Практическая работа 9

Тема: Тестирование программного продукта методом «черного ящика»

# 1. Цель работы

Научиться проводить тестирование «черного ящика» разными способами.

# 2. Теоретическая часть

Тестирование «черного ящика» (Black-box testing) – это метод тестирования программного обеспечения, при котором внутренняя структура, дизайн и реализация неизвестны тестировщику. Тестировщик опирается на требования и спецификации (письменные или устные) и проверяет функциональность программы без анализа кода.

Применение метода «черного ящика»:

1. Интеграционное тестирование – проверка взаимодействия компонентов программы.

2. Функциональное тестирование – проверка выполнения всех заявленных функций.

3. Стресс-тестирование – проверка системы под высокой нагрузкой.

4. Usability-тестирование – оценка удобства использования интерфейса.

5. Тестирование производительности – проверка скорости работы и потребления ресурсов.

6. Приемочное тестирование – проверка программы заказчиком.

7. Регрессионное тестирование – проверка, что новые изменения не вызвали ошибок в старом функционале.

8. Beta-тестирование – тестирование практически готовой программы реальными пользователями.

Метод «черного ящика» позволяет выявить:

- некорректные или отсутствующие функции;

- ошибки интерфейса;

- ошибки работы с внешними структурами данных;

- ошибки характеристик (память, трафик);

- ошибки инициализации и завершения программы.

Классы эквивалентности – это наборы данных с одинаковыми свойствами, на которых программа должна вести себя одинаково.

Правила формирования классов эквивалентности:

- Если условие ввода – диапазон n..m: допустимый класс V = {n..m}, недопустимые – {x < n}, {y > m}.

- Если конкретное значение a: допустимый класс {a}, недопустимые {x < a}, {y > a}.

- Если множество значений {a,b,c}: допустимый класс {a,b,c}, недопустимый {x ≠ a,b,c}.

- Булевы значения: true – допустимый, false – недопустимый.

Анализ граничных значений:

- Проверка значений на границах допустимого диапазона.

- Для диапазона n..m: тесты для n, m, n-1, m+1.

- Для дискретного множества: тесты для минимального и максимального значений, значений чуть меньше и чуть больше.

## 3. Таблица тестирования методом «черного ящика»

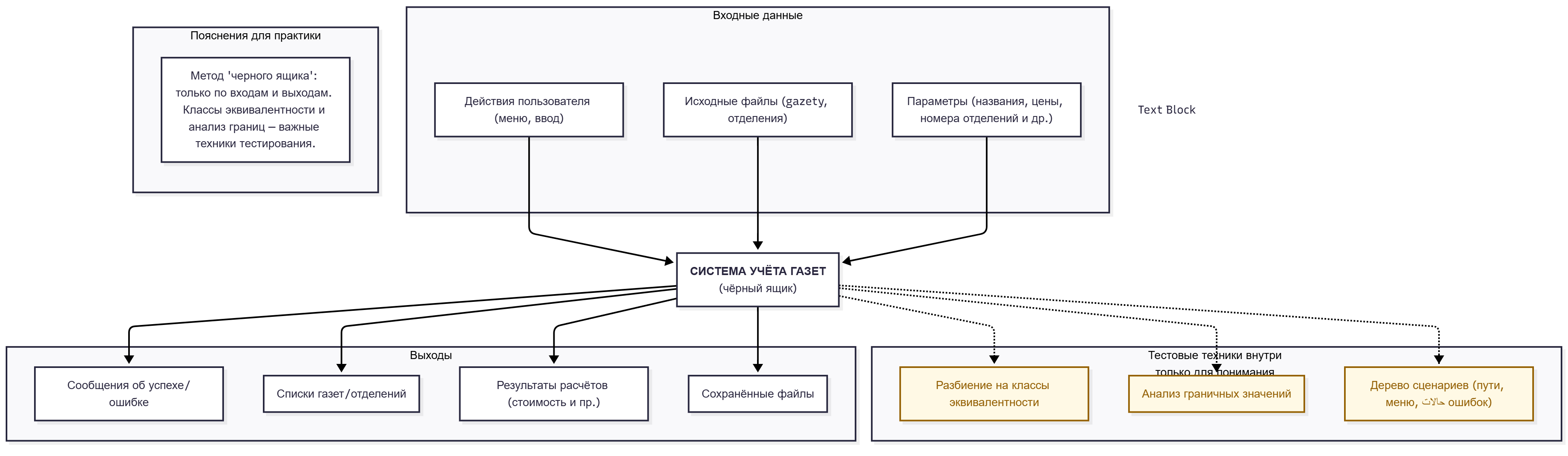
Классы эквивалентности:

|  |  |  |  |
| --- | --- | --- | --- |
| № | Условие | Класс эквивалентности | Тип |
| 1 | Номер отделения | 1 ≤ номер ≤ N (существующее) | допустимый |
| 2 | Номер отделения | <1 или >N (несуществующее) | недопустимый |
| 3 | Газеты в отделении | ≥1 газета | допустимый |
| 4 | Газеты в отделении | 0 газет | недопустимый |
| 5 | Цена газеты | ≥0 | допустимый |
| 6 | Цена газеты | <0 | недопустимый |

## 4. Тестовые варианты

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Назначение теста | Входные данные | Ожидаемый результат | Реакция программы | Вывод |
| 1 | Сумма для существующего отделения с несколькими газетами | Отделение №1, газеты: «Газета1» 50, «Газета2» 70 | 120 руб | 120 руб | Тест пройден |
| 2 | Сумма для отделения без газет | Отделение №2, газеты: отсутствуют | Сообщение «В отделении нет газет» | Сообщение корректно | Тест пройден |
| 3 | Отделение не существует | Отделение №99 (не добавлено) | Сообщение «Неверный выбор отделения» | Сообщение корректно | Тест пройден |
| 4 | Газета с нулевой ценой | Отделение №1, газета: цена 0 | Сумма учитывает 0 руб | 0 руб добавлено | Тест пройден |
| 5 | Газета с высокой ценой | Отделение №1, газета: цена 100000 | Сумма правильно рассчитана | 100000 руб | Тест пройден |

## 5. Блок-схема тестирования

****

# 6. Вывод

Тестирование методом «черного ящика» позволяет проверить корректность работы программы «Система учёта газет» без знания внутренней реализации кода. С помощью классов эквивалентности и анализа граничных значений выявлены ошибки при вводе данных и подсчёте стоимости. Программа корректно