

Санкт-Петербургский Национальный Исследовательский
Университет ИТМО
Мегафакультет Компьютерных Технологий и Управления
Факультет Программной Инженерии и Компьютерной Техники



ITMO UNIVERSITY

Вариант №Р3210173
Лабораторная работа №4
по дисциплине
Веб-программирование

Выполнил Студент группы Р32101
Лапин Алексей Александрович
Преподаватель:
Пашнин Александр Денисович

г. Санкт-Петербург
2022г.

Содержание

1	Текст задания:	3
2	Код	4
3	Вывод программы	6
4	Вывод:	8

1 Текст задания:

Лабораторная работа #4

Вариант 32101

Внимание! У разных вариантов разный текст задания!

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 набора компонентов [React Toolbox](#)
- 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 1074 пикселей.
- "Планшетный" - для устройств, ширина экрана которых равна или превышает 702, но меньше 1074 пикселей.
- "Мобильный" - для устройств, ширина экрана которых меньше 702 пикселей.

Стартовая страница должна содержать следующие элементы:

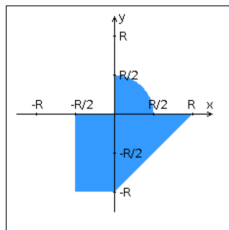
- "Шапку", содержащую ФИО студента, номер группы и номер варианта.
- Форму для ввода логина и пароля. Информация о зарегистрированных в системе пользователях должна храниться в отдельной таблице БД (пароль должен храниться в виде хэш-суммы). Доступ неавторизованных пользователей к основной странице приложения 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: Input (-5 ... 3) для координаты по оси X, Input (-3 ... 3) для координаты по оси Y, и Input (-5 ... 3) 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 Oracle.
- Для доступа к БД необходимо использовать Spring Data.

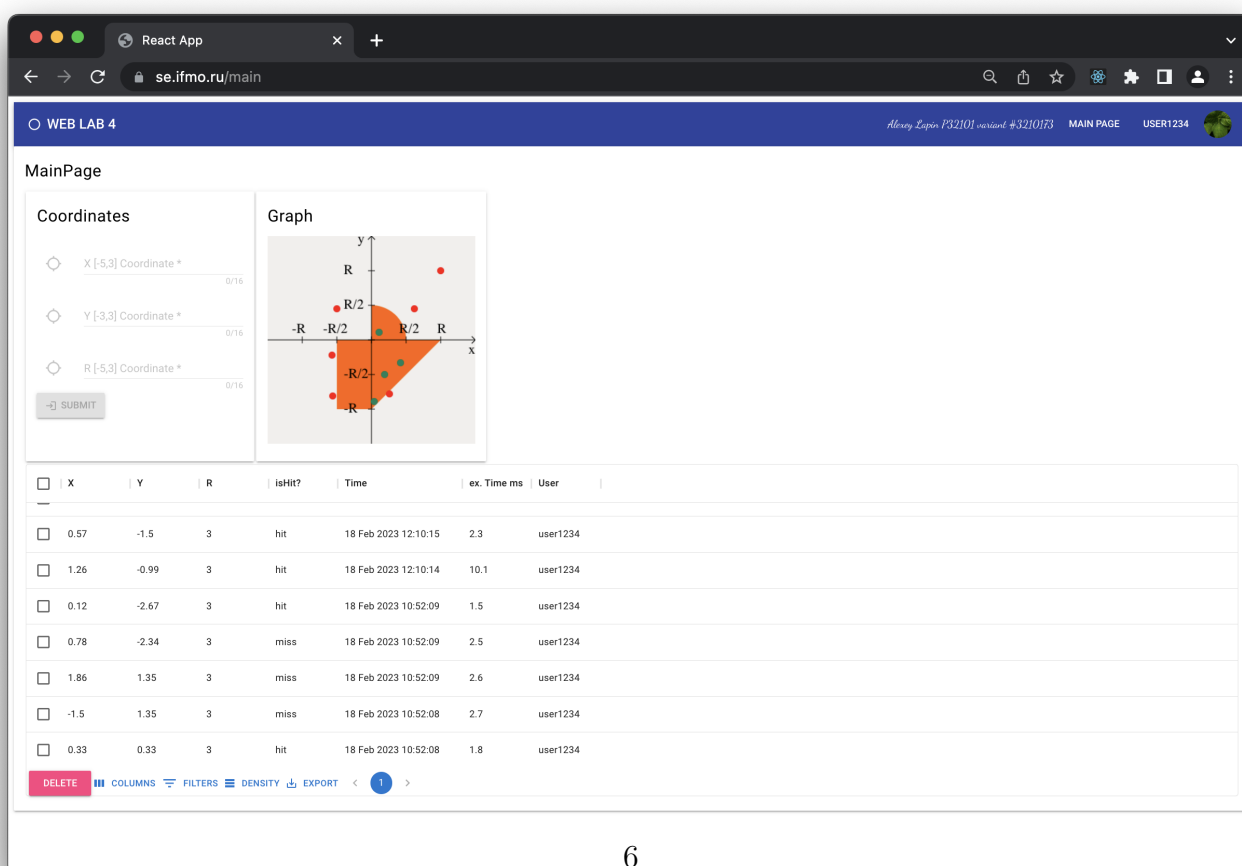
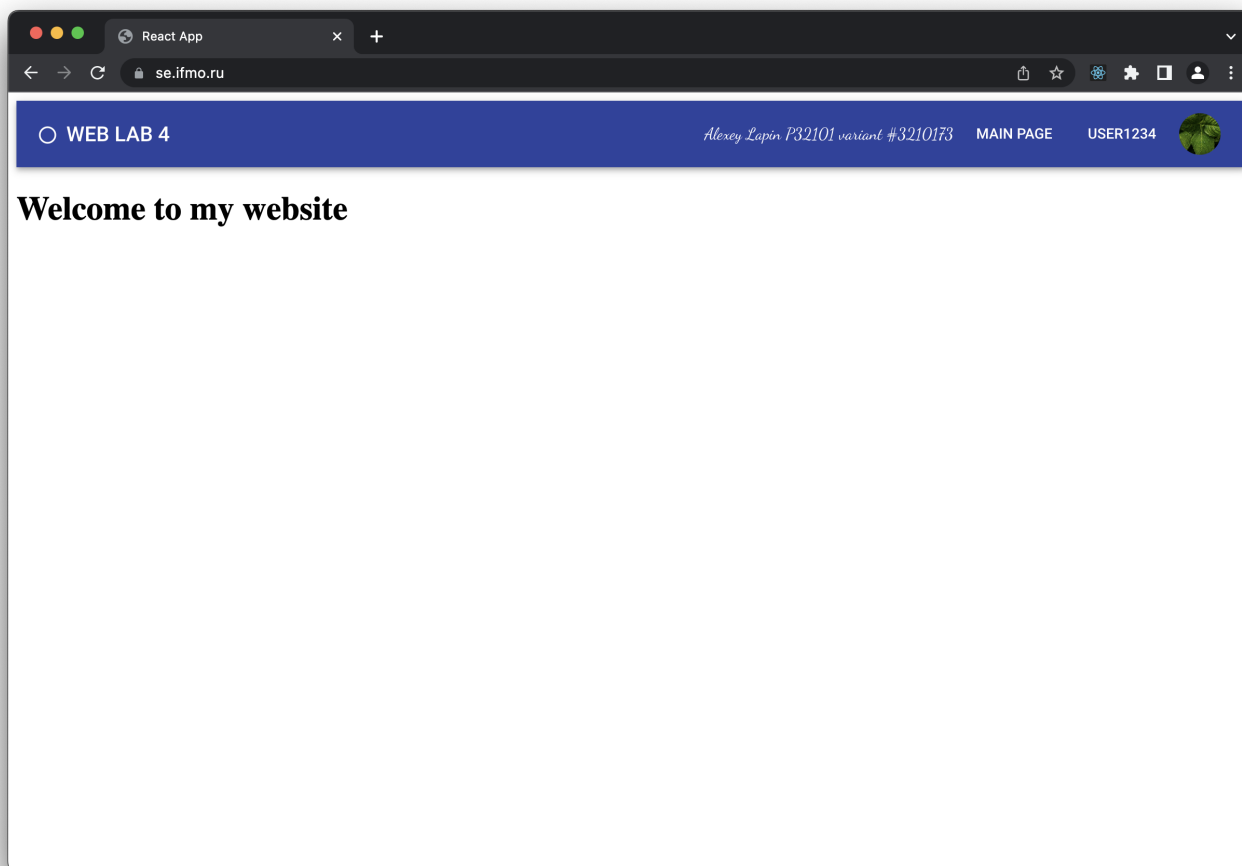


2 Код

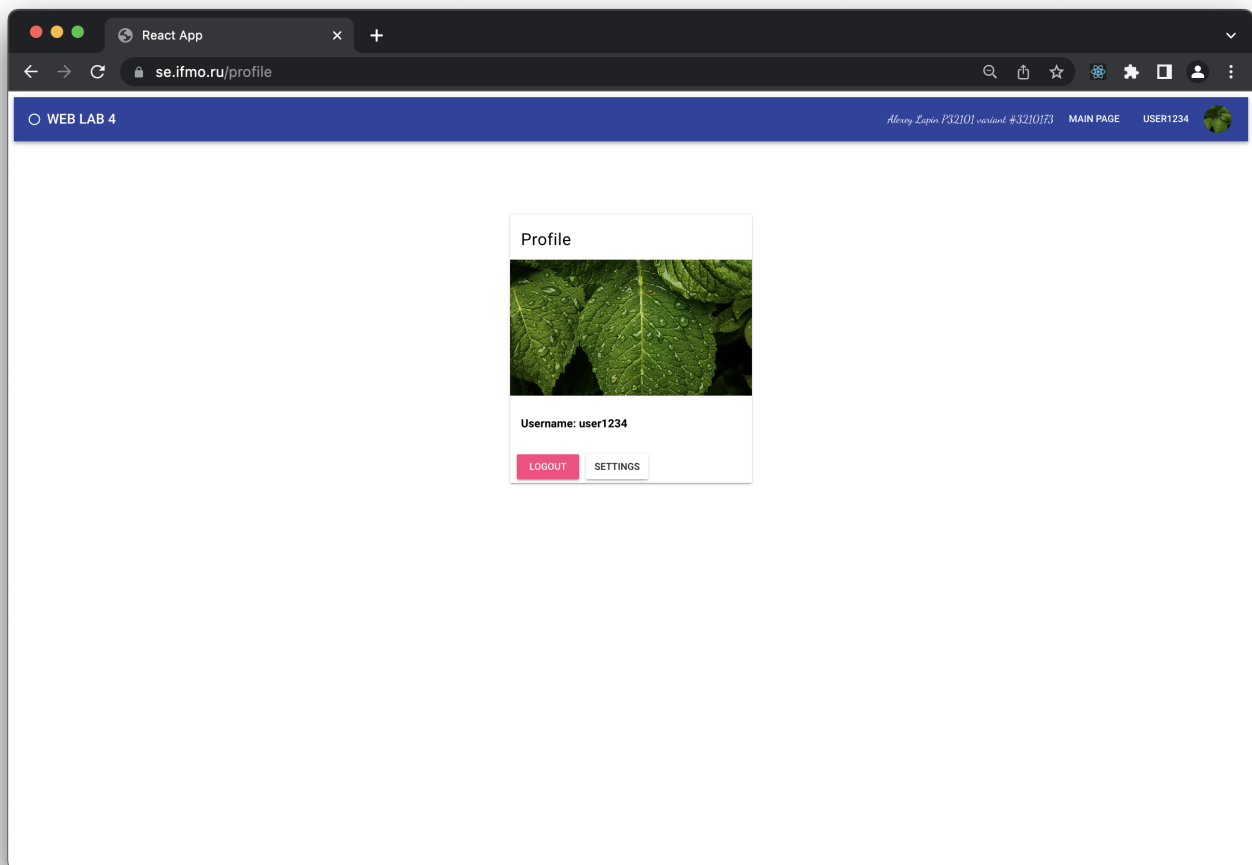
frontend: <https://github.com/AaLexUser/web-4-frontend>

backend: <https://github.com/AaLexUser/web-4>

3 Вывод программы







4 Вывод:

В этой лабораторной работе я познакомился с:

1. React
2. Redux
3. TypeScript
4. Java Spring