

Лабораторная работа №3

по курсу «*Веб-программирование*»

Вариант 3309

Работу выполнил:

Глушков Даниил Григорьевич

Группа Р3233

Преподаватель:

Горбунов М. В.

1 Задание

Переписать приложение из предыдущей лабораторной работы с использованием следующих технологий:

- Уровень back-end должен быть основан на Java EE (необходимо использовать EJB).
- Уровень front-end должен быть построен на Vue.js 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 1235 пикселей.
- «Планшетный» – для устройств, ширина экрана которых равна или превышает 699, но меньше 1235 пикселей.
- «Мобильный» – для устройств, ширина экрана которых меньше 699 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:

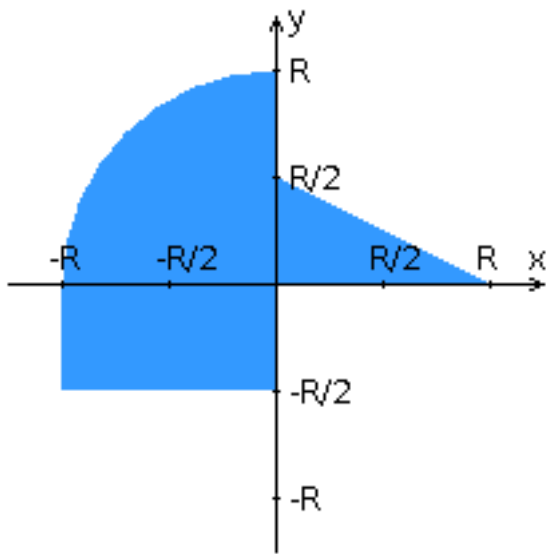
- Набор полей ввода для задания координат точки и радиуса области в соответствии с вариантом задания: Radio { '-4', '-3', '-2', '-1', '0', '1', '2', '3', '4' } для координаты по оси X, Text (-5 ... 3) для координаты по оси Y, и Radio { '-4', '-3', '-2', '-1', '0', '1', '2', '3', '4' } для задания радиуса 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.
- 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.

2 График



3 Исходный код

Исходный код располагается по адресу:

https://github.com/mrcat-pixel/web_lab4.git

4 Вывод

В процессе выполнения лабораторной работы я освоил азы Vue.js и использовал REST API.