Beet Seed

Завдання 1

|  | Що перевіряється | Коли застосовується | Обмеження | Особливості |
| --- | --- | --- | --- | --- |
| Функціональне тестування | Функціональні вимоги до системи та їх виконання | Після розробки функціональної частини програми або системи | Функціональне тестування зазвичай важливе для перевірки, чи виконуються конкретні функції програми | Тестування відбувається на основі конкретних вимог та очікуваних результатів |
| Нефункціональне тестування | Нефункціональні аспекти, такі як продуктивність, безпека, масштабованість тощо | Після реалізації функціональних вимог та перед випуском на ринок | Нефункціональні тестування можуть бути важкими для автоматизації та вимагати спеціалізованих інструментів | Тестування вимагає детальної специфікації вимог та критеріїв прийняття |
| Види тестувань пов’язаних зі змінними | Вплив змін у коді або архітектурі на існуючу функціональність та нефункціональні аспекти | Під час регулярних оновлень програми або системи | Пов’язане зі змінами тестування може бути витратним за часом, оскільки потребує перевірки функціональності в контексті змін | Тестування вимагає чіткої стратегії та процесу управління змінами для забезпечення безпеки та ефективності впроваджень |

Завдання 2

Регресійне тестування та ретестинг - це дві різні стратегії тестування, які використовуються для перевірки програмного забезпечення після внесення змін. Регресійне тестування спрямоване на перевірку, чи не вплинули зміни в програмі на її раніше працездатні функції. Воно зазвичай автоматизується та включає в себе виконання попередньо написаних тестів для забезпечення стабільності програми після внесення змін. З іншого боку, ретестинг - це процес повторного виконання вже пройдених тестів для перевірки виправлення дефектів або змін у програмі. Він не обов'язково пов'язаний зі змінами у програмному забезпеченні, а може бути проведений для перевірки стабільності після виправлення помилок. Основна відмінність полягає в тому, що регресійне тестування спрямоване на перевірку впливу змін на існуючий код, тоді як ретестинг перевіряє, чи вирішено виявлені раніше проблеми.

Beet Sprout

Завдання 1

Функціональне тестування, яке перевіряє конкретні функції або операції продукту, може бути проведене без безпосередньої перевірки нефункціональних вимог у деяких випадках. Проте, це може бути неповноцінним тестуванням, оскільки нефункціональні аспекти, такі як продуктивність, безпека, масштабованість, можуть бути критичними для користувачів та стабільності продукту в цілому. Таким чином, хоча функціональне тестування може забезпечити базовий рівень якості, для повного оцінювання продукту і його відповідності вимогам, необхідне також нефункціональне тестування. Такий підхід дозволить підвищити загальний рівень якості продукту та забезпечити кращий досвід користувача.

Завдання 2

Smoke (димове) тестування - це швидке та поверхневе тестування основних функціональностей програми або системи з метою перевірки її базової працездатності. Його мета - виявити серйозні проблеми, які можуть призвести до відмови в основних функціях, ще до того, як тестування перейде до більш глибоких або детальних етапів.

Димове тестування є доречним у випадках, коли потрібно швидко оцінити загальний стан програми або системи перед подальшими тестами. Воно особливо корисне у випадках, коли розробники зробили значні зміни у коді або додали нові функції, і необхідно швидко переконатися, що базова функціональність залишається стабільною.

Однак, димове тестування не завжди є доречним у всіх випадках. Наприклад, якщо продукт або система ще не розроблені або мають мінімальний функціонал, то димове тестування може бути непотрібним, оскільки немає базових функцій для перевірки. Також, якщо для продукту важливі не тільки основні функції, а й нефункціональні аспекти, наприклад, продуктивність або безпека, то може бути необхідно використовувати більш ретельні тести.

Отже, хоча димове тестування може бути доречним у багатьох ситуаціях, важливо аналізувати конкретний контекст і потреби проекту, щоб визначити, чи буде воно корисним у конкретній ситуації.