**Университет ИТМО**

**Факультет программной инженерии и компьютерных технологий**

**Лабораторная работа №4 по Web-Программированию**

**Выполнил: Богатов Александр Сергеевич**

**Группа: P3233**

**Вариант:** **3367**

**Преподаватель: Горбунов Михаил Витальевич**

**Санкт-Петербург**

**2021**

**Задание:**

Переписать приложение из [предыдущей лабораторной работы](https://se.ifmo.ru/courses/web#lab3) с использованием следующих технологий:

* Уровень back-end должен быть основан на ~~Java EE (необходимо использовать EJB)~~ Spring.
* Уровень front-end должен быть построен на [Angular 2+](https://angular.io/" \t "_blank) с использованием обычных полей ввода HTML
* Взаимодействие между уровнями back-end и front-end должно быть организовано посредством REST API.

Приложение по-прежнему должно включать в себя 2 страницы - стартовую и основную страницу приложения. Обе страницы приложения должны быть адаптированы для отображения в 3 режимах:

* "Десктопный" - для устройств, ширина экрана которых равна или превышает 1239 пикселей.
* "Планшетный" - для устройств, ширина экрана которых равна или превышает 866, но меньше 1239 пикселей.
* "Мобильный"- для устройств, ширина экрана которых меньше 866 пикселей.

**Стартовая страница должна содержать следующие элементы:**

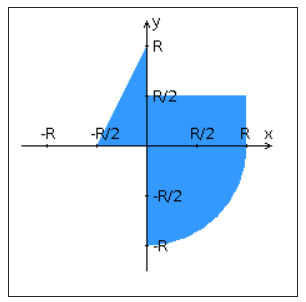
* "Шапку", содержащую ФИО студента, номер группы и номер варианта.
* Форму для ввода логина и пароля. Информация о зарегистрированных в системе пользователях должна храниться в отдельной таблице БД (пароль должен храниться в виде хэш-суммы). Доступ неавторизованных пользователей к основной странице приложения должен быть запрещён.

**Основная страница приложения должна содержать следующие элементы:**

* Набор полей ввода для задания координат точки и радиуса области в соответствии с вариантом задания: Select {'-4','-3','-2','-1','0','1','2','3','4'} для координаты по оси X, Text (-5 ... 3) для координаты по оси Y, и Select {'-4','-3','-2','-1','0','1','2','3','4'} для задания радиуса области. Если поле ввода допускает ввод заведомо некорректных данных (таких, например, как буквы в координатах точки или отрицательный радиус), то приложение должно осуществлять их валидацию.
* Динамически обновляемую картинку, изображающую область на координатной плоскости в соответствии с номером варианта и точки, координаты которых были заданы пользователем. Клик по картинке должен инициировать сценарий, осуществляющий определение координат новой точки и отправку их на сервер для проверки её попадания в область. Цвет точек должен зависеть от факта попадания / непопадания в область. Смена радиуса также должна инициировать перерисовку картинки.
* Таблицу со списком результатов предыдущих проверок.
* Кнопку, по которой аутентифицированный пользователь может закрыть свою сессию и вернуться на стартовую страницу приложения.

**Дополнительные требования к приложению:**

* Все результаты проверки должны сохраняться в базе данных под управлением СУБД Oracle.
* Для доступа к БД необходимо использовать ~~JPA~~ Spring Data.



**Исходный код:**

https://github.com/albogatov/web-lab4

**Вывод:**

В процессе выполнения данной лабораторной работы, были сделаны следующие выводы: frontend – вещь дьявольская, Angular – вещь немного нестабильная, Spring – вещь хорошая. Берегите себя и своих близких.