МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Казанский национальный исследовательский технический университет

им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации

Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА № 1

по дисциплине

Поддержка и тестирование программных модулей

Работу выполнил

Студент гр.4332

Салахов Камиль Ильгамович

Принял

Преподаватель Шумилкин А.О.

Казань 2024

**Цель работы**

Изучить метод тестирования «Черным ящиком»

**Задание на лабораторную работу**

Задание 1. Использовать ранее созданную программу по своей теме, и провести тестирование методом черного ящика. Минимум 5 тестов по каждой техники, и составление таблицы решений.

Задание 2**.**  Во время проведения тестирования описать этапы, тестовый сценарий, записать всё в таблицу (Таблица №1).

**Результат выполнения работы**

Таблица №1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Название теста | Описание сценария | Входные данные | Выходные данные | Удачное/неудачное тестирование | Предложения по исправлению найденных ошибок. | Пожелания пользователей |
| Корректные данные (Рисунок 1) | Ввести корректные значения во все поля | Имя:Иван  Возраст: 25  Email: [ivan@mail.com](mailto:ivan@mail.com)  Пароль: 12345678  Спорт: Футбол | Успешное сохранение данных | Удачное | - | - |
| Некорректный email (Рисунок 2) | Ввести email без '@' | Имя: Иван  Возраст: 25  Email: ivanmail.com  Пароль: 12345678  Спорт: Футбол | Данные не сохранятся и появится сообщение о некорректном вводе Email | Удачное | - | - |
| Пустые поля (Рисунок 3) | Оставить все поля пустыми | Пустые поля | Данные не сохранятся и появится сообщение о пустых полях | Удачное | - | - |
| Некорректный возраст (Рисунок 4) | Ввести отрицательный возраст | Имя: Иван  Возраст: -5  Email: [ivan@mail.com](mailto:ivan@mail.com)  Пароль: 12345678  Спорт: Футбол | Успешное сохранение данных | Неудачное | - | - |
| Слишком короткий пароль (Рисунок 5) | Ввести пароль менее 8 символов | Имя: Иван  Возраст: 5  Email: [ivan@mail.com](mailto:ivan@mail.com)  Пароль: 12345  Спорт: Футбол | Успешное сохранение данных | Неудачное | - | - |
| Некорректные символы в имени (Рисунок 6) | Ввести в поле "Имя" цифры | Имя: Иван123  Возраст: 25  Email: [ivan@mail.com](mailto:ivan@mail.com)  Пароль: 12345678  Спорт: Футбол | Успешное сохранение данных | Неудачное | - | - |

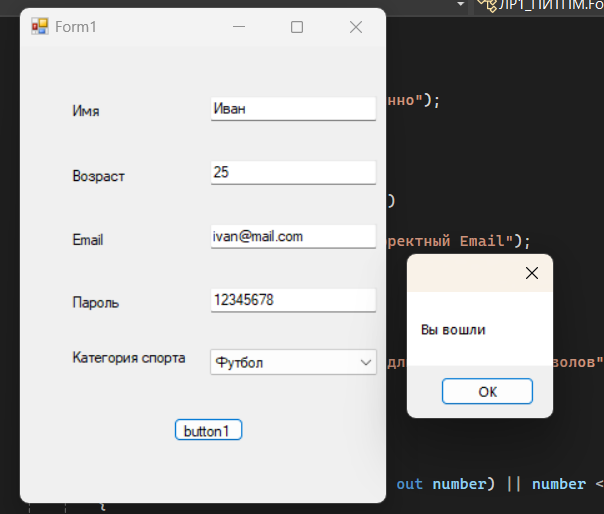
****

Рисунок 1 – Тест с вводом корректных данных

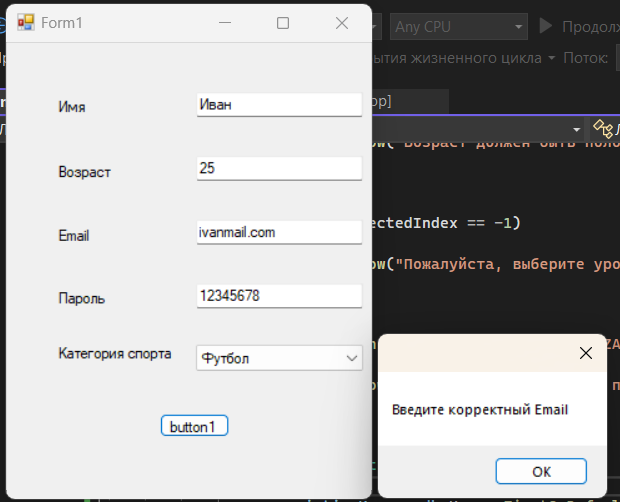
****

Рисунок 2 – Тест с некорректным email

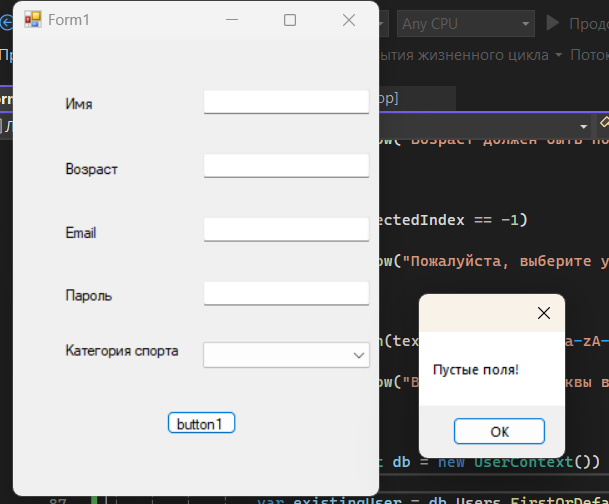
****

Рисунок 3 – Тест с пустыми полями

**Изображение выглядит как текст, снимок экрана, программное обеспечение, мультимедиа

Автоматически созданное описание**

Рисунок 4 – Тест с некорректным возрастом

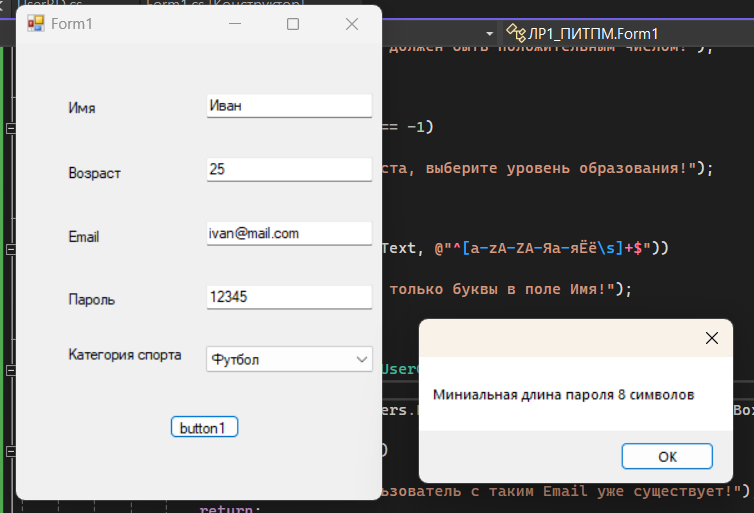
****

Рисунок 5 – Тест со слишком коротким паролем

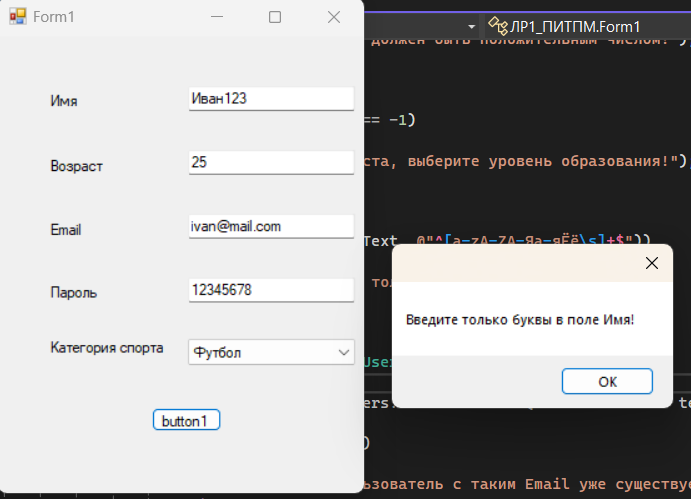
****

Рисунок 6 – Тест с некорректными символами в имени

**Ответы на вопросы**

1. Особенности метода «Черного ящика»

* Отсутствие доступа к коду: тестировщик работает с программой, как конечный пользователь, не зная ее внутреннего устройства.
* Ориентация на функциональность: метод основан на проверке того, как система выполняет требования и задачи, описанные в спецификациях.
* Использование входных и выходных данных: тестировщик проверяет, как программа реагирует на различные вводные данные и соответствует ли результат ожидаемому.
* Отсутствие влияния на дизайн: тестировщик не знает, как реализованы функции внутри программы, он только проверяет их поведение.

2. Достоинства и недостатки метода «Черного ящика»

Достоинства:

* Эффективность для функционального тестирования: метод позволяет проверить, насколько система выполняет заявленные функции.
* Минимизация зависимости от разработчиков: тестировщику не требуется знание кода и внутренней логики.
* Подходит для тестирования пользовательского интерфейса: позволяет проверить удобство и корректность работы системы для конечных пользователей.
* Масштабируемость: можно использовать на разных уровнях тестирования — от модульного до системного и приемочного.

Недостатки:

* Невозможность проверки внутренней логики: тестировщик не видит кода, поэтому некоторые ошибки в архитектуре программы могут остаться незамеченными.
* Ограниченное покрытие: можно пропустить ситуации, которые возникают из-за внутренних взаимодействий частей системы.
* Неэффективность при сложных внутренних ошибках: так как нет доступа к коду, трудно понять причину ошибок, связанных с взаимодействием между модулями.

3. Тестовый сценарий и спецификации

* Тестовый сценарий (test case) — это набор шагов, условий и ожидаемых результатов, который используется для проверки определенной функции или компонента программы. Он включает:
  + Описание условия для тестирования.
  + Порядок действий, который необходимо выполнить.
  + Ожидаемый результат при выполнении сценария.
* Спецификации (requirements specifications) — это документы, описывающие функциональные и нефункциональные требования к программному обеспечению. Спецификации используются для того, чтобы тестировщик знал, как программа должна работать и что именно должно быть проверено.

4. Техники тестирования в методе «Черного ящика»

* Эквивалентное разделение: заключается в разделении входных данных на классы эквивалентности, для которых система должна вести себя одинаково. Из каждого класса выбирается по одному тестовому значению для проверки.
  + Пример: для проверки возрастных ограничений (0-17, 18-60, 61+) можно взять по одному значению из каждого диапазона, а не проверять все возможные значения.
* Анализ граничных значений: тестирование на границах диапазонов допустимых значений, так как ошибки чаще возникают именно в этих точках.
  + Пример: если диапазон допустимых значений — от 1 до 100, проверяются значения 0, 1, 100, 101.
* Таблицы решений: представляют собой таблицу с комбинациями входных данных и соответствующих им ожидаемых результатов. Используются для тестирования сложных логических условий.
  + Пример: если система должна реагировать на комбинацию нескольких параметров (например, скидка на товар при определенных условиях), составляется таблица с возможными комбинациями и результатами.
* Метод состояний и переходов (State Transition Testing): используется для проверки систем, в которых состояние меняется в зависимости от предыдущих действий. Тестирование направлено на проверку корректности переходов между состояниями.
  + Пример: банкомат может находиться в состояниях «ожидание карты», «ввод ПИН-кода», «выбор операции». Каждый переход проверяется на правильность выполнения.
* Тестирование таблиц истинности (Decision Table Testing): техника направлена на тестирование логических условий и правил. Строится таблица, в которой записаны все возможные комбинации условий и действий.
  + Пример: если условие для выдачи кредита зависит от кредитной истории и дохода, таблица будет содержать комбинации этих условий и результат: одобрено или отказано.
* Комбинационное тестирование (Pairwise Testing): метод заключается в выборе всех возможных комбинаций двух входных параметров системы. Он полезен при тестировании системы с множеством параметров.

5. Особенность техники «Таблица решений»

Техника «Таблица решений» (или «Таблица истинности») используется для тестирования сложных логических условий, когда система должна выполнять действия в зависимости от множества параметров.

Особенности:

* Наглядность: таблица позволяет легко увидеть все возможные комбинации условий и действий.
* Полное покрытие вариантов: таблица содержит все возможные комбинации условий, что позволяет избежать пропуска вариантов.
* Простота в анализе сложных логик: тестировщик может увидеть, как изменения одного условия влияют на результат.

Таблица решений особенно полезна при тестировании бизнес-логики или ситуаций, где множество условий и их комбинаций влияют на поведение системы.