

Development of a Questionnaire System with a Response Analyzer

Hiroataka Azuma, Tokyo Denki University, Japan
Hironobu Takenouchi, Tokyo Denki University, Japan
Tatsuyuki Takano, Kanto Gakuin University, Japan
Osamu Miyakawa, Tokyo Denki University, Japan
Takashi Kohama, Tokyo Denki University, Japan

The Asian Conference on Education 2019
Official Conference Proceedings

Abstract

It has been increasing that the number of universities which try to give a bidirectional lecture by improving the contents of it from responses of students. To achieve this, instructors need to know them efficiently and easily. Questionnaires are general methods used to grasp the situation of students. A lot of universities have been adopting them. In general, instructors collect the results of questionnaires after their classes and examine them to improve their next lectures. However, it is difficult to understand students' questions and responses during classes since questionnaires are distributed after them. In order to solve this problem, a response analyzer can be used to make it easier to grasp students' reactions. A response analyzer is an educational device that aggregates students' situation and shows the results of it in real time. Instructors can fix their lectures according to them during their classes. In this study, a questionnaire system has been developed with the response analyzer. Adding to the existing questionnaires, this study's system enables instructors to make a single-question questionnaire and let students answer it. Instructors can know the results in real time with graphs and tables. They can improve their lectures not only after classes but also during them. Although there is also a problem that many of the existing response analyzers need specific devices and introduction costs, this system can be used through a browser and is easy to be introduced to a lecture since this system has been developed as a web application.

Keywords: Response analyzer, Questionnaire, Response, Question, Web application

iafor

The International Academic Forum
www.iafor.org

Introduction

The Ministry of Education, Culture, Sports, Science and Technology states that conventional single-directional lectures need to be changed to bidirectional ones where students can solve problems actively (The Ministry of Education, Culture, Sports, Science and Technology-Japan, 2012). In order to realize the active learning, instructors are required to encourage students to do the learning by pulling out their ideas and representations. For the reason, instructors should improve their lectures by understanding their students' reactions.

The level of knowledge and understandings for a lecture can vary according to each student. Especially at a university, there are a lot of single-directional lectures. Because of this, it is not easy to catch each student's questions and/or learning situation. Additionally, there are a few students who ask their instructors some questions in Japan since they usually care what other students think (Fuji & Yamaguchi, 2003).

One of the usual methods to grab students' reactions is one using a questionnaire. A questionnaire is a means to catch their situation adopted by a lot of universities. Instructors can improve their next lectures by viewing results of questionnaires done during class. This method can work to a certain degree; however, it cannot be possible to improve lectures during them. To understand students' levels in more detail, questionnaires should be used and the results of them should be reflected during class. But it is difficult to implement this because instructors' burdens are not light.

One of the solutions to the problem is to utilize a response analyzer to catch students' understandings easily. A response analyzer is an educational device that collects reactions from students and shows the results of them in real time. Because of this, instructors can improve their lectures during class.

In this study, a questionnaire system with a response analyzer has been developed. In addition to the existing functionality, this system also enables instructors make a questionnaire during class and get feedback in real time to improve their lectures.

A Response Analyzer

A response analyzer is an educational device that collects responses from a lot of learners instantly and quantitatively and shows the results. It shows aggregated results to an instructor after learners answer some questions using such as a clicker. An instructor can gain materials for deciding how s/he proceed his/her lecture, and s/he also can advise learners properly. Unlike existing way to collect reactions such as raising hand, learners can response secretly so their opinions are tolerant to others' ones. Learners may feel resistance to raise their hand for some reasons. This device can ease the feeling and make it possible to catch responses from learners instantly and efficiently.

But many types of a response analyzer such as a clicker system requires a special device and a software that cost a lot for both students and instructors (Ohinata et al., 2015). By considering device-independence, our system has been developed as a Web application.

System Overview

In this study, a questionnaire system with a response analyzer has been developed. This system is a Web application consisting of a server, a client on an instructor side, and a client on a student side.

This system has two types of a questionnaire: one that can contain multiple questions and one that has a single question. Like an existing questionnaire, a multiple-questioned one allows instructors to create a questionnaire before class, then submit it to students at an arbitrary time. A single-questioned one can be made during class when instructors want to know students' situation or to ask some questions. We call this single-questioned one a simple questionnaire. A simple questionnaire is submitted to students as soon as instructors have created it, and students are required to answer it. This is one of the response analyzer's functionalities that enables instructors to know their situation instantly.

This system has been assumed to be used in class in Japan so the language used in it is Japanese correspondingly. But it is replaced with English in examples of this paper.

A Client on An Instructor's Side

Main functionality of a client on a instructor's side contains creating a questionnaire and gaining students' reactions. There are differences in creation and management between a multiple-questioned questionnaire and a simple one. A multiple-questioned one can be managed per subject. And a unique ID will be assigned to a subject when an instructor types the name of it in this system (See Figure 1). A student can learn the subject and view and answer a questionnaire by entering an arbitrary ID assigned to a client on his/her side.

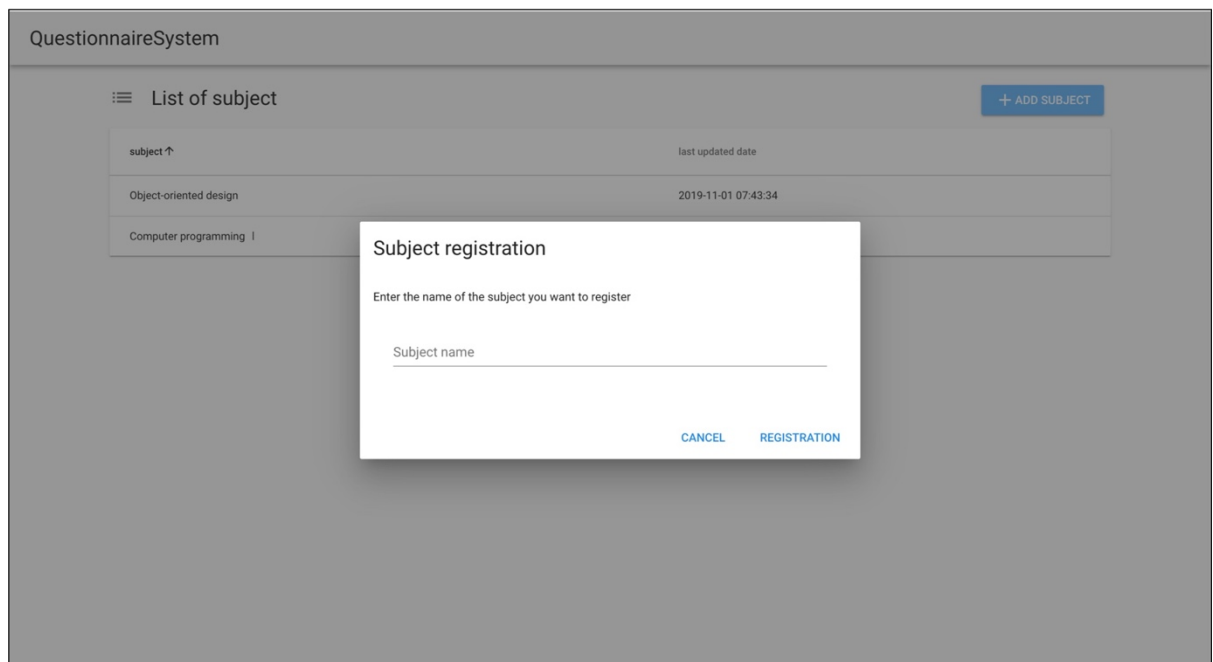
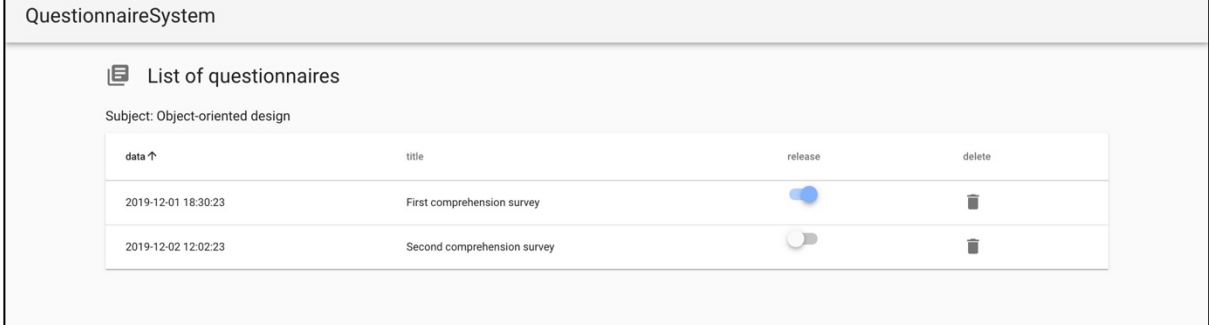


Figure 1: Subject registration on an instructor's side

Two or more multiple-questioned questionnaires can be made for each subject so they can be divided based on such as date and time, usage, and so on. The multiple-questioned questionnaires are also shown as a list for each subject, and an instructor can jump to a screen s/he wants to go to such as their names, date and time they were created, toggling of public and private, and their results (See Figure 2).



The screenshot shows a web interface titled "QuestionnaireSystem". Below the title is a section labeled "List of questionnaires" with a sub-header "Subject: Object-oriented design". A table lists two questionnaires. The first row has a date of "2019-12-01 18:30:23", the title "First comprehension survey", a blue toggle switch for "release", and a trash icon for "delete". The second row has a date of "2019-12-02 12:02:23", the title "Second comprehension survey", a grey toggle switch for "release", and a trash icon for "delete".

data ↑	title	release	delete
2019-12-01 18:30:23	First comprehension survey	<input checked="" type="checkbox"/>	
2019-12-02 12:02:23	Second comprehension survey	<input type="checkbox"/>	

Figure 2: List of questionnaires on an instructor's side

The state to be public or private means the questionnaires can be toggled to appear to students or not. After creating the multiple-questioned questionnaire, its state is going to be private automatically, so an instructor needs to toggle it to be public at an arbitrary time. Private ones cannot be shown in a client on a student's side, so they cannot be answered and their results cannot be viewed.

A simple questionnaire is a questionnaire containing a single question. This questionnaire appears to students as soon as this is created, and forces them to answer. Since a simple questionnaire contains only one question, can answer it without thinking about other ones.

The ways to create a multiple-questioned questionnaire and a simple one are the same (See Figure 3 and Figure 4). Available formats to answer are a single selection, a multiple selection, and a free description. A single selection requires only one from some options whereas a multiple selection need arbitrary numbers of answers. A free description allows an answerer to write letters s/he wants to freely. Arbitrary items of a multiple-questioned questionnaire can be toggled to be answered absolutely. If some items are toggled in that way, a student cannot send the questionnaire without answering them, in other words, those items must be answered. Since a simple questionnaire consists of only one question, it requires an answerer to answer absolutely.

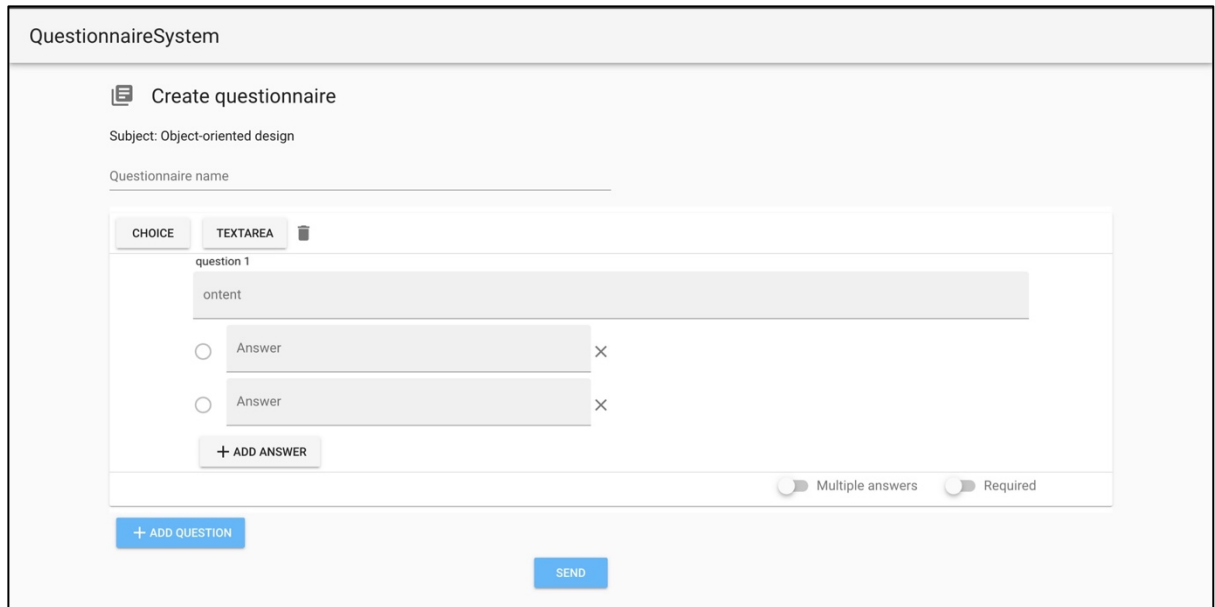


Figure 3: Screen to create a multiple-questioned questionnaire

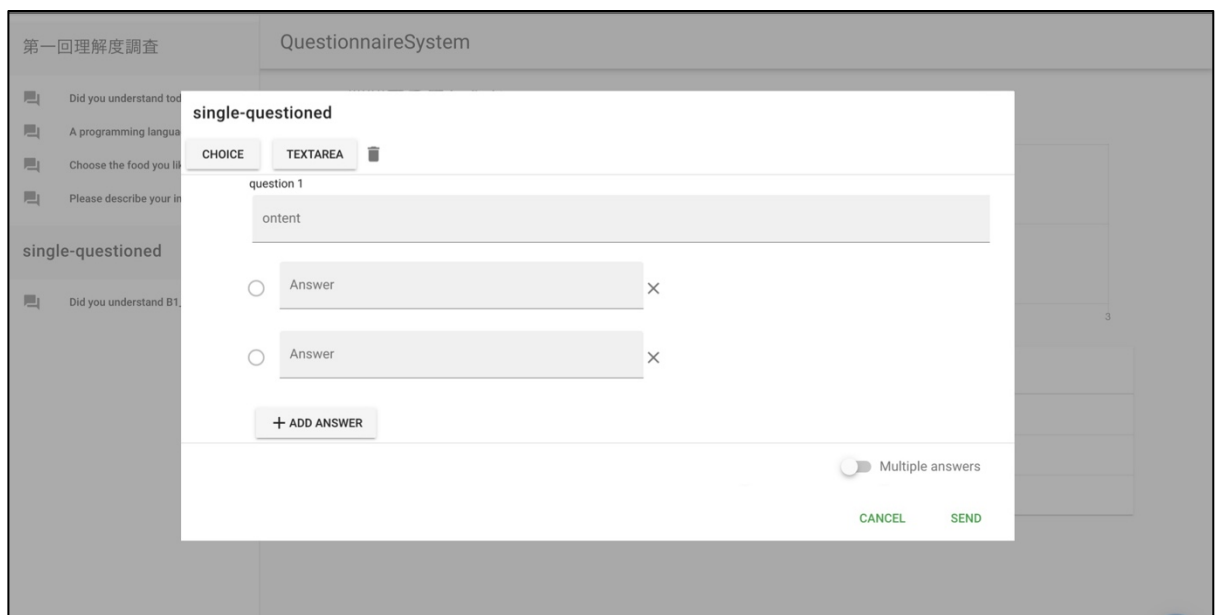


Figure 4: Screen to create a simple questionnaire

An instructor can view the situation of students on the screen of questionnaires' results (See Figure 5). On this screen, an instructor can choose which item to see at the bar on the left-hand side. The results of questionnaires are shown as graphs or tables. In terms of graphs, the aggregated results are shown as a bar graphs where maximum value is the number of students who take a subject. Because of this, an instructor can see the results graphically. In terms of tables, an instructor can view answered data in more details such as a student's name and ID number. And each column can be sorted so the instructor can control the table. Students who did not answer are shown as "Unanswered" so an instructor can monitor those students accordingly. Since the values of the results will be updated and shown in real time, an instructor does not need to operate this system to view their reactions during class.

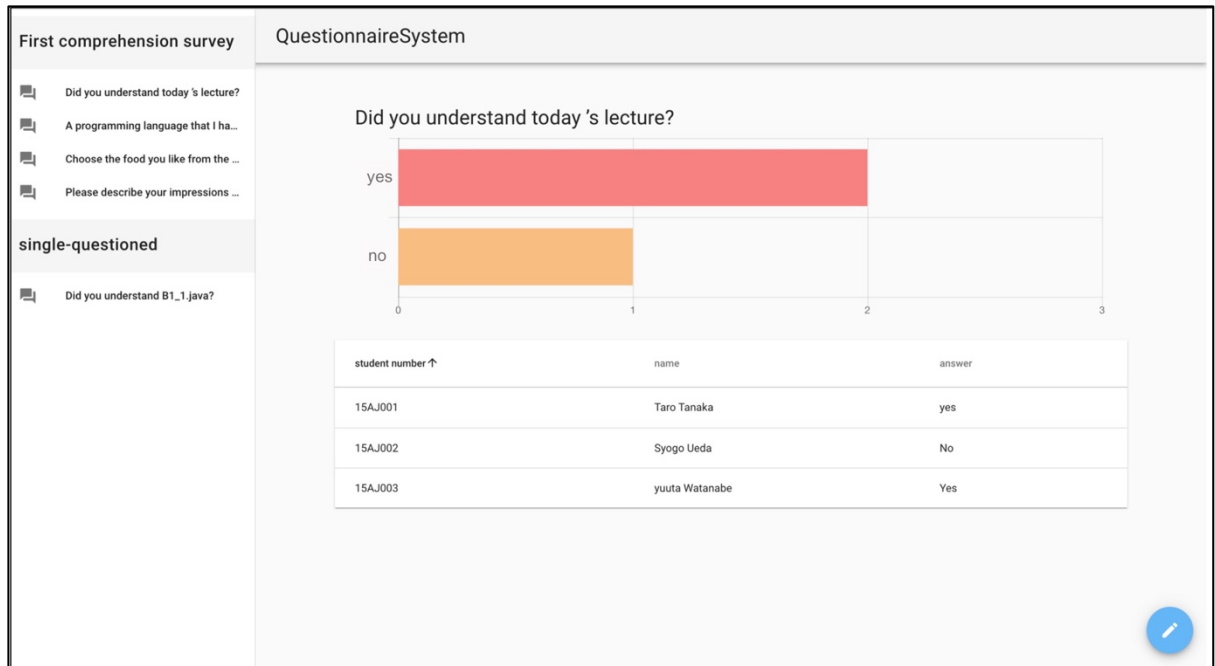


Figure 5: The results of a questionnaire on an instructor's side

A Client on A Student's Side

Main functionality of a client on a student's side contains answering a questionnaire and gaining the response status of a questionnaire. A unique ID is assigned to each subject and a student can answer a questionnaire and view the results of aggregation by entering the subject's ID in this system. Because of this, a client on a student's side needs to register a subject (See Figure 6). The registration is complete by typing a subject ID provided by an instructor.

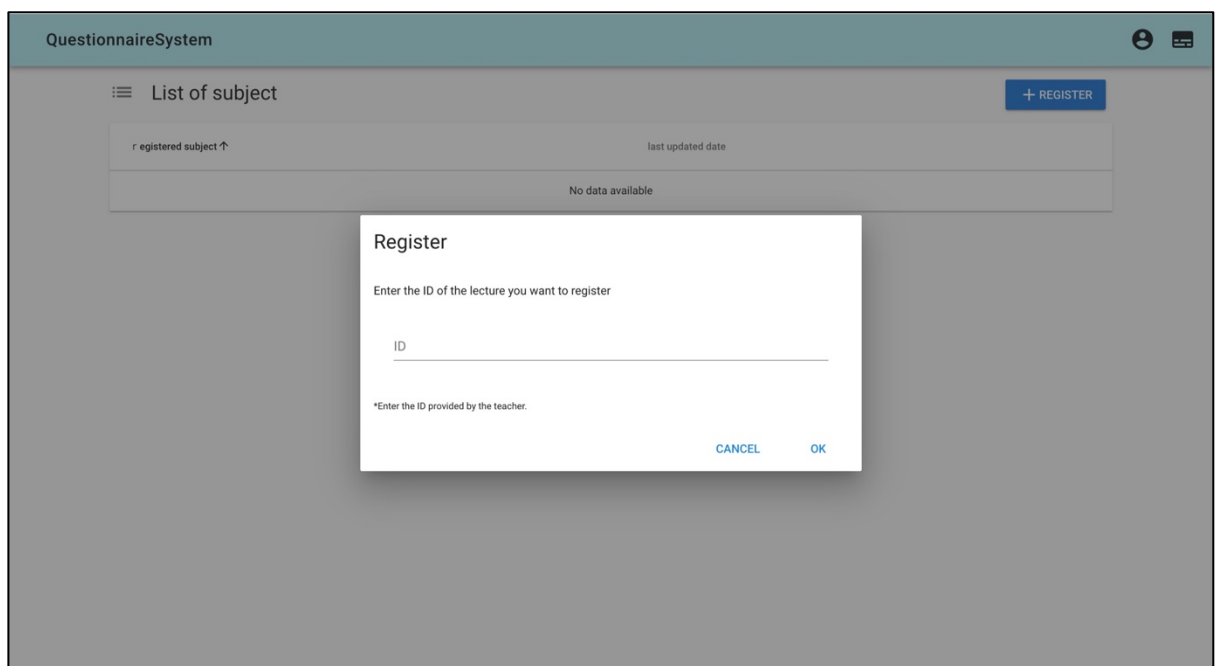


Figure 6: Subject registration on a student's side

Questionnaires are shown per subject (See Figure 7). Questionnaires in the figure are multiple-questioned ones created by an instructor. Only questionnaires created by a client on an instructor's side will appear.

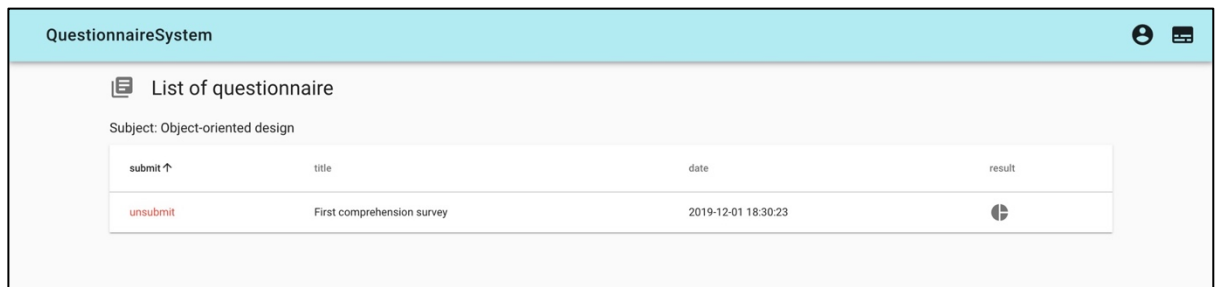


Figure 7: List of questionnaires on a student's side

By choosing one of the multiple-questioned questionnaires listed, a student can answer it (See Figure 8). How to answer varies according to the format of the question such as selection and description. This system confirms that items which are toggled to be answered absolutely before sending the results so it may returns feedback to force a student to fill them if there are unanswered ones (See Figure.9). This system sends the results and completes if all the questions that must be answered are finished.

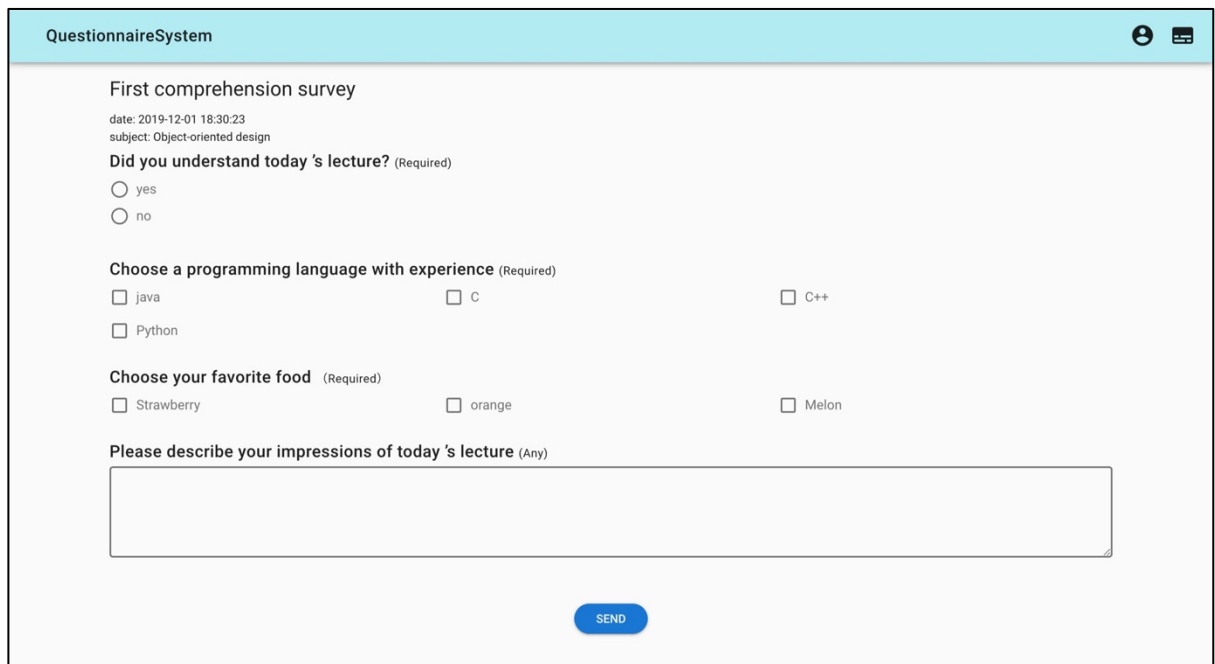


Figure 8: Screen to answer a multiple-questioned questionnaire

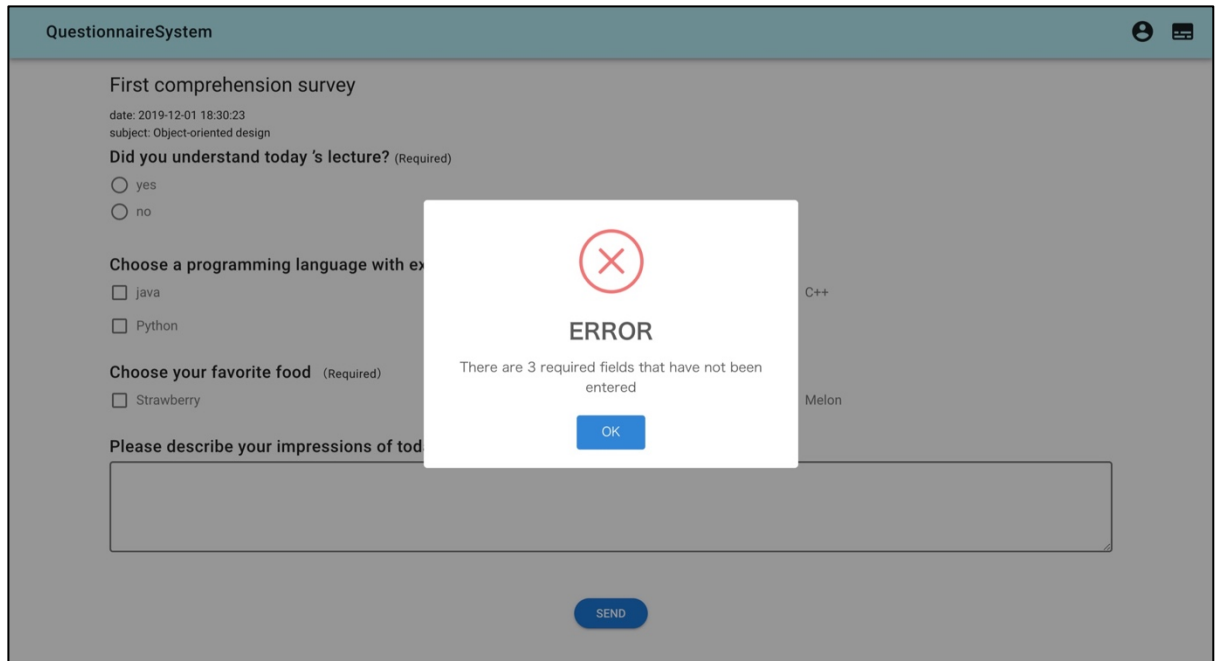


Figure 9: Feedback when questions that must be answered are unanswered

The results of a questionnaire will appear by selecting the pie chart on the right-hand side (See Figure 10). On the results screen, all the results for each question will be shown based on ratios in pie chart.



Figure 10: The results of a questionnaire on a student's side

When an instructor of a subject creates a simple questionnaire, all the students who take the subject will see a dialog to answer (See Figure 11). This dialog will never disappear unless a student answers it.

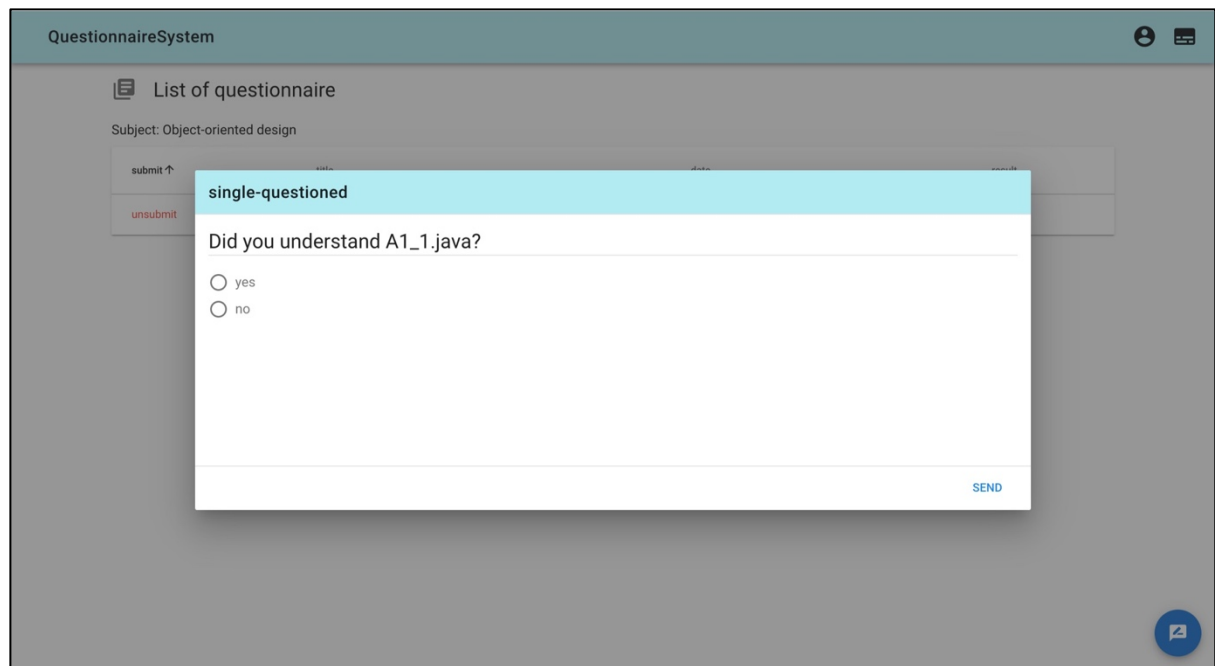


Figure 11: Screen to answer a simple questionnaire

Multidevice Supported

The number of users who use smartphones and/or tablets has been increasing recently. With this trend, it also has been increasing that the number of accesses to the same Web sites from a lot of devices with different screen sizes. To support the multidevice trend, there are demands to show Web sites in a proper screen size by introducing the responsive design even if a variety of devices access to the same one. Using the design, the usability of a device with a small screen will be improved.

By considering the spread of smartphones and classes without PCs, this system has adopted the responsive design for being accessed by a variety of devices. Supported screen sizes are for smartphones, tablets, and laptops. One example of differences of UIs among them is an icon button on a header part (See Figure 12). When you view this system using your laptop, the contents of functionality will be displayed while the cursor is hovering over the icon. But no cursors exist in a smartphone so you do not know what functionalities lie. Because of this, this system's UI for smartphones does not have the icon but have texts to show functionality to be chosen.

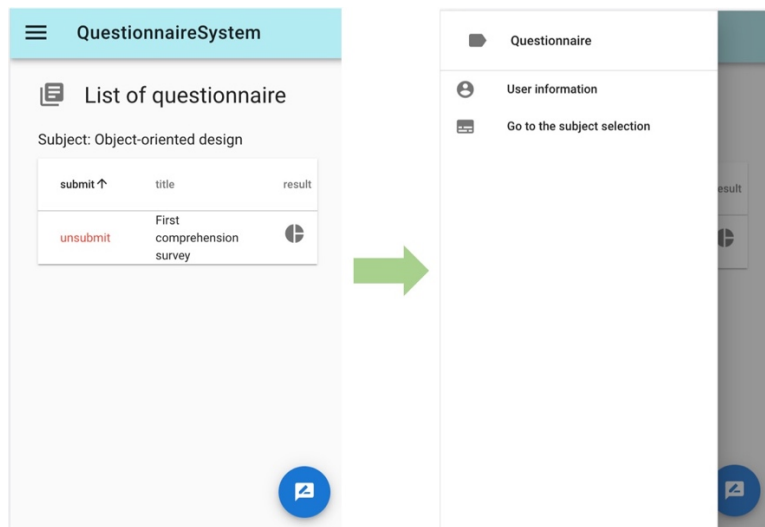


Figure 12: UIs of different devices

Server

This system's server has adopted REST API, and sends/receives information about a questionnaire using it. Data format is JSON. The server keeps data submitted by a client with time and ID. A client can receive necessary data from the server by calling the API.

There are two types of APIs: for an instructor and for a student. Instructors can create a questionnaire and delete it and students can answer it. The APIs vary according to the usage. And both of instructors and students can receive questionnaires, but instructors can see student's information on it whereas students cannot. Because of this, returned information varies according to the usage even if the same API is called. A client designates API based on a user and a usage to get necessary data.

Conclusion

In this study, a questionnaire system with a response analyzer has been developed by considering the device-independence. Using this system, instructors can grab students' situation instantly, and improve their lectures according to the demands. Students are free from limitation of devices to be able to answer the questionnaires. So instructors can improve their lectures more easily and teach more properly. Students also receive a setting where they can ask questions more freely and understand lectures well.

References

The Ministry of Education, Culture, Sports, Science, and Technology-Japan. (2012). Daigakukyoikubukai no shingi no matome ni tsuite (soan) [About the Summary of Deliberation at The Social Meeting of University's Education (Draft)]. Retrieved from http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo4/015/attach/1318247.htm, (Accessed on May 30th, 2018)

Toshie Fuji & Hiroyuki Yamaguchi, (2003). Daigakusei no jugyochu no shitsumonkodo ni kansurukenyu:gakusei ha naze jugyochu ni shitsumon shinainoka? [Study on Actions to Ask Questions of College Students during Class:Why do not they ask questions during class?].

Hiroshi Ohinata, et al., (2015). *Daigakukyoiku eno souhoko communication tool "clicker" no donyu -shikakushikentaisakujugyo eno tekiyo-* [Introduction of a Bidirectional Communication Tool 'Clicker' for University's Education -Application for Lectures of Qualification Test-], pp.161-168.

Contact email: 19JKM02@ms.dendai.ac.jp