World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Web-based program outcomes assessment and evaluation system with data analytics

Anna Marie M. Lamis and Jeffric S. Pisuena *

College of Computing and Information Sciences, State University of Northern Negros, Philippines.

World Journal of Advanced Research and Reviews, 2026, 29(03), 481-492

Publication history: Received on 31 January 2026; revised on 06 March 2026; accepted on 09 March 2026

Article DOI: https://doi.org/10.30574/wjarr.2026.29.3.0572

Abstract

Outcomes-Based Education requires higher education institutions to systematically monitor the attainment of program outcomes using credible and integrated evidence. However, many institutions still rely on fragmented data sources such as spreadsheets, paper records, and isolated digital tools, which limit timely interpretation and data-driven decision making. This study developed and evaluated a Web-Based Program Outcomes Assessment and Evaluation System with Data Analytics designed to centralize outcomes assessment data and support evidence-based academic program management. The system integrates multiple assessment sources including course grades, student surveys, employer competency evaluations, and licensure examination indicators within a unified relational database and web platform. Using Agile Software Development Life Cycle, the system was implemented as a full-stack web application that provides role-based access, automated reporting, interactive dashboards, and rule-based analytics for monitoring program outcomes attainment. Evaluation was conducted with 35 respondents consisting of administrators, students, and industry partners. Software quality was assessed using McCall's Software Quality Model, yielding an overall mean score of 4.96 indicating very good system quality. Usability evaluation using the Computer System Usability Questionnaire produced an overall mean of 6.03, reflecting positive user acceptance. The results demonstrate that the developed system effectively supports efficient outcomes assessment, centralized data management, and data-informed decision making in higher education environments.

Keywords: Outcomes-Based Education; Program Outcomes Assessment; Data Analytics; Web-based Information System; Higher Education Quality Assurance

1. Introduction

Outcomes-Based Education (OBE) and accreditation-oriented quality assurance require academic programs to demonstrate, through credible and traceable evidence, the extent to which students attain defined Program Outcomes (POs). In this framework, assessment is not limited to documenting student performance at the course level; rather, it involves consolidating multiple forms of evidence across courses, terms, and cohorts to determine whether the intended competencies of a program are being achieved. In the Philippine higher education context, this orientation is reinforced by the Commission on Higher Education through CHED Memorandum Order No. 46, series of 2012, which institutionalized an outcomes-based and typology-based quality assurance framework for higher education institutions. Similar reforms in other higher education systems have likewise emphasized measurable outcomes, accountability, and alignment between curricula, assessment practices, and workforce expectations [1, 2].

* Corresponding author: Jeffric S. Pisuena

Copyright © 2026 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution License 4.0.

In practice, however, the generation of PO evidence is often complex because it draws from heterogeneous sources, such as course grades, embedded assessment artifacts, student self-assessment surveys, and employer or industry feedback. These data are frequently collected across multiple academic periods and stored in fragmented spreadsheets, paper-based records, or isolated software tools. While such approaches may satisfy minimum documentation requirements, they often create recurring challenges in data completeness, traceability, consistency, and timely interpretation. Fragmented assessment processes can delay curricular interventions, weaken transparency in decision-making, and limit the ability of program administrators to monitor attainment trends longitudinally. These constraints are particularly significant in computing-related programs, where continuous curriculum refinement is necessary to keep pace with changing technical and professional standards.

To address these limitations, higher education institutions increasingly adopt web-based assessment and information systems that centralize data capture, automate reporting, and support more efficient outcomes monitoring. Research on digital and online assessment environments shows that such systems can improve the efficiency of assessment workflows, enhance feedback processes, and provide more structured support for monitoring student engagement and performance[3, 4]. Web-based assessment platforms are particularly valuable when they enable institutions to move beyond static record keeping toward integrated evidence management and data-informed interpretation of program performance [5]. In addition, the broader shift toward digital assessment in higher education, accelerated by post-pandemic educational transformation, has reinforced the importance of robust, accessible, and analytically capable platforms for academic evaluation and quality assurance [6, 7].

The growing integration of analytics into educational systems has further expanded the potential of outcomes assessment by enabling institutions to convert large volumes of assessment data into actionable insights. Learning analytics can support the identification of attainment patterns, emerging performance risks, and evidence gaps that may otherwise remain hidden in manually managed records. Studies have shown that analytics-enhanced systems can strengthen the interpretation of educational data, improve the monitoring of learning outcomes, and support more evidence-based academic decision-making [8, 9]. Likewise, digital assessment research suggests that the use of technology in classroom and program evaluation can improve responsiveness, reporting efficiency, and the overall quality of monitoring processes [10]. When aligned with OBE principles, such analytics capabilities may help institutions compare expected learning outcomes with actual attainment results and thereby support continuous quality improvement at the program level [11].

Despite these developments, many existing outcomes assessment implementations remain limited to data entry and report generation, with insufficient emphasis on integrating multiple evidence sources into a coherent PO attainment view, delivering interactive role-based analytics for different stakeholders, and producing exportable documentation. This gap is important because an outcomes assessment system is most useful not merely when it stores records, but when it enables administrators, program heads, and faculty to interpret evidence efficiently and act on it with confidence. Moreover, the value of such systems depends not only on their functional capabilities but also on their technical quality, maintainability, and usability for diverse user groups. Against this backdrop, the present study develops a Web-Based Program Outcomes Assessment and Evaluation System with Data Analytics that centralizes PO-related evidence, supports role-based assessment functions, and provides dashboards and rule-based decision support for interpreting attainment status, performance risks, and data collection priorities. In doing so, the study contributes a practical and evaluable digital infrastructure for supporting evidence-based continuous improvement, transparency, and accreditation-oriented program management in higher education.

1.1. Objectives of the Study

The main objective of this study is to develop a Web-Based Program Outcomes Assessment and Evaluation System with Data Analytics.

Specifically, it aims to:

- Develop a centralized system for collecting, managing, and evaluating program outcomes data;
- Design and integrate a data analytics and reporting module that provides visualization and interpretation of program outcomes for continuous quality improvement;
- Evaluate the system's quality using McCall's software quality model in terms of product operation, product revision, product transition;
- Evaluate the system's usability using the Computer System Usability Questionnaire.

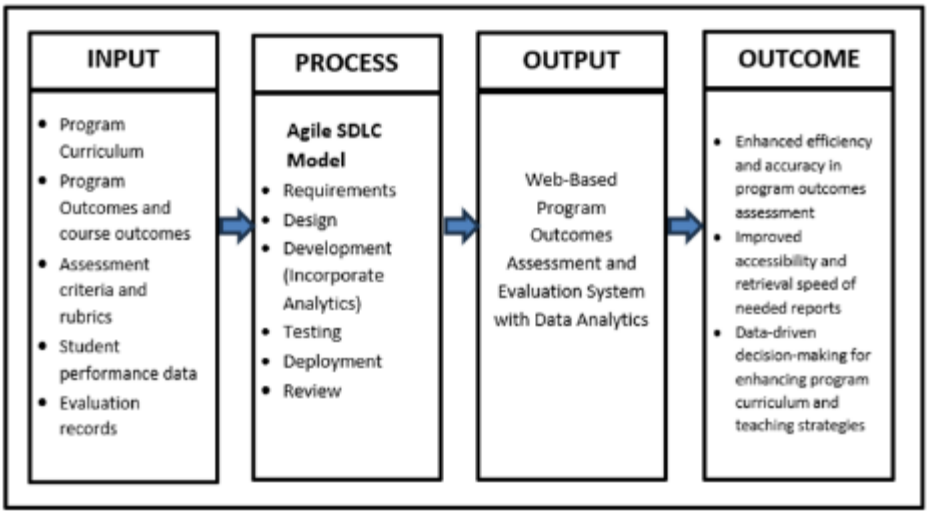

Figure 1 Conceptual Framework

Figure 1 presents an Input–Process–Output–Outcome (IPO) model that serves as the blueprint for developing the Web-Based Program Outcomes Assessment and Evaluation System with Data Analytics. The inputs consist of vital academic and institutional data, including the program curriculum, defined program and course outcomes, assessment criteria, student performance data, and evaluation records, which provide the foundation for system development. Guided by the Agile Software Development Life Cycle (SDLC) model, the process involves iterative stages of data gathering and analysis, system design, development, testing, deployment, and maintenance to ensure functionality and adaptability. The output is a functional web-based system that automates the assessment and evaluation of program outcomes while integrating data analytics for real-time monitoring and reporting. Ultimately, the outcomes demonstrate the system's effectiveness in enhancing efficiency and accuracy in outcomes assessment, improving accessibility and report retrieval speed, and promoting data-driven decision-making for continuous improvement of program curricula and teaching strategies.

2. Materials and Methods

2.1. Research Design

This study employed a descriptive developmental research design to guide the development and evaluation of the Web-Based Program Outcomes Assessment and Evaluation System with Data Analytics for the College of Computing and Information Sciences. The descriptive component evaluated system quality and usability by collecting feedback from faculty and program chairs using the Computer System Usability Questionnaire and McCall's Software Quality Model. The developmental component followed the Agile based Software Development Life Cycle, which included planning, requirements analysis, system design, development, testing, deployment, and refinement. This combined approach ensured that the developed system was functional and aligned with the requirements of outcomes-based education assessment and evaluation.

2.2. Research Instruments

Two standard instruments were used for the system evaluation in this study. The first is the McCall's Software Quality Model, which serves as the framework for evaluating software quality across three key aspects: product operation, which includes reliability, efficiency, and usability; product revision, which covers maintainability and testability; and product transition, which involves portability and flexibility. The second instrument is the Computer System Usability Questionnaire (CSUQ), developed by IBM, which was utilized to measure the system's usability in terms of system usefulness, information quality, interface quality, and overall user satisfaction.

2.3. Participants and Sampling

The respondents consisted of key stakeholders from the College of Computing and Information Sciences of the State University of Northern Negros. A total of 35 participants were involved, including one dean, three program heads, 29 students, and two employers. Academic administrators evaluated system functionality and institutional alignment, students assessed usability and user experience as primary end users, and employers provided industry perspectives

on workforce relevance. Respondents were selected through purposive sampling to ensure that participants had relevant expertise and familiarity with the system's operational context.

2.4. Data Gathering Procedure

The researchers first conducted a review of the existing manual assessment procedures to identify operational challenges and limitations. Based on the findings, the system was developed and subsequently subjected to evaluation. Two evaluation instruments were utilized in the study. McCall's Software Quality Model–based instrument was administered to expert evaluators to assess the quality attributes of the system, while the Computer System Usability Questionnaire (CSUQ) was distributed to end users to measure system usability and user satisfaction.

After the administration of the instruments, the collected responses were compiled and analyzed using descriptive statistical techniques, specifically the mean and standard deviation, to determine the overall evaluation of the system in terms of software quality and usability. These statistical measures were used to interpret the respondents' perceptions and provide quantitative support for the system evaluation results.

2.5. Software Development Method

Figure 2 The Agile Software Development Model

The development of the Web-Based Program Outcomes Assessment and Evaluation System with Data Analytics followed the Agile Software Development Life Cycle (SDLC) model, which emphasizes adaptive planning, iterative development, early delivery, and continuous improvement. The Agile model was chosen due to its flexibility and ability to accommodate user feedback during the development process ensuring that the system evolves based on the stakeholders' needs and functional requirements.

2.5.1. Requirements Phase

The requirements phase identified and documented the functional and non-functional requirements of the Program Outcomes Assessment and Evaluation System with Data Analytics. Existing program outcomes monitoring procedures at the College of Computing and Information Sciences of the State University of Northern Negros were reviewed, and consultations with the Dean and Program Heads were conducted to understand current workflows and data requirements. Based on these inputs, key system functions were defined, including user authentication, role-based access control, survey management, response submission, employer competency evaluation, and analytics reporting. The system was designed to support multiple user roles and incorporate nonfunctional requirements such as secure access, efficient database management, responsive interface design, and reliable data processing for analytics and reporting.

2.5.2. Design Phase

The design phase focused on translating the identified system requirements into a structured blueprint that guided the development of the PO Assessment and Evaluation System with Data Analytics. This phase defined the system structure, data organization, and interaction among system components to ensure that the implemented solution would meet the functional and operational requirements identified during the requirements phase.

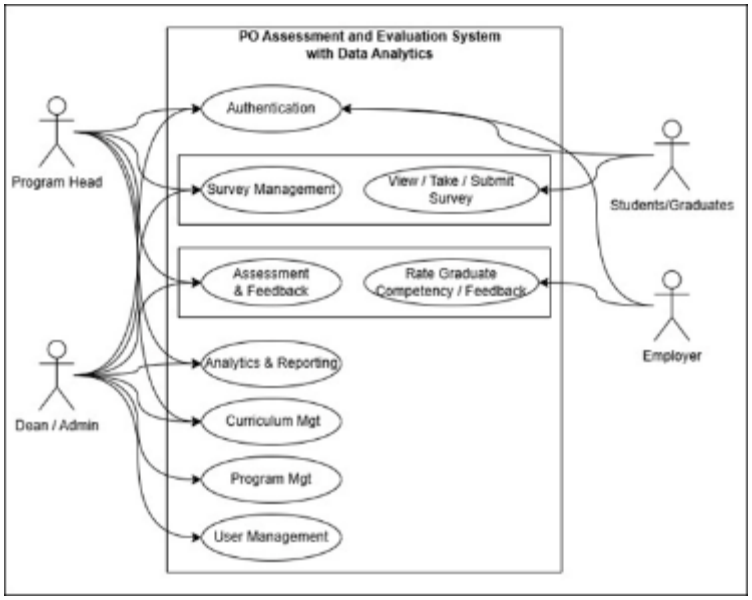

Figure 3 Use-Case Diagram

The use case diagram illustrates the interaction between users and the Program Outcomes Assessment and Evaluation System with Data Analytics. The main actors include the Dean or Administrator, Program Head, Students or Graduates, and Employers, each performing specific roles within the platform through secure role based authentication. Administrators and Program Heads manage core operations such as survey administration, assessment monitoring, analytics and reporting, and user management. Students and graduates submit survey feedback on program outcomes, while employers provide competency evaluations of graduates. Collected data are processed by the analytics module to generate reports and insights that support evidence-based decision making and continuous program improvement.

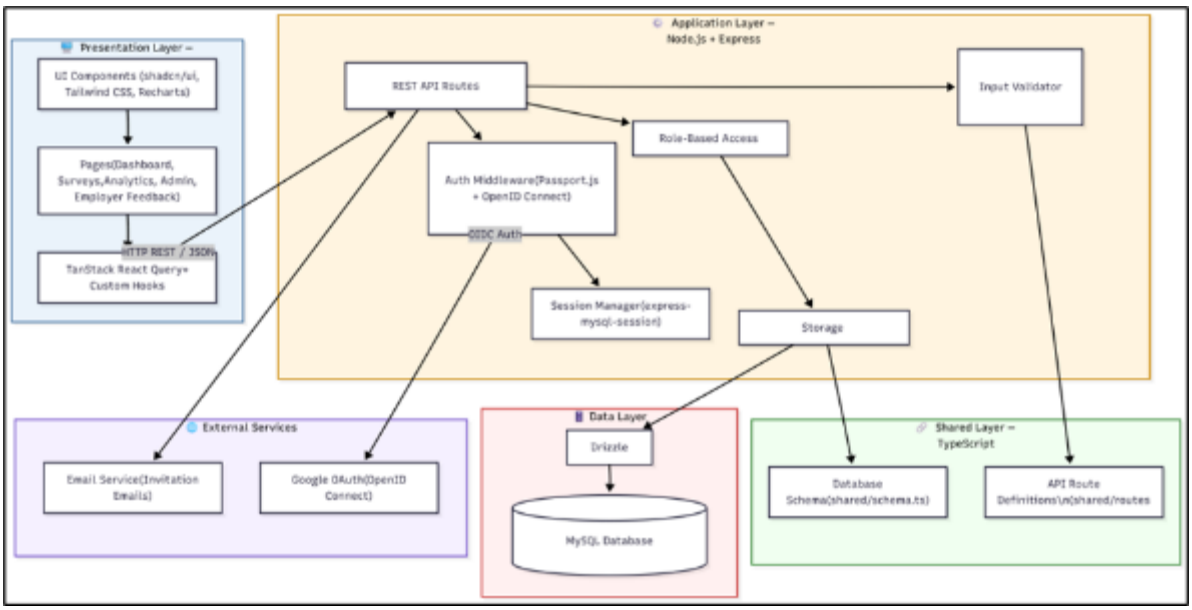

Figure 4 System Architecture

The system architecture follows a layered design consisting of presentation, application, data, and external service layers to enhance modularity, scalability, and maintainability. The presentation layer manages the user interface using React, ShadCN UI components, Tailwind CSS, and Recharts for data visualization, providing dashboards, survey management pages, analytics views, administrative modules, and employer feedback interfaces. Client server communication is handled through HTTP requests using TanStack React Query and custom hooks for efficient data retrieval and state management.

The application layer is implemented using Node.js with the Express framework, which handles core business logic, REST API services, request processing, role based access control, and authentication through Passport.js integrated with OpenID Connect. Session management is supported through express-mysql-session storage, while input validation ensures that all submitted data conforms to defined formats before processing.

The data layer utilizes a MySQL relational database to store academic records, survey responses, competency evaluations, and program outcome data, with database operations managed through Drizzle ORM to maintain schema consistency and efficient query handling. A shared TypeScript module centralizes database schema definitions and API route configurations to ensure consistent data structures across system components. Additionally, external services such as Google OAuth for identity verification and an email service for registration and system notifications are integrated to support secure authentication and communication.

2.5.3. Development Phase

The development phase involved the implementation of the PO Assessment and Evaluation System with Data Analytics based on the requirements and system design produced in the earlier phases. Following the Agile development approach, system components were implemented incrementally to ensure that each module functioned correctly and integrated properly with other system components. The development process focused on translating the system design into functional modules capable of supporting program outcomes assessment and evaluation.

System Implementation

The development of the Web-Based Program Outcomes Assessment and Evaluation System followed a structured Software Development Life Cycle to ensure systematic translation of institutional requirements into a functional information system. The SDLC framework organized the process into defined phases that supported iterative refinement and controlled implementation of system components [12]. During development, the platform was implemented as a full-stack web application using TypeScript across both client and server environments to maintain consistent data structures and improve maintainability. The client side interface was developed using React with component based modules for dashboards, administrative pages, survey interfaces, and analytics reporting views, while routing and query-based data fetching were used to synchronize interface states with server data. On the server side, Node.js with the Express framework handled application logic, authenticated requests, input validation, and coordination between system modules. RESTful application programming interfaces were implemented to manage operations such as retrieving and storing program outcome records, survey responses, employer feedback, and analytics summaries, following widely adopted service-based communication practices in web systems [13]. Data persistence was implemented through a MySQL relational database that stored users, program outcomes, course records, survey responses, employer competency ratings, and analytics outputs, with database operations managed through a dedicated data access layer to maintain separation between routing logic and persistence functions. Several non functional considerations were integrated during implementation, including authentication, role-based access control, and centralized schema definitions to support security, maintainability, and consistent data handling[14]. An integrated analytics module was also developed to generate dashboard visualizations and downloadable reports that support academic decision making and accreditation documentation. The completed system was deployed in a web server environment with configuration parameters managed through environment variables, enabling authorized stakeholders to access the platform through a standard web browser.

Data Analytics Module

The developed system incorporates a centralized data analytics and reporting module to support evidence-based monitoring of program outcomes. The analytics component operates through an authenticated application programming interface that retrieves structured data with optional filters such as program identifier and reporting year. Access is controlled through role-based authorization to ensure that program heads can only view analytics related to their assigned programs. Reporting years are dynamically derived from institutional data sources including course grade records, survey submissions, employer competency evaluations, and licensure examination data. Program outcome analytics are generated by integrating evidence from these repositories and aligning them with predefined course to program outcome mappings. Academic performance indicators are calculated using mean scores, and percentage-based grades are normalized to a five-point scale to ensure consistency with survey and employer evaluation ratings using linear transformation.

$$g_{0-5} = \left(\frac{g_{0-100}}{100}\right) \times 5$$

The system aggregates grades, survey feedback, and employer competency assessments to compute composite program outcome scores that are presented through dashboard scorecards and comparative reports. Descriptive statistics such as mean, variance, and standard deviation are also computed to support benchmarking, while Pearson correlation coefficient analysis was applied to examine relationships between assessment sources and overall outcome performance. In addition, the analytics module applies rule-based detection to identify low performing outcomes, declining trends, insufficient data samples, and inconsistencies across assessment sources. Outcome scores are finally categorized into attainment levels ranging from Highly Attained to Not Attained, enabling clearer interpretation for academic evaluation and decision making.

Algorithm for Rule-Based Program Outcomes Analytics

To support automated interpretation of Program Outcome indicators, the system employs a rule-based analytics algorithm that evaluates aggregated assessment data and generates recommendations for the analytics dashboard. The algorithm processes program outcome records derived from multiple sources including course grades, survey based self-assessments, employer competency evaluations, and board examination results when available. Each record contains computed indicators such as average scores, observation counts, and overall outcome attainment values, which are analyzed together with temporal trend data to identify patterns requiring academic intervention or program improvement. The procedure first computes descriptive statistics to establish baseline performance levels and classify outcomes into critical, moderate, or highly attained categories. It also evaluates the adequacy of available data by examining the number of records supporting each indicator and flags outcomes with insufficient evidence. Analytical checks are then applied to examine relationships and inconsistencies among assessment sources. Pearson correlation analysis identifies which evidence sources show the strongest association with overall outcome performance, while cross source variance analysis detects discrepancies between grades, surveys, and employer evaluations. Temporal trend analysis further compares recent indicators with historical values to identify significant performance increases or declines. When notable deviations exceed predefined thresholds, the system generates prioritized recommendations to support data-driven academic decision making and continuous program improvement.

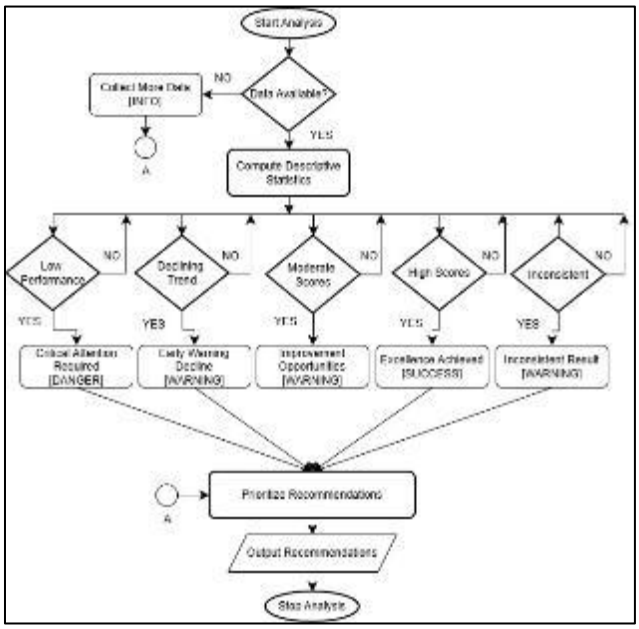

Figure 5 Analytics Decision Flow

Figure 5 shows the decision flow of the rule-based PO analytics algorithm showing sequential checks for data availability, attainment classification (low, moderate, high), sufficiency screening, correlation-based insight generation, cross-source inconsistency detection, trend alerts, and prioritization for dashboard output.

2.5.4. Testing Phase

The testing phase evaluated the quality, functionality, and usability of the developed Web-Based Program Outcomes Assessment and Evaluation System to determine whether it met required performance and user acceptance standards. Two evaluation instruments were employed. McCall's Software Quality Model was used to assess the technical quality of the system, including attributes such as accuracy, reliability, security, operability, and overall performance. This

evaluation was conducted by expert reviewers who examined the system's functional behavior and compliance with established software quality principles. In addition, the Computer System Usability Questionnaire was administered to end users to measure perceived usability, system usefulness, information quality, and interface quality using a seven-point Likert scale. The combined use of these instruments enabled a comprehensive assessment of both technical performance and user experience, ensuring that the system met software quality standards while providing an effective and user friendly platform for program outcomes assessment.

2.5.5. Deployment Phase

Upon completion of all development sprints and testing activities, the system was deployed on a web server environment for pilot implementation. The feedback gathered during this phase was used to enhance usability and refine analytical reports. Continuous monitoring and maintenance was done to ensure system reliability and to accommodate future enhancements.

3. Results and Discussion

The developed Web-Based Program Outcomes Assessment and Evaluation System successfully implemented a centralized platform for collecting, managing, and evaluating program outcomes evidence within a unified information system. The application integrates multiple data sources associated with outcomes assessment and consolidates them within a single relational database environment. Through a role-based web interface, authorized users can manage core academic entities such as programs, courses, and program outcomes, while also recording evidence used for outcomes evaluation.

The system architecture includes several operational modules responsible for data collection and management. These modules include course and program outcome management, enrollment and grade recording, survey administration for student or graduate self-assessment, and employer or industry feedback submission. Each module contributes structured data that are stored in the central database and later utilized for analytics and evaluation. Access to the system is controlled through authenticated sessions, and role-based restrictions ensure that users can only view or manage data associated with their assigned academic programs.

Table 1 Core System Modules for Program Outcomes Data Management

Module	Description	Data Managed
Program and Course Management	Allows administrators to define programs, courses, and academic structures	Program records, course information
Program Outcome Management	Enables creation and maintenance of program outcomes and mapping of outcomes to courses	PO definitions, PO-course mappings
Grade Entry Module	Records course grade data associated with enrolled students	Grade entries
Survey Management	Collects self-assessment responses related to program outcomes	Survey responses
Employer Feedback Module	Captures competency feedback from industry partners or employers	Employer evaluation ratings
Access Control and Authentication	Manages user authentication and role-based permissions	User roles and access rights

These integrated modules allow the system to maintain a centralized repository of program outcomes evidence. The centralized storage architecture enables the system to use the same datasets both for operational monitoring and for subsequent analytics computations.

The system integrates a dedicated analytics and reporting module that supports visualization and interpretation of program outcomes indicators. The analytics component is implemented as an interactive dashboard that retrieves processed analytics data from a secured application programming interface. Users can filter analytics results by academic program and reporting year, enabling flexible analysis of outcomes performance across different cohorts or academic periods.

Table 2 Analytics Outputs Generated by the Reporting Module

Analytics Component	Description	Purpose
PO-Level Analytics	Average scores and counts from grades, surveys, employer feedback, and board exam indicators	Evaluate attainment of individual program outcomes
Trend Analysis	Time-series indicators for grades, surveys, and employer feedback	Monitor changes in outcomes performance over time
Summary Statistics	Dataset totals such as number of courses, outcomes, grade records, and survey responses	Provide overview of available assessment data
Attainment Classification	Categorization of PO scores based on predefined thresholds	Standardize interpretation of PO attainment levels
Rule-Based Insights	Automated identification of low performance, trends, data sufficiency issues, and inconsistencies	Support decision making for program improvement

Visualization of outcomes indicators is implemented through several chart formats, including bar charts for comparing program outcomes performance, radar charts for profiling outcomes attainment across sources, pie charts for showing distribution of contributing evidence, and line or area charts for displaying temporal trends.

Figure 6 Analytics Dashboard displaying multi-source performance visualization

The analytics module transforms the centralized outcomes database into a functional decision-support tool that facilitates interpretation of program outcomes performance. Through the integration of multiple assessment sources, the system enables stakeholders to evaluate outcomes attainment using a more comprehensive evidence base rather than relying on a single data stream. The use of a unified scoring scale allows comparison among grades, surveys, and employer feedback indicators within the same analytical framework.

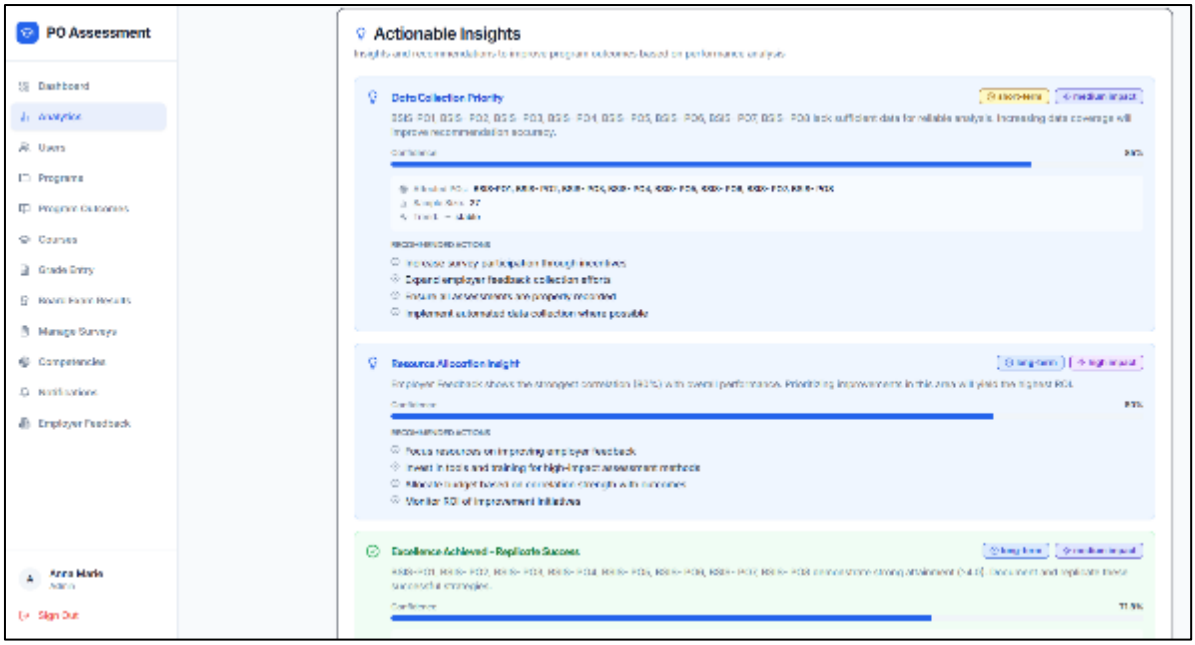

Figure 7 Analytics Dashboard displaying Actionable Insights

Another notable feature of the analytics module is the rule-based insights mechanism, which converts quantitative indicators into interpretable observations. By automatically flagging low-performing outcomes, identifying declining trends, and highlighting potential data limitations, the system assists stakeholders in prioritizing areas that require further investigation or intervention. Although the logic underlying these recommendations is deterministic rather than predictive, it offers transparent criteria that can be reviewed and adjusted according to institutional policies.

Table 3 Program Outcomes Attainment Classification

Score Range	Attainment Level
≥ 4.50	Highly Attained
4.00 – 4.49	Attained
3.00 – 3.99	Partially Attained / Needs Improvement
< 3.00	Not Attained

To assist interpretation of results, the system categorizes outcomes scores into attainment levels using predefined thresholds as shown in Table 3. The Program Outcomes Attainment Classification used in this study was author-defined for decision-support and reporting purposes, drawing on common weighted-mean interpretation practices used in educational assessment studies.

Table 4 Summary of McCall's Software Quality Evaluation

Quality Dimension	Mean Score	Interpretation
Product Operation	5.00	Very Good
Product Revision	5.00	Very Good
Product Transition	4.87	Very Good
Overall Mean	**4.96**	**Very Good**

The results of the McCall's Software Quality evaluation indicate that the developed system achieved very high quality ratings across all evaluated dimensions. The consistently high scores from expert evaluators demonstrate that the

system exhibits strong operational performance, maintainability, and adaptability, confirming that the implemented architecture and development approach effectively satisfied established software quality standards.

Table 5 Summary of CSUQ Results per Respondent Group

Respondent Group	System Usefulness	Information Quality	Interface Quality	Grand Mean	SD	Interpretation
Students	5.96	5.88	6.14	5.99	0.17	Agree
Employers	6.13	6.00	6.25	6.13	0.18	Agree
Dean	6.38	6.29	6.00	6.22	-	Strongly Agree
Program Heads	6.17	6.38	6.42	6.32	0.29	Strongly Agree
Overall Result	6.00	5.94	6.16	6.03	-	Agree

Students reported a grand mean of 5.99 (SD = 0.17), indicating consistent agreement on the system's usability and functionality. Employers also provided positive evaluations with a mean of 6.13, suggesting that the platform effectively supports user tasks and delivers reliable information. Higher ratings were observed from the Dean and Program Heads, with means of 6.22 and 6.32 respectively, reflecting strong agreement regarding system usability and interface effectiveness. Among the evaluated dimensions, interface quality obtained the highest mean score of 6.16, indicating high satisfaction with the system's layout, navigation, and interaction design. Overall, the system achieved a grand mean of 6.03, corresponding to an interpretation of agree, which demonstrates a high level of usability and user satisfaction.

4. Conclusions and Recommendations

This study developed and evaluated a Web-Based Program Outcomes Assessment and Evaluation System to support data-driven monitoring of program outcomes in outcomes based education environments. The system integrates multiple sources of assessment evidence, including course grades, survey responses, and employer feedback, into a centralized database that enables systematic data collection, storage, and analysis. Its architecture combines data management modules, a role based web interface, and an analytics dashboard that provides visualizations, trend analysis, and rule based insights to support evidence-based academic decision making and continuous program improvement.

Evaluation using McCall's Software Quality Model showed very high ratings across software quality dimensions, with an overall mean score of 4.96, indicating strong functionality, reliability, and maintainability. Usability assessment using the Computer System Usability Questionnaire also yielded positive results, with an overall mean score of 6.03 on a seven point scale. These findings indicate that the developed system meets software quality standards and provides an effective and user friendly platform for program outcomes assessment and evaluation.

Based on the findings of the study, several recommendations are proposed to further improve the Web-Based Program Outcomes Assessment and Evaluation System. Future development may expand the analytics module by incorporating advanced analytical approaches such as predictive analytics or machine learning to enable deeper analysis of program outcomes trends and provide proactive decision support for academic improvement. In addition, integration with existing institutional systems, including learning management systems, student information systems, and accreditation management platforms, is recommended to streamline data collection, reduce manual entry, and improve the consistency and efficiency of outcomes assessment processes.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

Statement of ethical approval

Ethical considerations were strictly observed throughout the conduct of this study to ensure integrity, transparency, and respect for all participants. Prior to data gathering, permission was obtained from the College of Computing and Information Sciences.

Statement of informed consent

Participation of respondents, including the dean, students and program chairs, was entirely voluntary, and informed consent was obtained to guarantee that they are fully aware of the study's purpose and procedures. All collected data, including feedback and evaluation results, was treated with the highest level of confidentiality and used solely for academic and research purposes. No personal or sensitive information was disclosed or linked to individual respondents. The development and evaluation of the system will adhere to ethical standards in research and information handling, ensuring that the process uphold fairness, data privacy, and respect for intellectual property.

References

[1] Ortega-Dela Cruz RA. Learners' attitude towards outcomes-based teaching and learning in higher education. Tuning Journal for Higher Education 2022;9:99–119. https://doi.org/10.18543/tjhe.1965.

[2] Cresencio M. Outcome-based education in open and distance learning. J Educ Elearn Res 2023;10:645–56. https://doi.org/10.20448/jeelr.v10i4.5043.

[3] Saracostti M, de Toro X, Rossi A, Lara L, Sotomayor MB. Implementation of a web-based system to measure, monitor, and promote school engagement strategies. A Chilean experience. Frontiers in Psychology 2022;13. https://doi.org/10.3389/fpsyg.2022.980902.

[4] Heil J, Ifenthaler D. Online Assessment in Higher Education: A Systematic Review. Online Learning 2023;27. https://doi.org/10.24059/olj.v27i1.3398.

[5] Bohra NS, Johri A, Wasiq M. Systematic approach of measuring program outcomes of management postgraduate program. Front Educ (Lausanne) 2024;9. https://doi.org/10.3389/feduc.2024.1404946.

[6] Montenegro-Rueda M, Luque-de la Rosa A, Sarasola Sánchez-Serrano JL, Fernández-Cerero J. Assessment in Higher Education during the COVID-19 Pandemic: A Systematic Review. Sustainability 2021;13:10509. https://doi.org/10.3390/su131910509.

[7] Lee VWY, Lam PLC, Lo JTS, Lee JLF, Li JTS. Rethinking online assessment from university students' perspective in COVID-19 pandemic. Cogent Education 2022;9. https://doi.org/10.1080/2331186x.2022.2082079.

[8] Caspari-Sadeghi S. Learning assessment in the age of big data: Learning analytics in higher education. Cogent Education 2022;10. https://doi.org/10.1080/2331186x.2022.2162697.

[9] Hernández-Campos M, Gonzalez-Torres A, García-Peñalvo FJ. Learning Outcomes Evaluation Through Learning Analytics Systems in Higher Education: A Systematic Literature Review. Sage Open 2025;15. https://doi.org/10.1177/21582440251347374.

[10] Sanusi N, Zulkifli H, Mohd I, Hamzah. Digital Technology in Classroom Assessment: A Bibliometric Study. TEM Journal 2025;14:551–61. https://doi.org/10.18421/TEM141-49.

[11] Siang SER, Reyes ARL, Dumdumaya CE. Reinforcing Learning Management System With OBE-Based Learning Analytics and RIASEC Integration. TEM Journal 2024:3346–58. https://doi.org/10.18421/tem134-69.

[12] Kyeremeh K. Overview Of System Development Life Cycle Models. Journal of Management and Science 2021;11:12–22. https://doi.org/10.26524/jms.11.3.

[13] Meshram SU. Evolution of Modern Web Services – REST API with its Architecture and Design. International Journal of Research in Engineering, Science and Management 2021;4:83–6.

[14] Le HT, Shar LK, Bianculli D, Briand LC, Nguyen CD. Automated reverse engineering of role-based access control policies of web applications. Journal of Systems and Software 2022;184:111109. https://doi.org/10.1016/j.jss.2021.111109.