

Факультет программной инженерии и компьютерной техники

Дисциплина: Веб-программирование

Лабораторная работа №4

Вариант № 8388

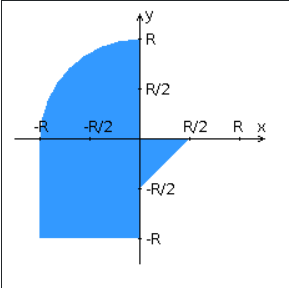
Выполнил: Васильев А. Ю.

№ группы: P3215

Преподаватель: Каюков И. А.

2021 год, 3 семестр

Текст задания.



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

* Уровень back-end должен быть основан на Spring.
* Уровень front-end должен быть построен на [React](https://facebook.github.io/react/" \t "_blank) + [Redux](http://redux.js.org/docs/basics/UsageWithReact.html" \t "_blank) (необходимо использовать ES6 и JSX) с использованием набора компонентов [Belle](http://nikgraf.github.io/belle/" \t "_blank)
* 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 1024 пикселей.
* "Планшетный" - для устройств, ширина экрана которых равна или превышает 793, но меньше 1024 пикселей.
* "Мобильный"- для устройств, ширина экрана которых меньше 793 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: ComboBox {'-3','-2','-1','0','1','2','3','4','5'} для координаты по оси X, TextInput (-3 ... 3) для координаты по оси Y, и ComboBox {'-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 Oracle.
* Для доступа к БД необходимо использовать Spring Data.

Исходный код.

<https://github.com/wizarsi/Spring-React-web-application>

Выводы по работе.

В ходе выполнения лабораторной работы, я научился создавать веб-приложения с использованием библиотеки React и фреймворка Spring. На практике была реализована правильная архитектура React приложения, функциональные компоненты, контейнерные, Business Logic Layer, Data server layer. Создал с помощью Spring Rest API приложение.