Słowniczek:

Testowanie statyczne

1. Podstawy testowania statycznego

Kandydat potrafi rozpoznać typy produktów pracy związanych z oprogramowaniem, które mogą być badane przy użyciu poszczególnych technik testowania statycznego. Kandydat potrafi opisać podając przykłady wartość testowania statycznego. Kandydat potrafi wyjaśnić różnicę między technikami testowania statycznego a technikami testowania dynamicznego z uwzględnieniem celów, typów identyfikowanych defektów oraz roli tych technik w cyklu życia oprogramowania.

1. Podstawy – testowanie statyczne (w odróżnieniu od dynamicznego) to testowanie bez uruchamiania systemu. Tutaj go w 99% przypadków OMAWIAMY, analizujemy, gdybamy, doszukujemy się błędów w logice. Robimy dużo spotkan, tworzymy dużo raportów itd.
2. Produkty pracy badane statycznie (czyli co tak możemy testować)
   1. Specyfikacje
   2. User Stories
   3. Architekturę systemu
   4. Kod źródłowy
   5. Testalia (ja pier…. – czyli wszystkie nasze TS i TC)
   6. Podręczniki użytkownika
   7. Stronę internetową produktu
   8. Umowy, plany
   9. Konfiguracje (kierownica – z prawej, z lewej)
   10. Diagramy UML
3. Ważna uwaga – przeglądy (review) mogą dotyczyń dowolnego z powyższej listy (w końcu pogadać można o wszystkim z tych rzeczy), ale analizę statyczną tylko tam, gdzie gadamy o konkretach (kod, architektura) i mamy jakieś dane pewne – np. ilość KM – o których chemy pogadać (review – tak sobie gadamy, inspekcja, analiza – mamy poważne spotkanie i musimy się trzymać planu i reguł)
4. Zalety testowania statycznego
   1. Często jak jesteśmy pogrążeni w technicznych aspektach tworzenia samochodu coś nam umyka, jesteśmy skupieni na szczególe a nie ogóle – wtedy dobrze spotkać się z zespołem i rzucić ogólne pytanie na temat kondycji systemu
   2. Jest tanim sposobem na szybkie wychwytywanie problemów (tanie bo gadając zapobiegamy temu, że ktoś coś źle zrozumiał i już rozpocznie z błędnym rozumowaniem pracę)
   3. Zapobiega defektom które wynikały by z niedociągnięć w projekcie
   4. Zwiększa efektywność pracy programostów – bo pozwala na wprowadzenie zasad dzięki którym komunikacja i ustalone systemy pracy (konwencje)
   5. Obniża koszty
   6. Usprawnia komunikację
5. Różnice między testowaniem dynamicznym a statycznym
   1. Mogą mieć te same cele ale inaczej się je wykonuje
   2. Dynamicznie – wykrywamy błędy już w uruchomionym systemie (ktoś już popełnił pomyłkę i wprowadził defekt)
   3. Statycznie – zapobiegamy wprowadzeniu błędów do systemu zanim ktoś popełni pomyłkę
6. Defekty usuwane przez testowane statyczne:
   1. Błędy w wymaganiach (klient źle wyraził o co mu chodzi)
   2. Błędy w projekcie (zapisaliśmy coś źle)
   3. Defekty w kodzie (oglądając oczami i czytając kod często znajdujemy pomyłki bez uruchamiania go)
   4. Odchylenia od standardów
   5. Złe specyfikacje interfejsów
   6. Słabe punkty zabezpieczeń
   7. Luki, nieścisłości
7. Dodatkowo – tylko! testowanie statyczne wykrywa defekty związane z utrzymywalnością kodu (modularyzacja, dobre praktyki itd. Itp. – omówię w nagraniu)
8. Proces przeglądów (Straszne gówno i są z tego pytania, kawa i jedziemy)

Kandydat potrafi podsumować czynności wykonywane w ramach procesu

przeglądu produktów pracy. Kandydat potrafi rozpoznać poszczególne role i obowiązki w przeglądzie formalnym. Kandydat potrafi wyjaśnić różnice między poszczególnymi typami przeglądów: przeglądem nieformalnym, przejrzeniem, przeglądem technicznym i inspekcją. Kandydat potrafi zastosować technikę przeglądu do produktu pracy w celu wykrycia defektów. Kandydat potrafi wyjaśnić, jakie czynniki decydują o powodzeniu przeglądu

1. W przeglądach w skrócie chodzi o to, by się spotykać (formalnie i nieformalnie) i omawiać nasz produkt na różnych etapach jego powstawania (przegląd ogólny, przegląd jak nam idzie z silnikiem, kołami, radiem – spotkania, spotkania, spotkania)
2. Celem przeglądów jest (jak wszystkiego tu) – wykrywanie defektów (statycznie), poszerzanie wiedzy, edukowanie uczestników, podejmowanie decyzji
3. Gdy planujemy zrobić przegląd to też ISTQB ma na to proces i kroki:
   1. Planowanie
      1. określamy co będziemy omawiać, przeglądamy i przygotowujemy dokumentację oraz ewentualnie metryki które pomogą nam w dyskusji
      2. Oszacowanie nakładu pracy i czasu ile potrwa przegląd
      3. Wskazujemy jak będzie przebiegał przegląd – kto co na nim będzie robił, przygotowujemy listę kontrolną itd.
      4. Wybieramy osoby które będą brały udział w przeglądzie
      5. W formalnych przeglądach – określamy kryteria wejścia – wyjścia (np. podczas inspekcji) – danych z którymi rozpoczynamy przegląd i danymi (celami) z którymi chcemy ten przegląd zakończyć.
   2. Rozpoczęcie przeglądu
      1. Rozsyłamy przygotowane informacje i formularze do uczestników
      2. Wyjaśniamy zakres przeglądu, role uczestników i co chcemy osiągnąć
      3. Odpowiadamy na pytania uczestników
   3. Przegląd indywidualny (indywidualne przygotowanie uczestnika)
      1. Przeglądamy co dostaliśmy do przejrzenia w związku z przeglądem
      2. Notujemy sobie swoje uwagi i ewentualne pytania
   4. Przekazanie informacji o problemach i ich analiza
      1. Na spotkaniu mówimy uczestnikom o znalezionych przez nas problemach
      2. Wspólnie analizujemy ww. defekty – wyznaczamy kto je rozwiąże
      3. Oceniamy rozmiar tych defektów, dokumentujemy jak będziemy sprawdzać jakość poprawek
      4. Oceniamy wnioski z przeglądu pod kątem kryteriów wyjścia (tego co przed spotkaniem założyliśmy sobie jako nasz cel)
   5. Usunięcie defektów i raportowanie
      1. Tworzymy raporty – co naprawiliśmy po spotkaniu
      2. Informujemy zainteresowanych o usunięciu defektu
      3. Aktualizujemy status defektu w narzędziu (np. Bugtrackerze)
      4. Zbieramy miary (jakość po wprowadzeniu poprawki)
      5. Sprawdzamy czy zmiana spełnia kryteria wyjścia z przeglądu ( co sobie założyliśmy na spotkaniu)
      6. Akceptujemy zmianę jak wszystko się zgadza
4. Role w przeglądzie (o to było dokładne pytanie w teście)
   1. Autor (Programista, mechanik)
   2. Kierownik (Szef programisty, mechanika – będzie mu przydzielał czas)
   3. Moderator (Osoba pilnująca czasu spotkania i byśmy nie gadali o pierdołach)
   4. Lider przeglądu (on odpowiada za przegląd, decyduje gdzie i kiedy się spotkamy)
   5. Przeglądający (inni programiści, mechanicy wnoszący jakiś wkład merytoryczny)
   6. Protokolant (Ktoś to musi zapisać)
5. Typy przeglądów
   1. Nieformalny
      1. Nie musi spełniać powyższych założeń i listy uczestników
      2. Spotykamy się w małym gronie osób zainteresowanych
      3. Szybko omawiamy problem, możemy ale nie musimy kończyć go pisemną dokumentacją lub wnioskami
   2. Przejrzenie (Walkthrough)
      1. Nie musimy przyjść przygotowani
      2. Spotkanie prowadzi Autor chcący nam coś pokazać/wyjaśnić
      3. Nie musimy mieć listy kontrolnej
      4. Musi być protokolant (spotkanie musi się skończyć jakimś pisemnym podsumowaniem)
      5. Bywa różny – od nieformalnego do bardzo formalnego
      6. Mechanik wprowadził nową zmianę w silniku – chce pokazać ją krok po kroku kolegom, żeby upewnić się, że o niczym nie zapomniał lub nie popełnił błędów. Chce mieć na końcu potwierdzenie pisemne, że każdy był, widział i akceptuje to co zrobił
   3. Przegląd techniczny
      1. Duże ważne spotkanie ważnych ludzi
      2. Uczestnicy to głównie osoby techniczne mające wiedzę w omawianym temacie
      3. Musi być protokolant
      4. Listy kontrolne – opcjonalne
      5. Każdy przed spotkaniem musi się przygotować
   4. Inspekcja
      1. Najgrubsze z grubych spotkań
      2. Najbardziej formalne
      3. Wszystko zgodnie z procesem opisanym wyżej (planowanie, rozsyłanie, zapraszanie, protokołowanie itd.)
      4. Każdy musi się przygotować
      5. Autor, Kierownik, Lider, Moderator, inni przeglądający (osoby techniczne), Protokolant
      6. Tworzymy dokument na końcu, obowiązują kryteria wejścia i wyjścia
6. Techniki przeglądu
   1. Ad hoc
      1. Mamy mało wskazówek co przeglądamy
      2. Spontaniczne
      3. Zależy od umiejętności przeglądających
      4. Ryzyko zgłaszania wielokrotnie tych samych problemów
   2. Oparty na liście kontrolnej
      1. Mamy liste kontrolną
      2. Odhaczamy co działa co nie działa
      3. Zaleta – dzięki liście mamy pokrycie (15 punktów, 10 ok 5 zle – 66%) i namacalne raporty
      4. Zaleta – systematyka przeglądu, zawsze sprawdzamy pod tym samym kątem
   3. Scenariusze i przebiegi próbne
      1. Otrzymujemy ustrukturyzowane wytyczne (…) mówiące – jak będziemy oglądać i przeglądać produkt.
      2. Dzięki scenariuszon – skupiamy się na przeglądzie na tym jak badana rzecz ma działać, co ma robić (taka walidacja)
      3. Dzięki scenariuszon – tak jak wyżej w liście kontrolnej – mamy systematykę i powatarzalne rezultaty
   4. Czytanie oparte na perspektywie
      1. Bawimy się w podział ról
      2. Ja jestem klientem, ty autorem – on wie co robi produkt
      3. Ja jestem doświadczonym użytkownikiem, ty świeżakiem, on jest starszym człowiekiem ona młodszym – każdy omawia produkt z tej perspektywy
      4. Każdy omawia problem ze swojej perspektywy
      5. Fajne, efektywne, rzuca nowe spojrzenie na produkt
7. Czynniki powodzenia związane z przeglądami – co robić by przegląd był udany i dał nam wiele wniosków
   1. Organizacyjne:
      1. Każdy przegląd ma swoje jednoznaczne cele
      2. Stosujemy różne techniki przeglądów
      3. Dbamy o to, by listy kontrolne były aktualne
      4. Planujemy przeglądy z wyprzedzeniem
   2. Kadrowe:
      1. W przeglądach uczestniczą eksperci i świeżaki
      2. Uczestnicy angażują się w przeglądy
      3. Testerzy są uważani za ważnych członków przeglądów
      4. Moderator dba o zwięzłość spotkania
      5. Protokolant zapisuje zwięźle wnioski i znalezione defekty
      6. Atmosfera – miła, ludzie uśmiechnięci