WALAILAK JOURNAL
http://wjst.wu.ac.th Applied Sciences

Theses and Capstone Projects Plagiarism Checker using Kolmogorov Complexity Algorithm

Marco Jr. DEL ROSARIO[1,*] and Julius SARENO[2]

[1]*Laguna State Polytechnic University, San Pablo City, Laguna, Philippines*
[2]*Technological University of the Philippines, Manila, Philippines*

([*]Corresponding author's e-mail: macky.delrosario@lspu.edu.ph)

Received: 4 March 2019, Revised: 11 July 2019, Accepted: 10 August 2019

Abstract

In education, students attempt to copy previous works and are relying on prepared solutions available on the Internet in order to meet their requirements. This action leads to plagiarism, which is becoming part of educational institutions' concern to reduce growing academic dishonesty. With regards to the aforementioned issue, this study aims to design and develop a plagiarism checker capable of registering documents, granting access to users, and calculating the similarity between documents. Thus, the software was constructed using HTML, PHP, JavaScript, CSS, and MySQL. The developed system is composed of three main modules; the Document Search which enables users to browse documents, the Document Registration which enables the administrator to add and manage the stored documents, and the document Comparison, which serves as the system plagiarism detection mechanism. The algorithm Normalized Compression Distance was used to measure similarity and the Boyer-Moore Algorithm to highlight the suspected plagiarized document. Moreover, tests were conducted to determine if the system is functioning as expected and to measure the accuracy of the output produced by the system. The developed system was evaluated using the ISO 25010 software quality model in terms of Product Quality and was rated by one hundred respondents. The system obtained a mean of 4.70 which is equivalent to "excellent" in descriptive terms. This validates that the objectives of the study were met and achieved. This further indicates that the system was developed according to its desired functions and requirements.

Keywords: Plagiarism checker, Boyer-Moore algorithm, Kolmogorov complexity algorithm, Normalized compression distance

Introduction

Plagiarism is the act of utilizing someone else's words or thoughts without offering credit to that particular individual. In other words, the deed of taking another person's ideas and passing them off as one's personal idea denotes the concept of "plagiarism". Helgesson and Eriksson expressed that the most common definition of plagiarism is made out of two sections [1]; to suit the work of another person and to take it as one's personal work by not giving appropriate recognition. As agreed by Reynolds [2], Plagiarism is the deed of stealing somebody's concepts or words and claim them as one's own. The rapid increase and availability of digital content and the evolution of the World Wide Web contributes to the practice of cutting and pasting paragraphs into papers and other documents which are lacking appropriate citation or quotation marks. Helgesson and Eriksson [1] concluded that in all academic institutions, plagiarism is a well-acknowledged and rising issue. It was predicted that plagiarism would have a considerable contribution to the number of serious nonconformities from proper research practice. As it is a growing educational concern, plagiarism has now turned into an indispensable character of an individual digital life [3].

Technology, specifically the internet, contributes greatly to sharing resources and information around the world. Having access to a vast amount of information on the internet leads to the increase of threat of unlawful copying and dissemination of someone's intellectual property. In this day and age, people are becoming dependent on the information available on the Internet for prepared solutions. In education, many are relying on this shortcut solution for writing assignments, research papers, and theses. Students attempt to copy previous works in order to pass or meet their requirements. With enhanced access to a vast amount of information and resources, plagiarism is increasing in academic institutions as an emerging form of academic dishonesty.

According to the study conducted by Eya [4], even academic institutions suffer from the effects of Plagiarism. Most students are unaware that their actions of copying and duplicating other person's intellectual property is not just unethical but actually a crime. Students submit their copied research papers to their faculty without knowing that they are plagiarizing. At the same time, most faculty members fall short in screening these research papers. Teachers should consider if these are the students' original ideas or at least proper credits to the persons whose ideas are used have been given. It is proven that course instructors tend to be frustrated with the issues of plagiarism [4].

Meanwhile, according to the study conducted by Badke [5], detecting plagiarism is vital to both the students and teachers. Badke [5] shared that he had a number of faculty members visiting him asking for approaches to examine a suspected plagiarized document. Some of his proposals were to use an online search engine and commercial plagiarism detection service. Conveniently, plagiarism detection tools are available to be used by the students. These plagiarism detection tools check the set of papers and compare them to the documents in the database to determine if they match a part or a portion of them in the database. However, there are times that some documents are compared to inappropriate document in the database of a plagiarism detection software.

With regards to this issue, an information system was developed to act as a repository where projects can be registered, stored, and retrieved for future use. According to an article in rsp.ac.uk website [6], in information technology, repository or digital repository is a mechanism that deals with and keep digital content. As agreed by Kim and Lee [7], having such storage leads to instituting ways for data governance. Henceforth, a necessary component of establishing data governance is devising a data repository. In addition to the perceived solution, a plagiarism detection mechanism was integrated into the system. This mechanism compares documents to the repository of theses and capstone projects to detect any future act of plagiarism.

Materials and methods

The IPO model used as the Conceptual Framework shown in **Figure 1** served as the guideline in developing the project. It includes the requirements needed as well as the processes that need to be undertaken.

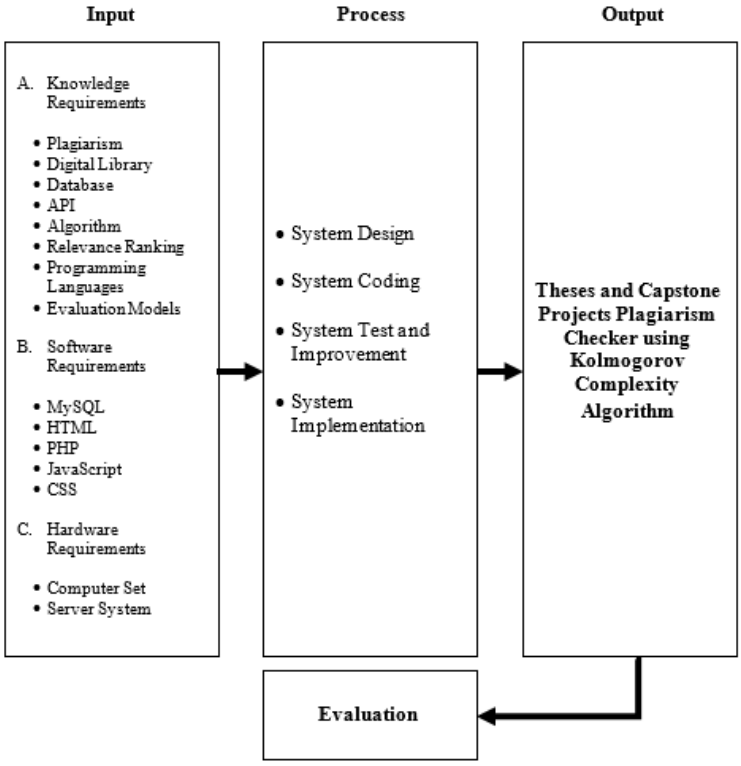

Figure 1 Conceptual Framework of the Project.

Project design

The project entitled "Theses and Capstone Projects Plagiarism Checker using Kolmogorov Complexity Algorithm" is a system that will help to resolve issues/problems related to research document plagiarism in an academic institution.

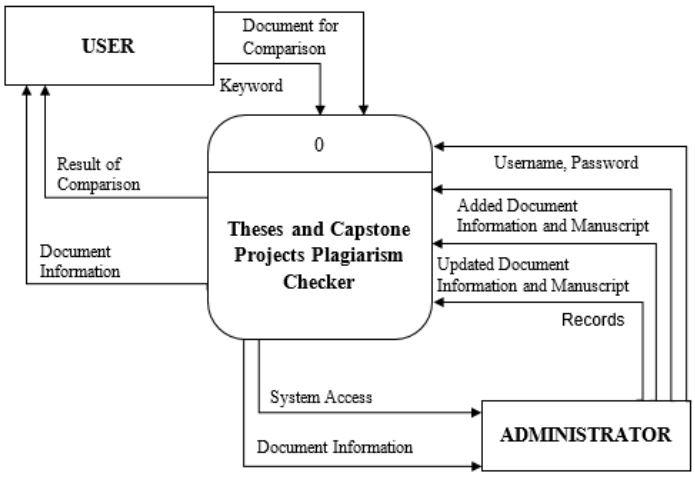

Figure 2 Context Diagram of the Project.

A system design is conceptualized based on the requirements of the system to obtain the objectives of the study. A context diagram of the system was prepared to illustrate the flow of data and the relationship of the system to the external entities, shown in **Figure 2**. The application does all the processing of the data gathered from the external entities to have an output that matches the needs of each entity. The process represents the system application of the project about a digital repository system of capstone projects and thesis with plagiarism detection capability. The lines connecting the entities to the process represent the features that are offered by the system to the entities. A user is given the opportunity to search for a document within the system's database as well as using the application's document comparison feature, while the Administrator is given the privilege to register a document that will be kept in the system's database. The administrator can also access the system database for updating records and documents. The system also allows the administrators to export the stored records and import records from other databases. Additionally, the administrator is capable of adding and updating user or administrator records.

Data flow diagram

The user and the administrator are the expected beneficiaries of the project. To have a valuable output from the system, a user or administrator should interact with the system. These interactions are done by inputting appropriate data and using the functions therein the system.

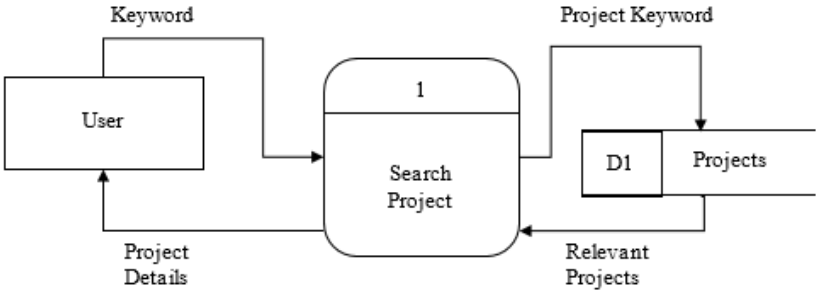

Figure 3 Document Search Module.

First of these functions is the search function or the document search module of the system. This allows the users to search for a project or a document by inputting a search keyword. The keyword will be used to search a document in the system's database. All projects that are relevant to the search keyword will be displayed and can be viewed by the end-users. The information provides the project details. **Figure 3** shows the data flow diagram of the document search module of the system.

Another function that a user can use is the developed system's capability to detect the suspected act of plagiarism on a document. The user can either fill up the text field of the content of a suspected plagiarized document or upload the document file itself.

The system will remove non-textual content and unnecessary text content on the submitted text or uploaded file. Non-textual contents are the images or tables while the unnecessary texts are commonly used words like punctuation marks and articles (a, an, the). After removing the unnecessary words, the processed document content will be compared with the registered documents in the system's database. The Boyer-Moore Algorithm for string matching is used to highlight the suspected plagiarized text [8-11]. According to Drozdek, Boyer-Moore Algorithm works by matching pattern (P) and Text (T) by contrasting them from the direction of right to left of T. It tries to solve the issue of efficiency by picking up speed and by skipping characters in T. Skipping characters in P tends to be more reasonable since the length of P is normally unimportant in comparison to the length of T [8,12-15]. The data flow diagram for the document comparison module is shown in **Figure 4**.

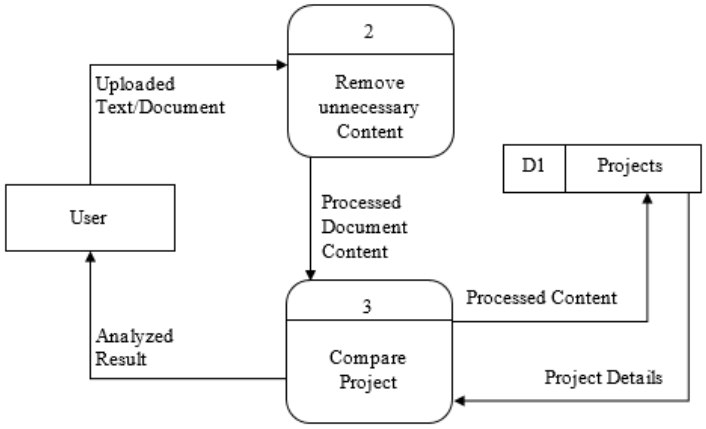

Figure 4 Document Comparison Module.

Figure 5 shows the pseudocode for identifying the similarity between documents. The system will accept the submitted document and this will be compared with the documents from the database which are stored in the string in both old and new array of variable. The array variable will separate every word in the document. In line 8, the system will search for identical words from the variable old in the variable new. Once a match is found, it will be marked as identical and will be included in the matrix variable, which contains both the matched and the mismatched words.

```
4  function boyer($old, $new){
5      $matrix = array();
6      $maxlen = 0;
7      foreach($old as $oindex => $ovalue){
8          $nkeys = array_keys($new, $ovalue);
9          foreach($nkeys as $nindex){
10                 $matrix[$oindex][$nindex] = isset($matrix[$oindex - 1][
                   $nindex - 1]) ?
11                     $matrix[$oindex - 1][$nindex - 1] + 1 : 1;
12             if($matrix[$oindex][$nindex] > $maxlen){
13                 $maxlen = $matrix[$oindex][$nindex];
14                 $omax = $oindex + 1 - $maxlen;
15                 $nmax = $nindex + 1 - $maxlen;
16             }
17         }
18     }
19
20     if($maxlen == 0) return array(array('d'=>$old, 'i'=>$new));
21     return array_merge(
22        boyer(array_slice($old, 0, $omax), array_slice($new, 0, $nmax)),
23        array_slice($new, $nmax, $maxlen),
24        boyer(array_slice($old, $omax + $maxlen), array_slice($new, $nmax
          + $maxlen)));
25 }
```

Figure 5 Pseudocode for Identifying Similarity.

```
27  function htmlBoyer($old, $new){
28      $ret = ''; // will hold the output string
29      $x = 0; // variable counter for number of words
30      $y = ""; // temporary storage
31      $boyer = boyer(preg_split("/[\s]+/", $old), preg_split("/[\s]+/",
        $new)); // boyer-Moore algorithm
32      foreach($boyer as $k){
33          if(is_array($k)){
34              if($x >=5){ // count if the number of words is greater than 5
35                  $ret .= "<mark style='background-color:
                    rgba(52,222,34,0.7);padding-top:3px;padding-bottom:3px;'>"
                    . $y . ' '."</mark>"; // highlight the return words
36              }
37              else{ // if the number of words is less than 5
38                  $ret .= $y.' '; // return words
39              }
40              $x = 0;
41              $y = "";
42              $ret .= (!empty($k['d'])?"".implode(' ',$k['d'])." ":'');
43          }
44          else {
45              $x++;
46              $y .= $k.' ';
47          }
48      }
49      return $ret; // output the words
50  }
```

Figure 6 Pseudocode for Highlighting Similarity.

Figure 5 only displays the code of the function of the system that identifies the similarity between documents. It does not include the control of the system to highlight the minimum number of words. This feature is presented in **Figure 6**, the pseudocode for highlighting similarity. In line 31, the result from identifying the similarity will be stored in variable boyer. Then an iterative process will be conducted to check the highlighted text if it is 5 words or more.

After that process, the system will generate an analyzed result that will be presented to the user. The Kolmogorov Complexity Algorithm using the Normalized Compression Distance (NCD) is used to measure the percentage of similarities between the documents. In the study entitled "Text comparison using data compression", the algorithm used to detect similarity between documents is the Kolmogorov Complexity [16]. Kolmogorov Complexity is one of the perfect measures for computation of the similarity of two strings in defined alphabet. However, the algorithm is incomputable [17,18]. Wortel [17], who also developed a system that is capable of plagiarism detection, applied the Normalized Compression Distance as the measure. Wortel's methods were centered on the theoretical concept of Kolmogorov Complexity [17]. In addition, Platos determined that approximation of Kolmogorov Complexity can be made by using compression. The idea is that a general-purpose compressor detects as much regularity in wide range of files as reasonably as possible. Thus, compressions may be thought of as approximations of the Kolmogorov Complexity, hence, NCD was used. NCD formula is illustrated in **Figure 7**.

$$NCD(x,y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}}$$

Figure 7 Normalized Compression Distance.

As shown in **Figure 8**, the variables *sx* and *sy* will contain the string of the submitted document and the document from the system database, respectively. Each document will be compressed, as seen in line 7 and line 10. At the same time, variables *min* and *max* were initialized using the value of *x* and *y*, respectively. The combination of both strings will also be compressed and stored in variable *xy*.

```
3   // sx,sy = strings to compare.
4   function ncd_new($sx, $sy) {
5
6       // compress submitted document
7       $x = $min = strlen(gzcompress($sx));
8
9       // compress document from the database
10      $y = $max = strlen(gzcompress($sy));
11
12      //concatenate string
13      $xy= strlen(gzcompress($sx.$sy));
14
15      // if x is shorter than y, swap min/max
16      if ($x>$y) {
17          $min = $y;
18          $max = $x;
19      }
20
21      // NCD Formula.
22      $res = ($xy-$min)/$max;
23
24      // transform result into percentage
25      return 100*round($res,2);
26  }
```

Figure 8 Pseudocode for Similarity Percentage.

The condition in line 16 will determine which string is shorter. If the content of *y* is shorter than *x* the value of *min* and *max* will be swapped, and the value of *min* and *max* is retained. Then, the formula for getting the NCD will be performed. The combination of *x* and *y* will be deducted by the value of the shorter string divided by the value of the longer string. Then, the result will be multiplied by 100 to get the percentage of similarity between documents using the Normalized Compression Distance.

An additional entity existing in the diagram is the administrator. An administrator is an important actor in the system because he is the person who can manipulate the records stored in the system database. The administrator is capable of adding documents by registering it through the Document Registration module. Furthermore, an administrator can update the documents or the information of a thesis or capstone project stored in the system's database. The flow of data in the Document Registration module is illustrated in **Figure 9**. The Administrator submits a completed registration form. The form is supplied with the information of a project to be registered. The system processes the details and will be saved to the system's database. Then the system will notify the administrator whether the project has been registered to the system's database or not.

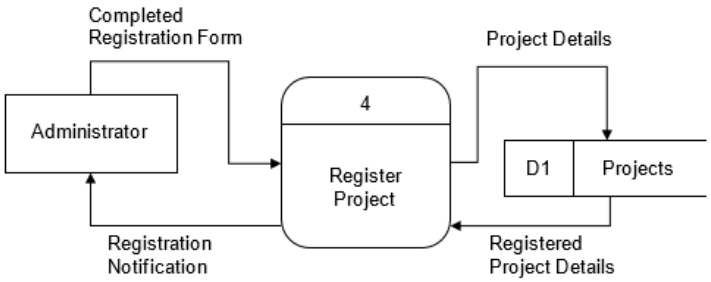

Figure 9 Document Registration Module.

Use case diagram

It shows the relationships, in a number of ways, that a user may interact with the system. In addition, there are different types of users shown in the diagram wherein users are referred to as Actors.

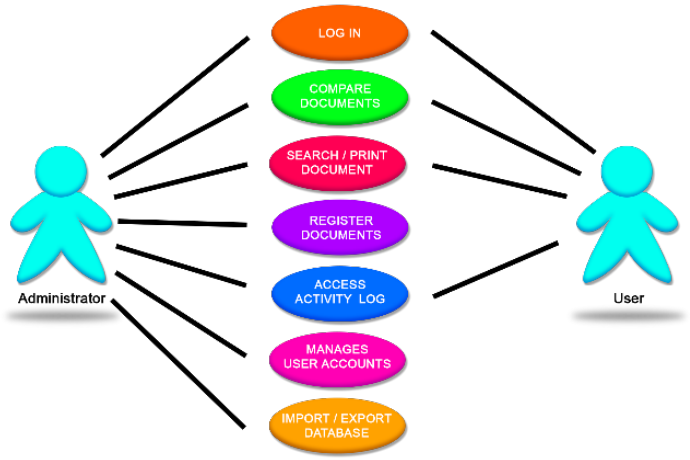

Figure 10 Use Case Diagram.

First of these actors is the Administrator. As depicted in **Figure 10**, the administrator is the person who manages and controls all the transactions of the system. The Administrator is the only actor who can access all three main modules of the system. The system administrator is the most privileged actor among other actors. The overall management of the system is given to the Administrators. They are the ones responsible for registering documents in the system and for managing the accounts of users in the system. The Administrators have full control over the modifications, updates, and database management of the system.

The expected administrator of the system is the Director and Chairperson of the Research and Development department from the different campuses of the implementing institution. Also, the College Research Coordinators are expected to act as administrators of the system.

As depicted in **Figure 10**, another actor is identified in the use case diagram. These actors are called the Users. The students, faculty members, and other researchers fall into this type of actor. The users are capable of accessing the system by logging in. Moreover, the users are capable of using the document

comparison and documents search features of the system. As included in the document search, users are able to view and print the document details and information.

User interface design

In the development of this project, the user interface design was designed as shown in **Figure 11**. It describes the layout of the system where it can interact with the users. The design was made to ensure that the users can easily access, comprehend and facilitate actions in using the elements that are visible in the design.

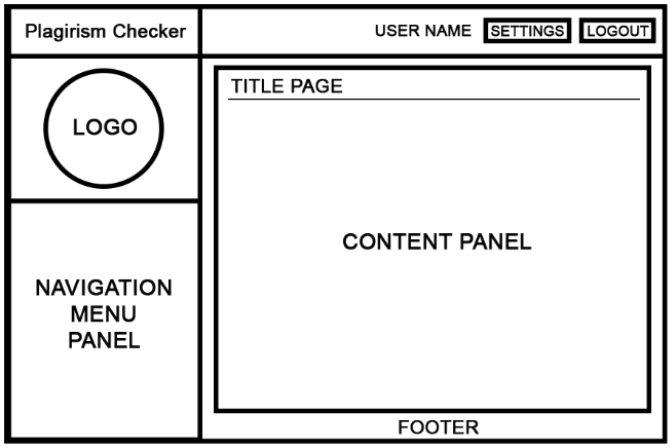

Figure 11 System General User Interface Design.

The system follows a general user interface design to be consistent and be predictable which can help the users in navigating the system easily. To elaborate, the interface was divided into four divisions. These are the header, left side panel, content panel, and the footer. As seen in **Figure 11**, there is a header panel wherein the acronym of the project is visible as well as the user's name and logout button. The left side panel of the system interface consists of the logo of the developed system and can be followed by the navigation menu panel. The navigational menu contains the three main modules of the system. The content panel contains the title of the displayed page as well as the information that goes on the page or feature, and the footer division which includes the title of the project and year of development.

Results and discussion

Test results

The developed system underwent the process of testing to ensure its quality in accordance with functionality and accuracy. The types of testing used were Functionality Test and Accuracy Test.

Functionality test

This type of testing checks if all functions of the system are working and are giving a correct output. In this testing, multiple test scenarios and test cases were prepared to be performed. **Table 1** shows the list of all test scenarios performed and the number of test cases for each scenario. Another data presented in the table is the remark indicating if the tests passed or failed.

Table 1 Summary of Result of Functionality Test.

FUNCTIONALITY	NUMBER OF TEST CASES	REMARKS
Login / Logout	5	5 Passed, 0 Failed
Document Registration	4	4 Passed, 0 Failed
Document Editor	4	4 Passed, 0 Failed
Document Search	12	12 Passed, 0 Failed
Document Comparison	4	4 Passed, 0 Failed
Import Record	4	4 Passed, 0 Failed
Export Record	2	2 Passed, 0 Failed
User Management / User Logs	9	9 Passed, 0 Failed
System Settings	5	5 Passed, 0 Failed

Legend:
Passed - indicates that the actual result of the test meets the expected result.
Failed - indicates that the actual result of the test was different from the expected result.

Table 1 describes that there were 8 test scenarios prepared in the conduct of the testing. Correspondingly, a total of 49 test cases were performed to determine the correctness and performance of the system. The purpose of each test case is to verify if every function that the system performs is conforming to the goals and requirements of the system. The last column in the table title "Remarks" indicates the number of passed and failed test cases. In this discussion, the testing that will be discussed are the three main modules of the system, namely, the document registration, document search, and document comparison.

Table 2 Document Registration Test Case.

TEST SCENARIO	Document Registration	**REMARKS**	PASSED
TEST CASE	Enter Valid Information and Select a File with valid File Format		
TEST STEPS			
1. Click Document Registration tab 2. Enter Title 3. Enter Author Name 4. Select an Institution 5. Select Focus (Program/Course) 6. Select Year 7. Choose a document to upload 8. Click Register			
EXPECTED RESULT		**ACTUAL RESULT**	
Document Registration must be successful		Document Registration successful	
POSTCONDITION			
Time and Date of Registration is stored in the Database. Display that the Document registration was successful			

As seen in **Table 2**, the document registration test case has 8 test steps. These steps are Click the document registration tab, then enter the document title and author name, followed by selecting an institution, project focus and year, select the document file to be uploaded then click the Register button. This test case expects a valid information and valid document with correct file format as inputs. A result

of the successful registration of a document is anticipated. After the execution of the test step, the system was able to successfully register the submitted document. Therefore, the test reflects that the function was working properly and that the system passed the test.

Table 3 Document Search Test Case.

TEST SCENARIO	Document Search	**REMARKS**	PASSED
TEST CASE	Enter Valid Search Key (with Relevant Document)		
TEST STEPS			
1. Click Document Search tab 2. Click Anywhere tab 3. Enter Search Keyword 4. Click Search Button			
EXPECTED RESULT		**ACTUAL RESULT**	
Must search relevant document in the Database		Searched relevant document in the Database	
POSTCONDITION			
Time and Date of Search is stored in the Database. Display all relevant documents.			

Next is the Document Search Module. There are 12 test cases performed and 4 test cases for each selection in the Document Search. **Table 3** represents the test case used in the Document Search Module. The expected result of the test is that the system must search a document in the database that is relevant to the keyword entered by the user. There are 4 steps in order to perform the test. Click the document search tab, then click anywhere tab, enter a search key and then click the search button. After performing the steps, the result of testing is that the system was able to search relevant document from the database. A postcondition of storing the date and time of searching was also accomplished. This means that the testing passed since it was able to meet the expected result.

Table 4 Document Comparison using File Upload Test Case.

TEST SCENARIO	Document Comparison (File Upload)	**REMARKS**	PASSED
TEST CASE	Select a File with a valid File Format		
TEST STEPS			
1. Click Document Comparison tab 2. Select a File 3. Click Compare Button			
EXPECTED RESULT		**ACTUAL RESULT**	
Document comparison process must be performed successfully		Document comparison process was performed successfully	
POSTCONDITION			
Time and Date of Comparison is stored in the Database. Display document comparison result.			

The test scenario regarding document comparison is divided into two, that is, document comparison using the file upload and the document comparison by means of entering text. The test case of Document

Comparison using file upload is shown in **Table 4**. Successful execution of the document comparison process was the anticipated result of the testing. The following steps were performed in order to determine if the system meets the desired outcome. Click the document comparison tab, select a file to be uploaded then click the compare button. After conducting the steps, an actual outcome of successfully performing the document comparison arrived. This means that the test in using document comparison by uploading a file as input passed. In addition, a postcondition of storing the time and date of comparison was successfully performed as well as displaying the document comparison result.

Table 5 Document Comparison using Text Input Test Case.

TEST SCENARIO	Document Comparison (Text Input)	**REMARKS**	PASSED
TEST CASE	Submit a Text		
TEST STEPS			
1. Click Document Comparison tab 2. Input Text 3. Click Compare Button			
EXPECTED RESULT		**ACTUAL RESULT**	
Document comparison process must be performed successfully		Document comparison process was performed successfully	
POSTCONDITION			
Time and Date of Comparison is stored in the Database. Display document comparison result.			

On the other hand, using the document comparison by means of entering text as input had a different test case. As shown in **Table 5**, the steps taken in the testing was that the document comparison tab was clicked, a text was inputted then the compare button was clicked. In performing these steps, the result of successfully performing the comparison process was expected. As it turns out, after following the step, the system successfully executed the comparison process by the use of the inputted text. For this reason, a passing remark was given to the test case.

Accuracy test

This type of testing assessed the system on the basis of the correctness of the outputs. Series of tests were conducted to measure the accuracy of the outputs of the developed system. The purpose of these series of tests was to measure the system capability in detecting similarities between documents.

Table 6 Accuracy Test Document Distribution.

Document Name	Remarks
A	Original Document
B	Original Document
C	Original Document
Stripped(A)	Removed unnecessary words in A
Stripped(B)	Removed unnecessary words in B
Stripped(C)	Removed unnecessary words in C
AB	Combined A and B
AC	Combined A and C
BC	Combined B and C
D	Copied from A
E	Copied from B
F	Copied from C

For this particular test, 12 documents were prepared. These documents were used as specimen samples in the four sets of tests. **Table 6** shows the documents as well as the characteristics of each document. There were 3 documents per set of tests. The first set of documents are the original documents where the other documents will be compared upon. The others are the following: documents stripped with unnecessary words, a combination of original documents, and documents that are copied from the original documents.

The first set of tests conducted in terms of accuracy was the system's capability to tokenize. According to Tatman, tokenization is one of the common tasks in Natural Language Processing (NLP) [19]. It is a process of breaking a string into tokens which in dawns small structure or units that can be used in tokenization. As agreed by Habert et. al., tokenization can be defined as the task of splitting a stream of characters into words. Moreover, this process can be associated with preliminary "cleaning procedures" for example is removing unusable labels, excluding "non-textual" items and eliminating parts that do not reside in natural languages [20]. These series of tests determine if the system was able to accurately tokenize the inputted documents considering the process of removing the unnecessary texts.

Table 7 Accuracy Test Set 1.

Submitted Document	Expected Result *	Actual Result *	Percentage
A	80	80	100 %
B	126	126	100 %
C	94	94	100 %

* - refers to the number of tokens.

As shown in **Table 7**, the original documents were submitted in the system. After the submission, the number of tokens recognized by the system were compared to the expected number of tokens per document presented in the second column. The expected number of tokens were manually counted. The results show that the system was able to produce the same number of tokens as the expected result. This indicates that the system accurately tokenized the submitted documents.

The second set of tests was performed to identify the system's capability to recognize identical documents. Three separate tests were administered. Documents A, B, and C were matched against themselves in every test. All the considered original documents were initially registered and then

submitted for comparison. Thus, a similarity result of 100 % was expected. As shown in **Table 8**, the average of the actual similarity results for all three tests is 100 %. This means that the system was able to detect identical documents.

Table 8 Accuracy Test Set 2.

Registered Document	Compared Document	Expected Result*	Actual Result*
A	A	100 %	100 %
B	B	100 %	100 %
C	C	100 %	100 %

* - refers to the similarity percentage generated by the system.

Identifying the capability of the system of recognizing the similarity between the registered document and the stripped form of the document was the purpose of the third set of tests. This test was conducted in order to determine if the system produces an accurate output as well as if the system was able to remove unnecessary words in the uploaded documents. Another three independent tests were conducted. Document A will be compared against the stripped document A. The same goes with documents B and C to be compared against stripped documents B and C, respectively. Every stripped document was then submitted for comparison. As shown in **Table 9**, the average similarity result for all three tests is 100 %. This indicates that the system was able to detect the similarity between the original form of registered document and the stripped form of the document.

Table 9 Accuracy Test Set 3.

Registered Document	Compared Document	Expected Result*	Actual Result*
A	Stripped(A)	100 %	100 %
B	Stripped(B)	100 %	100 %
C	Stripped(C)	100 %	100 %

* - refers to the similarity percentage of the system.

The purpose of the fourth set of tests is to determine if the system was able to identify a document that was combined with the original document. Six independent tests were conducted, comparing the combined documents of A and B, A and C, and B and C against the non-combined documents in each test. Each combined document is submitted for comparison. The result of comparing the combined documents may be easily assumed to be 50 %. But, because there is no known exact number of words or exact number or plagiarized parts in the document, 50 % would not be the expected result of this test. A result falling between the approximate range of 30 to 70 % was expected instead of 50 %.

Table 10 Accuracy Test Set 4.

Registered Document	Compared Document	Expected Result	Actual Result
A	AB	30 % - 70 %	45 %
A	AC	30 % - 70 %	53 %
B	AB	30 % - 70 %	65 %
B	BC	30 % - 70 %	63 %
C	BC	30 % - 70 %	47 %
C	AC	30 % - 70 %	58 %

* - refers to the similarity percentage of the system.

Table 10 displays the result of the test in determining the accurateness of the system if two documents are combined. The average similarity result of all three tests falls between the range of 30 - 70 %. This means that the system was able to detect similarity between the combined documents against the registered original documents.

The last set of tests aimed to identify if the system may possibly recognize actual copying which is the necessary requirement of the system. Three independent tests were conducted, comparing the document A against D, B against E, and C against F. All documents considered copied from the original document are submitted for comparison. The expected result was assumed to be greater than 20 %. The expected result may vary depending on the value of the similarity threshold for each document. The higher the number of suspected plagiarized words or phrases, the higher the similarity threshold is expected. As shown in **Table 11**, the similarity result for all three tests is greater than 20 % (assumed similarity threshold). This means that the system was able to detect actual copying activities in the submitted document.

Table 11 Accuracy Test Set 5.

Registered Document	Compared Document	Expected Result*	Actual Result*
A	D	> 20 %	74 %
B	E	> 20 %	44 %
C	F	> 20 %	59 %

* - refers to the similarity percentage of the system.

Series of tests have been described to demonstrate the accuracy of the system. Even though the results of the tests are particular with respect to the testing set of documents, it explained the general behavior of the system. It shows that the system was able to produce an acceptable and accurate measurement of similarity between the same documents and stripped documents. Correspondingly, in terms of the documents that are combined and copied, it is apparent that the system was able to compute and identify the similarity between documents.

Evaluation result

The performance of the project was evaluated using the ISO 25010 software quality model in terms of the Product Quality composition. There were 100 respondents who evaluated the developed project. The respondents consisted of IT practitioners, research personnel and students from the Laguna State Polytechnic University in the Philippines. All the respondents' ratings were consolidated and computed to get its quantitative and qualitative interpretation.

Table 12 Results of Respondents' Ratings of the System.

Criteria	Mean	Qualitative Interpretation
Functional Suitability	4.74	Excellent
Performance Efficiency	4.73	Excellent
Compatibility	4.72	Excellent
Usability	4.67	Excellent
Reliability	4.62	Excellent
Security	4.70	Excellent
Maintainability	4.71	Excellent
Portability	4.70	Excellent
Overall Mean	**4.70**	**Excellent**

Table 12 summarizes the evaluation results from the respondents' ratings showing the mean per criterion and the corresponding qualitative interpretation. The table also presents the overall mean by getting the average of all the means of the eight criteria.

The respondents' ratings regarding the Functional Suitability of the system obtained a mean of 4.74 which is equivalent to "Excellent" in qualitative interpretation. This indicates that the set of functions covers all the specified tasks and user objectives. Also, the functions provide the correct results with the needed degree of precision. Lastly, the functions facilitate the accomplishment of specified tasks and objectives.

In terms of Performance Efficiency, the system attained a mean of 4.73 which is equivalent to "Excellent" in the descriptive term. This indicates that the system is performing its functions to meet the required response, processing times and throughput rates. Similarly, the system meets the required types of resources to be used as well as meeting the maximum limit of the system.

In response to the system evaluation regarding Compatibility, it attains an average score of 4.72 which also means that the system is "Excellent" in this criterion. This points out that the system can perform its required functions efficiently while sharing a common environment and resources with other products, without inconvenient effect on other product. And, products or components can exchange information and use the information that has been exchanged.

The respondents' ratings regarding the Usability of the system obtained a mean of 4.67 which is interpreted as "Excellent". This means that the users can recognize whether a product or system is appropriate for their needs. The system enables the user to learn how to use it with effectiveness, efficiency in emergency situations. A product or system is easy to operate, control and appropriate to use. A product or system protects users against errors. The user interface enables pleasing and satisfying interaction for the user.

In terms of Reliability, the system achieves a mean score of 4.62 which means that the system is "Excellent" in the descriptive term. This points out that the system meets the needs for reliability under normal operation, the system is operational and accessible when required for use, the system operates as intended despite the presence of hardware or software faults, and it can recover the data directly affected and re-establish the desired state of the system whenever there is an interruption or a failure.

In terms of Security, the system attained a mean of 4.70 which is equivalent to "Excellent" in the descriptive term. This indicates that the system ensures that data are accessible only to those authorized to have access, it prevents unauthorized access to or modification of data and stores records which prove that an action has taken place. Also, the system was able to trace and identify the action of a user.

The respondents' ratings regarding the Maintainability of the system obtained a mean of 4.72 which is interpreted as "Excellent". This means that the system is composed of discrete components such that a change to one component has minimal impact on other components. The asset can be used in more than

one system. It is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified. The system can be effectively and efficiently modified without introducing defects or degrading existing product quality and test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.

In terms of Portability, the system attained a mean of 4.70 which is equivalent to "Excellent" in the descriptive term. This indicates that the system can effectively and efficiently be adapted for a different platform and can be successfully installed and/or uninstalled in a specified environment. It can also replace another specified software product for the same purpose in the same environment.

Moreover, the criterion Functional Suitability obtained the highest mean, while reliability got the lowest mean but still falls within the range of the scale value of “Excellent”. The overall mean generated for all the criteria contained in ISO 25010 software quality model in terms of the Product Quality composition as evaluation instrument yielded an average of 4.70 which validates that the system has attained its anticipated functions according to the requirements. This also indicates that the system is “Excellent”.

Description of the software developed

This study led to the development of the project which is an information system that functions as a digital repository of researches, theses and capstone projects’ documents for Laguna State Polytechnic University in the Philippines with plagiarism detection capability (**Figure 12a**). The project caters the user the basic features and functionalities of an information system allowing them to register, update, view, search and maintain the document records in the database using the document registration and document search module of the system.

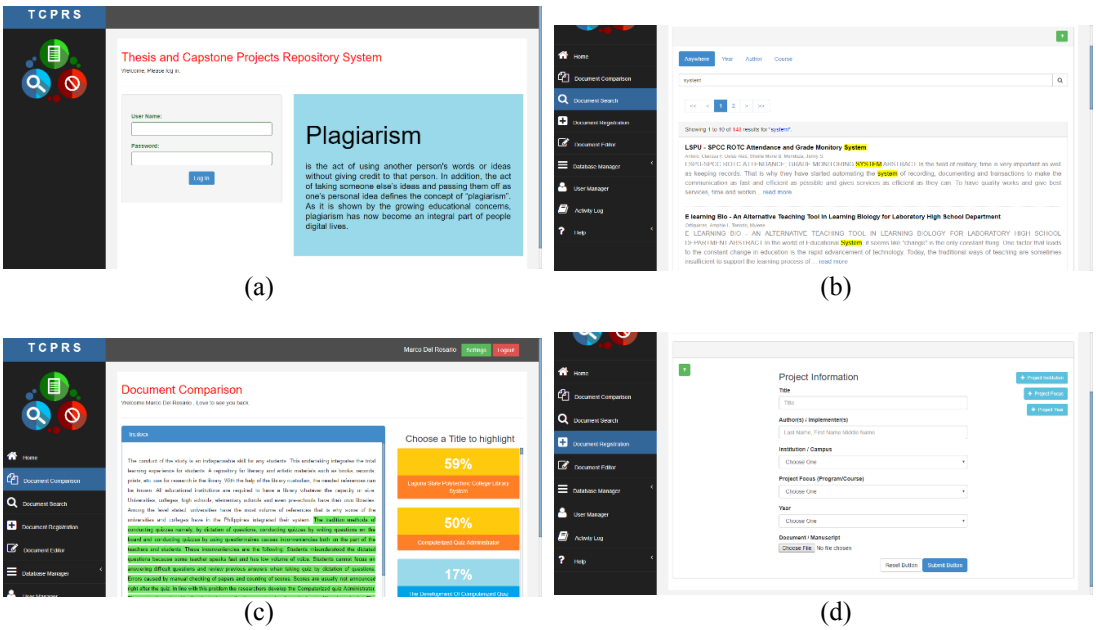

Figure 12 (a) Home Screen (b) Document Search (c) Document Comparison (d) Document Registration.

The document registration allows the administrator to increase the number of documents that can be found on the system (**Figure 12d**). It allows an administrator to add documents to the database as well as updating the information therein. On the other hand, the document search module allows the users to browse a document that was stored in the system database (**Figure 12b**). The system further provides the user selections in searching for a document. A user can perform this function by means of searching the author name, year, program/course, title or parts of its content. The information that can be browsed in the document search module is limited to the project title, author/s, year, project focus, database, and content.

Another feature offered by the system is its capability to detect plagiarized documents using the document comparison module (**Figure 12c**). This feature prompts the user to submit a document or a text which will be compared against the documents that were stored in the system. After comparing, a user will be given a similarity result. This result indicates the percentage of similarity measured using the Normalized Compression Distance algorithm. Furthermore, a user can view the similarity between documents by prompting the system to highlight the words, phrases or portions of the document that are found identical. However, the system was not able to compare document images, tables, and other non-textual content.

In addition, the system has the provision to allow importing and exporting records from the list of stored documents. The system is also capable of showing the log that records user activity. The system also enables the administrator to manage user information. It permits to add a new user, add another administrator, and edit their information as well as reset a password. Though, Laguna State Polytechnic University as the implementing institution, the developed system applies its features only to the documents available in the institution.

Conclusions

The following findings of the study were established in relation to the results of the conducted testing and evaluation with respect to the system functionality, accuracy and performance. The developed system was created according to the intended design, algorithm, and functionalities. The system is capable of offering functions and features for managing the thesis and capstone project documents at Laguna State Polytechnic University. The system was created in order to act as a digital repository system that is capable of providing the basic information system features such as registering, updating, viewing, searching and maintaining records preserved in the system's database. The system was made up of different modules. First is the document search which allows any user to browse for a document stored in the system. Second is the document registration module which allows the administrator to add documents. The document comparison module is the feature where documents are subjected to similarity detection to determine any possible plagiarism activity. The document comparison module was developed using Kolmogorov Complexity and Boyer-Moore Algorithms. Furthermore, the system was developed to give the administrator the authority to import and export document records. It also allows the administrator to revise the information on the registered document, to manage user account, and to monitor activities within the system. It was successfully designed using the following programming languages and software development tools such as PHP, JavaScript, CSS, HTML, and MySQL.

Moreover, different tests were conducted to identify the system's functionality and accuracy. The functions and features of the system were assessed using 49 test cases covering all the functions present in the system. After performing the test, the system was able to meet the expectations and considered functional. Likewise, in terms of accuracy, series of tests have been performed. Tests proved that the system was able to produce an acceptable and accurate measurement and was able to identify the similarity between documents in order to detect an act of plagiarism. The system was evaluated by 100 respondents composed of IT Practitioners, Research Personnel and students from Laguna State Polytechnic University. The performance of the system was evaluated using the ISO 25010 in terms of product quality composition which attained an overall mean of 4.70 equivalent to "Excellent" in qualitative term. This further proves that the developed system performs the expected functions, features, and requirements successfully and will be beneficial to an academic institution.

Acknowledgements

This study is a Thesis / Research Project submitted to the faculty of the College of Industrial Technology Graduate School in the Technological University of the Philippines, Manila, Philippines. The authors are indebted to the College of Computer Studies, to the respondents of the study, and to the student and faculty of the Laguna State Polytechnic University, San Pablo City, Laguna, Philippines.

References

[1] G Helgesson and S Eriksson. Plagiarism in research. *Med. Health Care Philos.* 2015; **18**, 91-101.
[2] G Reynolds. *Ethics in Information Technology*. Nelson Education, 2011.
[3] MJD Rosario. Student paper comparison system using Kolmogorov complexity and diff algorithm. *Thai J. Phys.* 2019; **36**, 9-27.
[4] J Eya. 2007, Development of Plagiarism Detector for the Family of C Program Source Codes. MIT Research Project. Technological University of the Philippines, Manila, Philippines.
[5] W Badke. Training plagiarism detectives: The law and order approach. *Online* 2007; **31**, 50-2.
[6] Repositories Support Project, Available at: http://www.rsp.ac.uk/start/before-you-start/what-is-a-repository, accessed July 2016.
[7] S Kim and W Lee. Global data repository status and analysis: Based on Korea, China and Japan. *Library Hi Tech*. 2014; **32**, 706-22.
[8] A Drozdek. *Data Structures and Algorithms in Java*. 2nd ed. Boston, Massachusetts: Course Technology, Thomson Learning, USA, 2010.
[9] G Barnett and LD Tongo. *Data Structures and Algorithms: Annotated Reference with Examples*. 1st ed. DotNetSlackers, 2007.
[10] K Mehlhorn and P Sanders. *Algorithms and Data Structures: The Basic Toolbox*. Springer Science & Business Media, 2008.
[11] WM Allen. *Data Structures and Algorithm Analysis in C++*. Pearson Education India, 2007.
[12] M Khairullah. Enhancing worst sorting algorithms. *Int. J. Adv. Sci. Tech.* 2013; **56**, 13-26.
[13] M Alam and A Chugh. Sorting algorithm: An empirical analysis. *Int. J. Eng. Sci. Innovat. Tech.* 2014; **3**, 118-26.
[14] Searching Algorithm from IDC Technologies, Available at: https://www.idc-online.com/technical_references/pdfs/.../Searching_ Algorithms.pdf, accessed January 2017.
[15] J Hopkins. Whiting School of Engineering, Boyer-Moore from Langmead Lab, Available at: https://www.langmead-lab.org/teaching-materials, accessed February 2016.
[16] J Platos, M Prilepok and V Snasel. *Text Comparison using Data Compression*. VSB-Technical University of Ostrava, 2013.
[17] LL Wortel. *Plagiarism Detection using the NCD*. Universiteit de Amsterdam, 2005.
[18] PMB Vitányi, FJ Balbach, RL Cilibrasi and M Li. *Normalized Information Distance*. Springer, Boston, MA, 2009, p. 45-82.
[19] R Tatman. Data Science 101 (Getting Started in NLP): Tokenization Tutorial, Available at: http://blog.kaggle.com/2017/08/25/data-science-101-getting-started-in-nlp-tokenization-tutorial, accessed July 2019.
[20] N Habert, A Gilles, M Adda-Decker, PBD Marëuil, S Ferrari, O Ferret, G Illouz and P Paroubek. Towards tokenization evaluation. *In*: Proceedings of the 1st International Conference on Language Resources and Evaluation, 1998, p. 427-31.