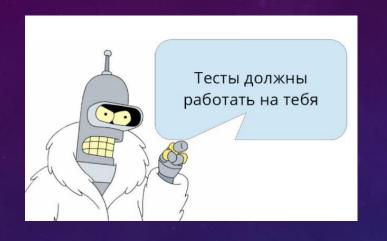
РГАТУ ВКР на тему

«Разработка плагина системы регистрации действий пользователя в браузере с генерацией кода для автоматизации тестирования»

09.03.04 Программная инженерия студент ипб-17 Куликова Э. В. руководитель Смирнов Н. В.

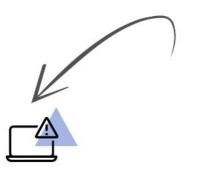
2. АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ



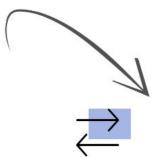
Автоматизированное

тестирование









Хотим избежать людских ошибок

Хотим тестировать быстрее

Хотим тестировать чаще

3. ЦЕЛИ И ЗАДАЧИ

<u>Цель</u> — упростить процесс автоматизации тестирования, сократить время на написания автоматизированных тестов и увеличить количество тестов.

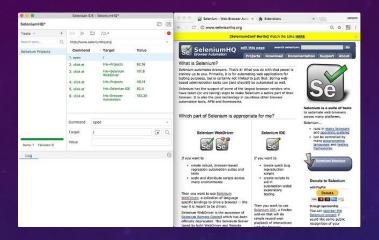
Задачи:

- Создать плагин, который будет регистрировать действия пользователей в браузере;
- Определять по каким иі-компонентам web-приложения совершено действие;
- Добавить генерацию *java-кода тестов* (шаблонов) для ui-компонентов адаптированного под корпоративную библиотеку *autotest-lib*.

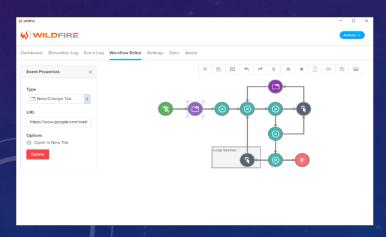
🖟 🗓 | Elements Console Sources Network Performance Memory Application Lighthouse Overrides Logger user actions

4. АНАЛОГИ

Selenium IDE



Wildfire.ai



Сравнительные критерии	Selenium IDE	Wildfire.ai
Поддерживает браузеры Google Chrome и Firefox	+	+
Записывает взаимодействия пользователя с веб-страницей	+	+
Запись и воспроизведение теста	+	+
Сохранение и загрузка скриптов для последующего воспроизведения	+	+
Возможность просматривания и моделирования журнала симуляции	_	+
Использование инструмента Selenium	+	+
Генерация кода теста	_	_

5. ОСНОВНЫЕ ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

- Регистрировать действия пользователей при работе в веб-приложении на странице браузера;
- Анализировать и обрабатывать полученные события с веб-страницы приложения;
- Определять с какими иі-компонентами и элементами веб-приложения взаимодействовал пользователь;
- Сохранять собранные данные о событиях в клиентское хранилище IndexedDB для последующей с ними работы;
- Превращать собранные действия в код на языке программирования java, адаптированный под корпоративную библиотеку автоматизации тестирования autotest-lib;
- Предоставлять пользователю через графический интерфейс следующие функции:
 - Начать генерацию кода автоматизированного теста;
 - Закончить генерация кода автоматизированного теста;
 - Обновить лог событий;
 - Скопировать сгенерированный код;
 - Посмотреть сгенерированный код автотеста и лог событий выбранной сессии.

6. ИСПОЛЬЗУЕМЫЕ ТЕХНОЛОГИИ И ИНСТРУМЕНТЫ

Pасширение в браузере Google Chrome



Инструменты разработчика DevTools

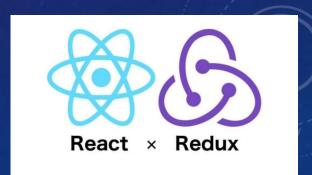








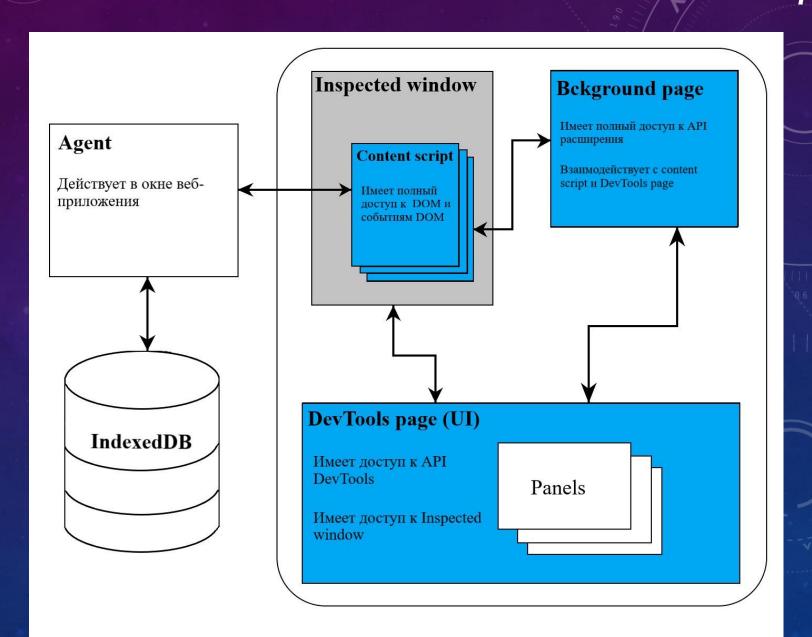




7. АРХИТЕКТУРА

Agent и DevTools общаются между собой посредством отправки сообщений через content-script и background:

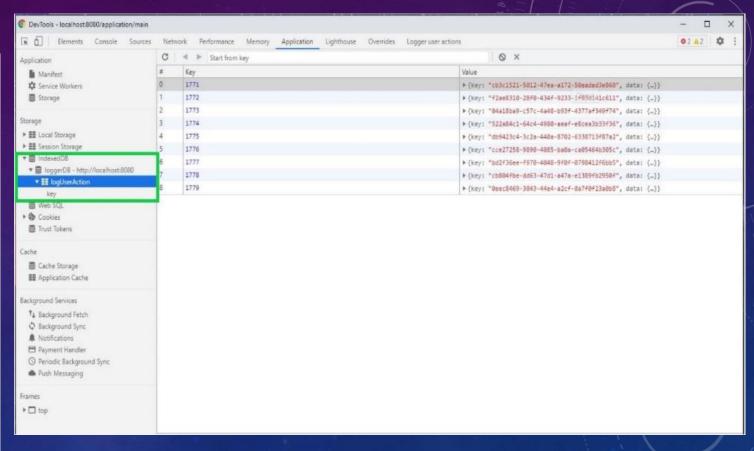
 $Agent \leftrightarrow content\text{-}script \leftrightarrow background \leftrightarrow UI.$



8. БАЗА ДАННЫХ INDEXEDDB

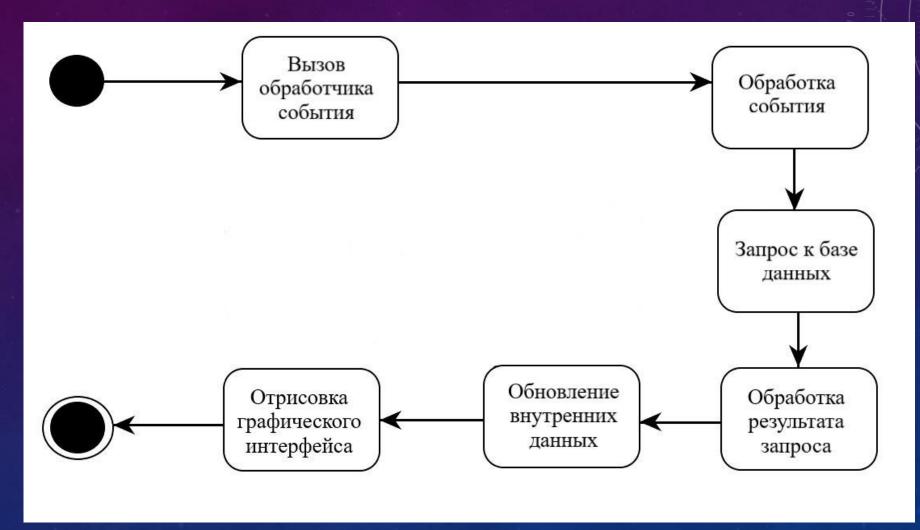
Сохранённый объект данных в IndexedDB

```
▼{key: "04a18ba9-c57c-4a48-b93f-4377af349f74", data: {...}}
▼ data:
   bigButton: undefined
   date: 1620726751258
    filterCaption: "Предметный класс"
    filterToolbar: undefined
    grid: undefined
    gridCellCol: undefined
    gridCellRow: undefined
    gridToolbar: undefined
    isFilter: true
    modalWindow: undefined
    modalWindowButton: undefined
    nameForm: "Конфигурация пакетной замены"
    quickFilter: undefined
    tab: "Конфигурация пакетной замены"
    tabContainer: "TCD_Filters"
   type: "change"
    value: ""
    xpath: "/HTML[1]/BODY[1]/TABLE[1]/TBODY[1]/TR[2]/TD[1]/DIV[1]/DIV[1]/DIV...
  key: "04a18ba9-c57c-4a48-b93f-4377af349f74"
```



После обработки действия пользователя, формируется объект данных для сохранения его в хранилище IndexedDB, чтобы потом превратить его в строчки java-кода автоматизированного теста, который будет проверять UI-компонент, с которым работал пользователь.

9. АЛГОРИТМ РАБОТЫ ПРИЛОЖЕНИЯ



10. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ ПОЛЬЗОВАТЕЛЯ

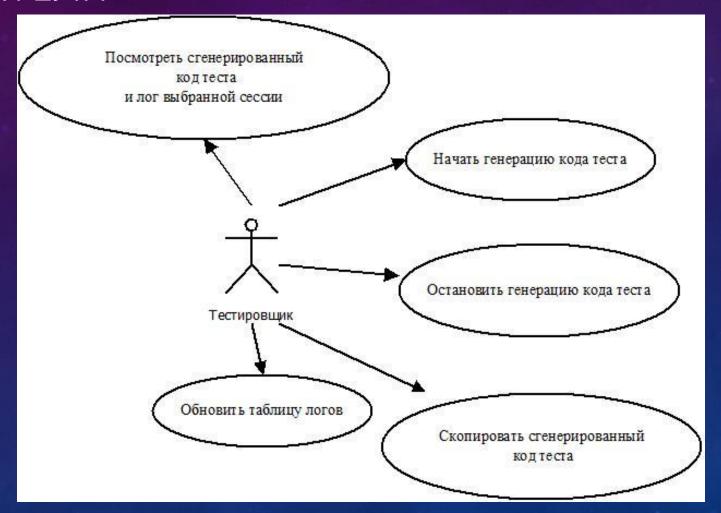
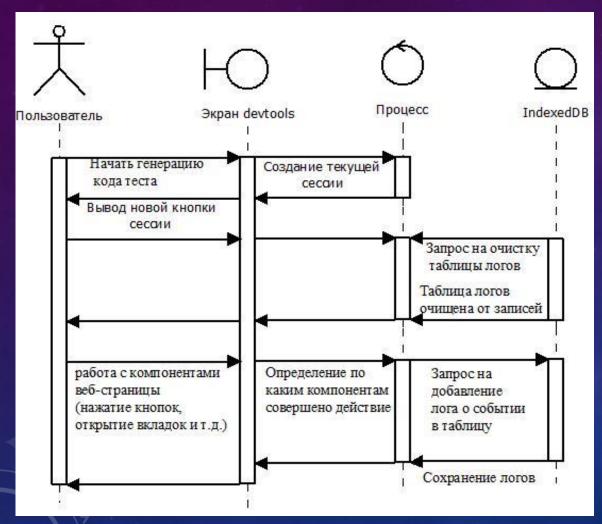


Диаграмма вариантов использования системы

11. ПРОЦЕСС РАБОТЫ ГЕНЕРАЦИИ КОДА АВТОТЕСТА



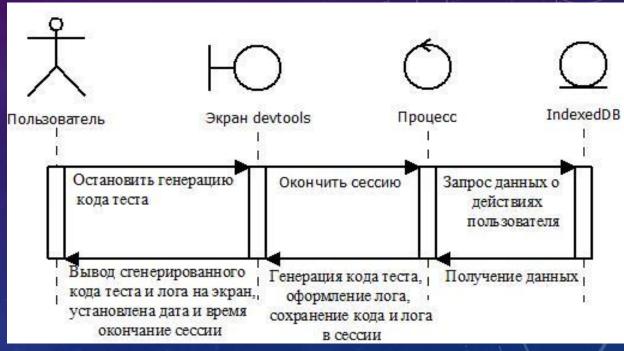
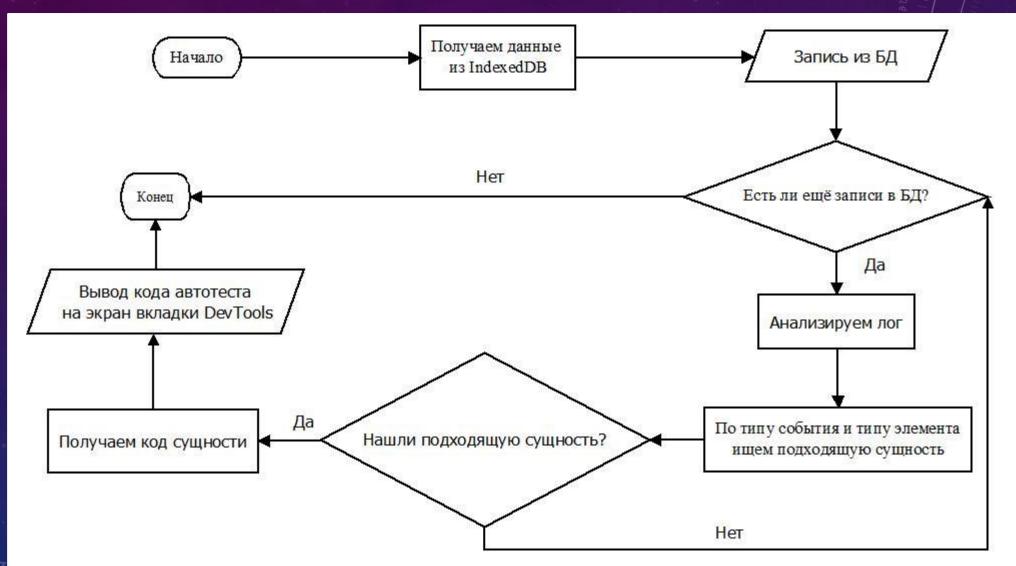


Диаграмма последовательности окончания генерации

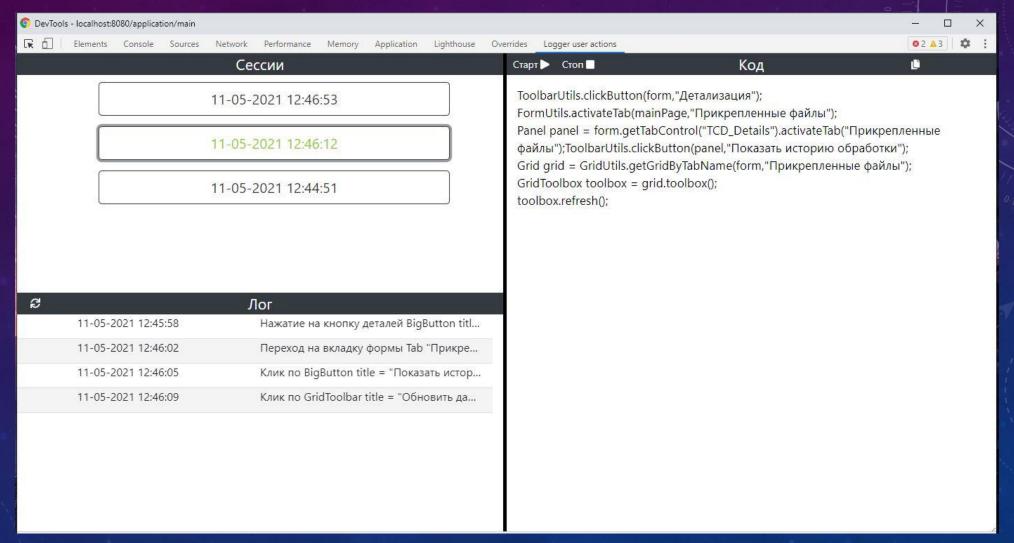
автотеста

Диаграмма последовательности старта генерации автотеста

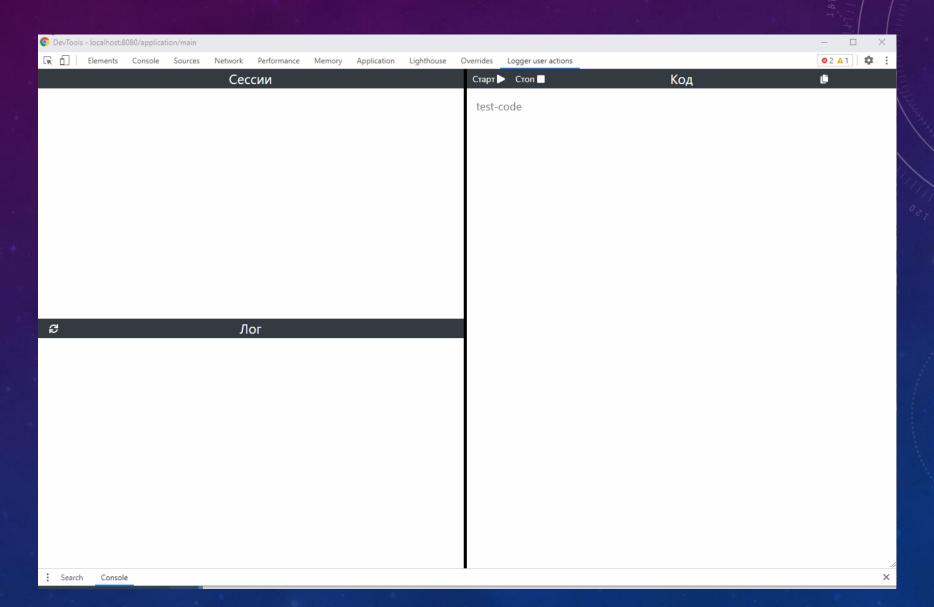
12. АЛГОРИТМ ГЕНЕРАЦИИ КОДА АВТОТЕСТА



13. ТЕСТИРОВАНИЕ ПРОГРАММЫ



14. ДЕМОНСТРАЦИЯ РАБОТЫ ПЛАГИНА



ЗАКЛЮЧЕНИЕ

- ✓ Спроектирован и разработан плагин (расширение) для браузера, который регистрирует действия пользователя в веб системах «НПО «Криста», для последующего преобразования в программный код, способный повторить эти действия в автоматизированном режиме с учетом различных тонкостей, разрабатываемых предприятием программных продуктов;
- ✓ Полученная система может использоваться специалистами предметной области для ускорения разработки функциональных автоматизированных тестов для интерфейса веб-приложений НПО «Криста».