

Практична робота № 2

Тема: Цикли тестування.

Мета: навчитися складати цикл тестування.

Теоретичні відомості:

Test cycle - це набір тестів з різних розділів тест-плану (різних типів тестування), відібраних для перевірки певної версії на певній фазі проекту; це також процес перевірки конкретної версії на відповідність тим вимогам до продукту, яким відповідає втілена у версії функціональність. У зв'язку із цим, сам Test cycle може змінювати свій зміст, тимчасові характеристики й необхідні ресурси. Як правило, Test cycle розростається від версії до версії. Однак у процесі розростання може знадобитися проведення окремого Test cycle для специфічної частини продукту. Більше того, може знадобитися проводити кілька Test cycles для одного продукту паралельно різними командами.

У даному розділі типи тестування пояснюються в тому порядку, в якому їх прийнято виконувати. Однак чіткої послідовності для виконання типів тестів немає.

Із чого складається Test cycle:

Перевірка на придатність до тестування (Sanity testing)

Даний тип тестів є одним із ключових типів у тестуванні й організації Test cycles. Призначення цих тестів полягає в попередній перевірці придатності нової версії для наступного повного тестування. Перевіряється базова функціональність і загальна стабільність продукту. У випадку успішного проходження даних тестів нова версія приймається для тестування. У протилежному випадку, відсилається на доробку із вказівкою того, що не працює.

Перевірка виправлень (fixes verification)

Ці тести призначені для перевірки виправлень дефектів, знайдених під час попередніх Test Cycles. Насправді, даний вид тестів не обов'язково виконувати відразу після прийому нової версії для тестування. Однак досить корисно дати швидку інформацію розроблювачам про стан виправлених дефектів перш ніж почнеться новий цикл.

Функціональне тестування (Functional testing) / тестування відповідності (Conformance testing, Correctness testing)

Ці тести можуть називатися по різному, однак, їхня суть проста – перевірка відповідності системи поставленим до неї вимогам.

Тестування інтерфейсу користувача (GUI testing)

Основною метою даних тестів є перевірка графічного інтерфейсу користувача на відповідність GUI-специфікаціям продукту (дотримання всіх розмірів, взаємне розташування, колірна гама, і т.д.).

Регресійне тестування (Regression testing)

Визначення успішності регресійних тестів (IEEE 610–90 “Standard Glossary of Software Engineering Terminology”) говорить: «повторне вибіркове тестування системи або компонент для перевірки зроблених модифікацій не повинне приводити до непередбачених ефектів». На практиці це означає, що якщо система успішно проходила тести до внесення модифікацій, вона повинна їх проходити й після внесення. Основна проблема регресійного тестування полягає в пошуку компромісу між наявними ресурсами й необхідністю проведення таких тестів по мірі внесення кожної зміни. Деякою мірою, завдання полягає в тім, щоб визначити критерії «масштабів» змін, з досягненням яких необхідно проводити регресійні тести.

Результати (Test outcome)

Результати є метою виконання Test cycle. В остаточному підсумку всіх цікавить, у якому стані знаходиться продукт, а особливо це цікавить менеджерів. Звітам про помилки/дефекти присвячений цілий курс, тому ми не будемо докладно зупинятися на цьому питанні. Варто лише відзначити, що повнота опису проблеми, доступність у поясненні, а також колекціонування допоміжної інформації (скріншоти, логи, конфігураційні файли і т.д.) істотно полегшують життя й прискорюють процес виправлення дефектів, а з ним і розробки.

Більшою мірою багатьох цікавить звіт по результатами всього тестування. Даний результат дає деякий зріз стану продукту на даний момент. Це те, заради чого й виконувався сам Test cycle.

Основна інформація, яку необхідно зібрати в одному документі:

- набір характеристик продукту, які були протестовані в даному Test cycle;
- час, витрачений на тестування, із вказівкою кількості людських ресурсів, задіяних у виконанні Test cycle;
- покриття тестами реалізованого функціонала по кожній характеристиці;
- загальна статистика знайдених дефектів у розрізі серйозності проблеми;
- детальна статистика знайдених дефектів для кожної характеристики;
- короткий опис стану кожної характеристики;
- подальші дії після закінчення Test cycle.

Варто звернути увагу на ще один дуже важливий документ, що призначений для перегляду стану продукту – це **Матриця Відповідності (Matrix of Traceability)**. Цей документ, як правило, заповнюється керівниками команд на стадії планування, а потім контролюється менеджером проекту.

Документ містить список вимог, для кожної вимоги вказується дата розробки, набір тестів, які покривають реалізацію даної вимоги (тільки ідентифікатори тестів), результати тестування на даний момент. Цей документ можна розширювати дуже сильно, але його призначення від цього не зміниться. Суть документа - подивитися, що вже реалізовано в продукті (або на якій стадії перебуває реалізація) і які проблеми були знайдені.

Не всі типи тестування необхідно, а часом і можливо проводити із самого початку. Із цієї причини деякі типи тестування включаються в Test Cycle на пізніх стадіях, а деякі згодом переходять у розряд регресійних.

Установочне тестування (Installation testing)

З назви походить, що дані тести проводяться з метою перевірки процедури інсталяції системи в цільовому оточенні.

Тестування продуктивності (Performance testing)

Спеціалізовані тести перевірки задоволення специфічних вимог, пропонованих до параметрів продуктивності. Існує особливий підвид таких тестів, коли робиться спроба досягнення кількісних меж, обумовлених характеристиками самої системи і її операційного оточення.

Навантажувальне тестування (Stress testing)

Необхідно розуміти відмінності між розглянутим вище тестуванням

продуктивності з метою досягнення її реальних (досяжних) можливостей продуктивності й виконанням програмної системи з підвищенням навантаження, аж до досягнення запланованих характеристик і далі, з відстеженням поведінки на протязі підвищення завантаження системи.

Порівняльне тестування (Back-to-back testing)

Одиничний набір тестів, що дозволяють порівняти дві версії системи.

Відновлювані тести (Recovery testing)

Ціль – перевірка можливостей рестарту системи у випадку непередбаченої катастрофи, що впливає на функціонування операційного середовища, у якому виконується система.

Конфігураційне тестування (Configuration testing)

У випадках, якщо програмне забезпечення створюється для використання різними користувачами (у термінах «ролей»), даний вид тестування спрямований на перевірку поведінки й працездатності системи в різних конфігураціях.

Тестування зручності й простоти використання (Usability testing)

Ціль – перевірити, наскільки легко кінцевий користувач системи може її освоїти, включаючи не тільки функціональну складову – саму систему, але і її документацію; наскільки ефективно користувач може виконувати завдання, автоматизація яких здійснюється з використанням даної системи; нарешті, наскільки добре система застрахована (з точки зору потенційних збоїв) від помилок користувача.

Практичне завдання:

Розробити комбінований, послідовний і паралельний цикл тестування для калькулятора.

Контрольні питання:

1. Що таке Test Cycle?
2. Опишіть Test Cycle з послідовним виконанням
3. Опишіть Test Cycle з паралельним виконанням
4. Опишіть Test cycle з комбінованим виконанням
5. З чого складається Test Cycle?