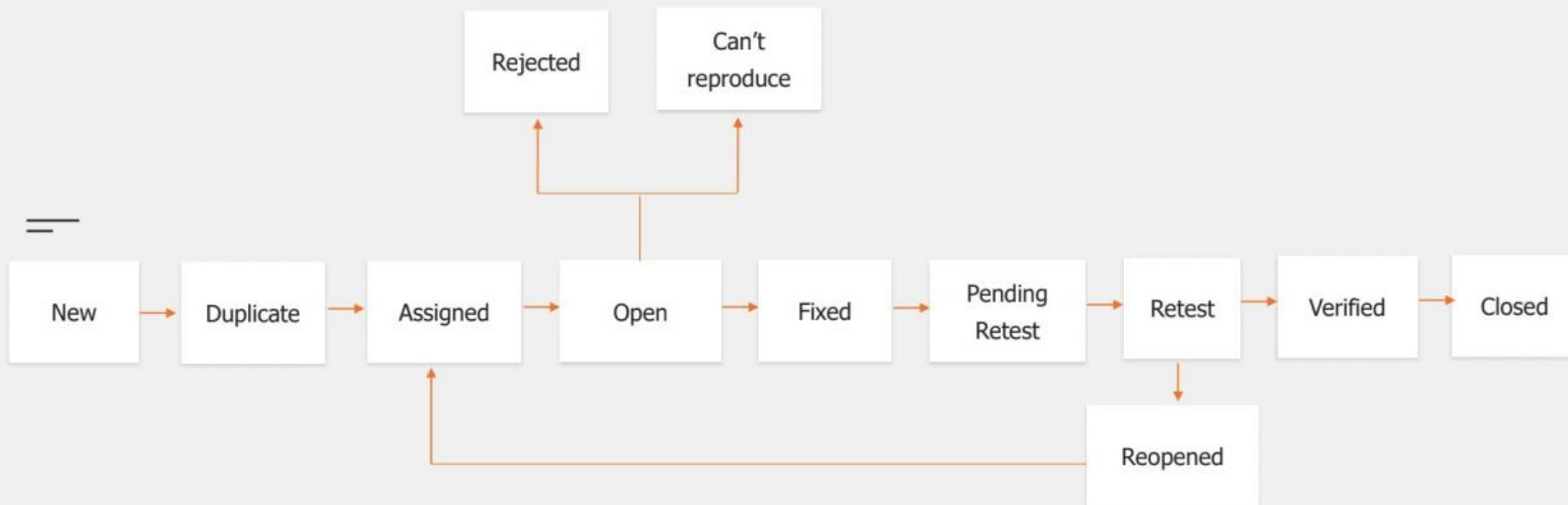


Сім принципів тестування

ЖИТТЄВИЙ цикл багу



Тестування показує наявність дефектів, а не їх відсутність

Тестування виявляє наявність дефектів, але не стверджує, що дефектів в ПЗ немає. При тестуванні знижується ризик присутності багів, які можуть бути у продукті, але все одно, якщо дефекти не були виявлені, тестування не доводить, що наше ПЗ написане без дефектів.

Вичерпне тестування неможливе

Тестування всього (всіх комбінацій вхідних даних та попередніх умов) неможливе, за винятком тривіальних випадків.

Тестування, що виконано раніше, заощаджує час та гроші

Щоб виявити дефекти на ранній стадії, дії з тестування повинні бути розпочаті якомога раніше в життєвому циклі розробки програмного забезпечення або системи, і повинні бути зосереджені на певних цілях.

Кластеризація дефектів (баги люблять дружити :))

Найчастіше менша частина функціоналу програмного забезпечення таїть у собі більшість помилок, дефектів, і невідповідностей з документацій. Якщо ми грамотно визначаємо мінімум найбільш значущих областей ПЗ та пріоритезуємо їх, то можемо з високою ймовірністю бути впевнені, що більшість критичних дефектів будуть знайдені в цьому мінімумі областей.

Парадокс пестициду

Якщо ті самі тести повторюються знову і знову, зрештою один і той же набір тест-кейсів більше не виявить жодних нових дефектів. Щоб подолати цей “парадокс пестицидів”, тест-кейси необхідно регулярно переглядати. Також потрібно писати нові тести для перевірки різних частин програмного забезпечення або системи, щоб знайти потенційного більше дефектів.

Тестування залежить від контексту

Тестування виконується по-різному, залежно від контексту. Наприклад, критично важливе для безпеки програмне забезпечення тестується не так, як сайт електронної комерції.

Омана про відсутність помилок

Виявлення та усунення дефектів не означає, що у системі більше немає дефектів. Наприклад, ретельне тестування всіх зазначених вимог та виправлення всіх виявлених дефектів може призвести до створення системи, яка буде важкою у використанні, не відповідатиме потребам та очікуванням користувачів або буде гіршою порівняно з іншими конкуруючими системами.