Automatic of Correction and Program Evaluation Using Web-Based Systems

Sunu Pinasthika Fajar
Informatics Department,
Faculty of Science and Technology
UIN Sunan Kalijaga
Yogyakarta, Indonesia
sunupf@gmail.com

Muhammad Dzulfikar Fauzi
Informatics Department,
Faculty of Science and Technology
UIN Sunan Kalijaga
Yogyakarta, Indonesia
dzulfikar1234567@gmail.com

Maria Ulfah Siregar
Informatics Department, Master Program
Faculty of Science and Technology
UIN Sunan Kalijaga
Yogyakarta, Indonesia
maria.siregar@uin-suka.ac.id

***Abstract*—Assistant or lecturer in a practicum on algorithmic learning courses it often takes a long time to correct a task collected by the practitioner. The number of tasks that must be corrected many will require a longer time. To simplify the work of the assistant or the lecturer developed a system that automatically corrects tasks that are collected by the practitioners with the black box testing method, so that the assistant only needs to publish the task and wait for the task to be corrected automatically and view or download the correction results. With the existence of this system, it is expected that the correction of tasks will be more fast, easy, and effective, and make students more able to get grades better because students are allowed to collect their assignments many times to be satisfied with the value.**

Keywords- auto-correcting; program; black box testing; lab work.

I. Introduction

A common thing is the consistency of human performance difficult to maintain, especially when compared to the engine. Like in terms of assessment answer is not a checkpoint. Especially for those who do not have standards definite judgment. Usually, there are differences in standards of assessment between answers corrected early with the final corrected answer. Sometimes standards are getting tougher, not infrequently the standard is getting slower. Especially if the number of answers that must be corrected is a large number. Consistency is increasingly difficult to maintain. Of course, this will harm students indirectly because getting a value that might not be what it should be. Not yet again if the value recap is needed for the final semester value calculation, performance lecturers are required to be greater because of the results of the correction of the answers that have not been recapitulated must be manually written into the file recap value.

These conditions often occur in practicums, including practicum programming learning, especially for classes with a number of students many and often students collect assignments each week in the form source code. This makes it very difficult for the lecturer or assistant to do the correction. You can imagine if the number of students reaches 50 people then the assistant or lecturer needs to correct 50 source codes and then test them one by one. Finally, the activity of assessing and recapitulating values becomes an activity that is not fun and finally correction was done at the end of the semester and make work pile up.

Researching automatic assessment for answers essay, use the hybrid method and apply it to Moodle. This system will check an essay answer by breaking the sentence into syllables. Then each syllable will be compared with the WordNet database, whether it is in accordance with the correct English spelling or not. If already the results will be saved into the system database and similarities from the WordNet database to the ontological structure [1].

Research on a system that simulates the correction of data errors transmitted between computers by the FEC method and implemented with Visual Basic. The input from this program is the bit that will be sent with a minimum length of 4 bits and a maximum of 20 bits. The essence of this system is hamming code that is inserted in the sent bits. This hamming code will be used for correction [2].

Researching spelling correction using Damerau Levenshtein Algorithm. This algorithm checks 4 types of spelling errors, namely one letter replacement, one letter insertion, one letter removal, and an exchange of two adjacent letters. The resulting output is a suggested word, which is a word that is the result of the correction of the word with the wrong spelling [3].

For that, a correction and automatic assessment system for a program that is made in practicum is really needed. Especially in world education. Besides facilitating lecturers and practicum assistants, as well as students also feel confident with the existing assessment because they have clear standards and good objectivity.

II. Problem Formulation and Purpose

A. The formulation of the problem

- How to make a program correction application automatically uses web-based applications.
- How to make a neat recap, so that it can facilitate lecturers or assistant to read it.
- How to make appropriate assessment standards.

B. The objectives of this study

- Design and implement a program correction application automatically using a web-based application.

III. Research Method

A. Data Colection

Data collection method used in this study is by conducting a literature study on the assessment variables have been chosen before, read books and use the internet especially the question and answer forum as a source of information.

B. Model Design

After conducting data collection than then the model of the system is to be created, namely the groove the work of the system that will be created and estimating the variable usage that has been obtained previously.

C. System Development

This research requires hardware and software for developing the planned system. Next hardware and the software used in this study are:

- Intel® Atom N270 @ 1.60 GHz processor (processing unit).
- 1014MB memory (temporary storage).
- 80GB hard drive (permanent storage).
- 64MB graphics card (display graphics).
- Linux Ubuntu 11.04 Desktop Edition (Operating System).
- Microsoft Windows XP Home Edition Service Pack 3 (Operating System).
- Apache PHP Server.
- MySQL (database server).
- Heidi SQL (MySQL IDE / Client).
- Geany (text editor).
- Notepad ++ (text editor).
- Pencil Project (Mockup tool).
- Mozilla Firefox 6.0, Opera, Chrome, Safari (web browser).
- CodeIgniter 2.0.2 (framework used).

D. System Development Method

In developing the system this is done with using the method that follows the following steps:

- Requirement Analysis and Specification Phase

  The phase to know what customer needs accurately and document them properly.

- Design Phase

  The phase of changing the existing documentation into the structure allows for implementation into a programming language.

- Implementation and Unit testing Phase

  The phase of implementing the design that has been made and does a small test on the existing module.

- Integration and System Testing

  The phase of checking the system as a whole, from beginning to end.

IV. Analysis And System Design

A. Requirement Analysis dan Specification Phase

Phase to accurately identify customer needs and document them well. System users are entities that will use the system. On this system's automatic correction system as:

1) Question Maker

The question maker is the user who is in charge of making questions in the system, where the question is a matter of which will be published and addressed to answering questions. Question maker too who has the right to make classes and add answer questions into it and see the value recap of the answerers about that registered.

2) Question Answer

The answer to the question is the user who has been registered by the question maker into the class that was created. Answering Questions also answers published questions where the answer will be calculated. Details of actors and activities in the system can be seen more clearly in Table 1.

TABLE 1. TABLE USERS AND ACTIVITIES

User	Activity
Question Maker	Register the System Managing Classes Make class Add Answering Questions Editing Classes Deleting Classes Manage Questions Making a Question Editing Questions Delete the question Download and view Value Recap Answering Questions
Question Answer	Answering Questions See the value

3) System Organizational Structure

The Automatic Correction System has a simple system structure of user systems. This system only has 2 types of users and the user is joined in a class. The organizational structure of the system can be seen in Figure 1.

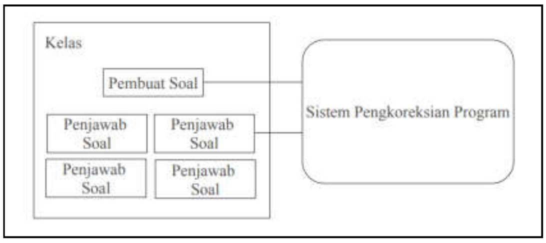


Figure 1. System organizational structure

B. Design Phase

The phase of changing the documentation in the structure that allows for implementation into a programming language. Modeling for the correction system uses UML.

1) Use Case Diagrams

The use case diagram describes the interaction between the system, the external system, and the user or in other words, describe who will use the system and in what way the user expects interaction with the system [4]. From the previous data collection, this diagram will consist of 2 actors and several use cases. The use case diagram of the system can be seen in Figure 2.

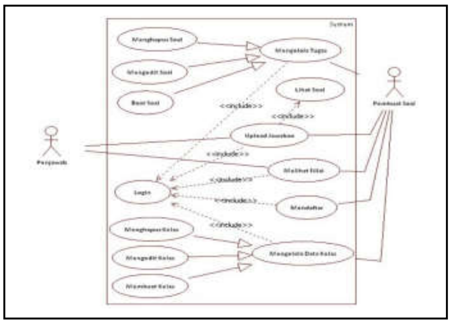


Figure 2. Activity Diagram for the Manage Problem process

2) Class Diagram

The class diagram of the system can be seen in Figure 3. The relationship between one class and another class. The relationship also indirectly describes the relationship in the database later.

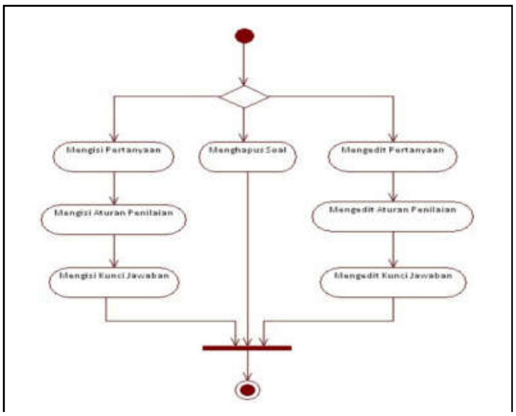


Figure 3. Class Correction System Diagram for Program Correction

3) *Activity Diagrams*

Activity Diagram describes business processes and sequence of activities in a process. Activity Structure Diagrams are similar to flowcharts in structured programming. This diagram can be made based on one process or several processes. Manage questions is one process that can only be done by the question maker. Question makers can do three activities when managing questions, namely making questions, editing questions and deleting questions. When the maker of the question makes a question then he has to fill in the question set the assessment rules, and make the answer key at once. Likewise when editing the question maker can edit the question or edit the assessment rules even edit the answer key. For more details can be seen in Figure 4.

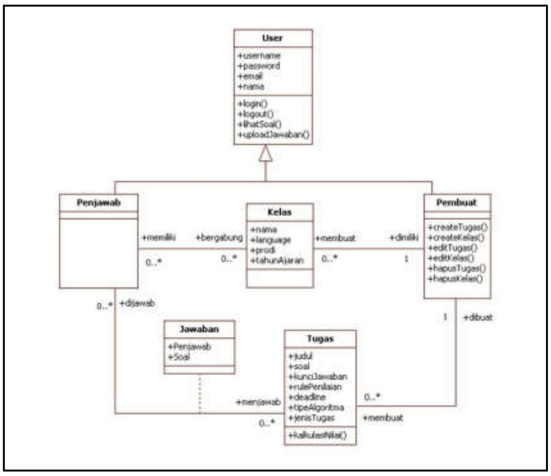


Figure 4. Use case Automatic Correction System diagram

The process of managing questions, this process can only be done by the question maker. Question makers can do three activities when managing classes, namely making classes, editing classes, deleting classes. When the maker of the question makes the class then he must fill in class information fill in the types of assignments in class and register students in the class. While editing the question maker can edit class information, or add or delete students. For more details can be seen in Figure 5.

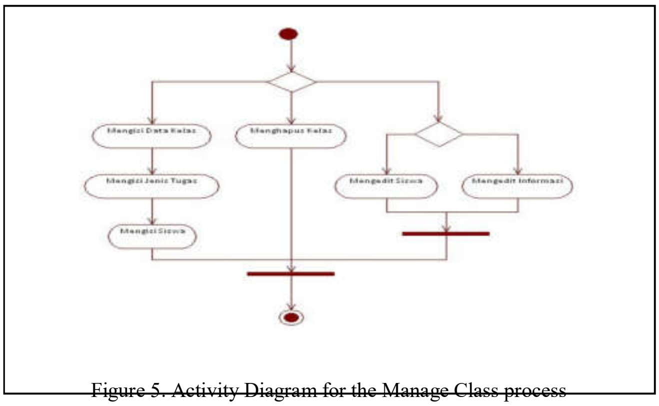


Figure 5. Activity Diagram for the Manage Class process

4) *User Interface*

The user interface is not just about buttons and menus the user interface is about interactions between users and applications. This means that the user interface design is not about how a product looks but rather how it works.

The design of the login page must have at least 4 elements, namely 2 useful text fields for filling in username and password and 1 button which is a confirmation button that the user wants to log in and 1 link that leads to the registration page making it easier for users who do not have an account. Figure 6 shows it.

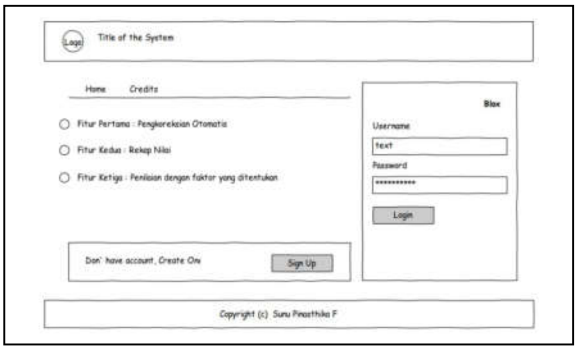

Logo
Title of the System
Home
Credits
○ Fitur Pertama : Pengkoreksian Otomatis
○ Fitur Kedua : Rekap Nilai
○ Fitur Ketiga : Penilaian dengan faktor yang ditentukan
Blox
Username
text
Password
****
Login
Don' have account, Create One
Sign Up
Copyright (c) Sunu Pinasthika F

Figure 6. Sketch of the Login Page Design

The page creating a new question must contain a form that generally has fields filled with attributes of the question itself such as titles, questions, deadlines, answer keys and so on. For more details, see Figure 7.

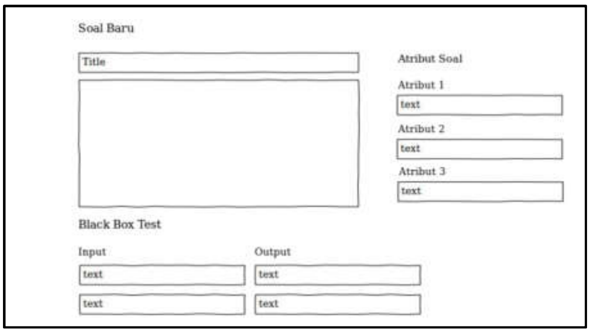
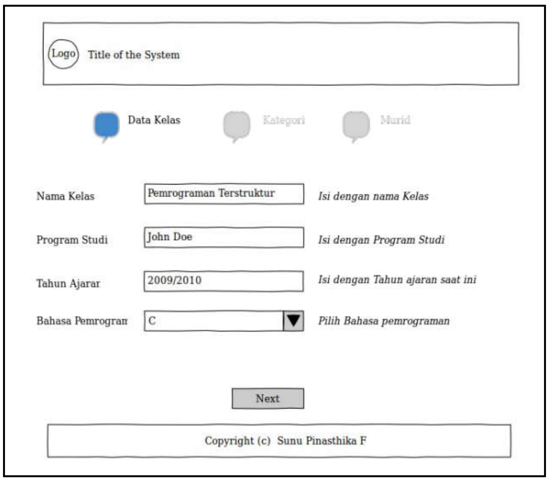


Figure 7. Sketch of Page Design for Problem

This page has many fields that must be filled. This is because when making a class the user must also create a task category and also students who will enter the class. It should be noted in this page user interface design is to overcome the problem of the number of forms available. Most users when opening a page full of input fields that must be filled in, will easily lose focus on certain forms because they are distracted by other forms. Therefore this page divides the form into three groups. So that the user can concentrate on filling out the first group form before filling in the second group form. Besides overcoming the problem of user focus, this method can also make the number of forms that must be filled as if reduced even though it really isn't. This is because each form group will only have a few forms to fill. It is shown on Figure 8.



Nama Kelas: Pemrograman Terstruktur — *Isi dengan nama Kelas*

Program Studi: John Doe — *Isi dengan Program Studi*

Tahun Ajarar: 2009/2010 — *Isi dengan Tahun ajaran saat ini*

Bahasa Pemrogram: C — *Pilih Bahasa pemrograman*

Copyright (c) Sunu Pinasthika F

Figure 8. Sketch of Page Design Making Class

V. IMPLEMENTATION AND TESTING SYSTEM

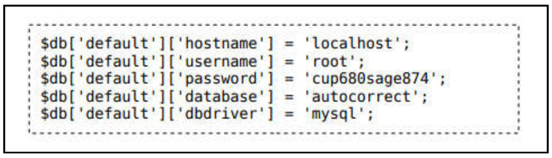
Implementation and testing unit is the phase of implementation design that has been made and does a small test of the existing modules.

A. Setup Environment System

In this study, the installation was carried out online and the OS used is Ubuntu 11.04 Desktop Edition. The programs installed are Apache, PHP, and MySQL.

B. Configuring System Connection with MySQL

This system was developed with the CodeIgniter 2.0.2 framework, so the connection configuration with the database was very easy. Just changing the database php file in the application /config folder to the following as shown on Figure 9.

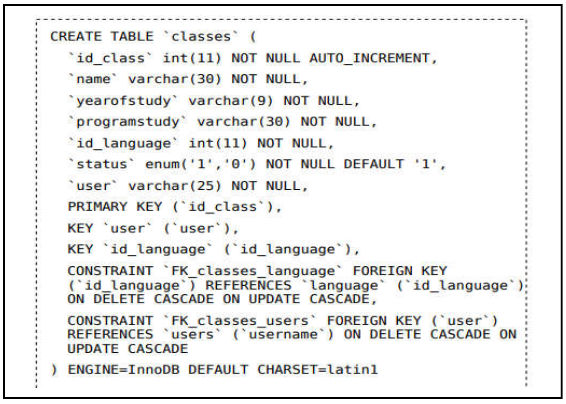


```
$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = 'cup680sage874';
$db['default']['database'] = 'autocorrect';
$db['default']['dbdriver'] = 'mysql';
```

Figure 9. Configure connection with database

C. Database Implementation

The database created will use the InnoDB storage engine. Because the InnoDB engine supports transaction features and foreign keys. This system has 13 tables (menus, users, biodata, language, classes, categories, students, questions, black box, assignment rule, grading rule, uploading log, and assignment grade). Each table has a relationship with another table and there is no relation at all. In Figure 10. Shows an example of the implementation of the command on SQL menus.

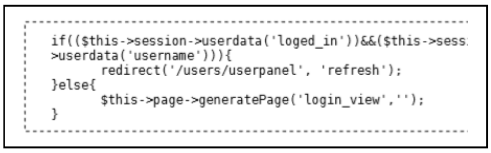


```
CREATE TABLE `classes` (
  `id_class` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(30) NOT NULL,
  `yearofstudy` varchar(9) NOT NULL,
  `programstudy` varchar(30) NOT NULL,
  `id_language` int(11) NOT NULL,
  `status` enum('1','0') NOT NULL DEFAULT '1',
  `user` varchar(25) NOT NULL,
  PRIMARY KEY (`id_class`),
  KEY `user` (`user`),
  KEY `id_language` (`id_language`),
  CONSTRAINT `FK_classes_language` FOREIGN KEY
(`id_language`) REFERENCES `language` (`id_language`)
ON DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT `FK_classes_users` FOREIGN KEY (`user`)
REFERENCES `users` (`username`) ON DELETE CASCADE ON
UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

Figure 6. Command on SQL menus

D. Design Implementaion

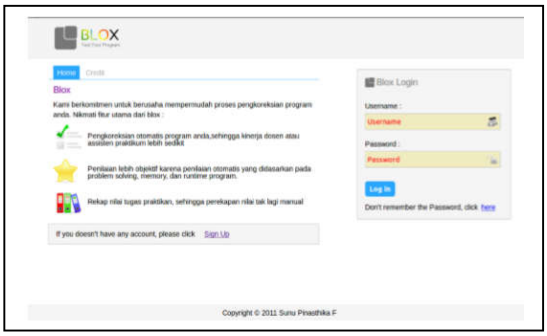
This page can only be accessed by users who are not logged in, if the user has logged in before, the user will be directed to the user-panel if he is a student. When a user accesses the system, the system will access the user controller first and call the method default is the index (). Then the system will check the login status and display the page to the user according to the login status, shown by Figure 11.



```
if(($this->session->userdata('loged_in'))&&($this->sess
>userdata('username'))){
      redirect('/users/userpanel', 'refresh');
}else{
      $this->page->generatePage('login_view','');
}
```

Figure 7. Code to display the login page

Login view is the name of a PHP file that will be displayed screen user. As a view file, login view file the majority contains HTML tags that govern how to display data as in Figure 12.



Blox

Kami berkomitmen untuk berusaha mempermudah proses pengkoreksian program anda. Nikmati fitur utama dari blox :

✓ Pengkoreksian otomatis program anda,sehingga kinerja dosen atau assisten praktikum lebih sedikit

Penilaian lebih objektif karena penilaian otomatis yang didasarkan pada problem solving, memory, dan runtime program.

Rekap nilai tugas praktikan, sehingga perekapan nilai tak lagi manual

If you doesn't have any account, please click Sign Up

Username : Username

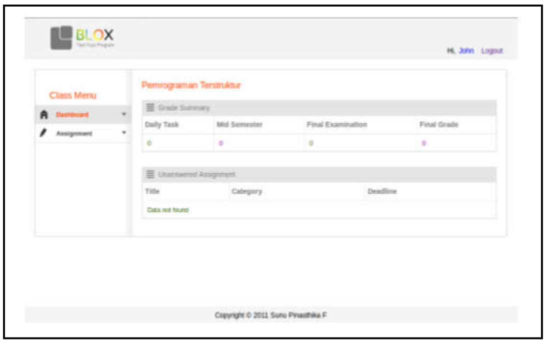
Password : Password

Don't remember the Password, click here

Copyright © 2011 Sunu Pinasthika F

Figure 8. Display of login pages

The dashboard, which is shown on Figure 13, is the page that the user first sees when first entering the user panel. On this page there are brief information about users, namely values and tasks have not been answered correctly.



Pemrograman Terstruktur

Grade Summary

Daily Task	Mid Semester	Final Examination	Final Grade
0	0	0	0

Unanswered Assignment

Title	Category	Deadline
Data not found		

Copyright © 2011 Sunu Pinasthika F

Figure 9. Display of the dashboard page

E. Integration and System Testing

The phase of checking the system as a whole, from the beginning to the end. Checking the correction system of this program will be carried out on the functional side of the system and usability of the system.

System testing will be carried out with black box testing techniques with examiners who are students and former informatics engineering students. After completing the test, the examiner will be asked to fill out a questionnaire. The testing of system functionality will follow the rules set by Table 2.

TABLE 2. FUNCTIONALITY TESTING RULES

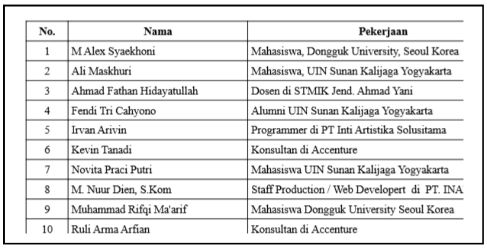
No	Points	Test Result Criteria
1	Login and Logout	The system can handle the user's login process, identify user status, and handle the logout process.
2	Answer Collection The system	Can handle the collection of answers made by the user show detailed questions, and filter files uploaded by the user.
3	Viewing Value System	Can display values to users according to user rights.
4	Profile Update System	Can handle profile updates made by users, either in the form of biodata or username and password.
5	Task Management The system	Can display tasks, and handle the processes carried out by the question maker such as making, editing, or sweeping away existing tasks.
6	Class Management The system	Can display classes, and handle the processes carried out by question makers such as creating, editing, or deleting classes that have been created.
7	Value Recap System	Can display the value of a class, or it can also display values based on certain questions to the question maker.

Usability testing will focus on the system user interface, whether the user interface can make the system easier to understand and use. For usability testing, tests will be carried out on things contained in Table 3.

TABLE 3. SYSTEM USABILITY TESTING TARGETS

No	Points	Explanation
1	Navigation and Buttons	Testing of the effects of laying, the shape and function of the navigation and the buttons in the system.
2	Form and Field	Testing of the effects of laying, form, validating a field.
3	Message / System Response	Testing of messages that arise due to a system that responds to user actions.
4	Layout	Testing of layout design and its effect on the ease of obtaining information.
5	Font	Testing of font selection and its effect on the ease of reading the information displayed.
6	Color	Testing the selection of colors and their effects on the user's comfort in obtaining information.
7	Media Access	Testing the system by opening it using frequently used browser browsers such as Mozilla, Opera, Chrome, and IE

The names of the examiners of this system can be seen in Figure 14. The system examiners are students and former informatics engineering students.



No.	Nama	Pekerjaan
1	M Alex Syaekhoni	Mahasiswa, Dongguk University, Seoul Korea
2	Ali Maskhuri	Mahasiswa, UIN Sunan Kalijaga Yogyakarta
3	Ahmad Fathan Hidayatullah	Dosen di STMIK Jend. Ahmad Yani
4	Fendi Tri Cahyono	Alumni UIN Sunan Kalijaga Yogyakarta
5	Irvan Arivin	Programmer di PT Inti Artistika Solusitama
6	Kevin Tanadi	Konsultan di Accenture
7	Novita Praci Putri	Mahasiswa UIN Sunan Kalijaga Yogyakarta
8	M. Nuur Dien, S.Kom	Staff Production / Web Developert di PT. INA
9	Muhammad Rifqi Ma'arif	Mahasiswa Dongguk University Seoul Korea
10	Ruli Arma Arfian	Konsultan di Accenture

Figure 10. List of Testers Name

VI. RESULTS AND DISCUSSION

From the tests carried out by 11 testers found in Figure 14. The results and also the following discussion:

TABLE 4. FUNCTIONALITY TESTING QUESTIONNAIRE RESULTS

No	Test Result Criteria	Yes	No
1	The system can display the login form when the user has not logged in.	11	
2	The system displays a user-panel page after the user has logged in.	11	
3	The system can handle the logout process properly	11	
4	The system can handle user profile updates, both biodata and username, and password	11	
5	The system displays the menu according to the status of the user who is logged in	10	1
6	The system displays a list of questions that have been made by assistants in the class	10	1
7	The system shows a detailed page of questions and forms to upload answers clearly.	10	1
8	The system can filter user uploaded files	11	
9	The system can upload student answers	11	
10	The system can display error messages for incorrect answers	11	
11	The system can display a successful message for the correct answer.	11	
12	The system can display the download link for the problem solution for questions that exceed the deadline.	9	2

13	The system can display the value of the logged user	11	

TABLE 5. SPECIFICALLY TESTING BY QUESTION MAKERS

No	Test Result Criteria	Yes	No
1	The system can store classes made by users into the database	11	
2	The system displays a page that lists the classes that have been created by the user	11	
3	The system can store questions made by the maker the question along with the answer key and the assessment rules into the database	11	
4	The system can display the final value of students in the class	11	
5	The system can display the values of students in the class based on certain questions	11	
6	The system can handle student value updates manually	11	
7	The system can display the form upload code answer for questions that exceed deadlines.	11	
8	The system can filter uploaded solution files	11	
9	The system can upload the solution file properly	11	
10	The system can save the final value recap into Microsoft Office Excel file	11	

From the data in Tables 4 and 5, we can see that the majority system functionality has been running well. However, some consider that the function has not run as it should, as happened in points 5-8 and 12 in the general testing section. This might happen because the difficulty testers find the function in question (in point 12 of the general testing section) or have difficulty finding comparative parameters to be able to answer the questionnaire questions (in points 5 and 6 of the general testing section).

Obtained from the 13 main functions of the system 8 functions are considered to have been running properly and correctly. Whereas the other 5 functions are considered not running properly and correctly, even though in point 12 of the general testing section more than 81.8% assume that the function has gone well and in points, 5-8 of the general testing section, 90.9% of testers assume the function has been going well.

VII. CONCLUSION

From the data in the tables based on the activities carried out by the author from data collection to the implementation of the program correction system, it can be taken conclusion as follows:

- Correction application and automatic assessment with the system web-based successfully created.
- The lecturer can see the value and recap the final value on the application correction and assessment with this web-based system.
- The system can assess using a rating factor often used in assessing a program.

REFERENCES

[1] Firdausiah, Andi Besse dkk. . Sistem Penilaian Otomatis Jawaban Essay Menggunakan Ontologi Pada Moodle. http://telkomnika.ee.uad.ac.id/n9/files/Vol.6No.3Des08/6.3.12.08.03.pdf. 2008. diakses pada 12 Juni 2011 pukul 19:36

[2] Hutauruk , Sindak. Perancangan Simulasi Koreksi Kesalahan Data Dengan Metode Fec Pada Komputer Berbasis Visual Basic. http://repository.upnyk.ac.id/370/1/A15_PERANCANGAN_SIMULASI_KOREKSI_KESALAHAN_DATA_DENGAN_METODA_FEC_PADA KOMPUTER_BERBASIS_VISUAL_.pdf. 2010. diakses pada 12 Juni 2011 pukul 20:31

[3] Sutisna, Utis dan Julio Adisantoso . Koreksi Ejaan Query Bahasa Indonesia Menggunakan Algoritme Damerau Levenshtein, http://repository.ipb.ac.id/bitstream/handle/123456789/43399/Utis%20Sutisna_abstract.pdf? sequence=2.2010. diakses pada 12 Juni 2011 puku 20:16

[4] Whitten, Jeffery L, Lonnie D. Bentley, dan Kevin C. Dittman, .Metode Desain dan Analisis Sistem. Yogyakarta : Andi, 2004.