Федеральное государственное образовательное бюджетное   
учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финуниверситет)**

**Факультет Информационных технологий и анализа больших данных**

**СЕМИНАРСКАЯ РАБОТА**

**по дисциплине** «Управление качеством программных систем»

**на тему:** «Методы тест-дизайна»

наименование темы или вариант задания

**Выполнила студентка** 3 курса,

группы ПИ21-7,

формы обучения \_\_\_\_\_очной\_\_\_\_\_\_\_\_\_

(очной, очно- заочной, заочной)

\_\_\_Денисова Ирина Сергеевна\_\_\_\_\_\_\_\_

(Ф.И.О. студента)

**Проверил преподаватель:**

Клочков Е.Ю.

|  |
| --- |
| Дата: 07 марта 2024г. |
|  |

Москва 2024 г.

**Содержание**

Введение 3

1. Список используемых тест-кейсов 4

2. Эквивалентные классы и граничные значения 5

2.1. Имя пользователя 5

2.2. Электронная почта 5

2.3. Пароль 5

2.4. Подтверждение пароля 5

3. Расчет количества тестов 6

Выводы 7

Источники 8

**Введение**

Цель работы: Изучить методы тест-дизайна и применить их для проектирования набора тест-кейсов для тестирования отдельной функциональности приложения, с целью минимизации количества тестов при максимальном покрытии возможных сценариев использования функционала.

Задание: Выбранная форма приложения для анализа - форма регистрации пользователя.

1. **Список используемых тест-кейсов**
   1. Ввод валидных значений во все поля.
   2. Ввод невалидного значения в поле "Имя пользователя".
   3. Ввод невалидного значения в поле "Электронная почта".
   4. Ввод невалидного значения в поле "Пароль".
   5. Ввод невалидного значения в поле "Подтверждение пароля".
   6. Ввод граничных значений для поля "Пароль" (минимальная и максимальная длина).
   7. Ввод граничных значений для поля "Имя пользователя" (минимальная и максимальная длина).
2. **Эквивалентные классы и граничные значения**
   1. Имя пользователя

Валидные значения: Имя пользователя должно содержать от 3 до 50 символов, включая только буквы и цифры. Это формирует один эквивалентный класс.

Невалидные значения: Имя пользователя, содержащее менее 3 символов или более 50 символов, а также имена, включающие специальные символы и пробелы, формируют другой эквивалентный класс.

* 1. Электронная почта

Валидные значения: Правильно оформленный адрес электронной почты (например, user@example.com) составляет валидный эквивалентный класс.

Невалидные значения: Адреса, не соответствующие формату электронной почты, или уже зарегистрированные адреса формируют невалидный эквивалентный класс.

* 1. Пароль

Валидные значения: Пароль должен быть длиной от 8 до 50 символов и содержать хотя бы одну цифру, одну заглавную и одну строчную букву, что формирует валидный эквивалентный класс.

Невалидные значения: Пароли, не соответствующие указанным требованиям, формируют невалидный эквивалентный класс.

* 1. Подтверждение пароля

Валидные значения: Подтверждение пароля, точно совпадающее с введенным паролем, составляет валидный эквивалентный класс.

Невалидные значения: Любое несовпадение с введенным паролем формирует невалидный эквивалентный класс.

1. **Расчет количества тестов**
   1. Ввод валидных значений для всех полей (1 тест): Этот тест проверяет, что форма регистрации работает корректно при вводе правильных данных.
   2. Ввод невалидных значений для каждого поля по отдельности (4 теста): Для каждого поля формы создается отдельный тест с невалидными данными, при этом остальные поля заполняются валидными значениями. Это помогает проверить обработку ошибок для каждого поля в отдельности.
   3. Тестирование граничных значений (дополнительно минимум 8 тестов): Граничные значения тестируются для поля "Имя пользователя" (2 теста: по одному для минимального и максимального количества символов) и "Пароль" (6 тестов: по два для минимальной и максимальной длины, а также для проверки требований к содержанию пароля).

Итого: 13 тестов.

**Выводы**

Применение методов тест-дизайна, таких как анализ эквивалентных классов и тестирование граничных значений, позволяет организовать тестирование более систематически и эффективно. Это не только способствует обнаружению большего количества ошибок с меньшим количеством тестов, но и помогает в оптимизации ресурсов, выделяемых на тестирование. Ключевым моментом является способность методов тест-дизайна обеспечить максимальное покрытие тестированием при минимальном количестве тестов, что особенно важно в условиях ограниченных временных и ресурсных рамок проекта.

**Источники**

1. Канер, С., Фолк, Дж., Нгуен, Х.К. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений.
2. Myers, Glenford J., 1946-The art of software testing / Glenford J. Myers, Corey Sandler, Tom Badgett. — 3rd ed.