

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ
Кафедра інформаційних технологій

ЯКІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ
МЕТОДИЧНІ ВКАЗІВКИ
для самостійної роботи здобувачів
014.09 Середня освіта (Інформатика)

Методичні рекомендації для самостійної роботи здобувачів

Одеса 2025

ЗМІСТ

Загальні положення	4
1 Контрольна робота № 1	5
2 Контрольна робота № 2	7
3 Приклад виконання контрольної роботи	9
4 Вимоги до оформлення контрольної роботи	10
1. Перелік посилань	12

ЗАГАЛЬНІ ПОЛОЖЕННЯ

У процесі підготовки фахівців з Інженерії програмного забезпечення за дисципліною «Якість програмного забезпечення та тестування», яка належить до обов'язкових дисциплін, визначальне значення має не тільки засвоєння теоретичних знань, але й практичне опанування основ забезпечення якості та тестування ПЗ.

Мета даних методичних вказівок – допомогти студентам самостійно підготуватися до виконання контрольних завдань з основних розділів дисципліни, засвоїти методи аналізу процесів якості ПЗ та тестування, навчитися розроблювати тестову документацію, оволодіти професійною термінологією.

Контрольні завдання призначені для перевірки знань студентів з основних тем, розглянуто також приклад розв'язання типових завдань.

Контрольні завдання призначені для проведення проміжного та підсумкового семестрового контролю, а також можуть бути використані під час підготовки та проведення державної атестації для студентів спеціальності 014.09 – Середня освіта (Інформатика).

1 КОНТРОЛЬНА РОБОТА №1

Варіант 1

4. Класифікація видів тестування.
5. Методи тестування.
6. Написати позитивний та негативний тест кейс для тестування поля «логін».

Варіант 2

4. Які види тестування відносяться до функціональних.
5. Метрики в тестуванні.
6. Написати позитивний та негативний тест кейс для тестування поля «пароль».

Варіант 3

4. Які види тестування відносяться до нефункціональних.

5. Забезпечення та контроль якості ПЗ (QA та QC).
6. Написати позитивний та негативний тест кейс для тестування форми авторизації.

Варіант 4

5. Які види тестування відносяться до пов'язаних зі змінами.
6. Характеристики якості ПЗ.
7. Написати позитивний та негативний тест кейс для тестування калькулятора.

Варіант 5

4. Тестування безпеки (Security and Access Control Testing).
5. Життєвий цикл ПЗ.
6. Написати позитивний та негативний тест кейс для функції обчислення кореня в калькуляторі.

Варіант 6

4. Тестування взаємодії (Interoperability Testing).
5. Структура баг репорту.
6. Написати позитивний та негативний тест кейс для роботи з кошиком в інтернет-магазині (додавання товару).

Варіант 7

4. Тестування установки (Installation Testing).
5. Структура тест плану.
6. Написати позитивний та негативний тест кейс для роботи з кошиком в інтернет-магазині (видалення товару).

Варіант 8

4. Тестування зручності користування (Usability Testing).
5. Тестова документація.
6. Написати позитивний та негативний тест кейс для роботи з кошиком в інтернет-магазині (редагування товару).

Варіант 9

4. Димове тестування (Smoke Testing).
5. Що таке чек лист? Навести приклад.
6. Написати позитивний та негативний тест кейс для функції обчислення суми двох чисел в калькуляторі.

Варіант 10

4. Регресійне тестування (Regression Testing).
5. Матриця трасування вимог (Traceability Matrix).
6. Написати позитивний та негативний тест кейс для функції обчислення добутку двох чисел в калькуляторі.

Варіант 11

4. Типи регресійного тестування.
5. Важливість і пріоритет дефекту.
6. Написати позитивний та негативний тест кейс для тестування форми реєстрації.

Варіант 12

4. Тестування зборки (Build Verification Test).
5. Життєвий цикл бага.
6. Написати позитивний та негативний тест кейс для тестування поля «пошук». Варіант 13

4. Санітарне тестування або перевірка погодженості/справності (Sanity Testing).
5. Мануальне та автоматизоване тестування? Коли має сенс застосовувати.
6. Написати позитивний та негативний тест кейс для тестування сайту ХНУРЕ.

Варіант 14

4. Навантажувальне тестування (Performance and Load Testing).
5. Характеристики якості ПЗ.
6. Написати позитивний та негативний тест кейс для тестування розкладу занять.

Варіант 15

4. Стресове тестування (Stress Testing).
5. Статичне та динамічне тестування.
6. Написати позитивний та негативний тест кейс для тестування сайту кінотеатру.

Варіант 16

4. Тестування стабільності або надійності (Stability / Reliability Testing).
5. Методи тестування.
6. Написати позитивний та негативний тест кейс для тестування сайту аеропорту.

Варіант 17

4. Об'ємне тестування (Volume Testing).
5. Дослідницьке тестування.
6. Написати позитивний та негативний тест кейси для перевірки валідації поля «phone».

Варіант 18

4. Функціональне тестування (Functional testing).
5. Тестування локалізації.
6. Написати позитивний та негативний тест кейс для перевірки поля введення email.

Варіант 19

4. Відмінності між навантажувальним та стресовим тестуванням.
5. Моделі життєвого циклу ПЗ.
6. Написати позитивний та негативний тест кейс для форми відновлення пароля.

Варіант 20

3. Аспекти тестування функціональності.
4. Статичне та динамічне тестування .

5. Написати позитивний та негативний тест кейс для тестування поля «User name».

2 КОНТРОЛЬНА РОБОТА № 2

Варіант 1

4. Техніки тест-дизайну.
5. Компонентне або модульне тестування (Component Testing or Unit Testing).
6. Протестувати ноутбук.

Варіант 2

4. Еквівалентний поділ (Equivalence Partitioning – EP).
5. Інтеграційне тестування (Integration Testing).
6. Протестувати аудиторію.

Варіант 3

4. Аналіз граничних значень (Boundary Value Analysis – BVA).
5. Системне тестування (System Testing).
6. Протестувати чашку.

Варіант 4

4. Вичерпне тестування (Exhaustive Testing – ET).
5. Приймальне тестування (Acceptance Testing).
6. Протестувати ліфт.

Варіант 5

4. Інспекції та наскрізні перегляди.
5. Функціональні та нефункціональні вимоги.
6. Протестувати банкомат.

Варіант 6

4. Види тест планів.
5. Бізнес-вимоги.
6. Протестувати ескалатор.

Варіант 7

4. Розробка через тестування (TDD – test driven development).
5. Бізнес-правила.
6. Протестувати мотузку.

Варіант 8

4. Системи відслідковування помилок (bug tracking systems).
5. Системне тестування (System Testing).
6. Протестувати будильник.

Варіант 9

4. Рівні інтеграційного тестування.
5. Вимоги користувачів.
6. Протестувати стіл.

Варіант 10

4. Підходи до інтеграційного тестування.
5. Властивості вимог до ПЗ.
6. Протестувати стол.

Варіант 11

4. Підходи до системного тестування.
5. Документи, що відображають вимоги.
6. Протестувати холодильник.

Варіант 12

4. Типи регресійного тестування.
5. Визначити різницю між поняттями «функціональні» і «нефункціональні» вимоги.
6. Протестувати аркуш паперу.

Варіант 13

4. Техніка тест дизайну Причина / Наслідок (Cause/Effect – CE).
5. Види вимог за рівнями.
6. Протестувати чайник.

Варіант 14

4. Техніка тест дизайну Передбачення помилки (Error Guessing – EG).
5. Види вимог за характером.
6. Протестувати космічну станцію.

Варіант 15

4. Градація Серйозності дефекту (Severity).
5. Способи представлення вимог.
6. Протестувати гітару.

Варіант 16

4. Градація Пріоритету дефекту (Priority).
5. Функціональні вимоги до ПЗ.
6. Протестувати степлер.

Варіант 17

4. Техніки тест-дизайну.
5. Інженерія вимог.
6. Протестувати тостер.

Варіант 18

4. Відмінність димового від санітарного тестування.
5. Аналіз вимог.
6. Протестувати старт-годинник.

Варіант 19

4. Різниця між error, bug, failure.
5. Нефункціональні вимоги до ПЗ.
6. Протестувати гамак.

Варіант 20

4. Системи відслідковування помилок.
5. Тестування документації.

6. Протестувати окуляри.

3 ПРИКЛАД ВИКОНАННЯ КОНТРОЛЬНОЇ РОБОТИ

Завдання:

4. Види тестування продуктивності.
5. Артефакти тестування.
6. Написати позитивний та негативний тест кейс.

4. Задачею тестування продуктивності (Performance testing) є визначення масштабованості додатка під навантаженням, при цьому відбувається:

- вимір часу виконання обраних операцій при визначених інтенсивностях виконання цих операцій;
- визначення кількості користувачів, що одночасно працюють з додатком;
- визначення границь прийнятної продуктивності при збільшенні навантаження (при збільшенні інтенсивності виконання цих операцій)
- дослідження продуктивності на високих, граничних, стресових навантаженнях.

Навантажувальне тестування (Load vs Performance Testing). В англomовній термінології ви можете так само знайти ще один вид тестування - Load Testing – тестування реакції системи на зміну навантаження (у межах припустимого). Load і Performance переслідують одну й ту саму мету: перевірка продуктивності (часів відгуку) на різних навантаженнях. Власне тому їх не розділяють.

Стресове тестування (Stress Testing) дозволяє перевірити наскільки додаток і система в цілому працездатні в умовах стресу і також оцінити здатність системи до регенерації, тобто до повернення до нормального стану після припинення впливу стресу. Стресом у даному контексті може бути підвищення інтенсивності виконання операцій до дуже високих значень або аварійна зміна конфігурації сервера. Також однією з задач при стресовому тестуванні може бути оцінка деградації продуктивності, у такий спосіб мети стресового тестування можуть перетинатися з цілями тестування продуктивності.

Тестування стабільності або надійності (Stability / Reliability Testing) – це перевірка працездатності додатка при тривалому (багатогадинному) тестуванні із середнім рівнем навантаження. Часи виконання операцій можуть грати в даному виді тестування другорядну роль. При цьому на перше місце виходить відсутність витоків пам'яті, перезапусків серверів під навантаженням і інші аспекти, що впливають саме на стабільність роботи.

Об'ємне тестування (Volume Testing) – це одержання оцінки продуктивності при збільшенні обсягів даних у базі дані додатки, при цьому відбувається:

- вимір часу виконання обраних операцій при визначених інтенсивностях виконання цих операцій;
- може вироблятися визначення кількості користувачів, що одночасно працюють із застосуванням.

5. У відповідність з процесами або методологіями розробки ПЗ, під час проведення тестування створюється і використовується певна кількість тестових артефактів (документи, моделі тощо). Найбільш поширеними тестовими артефактами є:

– План тестування (Test Plan) – це документ описує весь обсяг робіт з тестування, починаючи з опису об'єкта, стратегії, розкладу, критеріїв початку і закінчення тестування, необхідного в процесі роботи обладнання, спеціальних знань, а також оцінки ризиків з варіантами їх вирішення.

– Набір тест кейсів і наборів (Test Case & Test suite) – це послідовність дій, за якою можна перевірити чи відповідає тестована функція встановленим вимогам.

– Дефекти / Баг Репорт (Bug Reports / Defects) – це документи, що описують ситуацію або послідовність дій, яка призвела до некоректної роботи об'єкта тестування, із зазначенням причин і очікуваного результату.

6. Приклад позитивного та негативного тест кейсу:

Позитивний тест кейс		Негативний тест кейс (поле Контактна особа - NOK):	
Дія	Очікуваний результат	Дія	Очікуваний результат
1. Відкриваємо форму відправлення повідомлення	- Форма відкрита - Усі поля за замовчуванням порожні - Обов'язкові поля позначені - * - Кнопка «Відправити» не активна	1. Відкриваємо форму відправлення повідомлення	- Форма відкрита - Усі поля за замовчуванням порожні - Обов'язкові поля позначені - * - Кнопка «Відправити» не активна
2. Заповнюємо поля форми: - Тип звертання Консультація - Контактна особа йцукенгшщйцукенгшщйцук е - Контактний телефон +7-916-111-11-11 - Повідомлення	- Поля заповнені - Кнопка «Відправити» - активна (Enabled)	2. Заповнюємо поля форми: - Тип звертання Консультація - Контактна особа @#\$\$%^&.;?,> /№'"!()_}{ <~ - Контактний телефон (916)333-33-33 - Повідомлення йццуйцуйц(...)йцу - 1024 символу	- Поля заповнені - Кнопка «Відправити» - активна (Enabled)
3. Натискаємо кнопку «Відправити»	кор Якщо введені дані - ектні: - Повідомлення «Заявка відправлена» виведене на екран. Нова заявка з'явилася в списку на сторінці «Заявки».	3. Натискаємо кнопку «Відправити»	- Валідаційне повідомлення з усіма помилками виведено на екран: «У поле <i>Контактна особа</i> заборонене використання цифр і спец. символів.» Заявка НЕ з'явилася в списку на сторінці «Заявки».

ВИМОГИ ДО ОФОРМЛЕННЯ КОНТРОЛЬНОЇ РОБОТИ

Контрольна робота повинна мати атрибути: назва дисципліни, номер варіанта контрольної роботи, ПІБ студента, група. Далі за текстом контрольної роботи наводяться питання, що необхідно розкрити відповідно до завдання та результат у вигляді відповіді на поставлені запитання.

4 ПЕРЕЛІК ПОСИЛАНЬ

1. Канер, Кем, Тестування програмного забезпечення. Фундаментальные концепции менеджменту бизнес-приложений [Текст]: учеб. / Д. Фолк, Енг Кек Нгуен – К.: ДияСофт, 2001. – 544 с.
2. Калбертсон, Роберт Быстрое тестирование [Текст]: учеб. / К. Браун, Г. Кобб. – М.: «Вильямс», 2002. – 374 с.
3. IEEE Standard for a Software Quality Metrics Methodology Copyright ©, Institute of Electrical and Electronics Engineers, Inc. 2005.
4. The Guide to the Software Engineering Body of Knowledge (SWEBOK Guide) [Електронний ресурс]. – Режим доступу: <https://www.computer.org/web/swebok>.
5. Майерс Г. Дж. Искусство тестирования программ [Електронний ресурс] : пер. с англ. / Г. Дж. Майерс ; 3-е изд. – Москва : ООО "И.Д.Вильямс", 2012. – 272 с.

МЕТОДИЧНІ МАТЕРІАЛИ
для підсумкового контролю з дисципліни
«Якість програмного забезпечення та тестування»
для студентів усіх форм навчання
спеціальності 014.09 – Середня освіта (Інформатика)
Електронне видання

1. Забезпечення якості і контроль якості ПЗ.
2. Характеристики якості ПЗ.
3. Методологія розробки Rational Unified Process (RUP).
4. Зовнішня і внутрішня якість ПЗ.
5. Верифікація і валідація.
6. Вартість якості (Cost of quality).
7. Вартість помилки (Failure cost).
8. Модель Capability Maturity Model (CMM).
9. Модель Capability Maturity Model Integration (CMMI).
10. Види вимог. Вимоги до вимог.
11. Життєвий цикл дефекту.
12. Види тестування.
13. Рівні тестування.
14. Методи тестування.
15. Інспекції та наскрізні перегляди.
16. Тестові артефакти.
17. Тест-кейси і чек-листи.
18. Види тест планів.
19. Тест дизайн.
20. Техніки тест дизайну.
21. Розробка через тестування (TDD – test driven development).
22. Метрики в тестуванні.
23. Технологія швидкого тестування.
24. Автоматизація тестування.
25. Системи відслідковування помилок (bug tracking systems).
26. Навантажувальне і стресове тестування.
27. Димове тестування.
28. Санітарне тестування.
29. Тестування зручності використання.
30. Моделі життєвого циклу ПЗ.
31. PDCA-цикл.
32. Agile-методології.
33. Agile-маніфест.
34. Екстремальне програмування (XP).
35. Рефакторинг.
36. Особливості тестування в залежності від методології розробки ПЗ.
37. Основні етапи процесу створення ПЗ.

38. Каскадна та водоспадна модель. Визначення. Основні відмінності.
39. Інтеграційне та конфігураційне тестування. Визначення. Основні відмінності.
40. Тестування локалізації та інтернаціоналізації.
41. Бізнес-вимоги і бізнес-правила.
42. Статичне та динамічне тестування.
43. Альфа- та бета-тестування.
44. Термінологія поняття «помилка»: bug, defect, failure, fault, error.
45. Симптоми помилок в баг-репорті.
46. Типи регресійного тестування.
47. Компонентне (Component) та модульне тестування (Unit Testing).
Визначення. Відмінності.
48. Рівні інтеграційного тестування.
49. Підходи до інтеграційного тестування.
50. Підходи до системного тестування.