Advanced Operating Systems (CSG623)

Distributed Learning Management System with gRPC

Milestone 1 Report

1. Introduction

The project aims to develop a **Distributed Learning Management System (LMS)** using **gRPC** for backend communication. LMS will be used for interaction between students and Teachers. LMS includes the following features **authentication**, **assignment management**, **file upload and download**, and a **doubt-answering system**.

2. System Architecture

The LMS is implemented using a **client-server architecture** where the server handles all operations related to user management, assignment submission, grading, and doubt resolution.

The system's main components are:

- **gRPC Server**: Manages authentication, assignment uploads, submissions, and grading.
- **Protobuf Definitions**: Provides the structure for the communication protocol between the client and the server.
- **Storage**: JSON files are used to manage data related to users, assignments, submissions, and doubts.

Key Files and Directories:

- **Server-side JSON Files**:
  - ***users.json***: Stores user credentials and roles.
  - ***assignments.json***: Contains metadata about assignments.
  - ***submissions.json***: Keeps track of student submissions.
  - ***doubts.json***: Handles student queries and teacher responses.
- **Folders**:
  - ***assignments/***: Stores student-submitted files.
  - ***questions/***: Contains the assignment question files uploaded by the teacher.

3. Implemented Features

3.1. User Management

The system provides a **registration and login** system for students. A hardcoded teacher account is created:

- **RegisterStudent:** Used to register a new student into the system, saving their credentials in the users.json file.
- **Login:** This method validates a user's credentials and generates an authentication token for session management.

3.2. Assignment Management

Assignments are managed and stored in the form of PDFs:

- **CreateAssignment**: Allows teachers to upload new assignments (in PDF format), which are stored in the questions folder. Metadata related to the assignments is saved in the assignments.json file.
- **ViewQuestions**: Students can view the available assignments and download the assignment PDFs.

3.3. File Upload and Download

- **UploadFile**: Students can upload their assignment submissions to the server. The submitted files are saved in the assignment folder, with the file path saved in the submissions.json file.
- **DownloadFile**: Allows teachers to View and download a student's submission for grading.

3.4. Submission Management

The server tracks student submissions:

- **ViewSubmission**: Enables students to see their assignment submissions, including whether they have uploaded files for specific assignments.
- **ViewSubmittedAndPendingAssignments**: Shows students which assignments they have submitted and which assignments are still pending.

3.5. Grading

Teachers can assign grades to submitted assignments:

- **Grade**: Teachers can assign grades (A to E) for student submissions by updating the submissions.json file. If a student has not submitted an assignment, then FAIL grade is assigned automatically.

3.6. Doubt Handling

Students can submit doubts regarding their assignments, which can be answered by the teacher:

- **AddDoubt**: Allows students to submit doubts. These doubts are stored in the doubts.json file with the "unanswered" tag.
- **AnswerDoubt**: Teachers can respond to submitted doubts, moving them from "unanswered" to "answered".
- **ViewDoubt**: Students and Teachers can see "answered" and "unanswered" doubts.

4. Conclusion

In this milestone, we successfully laid the groundwork for the **Distributed LMS using gRPC** with core features like user management, assignment upload/download, grading, and doubt resolution. This project is on track, with the core components functioning as expected. We aim to build upon this work in the next milestone by refining existing features, improving user experience, and expanding the system's capabilities to support a broader range of LMS functionalities.

SUBMITTED BY-

1. *Pranjay Malhotra (2024H1030072P)*
2. *Sanskar Asati (2024H1030088P)*
3. *Shail Nilesh Shah (2024H1030074P)*