International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 6 527 - 534

Implementation of Excel Parser for Web Integration of Digital Evaluation System of CSIBER

Dr. Poornima G. Naik, Mrs. S. S. Jamsandekar, Mrs. K. S. Mahajan, Dr. R. S. Kamath, Mr. M. B. Patil
Department of Computer Studies
Chh. Shahu Institute of Business Education and Research
Kolhapur, India

***Abstract*—** Digitization and advancements in web technology have facilitated quick access to accurate data irrespective of its location and time zone. Time span becomes crucial in such cases since the data soon becomes stale. Further, web hosts data in variety of formats and sizes. Integrity, authenticity and confidentiality play a key role in data access. The main motto of web data accessibility is that data should be made available only to the person who is authorized to access it. CSIBER has recently installed Digital Evaluation System (DES) for automating the routine tasks of examination department. However, currently the software lacks in web interface. To address this issue, in the current paper, the authors have designed and developed an Excel parser in PHP which acts as an interface between DES and CSIBER's official website by enabling examinees a quick access to online mark statement. The parser is flexible enough to take care of minor differences in structures of Excel spreadsheets. The deployment folder structure is presented and different cases are explored.

Keywords- *Application Folder Structure, Digital Evaluation, Excel parser, web interface, algorithm, PHP*

*****

I. INTRODUCTION

In the era of digitization in order to facilitate easy accessibility to remote resources, information is made available on the web. The data is usually maintained in variety of databases at the server-side which is fetched and displayed on end user request. In contrast to this most often in organizations the internal data is maintained in Excel sheet in order to keep track of requisite data for ease in accessibility of required data. Most of the websites make such data publically available by simply allowing download facility. However if the file contain sensitive data it is highly desirable if the user is given access to the data for which he is authorized.

However such a scenario demands parsing of excel file before appropriate data is made available to the end user, and it would be much preferred if the data could be searched through a simple web interface. This is not a particularly difficult job, but requires a fair amount of work to do at the server side, typically it needs a database to be set up and manually import the Excel sheet data into a SQL database and make it available through a web application. Changes in either the spreadsheet or the database have to be done manually and had to be kept in sync. To make the input and output web pages look aesthetically pleasing, more time is required to be spent on web design. The whole task is extremely time consuming and cumbersome.

Rather than setting up a database and syncing it manually it would be better if one could connect the web application to excel sheet. This requires parsing the excel structure to retrieve the data of interest and publish it on web. Problem arises when data is maintained in different excel structures and formats. A dynamic algorithm is needed in order to identify and parse the file structures. This has encouraged the authors to develop a hassle free algorithm that parses the excel data and make it available on the web.

The algorithm is written using PHP Script since PHP is one of the most popular open source server-side scripting language which is attuned to different data formats.

Digital Evaluation System

Recently CSIBER has procured and set up a Digital Evaluation ERP System (DES) from CompServe Tech.Ltd, Kolhapur in association with Department of Computer Studies and Examination Department, CSIBER. The system has been successfully customized and implemented for different courses run by CSIBER since 2016. The DES, under which all answer scripts are scanned using high speed scanner (100 pages/min) and stored at the server side in the pdf format delimitated by unique bar-code corresponding to each answer script. Authorized evaluators get access to the uploaded answer scripts for evaluation and the marks are entered on the system and calculated automatically, thus eliminating any possibility of tabulation errors. In the pre-digitized era there were chances of error in coding and decoding, tabulation and compilation of marks and results. This system is unique of its kind and has less than 0.01 per cent margin of error unlike the conventional system where the margin of error is high. Through it, the examination section is eliminating mistakes to a large extent and completely eliminating the need for re-totaling. Thus the aforesaid system is "transparent, tamper-proof, efficient, cost-effective teacher and evaluator friendly." In future, the examination section is planning to incorporate web-based examination application entry.

In the current work, the authors have designed and developed an Excel parser in PHP for retrieving the required information from an Excel worksheet. The parser is flexible enough to incorporate minor differences in the structure of Excel sheet.

This paper is organized in six sections as follows section I gives the brief introduction ,section II reports the prior work ,section III describes the proposed work and its model, result and analysis is given section IV, section V reports conclusion and scope for future work.

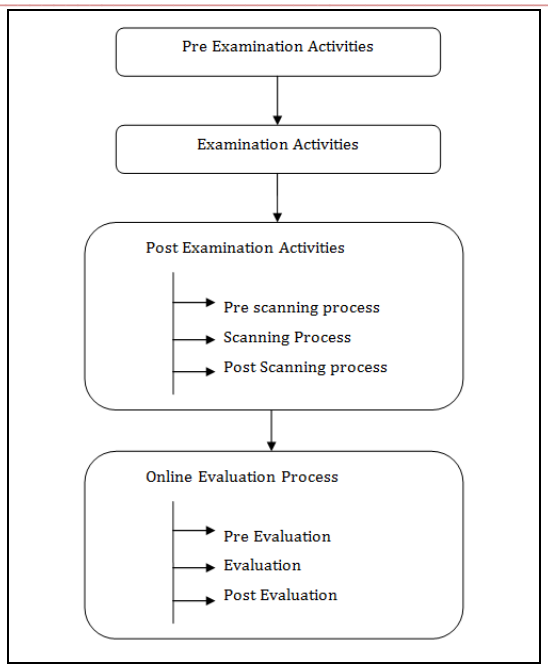

Figure 1. Activities under digital evaluation system

II. LITERATURE REVIEW AND PRIOR ART

In literature there exists a fistful of papers which deal with parsing of Excel spreadsheets. But most of them work with predefined structure and employ Java as underlying technology for implementation of parser. The existing parsers do not incorporate changes in structure of spreadsheets and as such demand a predefined static structure and break even with minor changes in the structure of the spreadsheet.

PHPExcel is a library written in PHP, provides a set of classes which allow writing to and reading from different spreadsheet file formats [1]. This project is built around Microsoft's OpenXML standard and PHP. The class SimpleXLXS can be used to parse and retrieve data from Excel XLS spreadsheet files [2]. It can parse a given Excel XLS file by extracting its contents files and parsing the contained XML spreadsheet file. The class provides functions to retrieve data for the spreadsheet worksheets, rows and cells.

SimpleExcel is a lightweight PHP library with simplistic approach for parsing/converting/writing tabular data from/to Microsoft Excel XML/CSV/TSV/HTML/JSON format [3]. This API comprises constructor, converter, parser, writer etc. Mohaideen Jamil posted article on parsing excel file in Java and how to export gridview contents to an excel sheet via a java web application [4].

Apache POI is an open source API can be used to read and write MS Excel files using Java [5]. Package includes classes such as HSSFWorkbook, HSSFSheet, XSSFWorkbook, XSSFSheet etc. JExcelAPI is a Java-based open source library allows reading, writing, and modifying Excel spreadsheets [6]. This was developed by Andrew Kahn and was released under the GNU Lesser General Public License.

Islam et al have developed an online examination system suitable for both Academic and Non-Academic examinations [7]. This application includes features such as user's registration, examination instruction, valid time of examination, time reminder, submission of the answer script, and release of the examination results. The system enables the admin or super admin to create a test from a question bank that comprises different subject areas. It is a multilingual system that permits users to conduct tests in their mother language. .

III. THEORETICAL FRAMEWORK

A. *Application Architecture*

The application architecture designed for deployment of web application on web hosting server and processing of end user HTTP request is shown in Figure 2. The information entered by an end user is passed in q query string which is extracted at server-side and the appropriate Excel file is selected based on the course selected by an end user. The Excel worksheet is parsed using PHP parser implemented by the authors before generating HTTP response to an end user.

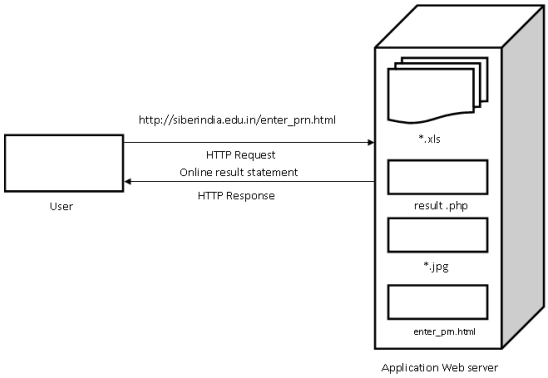

Figure 2. Processing of HTTP Request

B. *Problem Definition*

The Digital Evaluation software in use facilitates importing the result in Excel sheet which needs to be parsed before extracting the requisite information from it. However, it lacks web interface where the result can be accessed by an end user at any location and at any time. To enable the integration of existing examination software with institute's website, in the current work the authors have designed a web module which incorporates an Excel parser written in PHP which retrieves and compares the relevant data before presenting it to an end user. The following section describes the structure of an Excel file.

C. *Structure of Imported Excel Sheet*

The structure of an imported Excel sheet is shown in Figure 2. Such a worksheet is generated for each course run by the institute.

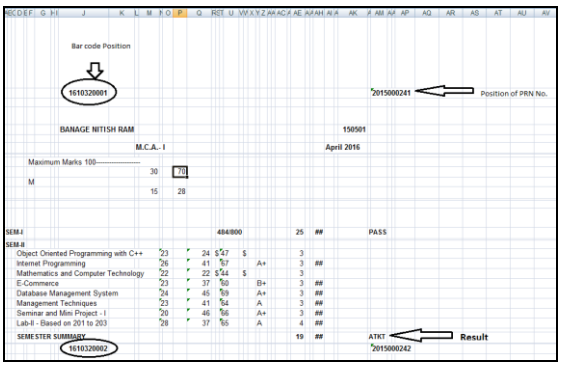

Figure 2. Structure of Imported Excel Sheet

As seen from Figure 2. the Excel sheet contains the required data at different column positions and the data is stacked one below other for different PRN No.'s assigned to different students. Further, the sheets generated for various courses slightly differ in their structure which is taken care of at code level.

The position of the data important for parsing Excel file corresponding to MCA-I-SEM-II course is shown in Table 1.

TABLE 1. COLUMNAR POSITION OF DATA IN EXCEL FILE

Component	Column Position
Bar Code	10
Result	39
PRN No.	39

Similar structure exists for other courses.

D. Control Flow Diagram

The control flow diagram for validation of PRN No., for taking care of different test cases and for displaying online mark sheet is presented in control flow diagrams Figure 4(a) – 4(c) connected by the connectors A and B.

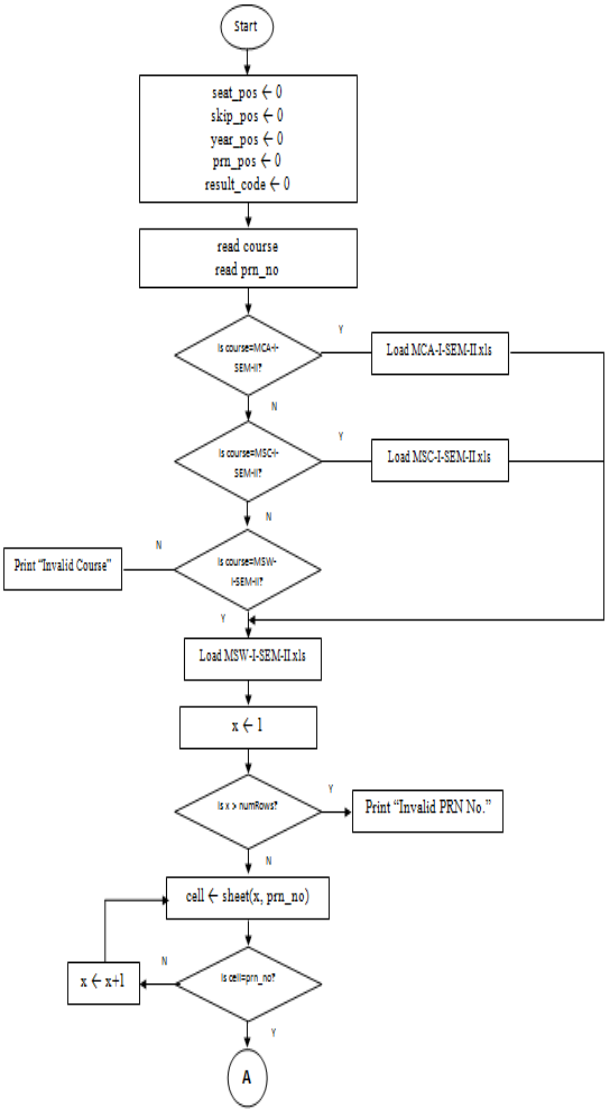

Figure 3(a) Control Flow Diagram for Validation of PRN No.

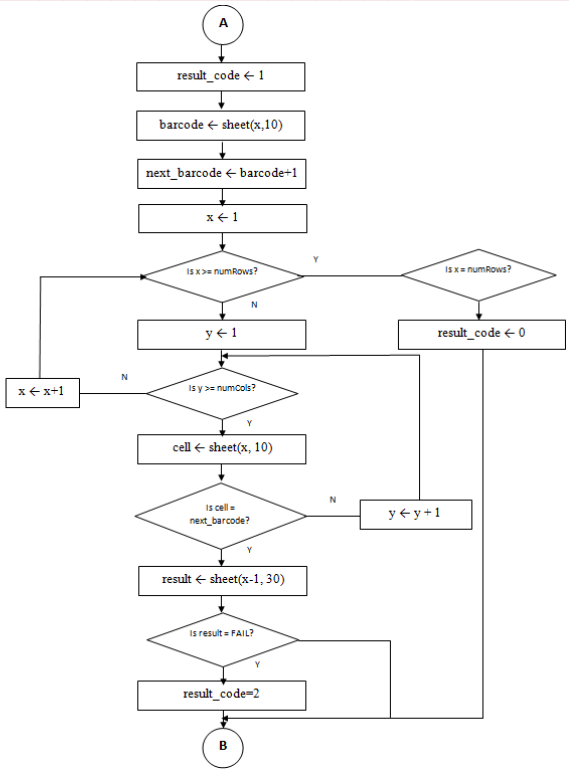

Figure 3(b) Control Flow Diagram for Different Test Cases

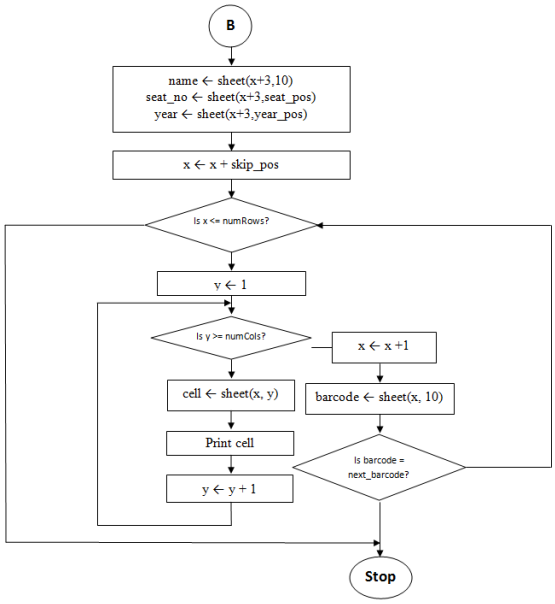

Figure 3(c) Control Flow Diagram for Displaying Online Mark Sheet

E. Proposed Algorithm :

The algorithm developed for parsing Excel file in C++ style is presented below:

/*Any high level language interfacing with Excel object library provides high level API for primitive functions such as loading Excel file in memory, counting number of rows and columns containing data and extracting data at the given address. Hence this algorithm assumes some standard functions as shown below:

Standard Functions of language L used in the Algorithm

getRows() - is function in a language L for returning number of rows containing data in a worksheet whose name is passed as an argument.

getcolumns() - is function in a language L for returning number of columns containing data in a worksheet whose name is passed as an argument.
getData() - is function in a language L which accepts the row and column address of a cell and returns the content stored at that address.

```
/* Global Variable Declaration */
int seat_pos;
int skip_pos;
int year_pos;
int prn_pos;
char course[50];
char sheet_name;
int result_code=1;

function displayResult(char SheetName[10])
{
/* To select appropriate Excel sheet for the course selected by an end user */
read course;
read prn_no;
if (course == “MCA-I-SEM-II”)
{
  read_excel(“MCA-I-SEM-II.xls”);
  skip_pos=12;
  seat_pos=37;
  year_pos=35;
  prn_pos=39;
}
else if(course == “MCA-I-SEM-II”)
{
  read_excel(“MSC-I-SEM-II.xls”);
  skip_pos=13;
  seat_pos=38;
  year_pos=36;
  prn_pos=40;
}
else if course == “MSW-I-SEM-II”)
{
  read_excel(“MCA-I-SEM-II.xls”);
  skip_pos=18;
  seat_pos=38;
  year_pos=36;
  prn_pos=40;
}
/* Initialize row position */
  x = 1;
/* Locate row containing PRN No.*/
numRows=getRows(sheet_name);
numCols=getColumns(sheet_name);
while ( x < numRows)
{
  cell=getData(x,prn_pos);
  if (cell==prn_no)
     break;
}
/* If search fails, PRN No. does not exist, otherwise x  contains the required row no.*/
if (x == numRows)
   ressultCode=0;
else
 /* x value now corresponds to the row containing the given PRN No. in an Excel sheet */
 {
          /* Get barcode corresponding to the given PRN No. */
          /* Barcode is located at column position 10 */
          barcode=getData(x,10);
          next_barcode=barcode+1;
          /* Check whether the result is fail */\
          x1=x;
          while (x1 < numrows)
          {
             y=1;
             while (y < numCols)
             {
                    cell=getData(x1,10);
                    if (cell == next_barcode)
                    {
                      /* Result is located at position 39 of previous row*/
                      result=getData(x-1,39);
                     if (result==”FAIL”)
                     {
                              result_code=2;
                              break 2;
                     }
                    }
                    y++;
             }
             x++;
          }
 }

 /* Check result_code and take appropriate action */
if (result_code==0)
{
  print “PRN No. does not exist”;
}
else if (result_code==2)
{
   print “Please Contact Examination Department”);
}
else
{
 /* Display Result */
 name = getData(x+3,10);
 seat_no=getData(x+3,seat_pos);
 year=getData(x+5,year_pos);
x=x+skip_pos;

/* display all rows from current barcode to next_barcode */
 while ( x < numRows)
 {
          y=1;
          while (y < numCols)
          {
                    cell=getData(x,y);
                    print(cell);
                    y++;
          }
          x++;
          barcode_cell=getData(x,10);
          if (barcode_cell == next_barcode)
                    break;
 }
}
}
```

The value returned by the displayResult() function can take one of the values 0, 1 or 2 as described in Table 2.

\

TABLE 2. POSSIBLE VALUES OF RESULT_CODE

result_code Value	Description
0	PRN No. does not exist or Course is invalid
1	Both PRN No. and Course are valid
2	Result Fail

IV. RESULT AND ANALYSIS

The algorithm proposed above is implemented in PHP and is tested for several test cases shown in Table 3.

TABLE 3. TEST CASES FOR WEB MODULE

Test Case Description	Test Case Outcome
PRN No. is invalid	Display message "Please Check PRN No. and Course Name"
PRN No. is valid but does not correspond to selected course	Display message "Please Check PRN No. and Course Name"
Both PRN No. and Selected Course are valid	Display Mark Statement corresponding to the PRN no.
Boundary Value Testing. PRN No. selected is the last PRN No. in the list.	Display Mark Statement corresponding to the PRN no.
Result is Fail	Display message "Please Contact Examination Department"

A. *Application Folder Structure*

The deployment folder structure for the web module is shown in Figure 5.

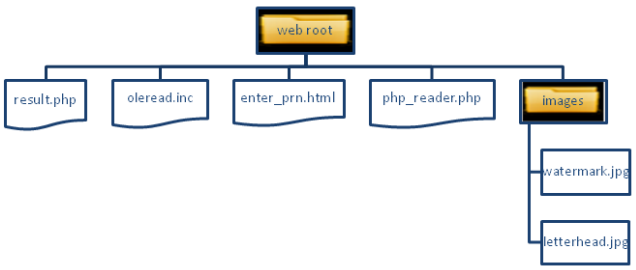

Figure 5. Deployment Folder Structure for Web Module

The GUI generated by the web module is shown in Figure 6.

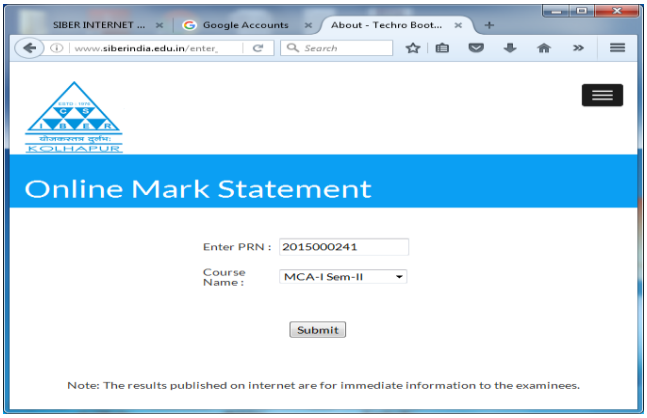

Figure 6. GUI Generated by Web Module

B. *Test Case 1 : PRN No. is invalid*

If the PRN No. entered by an end user is invalid and if it does not exist in an Excel worksheet, an appropriate error message is displayed to an end user as depicted in Figure 7.

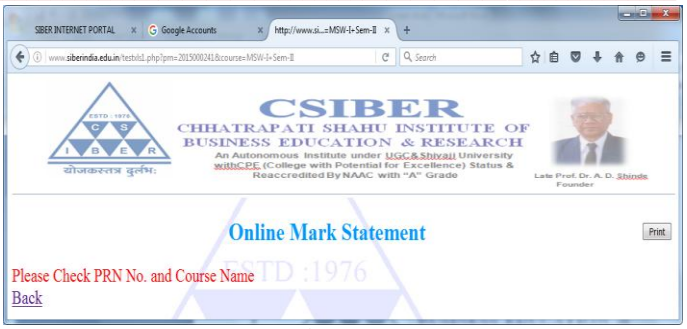

Figure 7. Output Generated for Test Case 1 (PRN No. is Invalid)

C. *Test Case 2 : PRN No. is valid but does not correspond to selected course*

If the correct PRN No. is entered by an end user but if the wrong course is selected, the same message as in Test Case 1 is communicated to an end user.

D. *Test Case 3 : Both PRN No. and Selected Course are valid*

In this case an online mark statement is presented to an end user which properly takes care of all his backlogs, if any as shown in Figure 8(a) and 8(b). The logic employed is as follows. The barcode is used as a delimiter for separating records. The barcode corresponding to the PRN No. entered by the user is tracked and the block between current barcode and next barcode is displayed to an end user.

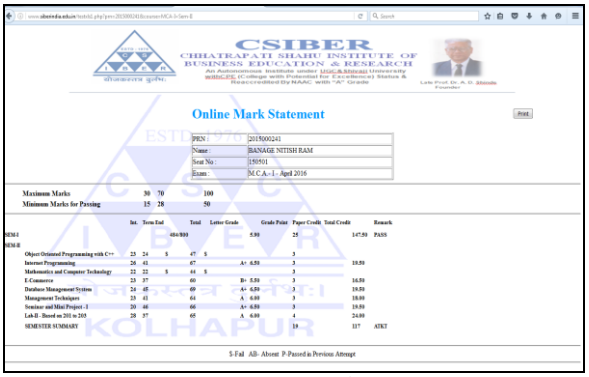

Figure 8(a). Output Generated for Valid PRN No.

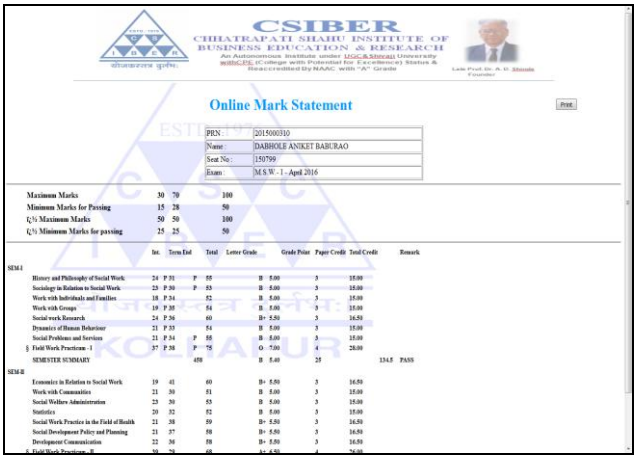

Figure 8(b). Output Generated for the Candidate with Backlogs

E. *Test Case 4 : Boundary Value Testing. PRN No. selected is the last PRN No. in the list.*

If the PRN No. entered by an end user happens to be the last one in an Excel sheet, then there is no bar code next to the current barcode in this case, and all the records till the end of the file are displayed to an end user.

F. Test Case 5 : Result is Fail

If the candidate is declared as fail, the output generated by the web module is depicted in Figure 9.

Figure 9. Output Generated by the Web Module for the Candidate Declared Fail

The print capability is added to the web module. The print preview generated by the module is shown in Figure 10.

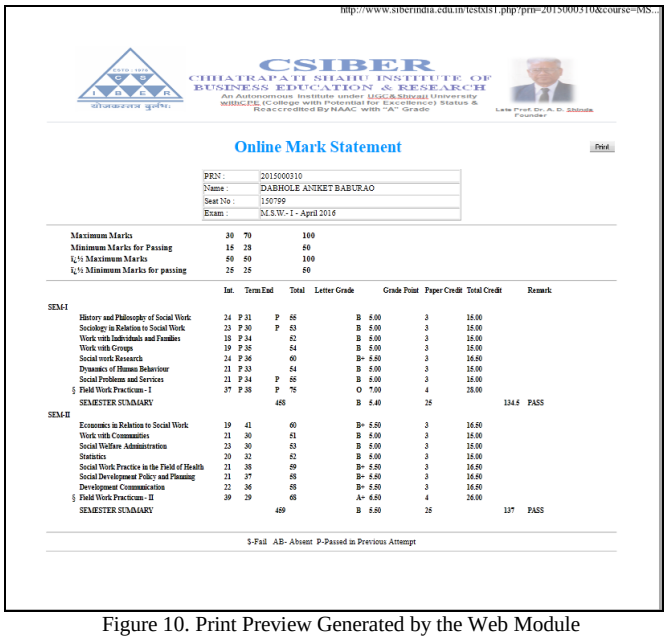

Figure 10. Print Preview Generated by the Web Module

V. CONCLUSION AND SCOPE FOR FUTURE WORK

In the era of information and communication technology, the software of all type and size provide some or other kind of web interface to extend the organization's reachability beyond its premises. In the current work, the authors have integrated DES software of the institute with existing institute's website. As a measure towards performance improvement, a flat file in Excel format is parsed for extracting relevant information to an examinee. An algorithm is designed and implemented in PHP for this purpose. The deployment folder structure is presented and various test cases are explored. The parser properly takes care of any structural differences pertaining to different courses.

In the current work, the differences in Excel spreadsheet structures is taken care of at code level which is hard coded in PHP parser. In future, the authors intend to separate out such data from the main parser and place it in an XML file which can be parsed before displaying on-line result to an end user. Such a code separation facilitates the parser to be most flexible to accommodate any such differences in future without any modification to a parser.

REFERENCES

[1] https://phpexcel.codeplex.com/
[2] http://www.phpclasses.org/package/6279-PHP-Parse-and-retrieve-data-from-Excel-XLS-files.html
[3] http://faisalman.github.io/simple-excel-php/api/0.3/
[4] http://www.simplecodestuffs.com/java-xls-xlsx-csv-quries/
[5] http://howtodoinjava.com/apache-commons/readingwriting-excel-files-in-java-poi-tutorial/
[6] http://www.javaworld.com/article/2074940/learn-java/java-app-dev-reading-and-writing-excel-spreadsheets.html
[7] Zahirul Islam, Mostafizur Rahman and Kabirul Islam, "Online Examination System In Bangladesh Context", International Journal of Science Environment and Technology, Vol. 2, No 3, 2013, 351 – 359

Appendix A
PHP Source Code

```
<?php
/* Global Variable Declaration */
$seat_pos=0;
$skip_pos=0;
$yy1=0;
$year_pos =0;
$valid=0;
$course;
echo "<html><body style=background-image:url(w3.png);background-repeat:no-repeat; align:center>";

include_once ("excel_reader.php");    // include the class
$GLOBALS['course'] = $_GET["course"];

// creates an object instance of the class, and read the excel file data

$excel = new PhpExcelReader;
if($GLOBALS['course']=="MCA-I Sem-II")
{
$excel->read('MCA-I-Sem-II.xls');
$GLOBALS['skip_pos']=12;
$GLOBALS['seat_pos']=37;
$GLOBALS['yy1']=39;
$GLOBALS['year_pos']=35;
}
else if($GLOBALS['course']=="MSc Envt-I Sem-II")
{
$excel->read('MSc-I-SEM-II.xls');
$GLOBALS['skip_pos']=13;
$GLOBALS['seat_pos']=38;
$GLOBALS['yy1']=40;
$GLOBALS['year_pos']=36;
}
else if($GLOBALS['course']=="MSW-I Sem-II")
{
$excel->read('MSW-I-SEM-II.xls');
$GLOBALS['skip_pos']=18;
$GLOBALS['seat_pos']=38;
$GLOBALS['yy1']=40;
$GLOBALS['year_pos']=36;
}

echo " <center><img src=lh.png width=1100 height =150></img></center>";

echo "<table><tr><td width= 1250><center><h1 style=color:#0099ff;>Online Mark Statement</h1></center></td> ";
```

```
echo "<td><button type=button onclick=window.print()
>Print</button></td></tr></table>";
// this function creates and returns a HTML table with excel rows and columns
data
// Parameter - array with excel worksheet data
function sheetData($sheet) {
  $GLOBALS['valid']=1;
  $prn = (int)$_GET["prn"];
  $re = '<table style="font-size:10pt; font-weight:bold">';
// starts html table
 $x = 1;

while($x <= $sheet['numRows']) {
   $yy = $GLOBALS['yy1'];
   $cell = isset($sheet['cells'][$x][$yy]) ? $sheet['cells'][$x][$yy] : '';
     if($cell ==$prn)
       break;
   $x++;
 }
// code for fail candidates

$y= 10;
$cell = isset($sheet['cells'][$x][$y]) ? $sheet['cells'][$x][$y] : '';
$cell_2 = $cell+1;
$xfail = $x;
 while($xfail <= $sheet['numRows'])
 {
  $y = 1;
  while($y <= $sheet['numCols'])
  {
    $cell = isset($sheet['cells'][$xfail][10]) ?
                $sheet['cells'][$xfail][10] : '';
    if ($cell == $cell_2)
    {
                        $x1 = $xfail-2;
                        $yfail =isset($sheet['cells'][$x1][39]) ?
              $sheet['cells'][$x1][39] : '';
 if ($yfail == 'FAIL')
 {
  $GLOBALS['valid']=2;
  break;
 }
}
$y++;
}
if($GLOBALS['valid']==2)
 break;
$xfail++;
}

if($x == $sheet['numRows']+1)
  {
    $GLOBALS['valid']=0;
    echo "<script> document.write('<font size=5 color=red>Please Check
PRN No. and Course Name'); document.write('<br><a href=
http://localhost/enter_prn.html> Back </a></font>');</script>";
}
else
{
if($GLOBALS['valid']==2)
  {
    echo "<script> document.write('<font size=5 color=red>Please Contact
Examination Department'); document.write('<br><a href=
http://localhost/enter_prn.html> Back </a></font>');</script>";
}
else
{
$y= 10;
$cell = isset($sheet['cells'][$x][$y]) ? $sheet['cells'][$x][$y] : '';
$cell_2 = $cell+1;
$name=  isset($sheet['cells'][$x+3][$y]) ? $sheet['cells'][$x+3][$y] : '';
$s = $GLOBALS['seat_pos'];
$seatno = isset($sheet['cells'][$x+3][$s]) ? $sheet['cells'][$x+3][$s] : '';
$course1= isset($sheet['cells'][$x+5][12]) ? $sheet['cells'][$x+5][12] : '';
$yr = $GLOBALS['year_pos'];
$year= isset($sheet['cells'][$x+5][$yr]) ? $sheet['cells'][$x+5][$yr] : '';
echo "<center><table border width=500>";
echo "<tr><td>PRN : ";
echo "<td>";
echo $prn;
echo "</tr>";

echo "<tr><td>Name : ";
echo "<td>";
echo $name;
echo "</tr>";

echo "<tr><td>Seat No : ";
echo "<td>";
echo $seatno;
echo "</tr>";

echo "<tr><td>Exam : ";
echo "<td>";
echo $course1." - ".$year;
echo "</tr>";

echo "</table></center><hr>";

// show max and min marks
echo "<table style= font-weight:bold;>";
echo "<tr>";
echo "<td width=40></td><td width=295> Maximum Marks</td>";
echo "<td width=30 >30</td>";
echo "<td width=110 >70</td>";
echo "<td>100</td>";
echo "</tr><tr>";
echo"<td width=40></td><td width=150> Minimum Marks for
    Passing</td><td>15</td><td>28</td><td>50</td></tr>";
if($GLOBALS['course']=="MSW-I Sem-II")
{
echo "<tr>";
echo "<td width=40></td><td width=295> § Maximum Marks</td>";
echo "<td width=30 >50</td>";
echo "<td width=110 >50</td>";
echo "<td>100</td>";
echo "</tr>";
echo "<tr>";
echo "<td width=40></td><td width=295> § Minimum Marks for
                        passing</td>";
echo "<td width=30 >25</td>";
echo "<td width=110 >25</td>";
```

 ISSN: 2321-8169
527 - 534

```
echo "<td>50</td>";
echo "</tr>";

}
if($GLOBALS['course']=="MBA-I Sem-II")
{
echo "<tr>";
echo "<td width=40></td><td width=295> # Maximum Marks</td>";
echo "<td width=30 > - </td>";
echo "<td width=110 >100</td>";
echo "<td>100</td>";
echo "</tr>";
echo "<tr>";
echo "<td width=40></td><td width=295> # Minimum Marks for
                        passing</td>";
echo "<td width=30 > - </td>";
echo "<td width=110 >50</td>";
echo "<td>50</td>";
echo "</tr>";
}
echo "</table><hr>";

$x=$x+$GLOBALS['skip_pos'];
 while($x <= $sheet['numRows'])
 {
  $re .= "<tr>";
  $y = 1;
  while($y <= $sheet['numCols'])
  {
   $cell = isset($sheet['cells'][$x][$y]) ? $sheet['cells'][$x][$y]
                : '';

    if($cell == "Remark")
      $re .= "<td></td><td>$cell</td>\n";
   else
    $re .= " <td>$cell</td>\n";
    $y++;
  }
  $re .= "</tr>\n";
  $x++;
  $y1=10;
  $cell_1 = isset($sheet['cells'][$x][$y1]) ?
            $sheet['cells'][$x][$y1] : '';
 if ($cell_1 == $cell_2)
  break;
 }//end if outer while

 return $re .'</table>';    // ends and returns the html table
}
// end of inner else
}
// end of outter else

}
$nr_sheets = count($excel->sheets);       // gets the number of worksheets
$excel_data = '';             // to store the the html tables with data of each sheet

// traverses the number of sheets and sets html table with each sheet data in
$excel_data
/*$excel_data .= '<h4>Sheet '. ($i + 1) .' (<em>'. $excel-
>boundsheets[$i]['name'] .'</em>)</h4>'. sheetData($excel->sheets[$i])
.'<br/>';*/
for($i=0; $i<$nr_sheets; $i++) {
 $excel_data .= sheetData($excel->sheets[$i]) .'<br/>';
}

//echo $excel_data;      // outputs HTML tables with excel file data
if($GLOBALS['valid']==1)
{

$excel_data = $excel_data."<hr><center>$-Fail   AB-
Absent  P-Passed in Previous Attempt   </center><hr>";
echo $excel_data;
}
echo "</body></html>";
?>
```