**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Сибирский государственный университет науки и технологий**

**имени академика М.Ф. Решетнева»**

Институт информатики и телекоммуникаций

Кафедра информационно-управляющих систем

ЛАБОРАТОРНАЯ РАБОТА №6

**«Тестовые сценарии системы»**

Руководитель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Доррер М.Г.

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка, дата)

Разработал:

Студент группы МПА21-01

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сацук М.М.

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата сдачи)

Красноярск, 2021

Test case (тест-кейс) – Это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или ее части. Более строго — формализованное описание одной показательной проверки на соответствие требованиям прямым или косвенным. Обязательно должен содержать следующую информацию:

* Идея проверки
* Описание проверяемого требования или проверяемой части требования
* Используемое для проверки тестовое окружение
* Исходное состояние продукта перед началом проверки
* Шаги для приведения продукта в состояние, подлежащее проверке
* Входные данные для использования при воспроизведении шагов
* Ожидаемый результат
* Прочую информацию, необходимую для проведения проверки

Таблица 1. Тест-кейс № A.1. - Проверка инвертирования произвольной строки

|  |  |  |
| --- | --- | --- |
| **Название:** | A.1. Проверка инвертирования произвольной строки | |
| **Функция:** | Инвертирование входной строки | |
| **Предусловие:** | ПО запущено. Открыто главное меню. | |
| **Шаги теста:** | **Ожидаемый результат** | **Результат теста:** |
| 1. Нажатие на кнопку «Lab #3» | Открылась форма (фрагмент) третьей работы |  |
| 1. Установка значение в поле «EditTextInputStroke» на “Auto-test” | В поле ввода записана строка “Auto-test” (без лишних пробелов, символов и т.п.) |  |
| 1. Нажатие на кнопку «Инвертировать» | Вызов действия кнопки установило обратное входное значение в текстовом поле «TextViewResultValue» |  |
| 1. Проверка соответствия данных из поля «TextViewResultValue» с заданной константой | Выходная строка соответствует заданной проверочной константе “tset-otuA” |  |
| 1. Нажатие на кнопку «ПерейтиВверх» | Корректное закрытие действующего фрагмента и открытие главного окна программы |  |

Таблица 2. Тест-кейс № A.2. – Проверка корректности финансовых расчётов

|  |  |  |
| --- | --- | --- |
| **Название:** | A.2. Проверка корректности финансовых расчётов | |
| **Функция:** | Вычисление прибыли | |
| **Предусловие:** | ПО запущено. Открыто главное меню. | |
| **Шаги теста:** | **Ожидаемый результат** | **Результат теста:** |
| 1. Нажатие на кнопку «Lab #5» | Открылась форма (фрагмент) пятой работы |  |
| 1. Установка значение в поле «EditTextNumberFinResult» на “35” | В поле ввода записана строка “35” (без лишних пробелов, символов и т.п.) |  |
| 1. Установка значение в поле «EditTextNumberFinCost» на “3” | В поле ввода записана строка “3” (без лишних пробелов, символов и т.п.) |  |
| 1. Выбор радио-кнопки под номером 2 | Выбрана только одна (вторая) радио-кнопка, остальные кнопки не задействованы |  |
| 1. Нажатие на кнопку «Расчет» | Вызов действия кнопки установило результат вычислений в текстовом поле «TextViewResultCalculate» |  |
| 1. Проверка соответствия данных из поля «TextViewResultCalculate» с заданной константой | Выходная строка соответствует заданной проверочной константе “26.56” |  |
| 1. Нажатие на кнопку «ПерейтиВверх» | Корректное закрытие действующего фрагмента и открытие главного окна программы |  |

Таблица 3. Тест-кейс № A.3. – Проверка корректности выдаваемого результата при заданном числе.

|  |  |  |
| --- | --- | --- |
| **Название:** | A.3. Проверка корректного результата при заданном возрасте | |
| **Функция:** | Отображение сообщения, соответствующего введенному возрасту | |
| **Предусловие:** | ПО запущено. Открыто главное меню. | |
| **Шаги теста:** | **Ожидаемый результат** | **Результат теста:** |
| 1. Нажатие на кнопку «Lab #1» | Открылась форма (фрагмент) пятой работы |  |
| 1. Установка значение в поле «EditTextNumberEditAget» на “15” | В поле ввода записана строка “15” (без лишних пробелов, символов и т.п.) |  |
| 1. Нажатие на кнопку «Кликни на меня» | Вызов действия кнопки установило результат вычислений в текстовом поле «TextViewTextResult» |  |
| 1. Проверка соответствия данных из поля «TextViewTextResult» с заданной константой | Выходная строка соответствует заданной проверочной константе “Вы слишком молоды!” |  |
| 1. Нажатие на кнопку «ПерейтиВверх» | Корректное закрытие действующего фрагмента и открытие главного окна программы |  |
| 1. Нажатие на кнопку «Exit» | Корректное закрытие ПО без ошибок |  |

**Автоматизированное тестирование Android приложения**

**Ranorex Studio** - это платформа автоматизации тестирования графического интерфейса, предоставляемая компанией по разработке программного обеспечения Ranorex GmbH. Фреймворк используется для тестирования настольных, веб-и мобильных приложений.   
Функции Ranorex Studio:

* Дизайн различных навыков, ролей и потребностей, автоматическая структура функции Ranorex, которая позволяет группам QA работать лучше всего.
* Ranorex Studio поддерживает проверку в ключе, вы можете обеспечить, чтобы каждый пользователь Лучше всего использовать навыки.
* Снижение уровня обучения с помощью интуитивно понятного программного обеспечения автоматизации тестирования, которое соответствует различным уровням навыков, с инструментами, не имеющим инструкций для начала, а также инструментами на основе кода для создания расширенного автоматического теста.
* Архив объектов Ranorex позволяет легко управлять всеми элементами пользовательского интерфейса. В этом месте.
* внимательно отслеживая влияние изменений программного обеспечения в системе путем интегрирования автоматического тестирования в среде CI/CD.
* Построение стабильного, надежного автоматического тестирования, которое обеспечивает быструю обратную связь от процесса тестирования и имеет низкие затраты на обслуживание-освобождает вас от низкоуровневых ручных тестов, перебор для более сложной реализации проверки приложения.

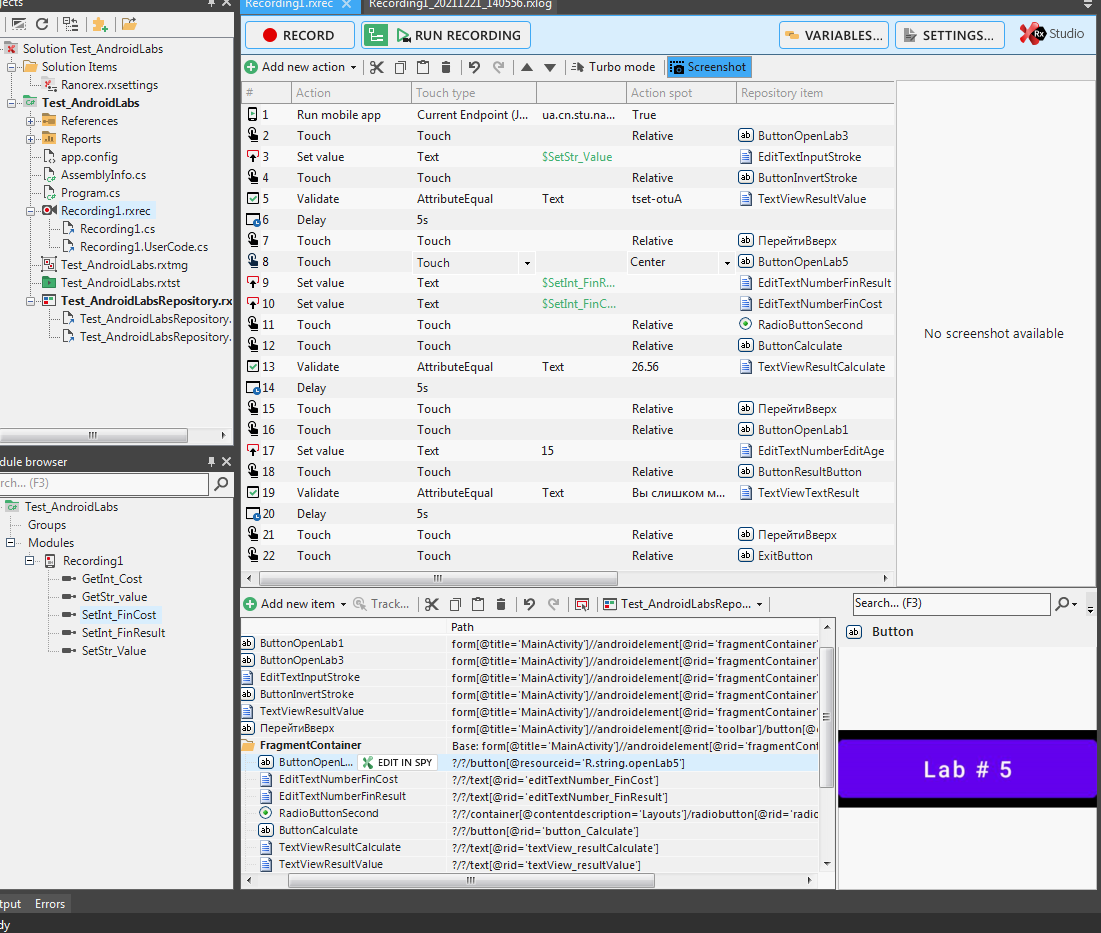


Рисунок 1 – Созданный алгоритм действий тестирования

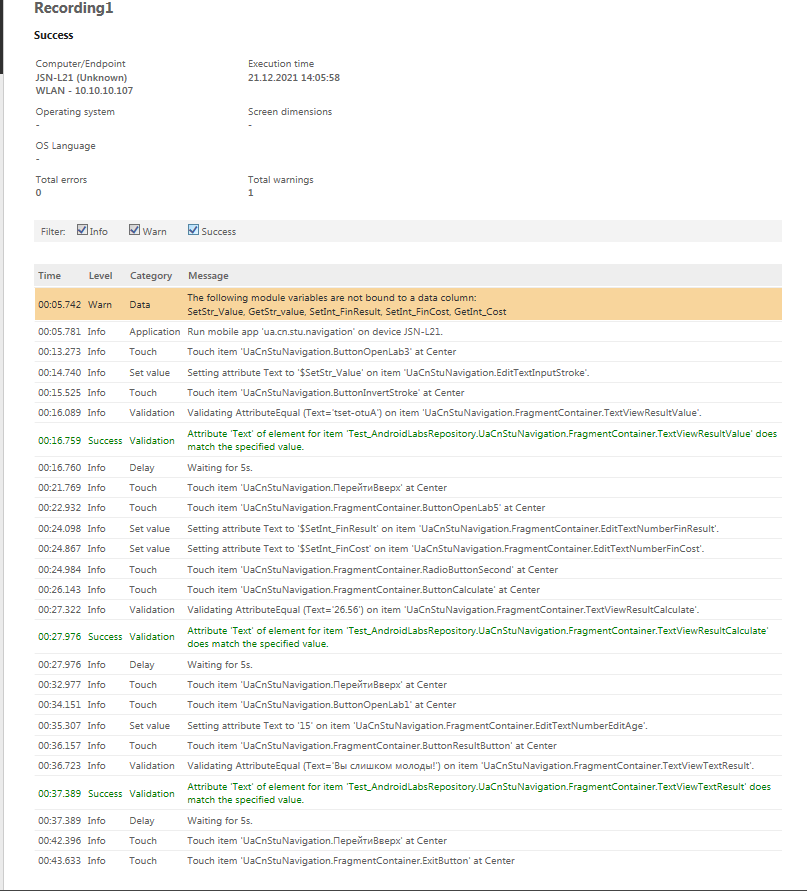


Рисунок 2 – Сгенерированный отчет пройденного тестирования