

Distributed Grading System

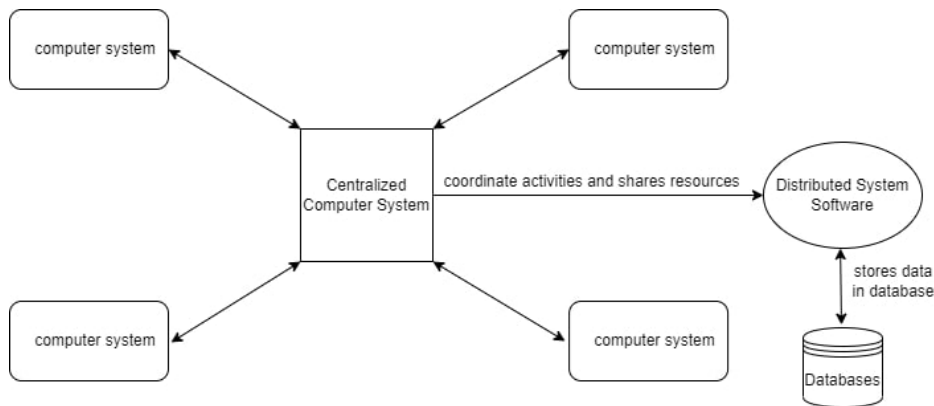
<https://github.com/Fcozer/Distributed-Grading-System.git>

Furkan Can Özer – 17290115
Murat Yıldırım – 17290132
Alptekin Yanık – 17290130
Elif Hale Gökçen - 19290803
ROZANA MAMOUN SALEEM - 20291229

Distributed Grading System

What is Distributed System?

The term "distributed system" refers to a group of independent computer systems that are linked through a centralized computer network using distributed system software despite being physically separated from one another. Each system's autonomous computers will interact with one another by exchanging files and resources while carrying out the duties given to them.



Computers can coordinate their efforts and share resources, such as hardware, software, data, etc., thanks to distributed system software.

Database: This is where the data that has been processed by each node or system of the distributed systems linked to the centralized network is kept.

Features of a Distributed System

The capacity to use any Hardware, Software, or Data anyplace in the System is known as resource sharing.

Openness: It is interested in system extensions and upgrades (i.e., How openly the software is developed and shared with others)

Concurrency is a feature of Distributed Systems, which deal with the same activity or functionality that can be carried out by various users who are located in different places. Each local system has unique resources and operating systems.

Scalability: It expands the system's scope as a number of processors interact with more users by adapting to enhance the system's responsiveness.

Fault tolerance: This refers to the system's capacity to continue functioning properly in the event of a hardware or software breakdown without suffering any performance degradation.

Transparency: As there should be privacy in every system, it hides the complexity of distributed systems from users and application applications.

Heterogeneity: Dispersed system components may differ from one another in terms of networks, computer hardware, operating systems, programming languages, and developer implementations.

Advantages of Distributed System

- Distributed applications exist by definition in distributed systems.
- Distributed systems share information among users who are spread out geographically.
- Resource Exchange (Autonomous systems can share resources from remote locations).
- It is more flexible and has a superior price-performance ratio.
- It features a faster throughput and a quicker reaction time.
- It is more resistant to component failure and has improved availability and dependability.
- It is extensible, allowing for both gradual growth and the extension of systems to more distant places.

Our Application

The number of students attending universities today is rising daily. As a result, when the academic staff wants to give students homework that is related to the courses they are taking, they add to their own and the research assistants' workloads. In addition, research assistants are given the responsibility of controlling assignments since institutions lack sufficient training personnel. The people tasked with conducting research are unable to do so because of this predicament. A project that can convert homework control into a distributed system is required as a remedy. The creation of a system for student self-grading is intended to address this issue. Distributed grading is a project that uses grade sharing, in which students assign grades to one another, to lighten the load on lecturers in classrooms when there are a lot of students. The goal of this project is to develop a fair and transparent grading system. All of the participants in this project who will take notes are situated to provide grades concurrently. Since there is a transparent grading system, it is expected that each student will gain mastery of the subject by carefully reviewing each assignment within the context of the course he or she has graded. It is also expected that students will establish the most precise cause-and-effect relationships on the subject. By offering transparency, this technique seeks to encourage participants to be fair and prevent potential conflicts in grading.

On the other hand, it aims to lessen the workload of faculty members in grading in universities where teaching assistants are scarce so that students can complete more varied and fruitful assignments and researchers and academicians can devote more time to their research, which they are unable to carry out due to the increasing homework load. Students will be able to examine, comment on, and grade the assignments of the students who have been allocated to them as part of the project. The identity of the students will be kept private throughout the interpretation and grading process, and only the students whose assignment has received comments will be able to see them. A professional will be present in this system in addition to the pupils. The lecturer who teaches the course is chosen as the expert by default, however other people can be in this role if they so desire. Students will be able to grade the amount of homework that the expert has set for each student to check. The required quantity of homework, which will be distributed at random to pupils, will be reviewed by them.

As a result, we decided to use Spring Boot, which is a Java framework on the backend, and React, a JavaScript framework, on the frontend, to design and manage this application of this capacity and complexity.

Spring Boot and Spring Security

Making standalone, professional Spring-based applications that you can "simply launch" is simple using Spring Boot.

In order to help you get started quickly, we take an opinionated stance on the Spring platform and third-party libraries. Spring setup is typically not necessary for Spring Boot apps.

A strong and incredibly configurable framework for access control and authentication is called Spring Security. In terms of protecting Spring-based apps, it is the de facto standard.

A framework called Spring Security is dedicated to giving Java applications authentication and authorization. Like all Spring projects, Spring Security's main strength lies in how easily it can be customized to fit specific requirements.

In our project, we paid attention to the limited privileges of users. There are 4 types of users in our system.

Admin: Has the authority to add other users to the system. User can add and delete.

Academician: Can see the assignments that students have uploaded. Can assign homework to students.

Assistant: Has the same authority as the academician attending the course.

Student: Can see the assignments given for the course. Can upload the answers of the incoming assignments to the submit page. Can score uploaded assignments.

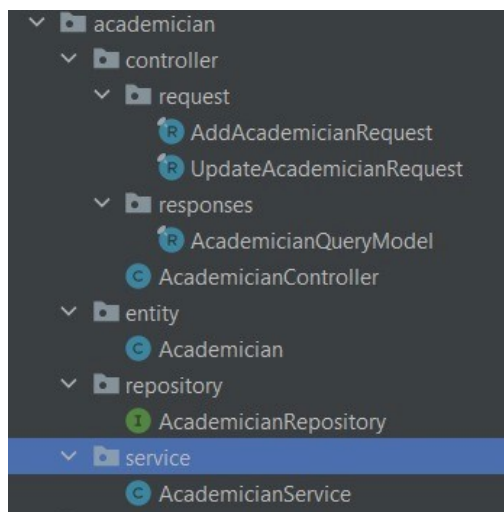
And also we have 4 more types of services in our system.

Lesson: After creating the users on the admin screen, the courses to be given to the academician users are assigned by the admin and the academicians are in a position to give their students homework through their lessons.

Homework: After the academicians log in to their systems with their own usernames and passwords, they create the homeworks based on the lessons given to them and forward these homeworks to the students taking the lesson.

Homework Submit: Students upload their assignments to the system through this service.

Homework Grade: Students score assignments uploaded through this service.



Here we give an example of the code architecture we have prepared in the backend. When we add a user or a functional service to the system, we split it into packages to write efficient and effective code. Each package contains four separate packages.

- Entity
- Repository
- Service
- Controller

Entity

It is aimed to write data types in the Entity package and create tables in the database. This package also establishes relationships with other services. In this way, data is retrieved from other services through the entity package and rusted. We add new tables to our database with the established relationship.

Repository

In this package, we use JPA and JPA is a standard that emerged in the Java programming language for associating Java classes with relational database tables. The purpose here is to match the class variables with the columns of the table and to perform database operations directly on the objects without writing SQL. At this point, JPA only sets a standard and does not take any action on the data itself.

Service

The service package is the package that we will use the data we have and process it according to our needs. Business rules are literally written in this package.

For example, in our application students do not just upload their homework to the system. They also grade their friends' assignments, and a student's assignment is scored by multiple students. It is precisely in this service that this data is collected and the score given by the students to a single student is calculated. After it is calculated, it is divided by the total number of students and the average score is calculated here.

Controller

In this package, we make the data we process ready for the user screen. In this package, a relationship is established with the interface via API. We establish our relationship between the two ends by using the Get, Post, Delete and Put methods.

To give an example from the picture above, there are two separate sub-packages for request and response in the Controller. When we use the post and put method, requests follow our AddAcademicianRequest and our UpdateAcademicianRequest. Also, when we use the get method, a response is sent from AcademicianQueryModel.

What is React and how we used in our application?

React is a front-end JavaScript toolkit that is free and open source for creating user interfaces based on UI components. It is kept up to date by Meta (previously Facebook) and a group of independent programmers and businesses. With frameworks like Next.js, React can be the foundation for single-page, mobile, or server-rendered applications. Making React apps typically necessitates the use of extra libraries for routing and specific client-side functionality because React is only concerned with state management and presenting that information to the DOM.

Information comes from the backend service written with Spring Boot to the frontend written with React. All operations such as displaying information properly, buttons directing the user to another page, etc. are possible with the frontend written in React.

Information comes from the backend service written with Spring Boot to the frontend written with React. All operations such as displaying information properly, buttons directing the user to another page, etc. are possible with the frontend written in React.

Some pages are viewed as common, according to user authorization. However, due to authorization differences between users, users see different pages even if they are viewing the same information. For example, in the system, a user with academic authority sees a page with buttons such as "addHomework" on the homework screen. However, since the student user is not authorized to assign homework to the class, he only sees the assigned homework on the homework page. Even if the Backend service pulls this data from a common table, on the Frontend these are two different pages.

Summary

Our system allows users to log in with their own user name and password. Each user has an authorization level. These privileges determine what users can and cannot do. As we mentioned before, there are 4 types of users. These are Admin, Academician, Student and Assistant, respectively. If we look at the system with a framework, we can briefly summarize it as follows:

The academician is the instructor of a course registered in the system and this course also has an assistant. The academic may assign an assignment for the course. Students enrolled in the course can view the assignment. When they complete the assignment, they can go to the Submit page to upload their answers and upload their assignments. The key point here is that they upload their assignments anonymously. Other users can only see the Id information of the assignments, but they cannot see who they belong to, so they can score them fairly. In short, there is an ID-based scoring system. In case the student objects to the grade, the Academician or the assistant can edit the grade.

References

- <https://www.geeksforgeeks.org/what-is-a-distributed-system/>
- <https://spring.io/projects/spring-boot>
- <https://spring.io/projects/spring-security>