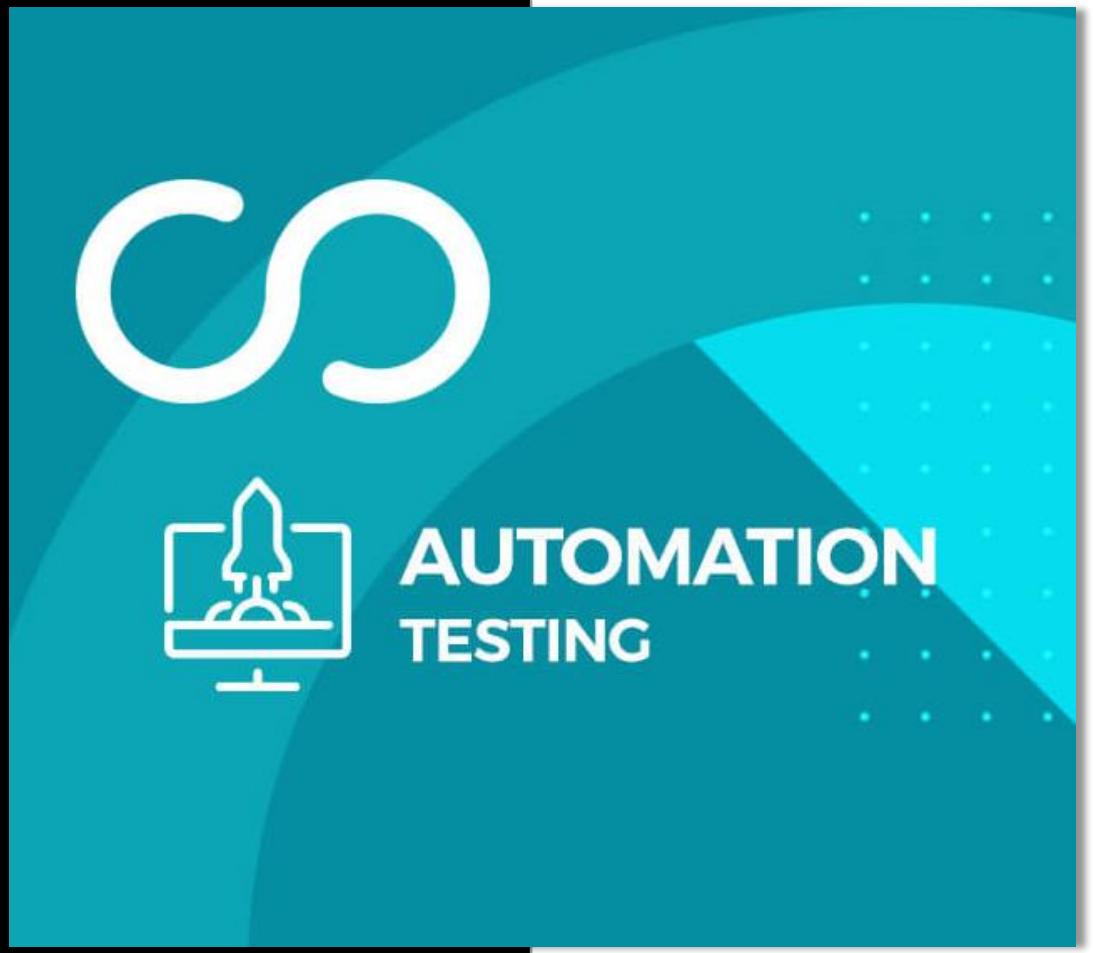
Кафедра ПИКС

Библиотека «kbv-testdriver» на базе Selenium Webdriver для автоматизации тестирования UI и API веб-сайтов



Выполнил студент группы 883871

БУЛЫШКИН Кирилл Викторович

Научный руководитель - проректор по учебной работе, канд.тех.наук, доцент кафедры ПИКС

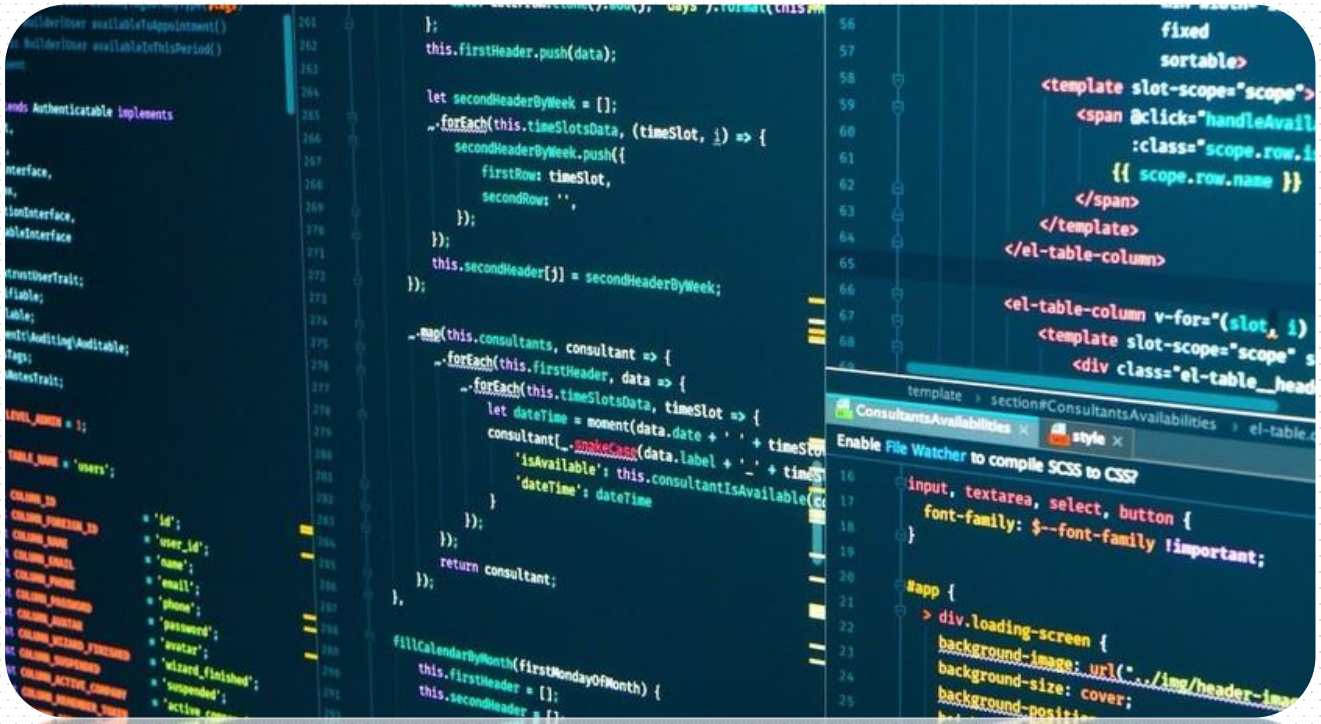
ШНЕЙДЕРОВ Евгений Николаевич

Актуальность выбранной темы

На текущий момент времени автоматизация тестирования является одним из наиболее бурно развивающихся направлений в IT-индустрии.

Качество разрабатываемых продуктов является крайне важным моментом, по причине высокой конкурентности рынка.

Функциональность разрабатываемой библиотеки



- Работа с графическими элементами веб-страниц.
- Проверка корректности функционирования веб-страниц.
- Возможность взаимодействия с программируемыми интерфейсами приложений.
- Возможность репрезентации итогов проведенных тестов помощью отчетов по результатам тестирования.
- Наличие подсистемы мониторинга работы библиотеки.

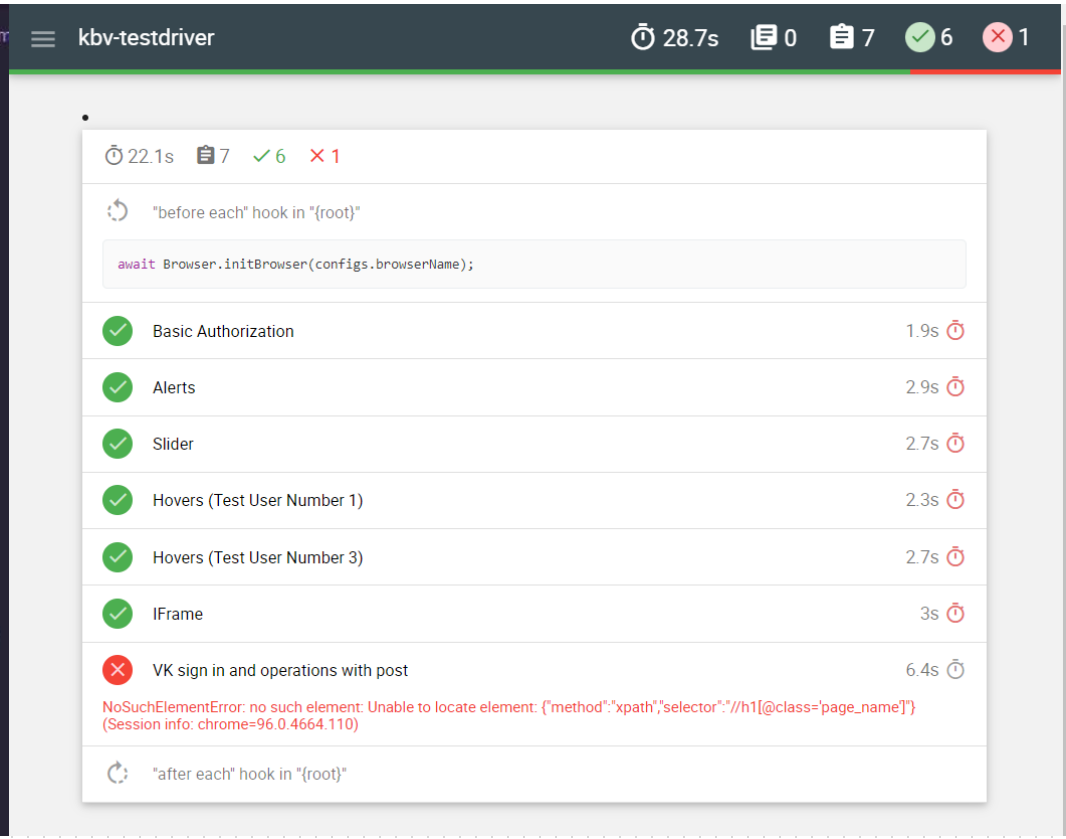
Технологии и инструменты, применяемые при разработке библиотеки

Алгоритм взаимодействия пользователя с разрабатываемой библиотекой

- Предполагаемый пользователь (тестировщик), будет взаимодействовать с разрабатываемой библиотекой из консоли используемой среды разработки.
- Для запуска тестов достаточно будет ввести команду npm test, после чего будет произведено открытие браузера и выполнение тестовых сценариев.
- Результаты выполнения тестов будут выведены в консоль.
- Также пользователь имеет возможность просмотра отчета по результатам выполнения тестов.

Управление библиотекой

```
JS test.js M ×  JS wallPage.js  JS requests.js ●  JS navigationBarPageLocators.js M  JS hooks.js  JS imagesComparing.js  mochawesome.htm
project > JS test.js > it('VK sign in and operations with post') callback
49
50  it('Slider', async () => {
51          await Browser.navigate(`${testData.host}${path.slider}`);
52          await Browser.windowMaximize();
53          const sliderPage = new SliderPage();
54          expect (await sliderPage.isDisplayed()).to.eql(testData.pageIsDisplaying);
55          const randomSliderPosition = RandomGenerators.getRandomValueFromArray(sliderValues);
56          await sliderPage.dragAndDropSlider(randomSliderPosition);
57          expect (await sliderPage.getSliderValue()).to.eql(sliderMap.get(randomSliderPosition));
58  });
59
60  hoversTestUserNumbers.forEach(function(userNumber) {
61
62          it(`Hovers (Test User Number ${userNumber})`, async () => {
63                  await Browser.navigate(`${testData.host}${path.hovers}`);
64                  await Browser.windowMaximize();
65                  const hoversPage = new HoversPage();
66                  expect (await hoversPage.isDisplayed()).to.eql(testData.pageIsDisplaying);
67                  await hoversPage.moveCursorToHover(userNumber);
68                  expect (await hoversPage.getHoverText(userNumber)).to.eql(testData.hoverText(userNumber));
69                  expect (await hoversPage.hoverLinkIsDisplayed(userNumber)).to.eql(testData.elementIsDisplaying);
70                  const hoverLinkHref = await hoversPage.getHoverLinkHref(userNumber);
71                  await hoversPage.hoverLinkClick(userNumber);
72                  expect (await Browser.getCurrentUrl()).to.eql(hoverLinkHref);
73                  await Browser.backToPreviousPage();
74          });
75
76  });
```



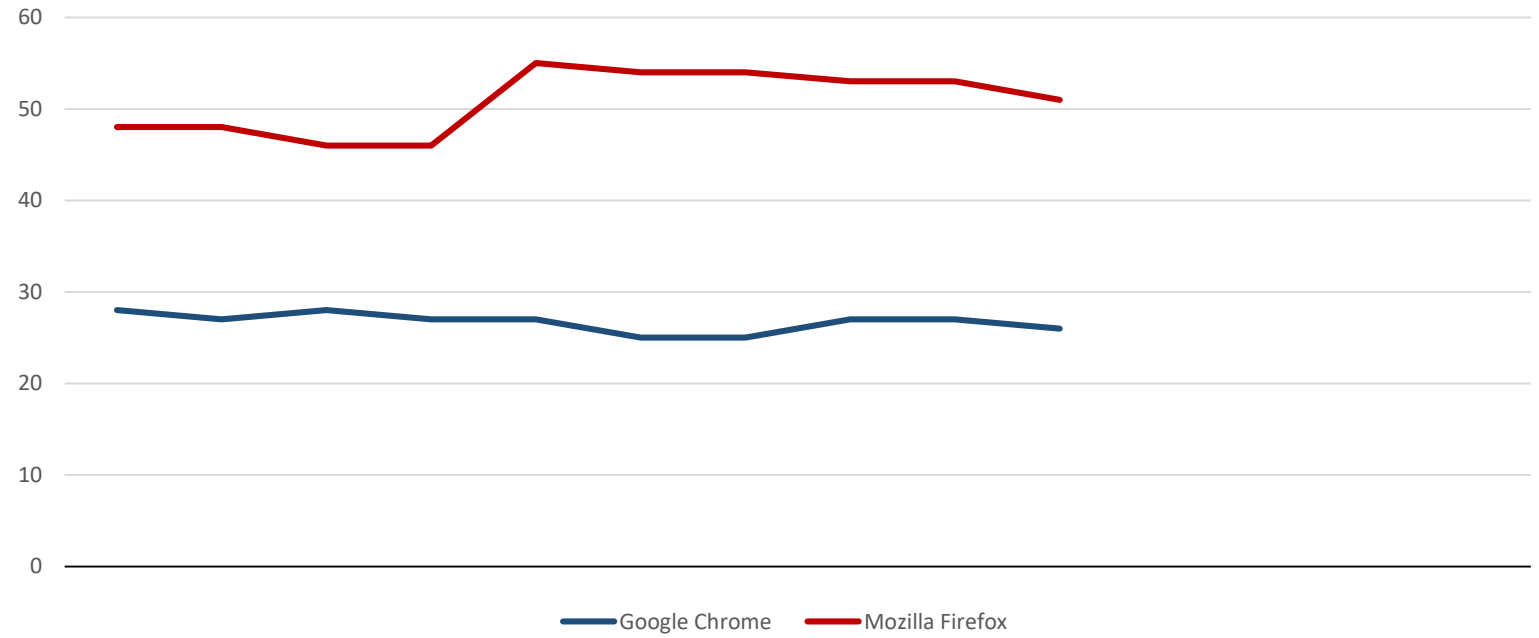
Пример автоматизированных тестов, созданных с помощью библиотеки «kbv-testdriver»

Пример отчета по результатам выполнения тестов

Оценка временных показателей работы библиотеки

Итерация, номер	1	2	3	4	5	6	7	8	9	10
Google Chrome, с	28	27	28	27	27	25	25	27	27	26
Mozilla Firefox, с	48	48	46	46	55	54	54	53	53	51

Временные показатели тестовых прогонов веб-сервиса «http://the-internet.herokuapp.com»



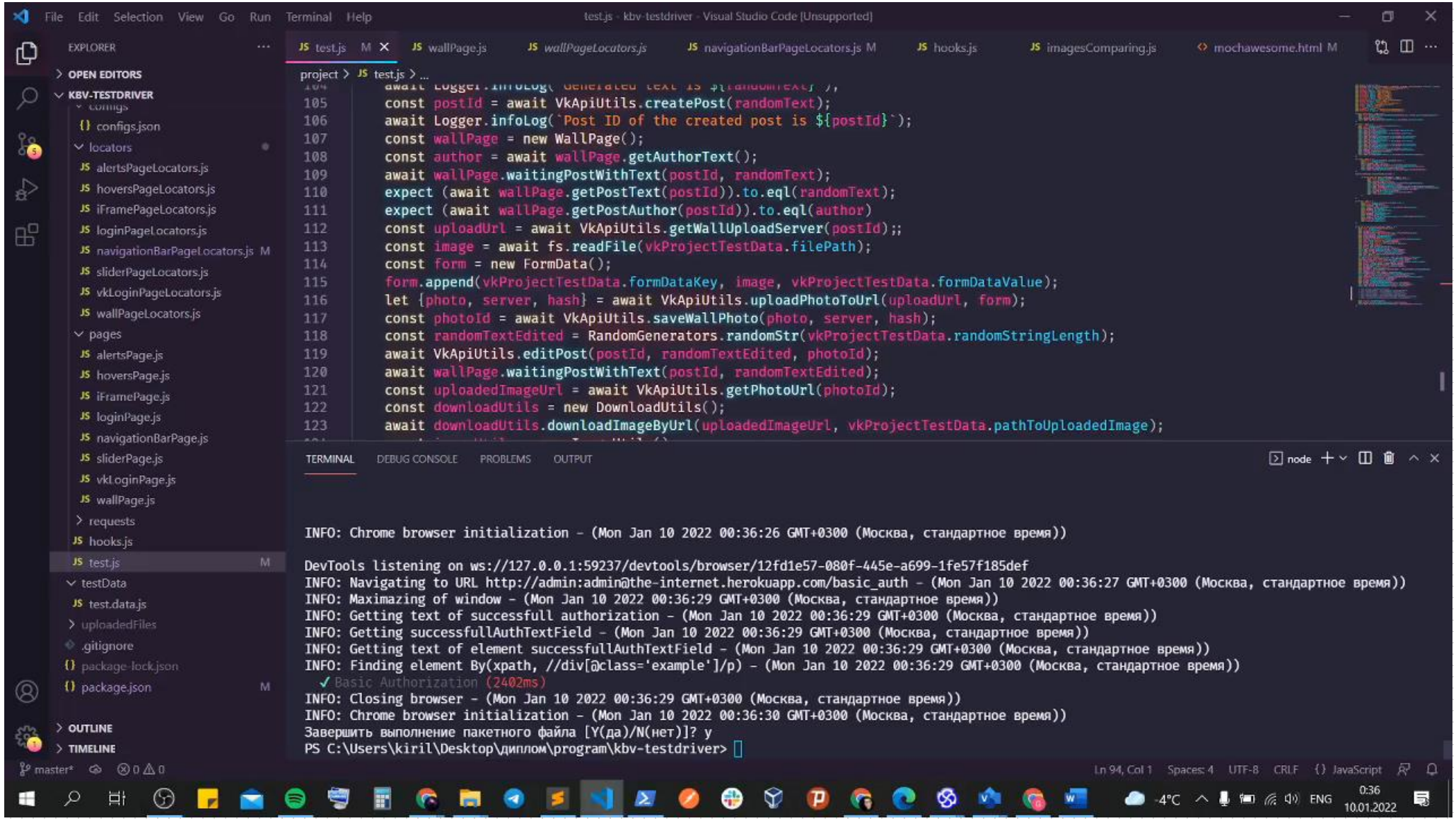
График, отображающий временные показатели при выполнении тестовых сценариев

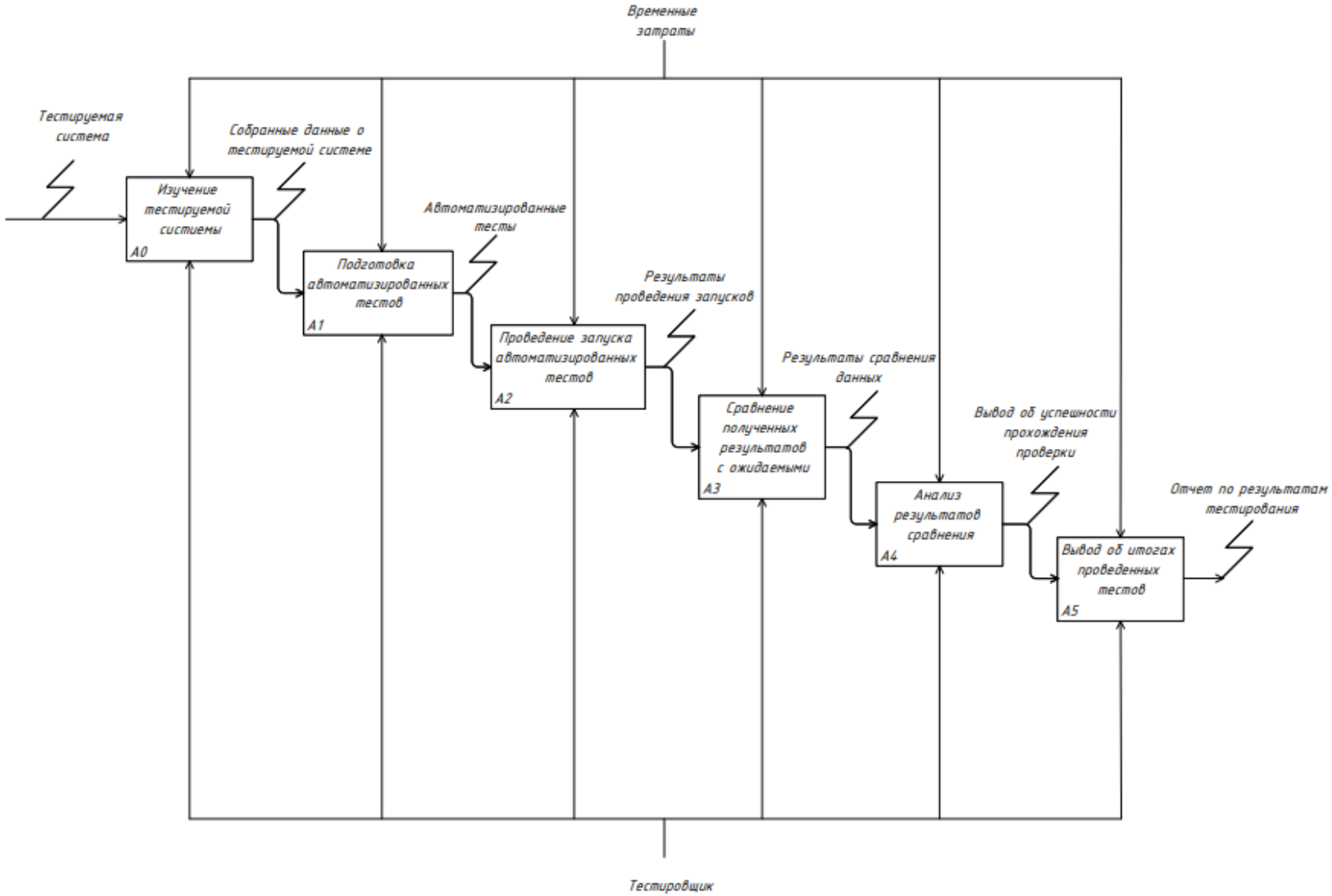
Заключение

- В ходе выполнения дипломного проекта была спроектирована и разработана библиотека «kbv-testdriver» на базе Selenium Webdriver для автоматизации тестирования UI и API веб-сайтов.
- Разработанная библиотека полностью соответствует техническим и экономическим требованиям.
- Разработанная библиотека полностью готова к использованию и реализации автоматизации тестирования веб-сайтов.

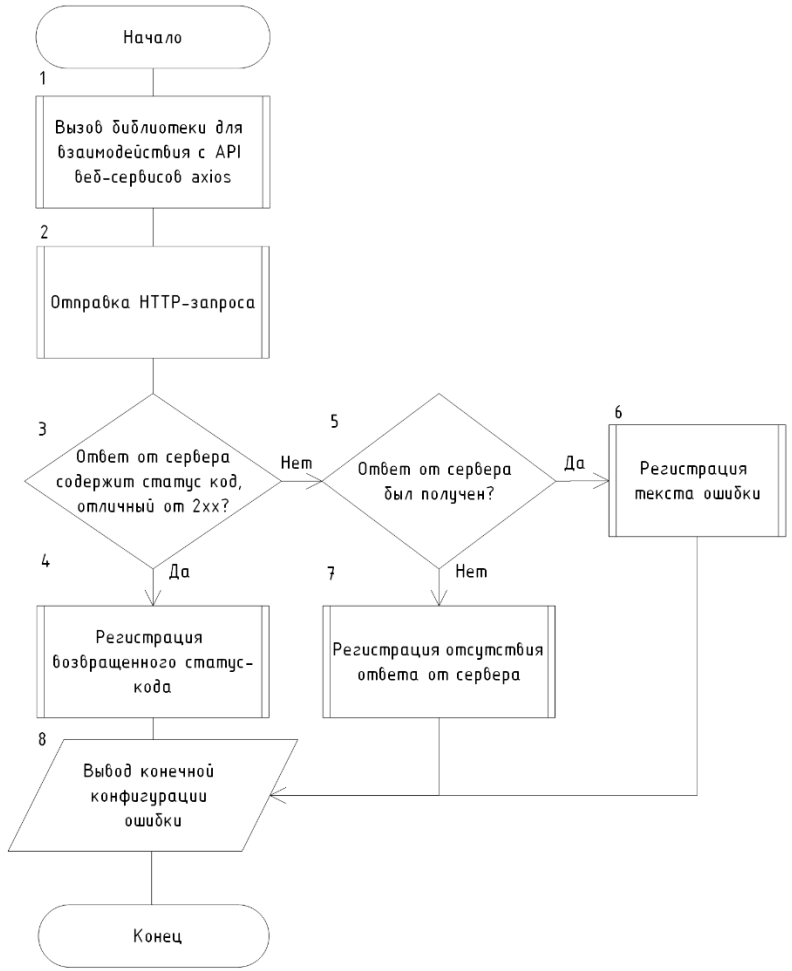
Спасибо за внимание!

Видеозапись работы библиотеки

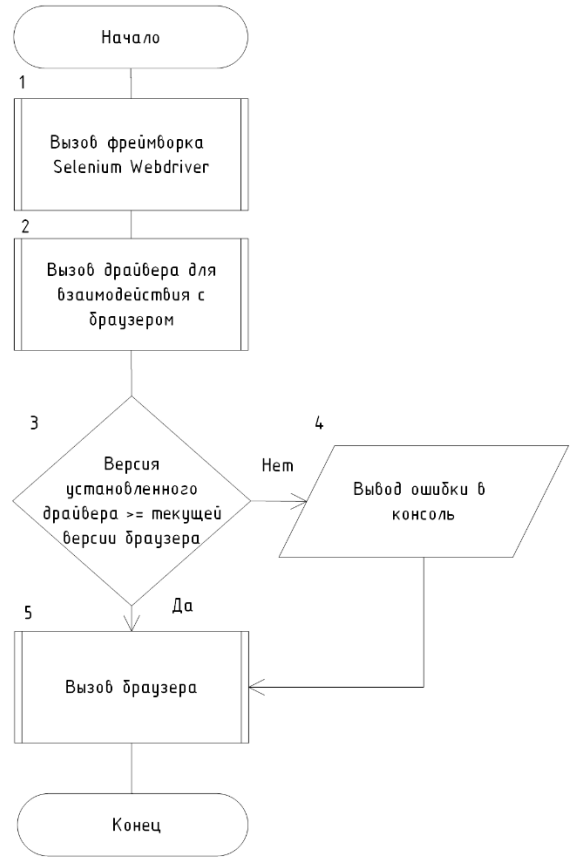



Временные затраты
Тестируемая система
Собранные данные о тестируемой системе
Автоматизированные тесты
Результаты проведения запусков
Результаты сравнения данных
Вывод об успешности прохождения проверки
Отчет по результатам тестирования
Изучение тестируемой системы
A0
Подготовка автоматизированных тестов
A1
Проведение запуска автоматизированных тестов
A2
Сравнение полученных результатов с ожидаемыми
A3
Анализ результатов сравнения
A4
Вывод об итогах проведенных тестов
A5
Тестировщик

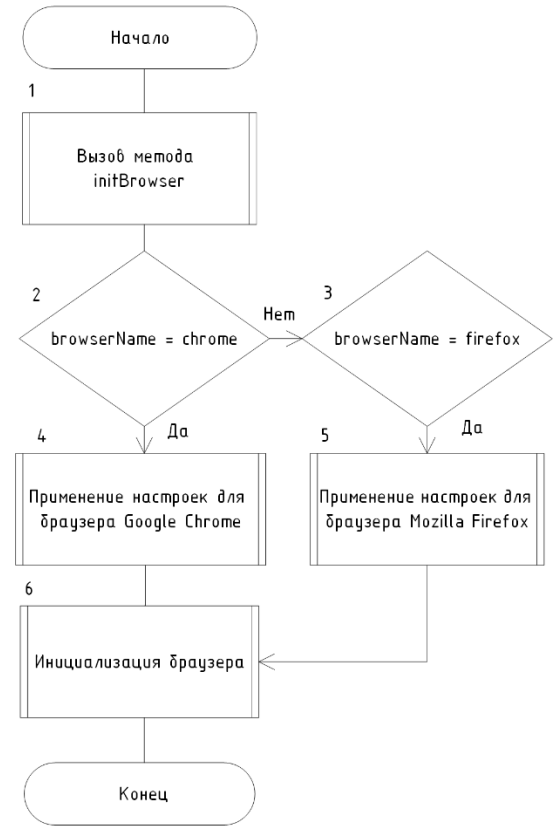
Обработка ошибок при API-вызове в библиотеке kbv-testdriver



Алгоритм вызова браузера библиотекой kbv-testdriver



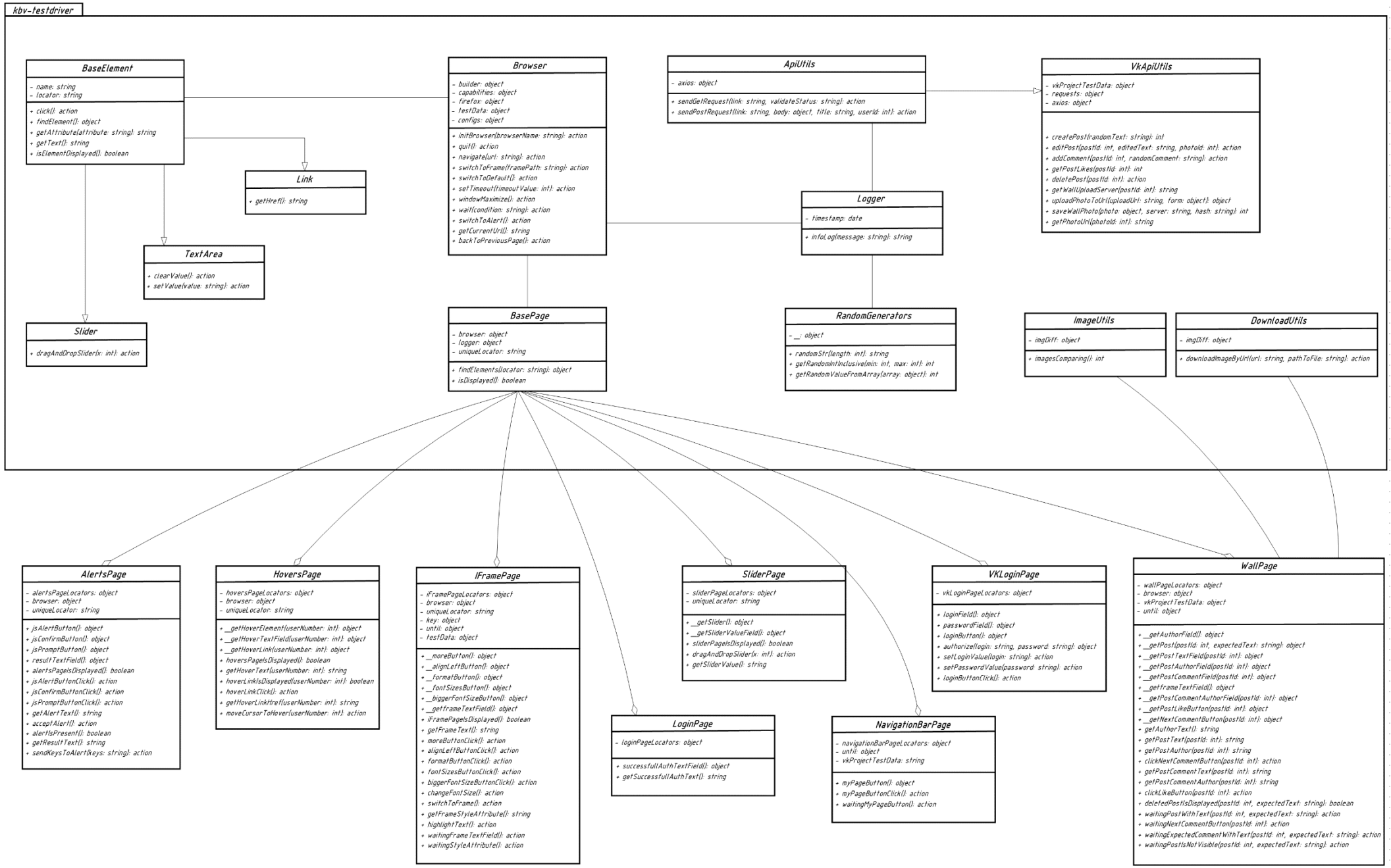
Алгоритм проверки и настройки выбранного пользователем браузера в библиотеке kbv-testdriver



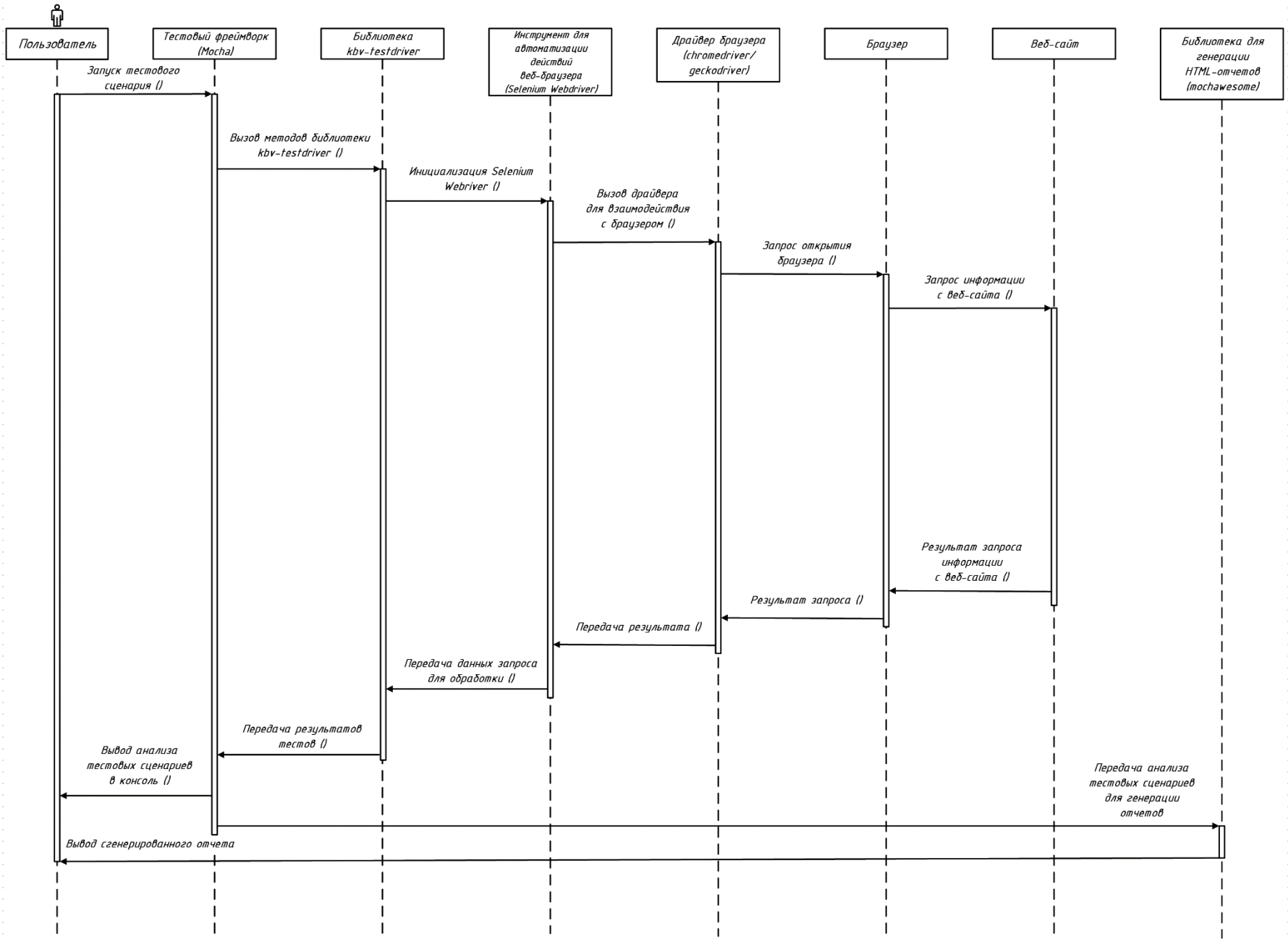
									ГУИР.88387-01 02 01
Изм.	Лист	N докум.	Подп.	Дата			Лит.	Масса	Масштаб
Разраб.		Бурмакин			Схемы алгоритмов				
Провер.		Шнейдеров					Лист	Листов 1	
Т.контр.		Шнейдеров							
Реценз.		Васильев							
Н.контр.		Жданович					ПИКС гр. 883871		
Утв.		Шнейдеров							

Формат А1

UML диаграмма классов



UML диаграмма последовательности (отработки тестового сценария)



Графический интерфейс отчета о результатах выполнения тестов

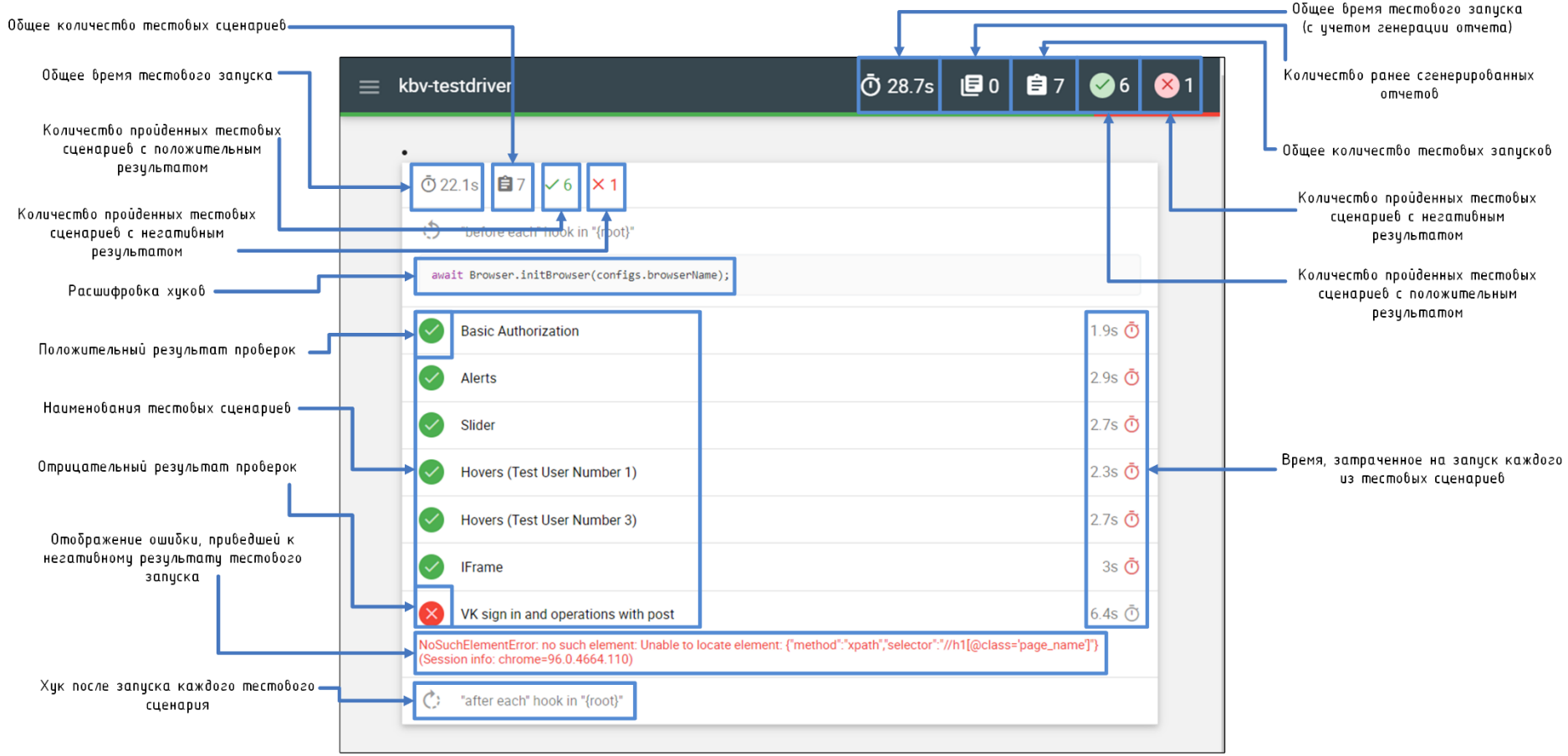


Рисунок 1 – Общий вид отчета

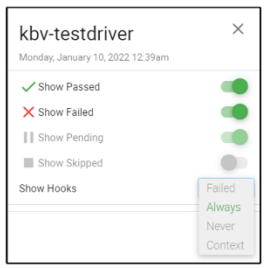


Рисунок 2 – Панель конфигурации отчета по результатам выполнения тестов

Рисунок 3 – Расшифровка тестового сценария с положительным результатом

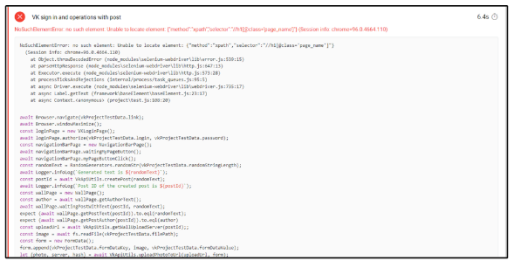


Рисунок 4 – Расшифровка тестового сценария с негативным результатом

Результаты оценки количественных показателей программного средства

Временные показатели запуска автоматизированных тестов веб-сервиса http://the-internet.herokuapp.com

Итерация, номер	1	2	3	4	5	6	7	8	9	10
Google Chrome, с	28	27	28	27	27	25	25	27	27	26
Mozilla Firefox, с	48	48	46	46	55	54	54	53	53	51

Выводы:

- все 20 запущенных подряд тестовых прогонов были успешно выполнены, что свидетельствует о высокой степени стабильности разработанных нами автоматизированных тестов и надежности при работе с библиотекой kbv-testdriver;
- разбежка во времени при запуске тестовых сценариев будет меньше при использовании браузера Google Chrome;
- тестовые запуски совершаются в среднем на 30 процентов быстрее в браузере Google Chrome.

Зарегистрированные показатели потребления оперативной памяти в ходе запуска автоматизированных тестов веб-сервиса http://the-internet.herokuapp.com

Вид нагрузки на систему	Объем свободной оперативной памяти, МБ
Минимальный расход оперативной памяти	74
Пиковый расход оперативной памяти	8

Выводы:

- если произвести расчеты относительно изначально доступной оперативной памяти (рис.3), потребление оперативной памяти, создаваемое тестовыми запусками при использовании библиотеки kbv-testdriver составляет 286 МБ;
- оперативная память расходуется умеренно, никаких дополнительных мероприятий по оптимизации не требуется.

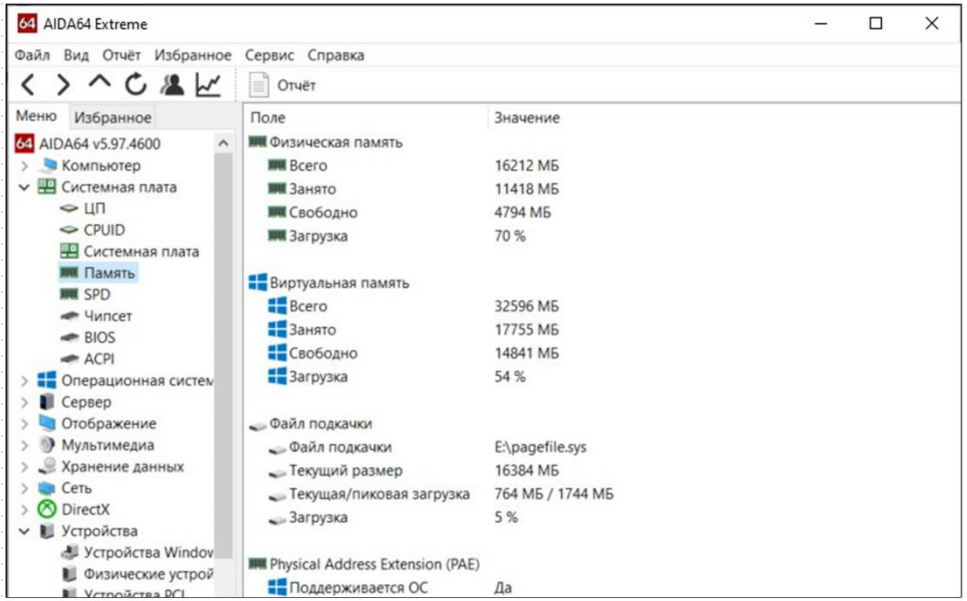


Рисунок 1 – Аппаратные характеристики оперативной памяти в AIDA64

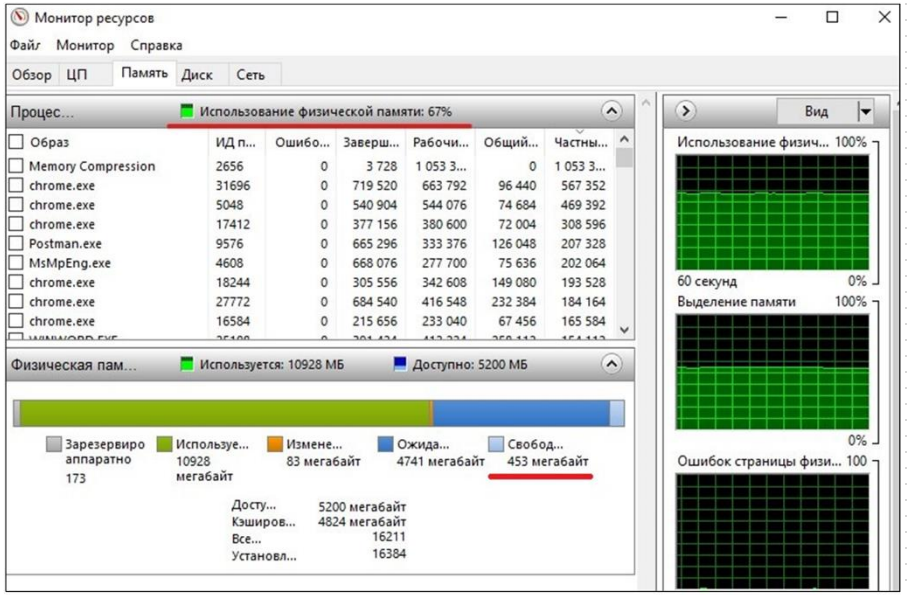


Рисунок 2 – Исходные показатели потребления оперативной памяти в Мониторе ресурсов