Екзаменаційна робота
з дисципліни
Якість програмного забезпечення та тестування
Студента ІПЗ-33 Гоши Давіда Олександровича

Білет 7

1ПЗ-33 Госса Давід Олександрович
Екзаменаційна робота з дисципліни
ЯПЗП

Білет № 7

1. Методи оцінки значень показників якості

Існують такі методи оцінки якості

1. Опитування та інтерв'ю: Отримання прямого та зворотнього зв'язку.
2. Показники ефективності: Використання аналітичних даних для статистики використання продуктів, тощо
3. Тестування: Використання тестування програмного забезпечення або продуктів для оцінки якості
4. Порівняльний аналіз: Порівняння з ринковими стандартами.
5. Експертні оцінки: Отримання відгуків від експертів у цій галузі
6. Скарги клієнтів: Вивчення скарг, що потребують покращення

2. Інтеграційне тестування. Задачі і цілі.

Інтеграційне тестування – це тестування рівня, на якому окремі модулі об'єднуються і тестуються як група. Метою цього рівня тестування є виявлення помилок у взаємодії між інтегрованими модулями.

Завдання:

Виявити помилки у взаємодії між компонентами. Переконатися, що інтегровані модулі добре працюють разом.

Протестувати інтерфейс між модулями

Знайти проблему, пов'язані з взаємодією між різними частинами системи.

Переконатися що всі частини системи працюють разом

3. Тестування зручності використання інтерфейсів користувача. Карти емпатії

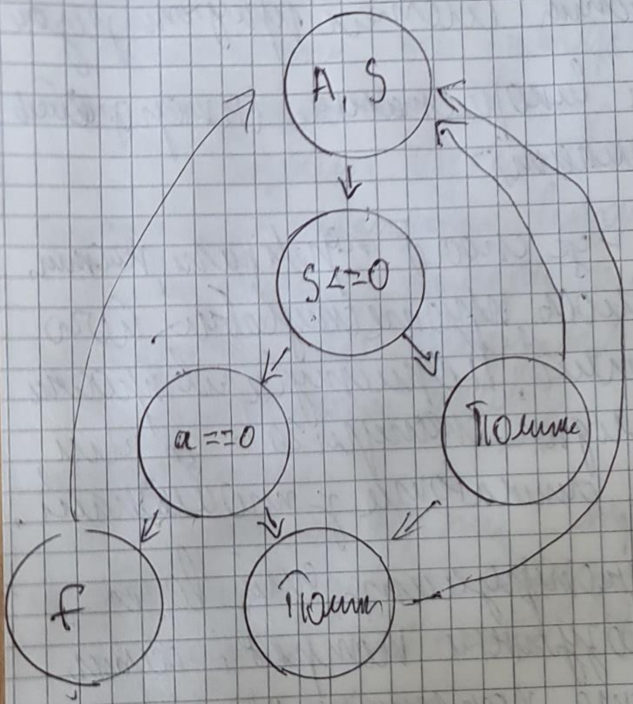
1) Юзабіліті-тестування - це спосіб перевірити, наскільки легко користуватися чимось, протестувавши його з реальними користувачами. Користувачів просять виконувати завдання, а дослідник спостерігає за ними, щоб побачити, де вони стикаються з проблемами.

Карта емпатії - це інструмент, який можна використовувати для розуміння потреб і бажань користувачів. Вона часто поділена на секції, що відповідають тому, що користувачі говорять, думають, роблять і відчувають. Метою карти емпатії - є отримати глибше уявлення про те, що відчуває користувач, щоб краще проектувати відповідно до його потреб

4. Побудувати мінімальну модель керуючого графу обчислення виразу, виділити необхідну кількість шляхів тестів, щоб протестувати код на 100%

$f = 12 - \frac{s}{2a-4}, s > 0$

Потоковий граф



Вихід з виду шляхів (Цикломатична складність)

$9 - 6 + 2 \cdot 0 = \boxed{3}$

Отже достатньо 3 тестових або ми впорядкуємо в усі границі значень для 100% протестованості. Зокрема 0 або від'ємні значення

Код:

КОД

```
def calculate(s, a):
    # Check if the conditions are met
    if s <= 0 or a == 2:
        return 'Invalid input'

    # If conditions are met, calculate the function
    f = 12 - s/(2*a - 4)
    return f

# Define your test cases
test_cases = [
    (1, 3),  # s > 0, a != 2
    (10, 5),  # s > 0, a != 2
    (100, 1),  # s > 0, a != 2
    (1, 2),  # s > 0, a == 2
    (-1, 3),  # s <= 0, a != 2
]

# Run the test cases
for s, a in test_cases:
    print(f'For s = {s}, a = {a}, f = {calculate(s, a)}')
```