

A System for Supporting Project-Based Exercise in Software Development with Facilities to Detect Students Automatically who Are Required Individual Tutoring

HIROAKI HASHIURA[†], KOTARO YAMASHITA[†], TATSUYA ISHIKAWA[†],
YUKA ISOZAKI^{*}, RYOTA CHIBA[†], YUICHI INOUE[†], SEIICHI KOMIYA[†]
Graduate School of Engineering[†]

Department of Information Science and Engineering^{*}

Shibaura Institute of Technology

Komiya Lab, 13F, Kenkyu-to, 3-7-5 Toyosu, Koto-ku, Tokyo

JAPAN

{m705107, m105109, m104010 101040, m107059, m107012, skomiya}@shibaura-it.ac.jp

<http://www.komiya.ise.shibaura-it.ac.jp/>

Abstract: - In recent years, large-scale-sizing and complication of software development are being followed wholeheartedly. On top of that fast delivery and low cost developments are being required as compared to up to now, and the environments, which surround software development are becoming more severe. For this reason, the industry is demanding high talented people with potentials of practical senses from the universities. For this reason, Shibaura Institute of Technology Information Engineering Department, in order to master the knowledge and technology for software development, is carrying out the group exercise class of the software development, which adopts a style more close to actual exercise. The group exercise can lower the burden of the teaching person regarding the instructions to students than the instructions for individual exercise, but it is difficult for giving instructions matching with every student's capacity. In addition, since the school hours assigned to software engineering are insufficient, it becomes important to prepare the environments where a student can exercise also out of school hours. Moreover, the goal of a exercise class is to acquire the knowledge and skill for software development on the individual level of a student. Therefore the authors having multi functions that were necessary for support to the group exercise and a high possibility to be included in the group exercise, developed group exercise supporting circumstances EtUDE[1] (Environment for Ultimate software Development Exercise) of the software development with function to automatically detect the student requiring individual instructions.

Key-Words: - Software Development Exercises, Software Development Environment,
Project-Based Exercise

1 Introduction

In recent years, large-scale-sizing and complication of the software development currently performed in the actual world are being enhanced by a technical advancement or diversification of the system forms. On top of that fast delivery and low cost developments are being required as compared to up to now, and the environments, which surround software development are becoming more severe. Training of the talented people who have more practical knowledge and the technology about software development with such a social background is important. However, in the companies belonging to the information industry, the pressing need of the training of such talented people is fully recognized. While busy with work in hand, the new comers are put into actual business exercise without enough technical training, or

workers are recruited from subcontractors, and employed in the software development project.

It is true for subcontractors also, the personnel are put into actual business exercise without enough training, and still the insufficient necessary personnel are provided by sub-companies, is the actual exercise.

That is, in the actual condition, the software skill person is studying technology not through corporate education but through actual exercise. With such situations, the companies are demanding the universities for the high talented people with potentials of practical senses. By the way, the form of organizing a project team and developing software is common in the actual world. So, it is not rare to liken the exercise of the software development performed by being divided into some groups in a exercise class etc. with a small project, and to make participants do a virtual experience for

a practical software development in it in educational facilities, such as universities. There, while participants utilize the knowledge of the software engineering acquired at the class etc., the practical capacity about a software development is supported by solving various problems about a software development actively, and on going.

Shibaura Institute of Technology (hereafter described as the Institute), in the department of information technology, in order to master the knowledge and technology of a software development, the practical software-development exercise (it is hereafter described as the exercise) which third graders perform by being divided into some project teams who consist of several persons is carried out from the 2003 fiscal year. The group exercise can lower the burden of the teaching person regarding the instructions to students than the instructions for individual exercise, since instruction are given per group, a group exercise is difficult for giving instructions united with every student's capacity.

In order to solve such a problem, the authors having multi functions that were necessary for support to the group exercise and a high possibility to be included in the group exercise, developed group exercise supporting circumstances EtUDE (Environment for Ultimate software Development Exercise) of the software development with function to automatically detect the student requiring individual instructions. This paper describes the system outlines of EtUDE, and the results of having performed the system.

Finally, the composition of this paper is shown below. In Chapter 2, the exercise states the target outline and aim of PBL. Chapter 3 describes related research of this paper. In Chapter 4, while clarifying the functions for which EtUDE was developed in order to solve the problems, which became clear in Chapter 3, it is shown clearly what kind of functions were offered. In Chapter 5, in order to realize such functions, what kind of architecture is there is clarified. In Chapter 6, the built system EtUDE is applied to actual practice business, and estimates the effectiveness of each offered function, and the effectiveness of the practice business, which introduced EtUDE. Chapter 7 describes the conclusion of this paper.

2 Background

The software-development exercise assumes that as limits of the properties of a subject, a participant has already acquired the knowledge and technology for programming and can use them to some extent

as a prerequisite. However, it is not easy to master programming technology and software engineering in four years of a university, although he is called a participant with majors in information technology.

Emphasis is put on the following portions of the software development exercise by the target group of this paper, although emphasis is only put on acquisition of programming technology by the programming exercise performed individually[2].

- It is necessary to perform a role assignment.
- Members need to communicate with each other.
- Develop software with a comparatively large scale.

These differences originate in the part whether a exercise has the side as a project.

DeMarco et al. [3] state that a software development project is high intellectual work, and many of the problems are more sociological than technical. This means that a project cannot be successful only with the members of technical skill.

However, a participant's expects as a first step in a software development exercise, learning the skills about a software development in many cases. This is considered to be a thing based on the way of thinking that technical skills, such as programming and modeling, are of primary importance as a software development. Thus, the participants do not recognize mostly which skills other than technical skill have grasped the success or failure of the software development.

For this reason, by making the participant learn the technical skills, the teaching person needs to point out existence of skills other than technical skills, and needs to highlight the importance of skills other than the technical skills in a software development project. Lapin et al. [4] have also pointed out the importance of such a curriculum.

In order to perform an exercise with such an element, it is necessary to take in the element of the software development as a current project not only in practice of programming exercise but in business. Such a software development exercise is called PBL (Project Based Learning), and is taken in by the software education of many universities by its importance.

It is indicated by Matsuura [5] that it needs to be based on the following points in carrying out PBL.

- It is necessary to teach efficiently as the prerequisite knowledge for Software Development to some extent until the latter half of 3rd grade.

- When a student sets up a subject himself, although it is interesting, realization is difficult, and it becomes a very easy setup in many cases, and there are sometimes a few failures or senses of accomplishment.
- Although PBL supports the power of performing a problem solving actively, utilizing the knowledge acquired at the lecture, in order to act as a group, an individual scholastic evaluation is difficult.
- Although it is dependent on the studying people number, it is necessary to prepare enough the support system by the tools, which support activity for every group, and teacher and TA.

Moreover, it is said that how Hayes et al. [6], Martin-Garcia et al. [7], evaluate the contribution, which the participant individual in a team performed to the team poses a big problem in a exercise. Since these problems exist, in PBL, it is difficult to build organization sufficiently by the small number of staff.

Moreover, in order to learn such skills, a certain amount of time is needed, but the school hours run short for this. In order to correspond to shortage of such school hours, it will work using the time besides school hours, but the members, which belong to the same group may not necessarily gather always at same place, or work at the same time, out of school hours. Furthermore, even if two or more members work separately, finally you have to make one product together. For this reason, without the restriction of time and place for each student, it is necessary to prepare the system, which supports communication between team members. Simultaneously, in order to support professors of a small number of people, it is also necessary for instructions to identify a required student preferentially.

In order to solve such a problem, the authors developed the system EtUDE (Environment for Ultimate software Development Exercise), which supports the group exercise of a software development, and it proposes raising the quality of a exercise by introducing into business.

2 Related Work

Since frequent communication of between a student and a teaching person is required unlike the usual lecture etc., a software-development group exercise has a large burden on the teaching person. For this reason, many systems, which support the exercise, have been proposed, so far.

(1) TeamSCOPE

Jang et al. [8] propose a system called TeamSCOPE, composing team members from all over the worlds, such as Asia, the United States, and Europe, and is conducting the experiment of performing the software development in offshore places.

Since the software development members, which are present in a remote place runs short of awareness, there are problems of real time communication and schedule adjustment of meetings are difficult, or gap arises in the product made in a mutual development phase or its interpretation. Then, Jang et al. implemented the function for compensating the shortage of awareness in TeamSCOPE. It evaluates what kind of influence compensating the shortage of awareness would have on a software development, and the importance of compensating the shortage of awareness is described.

(2) Waltz

Chiken et al. [9] point out that not only the communication between students in the group exercise of a software development, but the communication between students and a teaching person is also an important thing, and in addition to communication within a group, have developed the system Waltz with the communication support function between a teaching person and students. In addition, in the exercise lesson by Waltz, the teaching person is doing the inspection and the acceptance test for the student's product for progress evaluation in checkpoints, or in student's scholastic evaluation, and Waltz has a function, which supports these works Chiken et al. show clearly that the communication support function between students and a teaching person is utilized in the case of an inspection or the work of a test.

(3) GWSS

Matsuura et al. [5, 10-13] point out that the process of the software development in exercise are divided into four steps, requirements analysis phase, system-analysis phase, system design phase, and operation / test phase, and in the practice creating a deliverables with awareness of the purpose of these phases. It is said that it is important to make students study to carry out a software development in steps.

The guideline of work is given to each process so that participants can be conscious of a phase. The guideline is fundamentally made based on object-oriented development, and makes it the task to submit a deliverable and the last product after the

end of each phase. Matsuura et al. have prepared the template of a report for the system so that the scholarly students can create a deliverable and the last product smoothly.

Since the environment where these systems are applied differs, each system is equipped with the function required for the support for a group exercise although the support functions with which the system is equipped differ. However, since these systems are designed with emphasis on the side of a group exercise, they cannot identify a student to be taught preferentially. For this reason, the possibility that a student needs to be followed up individually will be lost in a group is high. Authors have made the exercise support environment equipped with the function which solves such a problem.

3 EtUDE

As described in Chapter 2, this system becomes important so that it can be used without each student having restrictions of time or a place. For this reason, EtUDE consists of group of tools for offering support, which cannot respond in a Web application system (EtUDE /Core) and a Web site application.

The image of the exercise using EtUDE is shown in Fig.1.

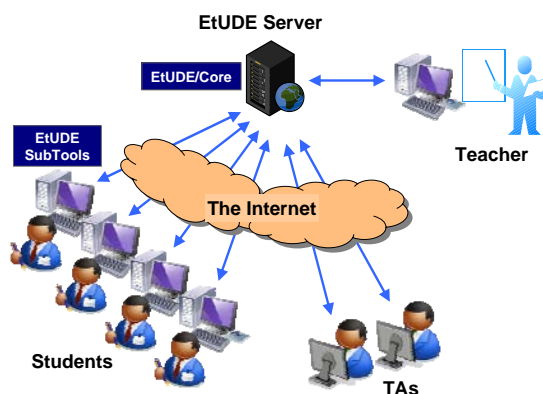


Fig.1: EtUDE System Overview

EtUDE/Core is working on an EtUDE server for 24 hours, and if the Internet connection environment are available, it can be used freely anytime. The design and the operational supportive tools are installed in the participant's terminal, and these tools can be used according to a process. Moreover, since the usage situation of the product of a exercise or a system is saved on a server, the teaching person and TA can support study of a participant, by referring to them.

Moreover, the functional outline of EtUDE is shown in Fig.2. Details of the functions are given below.

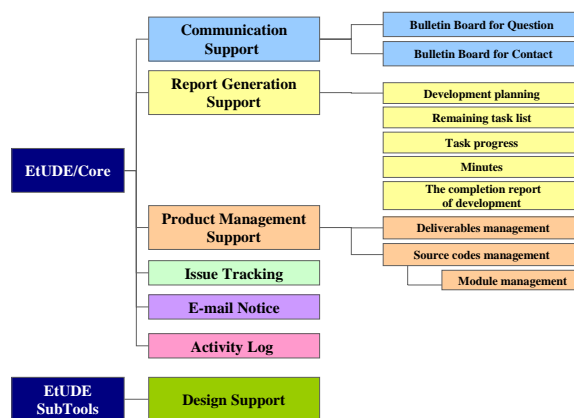


Fig.2: EtUDE Functional Outline

3.1 The functions of EtUDE/Core

3.1.1 Communication Support Function

The communication performed during the exercise session from the result which Hazeyama et al. [14] observed during the exercise lesson say that it became clear that there are three kinds with that objective and each of those can be classified as follows from the basis of communication between who and who.

(1) A question and its reply.

It is carried out in order to fill up the technically unknown points, and there are three kinds, communication within a group, communication with a teaching person and a student individually, and communication with a teaching person and a group.

(2) A proposal and its exchange of opinions.

It is the communication within a group.

(3) The contact and the check from the teaching person side.

It is the communication between a teaching person, a student individually, or a group.

It is the communication between a teaching person, a student individually, or a group.

In EtUDE/Core, a group bulletin board and two kinds of bulletin boards of questions / contacts which all participants use were implemented based on these three forms of communication.

Both of the bulletin boards have the advantage of being easy to hold the flow of the talk, even if a thread type bulletin board exchanges a comment in

the related subject unit called a thread and many topics are contributed to the bulletin board.

Furthermore, it is made easy to check visually the reply to which comment a certain comment is by indicating a thread tree form.



Fig.3: Top Screen of the Bulletin Board for Question / Contact

3.1.2 Report Generation Support Functions

In a software development, since almost all work is done in a worker's mind, it is the type where there are many things where the results of work are not seen visually. With such a feature is hard to grasp the progress situation, and there is the problem that it is not known how far is the plan behind the schedule. In order to lead a software development to a success, it is important to grasp the progress situation in a group as correctly as possible.

Since much work is required within the short exercise time, in the target exercise of this paper it is very difficult for completing work only in exercise time. Therefore, it is requested that participants should do much more work other than the exercise time.

However, since an exercise is 1 time per week in principle, while the participants are working out of practice time, a teaching person is unable to observe and get to know the progress situation of each group. Moreover, even if a teaching person works for grasp of progress during exercise time, it is also impossible to grasp the progress situation of all the groups correctly only within exercise time. For this reason, it will be necessary to grasp a participant's work situation by a certain method. For such problems, it is assumed that the authors should summarize the progress situation of a exercise within a group, and report to the teaching person periodically and if a problem exists should get more prompt support from the teaching person.

Based on this thinking, the support function of designing reports is made to be a practical form, which is close to IT engineer's everyday business.

However, for the situation where the participant are beginners of a software project "don't know what to report", the reporting items are made into a template so that beginners could also report. Therefore, participants are only required to fill the items in the prepared template, and can report now the information about the progress according to the state of affairs of the project. Moreover, the information is managed in a unified manner on the web, and is enabled to share the progress for making an inspection possible within a teaching person or a team. When it evaluates each attendance student, the data of such progress is useful to grasp the work, which each participant took charge of, and quantity of work, and also in order to judge how much the participant individual contributed to the group, it can be utilized.

By the exercise, the practice was divided into the following six phases and five kinds of reports were set in each.

- Development planning
- Remaining task list
- Progress management
- Minutes
- The completion report of development

3.1.3 Product Management Functions

In a software development, various works, such as planning, analysis, design, coding, and test, are done. The deliverables according to the work of minutes of the meeting, design specifications, source codes, test case tables, etc. are created in the process of these work.

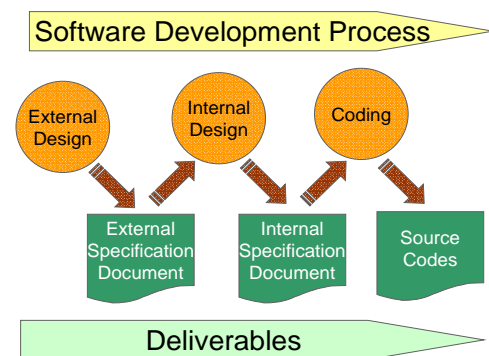


Fig.4: Software Development Process and Deliverables

These deliverables are referred to for execution of other work, and the following deliverables are generated. Thus, the deliverables are created one after another. Therefore, it is very important for a

software-development project to manage these deliverables appropriately and to enable them to share within a group. The relations between a development process and deliverables are shown in Fig.4.

Moreover, Matsuura et al. say that it is important that the contents of the product which the teaching person needs can be checked when grasping the progress situation of study or giving the suitable advice for the contents of study.

It is thought that support of version management (revision management) of a product is also required for the product containing a source code since revision of the contents is often performed with progress of development. Therefore, each function is equipped with the version management function. For this reason, all the products can take out the arbitrary files of not only the latest version but also the past.

In the exercise, since the group members are more than 20, it becomes a work burden very serious for the teaching person to manage all the products. Therefore, the product was managed in a unified manner electronically, the setup, which can be referred to any time from any place, is considered as necessity, so the product management function was developed. Further, the product management function consists of the following three functions.

(1) Deliverables management

A Deliverables management is a function to upload and management deliverables other than source codes.

(2) Module management

Since a source code has many files compared with other products and the frequency of changes is also high, to manage is a very difficult product. For this reason, the source code was made easy to manage by developing a module control function.

(3) Source codes management

In order to be able to version manage and to be able to manage a source code in a unified manner, the "source codes control function" was developed. A source code control function is for uploading a source code to the individual module created by the module controlling function.

3.1.4 Issue Tracking Function

If a test suitable after coding is done, it is sure that many bugs will be detected. In the limited time, in order to cope with bugs, the list of the bugs until that point of time is created, and debugging in order with a high priority is the basis. In that case, the role of a issue tracking function (bug tracking function) manages the information that takes charge of which

bug, and how much it has debugged, and supports debugging indirectly.

The bug management models, such as Bugzilla [15] which is a bug tracking tool of an open source, about the correction flow of the bug using a Issue Tracking Function, is referred.

3.1.5 E-mail Notice Function

Since EtUDE/Core is built by the Web site application system, if there is a PC connected to the Internet also out of exercise time, it can be worked. Conversely, if the situation of a project must be periodically checked with PC also out of lecture time, it can be said that such work will serve as a burden remarkable for a participant. In order to cope with such a problem, when the state of a project changes, EtUDE/Core is made with the function of notifying by e-mail. The mail address, which a participant wishes, can be set up in advance.

The timing of notifying the e-mail are as follows.

- When the message goes to school bulletin board
- At the time of upload of a product

3.1.6 Activity Log Acquisition Function

All the activity to all the users' system is recorded on EtUDE. Thereby, the use situation of a system etc. can be grasped. The function which records the response time of a system in addition to this is built, and it enables to check whether the system is operating in the range, which does not cause processing speed stress to the user.

3.2 EtUDE Sub Tools

The function of EtUDE described so far was the structure for supporting a development process over the whole. In order to raise the quality of a exercise further, in each process of a development process, the support corresponding to the problem according to each individual process is needed.

3.2.1 Design Supportive Tool

A class design is difficult work especially for beginners. It can be said that the cause is because the deliverables, which serves as an input of a class design is a requirement specification document of requirements analysis process. The requirement specification document was not formalized like the class diagram, and in almost all cases since it consists of the texts and charts, which were created by natural language, ambiguity is included in itself. That is, in order for a designer to perform a class design, it is necessary to read and solve a

requirement specification document, understanding the intentions of an order person and analyst. Moreover, considering the implement ability of a system, the designer needs to discover a class and needs to give the suitable name for it, it is necessary to set up relation and a multiplicity so that neither uselessness nor inconsistency may arise in structure, and the many matters must be taken into consideration. The property of such a class design also has big influence on acquisition of the skill of a design.

For this reason, a design supportive tool is introduced into the client side apart from EtUDE/Core, and is enabled to point out the errors of a design at an early stage by the exercise. The use image of this tool is shown in Fig.1.

4 Construction of System

4.1 Basic Principles of Systems Configuration

EtUDE/Core is assigned to exercise every year, and as an object of research it is always advancing. Therefore, it is required to change or add the function of EtUDE/Core repeatedly. Then, it becomes important to realize the system base which can respond to such changes.

In building the system base of EtUDE/Core, in order to attain the above-mentioned purposes, the following five statements of principles are set up.

- It is easy to reuse.
- It is easy to carry out maintenance.
- It is suitable for development by multi people.
- It is easy to do testing.
- It is easy to carry out extraction of a log.

In order to realize these statements of principles, four following exercises were introduced.

(1) Implementation by POJO (Plain old Java Objects)

POJO is what was produced as a comparison language with a heavy weight class, which needs succession of a specific class and is the general term of the class where it is not forced for the implementation of the specific interface and the succession of the class on the realization of the application. In EtUDE/Core, all the components are configured with POJO. With the advantage of lightweight implementation, the dependency between components can be lowered, and being able to adjust the system to new architecture easily.

(2) Implementation of Polymorphism interface

Martin said that in the good object oriented design, the implementation depends on abstraction and the implementation must not depend on implementation, and the importance of the Polymorphism using an interface is pointed out. By implementation, which utilized the interface, it becomes possible to operate dependencies from the outside, and the mock testing can be done easily.

(3) Separation and the modularization of the cross-sectional subject of concern

By separating the common functions crossing to every module, modular intensity can be made high. AOP [16] (Aspect Oriented Programming) is applied in as a mechanism for realizing this. AOP divides the function of a system into a core function (core concern) peculiar to each class, and the concerns (cross cutting concern) across boundaries, which appear repeatedly in two or more classes across boundaries at the execution time. It is a mechanism, which weaves concerns across boundaries into a core function (weaving).

In EtUDE/Core, the following functions are concerns across boundaries.

- Log
- Transaction processing
- Exception handling
- Security

(4) Practical use of a Design Patterns

Design Patterns [17] is what reorganize a typical problem happening at the time of an object-oriented program design by Gamma et al. and its solution, and consolidate for reuse. It is not necessary to think over the same solution to the same problem from the beginning repeatedly, and effects, such as shortening of a development period, are brought about as a result by using a design pattern. Moreover, since understandings common to design with surrounding people are obtained, it becomes possible to perform a design within a project smoothly.

4.2 System Architecture

Programming language of the system development was made as Java. It is because Java has a structure with great track record in the development field of Web application, and makes easy construction and employment maintenance of abundant class libraries, component sized technology like Java Beans, various frameworks, etc. The architecture of this system is shown in Fig.5.

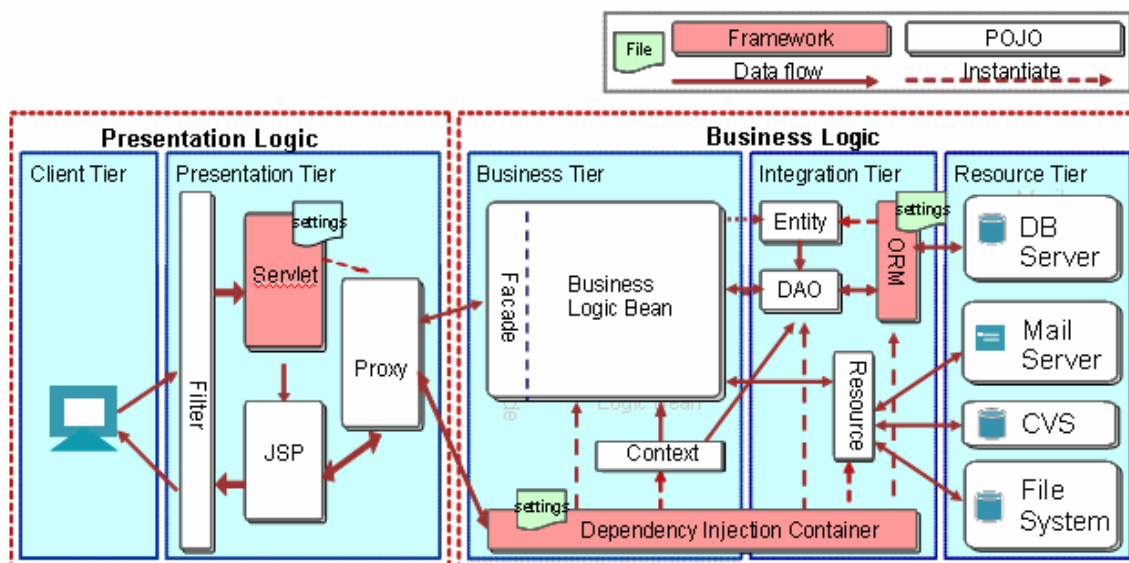


Fig.5: Overview of EtUDE/Core Architecture

4.3 Realization of EtUDE SubTool

In realization of an EtUDE SubTool, it is necessary to enable processing of a class diagram by a program. In this tool changing a class diagram into XMI [18](XML Metadata Interchange) data enables such processing. XMI is the specification for enabling it to exchange mutually with each tool etc. by creating the XML documentation of the model information. Specification is adjusted by OMG [19].

5 Evaluation

EtUDE was applied to the exercise lesson of the software development, and evaluation of the validity of each support function and the validity of the exercise lesson, which introduced EtUDE were evaluated. The log data about the use situation of EtUDE and the questionnaire survey was conducted at the time of the end of an exercise lesson and were used for analysis and evaluation. Moreover, analysis and evaluation of each product by matrices were also performed.

In addition, the contents of the exercise lesson do the analysis and the design of the Web application system of the "conference room reservation system" which is an exercise subject using the object-oriented software design method OOSE [20], and Java is used for program development. In addition, system design supportive tool JUDE [21] is used in the analysis and the design process by OOSE and in the program development environment Eclipse [22] is used in the programming process.

5.1 Evaluation of Communication Support Function

With evaluation of the effectiveness about the bulletin board function, evaluation of the effectiveness of the communication support was done. The evaluation to a bulletin board function used four-step evaluation (four points are the highest score) by the questionnaire about "working efficiency" and "ease of use." As a result, about "working efficiency" and "ease of use" of the bulletin board function, as shown in Fig.6, high evaluation was obtained from more than 80 percent of the students. Moreover, since there was no description of dissatisfaction also with the free entry column of a questionnaire, it can be judged that the support for communication was fully achieved.

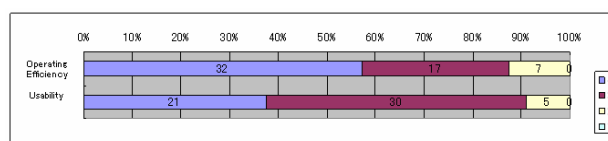


Fig. 6 Evaluation of bulletin board function

Moreover, it was asked what was used as communication means other than the bulletin board of EtUDE/Core. As a result, other means than "E-mail", " Instant Messaging", "mixi" [23] (As Japan's largest social networking site, mixi is the third most-popular Web site in Japan, behind only Yahoo! and Google.) was mentioned as alternative by some students. Considering that the evaluation about the working efficiency of a bulletin board is by no means low, but the judgment of the students is they want to use the communication method usually.

Since it is necessary to use EtUDE/Core so that a teaching person grasps the state of the project precisely, it is necessary to pursue the communication functions that are easy for students to use as communication means.

Table 1: Communication means other than for bulletin board in EtUDE/Core (multiple answers are possible)

| | # of replies |
|------------------------|--------------|
| E-mail | 13 |
| Instant Messaging | 12 |
| Telephone | 5 |
| Another bulletin board | 1 |
| Dialog only | 10 |
| Others | 6 |

5.2 Evaluation of Report Generation Support Function

It has evaluation of the validity about a progress management function, and is considered as evaluation of a report generation support function. As shown in Fig.7, about a progress management function, evaluation was not so high compared with other functions.

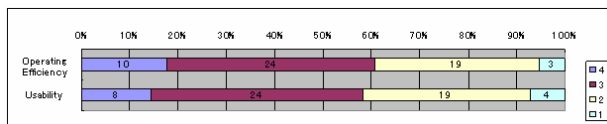


Fig. 7: Evaluation of Progress Management

Table 2: Precision of the Results of the Plan

| | # of Replies (%) |
|------------------------------------------------------|------------------|
| Progressed as Planned. (100 to 80%) | 1 (2%) |
| Progressed to Some Extent as Planned. (79 to 50%) | 19 (34%) |
| Seldom Progressed as Planned. (49 to 20%) | 31 (55%) |
| Plan was not Meaningful. (19 to 0%) | 5 (8%) |
| Plan was not formed. | 1 (2%) |

It is because there were many students puzzled since it did not act that a student manages a progress situation himself. Moreover, when a questionnaire ["whether work was able to be done as planned" (Table 2)] survey is conducted, it turns out that more than half the students answered "Seldom progressed as planned." This result can also be referred to as having lowered the evaluation of the function.

However, a student is required to make a plan and do progress management to study its importance. For this reason, it is necessary to experience progress management as soon as possible.

Moreover, in order that a teaching person might grasp the situation of each project easily, the students were asked to evaluate the progress management function by expressing the present progress situation of one's project by expressing in weather once at one week. The selection of the weather is five-step evaluation as "fine weather", "fine", "cloudy", "rain", "thunderstorm", and each is evaluated in the range 5 to 1, respectively. The average evaluation value of the progress situation of each project by this is shown in Table 3. Whenever it goes to next week, it turns out that evaluation is getting worse. The same tendency was seen in almost all projects.

Therefore when the cause of such an evaluation was analyzed; it was found that because of no exercise lesson in one week, the meeting with the group members could not be held in that week. Because of lower programming skill, it had taken more time for development than expected and lowered the evaluation of the progress of the project was found. That is, it turned out that a teaching person could grasp the progress situation of a project easily by making the progress situation of a project report as simplified evaluation when compared with weather expressions. It was found that the above progress management functions serve as support effective enough for a teaching person.

Table 3: Average of All the Projects About Progress Situation

| Date | Avg. |
|-------|------|
| 11/6 | 3.4 |
| 11/30 | 3.1 |
| 12/7 | 3.2 |
| 12/14 | 2.8 |
| 12/21 | 2.9 |
| 12/28 | 2.5 |
| 1/4 | 2.6 |

5.3 Evaluation of Product Management Support Function

Evaluation of a product controlling function is shown below. About working efficiency, as shown in Fig. 8 and Fig. 9, both of the functions were able to obtain high evaluation from nearly 80% of the students. Moreover, as a result of analyzing the total result of a log, it turned out that the work by

meeting was most actively utilized during the difficult winter vacation.

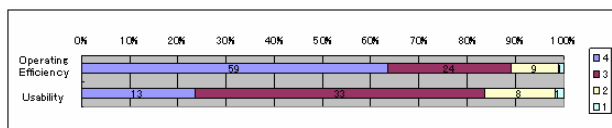


Fig. 8: Evaluation of Deliverables Management

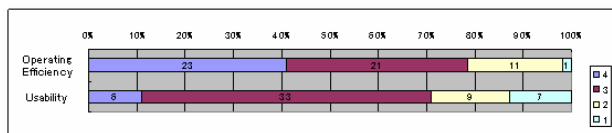


Fig. 9: Evaluation of Source Codes Management

5.4 Evaluation of Issue Tracking Function

Since the snags notice function was seldom utilized, so it led to low evaluation. This is guessed as there was not much time for implementation stage and the bug registration work was pending.

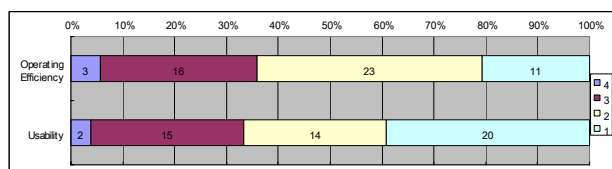


Fig. 10: Evaluation of Issue Tracking

Table 4: Evaluation of Issue Tracking

| | # of replies (%) |
|---------------------|------------------|
| It was helpful. | 2 (3%) |
| It was so-so role. | 8 (14%) |
| It was not helpful. | 9 (16%) |
| It was not used. | 37 (66%) |

5.5 Detect Students who are Required Individual Tutoring by a Design Supportive Tool

In this exercise, it evaluated that how many participants who need individual guidance were detected.

The example of this exercise performs the class design of the lending system of a library, showed the participant the robustness diagram of Fig.11 as data for a class design, and made modeling. Since many-to-many relations occur between books and a user in the case of this exercise, the participant has to avoid many-to-many relations by introducing an association class. A design supportive tool supports identifying the participants who cannot perform modeling appropriately for such a problem.

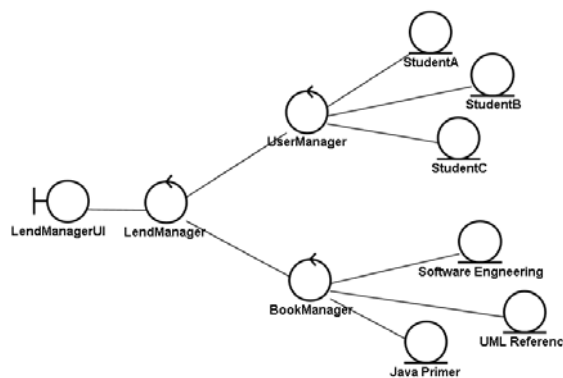


Fig. 11: the Example of Robustness Diagram

In order to check the validity of the tool, experiment was done with the following procedures.

(1) The 1st experiment

- Show robustness picture to the subjects.
- Make the subjects do modeling.

(2) The 2nd experiment

- Explain many-to-many relationship has bad effects, but don't give explanation of a related class.
- Make a subject do modeling.

(3) The 3rd experiment

- Give support by a design tool and point out many-to-many relation's existence part concretely.
- Make a subject do modeling.

Moreover, many-to-many relationship deployment was decomposed into the following elements in order to grasp the result of an experiment in more details.

- Indication of many-to-many relationship existence
- Creation of association class
- Creation of the control classes for association class, and setup of relations.
- A setup of the multiplicity of the relations between an entity class and an association class.
- A setup of the guidance possibility of the relation between an entity class and an association class.

Moreover, in order to investigate whether the subject person recognizes such a problem, result total was done after adding the revised number regardless of right or wrong. A result total is shown in Table 5.

Table 5: Result of Experiment on Many-to-Many Relationship Problem

| | Indication of Existence | Association Class Creation | | Management class creation | | Setup of Navigability | | Setup of Multiplicity | |
|-----|-------------------------|----------------------------|----------------------|---------------------------|----------------------|-----------------------|----------------------|-----------------------|----------------------|
| | | # of Descriptions | # of Correct Answers | # of Descriptions | # of Correct Answers | # of Descriptions | # of Correct Answers | # of Descriptions | # of Correct Answers |
| 1st | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 4.39% | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |
| 2nd | 87 | 59 | 59 | 59 | 19 | 42 | 26 | 49 | 26 |
| | 89.69% | 67.82% | 67.82% | 67.82% | 21.84% | 43.30% | 61.90% | 59.52% | 53.06% |
| 3rd | 94 | 87 | 79 | 87 | 67 | 49 | 36 | 86 | 34 |
| | 95.92% | 92.55% | 84.04% | 92.55% | 71.28% | 50% | 73.47% | 87.76% | 39.53% |

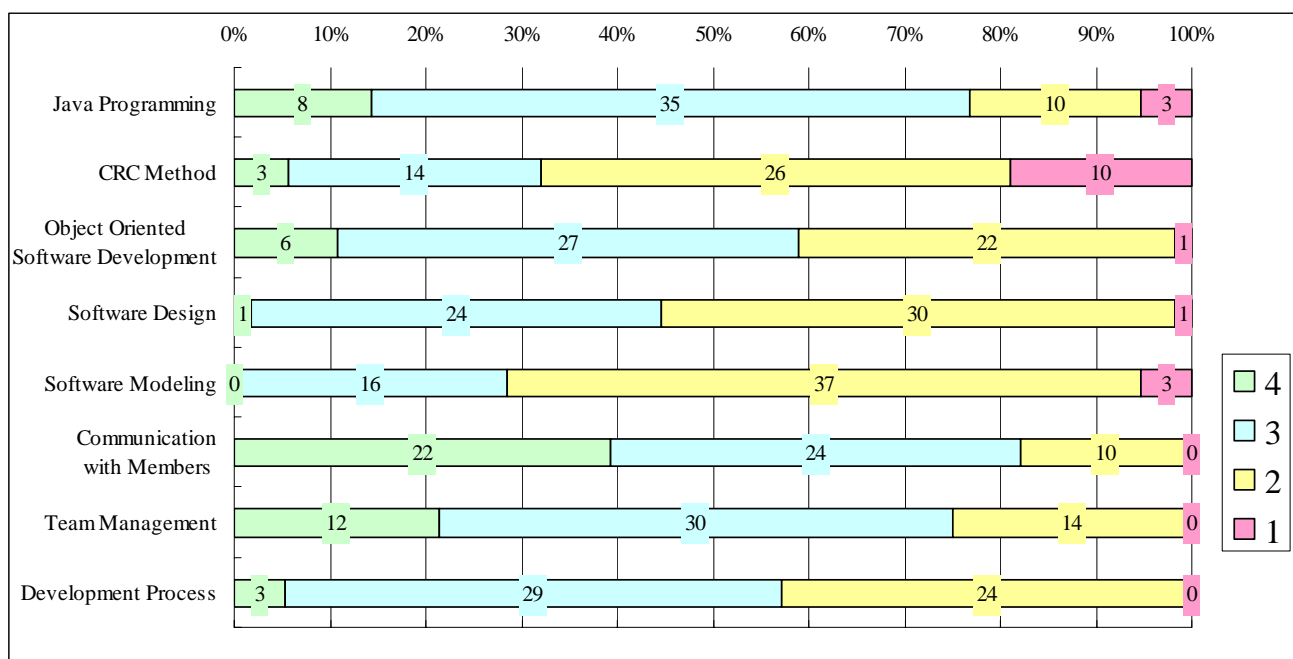


Fig.12: Evaluation According to Study Item of the Exercise

From the 1st result lowness of recognition many-to-many relationship problem can be seen. It is understandable that the problem is that it cannot recognize many-to-many relationship problem were not studied beforehand.

Moreover, from the 2nd experiment it shows that about half of the participants were unable to correct by themselves, and individual guidance is required for half of the participants, even if the bad effects of many-to-many relationship were explained.

From the 3rd experiment, the participants who need individual guidance were identified with a tool and individual follow up shows that about 90% of students can perform modeling now by paying attention to multiplicity. It shows that the tool has identified the participants appropriately who needs individual guidance. However, the final correct ratio is only 40%, and there is room for an improvement about the method of individual guidance.

5.6 Evaluation of Exercise Which Introduced EtUDE

In order to obtain overall evaluation of the exercise using EtUDE, the questionnaire, which evaluates in four steps each item currently studied by this exercise, was carried out. The result that there were many students who think they were able to make improvement in the skill about project management, such as "Communication with Members" and "Team Management", was obtained from the result of the questionnaire.

In addition, effects are high about the items of development process, and it is thought that the objectives of achieving skills other than the technical skills were almost attained. .

Moreover, when the questionnaire was taken about overall evaluation of the exercise, the result that it

was substantial from 80% or more of the participant was obtained as follows.

Table 6: Substantiality Degree Evaluation of the Exercise

| | # of Replies (%) |
|----------------------------|------------------|
| It was very substantial. | 7 (13%) |
| It was substantial. | 36 (68%) |
| It was seldom substantial. | 9 (17%) |
| It was not substantial. | 1 (2%) |

With the support of EtUDE, the progress grasp of the exercise that students performed was attained to some extent, and a teaching person was able to assign time now by instructions to students. Thereby, the high evaluation of the exercise class was confirmed.

6 Conclusion

This paper described the actual condition of the software engineering education in Japanese companies in the beginning, and in order to bring up the students with high faculty with the exercise feelings for which a company asks, it described carrying out the group exercise of a software development at the Institute. Next, in order to clarify the problems of software engineering education in a Japanese university and to resolve such problems, the system EtUDE that supports the group exercise of a software development was developed, and it was shown clearly that it has been introduced in the exercise classes. While describing the outlines of EtUDE, by applying EtUDE to a exercise class and evaluating it showed the validity of each support function, which EtUDE has. Furthermore, the validity of the exercise class, which introduced EtUDE was also shown. Moreover, the un-resolved problems were clarified.

References:

- [1] Tatsuya Ishikawa, Hiroaki Hashiura, Koutarou Yamashita, Tooru Kuwabara, Yumei Qiu, and Seiichi Komiya, "Environment for Ultimate software Development Exercise (EtUDE)," *Technical report of IEICE. KBSE*, vol.105, no.653, K BSE2005-48, 2006, pp.7-12. (In Japanese)
- [2] Soojin Jun, Seungbum Kim, Wongyu Lee, "On line Pair-Programming for Learning Programming of Novices," *WSEAS Transactions on Advances in Engineering Education*, Issue 9, Vol. 4, 2007, pp.187-192.
- [3] Tom DeMarco, and Timothy Lister, "*Peopleware: Productive Projects and Teams (2nd Edition)*," Dorset House Publishing, 1999.
- [4] Kristina Lapin, and Saulius Ragaisis, "Integrating team projects into the SE Curriculum", *WSEAS Transactions on Advances in Engineering Education*, Issue 3, Vol. 8, 2008, pp.104-110.
- [5] Saeko Matsuura, "Practical Software Engineering Education based on Software Development Group Experiments," *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education (E-Learn) 2005*, 2005, pp.1701-1708.
- [6] Jane Huffman Hayes, Timothy C. Lethbridge, and Daniel Port, "Evaluating Individual Contribution Toward Group Software Engineering Projects," *Proceedings of International Conference on Software Engineering (ICSE) 2003 Software Engineering Education and Training (SEET)*, 2003, pp.622-627.
- [7] Juan A. Marin-Garcia, and Jaime Lloret, "Improving Teamwork with University Engineering Students. The Effect of an Assessment Method to Prevent Shirking," *WSEAS Transactions on Advances in Engineering Education*, Issue 1, Vol. 5, 2008, pp.1-11.
- [8] Chyng-Yang Jang, Charles Steinfield, Ben Pfaff, "Virtual group awareness and groupware support: an evaluation of the TeamSCOPE system," *International Journal of Human-Computer Studies*, Vol. 56, Issue 1, Academic Press, 2002, pp.109-126.
- [9] Kunihiko Chiken, Atsuo Hazeyama, "Awareness Support in Group-based Software Engineering Education," *System Proceedings of the 10th Asia-Pacific Software and Applications Conference (APSEC2003)*, IEEE Computer Society Press, 2003, pp.280-289.
- [10] N. Kamigochi, and S. Matsuura, "A Learning Support Tool for Testing Java Programs," *Proceedings of the 25th conference on IASTED International Multi-Conference: Software Engineering (SE2007)*, 2007, pp.273-278.
- [11] S. Matsuura, and H. Kurihara, "Collaboration Support Model of Software Development Experiment," *Proc. of the 18th International Conference of Software Engineering and Knowledge Engineering*, 2006, pp.23-28.
- [12] Saeko Matsuura, and S. Atsuta, "Lesson Support System for Programming Exercise Lesson Improvement," *Proceedings of The 4th IASTED International Conference on Web-Based Education (WEB2005)*, 2005, pp.131-136.
- [13] Saeko Matsuura, "An Evaluation Method of Project Based Learning on Software Development

- t Experiment,” *Proceedings of the 37th ACM Technical Symposium on Computer Science Education*, 2006, pp.163-167.
- [14] Atsuo Hazeyama, Akiko Nakano, Sachiko Nakajima, and Keiji Osada, “Group Learning Support System for Software Engineering Education,” *Proceedings of the 2001 International Conference on Web Intelligence (WI2001)*, Vol. 2198, 2001, pp.568-573.
- [15] The Mozilla Organization, “*The Bugzilla Guide*”, <http://www.bugzilla.org/docs/3.0/html/lifecycle.html>.
- [16] Gregor Kiczales, John Lamping, Anurag Mendhekar, Chris Maeda, Cristina Videira Lopes, Jean-Marc Loingtier, John Irwin, “Aspect-Oriented Programming,” *Proceedings European Conference on Object-Oriented Programming (ECOOP)*, pp.220-242, 1997.
- [17] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, “*Design Patterns: Elements of Reusable Object-Oriented Software*,” Addison-Wesley Professional, 1995.
- [18] Object Management Group, Inc., “*XML Metadata Interchange (XMI) version 1.1*,” <http://www.omg.org/cgi-bin/apps/doc?formal/00-11-02.pdf>.
- [19] Object Management Group, Inc., “*The Object Management Group (OMG)*,” <http://www.omg.org/>.
- [20] Ivar Jacobson, Magnus Christerson, Patrik Jonsson, and Gunnar Overgaard, “*Object-Oriented Software Engineering: A Use Case Driven Approach*,” Addison-Wesley, 1992.
- [21] Change Vision, Inc., “*JUDE Community Site - UML Modeling Tool*,” <http://jude-users.com/en/>.
- [22] The Eclipse Foundation, “*Eclipse*,” <http://www.eclipse.org/>.
- [23] mixi, Inc. “*mixi*,” <http://mixi.jp/>. (In Japanese)