**Лабораторная работа №6 Сиразиева Яна Пи21-3**

***Автоматизация тестирования***

**Цель данной лабораторной работы** — разработка автотестов для проверки функциональности программной системы, включающей несколько React-компонентов. Автотесты должны обеспечить сопровождемость, надежность и структурированность системы, а также эмулировать поведение пользователя при взаимодействии с элементами пользовательского интерфейса.

**Описание реализованных автотестов:**

* Инструменты

Jest: для выполнения тестов.

React Testing Library: для эмуляции пользовательских взаимодействий и проверки рендеринга компонентов.

* Подходы

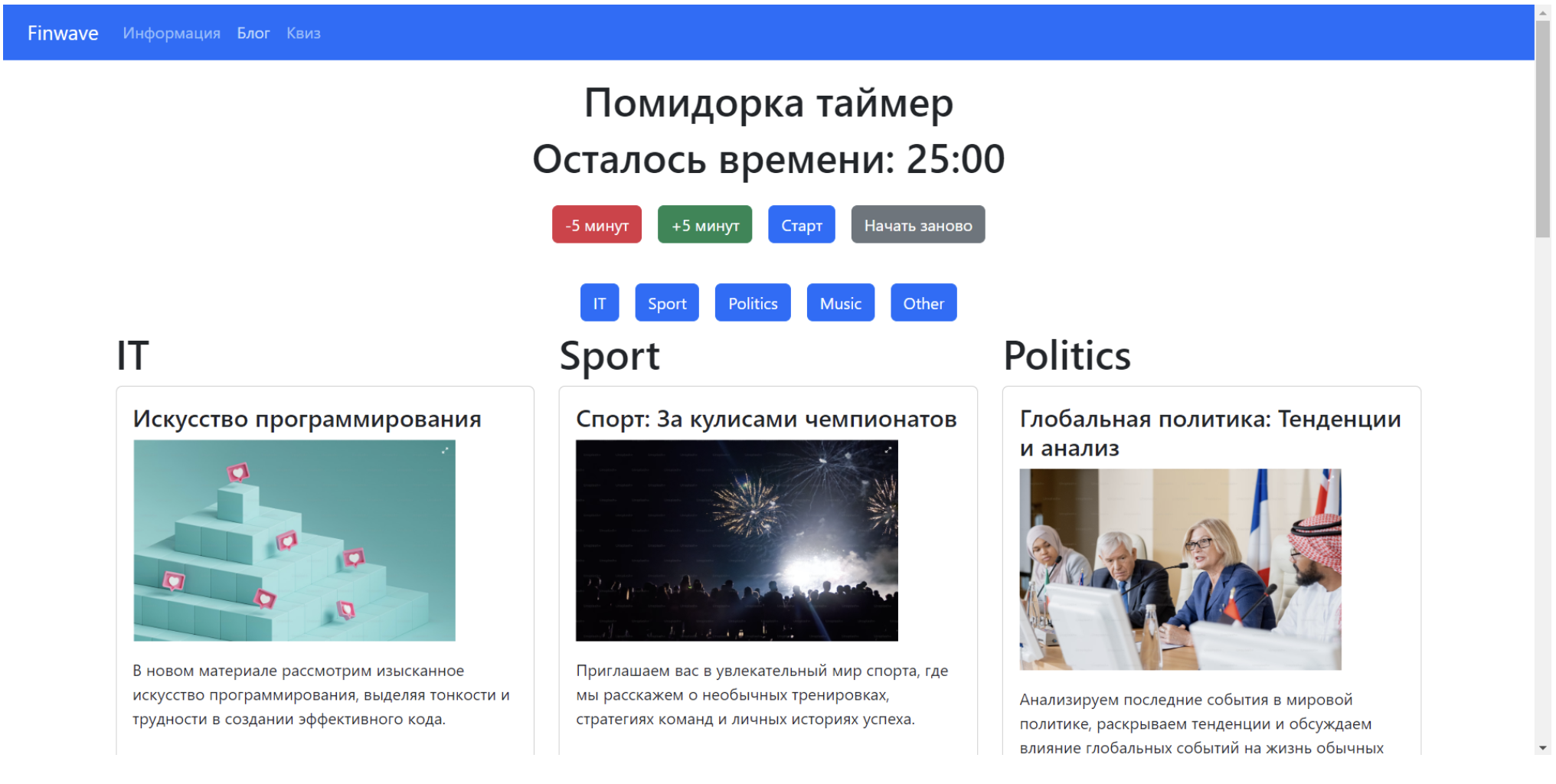
*- Модульное тестирование*: создание тестов для отдельных компонентов.

*- Интеграционное тестирование*: проверка взаимодействия нескольких компонентов.

*- Свойства сопровождаемости:* тесты структурированы и описаны таким образом, чтобы их можно было легко поддерживать и обновлять.

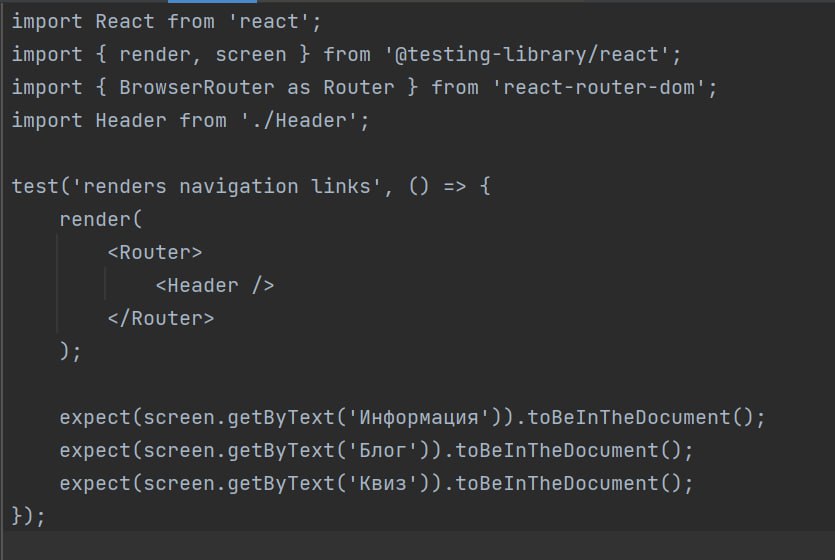
*- Надежность*: тесты покрывают ключевые функциональные элементы, чтобы минимизировать возможность упущения дефектов.

**Визуал проекта:**

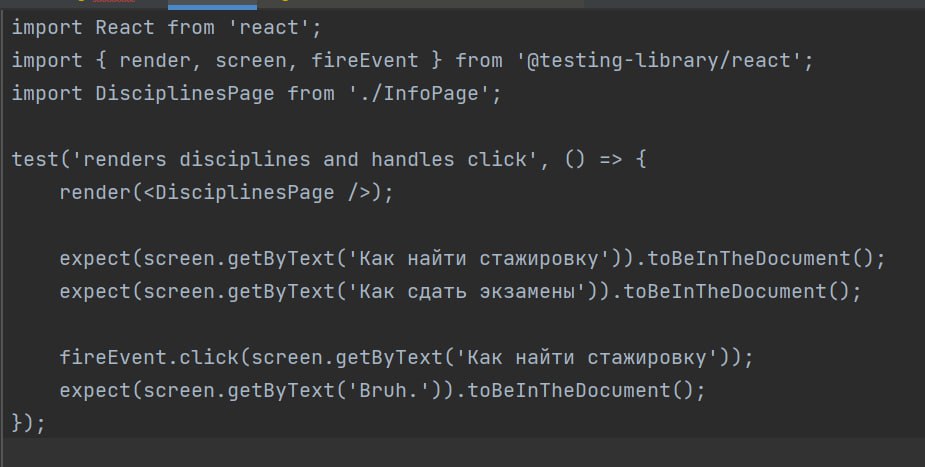


**Код автотестов**

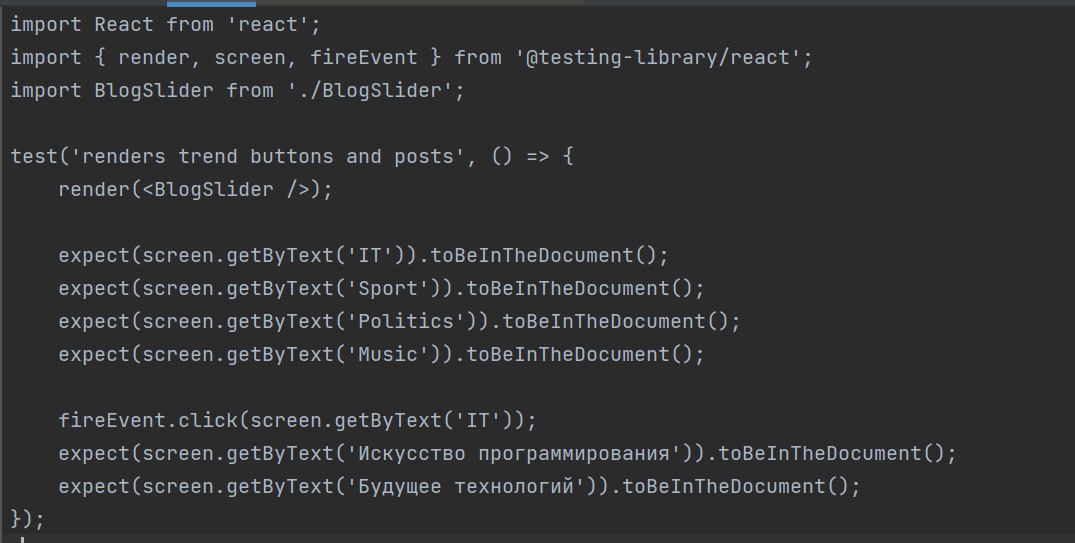
Header.test.js



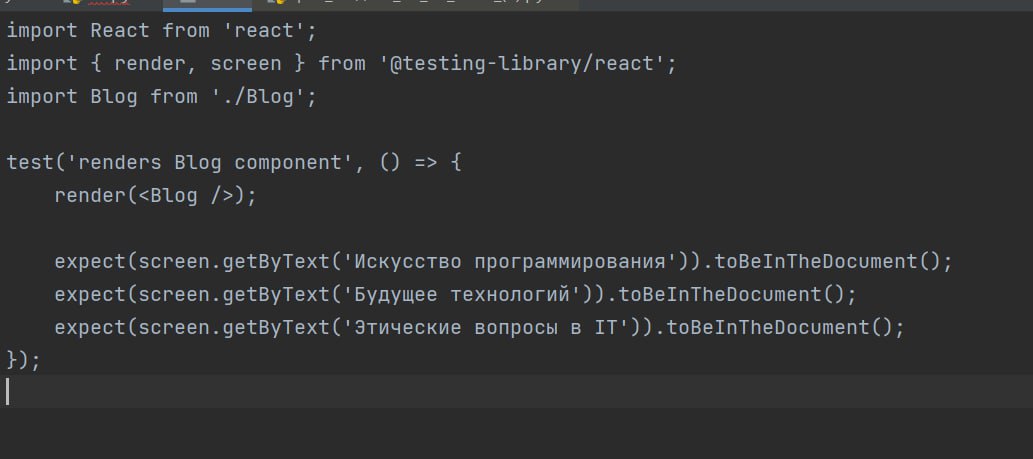
Blog.test.js



InfoPage.test.js



BlogSlider.test.js



**Отчет о тестировании:**

Выполненные тест-кейсы

* ***Header.test.js:*** проверка наличия навигационных ссылок.

Описание: Этот тест проверяет наличие элементов навигации в компоненте Header, чтобы убедиться, что все ссылки на страницы отображаются корректно.

Ожидаемый результат: Все ссылки отображаются корректно.

Фактический результат: Все ссылки отображаются корректно.

Статус: Успешный.

* ***Blog.test.js:*** проверка рендеринга блогов.

Описание: Этот тест проверяет, что компонент Blog правильно рендерит все блоги.

Ожидаемый результат: Все блоги отображаются корректно.

Фактический результат: Все блоги отображаются корректно.

Статус: Успешный.

* ***InfoPage.test.js***: проверка взаимодействия с дисциплинами.

Описание: Этот тест проверяет, что компонент DisciplinesPage правильно рендерит список дисциплин и корректно обрабатывает клики по элементам списка.

Ожидаемый результат: При клике на элемент списка отображается соответствующая информация.

Фактический результат: При клике на элемент списка информация не обновляется.

Статус: Неуспешный.

* ***BlogSlider.test.js***: проверка кнопок трендов и постов.

Описание: Этот тест проверяет, что компонент BlogSlider правильно рендерит кнопки трендов и соответствующие посты.

Ожидаемый результат: При клике на кнопку тренда отображаются соответствующие посты.

Фактический результат: При клике на кнопку тренда посты не обновляются.

Статус: Неуспешный.

Выявленные дефекты

**Defect 1: InfoPage:**

Описание: При клике на элемент списка дисциплин информация не обновляется.

Шаги для воспроизведения:

Открыть компонент DisciplinesPage.

Нажать на элемент списка "Как найти стажировку".

Ожидаемый результат: Отображается текст "Bruh."

Фактический результат: Текст не обновляется.

Предложенное решение: Проверить, правильно ли используется useState для обновления состояния.

**Defect 2: BlogSlider:**

Описание: При клике на кнопку тренда посты не обновляются.

Шаги для воспроизведения:

Открыть компонент BlogSlider.

Нажать на кнопку "IT".

Ожидаемый результат: Отображаются посты для тренда "IT".

Фактический результат: Посты не обновляются.

Предложенное решение: Проверить, правильно ли используется useRef для прокрутки к постам.

**Выводы по работе:**

Разработанные автотесты для компонентов React позволили выявить несколько дефектов, что подчеркивает важность тестирования для обеспечения качества программной системы. Выявленные дефекты в компонентах InfoPage и BlogSlider указывают на необходимость доработки функционала. Тесты помогли проверить корректность рендеринга элементов и взаимодействия пользователя с интерфейсом. В дальнейшем необходимо исправить выявленные ошибки и дополнительно протестировать систему для подтверждения их устранения.

**Используемые источники:**

1. React Testing Library // <https://testing-library.com/docs/react-testing-library/intro/>
2. React: тестируем компоненты с помощью Jest и Testing Library <https://habr.com/ru/companies/timeweb/articles/670480/>