Implementation of PHP Laravel Framework in the Development of Web Application for OBE-CQI Process

Camila Joy N. Bitas
Department of Electronics Engineering
College of Engineering, Technological University of the Philippines
Manila, Philippines
camilajoy.bitas@tup.edu.ph

Jan Cristian H. Dela Cruz
Department of Electronics Engineering
College of Engineering, Technological University of the Philippines
Manila, Philippines
jancristian.delacruz@tup.edu.ph

Janelle M. Eñola
Department of Electronics Engineering
College of Engineering, Technological University of the Philippines
Manila, Philippines
janelle.enola@tup.edu.ph

Mary Joy C. Laquindanum
Department of Electronics Engineering
College of Engineering, Technological University of the Philippines
Manila, Philippines
maryjoy.laquindanum@tup.edu.ph

Maria Philina F. Odango
Department of Electronics Engineering
College of Engineering, Technological University of the Philippines
Manila, Philippines
mariaphilina.odango@tup.edu.ph

Abstract—This paper describes the development of a web application that was designed to automate Outcome-Based Education (OBE) and Continuous Quality Improvement (CQI) processes at the Technological University of the Philippines (TUP) Manila. Hence, the application was developed using PHP's Laravel framework. The technology, therefore, assures automation for managing direct and indirect assessments on students, alumni tracking, and reporting efficiently and accurately. The system has friendly interfaces, strong authentication, notification, and a report with extensive data. The application's structure will be based on the Model-View-Controller architecture, in which its design strongly stresses a clear separation of data, presentation, and logic layers. This paper shows that the beauty of this application is that it has enabled all the educational management tasks in an institution to be unified in a single platform, thus enhancing the productivity and reliability of the university's administrative processes.

Keywords—OBE, CQI, direct assessments, indirect assessments, alumni tracking, PHP Laravel, Model-View-Controller

I. INTRODUCTION

Education is a powerful driver of development and one of the strongest instruments for reducing poverty and improving health, gender equality, peace, and stability. To improve education, an approach called Outcome-Based Education (OBE) - Continuous Quality Improvement (CQI) is introduced. This is where an emphasis is placed on an articulated idea of what students are expected to know and be able to do, that is, what skills and knowledge they need to have when they leave the school system. To simplify this process, a web application has been developed to make everything digitally and automatically [1].

This paper develops a web application designed for the OBE-CQI process using PHP Laravel and Python. PHP Laravel will serve as the primary framework for building the entire system. JavaScript will also be involved in retrieving redundant data from various forms. The Python API will be used to obtain the created report's Word document. Instructors can enter student grades via the web application, and the system will automatically generate the necessary results. Students are also granted access to personal accounts to keep track of their academic progress.

II. OBJECTIVES

This study aims to develop a user-friendly and intuitive interface using PHP Laravel. Having an easy-to-understand design and very straightforward buttons is crucial to easily navigate the web application without reading the manuals. Also, prioritizing a responsive layout is essential to access the web app on different devices.

Equally important is the goal of developing a resilient and efficient backend infrastructure for the web-based system. This includes designing a scalable architecture capable of handling fluctuating loads and ensuring that server-side procedures are optimized for speed and reliability. Developers can build a robust backend supporting complicated business logic and data management requirements using Laravel's built-in technologies, such as Eloquent ORM, for database interactions and its powerful routing system.

III. BACKGROUND OF THE PROBLEM

According to Wicaksono and Pakereng (2020) there is an urgent need for a unified web-based system that seamlessly integrates alumni tracking functionalities with the Outcome-Based Education (OBE) and Continuous Quality Improvement (CQI) processes to address the challenges of managing and integrating the massive amounts of data generated from these methodologies. An excellent option is to use PHP Laravel, a stable and adaptable web application framework. PHP has become the most popular scripting language among developers, leading to the growth of PHP framework [2]. Garcia-Dominguez et al., (2018) suggested that the use of a framework is often necessary for medium and large-scale projects. Instead of writing duplicate code, programmers can concentrate on the actual code needed for a project. Maintaining the framework's durability is imperative since it will eventually serve as the foundation for most advances made for the

organization. Working effectively is made possible using the Laravel framework, which stands out for its quick learning curve, accurate documentation, and community support. Moreover, it works better for big online projects that need to be completed quickly and with fewer resources [3].

The characteristics of PHP Laravel allow educational institutions to create an all-inclusive platform that not only makes OBE-CQI practices easier to apply but also easily integrates alumni tracking features. The system uses Python API Technologies and PHP to download survey and assessment reports. This journal focuses on exploring the development and implementation of such a system, with a particular emphasis on using PHP Laravel as the primary framework. This study intends to make a significant contribution to the field of educational technology and institutional management by investigating the technical nuances, design considerations, and possible advantages of using PHP Laravel in this context.

IV. REVIEW OF RELATED LITERATURE

Yamami et al. (2019) presented a novel approach to selecting web development frameworks, specifically PHP frameworks. It defines standards in six dimensions: intrinsic durability, industrialized solution, technological adaptability, strategy, technical architecture, and speed/efficiency. These criteria are then used to analyze and compare four popular PHP frameworks: Laravel, Symfony, Zend, and CodeIgniter. The evaluation considers framework maturity, community support, documentation, modular design, licensing, technical features (including ORM, templating, and AJAX support), and performance benchmarks. The research highlights each framework's merits and drawbacks, allowing developers to make an informed decision based on their project requirements [4].

Hossain (2019) examined the use of the PHP Laravel Framework for online application development, with a focus on a movie streaming service. The thesis looked at Laravel's core features, including routing, middleware, package management, and an authentication mechanism. Hossain demonstrated Laravel's adaptability for cloud environments by detailing its installation and configuration with tools like Cloud9. The study highlighted Laravel's role in optimizing development processes, decreasing redundancy, and improving code quality, as well as providing insights into its use for OBE-CQI [5]

Wicaksono and Pakereng (2020) demonstrated that the Laravel framework effectively addresses difficulties in traditional library management systems. They created a sophisticated library information system for SMK PGRI 2 Salatiga using Laravel capabilities such as MVC architecture and Eloquent ORM. This technology simplifies different library processes, improving efficiency and accuracy. Rigorous testing confirmed its dependability and use, enhancing library operations and knowledge dissemination. This emphasizes the need for current frameworks like Laravel in streamlining educational procedures, particularly library management systems, and developing OBE and CQI processes [6].

Chavan and Pawar (2021) analyzed the performance of Symfony, CodeIgniter, Phalcon, and Laravel PHP frameworks. Using identical blogging web apps across several frameworks, they examined parameters such as execution time and memory usage to provide insights. Laravel demonstrated scalability and economy with slightly faster and more memory-efficient performance than Phalcon. However, Phalcon proved to be lightweight and acceptable for larger applications. Heavy frameworks like Symfony and CodeIgniter performed well in terms of execution speed and memory usage. These insights assist developers in choosing PHP frameworks, particularly for online applications that support OBE and CQI procedures [7]

Novaliendry et al. (2021) explored the use of the PHP Laravel Framework as an E-Learning system in OBE-CQI procedures, to solve the limits of traditional face-to-face learning, notably during the COVID-19 epidemic. Their research uses structured Waterfall methodology and SDLC to create a versatile, accessible, and successful platform for web programming instruction. The Laravel framework's simplicity and flexibility are used to develop features like improved coding, schema construction, and ORM capabilities, which improve the online learning experience [8]

V. METHODOLOGY

This section presents the methods and procedures used in the development of a web application using PHP Laravel and Python API.

Requirement Analysis

The journey towards developing an efficient web application commences with an accurate requirement analysis. Engaging together with our Adviser and Co-adviser to facilitates a comprehensive understanding of the complexity involved in automating OBE and CQI processes. Through extensive discussions, key features such as forms of OBE-CQI, survey tools for student, alumni tracking, and report generation functionalities are identified and documented. Though, there is no specific requirements for the system itself, we still choose to use the PHP Laravel and Python API for its robust features, ease of use, and active community support make it an excellent choice for facilitating seamless integration with HTML, CSS, and JavaScript, aligning well with the project's requirements for efficient and effective web development.

Design Phase

In the design phase, the architectural blueprint of the web application is carefully designed with TUP logo and color as its inspiration. Emphasis is placed on scalability, modularity, and integration points for both frontend and backend components. The user interface design endeavors to strike a balance between functionality and aesthetics, with wireframes and mockups serving as visual aids to ensure consistency and usability. Meanwhile, database design revolves around structuring data efficiently using MySQL, ensuring optimal storage and retrieval mechanisms. Moreover, we also use the Figma as our tools in designing and referencing of our web application which it is a web-based design and prototyping tool used primarily for creating user interface (UI) designs, user experience (UX) designs,

and interactive prototypes for websites, mobile applications, and other digital products.

Development and Integration

Armed with a comprehensive roadmap, the development phase begins with setting up the Laravel project environment which is a pivotal step towards harnessing the power of PHP Laravel framework. The use of Model-View- Controller (MVC) architecture is embraced for this project, which can clearly characterize the separation between data, presentation, and logic layers. In web app development using a Model-View-Controller (MVC) architecture, the components are typically organized as follows:

- **Controller**: Handles user input and orchestrates the interaction between the model and the view.

- **Views**: Present data to the user in a human-readable format.

- **Model**: Represents the data and business logic of the application, interacts with the database.

The Front-end development sees the combination of HTML, CSS, and JavaScript to create dynamic and interactive user interfaces, bolstered by frameworks like Bootstrap and libraries such as Font Awesome.

On the back-end development, the PHP within the Laravel framework drives the implementation of logic functionalities and CRUD operations. Authentication and authorization mechanisms are also added to ensure secure access control. Integration with external services, including Java-based backend services for alumni tracking and Node.js for server- side scripting, enriches the web application's capabilities, elevating it beyond mere automation to a comprehensive educational management system. Additionally, with the use of CRUD operations on this web app development, it can basically manage data on a large scale. It stands for Create, Read, Update and Delete. It's a set of basic operations that are commonly used in database-driven applications to manage data:

1. Create: This operation involves adding new data records to a database. For example, in a web application, creating might involve filling out a form to add a new user to a user database.

2. Read: This operation involves retrieving data records from a database. It's about fetching data that already exists in the database. For instance, reading might involve fetching a list of all users from the user database.

3. Update: This operation involves modifying existing data records in a database. Users might update their information, such as changing their email address or password.

4. Delete: This operation involves removing data records from a database. For example, if a user decides to close their account, their record in the user database would be deleted.

Testing and Deployment

Rigorous testing forms the basis of quality assurance, encompassing unit testing, integration testing, and user acceptance testing (UAT). Unit tests scrutinize individual components for functionality and reliability, while integration testing validates seamless communication between frontend and backend modules. UAT sessions, featuring faculty and end- users, provide invaluable feedback, guiding refinements, and enhancements.
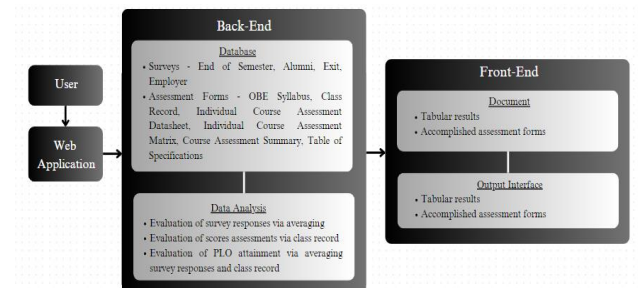

Figure 1: Block Diagram of the CQI System

VI. RESULTS

The results of the study show the relevant features of the web application. It includes student information and assessment systems, alumni tracking, reports and analytics, and automated notifications. The web application has different roles for each designated user. The topmost level in the hierarchical structure is the Admin. The admin accesses everything and maintains the whole system intact. It facilitates forms such as exit, end-of-semester surveys, alumni, and employer surveys while limiting the responses to the intended user lists and editing the availability of each form by choosing between active and inactive status.
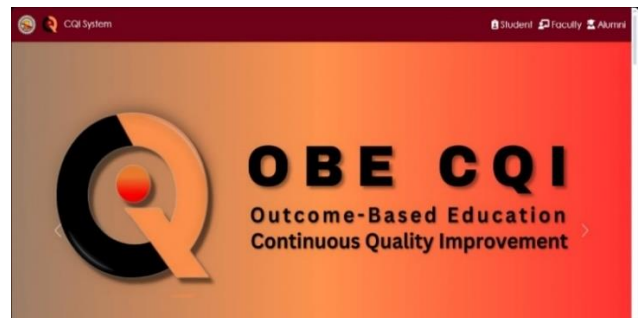
A. Homepage/Login page

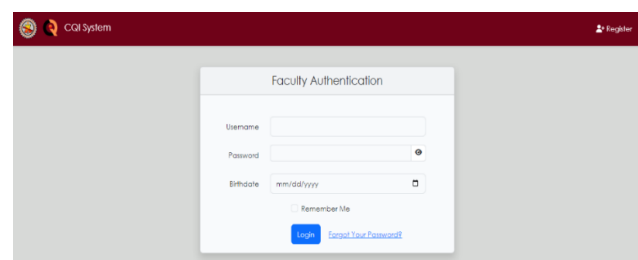**Figure 2**: Homepage of CQI System

Figure 3: Login page of CQI System

Figure 4: Sign-up page of CQI System

The home page consisted of the information needed for the accreditation, namely, TUP Mission and TUP Vision and about OBE CQI. The TUP officials and faculty per department were also included on this page. New users could register for the web application by providing their name, age, date of birth, ID number, email, username, and password. The login page allows users to log in to their accounts using information such as their username and password.

B. Survey Forms

Figure 5: End-of-Semester Survey

The figure illustrates a student's account containing the end-of-the-semester survey for different year levels. Each survey form has a designated indirect assessment questionnaire per subject, which has a progress bar that can be easily determined if it has been accomplished or not. Meanwhile, a notification system is established to inform the students about the availability of these forms every semester.

Figure 6: Exit Interview

The figure illustrates a student's account containing the exit interview for the graduating students. The survey form has a designated indirect assessment questionnaire, which collects feedback on the quality of education, teaching methods, curriculum, and overall student experience. Meanwhile, a notification system is established to inform the students about the availability of the form at the end of their studies.

Figure 7: Alumni Survey

Figure 8: Alumni Survey Sections

The figures show the form for the alumni survey which is composed of a set of questions intended for the graduates of the BSECE program in TUP-Manila. It also has a progress bar that tracks the accomplishment percentage of the user. For the automation of the notification system, the proponents have utilized cronjob, which can automatically send an email inviting alumni to answer this survey after five years (5) of program completion.

Figure 9: Employer Survey

The figure shows the form for the employer survey which is composed of a set of questions intended to gather feedback from employers about the performance and skills of graduates from the BSECE program in TUP-Manila. After the alumni answers the survey form which includes the name, position, email address, and contact number of the employer, an email will be sent automatically to the employer's email address with the link to the CQI system and universal login credentials.

Figure 10: Forms Maintenance

On the admin and faculty side, forms maintenance has been employed to the web application, which allows the stakeholders to add survey forms such as the end of the semester, exit interview, alumni, and employer's survey. As shown in the data table, the availability or status of the forms is changeable. On the left side of the table, various actions can be done to organize the questionnaires.

Figure 11: Forms Reports

The data acquired from the respondents are summarized in the form of reports, and a bar graph is utilized to represent the frequency of responses per question visually. It can also help faculty members track the number of students who have already responded to the survey forms.
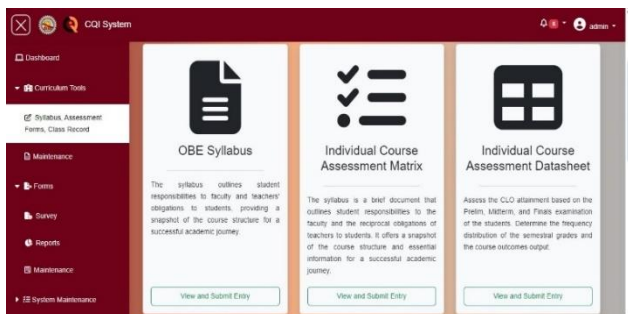
C. Curriculum Tools

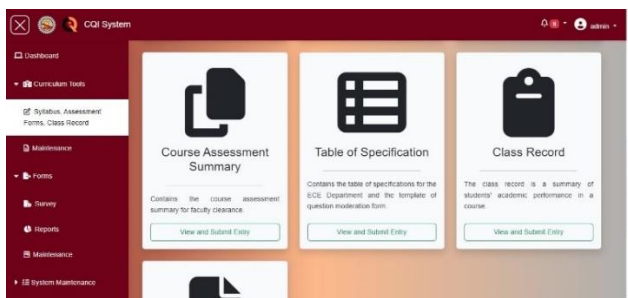**Figure 12:** Syllabus, Matrix and Datasheet

Figure 13: Summary, TOS, and Class Record

Figure 14: PLO Attainment

The figures show the OBE Syllabus, assessment forms, and class record for implementing OBE-CQI. The system integrated an auto-fetch feature wherein the information from one form will be fetched to the other forms with the same information. Faculty members can upload an Excel file to the class record to easily monitor the direct assessment for CLO attainment. These data will then be automatically sent to the PLO attainment form to monitor the attainment of the PLO.

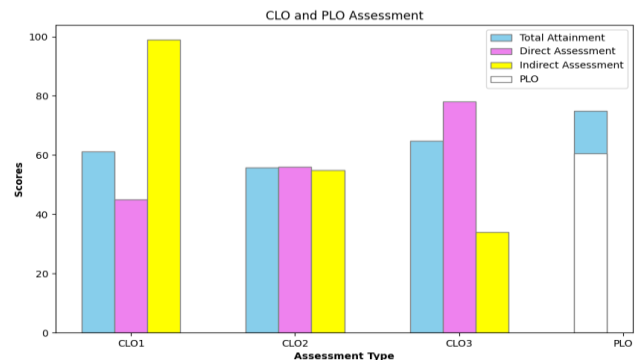### D. Reports for CLO and PLO attainment

Figure 15: Report-Generated Graph of CLO Attainment

The Word Document of the Individual Course Assessment Datasheet can display a graphical representation of the CLOs to visually represent the numerical data gathered from students' direct and indirect assessments. The datasheet assessment forms' UI can also display the bar graph representing the CLOs per subject. This graph will ease the comparison between the attainments of learning outcomes per year to achieve the CQI.
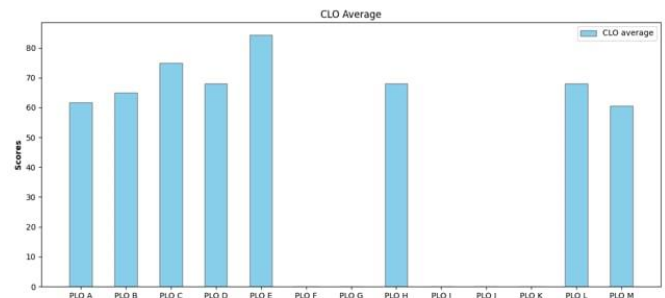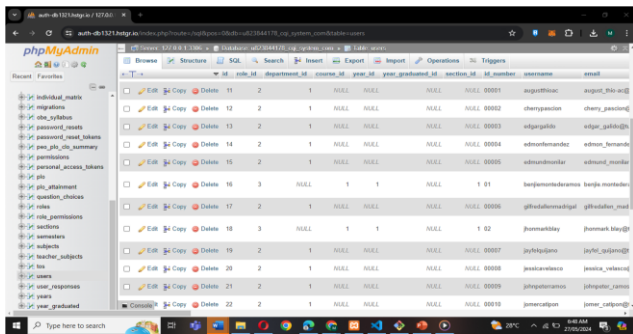

Figure 16: Report-Generated Graph of PLO Attainment

The Word Document of the PLO Attainment can display a graphical representation of the PLOs to visually represent the data gathered from CLOs. forms' UI can also display the bar graph representing the attainment per PLO. This graph will ease the comparison between the attainments of program learning outcomes per year to achieve the CQI.

E. Comprehensive Database for Students and Alumni

Figure 17: PHP Database

The figure shows the server side of the CQI Web application, which is responsible for storing and organizing user data. It consists of tables with different labels that classify the data accordingly. Each attribute in the database possesses a unique key identifier that maintains data constraints, integrity, and consistency to prevent errors. The comprehensive database is substantial to the University as it can store the historical data of students' course and learning outcomes while comparing the attainment from the previous batches to improve the curriculum throughout the years continually.

VII. CONCLUSION

The web application has efficiently handled the issues that TUP Manila faculty members confront, particularly in generating and maintaining survey and assessment forms. Previously, the process necessitated using Google Forms and managing seven distinct Word files, which was time-consuming and prone to errors and inconsistencies. All these duties have been streamlined and centralized with the new online application, resulting in considerable improvements in productivity and accuracy.

This application's creation utilizing the PHP Laravel framework allowed for the introduction of robust backend infrastructure, which is critical for system dependability and performance. Key functionalities like system maintenance have been effortlessly integrated, ensuring the application is always up to date and operating with minimal downtime. Furthermore, the backend architecture supports additional critical capabilities, laying a solid foundation for the application to handle complicated tasks effectively. Overall, this web application is a huge step forward in improving the operational efficiency and effectiveness of the TUP Manila faculty by making administrative responsibilities more manageable and less demanding.

VIII. REFERENCES

[1] Multidisciplinary Journals, "Role of Continuous Quality Improvement," Multidisciplinary Journals, May 2017. [Online]. Available: https://www.multidisciplinaryjournals.com/wp-content/uploads/2017/05/Full-Paper-ROLE-OF-CONTINUOUS-QUALITY-IMPROVEMENT.pdf

[2] E. Wicaksono, M. Pakereng, "Implementation of Laravel Framework in the Development of Library Information System (Study Case: SMK PGRI 2 Salatiga)," Jurnal Pilar Nusa Mandiri, Sep. 2020. [Online]. Available: https://doi.org/10.33480/pilar.v16i2.1666. [Accessed: Jun. 21, 2024].

[3] A. Garcia-Dominguez et al., "An Empirical Study on the Adequacy of MDE in Industry," *Journal of Software Engineering Research and Development*, 2018. [Online]. Available: https://jserd.springeropen.com/articles/10.1186/s40411-018-0050-8

[4] A. Yamami and M. Bennani, "A New Model for the Selection of Web Development Frameworks: Application to PHP Frameworks," *ResearchGate*, 2019. [Online]. Available: https://www.researchgate.net/profile/Abir-Yamami/publication/330656300_A_new_model_for_the_selection_of_web_development_frameworks_application_to_PHP_frameworks/links/5c4c8fee299bf12be3e5786c/A-new-model-for-the-selection-of-web-development-frameworks-application-to-PHP-frameworks.pdf

[5] S. Hossain, "Business Intelligence System: A New Solution for Small Businesses," *Theseus Repository*, 2019. [Online]. Available: https://www.theseus.fi/bitstream/handle/10024/171333/Hossain_Sakib.pdf

[6] V. Chavan and S. Pawar, "Performance Analysis of Symfony, CodeIgniter, Phalcon, and Laravel PHP Frameworks," 2021. [Online]. Available: https://journal.ijresm.com/index.php/ijresm/article/view/1420

[7] Novaliendry, et al., "Exploring the Use of PHP Laravel Framework as an E-Learning System in OBE-CQI Procedures," 2021. [Online]. Available: https://www.researchgate.net/publication/285550056_E-learning_of_PHP_based_on_the_solutions_of_real-life_problems