LMS - Learning Management System Specification It is necessary to create a Learning Management System for a university. The application should enable the administration of staff, students, and university study programs. The application needs to be implemented as a web application, which also serves as the university's webpage. It is expected that a large number of users will use the application simultaneously. System participants include students, teachers, and administrators, but unregistered users can also visit the university's webpage. Unregistered users can:

- View the university's webpage and see basic information about the university. This includes contact information, university location, description, and information about the rector.
- View individual faculty pages. Information includes contact details, faculty location, description, and information about the dean.
- View study programs at faculties, with a list of subjects and a description of the study program, as well as information about the program manager.
- View subject syllabi.
- View teaching materials for subjects.
- Register. Registered users can:
- Log in to the system.
- Edit their profile.
- Log out of the system. In addition to the functionalities of registered users, students can:
- View information about the subjects they are currently enrolled in.
- Access notifications for the subjects they are currently enrolled in.
- View their academic history, including previously completed subjects, the number of attempts for each subject, final grades, and the average grade, as well as the total number of earned ECTS credits during their studies.
- Register for exams for the subjects they are currently enrolled in (if the exam registration for the next session is active). In addition to the functionalities of registered users, teachers can:
- View information about the subjects they are currently teaching.
- Edit subject syllabi for the subjects they are currently teaching.
- Create schedules for outcomes by terms. Each term has an outcome or topic assigned.
- Define evaluation instruments (project assignments, tests, quizzes).
- Manage notifications for the subjects they are teaching.

- View a list of students for each subject they are teaching.
- Use student search by name, surname, student ID, enrollment year, average grade, or any subset of these attributes.
- View student data, including:
    - Basic information.
    - Average grade and earned ECTS credits during their studies.
    - Enrollments.
    - Passed exams (subjects) with grades and scores for each exam.
    - Failed attempts with grades and scores.
    - Registered exams.
    - Students' final papers.
- Enter grades for exams for the subjects where they are assigned as professors/lecturers. Grades can only be entered within 15 days after the exam date. Administrators can:
- Administer codebooks.
- Administer registered users.
- Administer study programs.
- Administer the organization.
- Add teachers and administrative staff. General Requirements:
- Enable data export for teachers and students in XML and PDF formats.
- Enable the addition of evaluation results (quizzes, exams, etc.) in the form of XML documents, either by specifying the path to the XML document for upload or by pasting the entire XML document into a form field. Provide validation for uploaded XML documents.
- Enable export of evaluation results in PDF format. General Requirements - Additional:
- Select one entity in the system and store it as RDF triples. When working with this entity, data should not be retrieved from the relational database but from the RDF store.
- To maximize points, implement the RDF store using TDB/Fuseki.
- Provide all CRUD functionalities and search by all properties for the selected entity.

Individual Tasks

Task 1: Implement a messaging system. Messages can be exchanged by all system participants. It should also be possible to have "group chats" by sending messages to multiple participants. Video calls and video conferences should also be possible.

Task 2: Implement a fully functional forum. All registered users can participate in the forum, while unregistered users can only browse. Users can subscribe to subforums and topics. When a new topic is created in a subscribed subforum or a new message is posted in a subscribed topic, subscribed users should be notified via email.

Task 3: Implement a notification system. When a teacher posts a notification for a subject they are teaching, all students currently enrolled in the subject and other teaching staff involved in the subject should be notified via email.

Task 4: Implement a file version tracking system, including the ability to revert to previous versions and compare the current and previous versions. This applies to all attachments in the LMS.

Task 5: Implement a career tracking system for teachers. When a teacher's academic rank changes in the system, a new entry should be generated that records the current rank and the history of all previous ranks chronologically. For each rank, information about the rank title, date of appointment, and end date should be maintained.

Task 6: Implement a system for creating and updating exam sessions. Implement the solution as an interactive calendar where the start and duration of exam sessions can be selected. Exam sessions cannot overlap. Access to these functionalities is restricted to administrators.

Task 7: Implement a reporting system for exam results. Display the number of students who passed, the number of students who failed, and the distribution of earned points divided into 10 categories. Display the distribution of points and the ratio of students who passed and failed graphically, using a Pie chart for the student ratio and a bar chart for the point distribution.

Task 8: Implement a system for checking prerequisites for enrolling in a study year. Check the total number of earned ECTS credits and whether the student has passed the prerequisites for subjects in the year they want to enroll in. The check is performed automatically for all students at the end of the semester (based on a specified date) or upon a special request from administrators. After the check is completed, students should be notified by email about whether they meet the enrollment criteria and any unmet prerequisites.

Task 9: Implement a system for creating class schedules. Implement the solution as an interactive table. When creating schedules, specify the subject, teacher, start time, duration, and classroom. It is not possible to exceed the teacher's workload limit of 30 hours per week, and when entering times for a subject, the number of lecture and exercise hours specified in the subject syllabus must be entered. After creating the schedule, validation should be performed to check for teacher assignment overlaps or classroom conflicts.