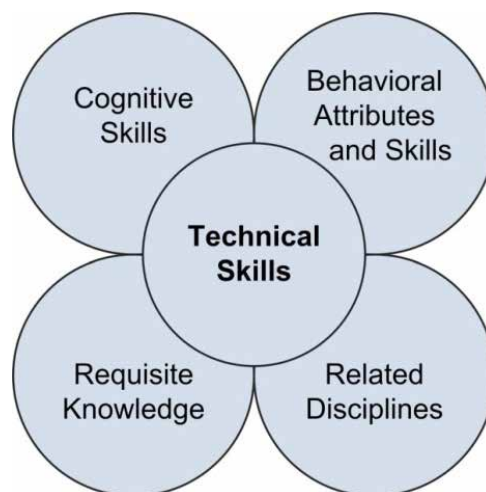


FIGURE 1. SWECOM Components (SWECOM, 2014)

Skills in Software Engineering Life Cycle Skill Area is organized based stages of System Development Life Cycle (SDLC). Steps in SDLC consists of planning, analyzing, designing, developing, testing and taking care of the application. Not all of the SDLC stages can be analyzed through the text of undergraduate thesis. The final stage which is preserving software cannot be found in undergraduate thesis.

TABLE 1. Software Engineering Life Cycle Skill Areas and Skills

LifeCycleSkillAreas	Skills
Software Requirements Skills References: [ACM 2004] [Laplante 2009] [Robertson 2012] [SWEBOK 2014] [Wiegiers 2013]	Software Requirements Elicitation Software Requirements Analysis Software Requirements Specification Software Requirements Verification and Validation Software Requirements Process and Product Management
Software Design Skills References: [IEEE 1016-2009] [IEEE 12207-2008] [IEEE 15528-2008] [SWEBOK 2014]	Software Design Fundamentals Software Design Strategies and Methods Software Architectural Design Software Design Quality Analysis and Evaluation
Software Construction Skills References: [ACM 2004] [Fowler 1999] [Hunt 1999] [McConnell 2004] [SWEBOK 2014]	Software Construction Planning Managing Software Construction Detailed Design and Coding Debugging and Testing Integrating and Collaborating
Software Testing Skills References: [IEEE 730-2002] [IEEE 829-2008] [IEEE 1012-2012] [Myers 2011] [SWEBOK 2014]	Software Test Planning Software Testing Infrastructure Software Testing Techniques Software Testing Measurement and Defect Tracking
Software Sustainment Skills References: [IEEE 12207-2008] [ISO/IEC/IEEE 24765:2010] [IEEE 828-2012] [Lapham 2006] [SWEBOK 2014]	Software Transition Software Support Software Maintenance
Source (SWECOM, 2014)	

Requirement Analysis

Planning, analyzing and designing steps are initial process in developing software in SDLC. At the planning step is more into administratif job, deciding scope and human resources getting involved in developing project. Analysis phase is steps to understand business process, then it is integrated to application that will be created. Analysis to capture user needs is a crucial point in creating application. If need analysis does not cover user needs, it will consequently affect the next stages. If that situation happens, the development of software should go back to initial step until the user needs are fulfilled. Moreover, the effect will also cause additional cost and human resources cost. Skills needed in analysis step is skill to determine user needs of the software. In SWECOM, skills determine software needs. The skills required to fulfill software needs are to extract software needs, to analyze software needs, to determine specifications of software needs. Haroon (2012) explains that strategy advised in doing managerial

process of need analysis is implementable techniques such as brainstorming of need extraction, discussing the project, negotiating the needs, having technical specification to document the needs, evaluating of need documentation to verify and validate the needs. It is also necessary to organize meeting for organizing the needs. In designing step, skills required based on SWECOM is basic knowledge of software engineering, method and software designing strategy, software designing architecture and skill in designing the quality control of the software.

Planning

Application is designed in designing step. Designing is proses of modelling of requirements which have been extracted in the form of abstraction. Abstraction in the format of diagram started from general level until more specific level. The objectives of modelling is to make sure business process integrated to the software can be understood by programmer.

Procedural approach in designing uses data flow diagram, zero diagram and detailed diagram in modelling application design. The emerging of object oriented-programming paradigm, Unified Modelling Language (UML) has been a standard in application designing of oriented-object programming. Diagram used to design application using UML is use case diagram, activity diagram, sequence diagram, class diagram and so on. Service-oriented designing has been adopted along with the emerging of cloud computing technology which provides various types of cloud service started from application, paltform and infrastructure service.

In designing step, skills required based on SWECOM is basic knowledge of software engineering, method and software designing strategy, software designing architecture and skill in designing the quality control of the software.

Development

Software construction is the collection of activities and processes for converting design specifications into functional software solutions that meet customer needs. At this stage, programming codes created in regards to designing of prior steps. Strategy and tool used have an important role at this stage. The selected programming languages and designing method is connected to the application platform and application architecture. Programming languages like Basic and ADA are used in procedural developing. A number of programming languages such as Visual Basic and CC++ can be used to program through procedural approach and object-oriented approach Blansit (2010). While creating programming code using Java is one hundred persen object-oriented approach.

Developing software using traditional approach generally adopts waterfall model. This model consists of six steps: planning, analyzing, designing, implementing, deployment and maintaining. Waterfall model in designing application conducted more structural process in which step by step must be followed rigidly. That is why many consider that this model is too rigid. New technique to organize more flexible in designing software has been developed, for example prototyping method(Floyd 1984). Waterfall model is developed further with incremental way using "coding and editing" approach. This model is well-known with Spiral model (1988)

A lot of applications are widely used in the field of business and trade. The fast growing of Internet makes more flexible in the form of iteratif and incremental development. This concept was started by the emerge of Rapid Application Development (RAD) in 1990s. Agile concept and Agile Manifesto offers more revolusioner development (Schwaber 2004, Holcombe 2008).

Testing

Testing steps in undergraduate thesis examination only black boxtests to ensure all functions in the applicaton run well. Every input will display output without knowing the details. In undergraduate thesis examination, testing of the progam is part of the examination itself. This testing is usually adhoc performed by committe members of the undergraduate thesis examination.

Of five steps in SDLC, there are four steps that can utilized to know technical skills of graduate students of technical informatics. Technical skills can be identified through collecting information from four steps in SWECOM models.

RESEARCH METHOD

This research employs qualitative approach and open coding method. Open coding is used to analyze the text. Open coding is a process to break up the data into unit meanings (Moghaddam, 2006). It is called open coding since at the analysis step the data analyzed through open concept and not focus on specific issues. Coding process is an analysis of textual content through giving labelling concepts for chunk of data, defining and developing a number of categories based property and dimension of qualitative data.

200 of undergraduate thesis samples are taken from 10 university libraries under Coordinator of Higher Education (Kopertis) Region Level III. Open coding conducted over manuscript using three steps; at first step analyze the introductory part of manuscript and the duration to finish undergraduate degree. At the second step, to review technical skills on the process of software engineering based on Life Cycle Skill Areas in Software engineering Competency Model (SWECOM) by IEEE Computer Society. At the third step, to identify the complexities of application base on the number of database table dan the number of actor/identity.

Further, selective coding conducted to determine competencies in software engineering. Straus and Corbin (1990) defines selective coding is "the process by which all categories are unified around a "core" category, and categories that need further explication are filled-in with descriptive detail." (p.14)

Research Object

Locations of this research are in ten universities under Coordinator for Higher Education (Kopertis) Level Region III. The location of universities that under coordinator of Kopertis III are Jakarta, Tangerang and Bekasi (figure 2). Universities that become object of this research are universities which own the most numerous students in taking Technical Informatics based on database in PPDIKTI 2015. The samples of thesis manuscript are taken from University of Esa unggul, USNI, Bina Nusantara University, Muhammadiyah University, Budiluhur University, STMIK Muhammadiyah, STT PLN, University of Indraprasta PGRI and University of Persada YAI.



FIGURE2. Kopertis 3

Sampling Technique

Sampling techniques used in this research is purposive sampling. Data are taken from 10 universities from 54 universities owning study programs of Technical Informatics. The selected universities are the universities which have the most populated students of Technical Informatics based on database of PPDIKTI in 2015. Samples of undergraduate thesis from each universities based on graduation year which is from 2016-2013. Not all university libraries have 2016 undergraduate thesis. Among university libraries which provides 2016 undergraduate thesis are Budi Luhur University, Mercubuana University, Binus University, Muhammadiyah University, university of Esa

Unggul, USNI and STT PLN. In STT PLN, USNI and University of Esa Unggul, the number of undergraduate thesis are very limited. In universities where 2016 undergraduate thesis have not been available (Table 2) due to administration reason their respective study programs. Many samples taken per year based on the quantity of undergraduate thesis on that year, the more undergraduate thesis in one year the more numerous samples are taken.

TABLE 2. Undergraduate Thesis campus per years

Campus/Year	2013	2014	2015	2016
USNI		8	11	1
Binus			11	9
Mercu buana		4	8	8
Esa unggul	5	4	9	2
Muhammadiyah		2	13	5
Budi luhur			1	19
STMIK Muh	14		6	
STT PLN		16	3	1
Unindra	1	12	7	
Univ. Pers Yai		6	14	

Data Analysis Technique

Manuscript sample analysis of undergraduate thesis conducted through open coding approach. There are three steps in conducting open coding. Firstly, it analyze titles, abstracts and approval sheets. At this stage, the open coding aims at determining and classifying topics of application based on software domain, software platform and software architecture. At approval sheet is necessary to know the approval dates of the undergraduates thesis. Based on the approval dates can be noticed how long a student spent to complete undergraduate degree. Secondly, open coding results classification of technical skills in process of software engineering based on Life Skill Areas within Software engineering Competency Model (SWECOM) by IEEE Computer Society. Review and coding step conducted in every stage of System Development Life Cycle (SDLC).

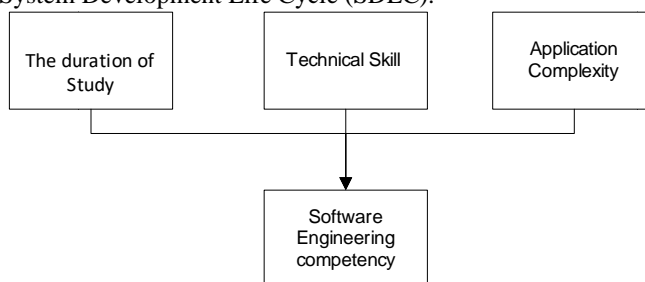


FIGURE 3. Research framework

Thirdly, classification the number of tables in database design, the number of actor dealing with application program to see the complexity of application. Further, selective coding is done to determine the competency software engineering of undergraduate students at Coordinator of Private Universities Level Region III. Competencies in software engineering can be known through analyzing the duration of the study, technical skills and complexity of application (figure 3).

RESULTS AND DISCUSSION

Well-qualified graduates are the ones who are able to complete their study in Technical Informatics on time. To complete undergraduate degree, a student has to take more or less 7 until 8 semesters. Nearly 50% of final assignment finished on time, however, many students do not graduate on time which is around 40% (Figure 4). The remaining graduates which is 10% finish their study earlier. The graduates who pass earlier are the ones who previously hold diploma degree and continue their study to undergraduate level.

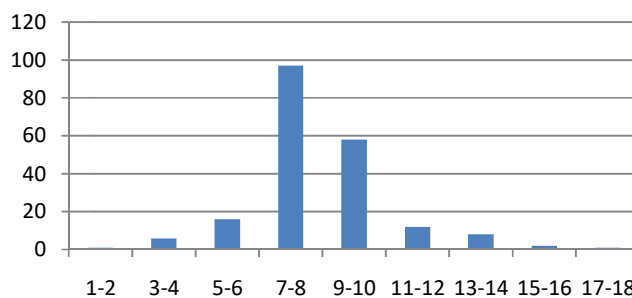
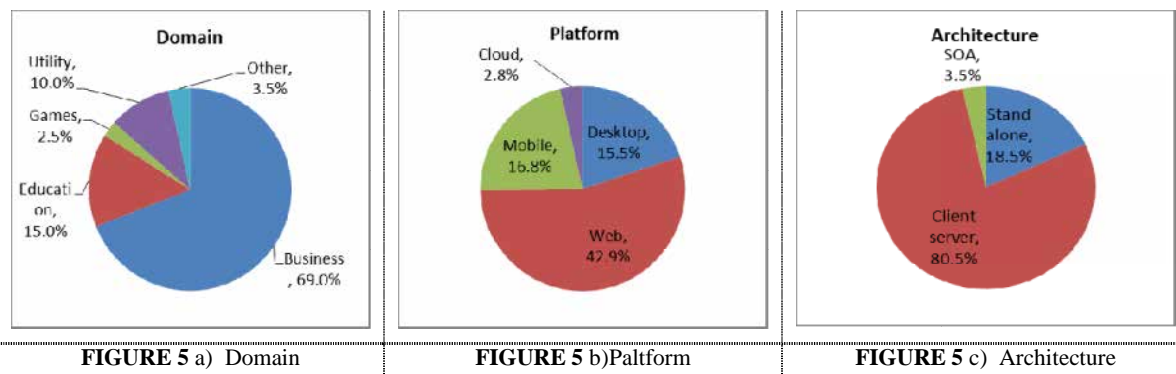


FIGURE 4. Total of Semesters to Complete the Thesis

Application created by graduates of Technical Informatics at Coordinator of Higher Education Region Level III is varied. Application is classified into four domains which are business applications, game applications, educational applications and utility applications and many other apps (Figure 5a). Applications in business is the most numerous application created by graduates of Technical Informatics (69%). Business application made to support business works. Inventory application, purchasing application, production and management information system are quite many created by students. Students also try to develop game application and all games created run on android based smartphone. Not all universities allow their students to create game application. Game application as research topic is merely found in two universities at Coordinator of Higher Education Region Level III. Other applications that do not include in the classification are related to the field of government and social matter.



Web platform and mobile platform is the most platform choose by the graduates of Technical Informatics. Some of the applications use two platforms; a web platform as server provides service while clients use mobile platform. Application with mobile platform varies, besides game application, there are some application for utilities and information and this application is usually utilized GPS, like small public minibuses transportation system (*angkot*) and tourism location system (Figure 5c).

Majority of graduates of Technical Informatics use client-server architecture. Client-server architecture is used to develop web application. While SOA architecture to develop service based application has not gained interest (Figure 5c).

The discussion above related to general knowledge in the field of software designing that has been done by graduates of Technical Informatics at Kopertis III. Further discussion is on technical skills of graduates based on SWECOM. The discussion begins with analysis step in SDLC.

At the analysis step, one of skills that is required to own by software engineer according to SWECOM is the skill to extract software needs. Skills are related to techniques in engineering requirement. Extraction of application needs for particular purpose is more suitable use interview and observation method while need extraction on applications that will be sold to public is more suitable use questionnaire. In the graduate thesis, a vast majority of students uses interview method and observation, meaning that the product of applications are created to fulfill certain purposes. Few graduates who use questionnaire (Figure 6 a)

Graduate skills in extracting software needs can be known from the transcript of the interview, unfortunately students rarely attach the transcript of the interview at the appendix part of their graduate thesis. Extraction towards software needs is much easier if it is obtained from the right key users. In doing extraction of software needs, many of graduates put the leaders of companies as key users. In fact, students should use head of divisions which is more knowledgeable about the process of their business. Take for an example, for warehouse application, the right key user as source of interview is head of warehouse not director of the company. If students make mistake in choosing the key user, it will be too risky in analyzing and extracting software needs.

The more users get involved in extracting the work needs, the easier to capture business process and the easier to integrate the work needs into the application. In undergraduate thesis of graduates of Technical Informatics at

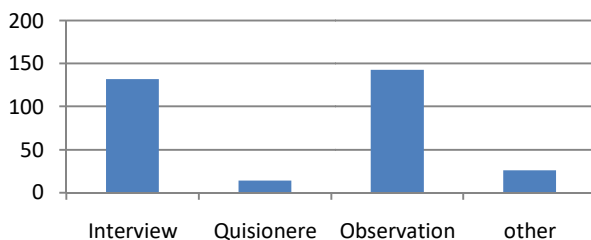


FIGURE 6 a) Requirement engineering method

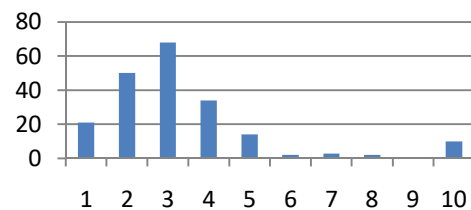


FIGURE 6 b) Total of Key users

Kopertis III, many of whom interview two or three key users for the applicaton with medium complexities. It would be better if those graduates get involved more key users (figure 6 b). The possibility of failure in analyzing and extracting software needs can be occurred if key user is not sufficient enough.

The next step is to see technical skills in designing software of Technical Informatics graduates. Skills required according to SWECOM are fundamental skills in designing, skills to choose strategies and method in designing software and skills in developing software design architecture. In this research, graduates have used a wide range of modelling (figure 7 a), but many of whom do not use the proper modelling strategies. Modelling with UML can be used for object-oriented software. In contrast, modelling with structured diagram should be used to design process-oriented software. (figure 7 b). The graduates has been able to develop software design architecture like process based design; the graduates are able to develop structure diagram start from contextual diagram, zero diagram and detailed diagram. Design architecture of diagrams in UML model can explain the software needs in detail.

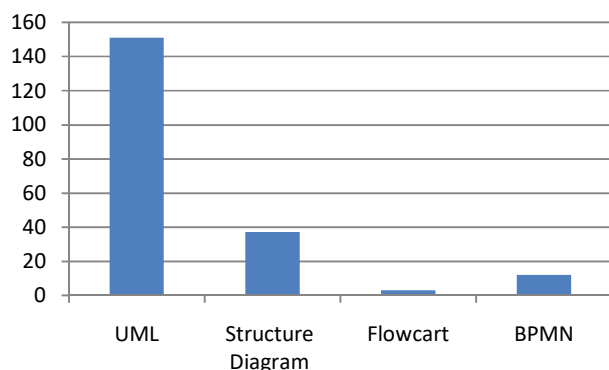


FIGURE 7 a) Modelling Tool

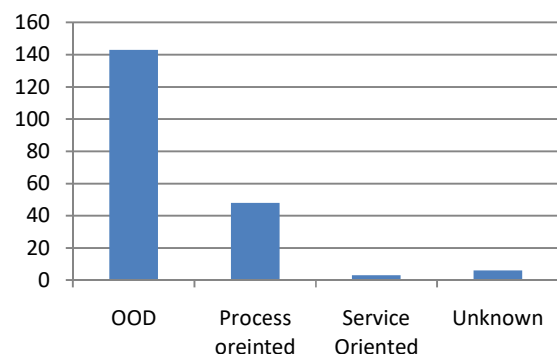


FIGURE 7 b) Design Method

The next skill is skill in designing software. This skill is related to technical skills in planning of software design. The planning includes planning to choose the tool like programming languages (8 a) and planning to choose method used (figure 8 b). At planning stage for choosing the tools, graduates are not able to determine the correct tool. Many of the undergraduate thesis explore desktop application design but they use Java for programming language. Java is made to develop distributed application in order to create application in networking system. Therefore, using Java to develop desktop application is not the right decision.

SWECOM does not explain competencies in regards to using tools because this is more into specific projects. However, to see skills in programming languages, graduates need to know types of programming languages used for application coding. It is broadly known that programming languages used are PHP and Java. The latest programming languages like Python has not been used. Most of the graduates use Waterfall method in developing software, while a few graduates that use agile method; there are only two universities use agile method or more precisely is called Scrum method.

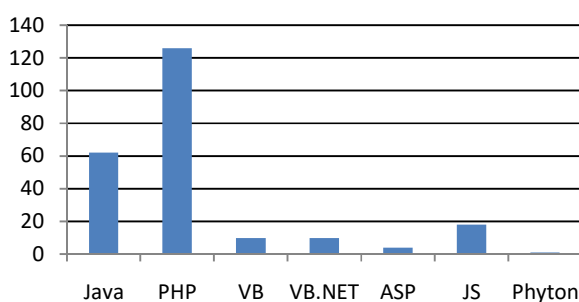


FIGURE 8 a) Program Language

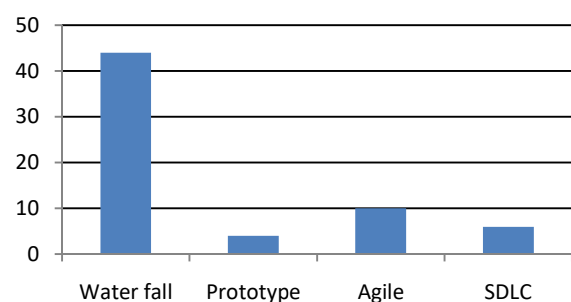


FIGURE 8 b) Development Method

Technical skills required to perform software testing according to SWECOM are skills to design test planning, skills to organize test infrastructure in form of tool and skill in operating software testing technique. Skills to test application is to assess the display and black box (figure 9). Generally, graduates merely perform display testing. They capture interface and explain functions related to the interface. Data on skills related to design planning test is not found. Also, data for tool testing is not found in the undergraduate thesis.

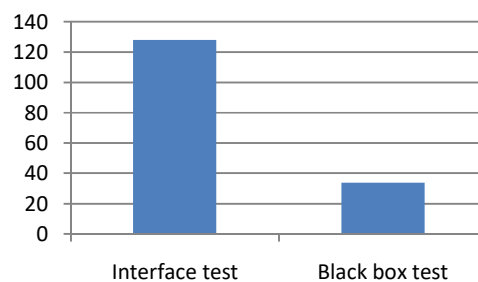


FIGURE 9 Application Test Technique

At the third step, open coding technique conducted to see complexities of application developed by graduates of Technical Informatics. Application complexities can be assessed through recognizing some parameters in building program. Some of the parameters are the number of tables in database design (figure 10 a), number of actor or entity dealing with designing (figure 10 b), Number of tables in database design varies and average the number of tables created by students is 6-7 tables. This research determine that application developed with the number of tables less than 10 is categorized as non complex application, while application with more than 20 tables is categorized as having high complexities. In this research, we do not find many application with tables more than 20.

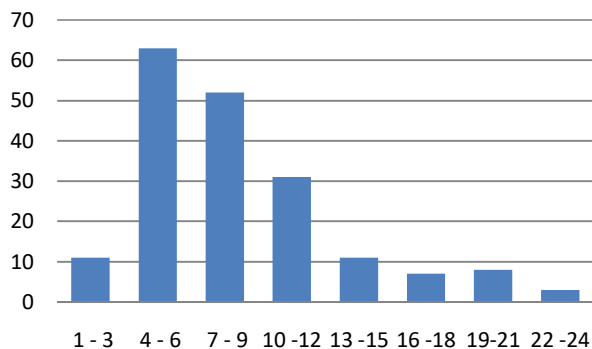


FIGURE 10 a) The Number of Database Table

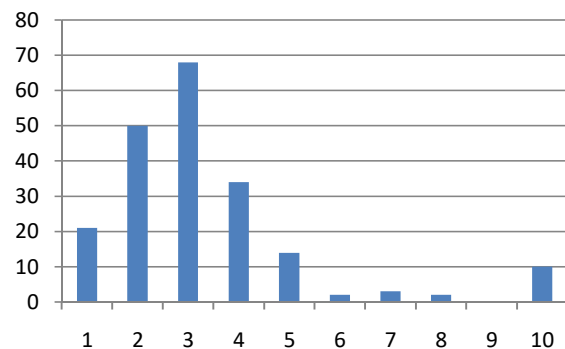


FIGURE 10 b) The Number of Actors/Entity

The more number of stakeholder get involved with the application, the more complex of application. Number of actor or entity in application design found in undergraduate thesis is not too many, 3 actor/entities in average. In mobile application, generally the quantity of actor is only two persons, user and administrator. Therefore, the quantity of actor cannot be used to know the complexities of mobile applications.

CONCLUSIONS

From this research we find that many final assignments with the topic of software engineering conducted undergraduate students of Technical Informatics do not finish their assignment on time, generally speaking they extend the tenure of their course to complete the final assignment. Based on the duration needed to complete the study can be concluded that the graduates are lack of skills in developing application program using tool and metode that they have learnt. Better practicum materials, more exercise and more intensive program can improve programming skills of Technical Informatics students.

Graduates have well-qualified competencies in application design, especially in design method and design architecture, however at level of analysis is necessary to improve their competency in choosing key user to extract software needs. At the stage of developing and testing, skills of the graduates need to be improved. In developing of planning of tools used, students should consider to use proper programming languages. Students are in need of knowledge about new method like Scrum to develop application. Through Scrum, students are able to experience working together in a team

There have not been many graduates who create complex application. To develop complex application many human resources are needed while at this one undergraduate thesis done by one student. It is necessary to have new innovation in doing final assignments, let say for example, Binus University that encourages their students to work in a team to finish complex application. Therefore, level of complexities can be improved and of course produces more useful applications.

Mobile applications is one of the software engineering that attract students' interest. And universities need to response this phenomena through designing teaching curriculum of software engineering for mobile applications. In the curriculum of software engineering, it is important to put big effort to improve programming techniques of mobile application and to enhance curriculum development in improving application design of mobile application. This kind of development will help government to support the grow of smartphone industry. Government has put software as a part of TKDN (The Level of local content). This new regulation, known as *Tingkat Kandungan Dalam Negeri or (TKDN)* should be fulfilled if companies want to market their 4G product in Indonesia. Therefore, to improve TKDN, stakeholders in Smartphone 4G should include local contents in their products. This policy gives tremendous opportunity for those who are dealing with mobile applications.

REFERENCES

1. A. Haron, M. Harun, M. N. Mahrim, S. Sahibuddin, N. H. Zakaria, and N. A. Rahman, "Understanding the requirement engineering for Organization:The Challenges", ICMME 8th International Conference, IEEE Explore pp. 561-567, 2012
2. A. M. Yusuf, Asesmen dan Evaluasi Pendidikan, Prenada Media, Jakarta (2015)
3. Bidang KKNI APTIKOM, Naskah Akademik Kerangka Kualifikasi Nasional Indonesia (KKNI) Rumpun Ilmu Informatika dan Komputer, APTIKOM Indonesia, 2014
4. A. Strauss & J. Corbin, Basics of qualitative research: Grounded theory procedures and techniques. Newbury Park, CA: Sage (1990).
5. B. D. Blansit, Object Oriented Programming: What is IT Talking About?, *Journal of Electronic Resources in Medical Libraries*, 7:1, 90-97, DOI:10.1080/15424060903585792(2010)
6. B. G. Glaser, and A. L. Strauss, , The Discovery of Grounded Theory: Strategies for Qualitative Research, Aldine, pp 271. (1967)
7. Floyd, C., "A Systematic Look at Prototyping," Approaches to Prototyping (Budde R., Kuhlenkamp K., Mathiassen L., and Zullighoven H., eds.), Springer Verlag, pp. 1-18. (1984)
8. H. Saiedian; D. J. Bagert; N. R. Mead, Software Engineering Programs: Dispelling the Myths and Misconceptions *IEEE software* (5) (2006)
9. I.M. Wirartha, Pedoman penulisan usulan penelitian, skripsi, dan tesis. Yogyakarta: Andi Offset, (2006)
10. K. M. Eisenhardt, Building Theories from Case Study Research, *Academy of Management Review*, vol. 14, no. 4, pp 532-550 (1989)
11. M. Apiola dan M. Tedre, New perspectives on the pedagogy of programming in a developing country context, *Computer Science Education*, Vol. 22(13) (2012)
12. N. Wirth, A Brief History of Software Engineering, *IEEE Annals of the History of Computing* (2006)
13. Okezone.com 9 Prodi Terfavorit di SBMPTN 2014 <http://news.okezone.com/read/2014/06/10/373/996894/9-prodi-terfavorit-di-sbmptn-2014> (2015)
14. P. Beynon-Davies, Information systems: An introduction to informatics in organisations. Palgrave Macmillan, (2002).
15. P. Ihanntola, T. Ahoniemi, V. Karavirta, and O. Seppälä, Review of recent systems for automatic assessment of programming assignments. In Proceedings of the 10th Koli Calling International Conference on Computing Education Research (pp. 86-93). ACM.
16. P. E. Widoyoko, Evaluasi Program pembelajaran, Pustaka Pelajar Yogyakarta (2009)
17. S. Gross, dan N. Pinkwart, Towards an Integrative Learning Environment for Java Programming. IEEE 15th International Conference on Advanced Learning Technologies 25-28(2015)

CERTIFICATE

This is to certify that

Kursehi Falgenti

has participated as

Presenter

in the

International Seminar on Mathematics, Science, and Computer Science Education

Theme:

“Harnessing Local Wisdom to Build Competencies of Excellence in Research
and Collaboration in the New Era of the ASEAN Economic Community”

Mathematics and Science Education Faculty
Universitas Pendidikan Indonesia

in Collaboration with

SEAMEO Regional Centre for QITEP In Science

Bandung

October 15th, 2016



Siti Fatimah, S.Pd., M.Si., Ph.D.
Dean of Mathematics and Science Education Faculty
Universitas Pendidikan Indonesia



Prof. Dr. Triyanta
Director

SEAMEO Regional Centre for QITEP in Science