UKM FILE MANAGEMENT SYSTEM

CUI KAI, KHAIRUL AZMI ABU BAKAR

Faculty of Information Science & Technology
Universiti Kebangsaan Malaysia
43600 Bangi, Selangor

Abstract

Kajian ini memfokuskan pada masalah yang wujud dalam pengurusan dokumen pengajaran Universiti Kebangsaan Malaysia (UKM), seperti penyebaran berbilang platform, kebenaran yang tidak teratur dan kemajuan yang tidak terkawal. Satu sistem pengurusan dokumen pengajaran berasaskan Web telah direka bentuk dan dibangunkan. Sistem ini menggunakan MySQL sebagai pangkalan data belakang, dan menggabungkan kawalan akses berdasarkan peranan, penjejakan status fail dan pengurusan versi untuk merealisasikan pengurusan terkumpul, berstruktur dan cekap bahan pengajaran. Melalui analisis perbandingan dengan Google Drive, Amazon Drive dan OneDrive, sistem ini menunjukkan prestasi yang cemerlang dari segi keselamatan, skalabiliti dan kelebihan penyebaran tempatan. Hasil akhir telah disahkan melalui pelbagai pusingan ujian, menunjukkan bahawa sistem ini mempunyai antara muka yang mesra pengguna, fungsi yang lengkap dan operasi yang stabil. Ia boleh meningkatkan kecekapan pengurusan dan tahap penstandaran dokumen pengajaran di institusi pengajian tinggi dengan berkesan, dan mempunyai nilai promosi dan aplikasi yang baik.

Kata kunci : Sistem Pengurusan Dokumen Pengajaran, Platform Web, Pangkalan Data MySQL.

Abstract

This study focuses on the problems existing in the teaching document management of the National University of Malaysia (UKM), such as multi - platform dispersion, chaotic permissions and uncontrollable progress. A web - based teaching document management system is designed and developed. The system uses MySQL as the back - end database, and combines role - based access control, file status tracking and version management to realize the centralized, structured and efficient management of teaching materials. Through comparative analysis with Google Drive, Amazon Drive and OneDrive, the system has outstanding performance in terms of security, scalability and advantages of local deployment. The final results have been verified by multiple rounds of tests, showing that the system has a user - friendly interface, complete functions and stable operation. It can effectively improve the management efficiency and standardization level of teaching documents in colleges and universities and has good promotion and application value.

Keywords： Teaching Document Management System, Web Platform, MySQL Database.

1.0 INTRODUCTION

With the development of educational informatization in colleges and universities, the standardized management of teaching documents has become an important basis for measuring teachers' performance and teaching quality (Elmasri & Navathe, 2020). At the National University of Malaysia (UKM), teachers are required to upload five types of teaching documents each semester, including course syllabuses, teaching plans, lecture notes, course materials, and summaries. However, at present, these documents are often submitted through channels such as Google Drive, UKMFolio, and emails, resulting in a fragmented process. This fragmentation leads to frequent problems such as version confusion and missing files.

In addition, administrators need to manually check the upload status of documents one by one, which results in a heavy workload and a high error rate. Meanwhile, the lack of a clear permission mechanism also makes the materials at risk of leakage. To solve these problems, UKM is in urgent need of an internally - designed teaching document management platform to realize the unified management of teaching tasks and document processing (Rosencrance, 2021).

This project builds a teaching document management system based on a Web platform. The back - end adopts the MySQL database, and the front - end ensures that teachers and administrators can efficiently upload, manage, and evaluate teaching materials within their respective permission scopes through role control, status tracking, and classification mechanisms.

2.0 LITERATURE REVIE

Colleges and universities are paying increasing attention to the management of teaching documents, and traditional methods of document submission and paper archiving can no longer meet the needs of large - scale course management. Studies have found that automatic assignment of teaching tasks, visualization of document status, and separate control of permissions are key to building an efficient teaching management system (Hoffer et al., 2016; Strauch et al., 2011).

The core of this system adopts the MySQL database, which has ACID transaction support, flexible data structure design, and stable concurrent access capabilities, making it suitable for long - term archiving and structured management of educational documents (DuBois, 2013; Silberschatz et al., 2019). Through database indexing, primary and foreign key constraints, and view control, it realizes the management of data integrity for documents across courses and semesters.

To demonstrate that this system is superior to mainstream cloud platforms, the project compared the performance of Google Drive, Amazon Drive, and OneDrive in terms of storage, security, permissions, and scalability. The results show that this system has obvious advantages in "embedded permissions", "high security", "structural design", and "support for high - concurrency processing".

Table 1 System comparison

	This system	**Google Drive**	**Amazon Drive**	**One Drive**
Account	√	√	√	√
Language	English	English	English	Multi-language support, including Chinese, English, French, German, Spanish, etc.
Security Performance	UKM development management, high security performance	The platform on the market has low security performance	SSL/TLS encryption, AES-256-bit storage encryption, no end-to-end encryption	TLS encryption, AES-256-bit encryption, end-to-end encryption (Vault feature)
storage capacity	High,100g for one person	15g is free, if you exceed 15g you need to pay extra	Storage capacity Free 5GB, paid plans start at 100GB	Free 5GB, paid plans from 100GB 1TB with Microsoft 365
Database	MySQL	NoSQL	AWS-based S3 service and distributed storage architecture	Based on Microsoft Azure Cloud Storage Platform
costs	Free	Storage capacity beyond a certain limit costs money	($1.99/month)	($1.99/month),

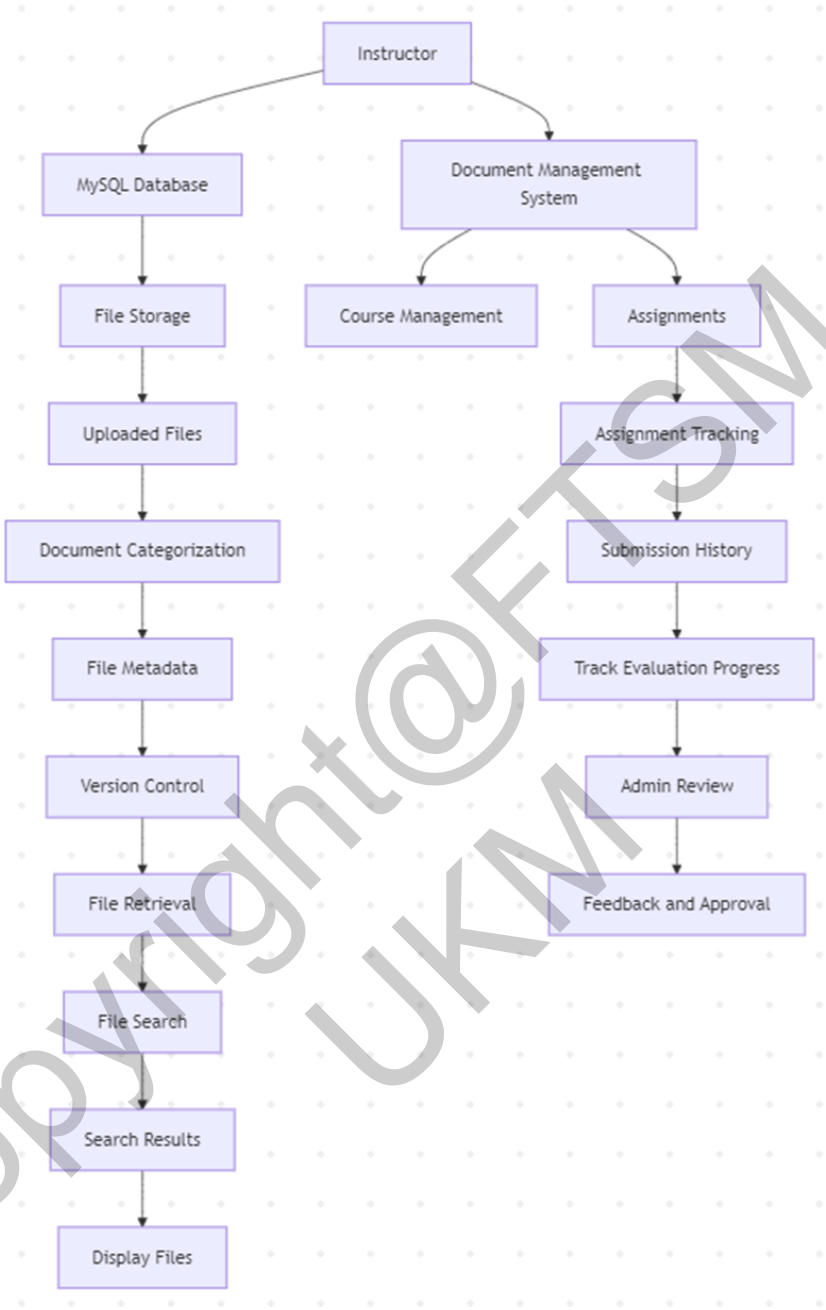
3.0 METHODOLOGY

This project adopts an agile development model for system construction, advancing system design, development, and optimization in phases. User feedback is incorporated into each iteration to ensure that the system design aligns with the actual operational processes of UKM teachers (Bartholomew, 2010).

In terms of system structure, the platform is implemented based on a B/S (Browser/Server) architecture. The front-end interface is concise and clear, supporting multi-

terminal browser compatibility; the back-end uses Java language to connect to the MySQL database and completes data interaction through RESTful APIs. The system logic includes teaching task assignment, file upload records, version management, and review processes, with all data managed uniformly in the database.

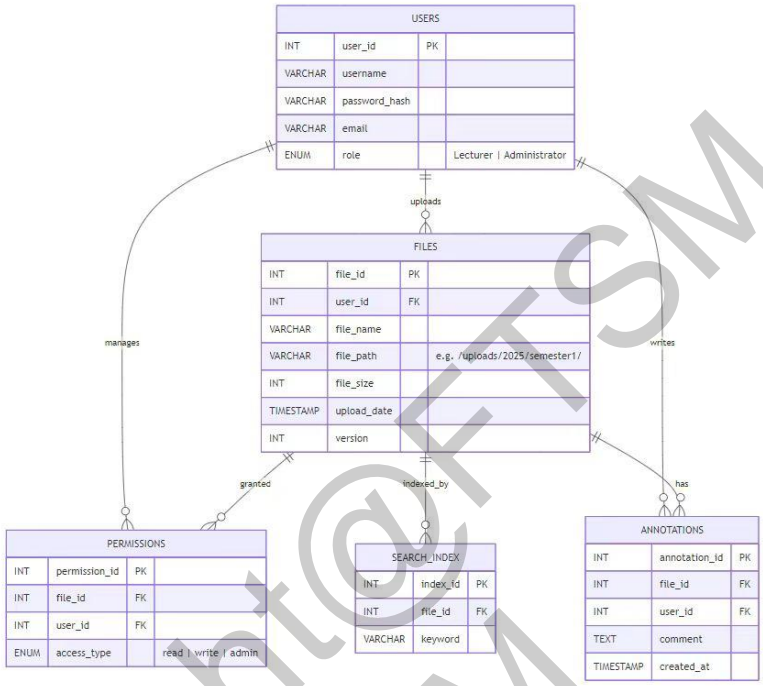
This flowchart presents the instructor operation process of the teaching document management system at the National University of Malaysia (UKM). Taking the instructor as the core role, work is carried out relying on the MySQL database and the Document Management System: On one hand, files are uploaded via File Storage, and go through successive steps such as Uploaded Files, Document Categorization, File Metadata, and Version Control, enabling File Retrieval and File Search, and finally Display Files. On the other hand, through the Course Management and Assignments modules, via Assignment Tracking, Submission History, and Track Evaluation Progress, it proceeds to Admin Review and obtains Feedback and Approval. It fully covers the entire process from file upload to management in teaching documentation, and from assignment distribution to review in the assignment process.



FIigure 1 Overall system flow

This is a database entity - relationship diagram, covering Users (USERS, including user ID, username, etc., with roles divided into Lecturer and Administrator), Files (FILES, including file ID, associated user ID, etc., recording upload information), Permissions (PERMISSIONS, managing the access permission types of users to files), Search Index (SEARCH_INDEX, associating files with keywords for searching), and Annotations (ANNOTATIONS, recording file - related annotations and creation

information). Each entity is associated through relationships, supporting data storage and interaction for business related to file management.

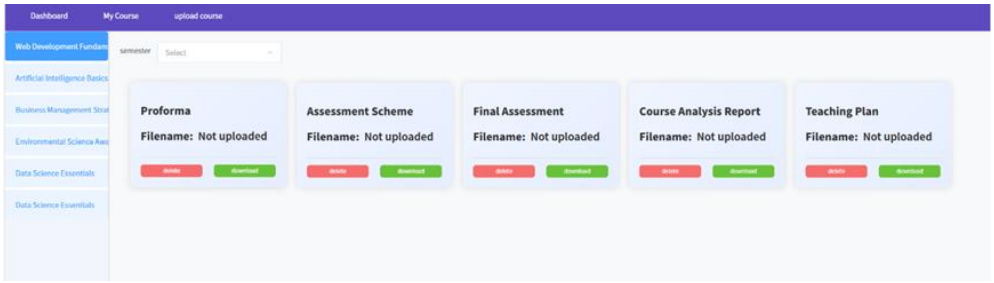


FIigure 2 ER diagram

4.0 RESULTS

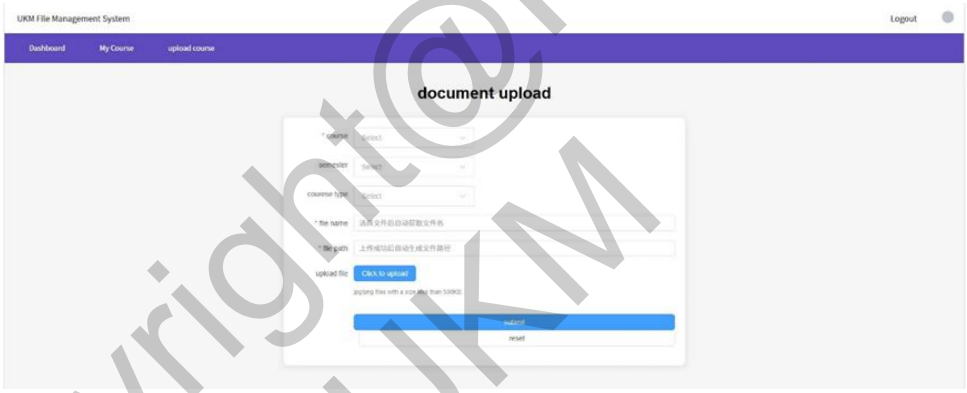
After the integration of the front - end and back - end of the system, a complete teaching file management function has been realized, including modules such as course allocation, file upload, file viewing, commenting, searching, and version control. The user operations are smooth, the interface is simple and intuitive, batch operations and status highlighting are supported, which effectively improves the efficiency of teaching task delivery.

This is the main interface of the system, the first page users see after logging in. It displays the system name "UKM File Management System" and the login status, with a logout button at the top. The sidebar provides clear navigation, allowing users to quickly access modules such as course management, instructor courses, semester management, and file management. The homepage design is simple, helping users efficiently find the functions they need and perform related operations, ensuring a good user experience.



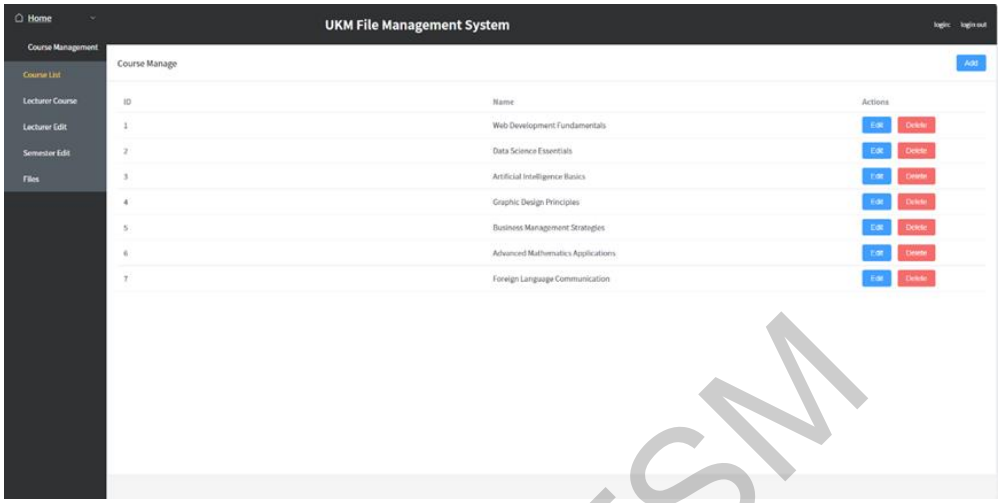
FIigure 3 Homepage

This is the course file upload interface, where lectures or administrator upload course-related files. Users can select course, semester, and course type, upload corresponding files, and specify file path and name. The page provides file size limitations to ensure uploaded file meets system requirements. Once uploaded, the file will be saved at the specified path for internal use and download by users.



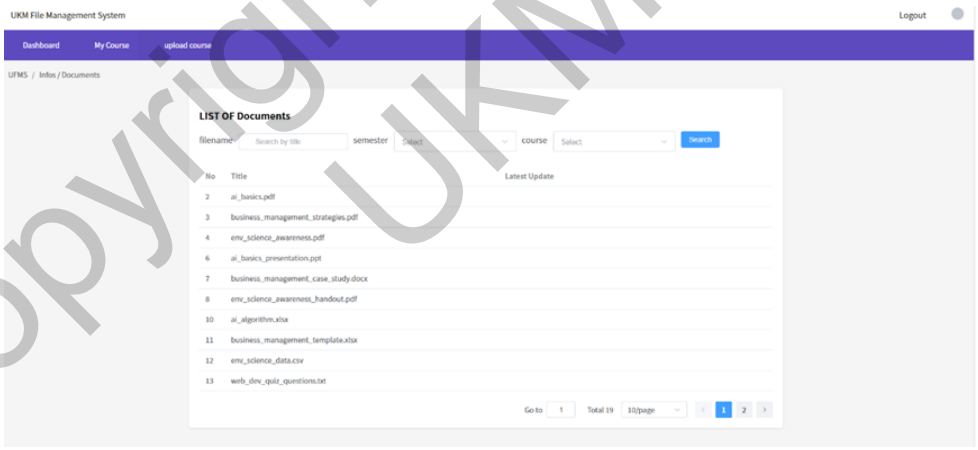
FIigure 4 File Upload Page

Figure 5 shows the course list management page, from which administrator can view detailed course details for all courses, including course ID and course name. By using the "Edit" and "Delete" buttons, the administrator can edit or remove course information. The "Add" button allows the administrator to add new courses, so the course information in the system is always up to date.



FIigure 5 Course List Management

Figure 6 illustrates the course file management interface of a course file, showing the file information of a course, such as file name, file type, and date of last file update. The file name may be clicked to download a file, and unwanted files may be deleted. This interface gives the teacher a single location to manage course-related files, with the capability to upload files and link them to other course data.



FIigure 6 Course File List Page

Through the design of the MySQL database, file data is stored in multiple tables, enabling unified query and management across semesters and courses. The system supports file formats such as PDF, DOC, and PPT, allows batch uploads, size limitations, automatic status recognition, and has a version number recording function. System testing covered aspects such as upload, download, search, concurrent response, and permission control. The test results show that the system demonstrates stability and compatibility in different network and device

environments and can meet the actual application requirements of teaching management (Kulshrestha & Sachdeva, 2014).

5.0 CONCLUSION

Studying abroad The UKM Teaching File Management System has successfully achieved the full - process digital management of teaching tasks and documents. Built on a Web platform and a MySQL database, the system covers core functions such as task allocation, document upload, permission control, review processes, and version recording. It has resolved the issues of dispersion, inefficiency, and insufficient security in the original system.

Compared with mainstream platforms, this system has a high degree of adaptability and customization capabilities. By adopting role - based permission control, status visualization, and structured data storage, it provides stable, efficient, and secure technical support for the teaching administration of UKM. Future expansion directions include: introducing AI - powered intelligent classification and content recognition, enhancing concurrent processing performance, and integrating message push and cross - campus collaboration functions (Angles & Gutierrez, 2008; Mell & Grance, 2011). Additionally, the system has the foundation for migration to cloud platforms, which can further improve its flexibility and scalability.

This project not only improves the quality of teaching document management at UKM but also provides a practical solution and implementation model for the modernization of digital governance and teaching evaluation in higher education institutions, making it highly valuable for reference and promotion.

6.0 REFERENCES

Angles, R., & Gutierrez, C. J. A. C. S. (2008). Survey of graph database models. ACM Computing Surveys, 40(1), 1–39. https://doi.org/10.1145/1322432.1322433

Bartholomew, D. (2010). MySQL High Availability: Tools for Building Robust Data Centers. O'Reilly Media.

Connolly, T., & Begg, C. (2015). Database systems: A practical approach to design, implementation, and management (6th ed.). Pearson Education.

DuBois, P. (2013). MySQL. Addison-Wesley.

DuBois, P. (2018). MySQL 8.0 Reference Manual. Oracle Corporation. https://dev.mysql.com/doc/refman/8.0/en/

Elmasri, R., & Navathe, S. B. (2020). Fundamentals of database systems (7th ed.). Pearson.

Greenwald, R., Stackowiak, R., & Stern, J. (2013). Oracle essentials: Oracle database 12c. O'Reilly Media, Inc.

Hoffer, J. A., Ramesh, V., & Topi, H. (2016). Modern database management (Vol. 4). Pearson.

Kulshrestha, S., & Sachdeva, S. (2014). Performance comparison for data storage - Db4o and MySQL databases. In Proceedings of the 2014 Seventh International Conference on Contemporary Computing (IC3) (pp. 329–334). IEEE. https://doi.org/10.1109/IC3.2014.6897194

Mell, P., & Grance, T. (2011). The NIST definition of cloud computing (Special Publication 800-145). National Institute of Standards and Technology. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf

Qin, L., Yu, J. X., & Chang, L. (2009). Keyword search in databases: The power of RDBMS. In Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data (pp. 681–694). https://doi.org/10.1145/1559845.1559917

Rosencrance, L. (2021). What is a file management system and how does it work? TechTarget. https://www.techtarget.com/searchcontentmanagement/definition/file-management-system

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). Database system concepts (7th ed.). McGraw-Hill Education.

Strauch, C., Sites, U.-L. S., & Kriha, W. J. L. N. (2011). NoSQL databases. Stuttgart Media University, 20(24), 79.

CUI KAI (A197931)
Dr. Khairul Azmi Abu Bakar
Faculty of Information Science & Technology
National University of Malaysia