Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Кубанский государственный технологический университет»

**Институт КСиИБ**

**Кафедра информационных систем и программирования**

Лабораторная работа №5

по дисциплине «Методология программной инженерии. Технологии, этапы и принципы управления качеством разработки программных комплексов »

Выполнил студент

группы 18-КМ-ПР

Тхам Конг Тон Бао

Преподаватель:

Попова Ольга Борисовна

Краснодар 2019

**Лабораторная работа №5**

**Использование средств автоматизации тестирования программного обеспечения**

Цель работы: разработать тесты для приведенных классов, провести регрессионное тестирование, выполнить профилирование программы**.**

**Реферат: Katalon solution – Инструмент для автоматизации тестирования веб/мобилных приложений и веб-сервисов**

**Введение**

Katalon Studio - эффективный инструмент для автоматизации процесса тестирования веб-приложений, мобильных приложений и веб-сервисов. Katalon Studio является потомком таких фреймворков, как Selenium и Appium. От последних он перенял множество преимуществ, связанных с комплексной автоматизацией тестирования программного обеспечения.

Чтобы начать работу с этим инструментом, вы можете одновременно иметь начальные знания в области тестирования программного обеспечения и быть настоящим гуру своего бизнеса. Люди, далекие от программирования, могут легко запустить свой проект для автоматизации тестирования (например, запустив функцию Object Spy для написания тестовых сценариев), а для программистов и опытных тестировщиков Katalon Studio окажется полезной с точки зрения экономии времени при написании новых библиотеки и поддержка существующих скриптов.

|  |  |
| --- | --- |
| **Беглый обзор** |  |
| Katalon Studio - это простое и мощное решение для автоматизации, созданное для тестировщиков во всем мире. Katalon Studio революционизирует использование сред автоматизации тестирования с открытым исходным кодом, таких как Selenium и Appium, устраняя их технические сложности, чтобы позволить разработчикам и QA эффективно настраивать, создавать, запускать, составлять отчеты и управлять своими автоматизированными тестами. Он также предлагает жизнеспособную альтернативу коммерческим решениям по автоматизации тестирования, которые недоступны для многих малых и средних команд.  обзор |  |

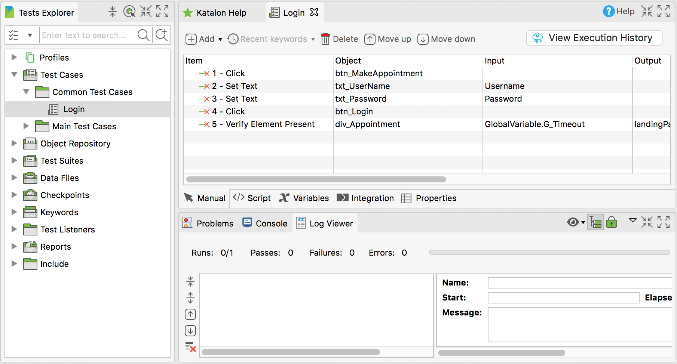


Рисунок 1 - Katalon Studio глабный экран

**Ключевые особенности Katalon Studio:**



Рисунок 2 – Katalon достижение

* Простое развертывание: Один комплексный деплоймент пакет содержит все необходимое для развертывания мощного инструмента автоматизации тестирования
* Быстрая и простая настройка. Katalon Studio не только легко устанавливается, но также имеет простую настройку среды тестирования. Тестировщики могут запускать свой первый тестовый скрипт довольно быстро, используя предварительно созданные шаблоны и тестовые скрипты, такие как репозитории объектов и библиотеки ключевых слов
* Результаты быстрее и лучше: встроенные шаблоны с понятными туториалами помогают тестировщикам быстро создавать и запускать скрипты. Они могут выполнять каждый шаг быстро и эффективно, будь это настройка проекта, создание и выполнение тестов или создание отчетов
* Гибкие режимы: новые тестировщики могут использовать recording и ключевые слова для создания автоматизированных тестов, в то время как у более опытных есть доступ к полной IDE для создания расширенных сценариев
* Простота использования: даже manual тестировщик с минимальным опытом программирования может с легкостью пользоваться преимуществами Katalon Studio
* Кросс-браузерное приложение: Katalon Studio поддерживает различные платформы: Windows 32 и 64 (7, 8 и 10) и OS X 10.5+

**Подробнее о Katalon Studio**

Katalon Studio - это удобное в использовании решение для автоматизации, которое отличается простотой и скоростью. Для команды автоматизаторов довольно удобно прилагать как можно меньше усилий для создания нового проекта, для выполнения тестов и мониторинга результатов выполнения. Каждый рабочий процесс предоставляет множество возможностей и кастомизаций для упрощения обслуживания и увеличения масштаба проекта:

* + Предварительно определенная структура артефактов: тест-кейсы, тест-сьюты, тест-объекты и отчеты. Тестировщикам больше не нужно тратить часы на их создание и поддержку
  + Кастомные ключевые слова обеспечивают гибкость в добавлении дополнительных ключевых слов для эффективного тестирования AUT для конкретных и сложных целей тестирования
  + Поддержка основных потребностей тестирования: Web, Mobile и [API](https://automated-testing.info/tags/api)
  + Выполняйте сразу несколько тест-сьютов с test-suite collection
  + Расширяйте текущий CI процесс с помощью режима консоли без лишних усилий. Запуск командной строки можно быстро осуществить при помощи опции “Generate Command Line for console mode’
* С лёгкостью отслеживайте результаты выполнения в виде таблицы или “дерева” как во время выполнения так и после.
* Подробный отчет Test Suite сокращает время анализа результатов. Вы можете экспортировать его в различные форматы, такие как CSV, PDF, HTML и сохранить его для последующего использования

**Навигация по линейному, упорядоченному рабочему процессу с помощью Katalon Studio:**

Начните

* Автоматическое создание тестов. Записывает действия и автоматически генерирует скрипты с использованием встроенных ключевых слов.
* High-end скриптинг: позволяет создавать сложные тестовые сценарии или настраиваемые ключевые слова с легкостью и эффективностью.
* Умный захват объекта: улучшенный рекордер эффективно определяет свойства объекта, чтобы максимизировать распознавание.

Выполняйте

* Выполнение теста стало мощнее: запускает тест-кейсы или тест-сьюты с использованием множества конфигураций и наборов данных.
* Универсальность в выполнении: предоставляет консоль интеграции CI с различными параметрами для удаленного выполнения. Выполняет тесты на нескольких браузерах и операционных системах локально, а также на Sauce Labs и BrowserStack.
* Гибкая обработка отказа и автоматическое повторное выполнение. Включает правила выполнения для автоматической обработки сложных потоков.
* Получайте отчёты
* Отчеты доступны в нескольких форматах: с расширенным протоколированием, отладочными данными и скриншотами
* Отчёты о выполнении заказа: интегрированы с процессом уведомлений .
* Расширенные логи Selenium и Appium: c возможностями более глубокого анализа для улучшения стратегии автоматизации.

**Поддерживайте**

* Умная поддержка тест-объектов: автоматически обновляет все связанные с этим тестовые примеры и комплекты при изменении объектов.
* Эффективная организация тестирования. Позволяет легко управлять и поддерживать тесты, данные и ключевые слова.
* Упрощенное взаимодействие внутри команд: Интегрируется с GIT, чтобы члены команды могли легко делиться артефактами и рабочей нагрузкой.

**Может ли Katalon Studio быть жизнеспособной альтернативой дорогостоящим коммерческим инструментам?**

Katalon Studio достаточно хорош по сравнению с коммерческими инструментами автоматизации тестирования, такими как UFT и Test Complete по нескольким причинам. Он представляет собой жизнеспособный, практически бесплатный вариант для таких инструментов, доступных на рынке:

**Katalon Studio**

* Никаких затрат на лицензирование и обслуживание
* Бесплатные Апгрейды
* Кросс-платформенное решение

**UFT/Test Complete**

* Плата за лицензирование и обслуживание достаточно высока
* Высокая плата за обновления и дополнительные модули
* Только под Windows

Katalon Studio был недавно опубликован, но получил широкое признание среди QA сообщества. Он предоставляет более легкий и быстрый способ тестирования для команд тестировщиков. Благодаря простому развертыванию, широкому набору функций, простоте использования и большому количеству вспомогательных ресурсов, Katalon Studio открывает новые горизонты в области автоматизации тестирования.

Чтобы создать новый проект API / веб-службы, в диалоговом окне «Новый проект» просто выберите тип API / веб-службы и укажите всю необходимую информацию о проекте.

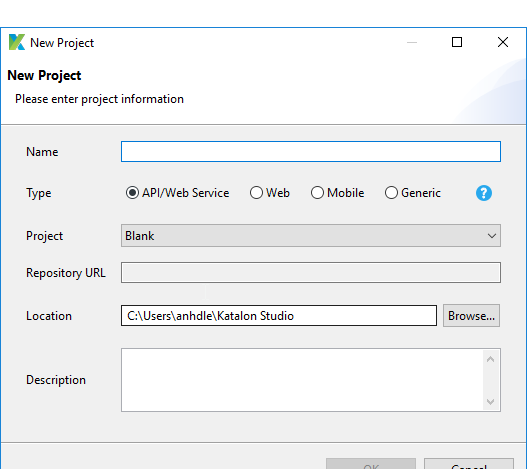
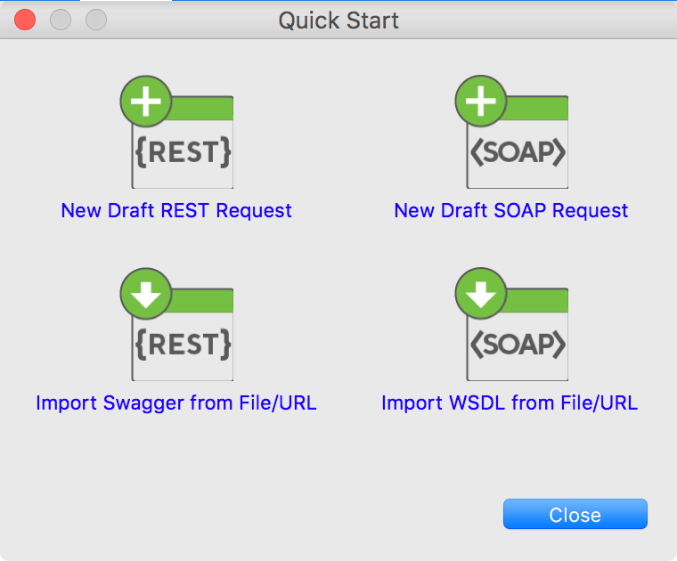


Рисунок 3 – Создать новый проект

Чтобы ускорить создание запроса веб-службы, Katalon Quickstart позволяет быстро создавать новые запросы или импортировать определение запроса из файла / URL-адреса одним щелчком мыши.



Варианты быстрого запуска включают в себя:

• Новый предварительный запрос REST / SOAP. Позволяет создать предварительный запрос для проверки различных данных перед сохранением в репозитории объектов. Все отправленные запросы будут сохранены на панели «История запросов».

• Импортировать Swagger 2.0 / WSDL из файла / URL-адреса: вы можете импортировать определение веб-службы из локального файла или URL-адреса.

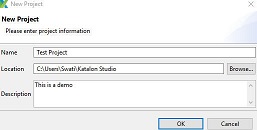
Katalon Studio загрузит указанную информацию в свой проект.Этот быстрый запуск API / веб-службы запускается автоматически при первом создании проекта нового API / веб-службы. Вы можете запустить его в любое время через Справка> Быстрый старт.

**Давайте создадим наш первый тест**

Ну что, раз мы зашли — давайте пробовать

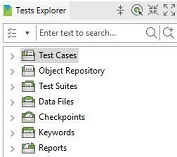
**Шаг #1: Создайте новый проект**

Перейдите в меню File-> New -> Create Project и введите название проекта в появившемся ниже окне:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio8.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Ваш проект откроется, и вы увидите следующие элементы, доступные в Test Explorer:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio9.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

**Шаг #2: добавление объектов в репозиторий**

Существует много подходов к созданию тест-кейса. Это может быть как запись вашего теста, так и генерация строк кода после добавления объекта. Давайте рассмотрим второй подход.

**Вот что нужно сделать:** Перейдите на вкладку «Object Repository», и для работы над веб-приложением, нажмите «Spy Web».

[http://www.software-testing.ru/images/stories/library/katalon/katalon-studio10-s.jpg](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio10.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Появится окно Object Spy. Введите стартовый URL и выберите, в каком браузере его открыть.

[http://www.software-testing.ru/images/stories/library/katalon/katalon-studio11-s.jpg](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio11.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

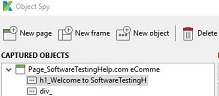
Это запустит URL в выбранном браузере. Я перехожу по «http://carguruji.com/shop» используя chrome.

Как вы можете видеть, каждый раз, когда вы наводите курсор мыши на объект на странице, он показывает способ захвата объекта:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio12.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Захватите все объекты, необходимые для теста, нажав «Alt + ~», теперь объекты будут отображаться в окне репозитория объектов под “Captured Objects”:

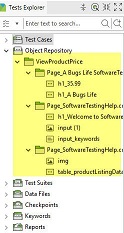
[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio13.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Затем нажмите на “Add to Object Repository”

http://www.software-testing.ru/images/stories/library/katalon/katalon-studio14.jpg

В итоге все необходимые для вашего теста объекты должны быть у вас в “Object Repository” структурированными по вашему выбору:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio15.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Теперь мы готовы добавить шаги к тесту.

**Шаг #3: Создаем тест-кейс**

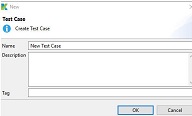
Я хочу, чтобы мой тест-кейс выполнял следующее:

* Запускал сайт
* Искал DVD
* Нажимал на DVD «Жизнь Жуков»
* Проверял, будет ли отображаться цена $35,99

Возможно, это не идеальный тест, но он подойдёт для демо-целей

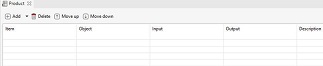
Итак, давайте создадим шаги в Katalon Studio.

Перейдите к Test Cases в Test Explorer. Щелкните правой кнопкой мыши и выберите  ‘*New ->Test Case*’. Введите имя тест-кейса и описание ниже. Я назову его «ViewProductPrice».

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio16.jpg)

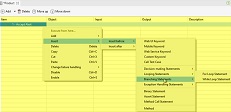
*Нажмите на картинку, чтобы увеличить изображение*

Откроется новый табличный формат:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio17.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Пользователям Selenium IDE покажется это довольно знакомым, так как выглядит оно очень похоже. Тем не менее, я нахожу это более близким по функционалу к HP UFT/QTP keyword view из-за возможности добавлять output variables, looping statements и т.д.

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio18.jpg)

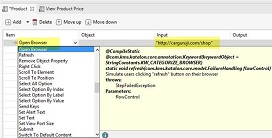
*Нажмите на картинку, чтобы увеличить изображение*

**Шаг #4: добавляем шаги тестирования**

Чтобы добавить  шаги тестирования нажмите «Add», и в создавшейся строке щелкните столбец «item», чтобы добавить ключевое слово / метод / действие, которое вы намереваетесь выполнить.

**Например:** Первое, что я хочу сделать, это «Открыть браузер» и перейти к определенному URL-адресу.

**Вот как вы можете это сделать:**

**[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio19.jpg)**

*Нажмите на картинку, чтобы увеличить изображение*

Итак, кликаем для выполнения соответствующих задач:

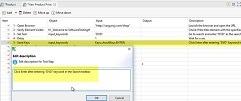
**Item**: выбрать, что делать

**Object**: где-на каком объекте

**Input**: что подавать на шаг

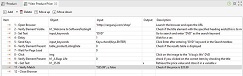
**Output**: где хранить возвращаемое значение

Аналогичным образом, продолжаем добавлять все шаги. Хорошая идея — добавить некоторое описание на каждый шаг, чтобы при будущем использовании тест-кейса чётко понимать, что тут и зачем. Для этого щелкните столбец описания и введите описание, как показано ниже:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio20.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

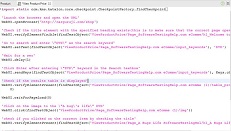
Вот так будет выглядеть ваш тест:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio21.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Как только готово — можете запускать.

Но сначала давайте я вам быстро покажу “Script view” данного скрипта. Нажмите на “Script” вкладку внизу страницы для активации.

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio22.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

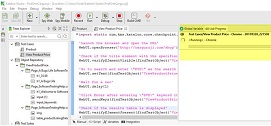
Пользователи, которым намного комфортнее работать с этим модом, могут использовать его вместо manual мода.

**Шаг #5: Запускаем ваш тест**

Для запуска вашего теста нажмите на “Run” в меню, так же вы тут можете выбрать браузер, в котором будет запускаться тест.

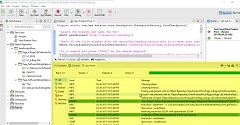
[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio23.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio24.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Результаты теста будут показаны как на примере ниже:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio25.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

**Шаг #6: для закрепления**

Теперь, когда вы выполнили свой первый тест, выполните следующие действия:

* Создайте больше тестов, добавьте их в тест-кейс и запустите их одновременно
* Просмотрите тест репорты, после выполнения тест-сьюта
* Параметрируйте ваши тесты
* Включите обработку ошибок
* Создайте кастомные ключевых слова /пакеты
* Вызывайте тесты и переиспользуйте код

**Новые возможности Katalon Studio**

**1) In-module помощь:** когда вы нажимаете на «?» в модуле тест-кейса, вы получаете страницу справки по документации, в которой отображается информация о ручном режиме тест-кейса. Аналогично, каждый модуль имеет удобную помощь в виде встроенного значка справки.

[http://www.software-testing.ru/images/stories/library/katalon/katalon-studio26-s.jpg](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio26.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

**2) Поддержка содержимого в режиме сценария:** я честно не знаю, что бы я делал, если бы IDE перестала подсказывать имена объектов и вызовы/синтаксис методов.

**3) Переименование тест-репортов:** если у нас нет сверхчеловеческой памяти, то узнать, какой тест прошел в какое время и какого числа будет довольно проблематично. В версии Katalon Studio 4.5 вы можете переименовать отчет о выполнении теста во что-то более человеческое и понятное.

**Что хорошего в Katalon?**

1) Простота установки. Работает c большинством операционных систем и браузеров: <https://docs.katalon.com/display/KD/System+Requirements>

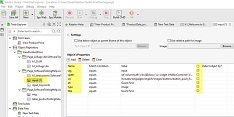
2) Web, Mobile, and Web Service тестирование — все в одном

3) Абсолютно бесплатное решение

4) Подходит как для опытных тестировщиков, так и для новичков

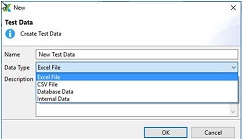
5) Отсутствует сложность установки и определения структуры как, например, в Selenium

6)Включает в себя надежное QTP-like UI/Object распознавание. Вы можете добавить несколько свойств для распознавания объекта, вместо того чтобы выбирать/угадывать, какой локатор работает лучше всего.

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio27.jpg)

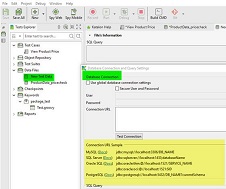
*Нажмите на картинку, чтобы увеличить изображение*

7) Тестовые данные могут быть созданы в самом инструменте или могут быть извлечены из Excel или базы данных.

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio28.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

Поддерживает следующие базы данных:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio29.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

8) Интегрируется с инструментами менеджмента требований и тестирования такими как JIRA и qTest. Отпадает нужда в тех. тул администраторах, так как интегрироваться довольно легко с помощью мастера быстрой настройки.

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio30.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

9) Выполнение нескольких тестов одновременно (как тест-сьют)

10) Интеграция с Git для управления исходным кодом и совместного использования

11) Надежная отладка и отчетность

12) Встроенная обработка ошибок. Больше никаких “try and catch” блоков, никаких сложных функций, чтобы аккуратно выйти из зафейленного скрипта. Определите, что должно произойти, когда ваш тест завершится с ошибкой в настройках проекта:

[](http://www.software-testing.ru/images/stories/library/katalon/katalon-studio31.jpg)

*Нажмите на картинку, чтобы увеличить изображение*

**Что можно было бы улучшить?**

Пожалуйста, обратите внимание:

* Пользовательский интерфейс слегка неуклюж
* Если в скрипт моде есть ошибка — вы не можете перейти в ручной режим