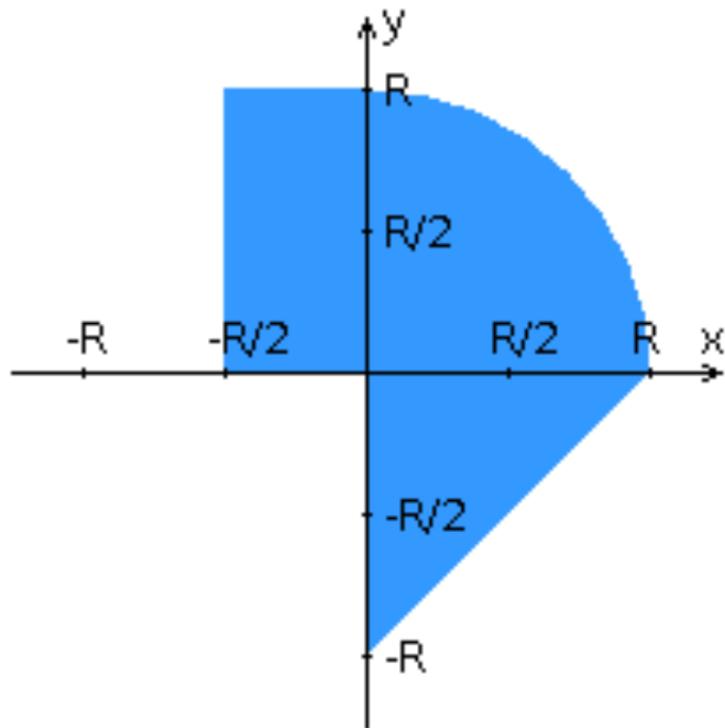


Отчет
по лабораторной работе №4
Веб-программирование
Вариант № 6

Группа
Выполнили:
Преподаватель

P3209
Исаева Александра-Ирина Антоновна
Барсуков Максим Андреевич



Задание

Переписать приложение из предыдущей лабораторной работы с использованием следующих технологий:

- Уровень back-end должен быть основан на Spring.
- Уровень front-end должен быть построен на React + Redux (необходимо использовать ES6 и JSX) 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 1211 пикселей.
- "Планшетный" для устройств, ширина экрана которых равна или превышает 734, но меньше 1211 пикселей.
- "Мобильный" для устройств, ширина экрана которых меньше 734 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: Button '-3','-2','-1','0','1','2','3','4','5' для координаты по оси X, Text (-3 ... 3) для координаты по оси Y, и Button '-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 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 HSQLDB.
- Для доступа к БД необходимо использовать Spring Data.

Репозиторий

https://github.com/isaxel/web/tree/main/my_app4

Выводы

В ходе выполнения лабораторной работы были освоены навыки разработки полноценного веб-приложения с использованием Spring Boot для создания REST API и React для построения пользовательского интерфейса. На практике реализованы аутентификация на основе JWT, валидация входных данных, пагинация и работа с базой данных.