Software Design Description (SDD)

Web-Based Student Note Sharing Platform

Nisa Begüm Burucu (64210025)

Utku Öztekin (64220012)

Elif Serra Uçar (64220058)

Deniz Kenan Ek (64210044)

Kerem Döleksöz (64210022)

Supervisor: Sibel Tarıyan Özyer

(Ankara Medipol University-Computer Engineering Department)

1. INTRODUCTION

1.1 Purpose

This document has been prepared to describe in detail the software design of the **Course Notes Upload and Sharing System**, which is developed as part of a university graduation project. The document follows the IEEE 1016 Software Design Description (SDD) standard and defines the architectural structure, components, data design, interfaces, and security mechanisms of the system.

The purpose of this SDD document is to clearly and comprehensively present how the software is designed for developers, the project advisor, and evaluators.

1.2 Scope

The developed system is a web-based course notes sharing platform designed for educational institutions. The system includes three different user roles: **Student**, **Instructor**, and **Administrator**.

- Students can view, download, comment on, and rate course notes.
- Instructors can create courses, upload course notes, and review comments.
- Administrators can manage users, courses, banners, and overall system settings.

The system consists of frontend, backend, and database layers.

1.3 Definitions, Acronyms, and Abbreviations

- **SDD**: Software Design Description
- **IEEE**: Institute of Electrical and Electronics Engineers
- **Admin**: System Administrator
- **Frontend**: User Interface Layer
- **Backend**: Server-side Business Logic Layer
- **API**: Application Programming Interface

1.4 References

- IEEE Std 1016-2009 – Software Design Description
- Software Engineering Course Materials

1.5 Document Overview

This document consists of the following main sections:

- System overview
- Architectural design
- Data design
- Component design
- Interface design
- Security considerations

2. SYSTEM OVERVIEW

The Course Notes Upload and Sharing System is a centralized web application in which users are authorized based on their roles. The system provides a secure platform where users can log in, access course materials, and interact with content.

The system is designed to be scalable, secure, and user-friendly.

3. DESIGN CONSIDERATIONS

3.1 Assumptions

- Users access the system via modern web browsers.
- Continuous internet connection is required.
- User-provided information is assumed to be accurate.

3.2 Constraints

- The database follows a relational model.
- Role-based authorization is mandatory.
- Passwords are not stored in plain text.

4. ARCHITECTURAL DESIGN

The system is designed using a **Three-Tier Architecture**:

4.1 Presentation Layer

- User interfaces
- Student, Instructor, and Admin panels
- HTML, CSS, and JavaScript-based structure

4.2 Business Logic Layer

- User management operations
- Course and note management
- Comment and rating processing

4.3 Data Layer

- Database management
- CRUD operations

5. DATA DESIGN

5.0 Academic Semester and Enrollment Structure

To accurately model a real university system, the software is designed with an **academic semester** structure. Courses are created for specific semesters and archived once the semester ends.

Students can access only the courses in which they are enrolled. The relationship between students and courses is modeled as a many-to-many relationship.

5.1 Database Schema

Users Table

- user_id (PK)
- name
- email
- password
- role

Courses Table

- course_id (PK)
- course_name
- instructor_id (FK)
- semester_id (FK)

Enrollments Table

- enrollment_id (PK)
- user_id (FK)
- course_id (FK)

Notes Table

- note_id (PK)
- course_id (FK)

- file_path
- upload_date
- version

Comments Table

- comment_id (PK)
- note_id (FK)
- user_id (FK)
- comment_text
- comment_date

Ratings Table

- rating_id (PK)
- note_id (FK)
- score

Banners Table

- banner_id (PK)
- title
- image_path

6. COMPONENT DESIGN

6.0 Role-Based Access Control (RBAC)

The system implements a **Role-Based Access Control (RBAC)** mechanism. Each user is allowed to perform only the actions permitted for their role. Authorization checks are enforced at both backend and frontend levels.

Authorization Matrix:

- **Administrator**:
    - Manage users (create, update, delete)
    - Approve instructor accounts
    - Manage courses and banners
    - View and remove all content
- **Instructor**:
    - Create and manage own courses
    - Upload, update, and delete notes for own courses
    - View comments related to own courses

- **Student**:
    - View enrolled courses
    - Download course notes
    - Add comments and ratings

6.1 User Management Component

- User registration
- Login and logout
- Role verification

6.2 Course Management Component

- Create and delete courses
- List courses

6.3 Note Management Component

- File upload
- File deletion
- Version control

6.4 Comment and Rating Component

- Add comments
- Moderate comments
- Rating functionality

7. INTERFACE DESIGN

7.1 User Interfaces

- Login and registration screens
- Student panel
- Instructor panel
- Administrator panel

7.2 System Interfaces

- RESTful API services

8. SECURITY DESIGN

- Password hashing
- Authorization checks
- Session management

9. ERROR HANDLING

- Invalid login warnings
- File upload errors
- Unauthorized access prevention

10. ENHANCED SYSTEM FEATURES

This section describes the top ten enhancements implemented to improve the system's functionality and academic value.

10.1 Role-Based Authorization (RBAC)

Ensures secure access control by validating user roles before executing any operation.

10.2 Course Enrollment System

Students must enroll in courses to access their content, reflecting real university workflows.

10.3 Academic Semester Management

Courses are organized by academic terms (Fall/Spring), and completed terms are archived.

10.4 Course Note Versioning

Multiple versions of the same course note are maintained, allowing access to previous versions.

10.5 File Type and Size Validation

Only approved file formats are allowed, and file size limits protect system performance.

10.6 Advanced Search and Filtering

Users can search and filter notes by course, instructor, upload date, or rating.

10.7 Notification System

Users receive system notifications when new course notes are uploaded.

10.8 Favorite Notes Feature

Students can bookmark frequently used notes for quick access.

10.9 Comment Moderation

Admins and instructors can review and remove inappropriate comments.

10.10 Statistics and Reporting Panel

The admin panel provides statistics such as most downloaded notes and most active instructors.

11.REFERENCES

[1] I. Sommerville, *Software Engineering*, 10th ed., Pearson Education, 2016.

[2] R. Pressman and B. Maxim, *Software Engineering: A Practitioner's Approach*, 8th ed., McGraw-Hill, 2015.

[3] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1994.

[4] M. Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3rd ed., Addison-Wesley, 2004.

[5] A. Dennis, B. Wixom, and D. Tegarden, *Systems Analysis and Design with UML*, 5th ed., Wiley, 2015.

[6] OWASP Foundation, *OWASP Top 10 – Web Application Security Risks*, 2023.