# Лабораторная работа №8 Автоматизация тестирования пользовательского интерфейса ПО

1. Цель работы
   1. Изучение механизмов автоматизации тестирования пользовательского интерфейса.
2. Литература
   1. Игнатьев, А. В. Тестирование программного обеспечения : учебное пособие для вузов / А. В. Игнатьев. — 4-е изд., стер. — Санкт-Петербург : Лань, 2025. — 56 с. — ISBN 978-5-507-50858-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/481331> (дата обращения: 21.05.2025). — Режим доступа: для авториз. пользователей.
3. Подготовка к работе
   1. Повторить теоретический материал (см. п.2).
   2. Изучить описание лабораторной работы.
4. Основное оборудование
   1. Персональный компьютер.
5. Задание
   1. Создать новый проект WPF, в котором реализовать простое окно авторизации с 2 полями ввода. При авторизации пользователя должен осуществляться переход на другую страницу с сообщением «Добро пожаловать, ЛОГИН, ваша роль - РОЛЬ». При некорректном вводе выводить сообщение об ошибке.

Список логинов, паролей и ролей пользователей хранить локально в коде (например, с помощью статического поля)

* 1. Создать новый проект типа проект тестирования xUnit, установить в нем пакеты FluentAssertions, FlaUI.Core, FlaUI.UIA3, загрузить последнюю версию программы FlaUIInspect из репозитория <https://github.com/FlaUI/FlaUInspect>
  2. Настройка тестов интерфейса
     1. В тестовом классе добавить два поля

private Application \_app;

private UIA3Automation \_automation;

* + 1. Задать элементам управления на форме свойство AutomationProperties.AutomationId
    2. В тесте инициализировать значения \_app и \_automation

\_app = Application.Launch("путь к exe");

\_automation = new UIA3Automation();

* + 1. Получить главное окно приложения

var window = \_app.GetMainWindow(\_automation);

* + 1. Получить необходимые элементы управления

var input = window.FindFirstDescendant(cf => cf.ByAutomationId("Имя")).AsTextBox();

* + 1. Взаимодействуя с полями ввода выполнить тест ПО.

При необходимости для анализа программы во время выполнения, уточнения наименований окон и другой информации используйте визуальное древо FlaUIInspect.

* 1. Разработать не менее 5 тестов для приложения, для наиболее полного покрытия различных сценариев использования (позитивных и негативных)
  2. Добавьте сохранение скриншотов программы во время выполнения тестов при помощи

var screenshot = window.Capture();

screenshot.Save("screenshot.png");

1. Порядок выполнения работы
   1. Повторить теоретический материал п. 3.1;
   2. Выполнить тестирование интерфейса ПО п. 5.1-5.7;
   3. Ответить на контрольные вопросы п. 8;
   4. Заполнить отчет п. 7.
2. Содержание отчета
   1. Титульный лист;
   2. Цель работы;
   3. Ответы на контрольные вопросы п. 6.3;
   4. Вывод по проделанной работе.
3. Контрольные вопросы
   1. Для чего используется свойство AutomationId у элементов управления?
   2. Как получить доступ к свойствам элемента управления с помощью FlaUI?
   3. Какие возможности предоставляет FlaUIInspect?