

Техніки тест-дизайну

Black box testing

Тестування, при якому ми не бачимо код програми, а бачимо тільки її реакцію на якісь дії

Equivalence Partitioning (Класи еквівалентності)

Визначаємо класи еквівалентності та створюємо тест-кейси для кожного класу. Окремо тестується 0. Тест кейси можна створювати як позитивні, так і негативні

Boundary Value Analysis (Аналіз граничних значень)

Доповнює класи еквівалентності на границі змін умов. Тобто, окрім одного представника еквівалентної групи, тестуємо також граничні значення між класами

Decision Table Testing (Таблиця прийняття рішень)

Використовуємо цю техніку, коли є декілька умов, але їх небагато. Створюємо таблицю та визначаємо всі можливі варіанти подій. Тестуємо окремо кожен з них.

Pairwise Testing (Попарне тестування)

Техніка використовується, коли є багато різних умов. Суть полягає в тому, що кожне значення одного параметра перевіряється і тестується на взаємодію з кожним значенням інших параметрів. Тестування проводиться за допомогою спеціальних програм

State Transition Testing (Аналіз переходу станів)

Дана техніка візуалізує всі можливі варіанти переходу з одного стану в інші. Для цього малюємо схему зі станами програми і малюємо всі варіанти можливих подій та переходу між станами. Тестуємо окремо кожен з них

Use Case Testing (Юз кейс тестування)

Для цієї техніки пишуться сценарії (юз-кейси) можливих дій користувачів в програмі і ці сценарії тестуються

White box testing

Тестування, при якому ми бачимо код програми і розуміємо який процес відбувається всередині продукту

Statement testing and coverage

Ця техніка передбачає створення тестів, які покривають як можна більше дій в коді програми

Decision testing and coverage

Передбачає створення тестів, які покривають як можна більше переходів/рішень між діями.

Experience based testing

Тестування, що базується на попередньому досвіді

Error Guessing

Передбачення помилок. Техніка базується на попередньому досвіді, на тому як програма працювала раніше і які помилки були у схожих програмах

Exploratory testing

Використовується коли мало документації і загалом не зрозумілі вимоги до продукту і небагато часу. Тобто ми самостійно починаємо досліджувати/розбиратись в роботі програми і починаємо тестувати

Check list based

Тестування проводиться на базі тих чек-лістів, які були створені раніше. Просто тестуємо програму і доповнюємо їх