МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное   
образовательное учреждение высшего образования  
«Самарский национальный исследовательский университет   
имени академика С.П. Королева»

(Самарский университет)

Институт информатики, математики и электроники

Факультет информатики  
Кафедра суперкомпьютеров и общей информатики

Отчет по лабораторным работам №№ 6

Дисциплина: «Enterprise Systems Development (Методы проектирования и поддержки требований к программному обеспечению)»

Тип предприятия: «University»

Выполнил: Харитонов Н. В.

Группа: 6133-010402D

Самара 2022

**ASSIGNMENT**

Objective: To study the business process implementation with Cuba Platform  
Tasks  
1. Create Cuba Platform project, which will implement one of business processes of previously selected organization.  
2. Add section to doc report image, which should include:  
• Selected business process description (several sentences). It is desirable that the business process be the same as business process described in assignment #4.  
• Instructions for running the application, prerequisites.  
• Step-by-step instruction with demonstration of business process automation using the application.  
3. Send zipped Cuba Project files and doc report to j-avdeev@yandex.ru  
Notice  
Use Cuba Platform documentation (for example https://doc.cuba-platform.com/bpm-latest/bpm.html).  
Yes, can use your preferable framework (not Cuba Platform) to create a web application to automate chosen business process.

**PROGRESS**

In the progress of the laboratory work the Jmix plugin was installed in Intellij IDEA. A new project was created with the "Jmix Project" option in Intellij IDEA. PostgreSQL was used in the project. Figures 1-2 demonstrate the process of creating the project.

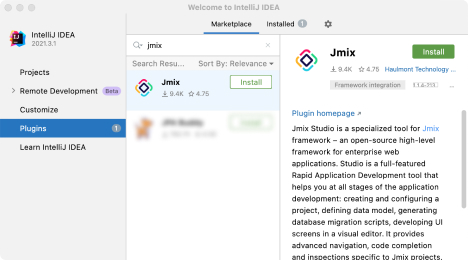


Figure 1 – Installing the plugin

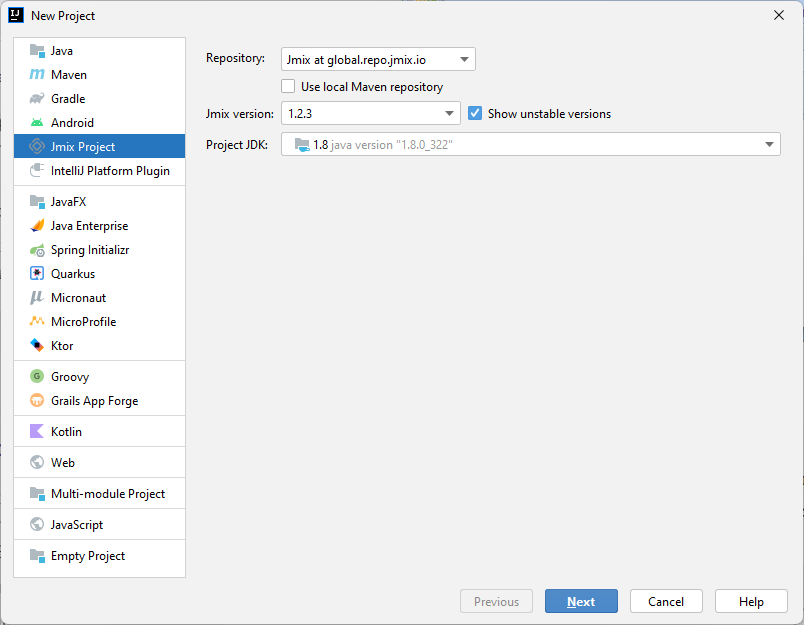


Figure 2 – Creating the project

The selected enterprise type is University. The following entities were created:

* Course – a university course which students can attend;
* Group – a group of students who have the same courses;
* Person – a parent class for the Student and Teacher classes;
* Student – a student of the university;
* Subject – a subject that can be taught in the university;
* Teacher;
* Online meeting – organized by a teacher for specific courses or groups.

An enumeration was also created – StudyYear – which represents ordinal number of the year of a group: 1, 2, 3 and 4.

The figure 3 shows the entity relationship diagram.

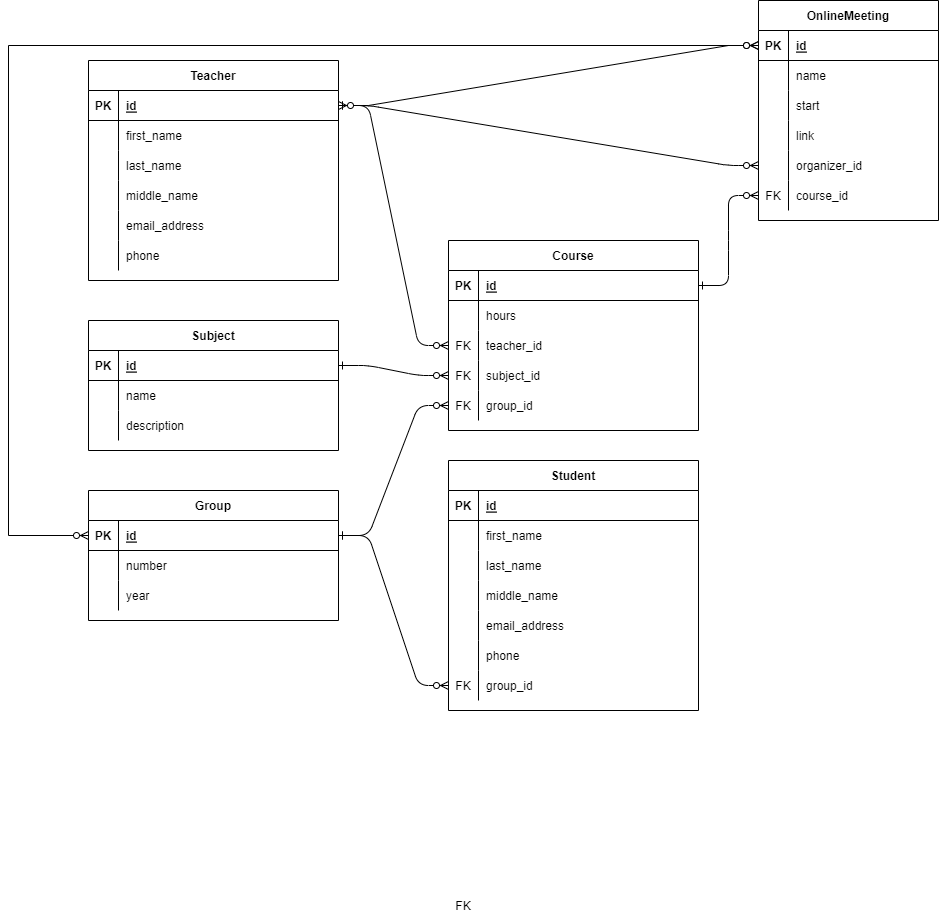


Figure 3 – The entity diagram

The support of the two languages was implemented by setting the “*jmix.core.available-locales*” property as shown on the figure 4.

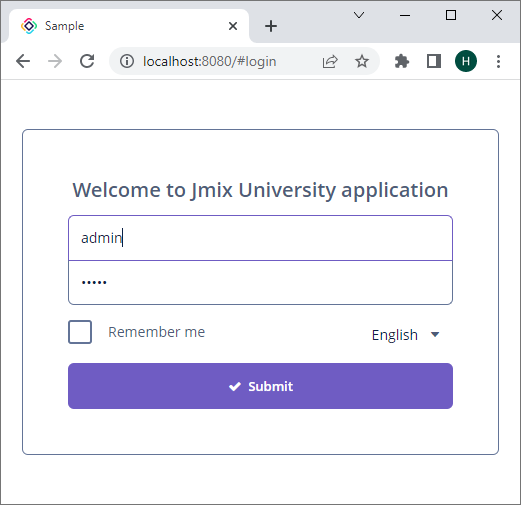
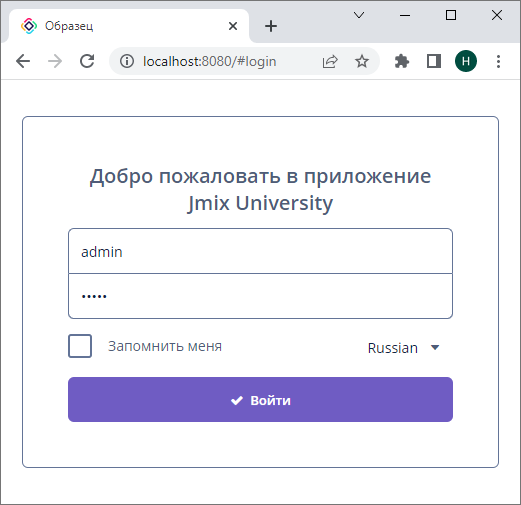
 

Figure 4 – Localization support

Three roles were created in the project:

* Admin – has full access to the site;
* Teacher – can view Courses, Students, Subjects, Teachers and Groups, can view and create Online Meetings;
* Student – can view Courses, Students, Subjects, Teachers, Groups and Online Meetings.

Figure 5 shows how the site is displayed for different roles.

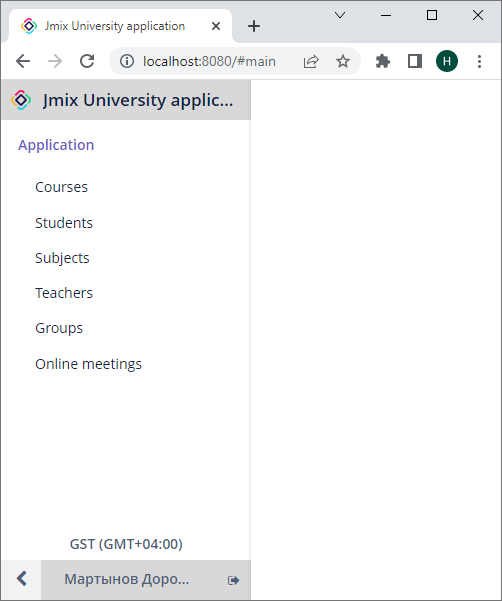
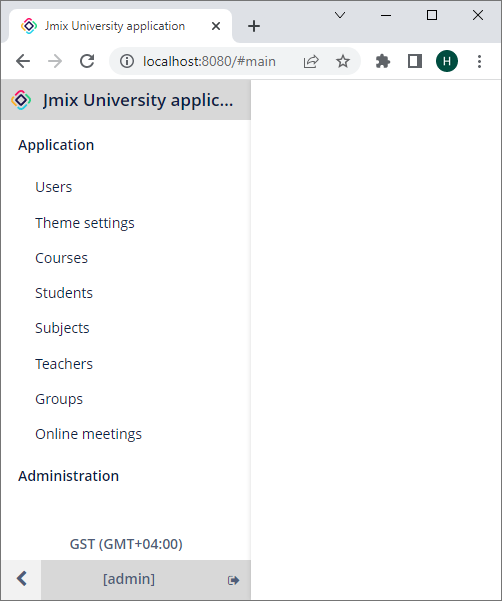


Figure 5 – Displaying the menu for different roles: admin (left), teacher and student (right)

The business process is simple. A user with the Admin role first creates teachers, subjects and groups. Then the user creates courses and assigns a teacher, a subject and a group to each course. Then the user creates students and distributes them into groups. The user can then update the data when needed, for example, when moving students to another group.

Prerequisites:

* JDK 1.8.0;
* Jmix 1.2.3;
* PostgreSQL 9.6.

Figures 6-15 demonstrate how a user with the Admin role can view and create different implemented entities.

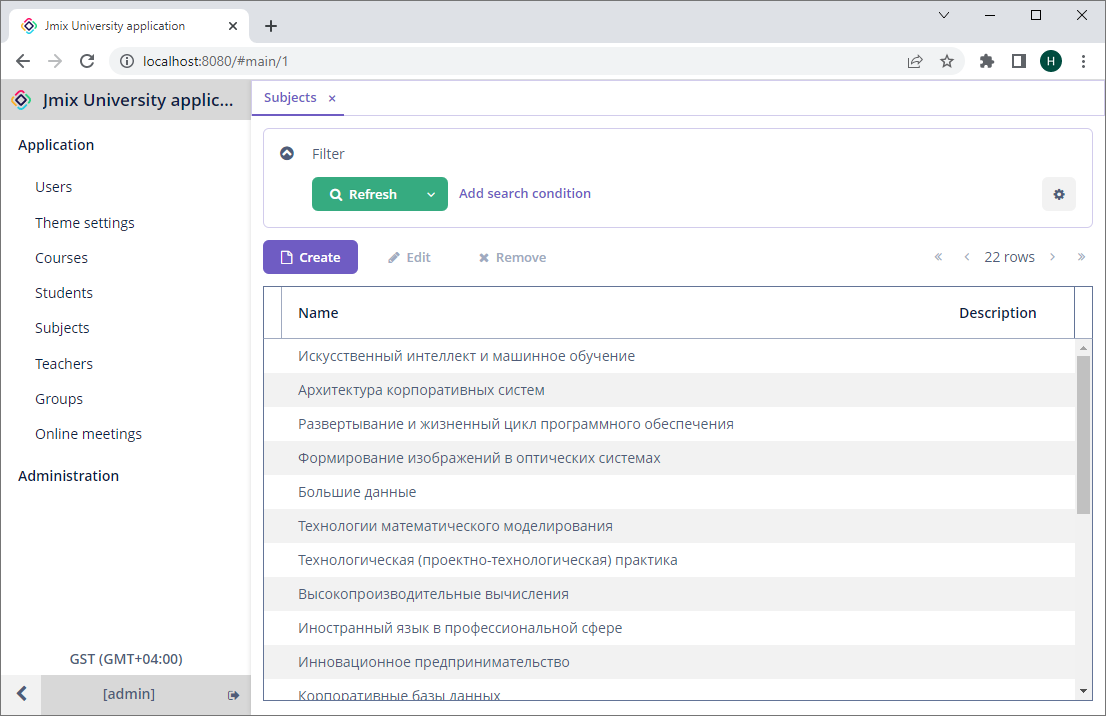


Figure 6 – Viewing subjects

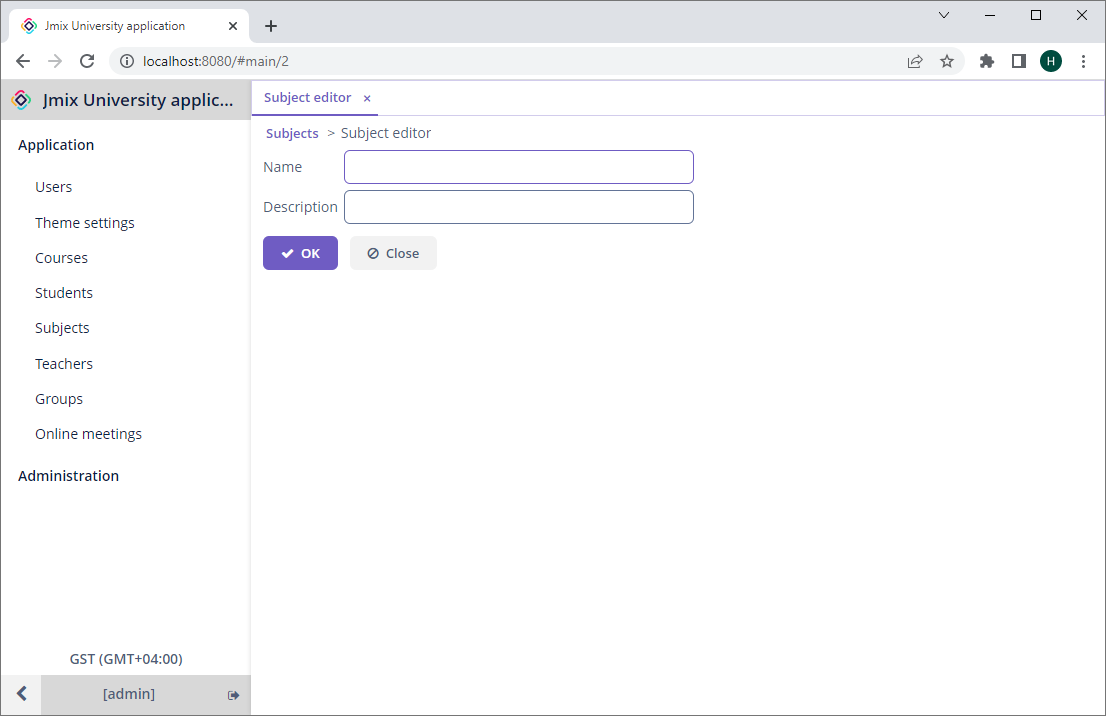


Figure 7 – Creating a subject

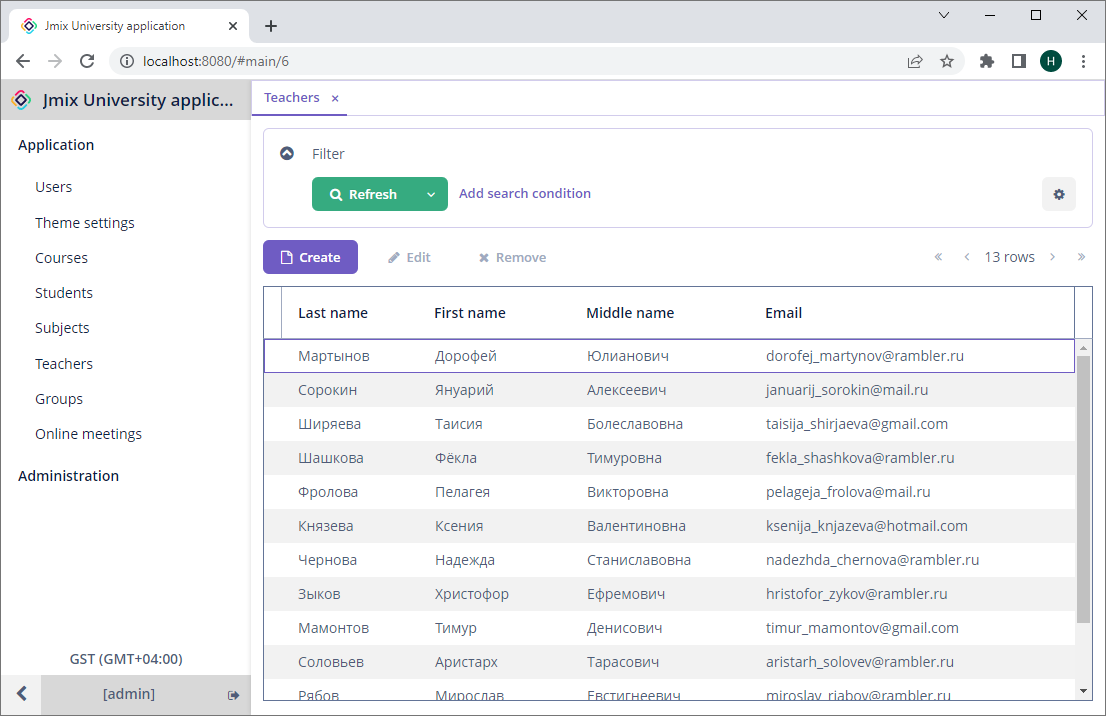


Figure 8 – Viewing teachers

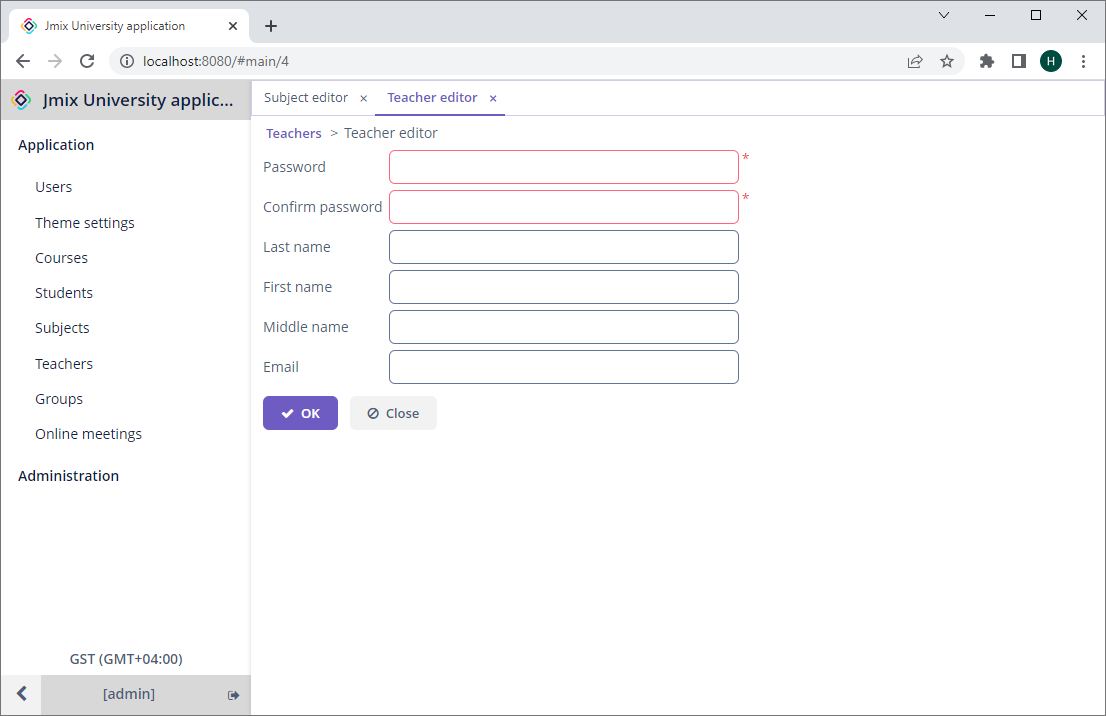


Figure 9 – Creating a teacher

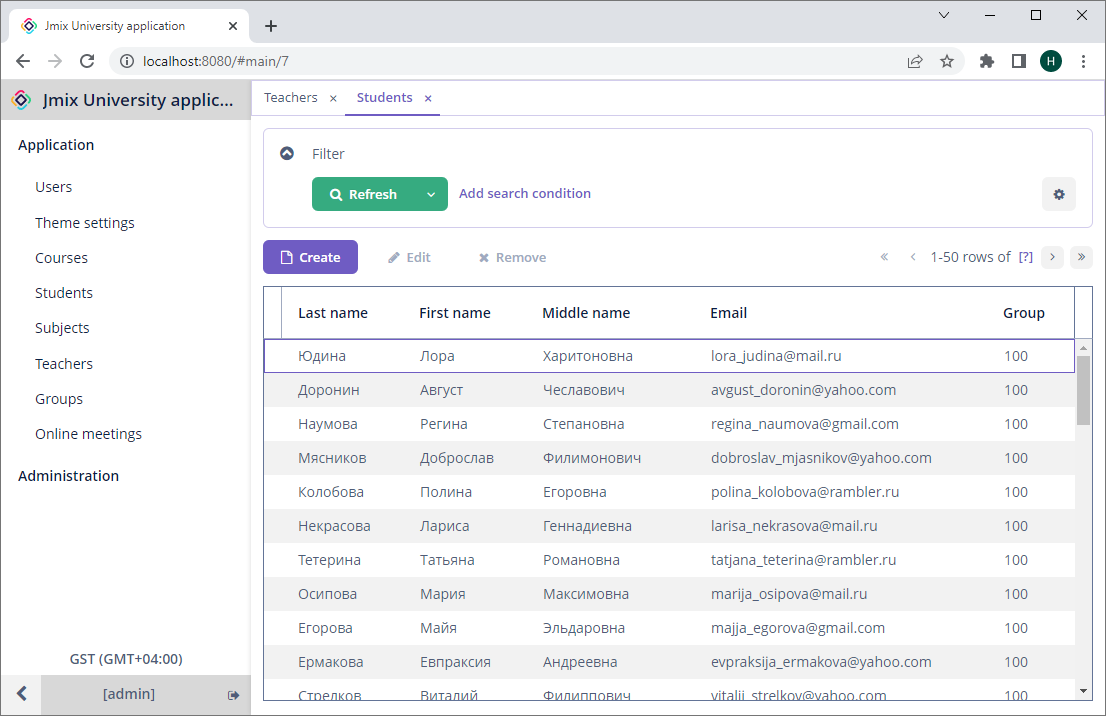


Figure 10 – Viewing students

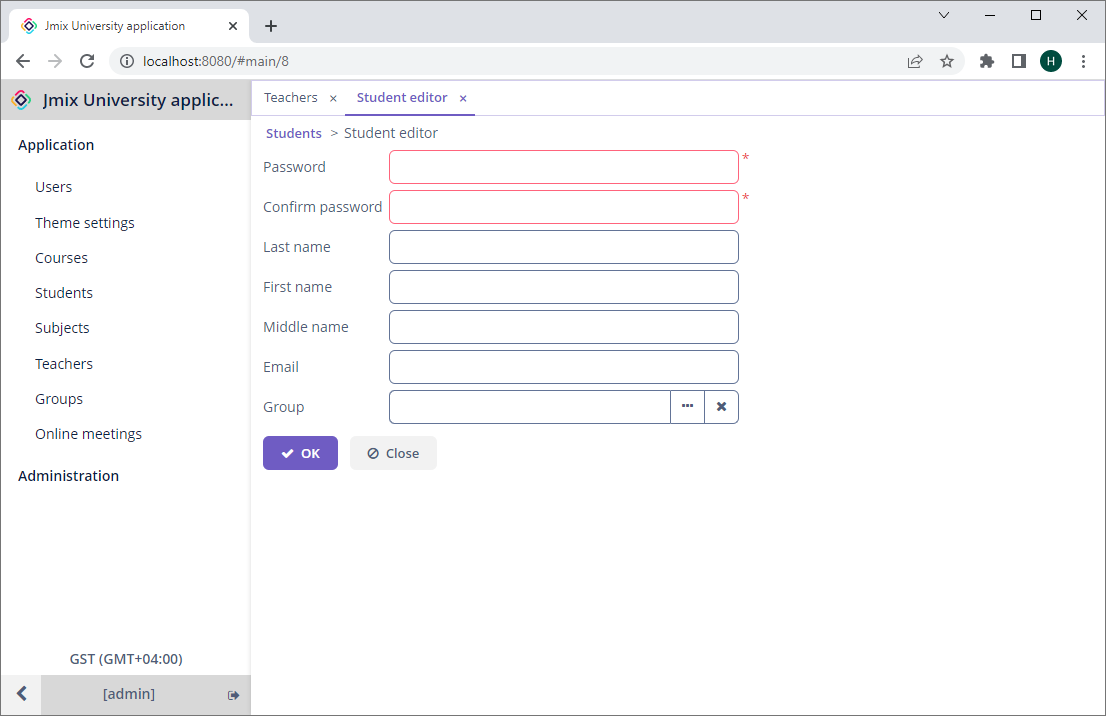


Figure 11 – Creating a student

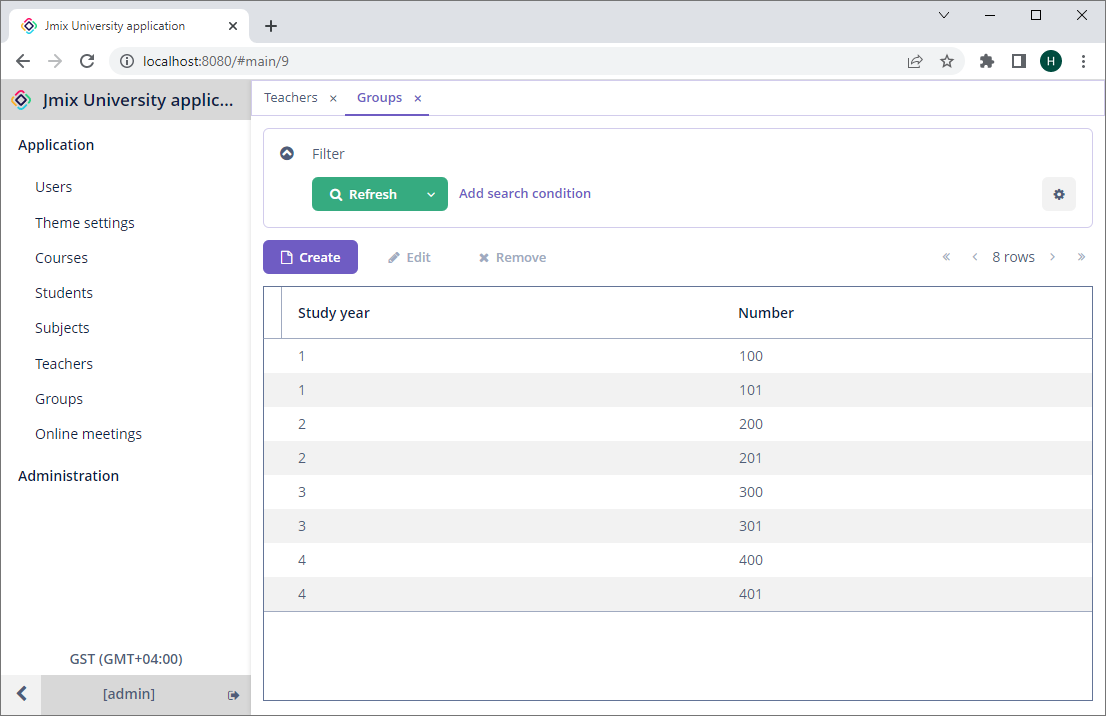


Figure 12 – Viewing groups

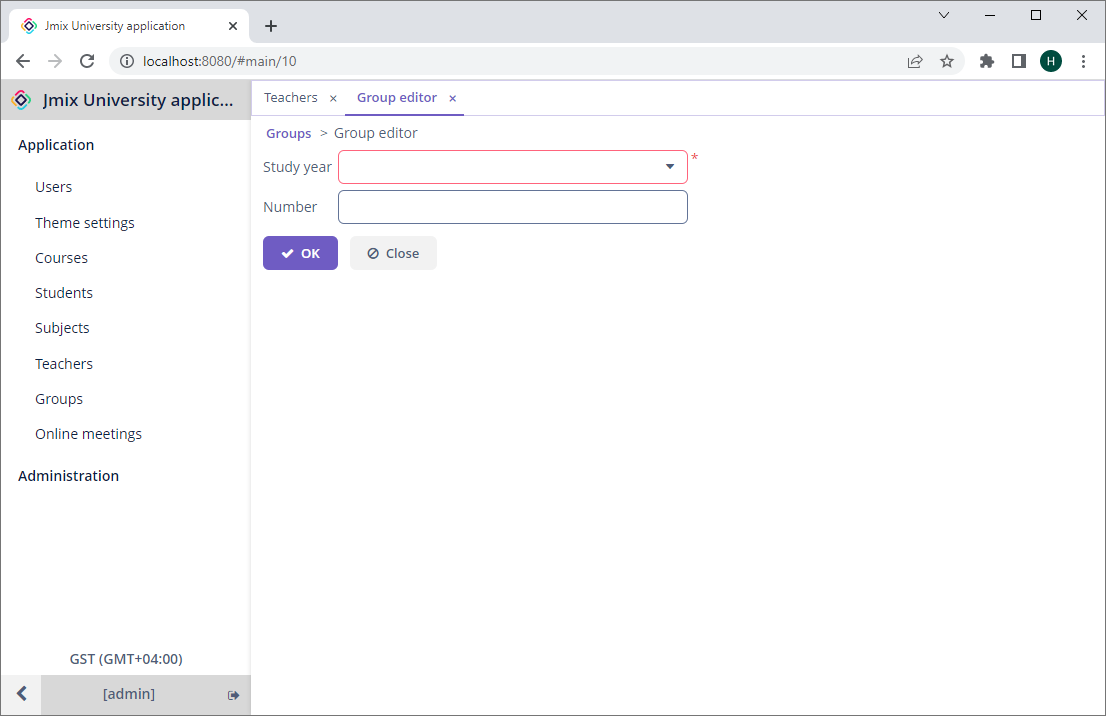


Figure 13 – Creating a group

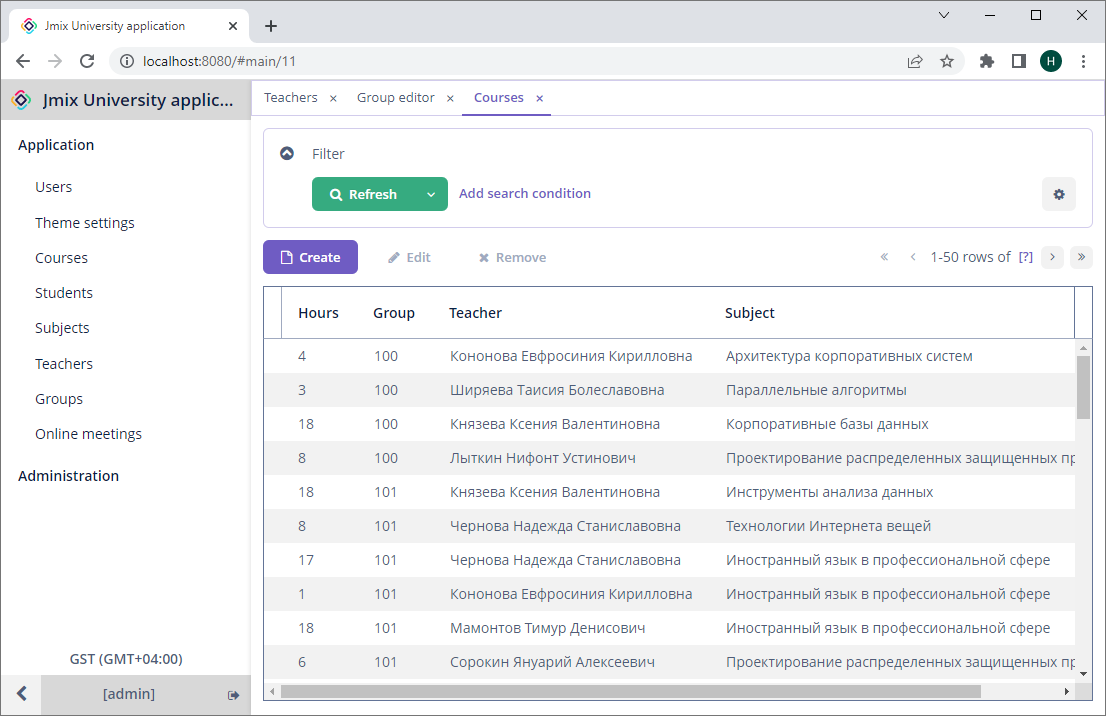


Figure 14 – Viewing courses

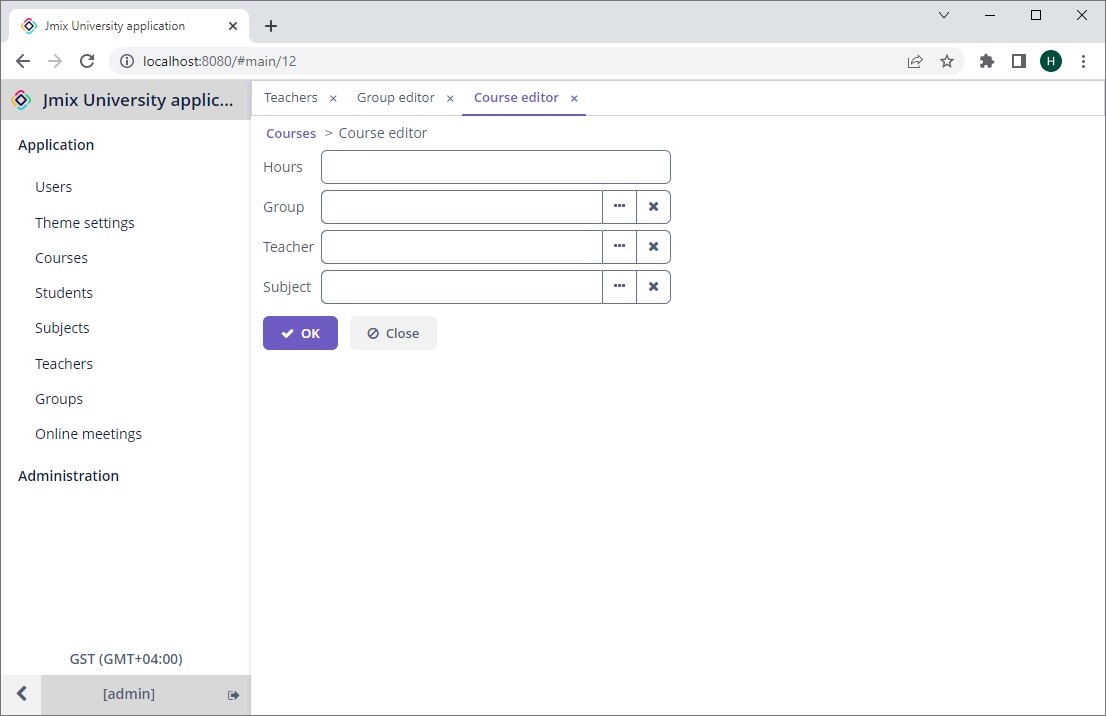


Figure 15 – Creating a course

Consider business logic of online meetings. An online meeting can be only created by a teacher. A student can only view online meetings. A teacher only sees online meetings organized by the teacher or in which the teacher participates, as demonstrated on the figure 16. A student only sees online meetings in which the student’s group participates.

When creating an online meeting the current teacher becomes the organizer of the meeting. The organizer cannot be changed. The organizer can either link the meeting to an existing course, in which case only the students who attend the course can see it, or add multiple groups to the meeting, in which case all the students of the added groups can see it. The organizer can also add other teachers to the meeting.

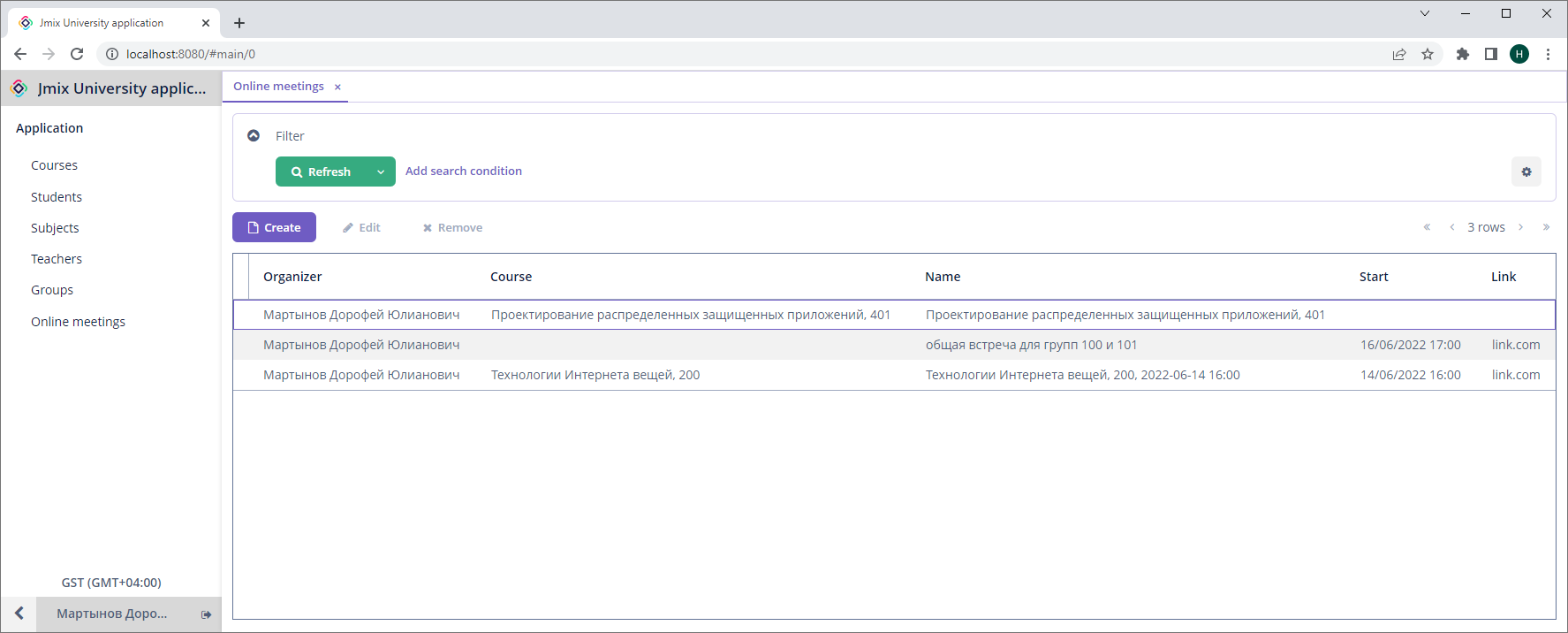


Figure 16 – Viewing online meetings

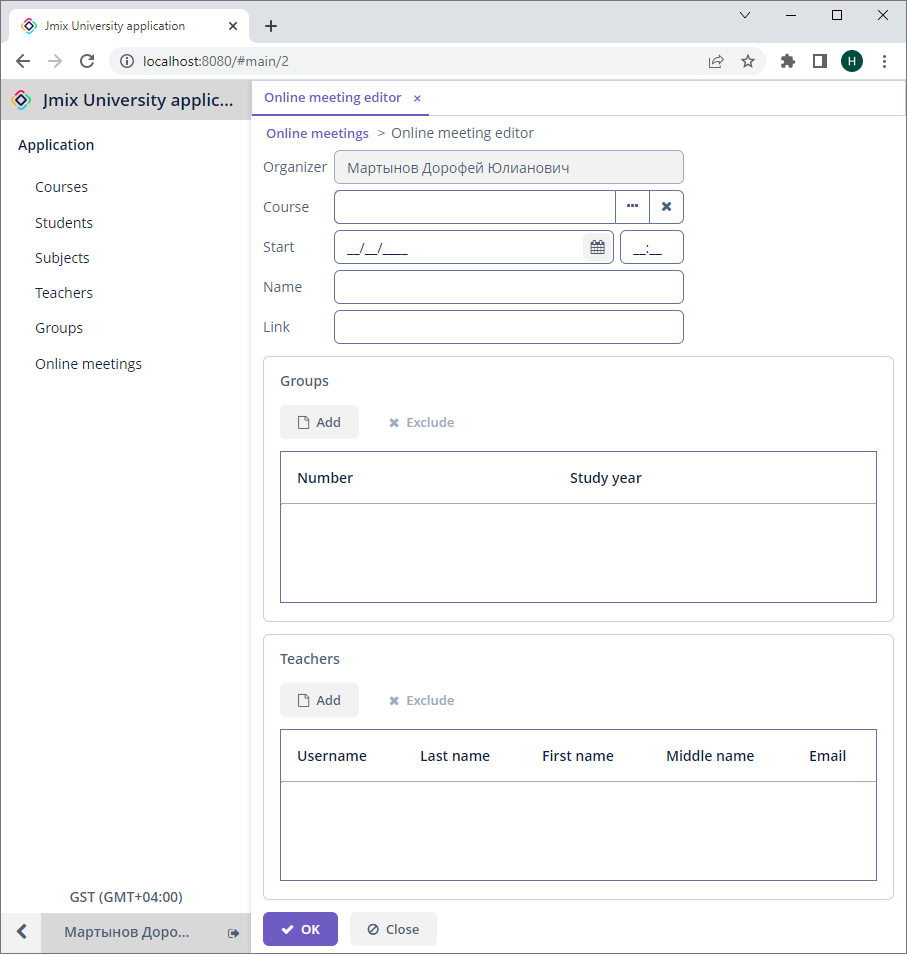


Figure 17 – Creating an online meeting

**CONCLUSION**

As a result, a project was created using the Jmix platform, Java and PostgreSQL. The project is a simple enterprise application for a university. It has multiple entities, screens and roles.