

System Design and Case Study Reporting on AQASYS: A Web-based Academic Quality Assurance System

Adel Alfozan¹, Mohammad Ali Kadampur²

Dean College of Engineering, Imam Mohammad Ibn Saud Islamic University, P.O.Box:5701 Riyadh 11432 KSA¹

Assistant Professor Department of Electrical Engineering, Imam Mohammad Ibn Saud Islamic University, Riyadh 11432 KSA²

Abstract—The demands of modern education have evolved from a teacher-centric requirement to a learner-centric requirement. Knowledge, skill, and competence are the most sought-after attributes of a graduate. Features such as the objective focus of learning, curriculum planning, a set of high expectations, and extended opportunities to the learner after completion of education are at the center of all the planning. It is all about skill-oriented, outcome-based standardization that has been infused by the societal stakeholders into the modern global education system to create a work-ready human capital. In this paper, a software product for academic quality assurance is presented. The software provides a generic framework to any educational institution that operates to implement known international standards of education. The software accepts the data and computes the quality parameters as per the selected standards. It has an analytical module that provides summary analytics and generates the course reports in the given format automatically. The software is tested with a case study and results are presented. The paper also presents the system design approach with discussion on the technologies selected for the development.

Keywords—*Outcome-based education; quality standards; automated software; system design; education technology; accreditation*

I. INTRODUCTION

In the last decade or two, the requirements of higher education particularly in professional education have shifted from a “teacher-centric” approach to a “learner-centric approach” [5]. In the traditional approach whatever teacher knew was used as the content or curriculum of the teaching. However, in the changing societal needs and driving forces of the job market, industry, and global changes, the learner needs to have knowledge, skill, and competence in the chosen areas of learning. The learner selects the domain of learning and sets objectives to achieve them at the end of the learning. This kind of pole shift and paradigm shift in teaching-&-learning has brought the educational institutes to frame new curricula, with defined objectives and clearly stated goals as “what the learner will be able to do” at the end of the learning? The standard educational institutes continuously loop around these

parameters and assess the level of achievement through their operational practices. This activity is given the word “Quality” [1][2] in most of the educational institutes that are signatories to the internationally recognized standards organizations [5].

In this work, a web-based academic quality management system is implemented. The paper describes the system design process and reports on the results after specific case study data is input to the system. The system is called AQASYS, an Academic Quality Assurance System.

The research questions on methods of quantifying Course Learning Outcomes (CLO’s), Program Learning Outcomes, the Difficulty level of the course, automated feedback assessments, CLO, PLO attainments are addressed. An automated CLO mapping module, design of a text-mining based Bloom’s level recommender module are built and integrated into the system. The system generates course-specific and program-specific results providing several summary analytic outputs of business value. The system addresses the issue of “closing the loop” to assess the committed objectives.

The system facilitates the academic quality assessment process with web-interface for the users and a reliable database at the backend. It is a secured, reliable, and all-time available resource for its users. The system provides all inherent advantages of web-based applications.

II. MOTIVATION

Most of the institutes that were visited and the faculty that were consulted, shared their experience of handling “academic quality” matters in their respective institutes and departments. Based on this the following major observations were made:

- 1) Each faculty was using some kind of excel sheet and running around for the templates and methods of calculations on assessments. There was absolute last-minute chaos, confusion, and lethargy in what they were doing.
- 2) There was no archive of centralized data repository to click and see the analytics of the past and explore it for future strategies.
- 3) Lack of sound research-driven statistical and mathematical foundations uniformly applied across the departments for assessment and achievements.

4) Plenty of “number in number out” kind of manipulated data to skew the academic quality for the personal advantage of the faculty or to the advantage of the department. The piety of the information was affected.

5) Irrational ways of mapping CLOs with the question paper text, resulting in the illusion of whole quality exercise. And finally,

6) Very poor regularity and ease of academic data management.

In light of these observations, the paper is presenting a web-based academic quality assurance system that has a centralized repository of academic data, research-driven academic foundation for all quality parameters, and above all ease of access with availability, confidentiality, and integrity.

The paper is organized in the following way: Section III presents a literature review and related work. Section IV lists the contributions of the paper. Section V presents in detail the software architecture used. Section VI addresses system design features. Section VII briefs about the development platform and the associated technologies used in the product development. Section VIII discusses the operational parameters and their formulae of computation. Section IX presents results and selected screenshots. The final section, Section X is the concluding section.

III. RELATED WORK

In the past two decades, there have been several research papers and software products attempting to illustrate and automate the process of academic accreditation [10] in higher education. This section reviews and compares systems like the one presented in this paper.

ABET Course Assessment Tool (ACAT) [4] is one of the earliest (2010) and decent attempts found to be made to automate the accreditation process. It is a web-based application designed to assist in the collecting of data and the generation of standardized reports.

Course and Student Management System (CSMS) [9] is a Web-based system based on the concept of an articulation matrix that maps various course outcomes to ABET program outcomes[6]. The articulation matrix is nothing but the course assessment matrix. It is a 2D table, in which rows represent learning objectives, and columns represent class activities conducted to meet those learning objectives. The articulation matrix is filled when the course learning objectives are achieved.

Program and Courses Outcomes System (PACOS) [13] is an initiation from Hong Kong University [12]. It is an open-source web-application. PACOS has various power levels of access with different user privileges. A user can perform CRUD (Create, Read, Update, Delete) operations on courses, course information, and program outcomes. PACOS manages data with excel files as inputs and displays result in the form of histogram and reports.

Program Outcome Assessment (POA) [3][11] is a web-based program developed using ASP.Net technologies to automate the program assessment. It supports a complete flow

from defining the program outcomes [6] to entering all course information in real-time. The program computes on-the-fly performs all the evaluations and presents the output in the form of graphs, color-coded tables, and a year, year to year comparison of various program outcomes for a span of five years.

Apart from the above work in which mainly academicians have demonstrated automated software programs for program assessment and accreditation, there are many software companies offering these services through their proprietary software products.

1) *CLOSO*: CLOSO [17] is an excellent, sustainable & affordable tool for monitoring the educational process and performance. It is a product from smart-accredit.com. It allows to plan and design assessments, generate course folders, collect and analyze feedbacks, display outcome attainments. It offers a free trial version for individuals for a limited period of testing the product.

2) *Contineo*: 'Contineo' [18] is a pioneering software platform for the implementation and administration of academic autonomy with the guided philosophy of the Outcome-Based Education System. It mainly focuses on the accreditation standards such as the National Board of Accreditation (NBA) followed in India and the subcontinent. Service is available through contineo.in for personalized customization in the product.

3) *Watermark*: [19] It is a Cloud-based software that helps higher education institutions manage assessment, accreditation, curriculum, course evaluation, & faculty activity. It is available at (<https://www.watermarkinsights.com/>)

4) *WEAVE*: [19] Weave is a cloud-based accreditation and assessment solution designed to help higher education institutions with program review, course planning, and more. The programmatic assessment functionality lets organizations analyze assessments and provide insight into student performance. It is available at (<https://weaveeducation.com/>).

There is another plethora of companies offering accreditation services spanning industry and academia. To list a few here [19] ARMATURE, Jetpack, Jura, SPOL, powerDMS, qualtrax, compWalk, DocTract, Submittable, WizeHive, and the list grows. It is to be noted that the companies listed in this paragraph however are not focusing on academic accreditation in particular but targeted for commercial business enterprises in general.

IV. CONTRIBUTIONS OF THE PAPER

Academic quality Assurance System (AQASYS) presented in this paper is different from most of the above products in the following major ways:

1) It has an intelligent box (iBOX) feature which automatically classifies assessment questions and assigns proper CLOs as per Bloom's taxonomy.

2) It maps back the assessment questions to Program Learning outcomes via the Course Learning Outcomes (CLO).

3) It produces the Course report automatically in each course by collecting relevant grade-book information, student feedback analysis, CLO achievement statistics and intelligent recommendations for course improvement wherever necessary.

4) It produces the summary analytics of the program for all courses of the program in a single view so that executives of the program can easily comprehend the course statistics.

5) It has in-built triggers to remind and notify the user about data entry delays so that the system can track the punctuality of the faculty in pursuing the quality matter in their academic functioning.

6) It has data archiving and data freezing facilities so that the data cannot be modified even by the data owner (faculty) after a specified due date. This helps to keep the sanctity of data and draw a line for “end of a cycle”.

7) It is generic in accepting program-dependent PLOs, CLOs, any number of students, any type of assessment, with time stamps and user foot-prints being recorded in the background.

8) It has a file upload facility to store quality files and course portfolios.

9) It provides data-isolation, user authentication, and ease of usage. The user need not bother about the computational details of quality standards.

10) Of course it is a web-based product with MVC architecture and accessible on desktop, tablet, and mobile as the design is made self-responsive.

V. THE SOFTWARE ARCHITECTURE

Considering the existing IT environment, user dynamics, software quality attributes that are required, design structure and business strategy employed in most of the higher education institutes a Model View Controller (MVC) framework was used in a client-server architecture, refer Fig. 1. The open source framework called *Codeigniter* [14] was selected for this purpose. It supports MVC development with PHP [7].

- **Model:** The Model contains only the pure application data, Model works as back-end. It deals with back-end operations and fetches data from the database and send it to the controller.
- **View:** A view is simply a web page, or a page fragment, like a header, footer, sidebar, etc The View presents the model’s data to the user. It works as front-end. It displays data and captures user actions, sends user actions to the controller.
- **Controller:** The controller works as an intermediary between model and view and controls the actions between them. It listens to events triggered by the view (or another external source) and executes the appropriate reaction to these events. Controllers decide how the HTTP requests are to be handled.

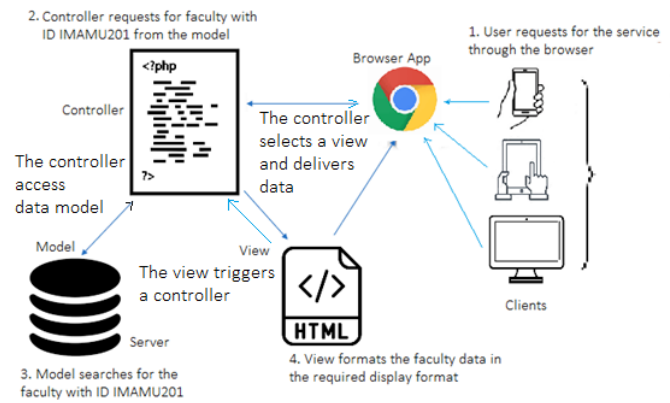


Fig. 1. MVC Client Server Architecture of AQASYS.

A. The AQASYS Interface

The AQASYS has five main modules with sub-modules as listed below:

- 1) Data entry module
 - a) Uploading from excel sheets
 - b) Enter by using GUI forms
 - c) Upload from URLs
- 2) Assessment Module
 - a) CLO Assessment
 - b) PLO Assessment
 - c) KPI Assessment
 - d) SO Assessment
- 3) Analytics Module
 - a) CLO achievement analysis
 - b) PLO achievement analysis
 - c) Filtered PLO attainment analysis
 - d) Feedback Analysis
- 4) Reporting module
 - a) Automatic course report generation
 - b) View charts
- 5) The AI ML module (iBOX)
 - a) Bloom’s taxonomy tool

Fig. 2 below shows the AQASYS block diagram.

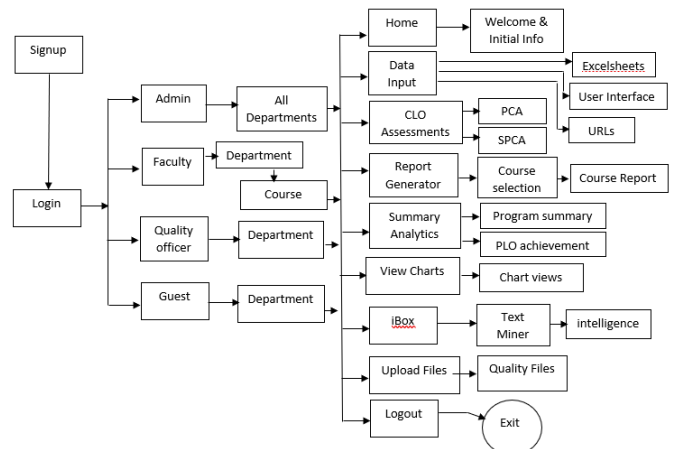


Fig. 2. Block Diagram of AQASYS.

VI. THE SYSTEM DESIGN

This section presents selected system design features of AQASYS.

A. The Use Cases

The use case diagram Fig. 3, describes how the different users interact in the AQASYS. Admin, Faculty, Student are all a type of users with different access privileges. New user is also a type of user who requires signup before getting access to AQASYS services. Each user can avail different services as indicated in the use case boundary.

B. The Class Diagram

The class diagram Fig. 4, describes the structure of AQASYS. The user class is the parent class for many user types such as student, faculty, admin, guest and the NewUser. Every user is associated with a department. A department has PLOs and courses. Each course has CLOs, KPIs and course assessments. CLOs contribute to PLOs of the department and they are associated with PLOs. The CLOs, KPIs and the assessments are mapped while producing the Table of

Specifications (ToS) on the course, example refer Table I. In order to perform mapping, AQASYS provides an intelligent auto-mapper class called iBOX. The iBOX is based on the Bloom's taxonomy. The iBOX class is associated with MappingTable class. Most of the operational methods are grouped in AqasysOp class.

C. The Data Model

This section describes the selected listing of databases and the table structures within each database. A sample of the model is shown in Fig. 5. The model has databases for each department and inside each department there are tables such as studentlist, facultylist, courses, courseallotment, plodefinitions, clodefinitions, kpidefinitions, tostable, etc. refer Fig. 6. There is a separate database for users of AQASYS. The application_users database has tables such as users, userlogs, departments, contactform. The database on Bloom data stores an incrementally growing data as a result of text processing in the AQASYS application. The Bloom data and associated algorithms add intelligence to the system through iBOX tab.

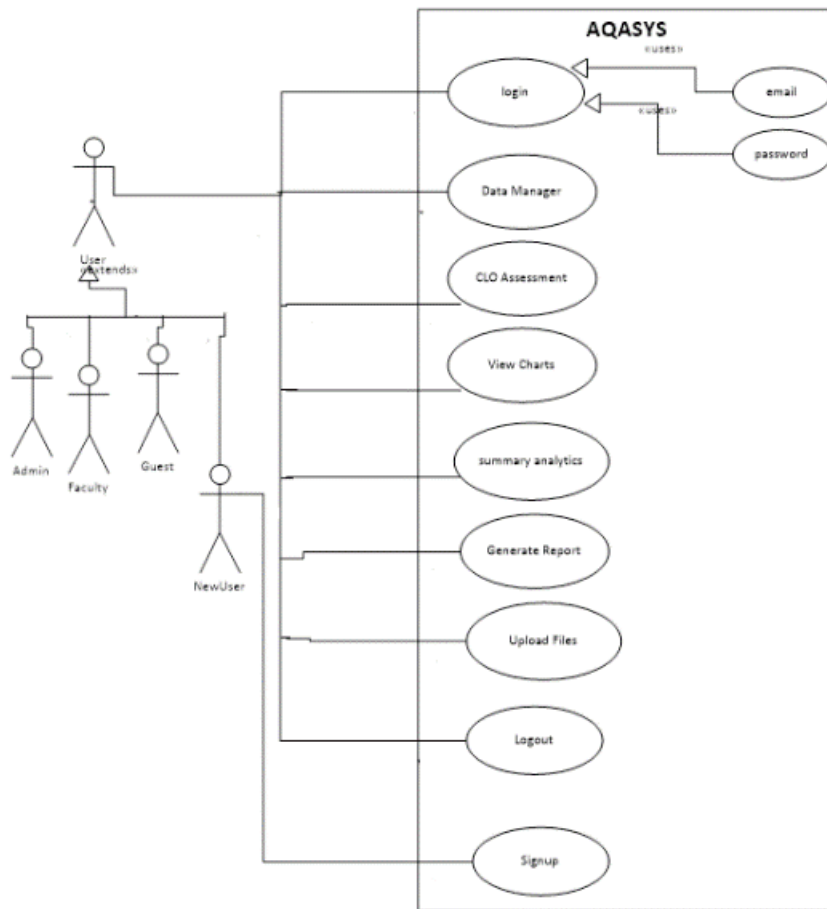


Fig. 3. Use Case Diagram of AQASYS.

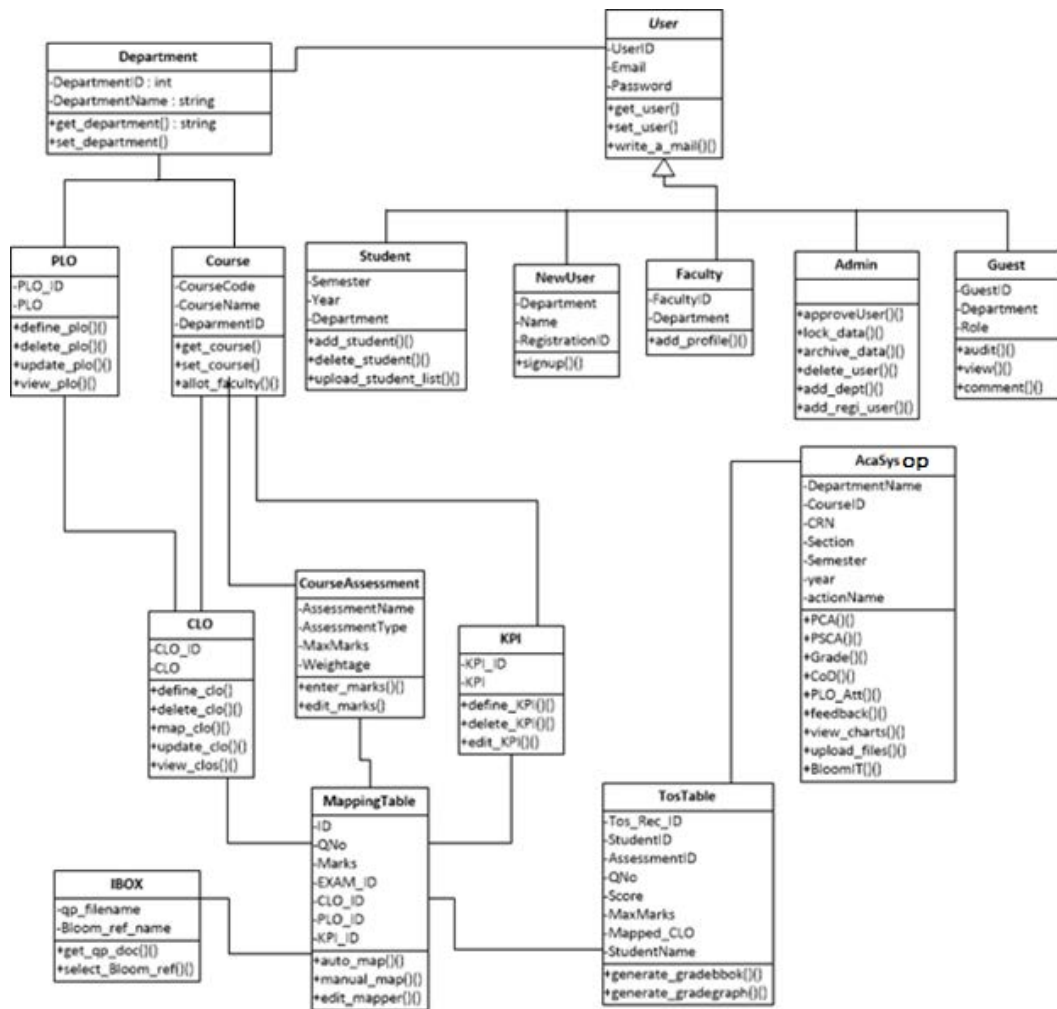


Fig. 4. The Class Diagram.

TABLE I. SELECTED LIST OF MVC METHODS

Table Column Head	View	Controller
generate_assessments.php generate_charts.php generate_cr.php get_clo.php get_clomselections.php get_clomselections_student.php get_coursecode.php get_exam.php get_feedbackgraph.php get_gradebookdata.php get_gradegraph.php get_kpi.php get_maxmarks.php get_plo.php get_plo_analytics.php.jsx get_so.php get_so_kpi_data_table.php get_summary_analytics.php get_summary_plo.php get_tostableyear.php get_userinfo.php import_coursedata.php	view_assessments.php view_charts.php form_cl_clo.php form_clo_definitions.php form_clodata.php form_clomdata.php form_clomtable.php form_evaluationchema.php form_exam_table.php form_gradebook.php form_kpi_definitions.php form_plo_definitions.php form_registered_students.php form_so_definitions.php form_so_kpi_data_table.php form_so_kpi_relationship.php	Login_controller.php Access_controller.php Delete_controller.php Update_controller.php Create_controller.php Display_controller.php Print_controller.php Logout_controller.php Signup_controller.php Validation_controller.php Bloom_controller.php Session_controller.php

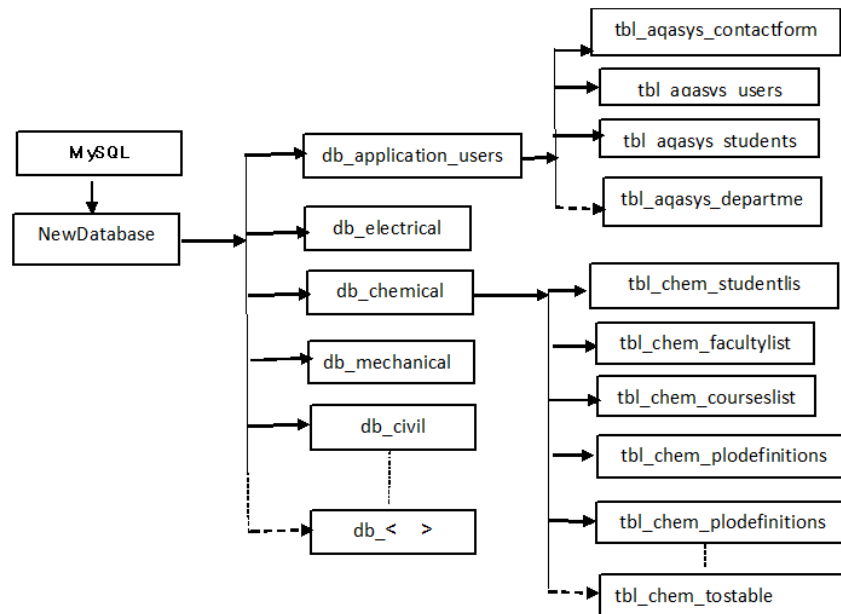


Fig. 5. The Block Diagram of the Data Model.

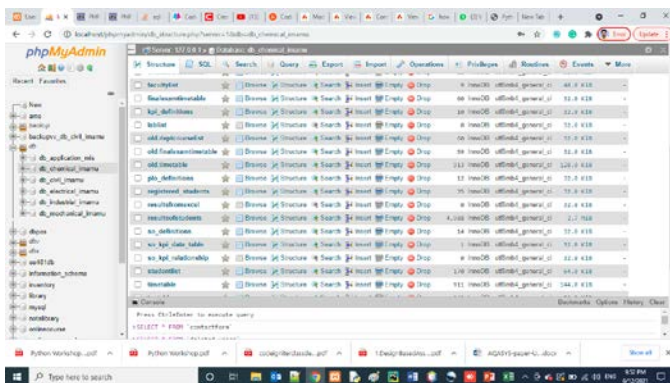


Fig. 6. The MYSQL View of the Data Model, a Sample Table Structure.

VII. THE DEVELOPMENT PLATFORM

Choosing a development platform and technology is always an intriguing question for the developer. After conducting good research, in this application, Codeigniter MVC framework [14] with PHP was chosen for the reason that the Codeigniter has a light footprint, flexible with MVC and non-MVC applications, supports MYSQL database and the learning curve is not very steep for the developer. Codeigniter has inbuilt security against Cross-Site Request Forgery (CSRF) and Cross-Site Scripting (XSS) attacks and provides content security. Finally, Codeigniter 4.0.3, PHP 7.2, HTML 5, CSS, JavaScript, AJAX, Bootstrap 4.

MYSQL are the combination of technologies used for the development.

VIII. THE OPERATIONS IN AQASYS

This section presents various operations and computations involved in AQASYS. These computational formulae are adopted from the academic quality standards and customized

to the institutional requirement. Refer the following Table II, to understand the computations of this section.

1) *Percentage CLO Attainment PCA (Pure)*: This calculation considers all the students who have appeared for the exam and obtained some score including the failure scores if any. This calculation doesn't exclude the failed student while calculating PCA hence the name PCA(Pure). It is basically the percentage of the CLO mark attained by the students as defined below:

$$PCA = \frac{\left\{ \sum_0^N MS_{CLO=j} \right\}}{\sum_0^k MX_{CLO=j}}$$

Where MS is the mark of each student and MX is Maximum marks allotted for that question of the assessment (CLO). N is the total number of students addressed by the jth CLO.

For example consider CLO 2.1:

$$PCA = \frac{2 + 4 + 4 + 4 + 3 + 1 + 0 + 4 + 3 + 4 + 3 + 2 + 5 + 6 + 6 + 5 + 6 + 4 + 5 + 1}{(10 + 10) \quad (4 + 6)}$$

PCA=0.36 Or 36%

Similarly, for CLO 1.1, 1.2 and 3.1 the % attainment values are

$$PCA = \frac{3 + 3 + 3 + 3 + 3 + 2 + 3 + 0 + 3 + 3}{(10) \quad (3)} = 0.866; 86.6\%$$

$$PCA = \frac{2 + 2 + 1 + 0 + 0 + 2 + 1 + 2 + 2 + 1}{(10)} = \frac{(2)}{(2)} = 0.65; 65\%$$

$$PCA = \frac{1 + 5 + 2 + 5 + 3 + 3 + 5 + 4 + 1 + 2}{(10)} = \frac{(5)}{(5)} = 0.62; 62\%$$

These values can be verified, referring to the Table II.

TABLE II. TABLE OF SPECIFICATIONS (TOS) EXAMPLE

	Q1	Q2	Q3	Q4	Q5
MaxMarks	4	5	6	2	3
CLO	2.1	3.1	2.1	1.2	1.1
StudentID					
S101	2	1	3	2	3
S102	4	5	2	2	3
S103	4	2	5	1	3
S104	4	5	6	0	3
S105	3	3	6	0	3
S106	1	3	5	2	2
S107	0	5	6	1	3
S108	4	4	4	2	0
S109	3	1	5	2	3
S110	4	2	1	1	3
PCA	36%	62%	36%	65%	86.6%
DL	27.5%	38%	28.3%	35%	13.3%
ODL	=28.43%				

2) *Percentage student CLO attainment (PSCA)*: It is a measure of x% of students who attained y% of CLO mark. As per the user requirements, the PSCA calculation excludes the records of failed students and classifies the graded students into four classes viz. Excellent (E), Adequate (A), Medium (M), Unsatisfactory(U). The target range for classifying into E, A, M, and U are dynamically adjustable with the slider scale Fig. 7. After classifying them into E, A, M, and U percentage numbers of (E+A) students are calculated. If this number is above a certain set target, then the corresponding CLO is declared as “attained”. The target selection is also dynamically adjustable as per the policy requirements of the quality committee. The following screenshot of the slider, (Fig. 7) and the specimen calculations make the PSCA calculation clear:

Slider:



Fig. 7. Criteria Setting Slider for CLO Assessments.

As per the above slider, the numbers indicate the upper limits for each range. That means the ranges for each class are as follows (Table III):

TABLE III. TABLE OF SPECIFICATIONS (TOS) EXAMPLE

Unsatisfactory (U)	0-25
Medium (M)	26-60
Adequate (A)	61-75
Excellent (E)	76-100
E+A target criteria	76-100
Target criteria (Unsatisfactory)	0-60

Target criteria: 0-60 unsatisfactory; 61-75 satisfactory; 76-100: Attained.

The following Table IV presents values of different operational parameters for the example data of Table II with E,A,M and U classification. Fig. 8 presents comparison of PSCA(Pure) and PSCA(adjusted). In the “adjusted value computation the records of failed students are not considered. This PSCA(adjusted) parameter, slightly skews the results towards PLO attainment advantage. Fig. 9 shows final values.

3) *Grading*: Grade Book generation is dependent on the complete marks submission in all assessments as planned for the course. Incomplete submission will not produce the grade book. The mark submission is to be done at each question level at each assessment for all the registered students. Any lapse is considered as incomplete data submission and grade book will not be generated. This strategy is adopted to collect the data from the root of the assessment source alleviating any high-level manipulation or adjustments to alter the grades. The grade ranges used are as below in Table V.

AQASYS generates course specific grade summary charts for the faculty login and grade summary charts for all the courses of the department for the administrator login. The sample grade summary screen shot is as shown in Fig. 10 below:

4) *PLO Attainments*: Before calculating the PLO attainments, it is important to understand the relation between CLO, PLO and course Assessments of the department. The following Entity-relationship diagram, Fig 11, shows relation between entities: department, course, assessment, PLOs and CLOs.

It means that a department will have many courses, a course will have many assessments. A course will have many defined CLOs. Each assessment will have many CLOs. And finally, many CLOs contribute to many PLOs.

a) *PLO Attainment in the course (CPL0)*: Now it is clear that, the CLOs of a course will contribute to one or more PLOs. There is “many to many “ (m:n) relation between CLOs and PLOs of the course. Therefore, while calculating the PLO attainment by a course, the contributions of each CLO to the PLO in question are to be considered. The attainment in i^{th} PLO is given by the average sum of all the contributing CLOs. The general formula used is as given below:

$$CPLO(i) = \frac{\sum_{j=0}^k V(i)_{CLO=j}}{k}$$

Where V(i) is the value of jth CLO contributing to ith PLO.

TABLE IV. TABLE OF SPECIFICATIONS (TOS) EXAMPLE

	Q1	Q2	Q3	Q4	Q5
Max Marks	4	5	6	2	3
CLO	2.1	3.1	2.1	1.2	1.1
StudentID					
S101	2=50% (M)	1=20% (U)	3=50% (M)	2=100% (E)	3=100% (E)
S102	4=100% (E)	5=100% (E)	2=33% (M)	2=100% (E)	3=100% (E)
S103	4=100% (E)	2=40% (M)	5=83% (E)	1=50% (M)	3=100% (E)
S104	4=100% (E)	5=100% (E)	2=33% (M)	0=0% (U)	3=100% (E)
S105	3=75% (A)	3=60% (M)	6=100% (E)	0=0% (U)	3=100% (E)
S106	1=25% (U)	3=60% (M)	5=83% (E)	2=100% (E)	2=66% (A)
S107	0=0% (U)	5=100% (E)	6=100% (E)	1=50% (M)	3=100% (E)
S108	4=100% (E)	4=80% (E)	4=60% (A)	2=100% (E)	0=0% (U)
S109	3=75% (A)	1=20% (U)	5=83% (E)	2=100% (E)	3=100% (E)
S110	4=100% (E)	2=40% (E)	1=16% (U)	1=50% (M)	3=100% (E)
Total Students	10	10	10	10	10
PSCA (Pure)	70%	40%	60%	50%	90%
Comments (Pure)	satisfactory	Unsatisfactory	Unsatisfactory	Unsatisfactory	Attained
PSCA (Adjusted)	87.5%	77.7%	85.7%	87.5%	100%
Comments (Adjusted)	Attained	Attained	Attained	Attained	Attained
DL	27.5%	38%	35%	35%	13.3%
ODL	29.76%				
Final CLO Assessment Results and Comments					
	PSCA (Pure)	Comments PSCA (Pure)	PSCA (Adjusted)	Comments PSCA (Adjusted)	
CLO 1.1	90%	Attained	100%	Attained	
CLO 1.2	50%	Unsatisfactory	87.5%	Attained	
CLO 2.1	65%	Satisfactory	86.6%	Attained	
CLO 3.1	40%	Unsatisfactory	77.7%	Attained	

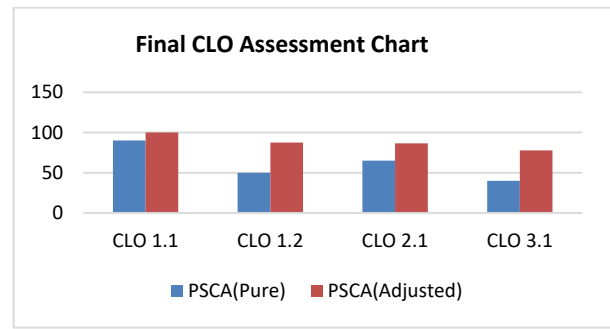


Fig. 8. Final Percentage CLO Attainments PSCA (Pure) and PSCA (Adjusted).

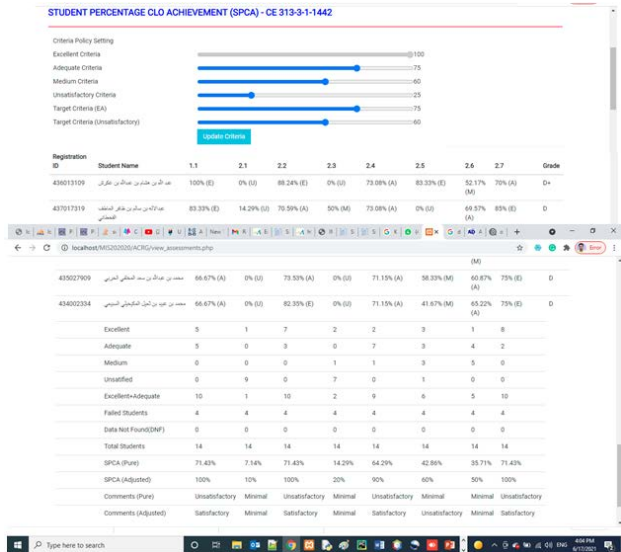


Fig. 9. Screen Shots from AQASYS: Final Percentage CLO Attainments PSCA (Pure) and PSCA (Adjusted).

TABLE V. GRADING POLICY AND RANGES

Range of Marks	Grade Assigned
100 >= Marks >= 95	A+
94 >= Marks >= 90	A
89 >= Marks >= 85	B+
84 >= Marks >= 80	B
79 >= Marks >= 75	C+
74 >= Marks >= 70	C
69 >= Marks >= 65	D+
64 >= Marks >= 60	D
59 >= Marks >= 0	F

b) PLO Attainment in the semester for the program (SPLO): This is the average of ith PLO attainment in all courses in the semester. The generalized formula that is used to calculate this parameter in AQASYS is as below:

$$SPLO(i) = \frac{\sum_{m=0}^N CPLO(i)}{N}$$

where, N is the total number of courses.

Fig. 12, Fig. 13, and Fig. 14, show instances of system results for this section.

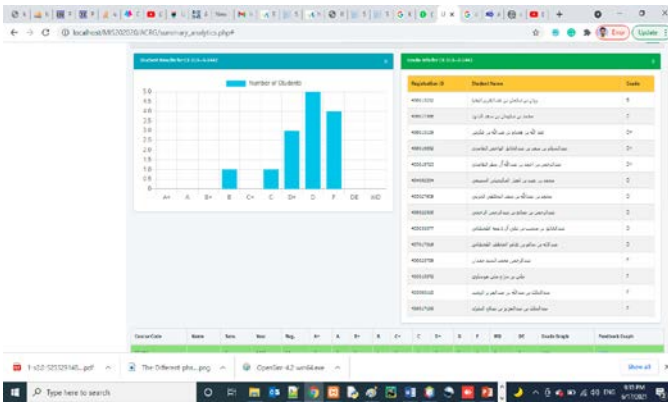


Fig. 10. Sample Grade Summary Chart Generated by AQASYS.

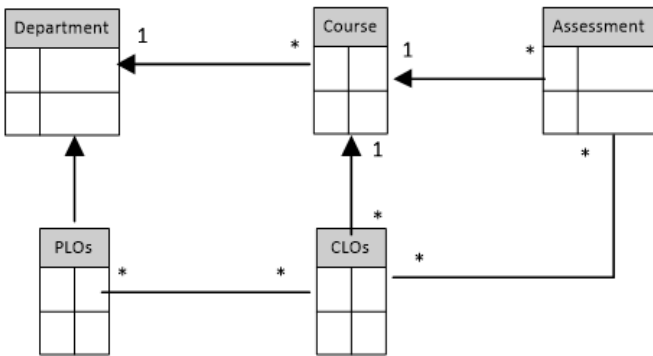


Fig. 11. Entity-Relationships in a PLO Calculation Scenario.

SUMMARY ANALYTICS

Department	Course code	CRN	Section	Semester	Year	Summary	Action
Civil	CE 313 - Reinforced Concrete Design		1	Second Se	144	Program PLO :	Go

Course code	Course CLOs	Section	Semester	Year	K1	S1	S2	S3	C1	C2	C3	PLO Attainment due to the course	Filtered PLO due to the course	
CE 210														
CE 211	2.1,2.3,2.4,2.5,2.6,2.8	1	Second Semester	1442		2.140 2.540 2.640						(S1) 2.1 = 40 (S2) 2.3 = 40 (S3) 2.4 = 40 (C1) 2.5 = 40 (C2) 2.6 = 40 (C3) 2.8 = 40	[?] 140/200 Graph View	
CE 213														
CE 221														
CE 221														
CE 241														
CE 241														
CE 241														
CE 311														
CE 313	1.1,2.1,2.2,2.3,2.4,2.5,2.6,2.7	1	Second Semester	1442		1.147.43 Avg(K1) = 71.43	2.147.14 2.271.43 2.342.58 2.428.57 2.547.43 Avg(S2) = 42.86	2.541.29 3.471.14 2.342.58 2.428.57 2.547.43				(S1) 2.1 = 71.43 (S2) 2.1 = 71.43 (S3) 2.1 = 71.43 (C1) 2.1 = 71.43 (C2) 2.1 = 71.43 (C3) 2.1 = 71.43	[?] 147.43 [?] 199.29 [?] 142.86 Graph View	
PLO attainment of the semester						71.43%	38.43%	42.86%				See PLO Graph link	****	
Filtered PLO Attainment of the semester						71.43%	36.31%	42.86%				See Filtered PLO Graph link	****	

Fig. 12. PLO Attainments (SPLO and CPLO).



Fig. 13. Graph Showing PLO Attainments (CPLO).



Fig. 14. Graph Showing PLO Attainments (SPLO).

5) *Difficulty Level(DL)*: This parameter is captured as Coefficient of Difficulty else-where[], however the parameter is not very convincing as it takes into account the only maximum scores. In this paper a new parameter is defined to capture the difficulty level. In this paper, the difficulty Level (DL) is the degree of difficulty of the assessment as felt by the students (the learner). It is defined as below:

$$DL = 1 - \frac{\sum_{i=0}^N M_i}{N * M_x}$$

Where M_i is the mark scored by i^{th} student, M_x is the maximum mark for the question (CLO) and N is the number of students. If all the students secure the allotted maximum mark then, the numerator of the fractional term in the above equation will be equal to its denominator resulting in the final value of $DL=0$, suggesting that the question paper was easy. If it is extremely difficult then all the students would score zero, resulting in $DL=1$ suggesting a very difficult question paper. The maximum value of DL is 1 and the minimum value is 0. DL may be expressed in a percentage value. Overall Difficulty Level (ODL) is the average of all the DL s.

6) *Survey collection & Analysis*: AQASYS collects the different kinds of surveys including course feedback surveys, alumni surveys and any survey as launched by the administrator. The system has two major provisions to upload the survey forms into the database.

- Through the excel sheets.
- Directly from the survey forms (example Google form).

After the due date of collection of survey data, the results are analyzed by the bar charts, pie charts, etc. Fig. 15 shows an instance of such a feedback graph as generated by the system.

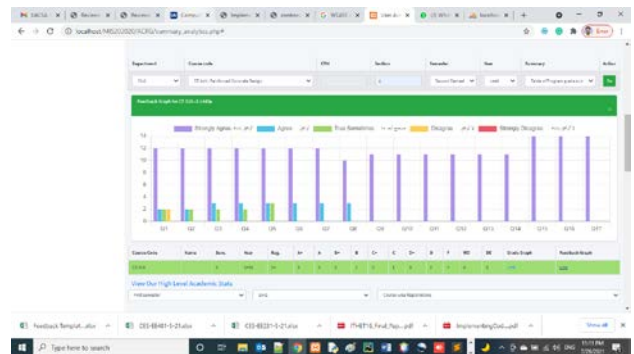


Fig. 15. Screen Shot of a Survey Graph Generated by AQASYS.

7) *CLO Mapping (CLOM)*: CLO mapping is the most important part of the quality assurance system. Generally, it was found that.

- During the curriculum design of most of the programs, the care that is due for defining the CLOs, is not given.
- While mapping the assessment questions to appropriate CLOs, the attitude and approach is very un-scientific to the educational philosophy. Most of the faculty sways hay.

The underpinning point here is that if an academic institution has symptoms as observed above, then the whole exercise of quality evaluation becomes just an exercise of number jargon that would remain far removed from the real state of academics in the organization.

Keeping these observations at the center of the design, AQASYS integrates an intelligent auto-mapping module called iBOX that assigns possible CLOs automatically to the questions of an assessment questionnaire. iBOX works on the text mining algorithms and takes into account Bloom's taxonomy to recognize action verbs.

a) iBOX Design: This section presents the design basics of iBOX (Fig 16). The goal is to first automatically identify the action verbs from the assessment questions and map them to CLOs, later map CLOs to the PLOs. The iBOX design explores Bloom's taxonomy and Action Verbs (refer to Table VI).

The iBOX algorithms, initially extract the action verbs from the given text document after verb extraction, the algorithms search Bloom's database for the matching verbs. If the matching verb is found in the database then, categorizes and label the document as per the scheme shown in Fig. 17, other- wise the algorithm allows human intervention to match the sentence with the action verb and Bloom's level. The algorithm updates Bloom's database with this newly learned

action verb. Thus iBOX is incrementally learning while it continues to automatically classify the CLO, PLO, and question sentences into different Bloom's levels. Fig. 16 shows the iBOX as it appears in AQASYS.

iBOX basically processes the input text files to identify the action verbs and match them to the verbs available in the Bloom database. If a proper match is found, it labels the CLO, PLO mapping automatically else it adds the new unmatched verb to the new verb list in the Bloom database. The new verb then becomes part of Bloom's verb list. The user is allowed to manually classify the new verb and insert it into an appropriate Bloom level list as an update. This way the iBOX follows incremental learning.

Three text files containing 1) The question paper 2) CLO definitions 3) PLO definitions are the inputs to a sentence segmentation and verb extraction module of iBOX.

This module uses Stanford parser [16] and wordNet database [8][15] to extract the exact verb from the sentences of the input documents. Table VII shows the decision making process that is inbuilt in the iBOX.

$V_q \rightarrow$ is a verb in the question paper

$V_c \rightarrow$ is a verb in the CLO definitions

$V_p \rightarrow$ is a verb in the PLO definitions

The extracted verbs are searched for their presence in the verb list of Bloom database. If they are present, then each verb (V_q, V_c, V_p) is compared with the members of the other verb lists corresponding to each Bloom's level (Remember, Understanding, Apply, Analyze, Evaluate, Design). If the verb under test is the same as the word in the other group then it is denoted as W_s (similar word), else if it is different, it is denoted as W_d (different word). It is to be noted that each of these verbs is already verified to be present in the Bloom database.

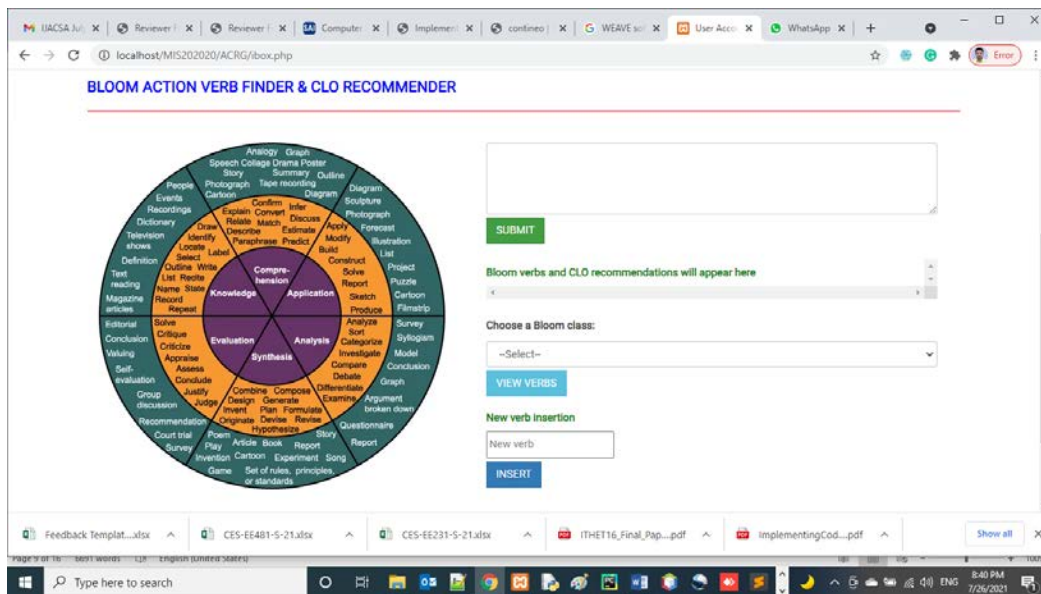


Fig. 16. iBOX-Bloom Level and CLO Recommender System.

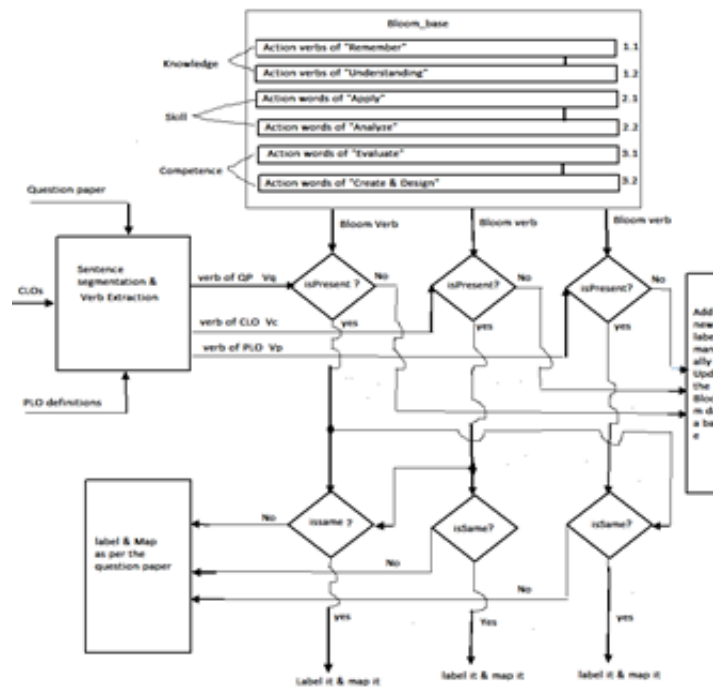


Fig. 17. Block Schematic of Decision making in iBOX Algorithms.

TABLE VI. ANDERSON’S REVISED BLOOM ACTION VERBS

Remember	Understand	Apply	Analysis	Evaluate	Design
Arrange	Classify		Analyze	Arrange	
Define	Convert		Appraise	Assemble	
Describe	Defend		Breakdown	Categorize	
Duplicate	Discuss	Apply	Calculate	Collect	Appraise
Identify	Distinguish	Change	Categorize	Combine	Argue
List	Estimate	Choose	Compare	Comply	Assess
Match	Explain	Compute	Contrast	Compose	Attach
Memorize	Express	Demonstrate	Criticize	Construct	Choose
Name	Extend	Discover	Diagram	Create	Compare
Order	Generalized	Dramatize	Differentiate	Design	Conclude
Outline	Give example(s)	Employ	Discriminate	Develop	Contrast
Recognize	Identify	Interpret	Distinguish	Devise	Defend
Relate	Indicate	Manipulate	Examine	Explain	Describe
Recall	Infer	Modify	Experiment	Formulate	Discriminate
Repeat	Locate	Operate	Identify	Generate	Estimate
Reproduce	Paraphrase	Practice	Illustrate	Plan	Evaluate
Select	Predict	Prepare	Infer	Prepare	Explain
State	Recognize	Produce	Model	Rearrange	Judge
	Rewrite	Relate	Outline	Justify	Interpret
	Review	Schedule	Point out	Reconstruct	Interpret
	Select	Show	Question	Relate	Relate
	Summarize	Sketch	Relate	Reorganize	Predict
	Translate	Solve	Select	Revise	Rate
		Use	Separate	Rewrite	Select
		Write	Subdivide	Set up	Summarize
			Test	Summarize	Support
				Synthesize	Value
				Tell	
				Write	

TABLE VII. DECISION MAKING IN iBOX

V _q	V _c	V _d	Labeling decision
W _s	W _s	W _s	L(W _s)
W _s	W _s	W _d	L(W _s)
W _s	W _d	W _s	L(W _s)
W _s	W _d	W _d	L(W _d)
W _d	W _s	W _s	L(W _s)
W _d	W _s	W _d	L(W _d)
W _d	W _d	W _s	L(W _d)
W _d	W _d	W _d	L(W _d)

The decision making uses the highest probability (dominant) principle and labels as per the maximum number of occurrences of the word. If W_s or W_d belongs to a specific Bloom’s list then the level of that list is marked and mapped to CLO and PLO along with the designated number labels. Example (Remember:1.1,Understanding:1.2,Apply:2.1, Analyze:2.2,Evaluation:3.1,Design:3.2) If the three verbs (V_q, V_c, V_p) that are present in the Bloom’s database with all being different and there is no similarity among them, then decision label will be marked as per the Bloom level of V_q, (refer Fig 17 and Table VII.). Table VIII shows the exact statements of SOs/PLOs and Table IX shows Bloom’s classification for those statements after recognizing action verbs and mapping them to the available verbs in the Bloom database.

TABLE VIII. SOs/PLOs:STATEMENTS

	Student Outcome (SO/PLO) Statement
a	An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline
b	An ability to analyze a problem, identify and define the computing requirements appropriate to its solution.
c	An ability to design , implements, and evaluate a computer-based system, process, component, or program to meet desired needs.
d	An ability to function effectively on teams to
e	An understanding of professional, ethical, legal, security and social issues and responsibilities
f	An ability to communicate effectively with a range of audiences.
g	An ability to analyze the local and global impact of computing on individuals, organizations, and society
h	Recognition of the need for and an ability to engage in continuing professional development.
i	An ability to use current techniques, skills, and tools necessary for computing practice.
j	An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
k	An ability to apply design and development principles in the construction of software systems of varying complexity

TABLE IX. BLOOM CLASSIFICATION INSTANCES FOR PLOS

SO's	Dominant action word identified	CLO Category	Recommended CLO Number	Member words from Bloom's table
a	apply	skill	2.1	Apply, Build Choose, Construct Develop Experiment With, Identify Interview Make, use Of, Model Organize Plan, Select Solve, Utilize
b	analyze	skill	2.2	Analyze, Assume Categorize Classify, Compare Conclusion Contrast Discover, Dissect Distinguish Divide, Examine Function, Inference Inspect, List Motive Relationships Simplify, Survey Take part in Test for, Theme
c	Design	competence	3.2	Adapt, Build Change, Choose Combine, Compile Compose, Construct, Create Delete, Design Develop, Discuss Elaborate, Estimate Formulate, Happen Imagine, Improve

SO's	Dominant action word identified	CLO Category	Recommended CLO Number	Member words from Bloom's table
				Invent, Makeup Maximize, Minimize Modify, Original Originate, Plan Predict, Propose Solution, Solve Suppose, Test Theory
d	function	knowledge	1.2	Classify, Compare Contrast Demonstrate Explain, Extend Illustrate Infer, Interpret Outline, Relate Rephrase, Show Summarize, Translate
e	understand	knowledge	1.1	Choose, Define Find, How Label, List Match, Name Omit, Recall Relate, Select Show, Spell Tell, What When, Where Which, Who Why
f	communicate	knowledge	1.1	Choose, Define Find, How Label, List Match, Name Omit, Recall Relate, Select Show, Spell Tell, What When, Where Which, Who Why
g	analyze	skill	2.2	Analyze, Assume Categorize Classify, Compare Conclusion Contrast Discover, Dissect Distinguish Divide, Examine Function, Inference Inspect, List Motive Relationships Simplify, Survey Take part in Test for, Theme
h	Recognize/engage	competence	3.1	Agree, Appraise Assess, Award Choose, Compare Conclude, Criteria Criticize, Decide Deduct, Defend Determine, Disprove Estimate, Evaluate Explain, Importance Influence, Interpret Judge, Justify Mark, Measure

SO's	Dominant action word identified	CLO Category	Recommended CLO Number	Member words from Bloom's table
				Opinion, Perceive Prioritize, Prove Rate, Recommend Rule on, Select Support, Value
i	Use	knowledge	1.2	Classify, Compare Contrast Demonstrate Explain, Extend Illustrate Infer, Interpret Outline, Relate Rephrase, Show Summarize, Translate
j	Apply	skill	3.1	Apply, Build Choose, Construct Develop Experiment With, Identify Interview Make, use Of, Model Organize Plan, Select Solve, Utilize
k	Apply	skill	3.1	Apply, Build Choose, Construct Develop Experiment With, Identify Interview Make, use Of, Model Organize Plan, Select Solve, Utilize

8) *Course report generation:* Utilizing all the information available about the course in the database AQASYS generates the academic course report in the required format automatically in a single click.

It intelligently inserts recommendations in the report wherever necessary. The automatic course report generation feature makes AQASYS unique from other competitive research in this field. Fig. 18 shows the user interface for course report generation and Fig. 19 shows the instance of the course report being generated.

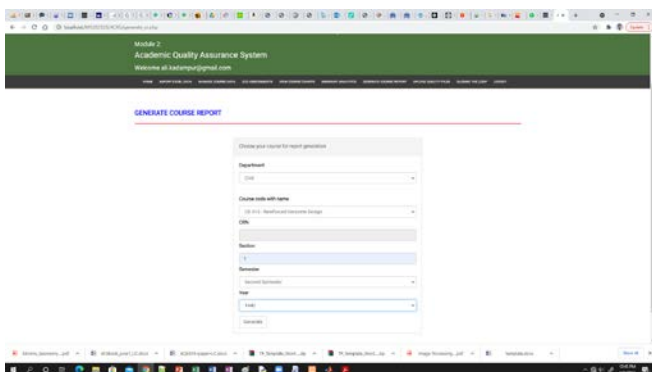


Fig. 18. Interface for Generating Course Report.

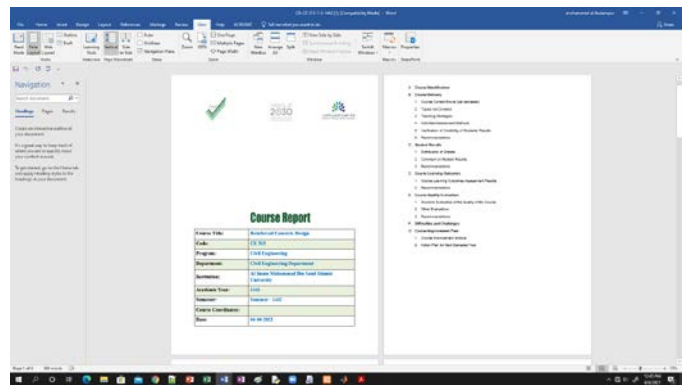


Fig. 19. Instance of System Generated Course Report.

IX. RESULTS AND SCREENSHOTS

Instances of AQASYS results are already presented in earlier sections wherever the discussion arose. In this section, few computed operational parameters are compared with manually computed values. Table X is the partial data reference table for CLO 1.1. The system computed parameters exactly match the expected values. Refer to results in Table XI.

TABLE X. EXAMPLE MARKS OF 14 STUDENTS FOR CLO 1.1

CLO 1.1	2	3	2.5	3	2	2	3
Max Marks:3	3	3	0	2	3	2	2

TABLE XI. PARAMETERS COMPARED

SNo	Parameter	Manually computed value	System generated value
1	PCA (Pure); 1.1	77.4%	77.4%
3	PSCA (Pure)	92.6%	92.6%
4	PSCA(adjusted)	100%	100%
5	DL	22.61%	22.61%
Verified parameters			
7	GRADE summary	✓	✓
8	PLO attainment at courses level	✓	✓
9	PLO attainment at Semester level	✓	✓
10	Filtered PLO attainment at course level	✓	✓
11	Filtered PLO attainment at Semester level	✓	✓

Fig. 20 shows instance of grade results and Fig. 21 shows instance of PLO attainments.

1) *AQASYS Performance evaluation:* In order to evaluate the performance of the application, the following few application performance metrics were considered. Refer Table XII. The Appdex metric, as defined below:

$$Appdex = \frac{satisfiedcount + \frac{ToleratingCount}{2}}{Totalsamples}$$

was considered to evaluate “user satisfaction”.

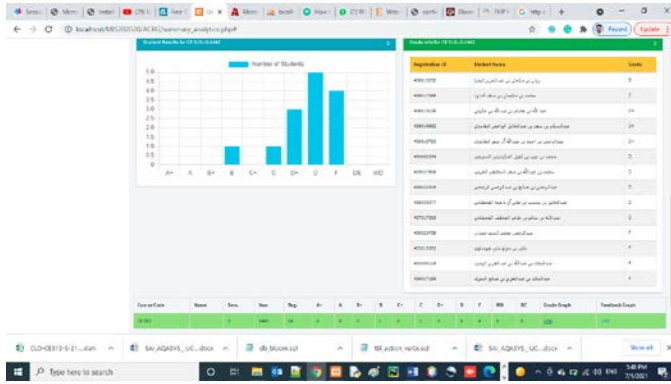


Fig. 20. Instance of Grade Results.

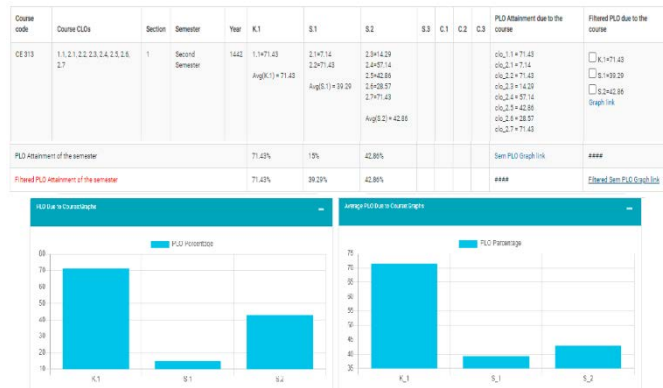


Fig. 21. Instance of PLO Attainments.

TABLE XII. APPLICATION PERFORMANCE METRICS

SNo	Parameter	Value
1	App user total count	48
2	App user satisfied count	41
3	Tolerating count	7
4	Appdex	0.927
5	Observed max number of concurrent users	48
6	Application availability	100% (Pinged all the time in a 72 hours of monitoring)

The count of concurrent users was observed to understand the “request rate” and the load that the application can handle. The application was found to be performing quite satisfactorily. “Availability” of web-application like the one in this paper is very important from the user's perspective. The application was “pinged” at an interval of a minute regularly for a duration of 120 hours. The log results showed successful “ping” all the time. The application was highly available.

2) AQASYS selected screenshots: The following figures, Fig 22 through Fig. 28, show various instances of AQASYS output as depicted against their names.

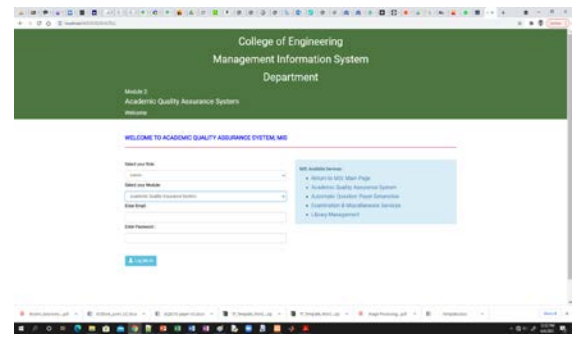


Fig. 22. Instance of Login Page.

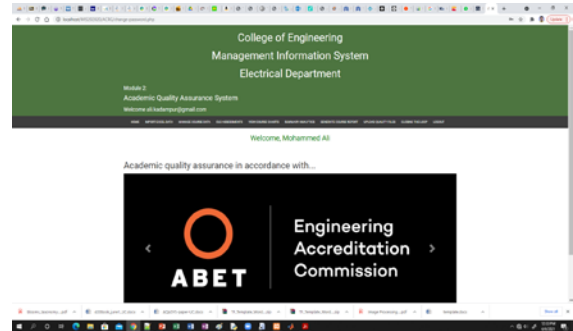


Fig. 23. Instance of Home Page.

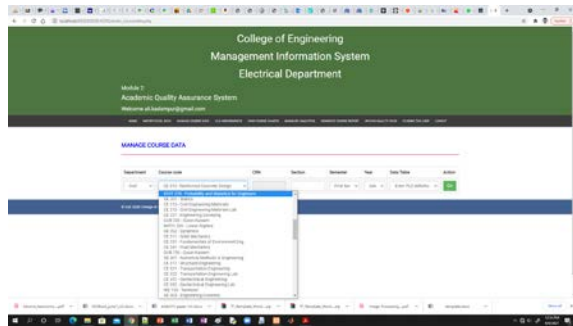


Fig. 24. Instance of Course Selection.

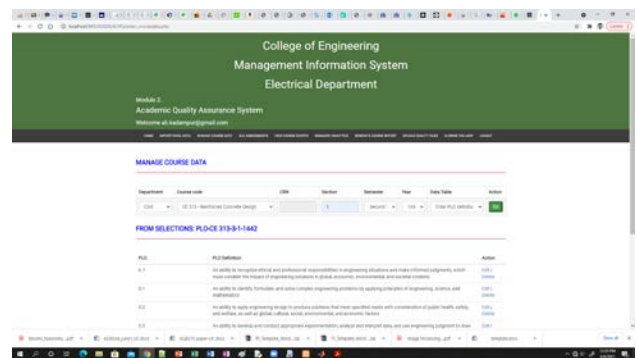


Fig. 25. Instance of CLO Definitions.

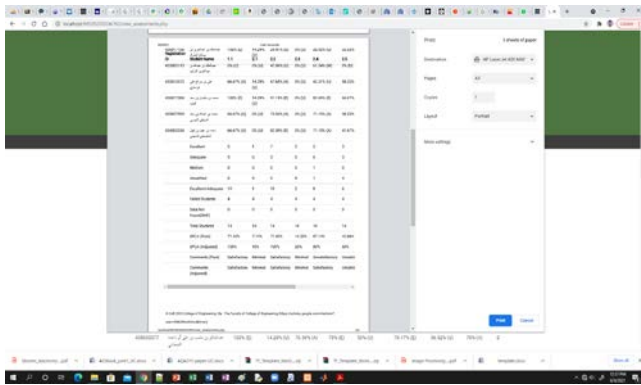


Fig. 26. Instance of PSCA Computations.

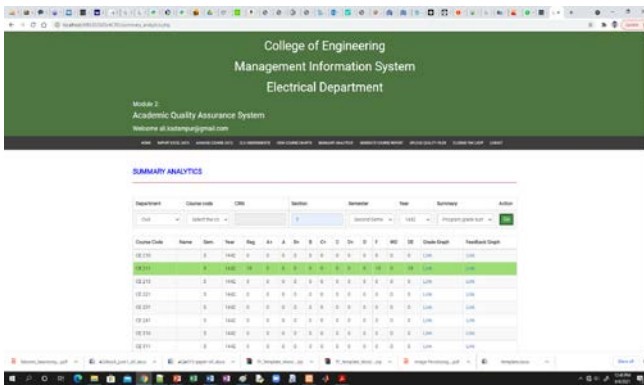


Fig. 27. Instance of Summary Analytics.

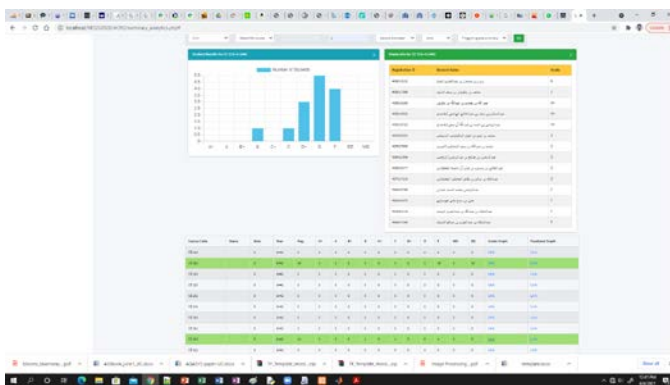


Fig. 28. Instance of Grade Summary Display.

X. CONCLUSION

This paper presented a web-based academic quality assurance system. The unique features of the system included automatic course report generation and intelligent Bloom's level and CLO-PLO mapping module, that automatically assigns appropriate Bloom levels to the text presented to it by matching the action verbs. The paper presented the content from the system designer's perspective as well as from the user's perspective. The system was launched in the institutional LAN on a pilot basis and real-time data was loaded onto the system. Different operational parameters such as PCA, PSCA, DL, Course level PLO attainment (CPLO), Semester PLO attainment (SPLO), course-wise grade details, and other summary analytic were tested. The computed

parameters were found to be as expected in the quality standards.

This system would help the faculty, quality committee, and the administration to measure the learning quality in the selected program according to the ABET and NCAAA standards. The faculty need not worry about the clumsy calculation formulae and heaps of excel-sheets and run around seeking last-minute templates to stitch the quality. The faculty now can focus on teaching, followed by assessing and entering the scores of the students. All the calculations related to quality are carried out by the system automatically. The system provides summary charts and tabulates data outputs. As the application uses a client-server architecture, the server database archives the historical data and is available for viewing at any point in time.

The application makes the matters of handling of academic quality, a seamlessly an easy task with features of reliability, security, availability, and privileged access being embedded into the system.

ACKNOWLEDGMENT

The authors acknowledge the support of department heads and colleagues of the college of engineering Imam Mohammad Ibn Saud Islamic University, Riyadh, KSA. Particular thanks are extended to Dr. Aun Bashir, Dr. Farid Fadellah, Dr. Ali Hennache, Dr. Rashid Khan for their support in providing data and testing the application.

REFERENCES

- [1] Burge LL, Leach RJ. An advanced assessment tool and process. In Proceedings of the 41st ACM technical symposium on Computer science education 2010 March 10: 451-454. ACM.
- [2] Costreie S, Ianole R, Dinescu R. An Evaluation of the Quality (Assurance) Evaluation-Case Study: The University of Bucharest. Quality Assurance.
- [3] Khwaja Amir. A Web-Based Program Outcome Assessment Tool. 2018 21st Saudi Computer Society National Computer Conference (Ncc). 345 E 47th St, New York, Ny 10017 Usa: IEEE; 2018.
- [4] E. Essa, A. Dittrich, S. Dascalu, and F. C. Harris Jr., "ACAT: A WebBased Software Tool to Facilitate Course Assessment for ABET Accreditation," Proceedings of the 2010 Seventh Intl. Conf. on Info. Tech.: New Generations (ITNG), April 12-14, 2010, pp. 88-93.
- [5] ABET (Accreditation Board for Engineering and Technology). (2015). "Criteria for accrediting engineering programs." (<http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2016-2017>) (Nov. 9, 2015).
- [6] Ketchman K, Dancz CLA, Burke RD, Parrish K, Landis AE, Bilec MM. Sustainable Engineering Cognitive Outcomes: Examining Different Approaches for Curriculum Integration. J Prof ISSUES Eng Educ Pract. 2017;143. [https://doi.org/10.1061/\(ASCE\)EI.1943-5541.0000324](https://doi.org/10.1061/(ASCE)EI.1943-5541.0000324).
- [7] Development reference and learning resources from : <https://phpgurukul.com/codeigniter-introduction/>.
- [8] Silveira, N., Dozat, T., Marneffe, M., Bowman, S., Connor, M., Bauer, J., & Maning, C.: A Gold Standard Dependency Corpus for English. Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014), 2897-2904 (2014).
- [9] S. A. ElRahman and S. S. Shabanah, "Course and Student Management System Based on ABET Computing Criteria," I. J. of Info. Engineering and Electronic Business, 3, 2016, pp. 1-10.
- [10] Accreditation," Proceedings of the 2010 Seventh Intl. Conf. on Info. Tech.: New Generations (ITNG), April 12-14, 2010, pp. 88-93.
- [11] A. Khwaja, B. Alfalah, F. Alharbi, and R. Alarfaj, "WebCOAT: A Web-Based Course Outcome Assessment Tool," Proc. of the 2016

- GlobalConf. on Teaching and Learning with Tech. (CTLT 2016), June 16-17, 2016, pp. 215-220.
- [12] Center for education innovation Hong Kong: <http://cei.hkust.edu.hk/teaching-resources/outcome-based-education/school-resources/school-business-and-management>.
- [13] Programs and Courses Outcomes System (PACOS). <http://pacos.ust.hk/> [Last Retrieved: Feb 20, 2018].
- [14] Jamie Rumbelow "The codeigniter handbook " Efendi Publishing, Vol-1, ISBN: 978-0-9571791-1-0.
- [15] G. Miller, R. Beckwith, +2 authors K. Miller "Introduction to WordNet: An On-line Lexical Database" International Journal of Lexicography 1990, DOI:10.1093/IJL/3.4.235 Corpus ID: 2146137.
- [16] Christopher D. Manning, M. Surdeanu, John Bauer, J. Finkel, Steven Bethard, David McClosky " The Stanford CoreNLP Natural Language Processing Toolkit " ACL publisng 2014, DOI:10.3115/v1/P14-5010Corpus ID: 14068874.
- [17] Accreditation Support Software <https://www.smart-accredit.com/index.php>.
- [18] Outcome Based Education Software <http://www.contineo.in/>.
- [19] Weave education <https://www.capterra.com/accreditation-management-software/>.