

A Web Portal for Managing Educational Resources

Daniil Honcharenko, Radoslav Vargic

Slovak University of Technology in Bratislava, Faculty of Informatics and Information Technology, 84104, Bratislava
xhoncharenko@stuba.sk, radoslav.vargic@stuba.sk

Abstract – This paper presents the design and implementation of a centralized web portal for study materials management that addresses the challenges of digital transformation in education. Traditional methods of distributing educational materials suffer from fragmentation, limited organization capabilities, and accessibility issues. Our solution offers a hierarchical folder structure, robust user role management, and specialized viewing capabilities for various file formats. Built on Spring Boot with a layered MVC architecture, the system supports multiple file formats (PDF, video, ZIP) and provides role-based access control. User testing confirms improved accessibility and organization of study materials compared to existing solutions. The system effectively balances administrative flexibility with user accessibility, creating an efficient and centralized platform for educational content management without the complexity of full learning management systems.

Keywords - Study Materials Portal; Spring Boot; Web Application; File Management; User Roles; Web Development

I. INTRODUCTION

The digital transformation of education has significantly accelerated in recent years, shifting focus from traditional paper-based learning materials to digital resources. This transition presents both opportunities and challenges for educational institutions. While digital materials offer enhanced accessibility and distribution potential, many institutions lack effective tools for organizing, managing, and distributing these resources in a structured and user-friendly manner.

Current approaches often rely on ad-hoc methods such as email distribution, generic cloud storage, or basic static HTML pages. These solutions typically lack proper organization, version control, role-based access management, and specialized viewing capabilities needed in educational settings. Additionally, full-featured Learning Management Systems (LMS) like Moodle often represent excessive complexity when the primary need is centralized study materials management.

The existing Study Materials Portal at FEI STU (psm.fei.stuba.sk) illustrates these limitations. While providing basic access to materials, it relies on static HTML pages with manual content management, lacks user roles and permissions, and offers minimal dynamic functionality. This traditional approach, while simple and performant, fails to meet the evolving needs of modern educational institutions.

This paper presents the design and implementation of a specialized web portal for study materials management that bridges the gap between overly simplified static solutions and complex LMS platforms. The system offers hierarchical organization of materials, role-based access control, online preview capabilities, and a user-friendly interface while maintaining operational simplicity.

II. GOAL OF THE WORK

The primary objective of this work is to design and implement a functional prototype of a web portal for managing study materials that addresses the limitations of existing solutions. The portal aims to provide the following capabilities:

- Create a centralized and user-friendly environment for content creators (teachers, administrators) to upload and organize study materials efficiently
- Implement a logical hierarchical structure of folders and courses for effective content organization
- Support various file formats (primarily PDF, video, ZIP) with specialized viewing capabilities
- Incorporate a link-type item system that enables redirections to external resources or internal portal locations
- Develop a role-based user system (administrator, teacher, student) with clearly defined permissions
- Provide a clean, intuitive interface for both content managers and students accessing materials

The solution seeks to find the optimal balance between sophisticated content management capabilities and interface simplicity, avoiding the overengineering typical of comprehensive LMS while offering significantly more functionality than static web pages

III. WORK METHODOLOGY AND RESEARCH METHODS

The development process combined analysis, requirements engineering, design, and implementation phases. As shown in Fig. 1, our methodology followed a systematic approach to ensure comprehensive coverage of all aspects of the system development.

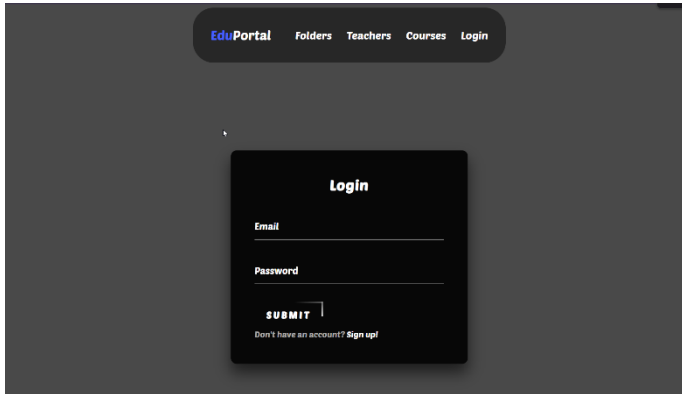


Figure 1. Login page

- Course management within folders with ownership and teacher assignments
- File management with metadata, download, and deletion capabilities
- Content browsing with online previews for supported formats

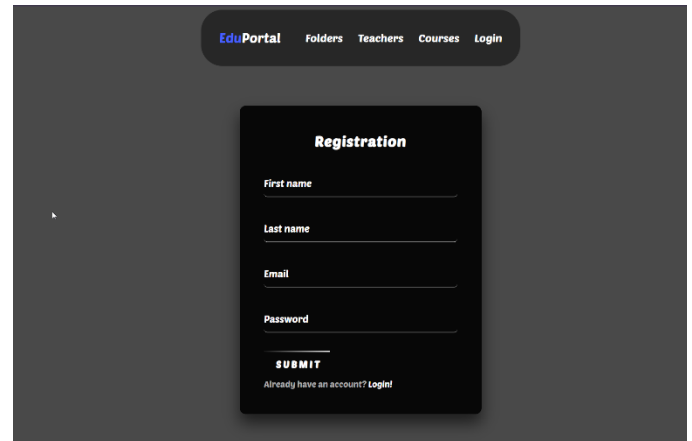


Figure 2. Registration Page

A. Analysis of Existing Solutions

The first phase involved a comprehensive review of existing study materials management platforms to identify their strengths and limitations.

The current PSM (Study Materials Portal) at FEI STU uses static HTML pages with limited functionality and manual content management. While offering simplicity for end users and low infrastructure requirements, it lacks flexibility, dynamic functionality, user roles, and proper content organization tools.

Learning Management Systems like Moodle provide comprehensive educational capabilities including course management, assignment submission, grading, and discussion forums. While highly flexible and feature-rich, these systems introduce significant complexity, administration overhead, and focus on course-centric rather than material-centric organization.

The analysis revealed a clear gap for solutions that offer structured content management without the full complexity of LMS platforms.

B. Requirements Specification

Based on the analysis and stakeholder inputs, functional and non-functional requirements were defined:

Functional Requirements:

- User and role management with registration, authentication, and role requests, as illustrated in Fig. 2 showing the registration interface
- Hierarchical folder management with ownership and link-type capabilities

Non-Functional Requirements:

- Security through authentication, password hashing, and access control
- Usability through intuitive interfaces and navigation
- Scalability to accommodate future expansions
- Reliability and system stability
- Technology constraints: Java, Spring Boot, Thymeleaf, MySQL

C. System Architecture Design

The system was designed using a layered architecture approach combined with MVC pattern to ensure modularity, scalability, and separation of concerns:

- Presentation Layer: Controllers, Views (Thymeleaf), Model
- Service Layer: Business logic components
- Data Layer: Entities and repositories (Spring Data JPA)
- Domain Layer: Entity definitions
- Security Layer: Authentication and authorization (Spring Security)

This architecture ensures clear separation between database interactions, business logic, and presentation, facilitating maintenance and future extensions.

D. Implementation Data Model Design

A comprehensive data model was designed to support all system requirements, as shown in Fig. 3:

- users: User accounts with personal information and role assignments
- folder: Hierarchical folder structure with support for link-type folders
- course: Course definitions within folders, potentially link-type
- file_metadata: Information about uploaded files with references to courses
- teacher_course: Many-to-many relationship between teachers and courses
- license_requests: Tracking requests for teacher role upgrades
- confirmation_token: Supporting email confirmation functionality

The model was designed to efficiently represent the relationships between entities while supporting all required functionality.

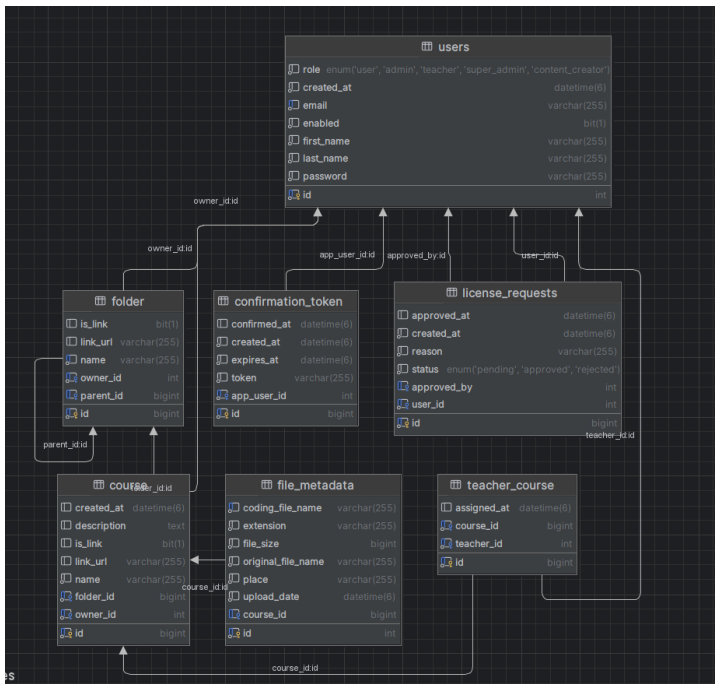


Figure 3. Database entity relationship diagram showing the relationships between key entities in the system

E. Implementation Approach

- Core user management with registration and authentication
- Folder and course management capabilities
- File upload and management functionality
- Online preview features for supported formats
- Role-based access control implementation
- User interface refinement

Each component was developed and tested incrementally to ensure proper functioning before integration.

IV. PROPOSED SOLUTION

The implemented solution features a modern web-based system built on proven technologies:

Technology Stack:

- Backend: Java with Spring Boot, Spring Data JPA, Hibernate, Spring Security
- Frontend: Thymeleaf templates with HTML, CSS, JavaScript, and Bootstrap
- Database: MySQL for persistent data storage
- Build Tool: Apache Maven

Key Components:

- Registration with email verification
- Role-based authentication (Student, Teacher, Administrator)
- Role upgrade request workflow
- User profile management

Content Organization:

- Hierarchical folder structure with unlimited nesting
- Course containers within folders
- Special link-type items for both folders and courses
- Teacher assignments to courses

File Management:

- Upload interface with metadata capture
- File storage with appropriate access controls
- Download capabilities with access verification
- Online preview for supported formats (PDF, video)

Security Implementation:

- Password hashing using BCrypt
- CSRF protection
- Input validation
- Role-based access control
- Session management

As demonstrated in Fig. 4, the solution provides an intuitive folder browser interface with nested folders and search functionality, enabling efficient navigation through the content hierarchy. This interface exemplifies our approach to providing streamlined access to educational materials with clear organization and contextual actions based on user roles and permissions..

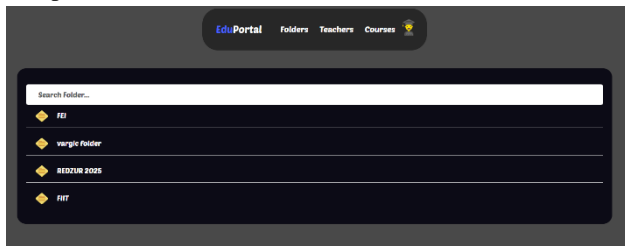


Figure 4. EduPortal – folder browser UI with nested folders and search functionality

V. BENEFITS AND EVALUATION

The developed system offers numerous advantages over existing solutions:

Compared to Static HTML Portals:

- Role-based access instead of open access
- Structured content organization
- Dynamic content management without HTML knowledge
- Online preview capabilities
- User management and personalization

Compared to Full LMS Solutions:

- Simplified interface focused on materials management
- Lower administrative overhead
- More flexible content organization (not strictly course-centric)
- Reduced complexity for both administrators and users
- Faster learning curve for content creators

General Benefits:

- Centralized content repository with consistent access patterns
- Clear ownership and permissions model
- Structured browsing experience
- Fast and responsive interface
- Minimal infrastructure requirements

Initial testing with a representative user group confirms improved accessibility and organization of study materials compared to existing solutions, with particularly positive feedback on the hierarchical structure and online preview capabilities.

VI. CONCLUSION

This work addresses the need for effective study materials management in educational institutions undergoing digital transformation. The implemented web portal bridges the gap between overly simplified static solutions and complex LMS platforms, providing hierarchical content organization, role-based access control, and specialized viewing capabilities.

The layered architecture and MVC pattern ensure separation of concerns and facilitate future extensions. The role-based system effectively balances administrative flexibility with proper access controls. The online preview capabilities enhance user experience by reducing the need to download files before viewing content.

Future work could focus on enhanced search capabilities, integration with external authentication systems, mobile application development, and expanded file format support. The current implementation establishes a solid foundation for these extensions while already providing significant improvements over existing solutions.

ACKNOWLEDGEMENT

This paper was supported by NEXT (ERASMUS-EDU-2023-CBHE-STRAND-2), EULIST (ERASMUS), CYB-FUT (ERASMUS+).

REFERENCES

- [1] VMware, Inc. "Spring Framework Documentation." [Online]. Available: <https://spring.io/projects/spring-framework>
- [2] VMware, Inc. "Spring Boot Documentation." [Online]. Available: <https://spring.io/projects/spring-boot>
- [3] VMware, Inc. "Spring Security Reference." [Online]. Available: <https://docs.spring.io/spring-security/reference/>
- [4] Oracle Corporation. "MySQL Documentation." [Online]. Available: <https://dev.mysql.com/doc/>
- [5] Thymeleaf Team. "Thymeleaf Documentation." [Online]. Available: <https://www.thymeleaf.org/documentation.html>
- [6] Bootstrap Team. "Bootstrap Documentation." [Online]. Available: <https://getbootstrap.com/docs/>
- [7] Mozilla Foundation. "MDN Web Docs: HTML." [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTML>
- [8] Mozilla Foundation. "MDN Web Docs: CSS." [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/CSS>
- [9] Mozilla Foundation. "MDN Web Docs: JavaScript." [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [10] Java EE. "Jakarta EE Specification." [Online]. Available: <https://jakarta.ee/specifications/platform/>