



KVPD E-Learning System

Muhammad Najmi Aiman Jantan¹, Noraini Ibrahim^{1*}

¹ Faculty of Computer Science and Information Technology

Universiti Tun Hussien Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: noraini@uthm.edu.my

DOI: <https://doi.org/10.30880/aitcs.2025.06.01.039>

Article Info

Received: 15 June 2024

Accepted: 8 May 2025

Available online: 30 June 2025

Keywords

College Vocational Port Dickson,
Learning Management System, E-
Learning

Abstract

The research addresses the need for enhancing the Kolej Vokasional Port Dickson (KVPD) E-Learning System by identifying and mitigating weaknesses inherent in the traditional KVPD learning method. Various challenges are identified, including the pace of teaching, financial burdens on students, accessibility of learning materials, and the assessment of students' skill levels. To tackle these issues, the Prototyping Model with Evolutionary Prototyping is employed. Several key functionalities of the system are identified encompassing use cases such as login, register account, manage user, manage course, manage learning material, manage assessment and assignment, record grade and generate report. A comparative analysis of system features is conducted to pinpoint areas requiring improvement. The findings contribute significantly to refinement E-Learning systems, fostering a more user-centric and efficient educational experience. The testing results demonstrate a successful implementation, with all criteria meeting or exceeding expected standards. Consequently, the system ensures heightened teacher involvement, alleviates financial strain on students, simplifies access to learning materials, and facilitates accurate assessment of students' skill levels. In conclusion, the study underscores the importance of continuous evolution in educational technology. Insights gained pave the way for future research endeavours aimed at further enhancing E-Learning systems, thereby enriching the educational landscape.

1. Introduction

Learning can be defined as the process of acquiring knowledge, skills, behaviors, or understanding through study, experience, or teaching [1]. It involves the modification of an individual's existing knowledge or behavior based on new information, experiences, or interactions with the environment. The way people are currently taught could be made more satisfying. Far more is expected from education by people now than what is often provided in textbooks and lectures, especially in the digital age when so much information is readily accessible online and on mobile devices [2]. The online learning system, sometimes known as an e-learning system, is a distance learning training system that makes use of cutting-edge Internet technology to motivate the instructor to instruct interested pupils [3]. Just logging into their system gives them enough incentive to do so, and the classes can even be taken online using smartphones and tablets if students want to learn on the road.

The Kolej Vokasional Port Dickson (KVPD) learning system faces challenges due to teachers' prioritization of delivering content quickly, which may leave some students behind. This can result in limited elaboration and teacher focus. Additionally, the rising costs of educational materials and textbooks limit access to essential resources, causing financial burdens for students. Traditional methods of distributing learning materials face challenges due to technological limitations, increasing student enrollment, and department expansion. Teachers

This is an open access article under the CC BY-NC-SA 4.0 license.



must also track and address students' Sijil Kemahiran Malaysia (SKM) levels, which can be hindered by class size, lack of standardized assessment tools, and the pressure to cover curriculum content efficiently. These factors can lead to difficulties related to the SKM level in KVPD.

Thus, the KVPD E-Learning System is proposed with a multi-faceted objective aimed at leveraging object-oriented principles to analyze and design a robust web-based platform for educational purposes. The primary goal is to facilitate seamless access to educational resources for teachers, administrators, and students within the College Vocational Port Dickson (KVPD) community. The system's scope encompasses key modules such as Login, Manage User, Assign Course and Class, Manage Course, Manage Learning Material, Manage Assessment and Assignment, Record Grade, Track SKM Level, Generate Report, and Manage Feedback. Each module caters to specific functionalities catering to the diverse needs of administrators, teachers, and students. The expected outcome of the KVPD E-Learning System is to bring about transformative improvements in the educational landscape. This includes improved access to education, providing an enhanced and personalized learning experience, offering flexibility and convenience in learning, potential cost savings, robust assessment and analytics capabilities, seamless integration with existing systems, and a feedback mechanism for continuous improvement. Success will be gauged by the system's ability to align with the institution's goals, enhance the educational experience, and adapt to the evolving needs of the College Vocational Port Dickson community. Ultimately, the proposed E-Learning System aims to establish a dynamic and efficient platform that contributes to the overall advancement of education at KVPD.

The paper will be structured as follows: A comprehensive review of related works is presented in Section 2. Section 3 details the methodology employed in the research. The outcomes of the system are delved into in Section 4. Finally, Section 5 offers a discussion on the concluding aspects of the project.

2. Related Works

An overview of the related works concerning the KVPD E-Learning System is provided in this section.

2.1 Technology Used

The technology used for this web-based system is Hypertext Preprocessor (PHP). Hypertext Preprocessor (PHP) was previously known as Personal Home Page and was released in 1995 [4]. PHP is widely used in website development because it can generate dynamic content through database connections, XML documents, graphics, PDF files, and others. Besides, PHP is able to run scripts from the command line [5]. The scripting language is JavaScript, which either runs on client-side or server-side [6]. Client-side JavaScript runs on the web browser of the client machine, but server-side JavaScript runs on the server. Client-side JavaScript is usually implemented together with HTML and CSS. However, server-side JavaScript uses PHP, Java, Python, and Ruby. Nowadays, JavaScript improves the interaction between web applications and humans [7]. For the database, Structured Query Language (SQL) is a standard programming language used to manage relational databases and do different things with the data they store. SQL was first made in the 1970s and is now commonly used by database administrators, developers writing scripts to integrate data, and data analysts setting up and running analytical queries [8]. SQL is a method of communication between humans and relational database management systems. It was developed by IBM researchers Raymond Boyce and Donald Chamberlin in the 1970s [9]. For the framework, Laravel is an open-source PHP framework, which is robust and easy to understand [10]. Laravel uses a model-view-controller design pattern and incorporates tools from various PHP frameworks, such as CodeIgniter and Yii, and Ruby on Rails. It simplifies web app creation, making it faster and more organized. Knowledge of Core PHP and Advanced PHP is beneficial. Laravel is safe and secure, making it a trendy web framework. It has more than 70,000 stars on GitHub and has become an in-demand skill on the job market [11]. Lastly, extra technology is the Badges System, which is Badges are a helpful tool for mentorship and learning help. In this situation, the mentor can plan, record, and report on the student's education using the system as a supporting tool. Students can use this system independently to organize, track, and evaluate their learning and celebrate their successes with others. Training for students upon entrance, midterm, and re-entry may incorporate this badge system [12].

2.2 Comparison with Existing System

A few systems are being compared, including UFUTURE, PutraBLAST, AUTHOR, and a new system proposed called KVPD E-Learning System. Eleven features were compared, such as the Login, the Register Account, Assign Course and Class, the Manage User, the Manage Course, the Manage Learning Material, the Manage Assessment and Assignment, the Report Grade, the Track SKM Level, the Generate Report, and the Manage Feedback. Table 1 shows a comparison of the different features that are already in place.

Table 1 Comparison between existing systems

Features	UFUTURE [13]	PutraBLAST [14]	AUTHOR [15]	KVPD E-Learning System
Login	√ (matric number and password)	√ (matric number and password)	√ (matric number and password)	√ (matric number and password)
Manage Course and Class	√	√	√	√ (manual and using excel)
Manage User	√ (edit number phone, etc.)	√ (edit picture)	X	√ (edit picture and anything else)
Manage Course	√	√	√	√ (edit overview)
Manage Learning Material	√ (add file, picture) √ (have assignment)	√ (add file, picture, video) √ (have assignment)	√ (add file, picture, video) √ (have assignment)	√ (add file, picture, video) √ (have assignment)
Assessment and Assignment	individual, group, quiz and test)	individual, group, quiz and test)	individual, group, quiz and test)	individual, group, quiz and test)
Record Grade	√ (overall grade)	√ (overall grade)	√ (only assignment and assessment)	√ (overall grade)
Track SKM Level	X	X	X	√ (track SKM approval)
Generate Report	X	X	√ (can print all submission)	√ (can generate graph of teacher)
Manage Feedback	√ (have feedback section)	√ (give feedback on teacher)	X	√ (give feedback on learning materials)

Commented [N.1]: Instead letak √ or X, pelajar tak tunjuk kan hasil

3. Methodology

The Prototyping Model is a software development model that emphasizes the creation of a prototype early in the development process. In this model, a prototype is built, evaluated, and refined iteratively until it eventually evolves into the final product. The key idea behind the Prototyping Model is to involve users in the design and development process to gather feedback and refine the system based on user input. There are four types of prototype models which are rapid throwaway, evaluation, incremental, and extreme prototyping. There are plenty of advantages when implementing prototyping model such as errors and defects can be detected in early phase of prototyping, thus can be fixed before the deployment of the finished product. Other than that, prototyping also gives users a better understanding of how the product works [16]. The Evaluation Prototyping phase is a crucial step in the Prototyping Model of software development. It involves the thorough assessment of the prototype by stakeholders to gather feedback on functionality, usability, and adherence to requirements. During this phase, users actively engage with the prototype, and evaluations include usability testing and system performance analysis. The iterative nature of the model allows for continuous refinement based on insights gained from evaluations, facilitating the creation of a final product that better meets user needs. This phase plays a key role in mitigating risks, validating the prototype, and ensuring the successful development of the desired software product.

The Prototyping Model's Planning Phase provides a structure for the whole development process. Its main goal is to create a clear knowledge of the requirements, scope, and objectives of the project. The Analysis Phase is where the system requirements are defined and developed. It involves a thorough analysis of the initial needs that were acquired in the planning stage. The Design Phase is essential for enhancing the first prototype that was made in the Analysis Phase. The main objective is to improve the user interface, functionality, and overall design based on input from stakeholders and users. The enhanced prototype is turned into a working system during the crucial Implementation Phase. The main goal is to realize the functions and design, producing a workable model that reflects the desired features and user interactions found through prototyping. [17]. The six phases of this prototyping model for this project are shown in Table 2.

Table 2 The Phases of Prototyping Model

Phase	Task	Output
Planning	<ul style="list-style-type: none"> Propose project. Task scheduling. Identify problem, scope and objective. 	<ul style="list-style-type: none"> Proposal Gantt Chart

Table 2 (cont)

Phase	Task	Output
Analysis	<ul style="list-style-type: none"> Collect and analyze the information. 	<ul style="list-style-type: none"> Swimlane diagram (To-be-model) Use case diagram Use case specification Activity diagram Sequence diagram Class diagram Requirement Definition
Design	<ul style="list-style-type: none"> Design system architecture Design user interface of the system by using PHP Design database 	<ul style="list-style-type: none"> Program code System architecture Data scheme and data dictionaries User interface
Prototype Implementation (Prototype 1)	<ul style="list-style-type: none"> Programming Develop prototyping 1 and connection with database 	<ul style="list-style-type: none"> Prototyping 1
Prototype Implementation (Prototype 2)	<ul style="list-style-type: none"> Programming Develop prototyping 2 and connection with database 	<ul style="list-style-type: none"> Prototyping 2
System Implementation	<ul style="list-style-type: none"> Implement all functions and features. Test the system 	<ul style="list-style-type: none"> System Test case Requirement Matrix
		Traceability

3.1 Analysis

This section presents the results of the system analysis, depicted through various visual representations. These include a swimlane diagram, a use case diagram, the corresponding sequence diagram, an activity diagram, a class diagram, and a comprehensive requirement definition. These visual aids collectively convey the findings and insights derived from the analysis of the system.

3.1.1 Swimlane diagram

The swimlane diagram describes the overall process of the system for the three (3) users of the system. The swimlane diagram of proposed is displayed in Fig. 1.

3.1.2 Use case diagram

A use case diagram is a graphical representation in the field of software engineering that depicts the interactions between various actors (users or external systems) and a system, emphasizing the different ways in which these actors interact with the system to achieve specific goals or perform tasks. The use case diagram represents the overall activity of the KVPD E-Learning System. The purpose of this diagram is to illustrate the dynamic parts of the system. The use case diagram that has been proposed is displayed in Fig. 2.

The KVPD E-Learning System is an online education platform designed for students, teachers, and administrators. It offers various features to make learning and academic management easier. In the login module, users like teachers, administrators, and students can log in using their student or teacher IDs and passwords. There's also a password recovery option. The register account module allows students to create new accounts by providing details like name, email, student ID, and password, with verification of active student IDs. The manage user module lets users edit profiles and upload profile pictures. In the assign course and class module, administrators can assign up to 10 courses to students and link specific teachers to courses and classes. Students are also assigned to classes, mainly focusing on Form 5 classes (3 classes), and teachers are assigned to these classes. The system includes a module for managing courses, enabling students to browse and search for semester courses and teachers to edit and update their courses. Learning materials like lecture notes and videos can be accessed through the manage learning material module, which also allows teachers to organize these resources. The system facilitates assessment and assignment management through the manage assessment and assignment

module. Students can access online assignments and quizzes, and teachers can manage them, set due dates, and track student progress. Grades are recorded in the record grade module, where students can view their grades, and teachers can manage grading for assignments, quizzes, and courses. The track SKM level module lets teachers approve student submissions for each SKM level, while students can view their SKM level progress. The generate report module allows teachers and students to generate various reports, including submission reports, achievement badge lists, and grading marks reports. Finally, the manage feedback module enables students to provide feedback on learning resources, fostering communication and improvement within the KVPD E-Learning System. The detailed for each use case proposed by corresponding sequence diagram at **Appendix A**.

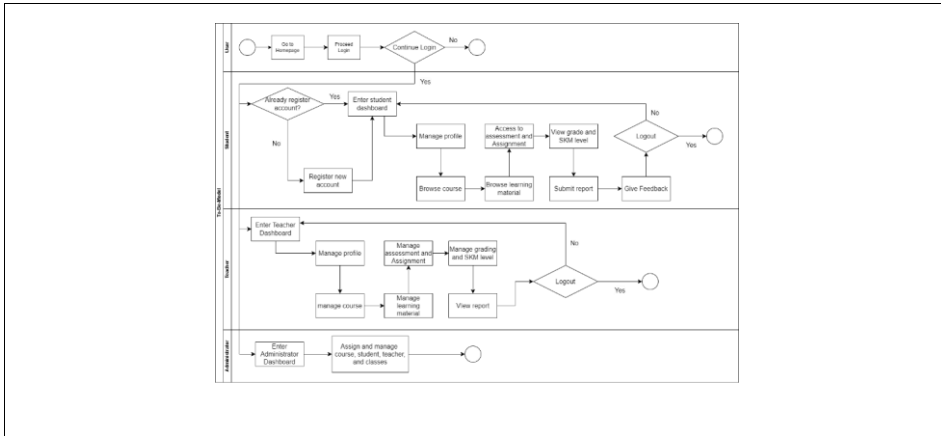


Fig. 1 TO-BE Swimlane Diagram

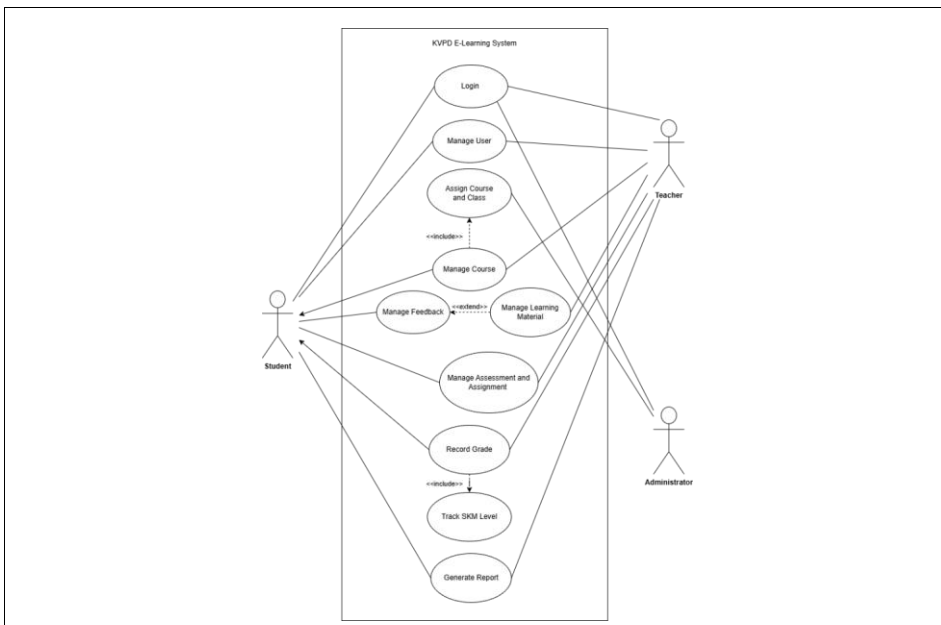


Fig. 2 Use case Diagram

3.1.3 Class diagram

A class diagram is a visual representation in the Unified Modelling Language (UML) that illustrates the structure and relationships within a system by depicting classes, their attributes, methods, and the associations between classes. Generally, there are 10 classes in this system which is displayed in Fig. 3.

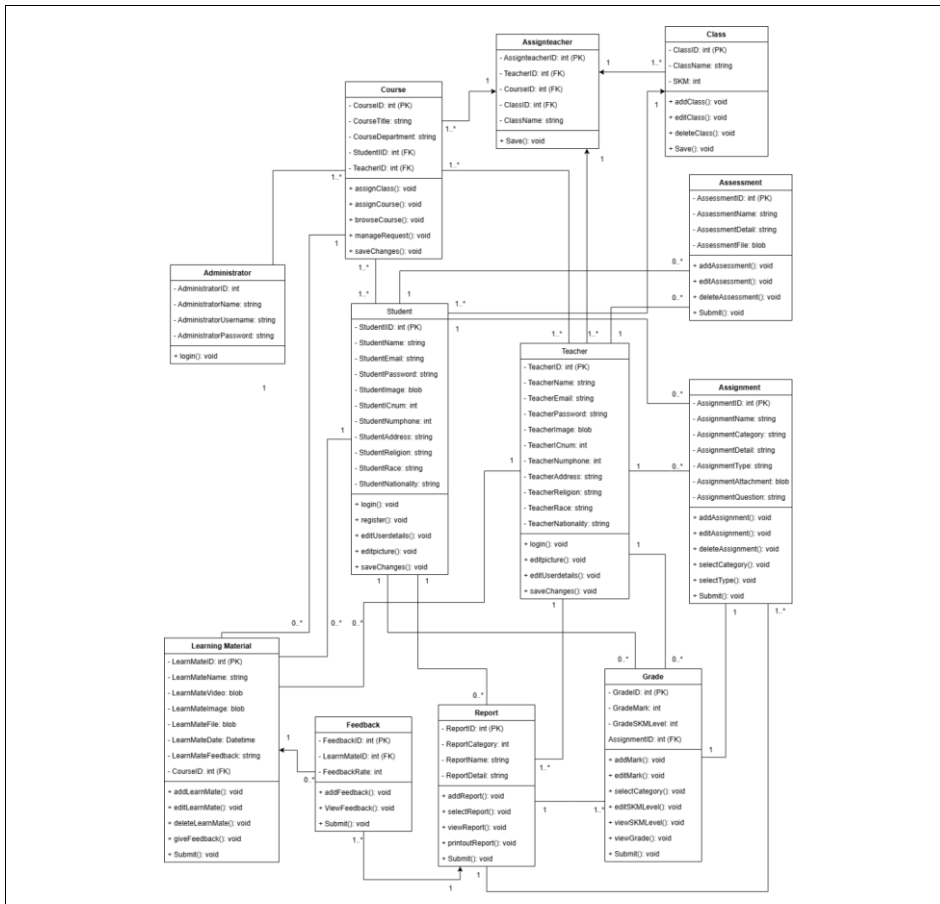


Fig. 3 Class Diagram

3.1.4 Requirement Definition

The requirements the developed system must meet or what the users expect from the suggested system are found through requirement analysis. Also, these requirements are essential for figuring out the general quality and usefulness of the system because they affect both. The following parts will discuss the requirements analysis, including user requirements, functional and non-functional requirements, and more. The many valuable features of the KVPD E-Learning System are shown in Table 3.

Table 3 System's functional modules

No.	Module	Function	User
1	Login	Login to the system.	Student, Teacher and Administrator
2	Assign Course and Class	Register course, Assign teacher and student to each class and course.	Administrator
3	Manage User	Edit and update user profile.	Student and Teacher
4	Manage Course	Browse, search, and manage course.	Student and Teacher

Table 3 (cont)

No.	Module	Function	User
5	Manage Learning Material	Browse, search, and manage learning material.	Student and Teacher
6	Manage Assessment and Assignment	Browse, search, and manage assessment and assignment.	Student and Teacher
7	Record Grade	View and manage grade and show achievement badge based SKM level.	Student and Teacher
8	Track SKM Level	View SKM level progress and approves for each submission for SKM level.	Student and Teacher
9	Generate Report	Submission report and show the list of students by achievement badge, grading marks report and show report feedback.	Teacher
10	Manage Feedback	Make feedback on learning materials	Student

Functional requirements of the KVPD E-Learning System are shown in Table 4.

Table 4 Functional requirements

No.	Module	Functional Requirements
1	Login	<ul style="list-style-type: none"> The system shall allow users enter their active ID as their username and password. The system shall allow student to reset or recovery password.
2	Assign Course and Class	<ul style="list-style-type: none"> The system shall allow administrator to register courses for each student. The system shall allow administrator to assign teacher for each course. The system shall allow administrator to assign student for each class. The system shall allow administrator to assign teacher for classes.
3	Manage User	<ul style="list-style-type: none"> The system shall allow users to edit and update their user profile. The system shall allow users to upload profile picture.
4	Manage Course	<ul style="list-style-type: none"> The system shall allow students to browse and search for their courses and view a list of enrolled courses. The system shall allow teacher to manage the course and that be enroll for their students.
5	Manage Learning Material	<ul style="list-style-type: none"> The system shall allow student to access lectures notes, or other education resources like teaching video. The system shall allow teacher to manage the learning materials.
6	Manage Assessment and Assignment Module	<ul style="list-style-type: none"> The system shall allow student students to access the online assignment, quiz, test and submit individually or in a group. The system shall allow Teacher be able to manage the online assignment, quiz, and the due date of the assignment. The system shall allow teacher to track student assignment progress. System shall classify and filter the assessment based on SKM level.
7	Record Grade	<ul style="list-style-type: none"> The system shall allow students to view their grade of their assignment and course. The system shall allow teacher to manage the grading of their course, assignment, quiz, and test. System should show student list and give the achievement badge based on achieve of SKM level.
8	Track SKM Level	<ul style="list-style-type: none"> The system shall allow teacher approves for each submission for each SKM Level The system shall allow student to view SKM level progress.
9	Generate Report	<ul style="list-style-type: none"> The system shall allow student to submit their report. The system shall allow teacher print out the list of the students by achievement badge. The system shall allow teacher grading marks report.
10	Manage Feedback	<ul style="list-style-type: none"> The system shall allow student to give feedback of learning materials.

Non-functional requirements of the KVPD E-Learning System are shown in Table 5.

Table 5 *Non-functional requirements*

No.	Requirement	Non-Functional Requirements
1	Performance	<ul style="list-style-type: none"> The system shall respond to user interactions within 5 seconds under normal load conditions.
2	Security	<ul style="list-style-type: none"> The password is encrypted in the database
3	Availability	<ul style="list-style-type: none"> The system should be accessible during standard operating hours, excluding scheduled maintenance periods.
4	Usability	<ul style="list-style-type: none"> The user interface must be designed to be intuitive, requiring no more than 30 minutes of training for a new user to navigate and use the key features of the system effectively.
5	Reliability	<ul style="list-style-type: none"> The system should provide a reliable environment to teacher, student, and administrators. The system shall handle the entry of learning materials by teachers

User Requirement Analysis of the KVPD E-Learning System is shown in Table 6.

Table 6 *User Requirement Analysis*

No.	User Requirement
1	Users should be able to log in using their active ID as their username and a password.
2	Students should be able to reset and recover their password.
3	Administrator should be able to register courses for each student, teacher to classes, students to classes, teachers to courses.
4	Users should be able to edit and update their user profile.
5	Students should be able to browse and search their course.
6	Teachers should be able to manage the courses they are assigned to.
7	Students should be able to access the learning materials.
8	Teachers should be able to manage learning materials.
9	Students should be able to access online assignments, quizzes, tests, and submit them individually or as a group.
10	Teachers should be able to manage online assignments, quizzes, and set due dates, track student assignment progress.
11	Students should be able to view their grades for assignments and courses.
12	Teachers should be able to manage the grading of courses, assignments, quizzes, and tests.
13	Teachers should be able to approve each submission for each SKM level.
14	Students should be able to view their SKM level progress.
15	Students should be able to submit their reports.
16	Teachers should be able to print out the list of students with achievement badges.
17	Teachers should have access to a grading marks report.
18	Students should be able to give feedback on learning materials.

3.2 Design

This section presents the results of the system design and database is represented in two (2) outcomes. They are system architecture and schema table for the database.

3.2.1 System Architecture

In this project, Model-View-Controller (MVC) architecture is used. The system can be separated into 3 parts, the model that is responsible for the data management, views related to user interface (UI), and the controller that manage the event handling in the system [12]. Fig. 4 shows the package diagram of the system that shows the model, view, and controller of the system.

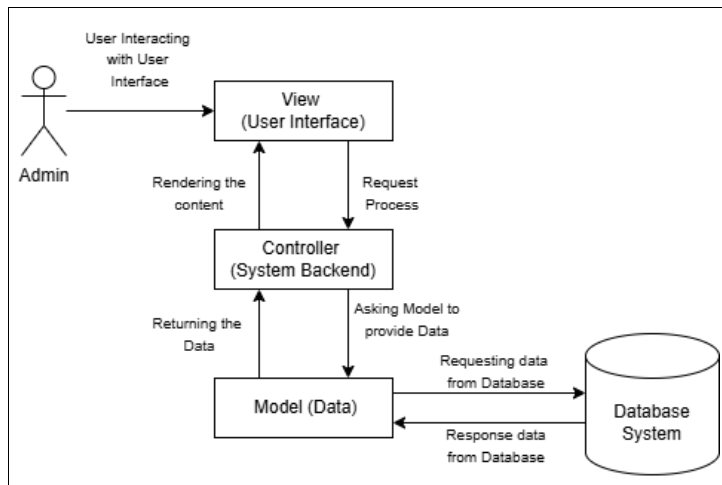


Fig. 4 System's Architecture

The Model-View-Controller (MVC) architectural pattern is a design structure used to separate an application into three interconnected components. The Model represents the data and business logic of the application. It manages data, responds to queries from the View, and updates based on commands from the Controller. The View is the user interface of the application, displaying data to the user and sending user commands to the Controller. The Controller acts as an intermediary between the Model and the View. It receives user input from the View, processes it, and updates the Model accordingly, ensuring the View displays the updated data. This separation of concerns helps in organizing code, making it more modular and easier to maintain.

3.2.2 Database schema

A database schema is a logical blueprint or structural design that represents the organization of data in a database.

- i. student (StudentID (int), StudentName (nvarchar), StudentEmail (nvarchar), StudentPassword (nvarchar), StudentImage (nvarchar), StudentICNum (nvarchar), StudentNumphone (int), StudentAddress (nvarchar), StudentReligion (nvarchar), StudentRace (nvarchar), StudentNationality (nvarchar))
- ii. teacher (TeacherID (int), TeacherName (nvarchar), TeacherEmail (nvarchar), TeacherPassword (nvarchar), TeacherImage (nvarchar), TeacherICNum (nvarchar), TeacherNumphone (int), TeacherAddress (nvarchar), TeacherReligion (nvarchar), TeacherRace (nvarchar), TeacherNationality (nvarchar))
- iii. class (ClassID (int), ClassName (nvarchar), SKM (int))
- iv. assignteacher (AssignteacherID (int), TeacherID (int), CourseID (int), ClassID (int), CourseOverview (nvarchar))
- v. course (CourseID (int), CourseTitle (nvarchar), CourseDepartment (nvarchar), StudentID (int), TeacherID (int))
- vi. administrator (AdministratorID (int), AdministratorName (nvarchar), AdministratorUsername (nvarchar), AdministratorPassword (nvarchar))
- vii. assessment (AssessmentID (int), AssessmentName (nvarchar), AssessmentDetails (nvarchar), AssessmentFile (nvarchar))
- viii. Assignment (AssignmentID (int), AssignmentName (nvarchar), AssignmentCategory (nvarchar), AssignmentDetails (nvarchar), AssignmentType (nvarchar), AssignmentAttachment (nvarchar), AssignmentQuestion (nvarchar))
- ix. grade (GradeID (int), GradeMark (int), GradeSKMLevel (int), AssingmentID (int))



Fig. 7 Interface Manage User

Fig. 8 Code Segment Manage User

4.1.3 Assign Course and Class

Administrators have two methods for assigning teachers: manually or in bulk using an Excel file. For manual assignments, admins select each teacher and click "Assign," specifying the course and class. The bulk assignment method allows admins to upload an Excel file with multiple teacher assignments, speeding up the process. For assigning students, admins individually edit each student's profile, selecting the course department for each. Unlike teachers, there is no bulk assignment option for students, requiring detailed and customized assignments for each as shown in Fig. 9. The code snippet provided manages both bulk and manual teacher assignments. For bulk assignments, it processes the uploaded file, ensuring it is in a valid format ('cls', 'csv', 'xlsx'). It extracts data from each row, retrieves the necessary IDs from the database, checks for existing assignments, and inserts new records if no duplicates are found, providing feedback on success or errors. For manual assignments, the code retrieves form data, fetches IDs for the teacher, course, and class, verifies their existence, checks for existing assignments, and inserts new records if needed. It provides feedback on the success or failure of the assignment, including any encountered errors. This ensures a robust and user-friendly process for managing teacher assignments as shown in Fig. 10.



Fig. 9 Interface Assign Course and Class

Fig. 10 Code Segment Assign Course and Class

4.1.4 Manage Course

In the "Manage Course" section, users can click on or search for assigned courses. The system displays Course Title, Class Name, and Course Overview, along with a list of enrolled students. Teachers can edit the Course Overview to keep it relevant. In the student view, students see detailed course information, including Course Title, Class Name, and Course Overview, plus a profile of the assigned teacher as shown in Fig. 11. The code snippet updates the Course Overview on form submission by sanitizing input to prevent SQL injection, then executing an SQL update query. If successful, course details are refreshed; otherwise, an error message is logged or displayed, ensuring course information remains current and accurate as shown in Fig. 12.



Fig. 11 Interface Manage Course

Fig. 12 Code Segment Manage Course

4.1.5 Manage Learning Material

Upon selecting a course, teachers can manage course materials in the Folder section. They can add new folders by specifying a name and status (visible or hidden) as shown in Fig. 13. In Student View, only visible folders are shown. Teachers can add files (PDFs, Word documents, Excel sheets, videos, images) to folders by selecting the file and clicking "Add File." This keeps materials organized and accessible to students. The provided code handles adding new folders and files. For folders, it retrieves the name, status, and current date, then inserts this data into the database. For files, it retrieves file details, uploads the file to a designated folder, and inserts the file information into the database. Successful operations refresh the page to reflect changes, while errors are handled and displayed for debugging as shown in Fig. 14.

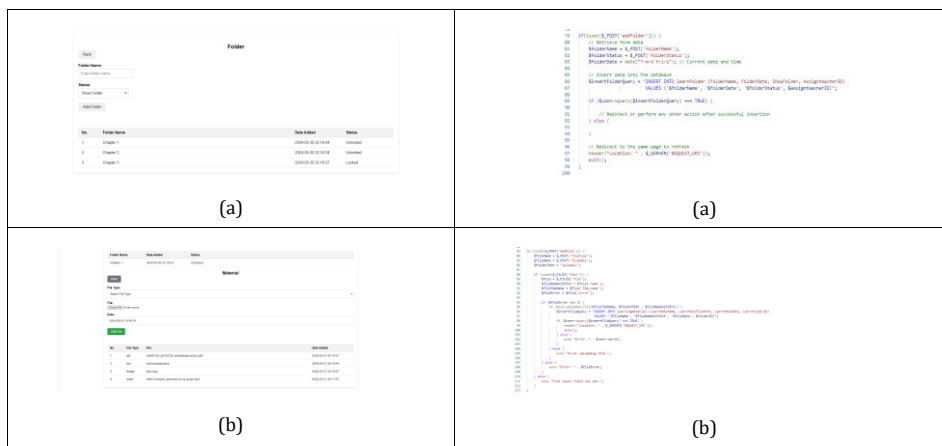


Fig. 13 Interface Manage Learning Material

Fig. 14 Code Segment Manage Learning Material

4.1.6 Manage Assessment and Assignment

Upon selecting an assignment, students are guided to the Submission Place, a centralized hub tailored for managing assignments efficiently. Here, they find crucial information such as the assignment's Due Date, fostering organization and awareness. The user-friendly interface facilitates seamless file uploads through a simple "Choose File" button, allowing students to add or remove files as needed before finalizing their submission. For teachers, accessing student submissions is effortless with the "View Submission" feature. Clicking on this option leads to a dedicated page where educators can effectively manage submissions. This page provides essential details like the Submission Due Date, enabling teachers to monitor assignment deadlines with precision. Moreover, it presents a comprehensive table listing all students in the class, complete with vital information such as Student Name, Matriculation Number, File Name, Date of Submission, and Submission Status. This comprehensive overview ensures efficient tracking of student progress and submission statuses, distinguishing between on-time and late submissions as shown in Fig. 15.



Fig. 15 Interface Manage Assignment

In the Manage Assessment interface, teachers can select either a Quiz or a Test, similar to the assignment management interface. They can add a new assessment folder by clicking "Add Assessment" and input details like Assessment Name, Status (choosing between "Show Assessment" or "Hide Assessment" for visibility control), Start Date, End Date (to define availability), Type of Assessment ("Question" or "Attachment"), and Marks (maximum score). Once details are entered, teachers save by clicking "Add Assessment". In the student's view, only assessments marked "Show Assessment" are visible. Colors denote assessment status: white for ongoing, red for upcoming, and grey for concluded assessments, helping students track availability and status. Teacher can edit assessment details by clicking "Edit," adjusting aspects like Assessment Name, status, start/end dates, and marks. After changes, they save by clicking "Save". Additionally, they can manage supplementary files by clicking "Add File," selecting desired files, and uploading. For students, selecting an ongoing quiz reveals a submission area with an End Date countdown, aiding time management. They can download the assessment file, upload completed work, and delete previous submissions, ensuring efficient submission management. This user-friendly interface prioritizes seamless assessment completion, minimizing technical hurdles for students as shown in Fig. 16.



Fig. 16 Interface Manage Assessment

4.1.7 Record Grade

The Record Grade Module stands as a beacon of efficiency for educators. This multifaceted tool equips teachers with the means to meticulously manage student grades across a spectrum of assessment types, including assignments, quizzes, and tests. Its intuitive interface expedites the grading process, allowing teachers to update grades seamlessly while maintaining pinpoint accuracy. Meanwhile, students benefit from a transparent view of their academic progress via Student View, providing them with valuable insights into their performance across various assessments as shown in Fig. 17. PHP script operates diligently to ensure the integrity and reliability of grade submissions. This script serves as the backbone of the grading system, meticulously validating and processing each grade entry to uphold the highest standards of accuracy. Thus, within the bustling ecosystem of education technology, the Record Grade Module emerges as a vital tool, fostering a harmonious exchange of knowledge and feedback between educators and students as shown in Fig. 18.



Fig. 17 Interface Record Grade

Fig. 18 Code Segment Record Grade

4.1.8 Track SKM Level

In the Track SKM Level module, teachers gain access to a comprehensive list of enrolled students for each selected course. The table showcases vital student details such as matriculation number, name, total marks, grade, and SKM approval status, providing a holistic overview of student performance and SKM submission status as shown in Fig. 19. Students, on their end, can view a table listing their enrolled courses for the respective SKM Level. Each row displays course title, total marks achieved, grade obtained, and the status of their SKM submission. The provided code manages SKM submission approval for students. It updates approval status for individual students by handling POST requests containing studentID and approveStatus, and for all students associated with a teacher via POST requests with

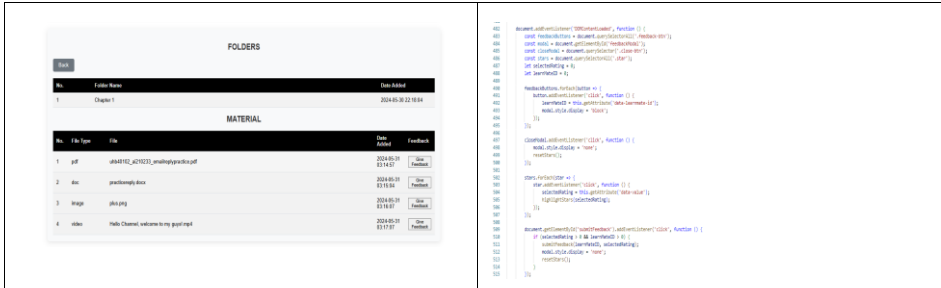


Fig. 23 Interface Manage Feedback

Fig. 24 Code Segment Manage Feedback

4.2 Testing

The test case for ten (10) modules are the Login, Manage User, Assign Course and Class, Manage Course, Manage Learning Material, Manage Feedback, Manage Assessment and Assignment, Record Grade, Track SKM Level, Generate Report. All the features to be tested are listed as shown in Table 7.

Table 7 List of Test Cases

No.	Test Cases	Description	Status
TC_100			
Login			
1.	TC_100_01	A registered student login with a valid Matric Number and Identity Card Number and is redirected to the student homepage.	Pass
2.	TC_100_02	A registered teacher login with a valid username and Identity Card Number and is redirected to the teacher homepage.	Pass
3.	TC_100_03	An administrator logs in with a valid username and password and is redirected to the admin dashboard.	Pass
4.	TC_100_04	A user enters an incorrect username or password, and the system displays an error message: "Invalid username and password."	Pass
TC_200			
Manage User			
5.	TC_200_01	A teacher edits personal information (Name, Email, IC Number, Phone Number) and clicks save. The system displays "Update successfully!".	Pass
6.	TC_200_02	A student edits personal information (Name, Email, IC Number, Phone Number) and clicks save. The system displays "Update successfully!".	Pass
7.	TC_200_03	A teacher updates their password by entering the current password, new password, and confirming the new password. The system displays "Password changed successfully!".	Pass
8.	TC_200_04	A student updates their password by entering the current password, new password, and confirming the new password. The system displays "Password changed successfully!".	Pass
9.	TC_200_05	A user uploads a new profile picture, and it is successfully displayed on their profile.	Pass
10.	TC_200_06	A user deletes their profile picture, and the default picture is displayed.	Pass
TC_300			
Assign Course and Class			
11.	TC_300_01	An admin manually assigns a teacher to a course and class. The teacher is listed as assigned in the system.	Pass
12.	TC_300_02	An admin assigns teachers using an Excel file. All teachers are listed as assigned according to the file.	Pass
13.	TC_300_03	An admin assigns a student to a course department. The student is listed as assigned in the system.	Pass
TC_400			
Manage Course			
14.	TC_400_01	A teacher views assigned courses, edits the Course Overview, and saves changes. The updated overview is displayed.	Pass
15.	TC_400_02	A student views their enrolled courses, and the Course Overview is displayed accurately.	Pass
16.	TC_400_03	A student views the profile of the assigned teacher, and all relevant information is displayed.	Pass

Table 7 (cont)

No.	Test Cases	Description	Status
TC_500			
Manage Learning Material			
17.	TC_500_01	A teacher adds a new folder, sets it to "Show Folder", and the folder is displayed in the student view.	Pass
18.	TC_500_02	A teacher uploads a file to a folder, and the file is accessible to students.	Pass
19.	TC_500_03	A student accesses learning materials from a displayed folder.	Pass
TC_600			
Manage Feedback			
20.	TC_600_01	A student gives feedback on learning material using the star rating system. The feedback is submitted successfully.	Pass
21.	TC_600_02	A teacher views the feedback received for learning material. The feedback is displayed correctly.	Pass
TC_700			
Manage Assessment and Assignment			
22.	TC_700_01	A teacher adds a new assignment, sets it to "Show", and sets a due date. The assignment is displayed in the student view.	Pass
23.	TC_700_02	A student uploads an assignment before the due date. The submission is recorded successfully.	Pass
24.	TC_700_03	A teacher edits an existing assignment and saves changes. The updated assignment details are displayed.	Pass
25.	TC_700_04	A teacher creates a new assessment (Quiz or Test), sets it to "Show Assessment", and sets start and end dates. The assessment is displayed in the student view.	Pass
26.	TC_700_05	A student submits an assessment within the specified dates. The submission is recorded successfully.	Pass
27.	TC_700_06	A teacher views student submissions for an assessment. All submission details are displayed.	Pass
TC_800			
Record Grade			
28.	TC_800_01	A teacher records grades for an assignment. The grades are saved and displayed in the student's record.	Pass
29.	TC_800_02	A teacher views and edits the overall grade for a student. The changes are saved and displayed.	Pass
30.	TC_800_03	A student views their assignment and assessment marks. The marks are displayed correctly.	Pass
TC_900			
Track SKM Level			
31.	TC_900_01	A teacher views the SKM approval status for students. The status is displayed accurately.	Pass
32.	TC_900_02	A student views their total marks, grade, and SKM approval status for enrolled courses. The information is displayed correctly.	Pass
TC_1000			
Generate Report			
33.	TC_1000_01	A teacher generates a bar chart showing the total number of students in each course. The chart is displayed accurately.	Pass
34.	TC_1000_02	A teacher generates a horizontal bar chart showing student feedback on learning materials. The chart is displayed accurately.	Pass
35.	TC_1000_03	A teacher generates a stacked bar chart showing assignment submission statuses. The chart is displayed accurately.	Pass
36.	TC_1000_04	A teacher generates a pie chart showing the distribution of student grades in a selected course. The chart is displayed accurately.	Pass

A total of 36 test cases had been tested for the KVPD E-Learning System. The system has been passed successfully. The overall result of the test cases as shown in Table 8.

Table 8 Overall Result of the Test Cases

Test Case ID	Total Test Cases	Total Pass	Total Fail
TC_100	4	4	0
TC_200	6	6	0
TC_300	3	3	0
TC_400	3	3	0
TC_500	3	3	0
TC_600	2	2	0
TC_700	6	6	0

Commented [NB12]: Betulkan table format

Test Case ID	Total Test Cases	Total Pass	Total Fail
TC_800	3	3	0
TC_900	2	2	0
TC_1000	4	4	0
Total	36	36	0

The pie chart summarized the pass rate of the overall result as shown in Fig. 25. This pie chart shows the passing rate is 100%.

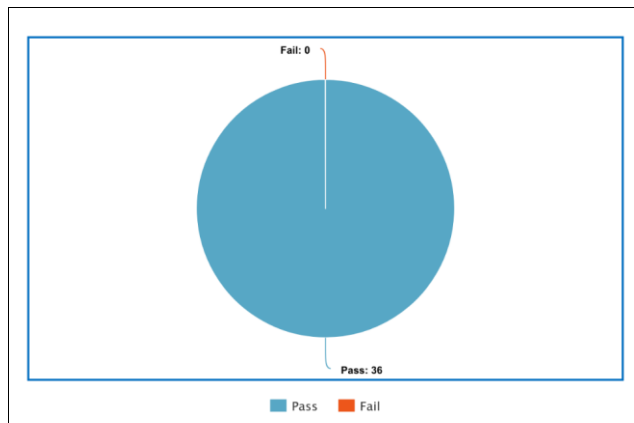


Fig. 25 Pie Chart of Overall of Test Cases

Conclusion

In conclusion, this research focuses on enhancing the KVPD E-Learning System. The Prototyping Model methodology is employed for systematic development, spanning six (6) phases dedicated to specific use cases. The system's technology stack includes Hypertext Preprocessor (PHP), JavaScript, Structured Query Language (SQL), and the Laravel framework. Additional technology, such as the Badges System, enhances the learning and mentorship experience. The detailed system requirements, both functional and non-functional, are outlined, covering modules such as Registration, Login, Course Management, Learning Material Management, Assessment, Grading, SKM Level Tracking, Reporting, and Feedback. The user requirements analysis emphasizes the needs of students, teachers, and administrators, ensuring the system caters to their specific functionalities and interactions. The study concludes by highlighting the importance of continual evolution in educational technology and suggests insights for future research in this dynamic field.

Acknowledgement

The author would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: **study conception and design:** Muhammad Najmi Aiman Jantan, Noraini Ibrahim; **data collection:** Muhammad Najmi Aiman Jantan; **analysis and interpretation of results:** Muhammad Najmi Aiman Jantan, Noraini Ibrahim; **draft manuscript preparation:** Muhammad Najmi Aiman Jantan, Noraini Ibrahim. All authors reviewed the results and approved the final version of the manuscript.

References

- [1] Inchainge, B. V., "Learning: What is it?," Inchainge, Jan. 19, 2023. [Online]. Available: <https://inchainge.com/knowledge/learning/>.
- [2] A. Haleem and K. Abid, "Understanding the role of digital technologies in education: A Review," Sustainable Operations and Computers, May 23, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2666412722000137>.
- [3] S. Tamm, "What is the definition of e-learning?," E-student, Sep. 21, 2023. [Online]. Available: <https://e-student.org/what-is-e-learning/>.
- [4] J. Samra, "Comparing performance of plain PHP and four of its popular frameworks," 2015.
- [5] K. Tatroe and P. MacIntyre, Programming PHP: Creating Dynamic Web Pages, O'Reilly Media, 2020.
- [6] P. A. Gardner, S. Maffei, and G. D. Smith, "Towards a program logic for JavaScript," in Proceedings of the 39th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, pp. 31-44, Jan. 2012.
- [7] A. M. Fard and A. Mesbah, "Jsnoise: Detecting javascript," Sep. 2013.
- [8] P. Loshin and J. Sirkin, "What is structured query language (SQL)?," Data Management, Feb. 7, 2022. [Online]. Available: <https://www.techtarget.com/searchdatamanagement/definition/SQL>.
- [9] D. Florescu and G. Fourny, "JSONiq: The history of a query language," IEEE Internet Computing, vol. 17, no. 5, pp. 86-90, 2013.
- [10] "T. Point, "Laravel - Overview," Online Tutorials, Courses, and eBooks Library, 2006. [Online]. Available: https://www.tutorialspoint.com/laravel/laravel_overview.htm.
- [11] A. Jalli, "What is Laravel?," Built In, 2022. [Online]. Available: <https://builtin.com/software-engineering-perspectives/laravel>.
- [12] European Badge Alliance, "Long-term school exchanges," 2017. [Online]. Available: <https://ebawebsite.net/badge-systems/school-exchange-badges/>.
- [13] UFUTURE, "Ufuture," Dec. 28, 2023. [Online]. Available: <https://ufuture.uitm.edu.my/home/>.
- [14] PutraBLAST, "PutraBLAST On Campus semester 1, 2023/2024," Dec. 28, 2023. [Online]. Available: <http://putrablastdk.upm.edu.my/>.
- [15] "Author UTHM academic online resources global learning system uthm," Dec. 28, 2023. [Online]. Available: <https://author.uthm.edu.my/>.
- [16] S. Lewis, "What is The prototyping model?," CIO, Jun. 9, 2023. [Online]. Available: <https://www.techtarget.com/searchcio/definition/Prototyping-Model#:~:text=The%20prototyping%20model%20is%20a.or%20product%20can%20be%20developed>.
- [17] W. R. Bischofberger and G. Pomberger, Prototyping-Oriented Software Development: Concepts and Tools, Springer Berlin Heidelberg, Germany, 2012.