**Имя проекта: «Encode»**

**Суть проекта**: разработка инструмента, позволяющего оптимизировать кодирование и декодирование текстовой информации.

**Цели проекта:**

1. Исключение необходимости ручного кодирования и декодирования текста.
2. Сокращение времени работы с кодируемым или декодируемым текстом.

**Метрики достижения целей:**

1. Автоматическое преобразование исходного текста в закодированный формат и/или исходного текста в декодируемый формат
2. Сокращение времени работы с исходными и конечными данными за счёт автоматического преобразования.

**Риски:**

Высокая сложность безошибочного преобразования исходной информации ввиду иных кодировок.

**Уровень пользовательских требований:**

Программа рассчитана на одного человека. Имеется интерфейс, включающий в себя окно для ввода исходных данных, окно для вывода конечных данных, кнопки преобразования текста, меню с возможностью открывать и сохранять файлы. Кнопки переключения отвечают за формат кодирования: кнопка «Ascii code» отвечает за кодирование в Аски код, кнопка «SQCode» за кодирование в SQ код.

**Системные характеристики**

* СХ-1: Приложение является оконным.
* СХ-2: Для работы приложение использует компилятор Visual C++.
* СХ-3: Приложение устанавливается на Windows 7 и выше.
* СХ-4: Объём приложения не более 220 Мб.

**Пользовательские требования**  
ПТ-1: Запуск и установка приложения:

* ПТ-1.1: Запуск приложения производится с исполняемого файла Encode.exe
* ПТ-1.2: Остановка приложения производится с кнопки закрытия окна

ПТ-2: Конфигурирование приложения:

ПТ-2.1: Конфигурирование приложения сводится к указанию пути в файловой системе для открытия или сохранения.

ПТ-2.2: Целевой формат кодирования: Ascii-код и SQcode.

ПТ-3: Сохранение итоговых данных

ПТ-3.1: В процессе работы приложение должно сохранять итоговую информацию в указанный пользователем файл.

**Атрибуты качества**

* АК-1: Производительность

АК-1.1: Приложение должно обеспечивать хорошую скорость преобразования данных не менее 5 Мб/сек на аппаратном обеспечении, эквивалентном следующему: процессор i5, 8Гб оперативной памяти, средняя скорость чтения/ записи на диск 40Мб/сек.

* АК-2: Устойчивость к входным данным

АК-2.1: Поведение приложения в ситуации обработки файлов с нарушениями формата определено в ДС-2.

АК-2.2: Требования относительно размеров обрабатываемых файлов изложены в ДС-4.

**Ограничения**

* О-1: Приложение разрабатывается на языке программирование C++, использование которого обусловлено возможностью заказчика осуществлять поддержку приложения своими силами.
* О-2: Ограничения относительно версии и настроек компилятора Visual C++ отражены в пункте ДС-1 раздела «Детальные спецификации».
* О-3: Процедуры установки и настройки компилятора Visual C++ выходят за рамки данного проекта и не описываются в документации.

**Детальные спецификации**

* ДС-1: Компилятор Visual C++

ДС-1.1: Минимальная версия: 14.0.23918.0.  
ДС-1.2: Для работы приложение должна быть установлена Visual Studio 2022

* ДС-2: Параметры окна ввода

ДС-2.1: При отсутствии формата кодирования приложение не будет совершать дальнейших действий.

ДС-2.2: При открытии файла с кодировкой, отличной от UTF-8 UTF-16 приложение должно принимать за исходные данные только название файла.

ДС-2.3: При сохранении файла с расширением, отличающимся от TXT, DOCX приложение должно выдавать ошибку о невозможности сохранения файла.

* ДС-3: Сообщения

ДС-3.1: Сообщение о невозможности сохранить файл: «Не удалось сохранить файл».

* ДС-4: Форматы и размеры файлов:

ДС-4.1: Приложение должно обрабатывать текстовые файлы на русском и

английском языках в следующих исходных кодировках: UTF-16, UTF-8.

Обрабатываемые файлы могут быть представлены в следующих форматах,

определяемых расширениями файлов:

Plain Text (TXT), Word Document (DOCX)

ДС-4.2: Приложение должно обрабатывать файлы размером до 30 МБ (вклю-

чительно), игнорируя любой файл, размер которого превышает 30 МБ.

Ответы на контрольные вопросы

1. Объясните понятие «жизненный цикл программного обеспечения».

* период времени, который начинается с момента принятия решения о необходимости создания программного продукта и заканчивается в момент его полного выхода из эксплуатации.

1. Какую документацию нужно тестировать?

* Документация, выполненная в строгом соответствии со стандартами (ГОСТ, ISO): техническим заданием (ТЗ), спецификациями, требованиями к программному обеспечению, инструкциями, и т. д.
* Документация, выполненная по внутренним правилам команды или заказчика (документация, основанная на стандартах, сгенерированная документация).

1. На каком этапе ЖЦ разработки нужно тестировать документацию?

* Допускается создание документации на каждом этапе жизненного цикла.

1. Из каких шагов состоит тестирование документации?

* Определить объем и вид документов, требующихся для конкретного вида жизненного цикла разработки и выбранной методологии разработки.
* Определить цели тестирования документации.
* Выбрать существующие документы и провести их анализ одним из выбранных способов.
* Оценить документацию по критериям, сформировать отчет о тестировании документации.

1. Когда тестирование документации оправдано?

* Цель- ускорение процедуры тестирования. От детальности и точности описания функциональности зависит то, как быстро тестировщик сможет выполнить тестирование.
* Цель- использование полной документации в качестве источника для создания тестовой документации и проектирования тест-планов.
* Цель- сокращение затрат на техническую поддержку.

ЛАБОРАТОРНАЯ РАБОТА №2

Работа с классификацией видов тестирования

Цель работы: Применить виды тестирования к своему проекту

**Проект: Encode**

**Модульное тестирование:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Модуль | | Функц. тест | Юза-ити | «+» тест | «-» тест | Чёрный ящик | Надежность |
| Ввод данных | | + | + | Буквы +  Цифры + | Спец.символы +  NULL + | + |  |
| Вывод данных | | + | + |  |  | + |  |
| Кнопки | Кодирование | + | + | Ascii-code + SQCode + |  | + |  |
| Декодирование | + | + | Ascii-code + SQCode + |  | + |  |
| Открыть | + | + | Txt+  docx + |  | + | + |
| Сохранить | + | + | Txt+  docx + | + | + | + |
| Ascii-code | + | + |  |  | + |  |
| SQCode | + | + |  |  | + |  |

**Интегрированное тестирование:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Шаг | Функц. тест | Юза-ити | Поз.тест | Негатив.тест | Чёрный ящик | Надежность |
| Открыть+Ввод данных | + | + | txt+  docx + | + | + | + |
| Вывод данных +Сохранить | + | + | txt+  docx + | + | + | + |
| Кодирование+ Ascii | + |  |  |  | + |  |
| Кодирование+ SQCode | + |  |  |  | + |  |
| Декодирование+ Ascii | + |  |  |  | + |  |
| Декодирование+ SQCode | + |  |  |  | + |  |

**Системное тестирование:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Инсталляция | Совместимость | Конфигурация | Приём. тест |
| Encode | Интерфейс +  Стабильность работы + | Windows 7 +  Windows 8 +  Windows 10 +  Windows 11 + | Процессор i5+  Скорость чтения 40Мб/с +  8 Гб оперативной памяти + | + |

**Тип тестирования:** функциональный:

**Метод тестирования:** статический (программный код, документация).

**Виды тестирования:** по объекту (функциональное, инсталляция, ресурсные тесты, юзабилити, совместимость), по признаку позитивности, по степени автоматизации (ручное), по времени проведения (приёмочное тестирование).

**По степени автоматизации: ручное тестирование**

Ответы на контрольные вопросы:

1. Определяется ли качество ПО качеством программного кода?

* Качество программного кода может быть оценкой качества только в статическом методе тестирования. В остальных случаях разработчик понимает качество по-своему.

1. Какие существуют виды тестирования?

* Функциональное, ручное, автоматизированное, полуавтоматизированное, позитивное, негативное,альфа, smoke-тестирование, приемочное тестирование, регрессивное тестирование.

1. Какие существуют типы тестирования?

* Функциональный (что система делает), нефункциональный (как система это делает)

1. Какие существуют методы тестирования?

* Статические (то, что проверяется без непосредственного выполнения кода), динамические (функциональность системы).

1. Какие существуют уровни тестирования?

* Компонентный, интеграционный, системный уровни тестирования.

ЛАБОРАТОРНАЯ РАБОТА № 3

Cоздание тестовой документации.

Цель: Проверить сайт по написанному тест-сьюту

**Сайт: sokolov.ru**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Автор | Приоритет | Заголовок | Список тест-кейсов |
| 11 | Юлия | 1 | Модуль пользователя | * 1. Авторизация   2. Редактирование профиля |
| 22 | Юлия | 1 | Модуль товаров | * 1. Получение новинок   2. Получение товаров в категории   3. Получение товара |
| 33 | Юлия | 1 | Модуль корзины | * 1. Добавление товара в корзину   2. Увеличение количества товара в корзине   3. Уменьшение количества товара в корзине |
| 44 | Юлия | 1 | Модуль оформления заказа | * 1. Ввод промокода   2. Оформление заказа   3. Оформление заказа |

Таблица 1. Тест-сюьт

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Описание  (Тип) | Предусловия | Шаги | Ожидаемый  результат | Фактический результат |
| 1.1 | Авторизация (позитивный) | 1. Пользователь находится на странице входа в личный кабинет 2. Пользователь был ранее зарегистрирован в системе | 1. Ввести в поля «Логин» и «Пароль» логин и пароль пользователя 2. Нажать кнопку «Войти» | 1.Пользователь перенаправляется на страницу личного кабинета | 1. + |
| 1.2 | Редактирование профиля (негативный) | 1. Пользователь находится на странице личного кабинета | 1. Нажать на пункт   «Профиль»   1. Получить страницу редактирования профиля 2. Нажать на кнопку   «Изменить»   1. Ввести в поля   «Пароль» и «Подтвердите пароль» разные значения   1. Нажать на кнопку   «Сохранить» | 1. Пользователю отображается сообщение «Пароли не совпадают» | 1. + |
| 2.1 | Получение новинок (Позитивный) | 1. Пользователь находится на одной из страниц магазина | 1. Пользователь нажимает на название интернет-магазина, расположенного в шапке сайта | 1. Пользователю отображается главная страница магазина со списком новинок.  2. Пользователю отображается главная страница магазина выгодными предложениями. | 1. + 2. + |
| 2.2 | Получение товаров в категории (Позитивный) | 1. Пользователь находится на   Одной из  следующих страниц: на главной странице, на странице  «Корзина»,на странице входа в личный кабинет, на странице регистрации, на странице каталога  товаров,на странице товара | 1. Нажать на пункт меню категорий | 1. Пользователь перенаправляется на страницу категорий  2. На странице категорий отображается заголовок, совпадающий с названием ранее выбранного пункта меню | 1. + 2. + |
| 2.3 | Получение товара (Позитивный) | 1.Пользователь находится на  одной из  следующих страниц: на главной странице, на странице  «Корзина»,на странице категорий  2.На данной странице  отображается товар. | 1. Нажать на кнопку  «Товар» | 1.Пользователь перенаправляется на страницу товара  2. Информация о товаре на странице товара должна совпадать с  информацией на странице, с которой был осуществлен переход. | 1. + 2. +   (Рис. 1-2) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Описание (Тип) | Шаги | Шаги | Ожидаемый результат | Фактический результат |
| 3.1 | Добавление товара в корзину (Позитив ный) | 1. Пользователь находится на одной из следующих страниц: на главной странице, на странице категорий, на странице товара 2. На данной странице отображается товар | 1. Нажать на кнопку «Добавить в корзину» | 1. Страница, на которой находится пользователь, обновляется 2. Пользователю отображается сообщение «Товар в корзину успешно добавлен» 3. При переходе на страницу «Корзина» отображается добавленный товар | 1. - 2. + 3. +   (Рис. 4) |
| 3.2 | Увеличен ие количест ва товара в корзине (Позитивный) | 1. Пользователь находится на странице «Корзина» | 1. Нажать на кнопку «+» | 1. В колонке «Кол- во» в строке товара увеличивается значение на 1 | 1. + |
| 3.3 | Уменьше ние количест ва товара в корзине (Негативный) | 1. Пользователь находится на странице «Корзина» 2. Значение в колонке «Кол- во» в строке товара равно 1 | 1. Нажать на кнопку «-» | 1. Товар убирается из «Корзины»  2. Пользователю доступна кнопка «Отменить удаление». | 1. + 2. + |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Описание (Тип) | Шаги | Шаги | Ожидаемый результат | Фактический результат |
| 4.1 | Ввод промокода. (Позитив ный) | 1. Пользователь переходит к оформлению заказа. 2. Пользователь вводит промокод в поле «Введите промокод». | 1. Нажать на кнопку «Применить» | 1. Введенный промокод сохраняется и отображается сверхку поля «Введите промокод». 2. Сумма к оплате/ сумма баллов меняется в окне «К оплате». | 1. + 2. +   (Рис. 3) |
| 4.2 | Оформление заказа(Негативный) | 1. Пользователь не выбирает способ получения заказа  2. Пользователь не указывает адрес доставки | 1. Нажать на кнопку « Оформить заказ », не выбирая способ получения из пункта «Доставка».  2. Выбрать способ получения « Курьер» в пункте «Доставка», не указывать адрес, нажать на кнопку «Сохранить адрес». | 1. У пользователя появляется сообщение: « Для оформления заказа добавьте адрес доставки».  2. Окошко с пустым адресом светится красным, появляется сообщение «Уточните адрес». | 1. + 2. + |
| 4.3 | Оформление заказа (Позитивный) | 1. Пользователь находится на странице оформления заказа. 2. Пользователь заполняет необходимые данные для оформления заказа: доставка, оплата, покупатель. | 1. Нажать на кнопку «Оформить заказ». | 1.Появляется сообщение «Заказ оформлен».  2. Пользователь переходит на страницу для отслеживания статуса заказа.  3. Товар убирается из «Корзины». | 1. + 2. + 3. + |

Таблица 2. Тест-кейсы

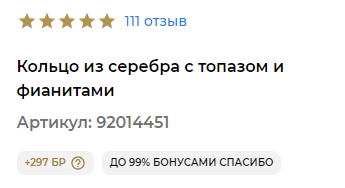
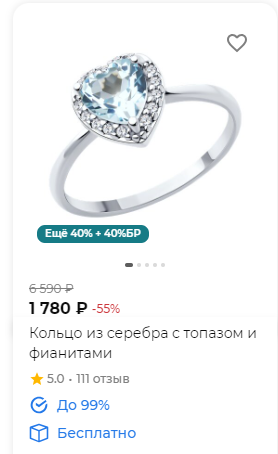


Рис. 1-2 – дополнение к п. 2.3 тест-сьюта



Рис. 3 – дополнение к п. 4.1 тест-сьюта

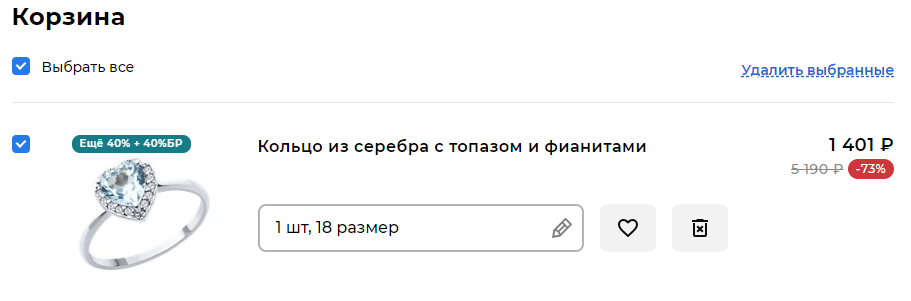


Рис. 4 – дополнение к п. 3.1 тест-сьюта

**Ответы на контрольные вопросы:**

1. Приведите пример негативных тест-кейсов для трех видов тестирования.

* Ввод пустых значений, ввод отличных друг от друга данных, нажатие кнопки до заполнения данных.

1. Перечислите требования к тест-плану.

* – режимы тестирования, график тестирования, критерии начала тестирования, критерии окончания тестирования, окружение тестируемой системы, необходимое оборудование и программные средства, возможные риски и пути их решения.

1. Перечислите требования к тест-сьютам.

* ID – идентификатор,
* Автор,
* Приоритет выполнения,
* Краткое описание,
* Список тест-кейсов.

1. Перечислите требования к тест-кейсам.

* Уникальный идентификационный номер. По этому значению на тест-кейс будут ссылаться из других документов.
* Краткое описание.
* Пошаговые мероприятия.
* Ожидаемый результат.
* Действительный (фактический) результат.

1. Какова связь этапа жизненного цикла разработки программного обеспечения и вида тестовой документации?

* Внесение ясности для заказчика. Общение с другими командами, понимание желаний заказчика напрямую влияют на качество тестовой документации и в дальнейшем на ПО.

ЛАБОРАТОРНАЯ РАБОТА № 4

Методы тест-дизайна.

Цель: Разработать тест-сьют для своего проекта.

**Форма авторизации сайта: rutube.ru**

**Позитивные классы эквивалентности:**

1. Корректные данные:

* Правильный адрес электронной почты (текстовый формат)
* Правильный пароль (текстовый формат, не менее 8 символов, использование верхнего регистра)

**Негативные классы эквивалентности:**

1. Некорректные данные:

* Неверный адрес электронной почты
* Неверный пароль (не соответствует минимальной длине, не имеет букв верхнего регистра).

1. Пустые поля
2. Специальные символы

* Пробелы, символы, не разрешенные в email.
* Пробелы, символы, не разрешенные в пароле.

1. Граничные значения:

* Электронная почта превышает максимальную длину

**Расчет количества тестов:**

|  |  |
| --- | --- |
| Поле «Эл. Почта» | Поле «Пароль» |
| ihavebeenworking@mail.ru + | IFOns82kfjpw + |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ + (Введите почту) | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_+ (введите пароль) |
| @@com.ru + (Введите правильную почту) | Ifj3 + (Пароль должен состоять минимум из 8 символов) |
|  | 5iehsk75h + (Пароль должен содержать буквы верхнего регистра) |

Как результат, количество тестов по данному методу составит 7.

**Ответы на контрольные вопросы:**

1. Опишите методику выделения эквивалентных классов.

* Основная идея заключается в том, чтобы разделить все возможные входные данные на группы (классы), которые обрабатываются системой одинаково и приводят к одинаковым результатам. Это позволяет сократить количество тестов, необходимых для проверки функциональности системы.

1. В чем цель тестирования граничных значений?

* Цель тестирования граничных значений заключается в выявлении ошибок, которые могут возникнуть на границах допустимых диапазонов входных данных.

1. Что такое методика черного ящика?

* Подход к тестированию программного обеспечения, при котором тестировщик не имеет доступа к внутренней структуре или коду приложения.

1. В чем разница между методикой черного, белого и серого ящиков?

* Разница между методиками черного, белого и серого ящиков заключается в доступе к внутренней структуре системы и подходах к тестированию.

1. Что представляет собой тест-дизайн?

* Процесс разработки и создания тестов для оценки функциональности и качества программного обеспечения.

# ЛАБОРАТОРНАЯ РАБОТА №5

# Ручное тестирование

Цель: Провести ручное тестирование по проекту.

**Проект: Encode**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Автор | Приоритет | Заголовок | Список тест-кейсов |
| 11 | Юлия | 1 | Модуль ввода данных | * 1. Ввод данных вручную   2. Ввод данных вручную   3. Открытие файла |
| 22 | Юлия | 1 | Модуль преобразования данных | * 1. Кодирование данных   2. Декодирование данных |
| 33 | Юлия | 1 | Модуль сохранения данных | * 1. Копирование данных   2. Сохранение данных |

Таблица 1 – Тест-сьют проекта

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Описание  (Тип) | Предусловия | Шаги | Ожидаемый  результат | Факт. результат |
| 1.1 | Ввод данных вручную  (позитивный) | 1.Пользователь обращается к окну ввода данных в программе | 1. Ввести набор символов (цифры, буквы, пробелы). | 1. В окне ввода отображается введенная пользователем информация. | 1. + |
| 1.2 | Ввод данных вручную  (негативный) | 1.Пользователь обращается к окну ввода данных в программе | 1. Ввести набор специальных символов. | 1. В окне ввода отображается введенная пользователем информация. | 1. + |
| 1.3 | Открытие файла (Позитивный) | 1.Пользователь обращается к меню, к пункту «Открыть». | 1. Пользователь открывает в окне обзора любой текстовый файл формата txt, docx. | 1. В окне ввода отображается прочитанная из файла информация. | 1.+ |
| 2.1 | Кодирование данных (Позитивный) | 1. Пользователь выбирает одну из кодировок: Ascii, SQCode. | 1. Нажать на кнопку «Encode» | 1. В окне вывода данных появляется информация в указанной пользователем кодировке.  2. В окне ввода данных сохраняется исходная информация. | 1. +  2. + |
| 2.2 | Декодирование данных (Позитивный) | 1.Пользователь выбирает одну из кодировок: Ascii, SQCode. По очереди. | 1. Нажать на кнопку «Decode» | 1. В окне вывода данных появляется информация в понятной для человека форме.  2. В окне ввода данных сохраняется исходная закодированная информация. | 1. +  2. +  3. +  4. + |
| 3.1 | Копирование данных (позитивный) | 1. Пользователь обращается к одному из окон программы: окно ввода, окно вывода. | 1. Выделить фрагмент данных.  2. Нажать ПКМ–>Копировать | 1. В буфер обмена попадает фрагмент данных, выделенной пользователем. | 1.+ |
| 3.2 | Сохранение данных (позитивный) | 1. Пользователь обращается к меню, к пункту «Сохранить». | 1. Выбрать место сохранения файла.  2. Выбрать расширение файла txt или docx. | 1. В окне вывода данных остаются данные.  2. По выбранному месту находится сохраненный файл заданного расширения.  3. Сохраненный файл открывается и отображает данные из окна вывода. | 1. +  2. +  3. + |

Таблица 2 – Тест-кейсы проекта

**Ответы на контрольные вопросы:**

1. Какие виды тестирования можно выполнять в ручном режиме?

- Модульное, интеграционное, системное тестирование, приёмочное тестирование, юзабилити-тестирование.

1. Какая документация участвует в ручном тестировании?

- Тест-план, тест-сьют.

1. Может ли существовать программная система, свободная от дефектов?

- Нет, ввиду многих факторов: ограниченность ресурсов, ошибки в коде, изменение требований, недостаточное тестирование.

1. Каковы требования к описанию дефекта?

- Наличие идентификатора дефекта, его описание, шаги для выполнения, ожидаемый и фактический результаты, приоритет, указание среды тестирования, а также подкрепление описания скриншотами или логом.

1. Каков жизненный цикл дефекта?

- Новый (тестировщик фиксирует дефект), назначен (дефект назначается разработчику для исправления), открыт (разработчик начинает работать над дефектом), исправлен (разработчик вносит изменения и тестирует их), к повторному тестированию (исправленный дефект возвращается к тестировщику), проверен (проверка тестировщиком устранения дефекта), закрыт (исправление подтверждено).

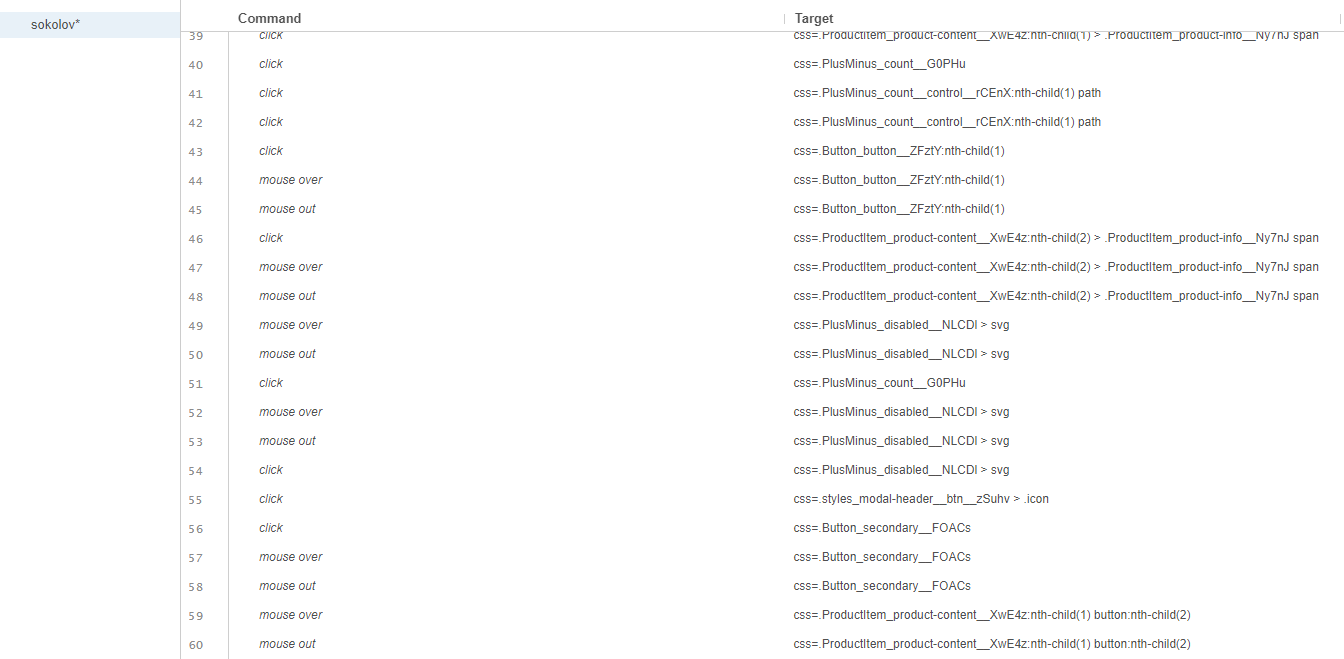
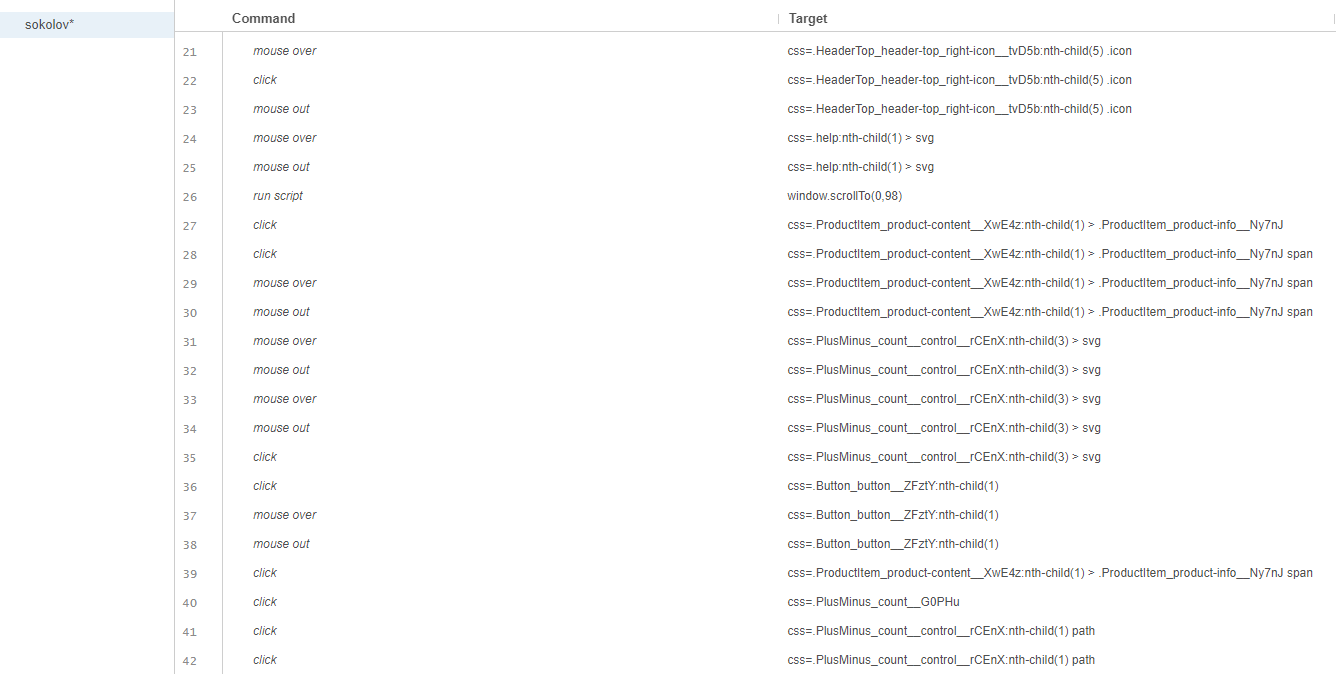
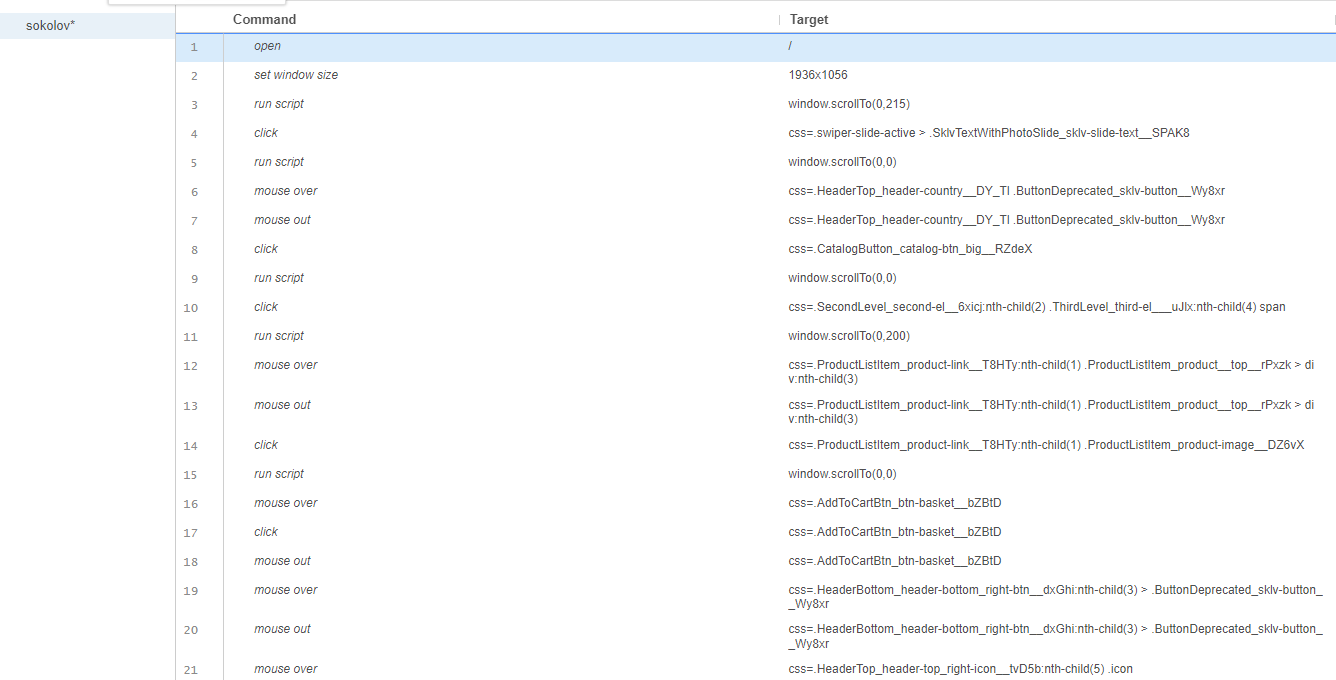
# ЛАБОРАТОРНАЯ РАБОТА №6

# Автоматизация тестирования

Цель: Провести автотест для сайта Sokolov

Расширение браузера: selenium ide

В автотестах были проведены тесты из тест-сьюта, созданного в предыдушей лабораторной работе. В проект записываются все действия системы на работу пользователя.



**Ответы на контрольные вопросы:**

1. Каковы цели использования автоматизированного тестирования?

* сокращение времени тестирования, увеличение охвата тестирования,исключение человеческого фактора,оптимизация затрат

1. В каких случаях можно использовать автоматизированное тестирование?

* когда проект большой и требует регулярного выполнения множества тестов, Для тестирования производительности, где требуется симуляция большого количества пользователей или нагрузок на систему,В проектах с высокими рисками, где важно минимизировать человеческий фактор и повысить надежность тестирования

1. Каковы недостатки автоматизированного тестирования?

* высокие первоначальные затраты, временные затраты на разработку, необходимость регулярного обновления

1. Что такое локатор?

* элемент в автоматизации тестирования, который используется для нахождения интерфейсных элементов (например, кнопок, полей ввода) на веб-странице или в приложении. Локаторы помогают инструментам автоматизации взаимодействовать с этими элементами для выполнения действий (например, кликов или ввода данных) во время тестирования.

1. Каковы цели нагрузочного тестирования?

* проверка производительности приложения, оценка масштабируемости

# ЛАБОРАТОРНАЯ РАБОТА №7

# Функциональные и нефункциональные требования

Цель: Заполнить таблицы с функциональными и нефункциональными требованиями.

Таблица 1. О сайте «Sokolov»

|  |  |
| --- | --- |
| **Функциональные требования** | **Нефункциональные требования** |
| Аутентификация пользователей: сайт позволяет пользователям создавать учётные записи, входить в систему, получать код подтверждения | Производительность: сайт должен загружать веб-страницу менее, чем за 4 секунды, даже с учетом одновременного использования 1000 пользователями |
| Работа с корзиной покупок: сайт должен позволять пользователям добавлять товары в корзину, просматривать, изменять, удалять её содержимое, переходить к оформлению заказа. | Безопасность: сайт должен соответствовать отраслевым стандартам безопасности и протоколам шифрования |
| Работа с оформлением заказа: сайт должен позволять пользователю использовать промокоды на скидку, выбирать способ получения (доставка или самовывоз). После оформления заказа пользователю должно быть позволено посмотреть статус заказа, а также предоставлена возможность отмены заказа или переноса даты доставки. | Пользовательский интерфейс: сайт должен загружать веб-страницу полноценно на различных устройствах и системах |
| Функциональность поиска: сайт должен предоставлять пользователю функцию поиска, которая позволяет находить продукты по ключевым словам. | Обслуживание: сайт всегда должен обслуживаться технически |
| Функциональность фильтра: сайт должен предоставлять пользователю функцию сортировки, которая позволяет находить лекарства по выбранным фильтрам. | Доступность: сайт должен быть доступен пользователям с ограниченными возможностями |

Таблица 2. О вашем проекте по Базе Данных «Аптечная сеть»

|  |  |
| --- | --- |
| **Функциональные требования** | **Нефункциональные требования** |
| Аутентификация пользователей: система позволяет работникам, входить в систему (для фармацевтов: по номеру сотрудника и паролю, для администраторов: по логину и паролю) | Производительность: система должна обрабатывать запросы менее, чем за 2 секунды, даже с учетом одновременного использования 3 фармацевтами |
| Управление лекарствами: система должна позволять фармацевтам добавлять принятые лекарства в корзину, просматривать, изменять количество, заказывать отсутствующие позиции. | Безопасность: система должна соответствовать отраслевым стандартам безопасности и протоколам шифрования |
| Обработка заказов: система должна позволять фармацевтам выбирать адрес аптеки, службу доставки и количество заказываемых позиций. После оформления запроса фармацевту должно быть позволено посмотреть статус, а также предоставлен номер сотрудника службы доставки, который будет доставлять заказанные позиции в аптеку. | Обслуживание: система должна обслуживаться технически в случае изменения аппаратного обеспечения в аптеке или запроса от фармацевтов или администраторов |
| Функциональность поиска: система должна предоставлять фармацевту функцию поиска, которая позволяет находить лекарства по ключевым словам. | Надежность: минимальное время простоя системы, высокая доступность |
| Функциональность фильтра: система должна предоставлять фармацевту функцию сортировки, которая позволяет находить лекарства по выбранным фильтрам. |  |

**Ответы на контрольные вопросы:**

1. Каковы цели выявления функциональных требований?

- Для определения требуемых от системы функций, для оценки объема и ресурсов, обеспечение четких критериев тестирования

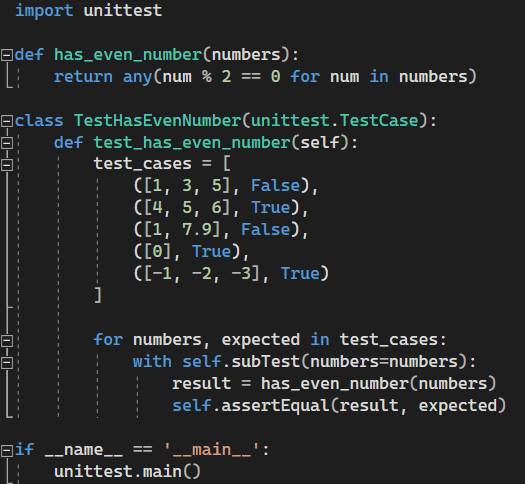
2. Каковы цели выявления нефункциональных требований?

- Для определения атрибутов качества системы, улучшения проектирования, обеспечения основы для тестирования, гарантирующие, что все аспекты качества будут проверены.

# ЛАБОРАТОРНАЯ РАБОТА №8

# Типы вопросов при выявлении требований

Цель: Создать автотест, дописать уточненные требования.



Данный автотест будет вводить поочередно наборы чисел с четными/нечетными значениями и ноль. В случае, если программа правильно определит наличие чётных чисел, автотест пройден. Результат автотеста говорит о том, что программа написана правильно.

|  |  |
| --- | --- |
| Открытые вопросы | Закрытые вопросы |
| И: Сколько чисел будет входить в вводимый набор?  З: Около 30 символов | И: В наборе чисел будут отрицательные?  З: Да |
| И: Сколько пользователей могут иметь одновременный доступ к программе?  З: Максимальное число пользователей- 4 человека | И: В наборе чисел все числа- целые?  З: Да |
| И: Какой диапазон чисел будет вводиться?  З: От -1000 до 1000 |  |
| И: Как происходит процесс ввода чисел?  З: Набор вводится в консоль с клавиатуры |  |
| И: Как на ваш взгляд должны сохраняться введенные данные?  З: Они должны сохраняться в лог-файле, вместе с результатами программы |  |