**ФПИиКТ**

**Отчет по лабораторной работе №4**

**По веб-программированию**

**Вариант №77626**

Ибадуллаев Алибаба Эльбрус оглы

Группа: P3215

Преподаватель: Иван Каюков

Санкт-Петербург

2021г.

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).
* Уровень front-end должен быть построен на [Angular 2+](https://angular.io/) с использованием набора компонентов [PrimeNG](https://www.primefaces.org/primeng/#/)
* 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 1263 пикселей.
* "Планшетный" - для устройств, ширина экрана которых равна или превышает 831, но меньше 1263 пикселей.
* "Мобильный"- для устройств, ширина экрана которых меньше 831 пикселей.

**Стартовая страница должна содержать следующие элементы:**

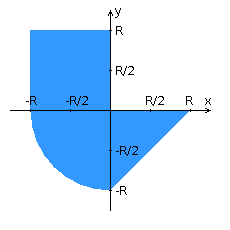
* "Шапку", содержащую ФИО студента, номер группы и номер варианта.
* Форму для ввода логина и пароля. Информация о зарегистрированных в системе пользователях должна храниться в отдельной таблице БД (пароль должен храниться в виде хэш-суммы). Доступ неавторизованных пользователей к основной странице приложения 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: RadioButton {'-3','-2','-1','0','1','2','3','4','5'} для координаты по оси X, Slider (-3 ... 5) для координаты по оси Y, и RadioButton {'-3','-2','-1','0','1','2','3','4','5'} 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 зависить от факта попадания / непопадания в область. Смена радиуса также должна инициировать перерисовку картинки.
* Таблицу со списком результатов предыдущих проверок.
* Ссылку, 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 PostgreSQL.
* Для доступа к БД необходимо использовать JPA.



Исходный код:

<https://github.com/AliIbadullayev/web>4

Вывод:

Я приобрел знания в сфере веб разработки на Java EE и Angular 2. Научился конфигурировать серверную часть приложения на Wildfly. Приобрел знания в сфере создания restful веб приложения.