

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НАУЧНО-ОБРАЗОВАТЕЛЬНАЯ КОРПОРАЦИЯ ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ



**Лабораторная работа  
по информационным системам №1**

**Вариант 368949**

**Выполнил:**

Степанов Арсений Алексеевич

**Группа:**

Р3318

**Преподаватель:**

Николаев Владимир Вячеславович

Санкт-Петербург, 2025г.

# Задание

Реализовать информационную систему, которая позволяет взаимодействовать с объектами класса Organization, описание которого приведено ниже:

```
1 public class Organization {
2     // Поле не может быть null
3     // Значение поля должно быть больше 0
4     // Значение этого поля должно быть уникальным
5     // Значение этого поля должно генерироваться автоматически
6     private Long id;
7
8     // Поле не может быть null
9     // Строка не может быть пустой
10    private String name;
11
12    // Поле не может быть null
13    private Coordinates coordinates;
14
15    // Поле не может быть null
16    // Значение этого поля должно генерироваться автоматически
17    private java.util.Date creationDate;
18
19    // Поле может быть null
20    private Address officialAddress;
21
22    // Значение поля должно быть больше 0
23    private float annualTurnover;
24
25    // Значение поля должно быть больше 0
26    private int employeesCount;
27
28    // Поле не может быть null
29    // Значение поля должно быть больше 0
30    private Float rating;
31
32    // Поле может быть null
33    private OrganizationType type;
34
35    // Поле может быть null
36    private Address postalAddress;
37 }
```

```
1 public class Coordinates {
2     // Значение поля должно быть больше -469
3     private float x;
4
5     // Значение поля должно быть больше -440
6     // Поле не может быть null
7     private Long y;
8 }
```

```

1 public class Address {
2     // Поле не может быть null
3     private String street;
4
5     // Длина строки не должна быть больше 24
6     // Поле может быть null
7     private String zipCode;
8
9     // Поле не может быть null
10    private Location town;
11 }

```

```

1 public class Location {
2     private int x;
3
4     private float y;
5
6     // Поле не может быть null
7     private Double z;
8 }

```

```

1 public enum OrganizationType {
2     COMMERCIAL,
3     PUBLIC,
4     GOVERNMENT,
5     TRUST,
6     OPEN_JOINT_STOCK_COMPANY;
7 }

```

Разработанная система должна удовлетворять следующим требованиям:

1. Основное назначение информационной системы - управление объектами, созданными на основе заданного в варианте класса
2. Необходимо, чтобы с помощью системы можно было выполнить следующие операции с объектами: создание нового объекта, получение информации об объекте по ИД, обновление объекта (модификация его атрибутов), удаление объекта. Операции должны осуществляться в отдельных окнах (интерфейсах) приложения. При получении информации об объекте класса должна также выводиться информация о связанных с ним объектах
3. При создании объекта класса необходимо дать пользователю возможность связать новый объект с объектами вспомогательных классов, которые могут быть связаны с созданным объектом и уже есть в системе
4. Выполнение операций по управлению объектами должно осуществляться на серверной части (не на клиенте), изменения должны синхронизироваться с базой данных

5. На главном экране системы должен выводиться список текущих объектов в виде таблицы (каждый атрибут объекта - отдельная колонка в таблице). При отображении таблицы должна использоваться пагинация (если все объекты не помещаются на одном экране)
6. Нужно обеспечить возможность фильтровать/сортировать строки таблицы, которые показывают объекты (по значениям любой из строковых колонок). Фильтрация элементов должна производиться только по полному совпадению
7. Переход к обновлению (модификации) объекта должен быть возможен из таблицы с общим списком объектов и из области с визуализацией объекта (при ее реализации)
8. При добавлении/удалении/изменении объекта, он должен автоматически появиться/исчезнуть/измениться в интерфейсах у других пользователей, авторизованных в системе
9. Если при удалении объекта с ним связан другой объект, связанные объекты должны удаляться
10. Пользователи должны иметь возможность просмотра всех объектов. Для модификации объекта должно открываться отдельное диалоговое окно. При вводе некорректных значений в поля объекта должны появляться информативные сообщения о соответствующих ошибках

В системе должен быть реализован отдельный пользовательский интерфейс для выполнения специальных операций над объектами:

1. Вернуть массив объектов, значение поля name которых начинается с заданной подстроки
2. Вернуть массив объектов, значение поля rating которых меньше заданного
3. Вернуть массив объектов, значение поля rating которых больше заданного
4. Объединить организации, создав новую и зачислив в неё всех сотрудников двух исходных. Пользователь должен указать исходные организации, имя и адрес новой организации
5. Реализовать поглощение одной организацией другой без увольнения сотрудников

Представленные операции должны быть реализованы в рамках компонентов бизнес-логики приложения без прямого использования функций и процедур БД

Особенности хранения объектов, которые должны быть реализованы в системе:

1. Организовать хранение данных об объектах в реляционной СУБД (PostgreSQL). Каждый объект, с которым работает ИС, должен быть сохранен в базе данных

2. Все требования к полям класса (указанные в виде комментариев к описанию классов) должны быть выполнены на уровне ORM и БД
3. Для генерации поля id использовать средства базы данных
4. Для подключения к БД на кафедральном сервере использовать хост pg, имя базы данных - studs, имя пользователя/пароль совпадают с таковыми для подключения к серверу

При создании системы нужно учитывать следующие особенности организации взаимодействия с пользователем:

1. Система должна реагировать на некорректный пользовательский ввод, ограничивая ввод недопустимых значений и информируя пользователей о причине ошибки
2. Переходы между различными логически обособленными частями системы должны осуществляться с помощью меню
3. При добавлении/удалении/изменении объекта, он должен автоматически появиться/исчезнуть/измениться на области у всех других клиентов

При разработке ИС должны учитываться следующие требования:

1. В качестве основы для реализации ИС необходимо использовать Spring MVC
2. Для создания уровня хранения необходимо использовать Hibernate
3. Разные уровни приложения должны быть отделены друг от друга, разные логические части ИС должны находиться в отдельных компонентах

## Выполнение

Для реализации фронтенд-составляющей информационной системы был использован фреймворк Svelte Kit и библиотека компонентов ShadCN, система сборки Vite. Для бэкенд-составляющей был использован фреймворк Spring MVC с Hibernate в качестве ORM

Для локального развёртывания приложения каждый отдельный сервис был обернут в Docker-контейнер с последующим использованием в системе развёртывания контейнеров Docker-Compose

Исходный код содержится в репозитории на GitHub:

1. Фронтенд – Ссылка на код в репозитории
2. Бэкенд – Ссылка на код в репозитории
3. Диаграмма классов – Ссылка на изображение

# Скриншоты пользовательского интерфейса

Вход

Введите свои данные чтобы продолжить

Имя пользователя

armemius

Пароль

\*\*\*\*\*

Повтор пароля

\*\*\*\*\*

Регистрация

Есть аккаунт? Войти

Лабораторная #1

Предметная область

Организации

Координаты

Локации

Адреса

Управление

Операции

Исходники

CN armemius  
Пользователь

Главная > Операции

Префикс Рейтинг+ Рейтинг- Поиск по префиксу

Слияние организаций Поглощение организации

ID	Имя	Координаты	Дата создания	Официальный адрес	Выручка за год	Сотрудники	Рейтинг	Тип организации	Адрес для корреспонденции
----	-----	------------	---------------	-------------------	----------------	------------	---------	-----------------	---------------------------

Слияние организаций

Объединить две организации, создав новую с указанным именем и адресом

Имя\*

Адрес\*

Выбрать

+

Орг. #1\*

Выбрать

Орг. #2\*

Выбрать

Выполнить операцию

Лабораторная #1

Предметная область

Организации

Координаты

Локации

Адреса

Управление

Операции

Исходники

CN armemius  
Пользователь

Главная > Предметная область > Координаты

ID	X1	Y
71	-468	-439
69	65	789
45	869.5	654654645
61	2343	23423432432432
64	2343	23423432432432
65	2343	23423432432432
47	2343	23423432432432
67	234324242000000000	23423432432432
66	234324242000000000	23423432432432
62	234324242000000000	23423432432432

+ Добавить новый элемент

Строк на странице 10

Страница 1 из 2

## Выводы

В ходе работы была разработана и протестирована информационная система на основе заданного техзадания. Были углублены знания касательно отдельных аспектов работы экосистемы Spring и Svelte Kit. Изучены новые инструменты для работы с фронтендом, бэкендом, базами данных и инфраструктурой