

Certyfikowany tester
Przykładowy egzamin
poziomu podstawowego

Wersja 2.3 2015

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

Spis treści

PYTANIA.....	4
Pytania "Podstawy testowania"	5
Pytania "Testowanie w cyklu życia"	7
Pytania "Testowanie statyczne"	9
Pytania "Techniki projektowania testów"	10
Pytania "Zarządzanie testowaniem"	17
Pytania "Testowane wspomagane narzędziami"	20
ODPOWIEDZI I UZASADNIENIA.....	22
Odpowiedzi "Podstawy testowania"	23
Odpowiedzi "Testowanie w cyklu życia"	28
Odpowiedzi "Testowanie statyczne"	32
Odpowiedzi "Techniki projektowania testów"	34
Odpowiedzi "Zarządzanie testowaniem"	45
Odpowiedzi "Testowane wspomagane narzędziami"	50

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

Wstęp

To jest przykładowy egzamin. Ma on na celu pomóc kandydatom przygotować się do egzaminu. Dostarcza zbiór pytań, których format jest podobny do PRAWDZIWEGO egzaminu na certyfikat ISTQB® Certyfikowany Tester Poziomu Podstawowego.

Nie wolno używać tych pytań podczas PRAWDZIWEGO EGZAMINU.

1. Dowolna osoba lub firma szkoleniowa może używać tego Przykładowego Egzaminu podczas kursu szkoleniowego, o ile jest potwierdzone, że ISTQB® jest źródłem i ma prawa własności do tych materiałów.
2. Dowolna osoba lub grupa osób może używać Przykładowego Egzaminu jako podstawy do artykułów, książek lub innych podobnych pism, o ile jest potwierdzone, że ISTQB® jest źródłem i ma prawa własności do tych materiałów.
3. Każda Rada Krajowa (National Board) uznawana przez ISTQB może przetłumaczyć ten Przykładowy Egzamin i go udostępniać, o ile o ile jest potwierdzone, że ISTQB® jest źródłem i ma prawa własności do tych materiałów.
4. Dla wszystkich pytań tylko jedna odpowiedź jest prawidłowa chyba, że w pytaniu jest zapisane inaczej.

Informacje ogólne:

Liczba pytań: 40

Czas trwania egzaminu: 60 minut

Sumaryczna ilość punktów: 40

Ilość punktów zaliczająca egzamin: 26 (lub więcej)

Procentowy poziom zaliczenia egzaminu: 65% (lub więcej)

Wersja 2014

Tłumaczenie: Magdalena Ślęzak, Piotr Ślęzak

Przegląd: Lucjan Stapp

Wersja 2.3 2015

Tłumaczenie: Lucjan Stapp

Przegląd: Adam Roman, Jan Sabak, Karolina Zmitrowicz

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

PYTANIA

Pytania "Podstawy testowania"

Pytanie 1 K1

Które z poniższych zdań NAJLEPIEJ opisuje jedną z siedmiu podstawowych zasad testowania?

- a) Testy automatyczne są lepsze niż testy manualne, gdy chcemy uniknąć testów gruntownych.
- b) Testowanie gruntowne jest, przy odpowiednim wysiłku i narzędziach, wykonalne dla każdego oprogramowania.
- c) Jest po prostu niemożliwe przetestowanie wszystkich kombinacji wejścia/wyjścia dla aplikacji softwarowej.
- d) Celem testowania jest wykazanie braku usterek.

Pytanie 2 K1

Które z poniższych zdań opisuje NAJWAŻNIEJSZY cel zespołu testowego?

- a) Rozstrzygnięcie, czy wykonano dostatecznie wiele testów modułów w czasie testów systemowych.
- b) Spowodowanie tak wielu awarii jak to tylko możliwe, aby można było zidentyfikować i naprawić usterki.
- c) Udowodnienie, że wszystkie usterki zostały zidentyfikowane.
- d) Udowodnienie, że pozostawione usterki nie spowodują żadnej awarii.

Pytanie 3 K1

Które z poniższych zadań będzie – Twoim zdaniem – wykonywane podczas analizy i projektowania testów według podstawowego procesu testowego?

- a) Zdefiniowanie celów testowania.
- b) Przeglądanie podstawy testów.
- c) Tworzenie zestawów testów z procedur testowych.
- d) Analizowanie wniosków z poprzednich projektów dla naprawy procesu.

Pytanie 4 K2

Poniżej podana jest lista problemów, które można zaobserwować podczas testowania lub pracy operacyjnej. Który z nich jest awarią?

- a) Produkt zepsuł się, gdy użytkownik wybrał opcję w okienku dialogowym.
- b) Plik z kodem źródłowym dołączony do nowej wersji (build) był w złej wersji.
- c) Algorytm używał złych danych wejściowych.
- d) Deweloper nieprawidłowo zinterpretował wymaganie algorytmu.

Pytanie 5 K1

Która z poniższych sytuacji lub akcji – o ile zostanie zauważona podczas przeglądów lub testów – może prowadzić do problemów (lub konfliktu) w mieszanym zespole programistów i testerów?

- a) Testerzy i programiści są dociekliwi i skupieni na znajdowaniu defektów.
- b) Testerzy i programiści są dostatecznie wykwalifikowani, by znajdować usterki i awarie.
- c) Testerzy i programiści przedstawiają defekty jako krytykę osób, a nie produktu.
- d) Testerzy spodziewają się, że w oprogramowaniu mogą być defekty, których programiści nie znaleźli i nie naprawili.

Pytanie 6 K2

Które z poniższych zdań są PRAWDZIWE?

- A. Testerzy mogą być zobowiązani do przetestowania oprogramowania pod kątem wymagań prawnych lub kontraktowych.
 - B. Testowanie oprogramowania jest głównie potrzebne by poprawić jakość efektów pracy programistów.
 - C. Rygorystyczne testowanie i naprawa znalezionych usterek może pomóc w zmniejszaniu ryzyka pojawiania się problemów w środowisku produkcyjnym.
 - D. Rygorystyczne testowanie jest czasami stosowane by udowodnić, że wszystkie awarie zostały wykryte.
- a) A, B i C są prawdą; D są fałszem.
 - b) A jest prawdą; B, C i D są fałszem.
 - c) A i C są prawdą; B i D są fałszem.
 - d) C i D są prawdą; A i B są fałszem.

Pytanie 7 K2

Które z poniższych zdań poprawnie opisuje różnicę pomiędzy testowaniem a debugowaniem?

- a) Testowanie namierza i identyfikuje źródło usterek. Debugowanie analizuje awarie i proponuje działania prewencyjne.
- b) Testowanie dynamiczne ujawnia awarie spowodowane usterekami. Debugowanie odszukuje, analizuje i usuwa powody awarii z oprogramowania.
- c) Testowanie usuwa usterki. Debugowanie identyfikuje przyczyny awarii.
- d) Testowanie dynamiczne zapobiega przyczynom awarii. Debugowanie usuwa awarie.

Pytania "Testowanie w cyklu życia"

Pytanie 8 K1

Które z poniższych zdań NAJLEPIEJ opisuje testowanie нефункционалне?

- a) Testowanie нефункционалне to proces testowania zintegrowanego systemu by sprawdzić, czy spełnia on wyspecyfikowane wymagania.
- b) Testowanie нефункционалне to proces testowania sprawdzający zgodność systemu ze standardami kodowania.
- c) Testowanie нефункционалне to testowanie bez odwoływania się do wewnętrznej struktury systemu.
- d) Testowanie нефункционалне to testowanie takich атрибутów systemu jak użyteczność, niezawodność czy pielęgnowalność.

Pytanie 9 K1

Co jest istotne podczas pracy z modelem wytwarzania oprogramowania?

- a) Jeżeli to konieczne, dostosować model do kontekstu projektu i cech produktu.
- b) Wybrać model wodospadowy (kaskadowy), ponieważ to najbardziej sprawdzony model.
- c) Rozpocząć z modelem V, a potem przejść do modelu iteracyjnego lub przyrostowego.
- d) Zmienić strukturę organizacji, by dostosować się do modelu, a nie odwrotnie.

Pytanie 10 K1

Które z poniższych jest cechą dobrego testowania i stosuje się w dowolnym modelu cyklu życia wytwarzania oprogramowania?

- a) Testy akceptacyjne to zawsze ostatni stosowany poziom testów.
- b) Wszystkie poziomy testów są zaplanowane i wykonywane dla każdej cechy wytwórczej.
- c) Testerzy są angażowani, gdy tylko pierwszy fragment kodu może być wykonany.
- d) Dla każdej aktywności wytwórczej istnieje odpowiadająca jej działalność testowa.

Pytanie 11 K1

Które z poniższych jest przykładem testowania pielęgnacyjnego?

- a) Sprawdzenie, czy poprawiono defekty podczas tworzenia nowego systemu.
- b) Sprawdzanie rozszerzeń dla działającego w środowisku docelowym systemu.
- c) Rozstrzyganie skarg dotyczących jakości systemu, zgłaszanych podczas testów akceptacyjnych.
- d) Integracja funkcji podczas tworzenia nowego systemu.

Pytanie 12 K2

Które z poniższych zdań są PRAWDZIWE? [K2]

- A. Testowanie regresywne i retestowanie to to samo.
 - B. Testy regresji pokazują, czy wszystkie wykryte awarie zostały rozwiązane.
 - C. Testy regresji są dobrym kandydatem do automatyzacji.
 - D. Testy regresji wykonuje się, by wykryć defekty będące wynikiem zmian w programie.
 - E. Testy regresji nie powinny być wykonywane podczas testów integracyjnych.
-
- a) A i B są prawdą
 - b) A, C i E są prawdą
 - c) C i D są prawdą
 - d) B, D i E są prawdą

Pytanie 13 K2

Które z poniższych porównań testowania modułowego i testowania systemowego są PRAWDZIWE?

- a) Testy modułowe sprawdzają działanie modułów, obiektów lub klas, które można testować w izolacji, podczas gdy testowanie systemowe sprawdza interfejsy pomiędzy modułami i interakcje różnych części systemu.
- b) Przypadki testowe dla testowania modułowego są zwykle wyprowadzane ze specyfikacji modułów, specyfikacji projektowej lub z modelu danych, podczas gdy przypadki testowe dla testów systemowych są zwykle wyprowadzane ze specyfikacji wymagań, specyfikacji funkcjonalnej lub przypadków użycia.
- c) Testowanie modułowe ogranicza się tylko do atrybutów funkcjonalnych, podczas gdy testowanie systemowe ukierunkowane jest na cechy funkcjonalne i niefunkcjonalne.
- d) Za testowanie modułowe na ogół odpowiadają testerzy, podczas gdy za testy systemowe na ogół odpowiadają użytkownicy systemu.

Pytania "Testowanie statyczne"

Pytanie 14 K1

Które z poniższych są głównymi fazami przeglądu formalnego?

- a) Inicjacja, status, indywidualne przygotowanie, spotkanie przeglądowe, poprawki, sprawdzenie.
- b) Planowanie, indywidualne przygotowanie, spotkanie przeglądowe, poprawki, zamknięcie, sprawdzenie.
- c) Planowanie, rozpoczęcie, przygotowanie indywidualne, spotkanie przeglądowe, poprawki, sprawdzenie.
- d) Indywidualne przygotowanie, spotkanie przeglądowe, poprawki, zamknięcie, sprawdzenie, analiza przyczynowo – skutkowa.

Pytanie 15 K2

Który z poniższych typów przeglądów NAJLEPIEJ nadaje się do przeglądania modułów krytycznych ze względu na bezpieczeństwo w projekcie software'owym?

- a) Przegląd nieformalny.
- b) Przegląd koleżeński.
- c) Inspekcja.
- d) Przegląd.

Pytanie 16 K1

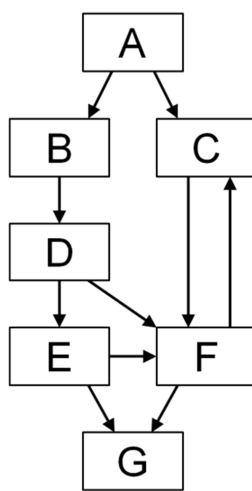
Które z poniższych zdań o analizie statycznej jest FAŁSZEM?

- a) Wspomagana narzędziami analiza statyczna może być używana jako środek zapobiegawczy w odpowiednim procesie.
- b) Wspomagana narzędziami analiza statyczna może znaleźć usterki, które nie dadzą się łatwo wykryć w testowaniu dynamicznym.
- c) Wspomagana narzędziami analiza statyczna skutkuje ograniczeniem kosztów przez wczesne znajdowanie usterek.
- d) Wspomagana narzędziami analiza statyczna to dobra metoda na wymuszanie awarii w oprogramowaniu.

Pyania "Techniki projektowania testów"

Pytanie 17 K4

Jednym z celów testowania w projekcie jest osiągnięcie 100% pokrycia decyzji. Następujące trzy testy zostały wykonane dla grafu przepływu sterowania pokazanego na poniższym rysunku.



Test α pokrywa ścieżkę: A, B, D, E, G.

Test β pokrywa ścieżkę: A, B, D, E, F, G.

Test γ pokrywa ścieżkę: A, C, F, C, F, C, F, G.

Które z poniższych zdań odnoszących się do celu pokrycia decyzji jest poprawne?

- a) Decyzja D nie jest w pełni przetestowana.
- b) Osiągnięto 100% pokrycia decyzji.
- c) Decyzja E nie jest w pełni przetestowana.
- d) Decyzja F nie jest w pełni przetestowana.

Pytanie 18 K3

Podczas testowania znaleziono usterkę: system padł podczas ściągania danych klienta z serwera. Usterka została naprawiona przez poprawienie kodu, który sprawdzał dostępność sieci podczas transferu danych. Istniejące przypadki testowe pokrywały 100% instrukcji w istniejącym module. Kilka nowych testów zostało zaprojektowanych, dodanych do zestawu testowego i wykonanych, by sprawdzić poprawkę i zapewnić bardziej obszerne pokrycie.

Jakie typy testów opisane są w powyższym akapicie?

- A. Testowanie funkcjonalne.
 - B. Testowanie strukturalne.
 - C. Retestowanie.
 - D. Testowanie wydajnościowe.
-
- a) A i B, ale nie C i D.
 - b) A i C, ale nie B i D.
 - c) A, B i C, ale nie D.
 - d) B, C i D, ale nie A.

Uzasadnienie

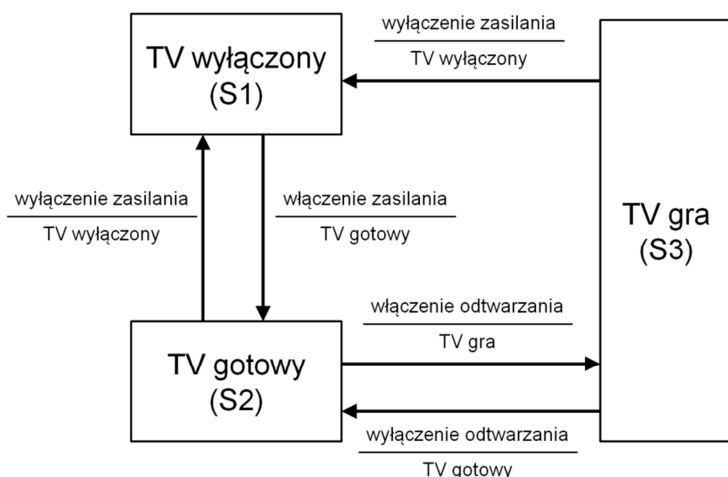
- A. Jest prawdziwe. Ściąganie danych klienta z serwera to typowe testowanie funkcjonalne.
- B. Jest prawdziwe. W opisie problemu jest powiedziane: „*Istniejące przypadki testowe pokrywały 100% instrukcji w istniejącym module*”, testowanie instrukcji to testy strukturalne.
- C. Jest prawdziwe. W opisie problemu jest powiedziane: „*By sprawdzić poprawkę i zapewnić bardziej obszerne pokrycie, kilka nowych testów zostało zaprojektowanych, dodanych do zestawu testowego i wykonanych*”, a więc mamy do czynienia z retestami.
- D. Jest nieprawdziwe. W opisie problemu nie ma mowy o testach wydajnościowych.

Stąd

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) POPRAWNE
- d) NIEPOPRAWNE

Pytanie 19 K3

Które z poniższych zdań o danym diagramie stanów i tablicy przypadków testowych jest PRAWDZIWE?



Przypadek testowy	1	2	3	4	5
Stan początkowy	S1	S2	S2	S3	S3
Wejście	Włączenie zasilania	Wyłączenie zasilania	Włączenie odtwarzania	Wyłączenie odtwarzania	Wyłączenie zasilania
Wynik oczekiwany	TV jest gotowy	TV wyłączony	TV gra	TV jest gotowy	TV wyłączony
Stan końcowy	S2	S1	S3	S2	S1

- Zaprojektowane przypadki testowe mogą być wykorzystane do pokrycia poprawnych i niepoprawnych przejść na diagramie stanów.
- Zaprojektowane przypadki testowe pokrywają wszystkie możliwe poprawne przejścia na diagramie stanów.
- Zaprojektowane przypadki testowe pokrywają tylko wybrane przejścia na diagramie stanów.
- Zaprojektowane przypadki testowe reprezentują sekwencje par przejść między stanami na diagramie stanów.

Pytanie 20 K2

Które z poniższych zdań są PRAWDZIWE w odniesieniu do techniki podziału na klasy równoważności?

- A. Podział możliwych wejść na klasy, w taki sposób, że dla wszystkich elementów danej klasy spodziewamy się podobnego zachowania systemu.
- B. Wykorzystuje zarówno poprawne jak i niepoprawne klasy równoważności.
- C. Musi używać w testach przynajmniej dwóch wartości z każdej klasy równoważności.
- D. Może być używana do testowania klas równoważności wejść wprowadzanych tylko z Graficznego Interfejsu Użytkownika.

- a) A, B i D są prawdą; C jest fałszem.
- b) A jest prawdą; B, C i D są fałszem.
- c) B i C są prawdą; A i D są fałszem.
- d) A i B są prawdą; C i D są fałszem.

Pytanie 21 K1

Która z poniższych list technik projektowania testów może być w całości zakwalifikowana jako techniki czarnoskrzynkowe?

- a) Podział na klasy równoważności, testowanie w oparciu o tablicę decyzyjną, testowanie w oparciu o przejścia pomiędzy stanami, analiza wartości brzegowych.
- b) Podział na klasy równoważności, testowanie w oparciu o tablicę decyzyjną, pokrycie instrukcji, testowanie w oparciu o przypadki użycia.
- c) Podział na klasy równoważności, pokrycie decyzji, testowanie w oparciu o przypadki użycia.
- d) Podział na klasy równoważności, pokrycie decyzji, analiza wartości brzegowych.

Pytanie 22 K3

Należy wyliczyć premię pracownika. Premia nie może być ujemna, ale może być wyliczona na zero. Premia wyliczana jest w oparciu o staż pracy. Są następujące kategorie zatrudnienia: zatrudniony na okres nie większy niż 2 lata; więcej niż 2 ale mniej niż 5; 5 do 10 lat; więcej niż 10 lat. W zależności od okresu zatrudnienia, pracownik otrzymuje odmienny poziom premii. Ile poprawnych klas równoważności jest potrzebnych do testowania obliczeń premii?

- a) 3
- b) 5
- c) 2
- d) 4

Pytanie 23 K2

Które z poniższych zdań opisujących korzyści z projektowania przypadków testowych z przypadków użycia jest prawdą, a które fałszem?

- A. Projektowanie przypadków testowych z przypadków użycia wspomaga testy systemowe i testy akceptacyjne.
- B. Projektowanie przypadków testowych z przypadków użycia wspomaga tylko testy automatyczne.
- C. Projektowanie przypadków testowych z przypadków użycia wspomaga testy modułowe.
- D. Projektowanie przypadków testowych z przypadków użycia wspomaga testowanie integracji.

- a) A i D są prawdą; B i C są fałszem.
- b) A jest prawdą; B, C i D są fałszem.
- c) A i B są prawdą; C i D są fałszem.
- d) C jest prawdą; A, B i D są fałszem.

Pytanie 24 K1

Która z poniższych opcji jest najlepszą podstawą do testowania atakiem usterek?

- a) Doświadczenie, dane o usterkach i awariach, wiedza o awariach oprogramowania.
- b) Identyfikacja ryzyka wykonana na początku projektu.
- c) Przypadki użycia wyprowadzone z przepływów biznesowych przez ekspertów dziedzinowych.
- d) Wyniki testów i porównanie z wynikami oczekiwanymi uzyskanymi z istniejącego systemu.

Pytanie 25 K1

Pracujesz w projekcie, w którym jest uboga specyfikacja i duża presja czasu. Które spośród poniższych jest najlepszym podejściem do testów w takiej sytuacji?

- a) Testowanie w oparciu o przypadki użycia.
- b) Pokrycie instrukcji.
- c) Testowanie eksploracyjne.
- d) Testowanie decyzji.

Pytanie 26 K1

Która z poniższych technik jest techniką opartą na strukturze?

- a) Testowanie decyzji.
- b) Analiza wartości brzegowych.
- c) Podział na klasy równoważności.
- d) Testowanie przejść pomiędzy stanami.

Pytanie 27 K3

Rozpoczynasz testowanie programu w oparciu o specyfikację. Program `calcNWD (A, B)` wylicza największy wspólny dzielnik (NWD) dwóch liczb całkowitych A i B, większych od zero.

Zdefiniowano następujące przypadki testowe (TC):

TC	A	B
1	1	1
2	INT_MAX	INT_MAX
3	1	0
4	0	1
5	INT_MAX - 1	1
6	1	INT_MAX - 1

gdzie INT_MAX jest największą liczbą całkowitą. Która technika testowa została zastosowana do zdefiniowania przypadków testowych 1 - 6?

- a) Analiza wartości brzegowych.
- b) Testowanie przejść między stanami.
- c) Testowanie w oparciu o przypadki użycia.
- d) Testowanie w oparciu o tablicę decyzyjną.

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

Pytanie 28 K3

Pracownik organizacji otrzymuje premię, jeżeli pracuje dłużej niż jeden rok i osiągnął uprzednio uzgodniony z nim cel.

Do testowania systemu zaprojektowana została następująca tablica decyzyjna:

		T1	T2	T3	T4	T5	T6	T7	T8
WARUNKI									
W1	Zatrudnienie ponad 1 rok	TAK	NIE	TAK	NIE	TAK	NIE	TAK	NIE
W2	Uzgodniono cel?	NIE	NIE	TAK	TAK	NIE	NIE	TAK	TAK
W3	Osiągnięto cel?	NIE	NIE	NIE	NIE	TAK	TAK	TAK	TAK
AKCJA									
A1	Wypłata premii?	NIE	NIE	NIE	NIE	NIE	NIE	TAK	NIE

Które z przypadków testowych można wyeliminować z powyższej tablicy decyzyjnej ze względu na niemożliwość ich zajścia?

- a) T1 i T2.
- b) T3 i T4.
- c) T7 i T8.
- d) T5 i T6.

Pytania "Zarządzanie testowaniem"

Pytanie 29 K1

Które z poniższych NAJLEPIEJ opisuje podział zadań pomiędzy liderem testów a testerem?

- a) Lider testów planuje aktywności testowe i wybiera odpowiednie standardy postępowania, podczas gdy tester wybiera narzędzia, które będą użyte i sprawuje nad nimi nadzór.
- b) Lider testów planuje, organizuje i zarządza aktywnościami testowymi, podczas gdy tester specyfikuje i wykonuje testy.
- c) Lider testów planuje, monitoruje i zarządza aktywnościami testowymi, podczas gdy tester projektuje testy i decyduje o zatwierdzeniu kryteriów zakończenia.
- d) Lider testów planuje i organizuje testowanie oraz specyfikuje przypadki testowe, podczas gdy tester priorytetyzuje i wykonuje testy.

Pytanie 30 K2

Które z poniższych może zostać zakwalifikowane jako ryzyko produktowe?

- a) Niska jakość wymagań, projektu, kodu i testów.
- b) Problemy polityczne, a także opóźnienia - głównie w złożonych obszarach produktu.
- c) Obszary podatne na błędy, potencjalne szkody dla użytkownika, słabe parametry jakościowe produktu.
- d) Problemy w definiowaniu poprawnych wymagań, obszary możliwych awarii w oprogramowaniu lub w systemie.

Pytanie 31 K2

Która z poniższych są typowymi kryteriami zakończenia testów?

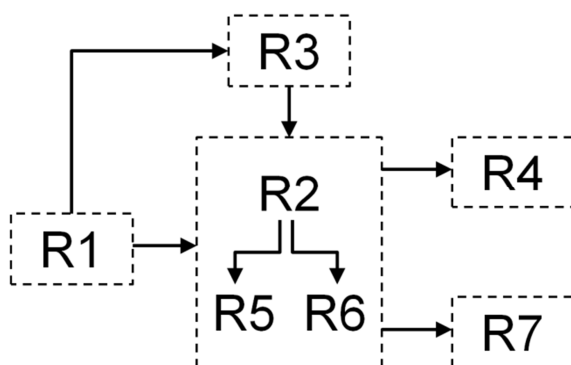
- a) Miary pokrycia testowego, miary niezawodności, koszt testów, harmonogram, stan naprawy usterek i pozostałe ryzyko.
- b) Miary pokrycia testowego, miary niezawodności, stopień niezależności testerów i złożoność produktu.
- c) Miary pokrycia testowego, miary niezawodności, koszt testów, dostęp do testowalnego kodu, czas do wypuszczenia produktu na rynek i kompletność (gotowość) produktu.
- d) Czas do wypuszczenia produktu na rynek, pozostałe usterek, kwalifikacje testerów, stopień niezależności testerów, miary pokrycia testowego oraz koszt testów.

Pytanie 32 K3

Jako lider testów masz następujące wymagania do przetestowania:

- R1 – Nieprawidłowości procesu
- R2 – Synchronizacja
- R3 – Potwierdzenie
- R4 – Problemy
- R5 – Dane finansowe
- R6 – Diagram danych
- R7 – Zmiany profilu użytkownika

Wymagania są logicznie zależne ($A \rightarrow B$ oznacza, że B jest zależne od A).



Jaki harmonogram wykonywania testów zaproponujesz zgodnie z zależnościami pomiędzy wymaganiami?

- a) $R3 > R2 > R1 > R7 > R5 > R6 > R4$
- b) $R2 > R5 > R6 > R4 > R7 > R1 > R3$
- c) $R1 > R3 > R2 > R5 > R6 > R4 > R7$
- d) $R1 > R2 > R5 > R6 > R3 > R4 > R7$

Pytanie 33 K1

Jaka jest korzyść z niezależnego testowania?

- a) Zostanie wykonane więcej pracy, bo testerzy nie przeszkadzają deweloperom przez cały czas.
- b) Niezależni testerzy starają się być bezstronni i znajdują inne usterki niż deweloperzy.
- c) Niezależni testerzy nie potrzebują dodatkowego wykształcenia i treningu.
- d) Niezależni testerzy redukują wąskie gardło w procesie zarządzania incydentami.

Pytanie 34 K1

Które z poniższych może być zakwalifikowane jako ryzyko projektowe?

- a) Braki umiejętności i niedobór personelu.
- b) Niedostateczne właściwości oprogramowania (np. użyteczność).
- c) Podatność dostarczonego oprogramowania na awarie.
- d) Możliwe problemy niezawodności (usterki).

Pytanie 35 K2

Poproszono Cię – jako lidera testów – o końcowy raport z testów. Ograniczając się do aktywności testowych, co powinno być NAJWAŻNIEJSZĄ informacją zawartą w Twoim raporcie, zgodnie z normą IEEE 829?

- a) Liczba wykonanych przypadków testowych i ich wyniki.
- b) Podsumowanie podstawowych aktywności testowych, zdarzeń i ich status w stosunku do spełnienia zamierzonych celów.
- c) Całkowite oszacowanie każdego elementu pracy deweloperów.
- d) Szkolenia odbyte przez członków zespołu testowego, które wspierały wysiłek testowy.

Pytanie 36 K3

Jesteś testerem w projekcie tworzącym oprogramowanie krytyczne ze względu na bezpieczeństwo. Podczas wykonywania testów stwierdziłeś/aś, że jeden z oczekiwanych wyników nie został osiągnięty. Pisziesz raport o incydencie, który miał miejsce. Jak sądzisz, jaka będzie NAJBARDZIEJ istotna informacja w tym raporcie zgodnie z normą IEEE 829?

- a) Wpływ, opis incydentu, data i czas, twoje nazwisko.
- b) Unikalny identyfikator zgłoszenia, specjalne wymagania i dane osoby, która wykryła defekt.
- c) Przekazywane elementy (ang. transmitted items), twoje nazwisko i twoje odczucia nt. prawdopodobnego źródła błędu.
- d) Opis incydentu, środowisko, oczekiwane wyniki testów.

Pytania "Testowane wspomagane narzędziami"

Pytanie 37 K1

Z poniższej listy wybierz zalecane zasady wprowadzania w organizacji wybranego narzędzia testowego

- A. Wdróż narzędzie w całej organizacji w tym samym czasie.
- B. Rozpocznij od projektu pilotażowego.
- C. Dostosuj i popraw procesy, by dopasować użycie narzędzia.
- D. Zapewnij szkolenia i wsparcie dla nowych użytkowników.
- E. Niech każdy zespół określi własne metody używania narzędzia.
- F. Monitoruj, czy koszty nie przekroczyły początkowych kosztów nabycia.
- G. Zbieraj wnioski z użycia narzędzia od wszystkich zespołów.

- a) A, C, D, E
- b) B, E, F
- c) B, C, D, G
- d) A, F, G

Pytanie 38 K1

Które z poniższych NAJLEPIEJ charakteryzuje narzędzia do wykonywania testów w oparciu o słowa kluczowe?

- a) Tablica zawierająca dane wejściowe do testów, słowa opisujące akcje oraz wartości oczekiwane sprawdzające działanie testowanego systemu.
- b) Działania testerów nagrane w skrypcie, który jest wykonywany wielokrotnie.
- c) Działania testerów nagrane w skrypcie, który jest wykonywany wielokrotnie z różnymi zbiorami testowych danych wejściowych.
- d) Możliwość logowania wyników i porównania ich z wynikami oczekiwanymi, przechowywanymi w zbiorze tekstowym.

Pytanie 39 K1

Które z poniższych NIE jest celem Projektu Pilotażowego do oceny narzędzia?

- a) Oszacowanie, na ile narzędzie pasuje do istniejących procesów i praktyk.
- b) Określenie sposobu użycia, zarządzania, przechowywania i pielęgnacji narzędzia i testaliów.
- c) Oszacowanie, czy korzyści będą osiągnięte przy rozsądnych kosztach.
- d) Redukcja odsetka wykrytych usterek w Projekcie Pilotażowym.

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

Pytanie 40 K2

Poniżej dana jest lista celów poprawy efektywności testów, które to cele zamierza osiągnąć organizacja wytwarzająca i testująca oprogramowanie. Który z tych celów najlepiej wspomogą narzędzia do zarządzania testowaniem?

- a) Budowanie powiązań pomiędzy wymaganiami, testami i błędami.
- b) Optymalizacja zdolności testów do wykrywania usterek.
- c) Szybsze rozwiązywanie usterek.
- d) Automatyzacja wyboru tych przypadków testowych, które będą wykonane.

ODPOWIEDZI I UZASADNIENIA

Odpowiedzi "Podstawy testowania"

Pytanie 1 K1

Które z poniższych zdań NAJLEPIEJ opisuje jedną z siedmiu podstawowych zasad testowania?

- a) Testy automatyczne są lepsze niż testy manualne, gdy chcemy uniknąć testów gruntownych.
- b) Testowanie gruntowne jest, przy odpowiednim wysiłku i narzędziach, wykonalne dla każdego oprogramowania.
- c) Jest po prostu niemożliwe przetestowanie wszystkich kombinacji wejścia/wyjścia dla aplikacji softwarowej.
- d) Celem testowania jest wykazanie braku usterek.

Uzasadnienie

- a) NIEPOPRAWNE - Testy automatyczne nie zawsze są lepsze niż manualne, a często są wręcz niemożliwe do wykonania.
- b) NIEPOPRAWNE – Nie da się wykonać testów gruntownych, niezależnie od nakładu pracy włożonego w testy (zasada 2).
- c) POPRAWNE – W zasadzie 2 jest powiedziane: Przetestowanie wszystkiego (wszystkich kombinacji wejść i warunków początkowych) jest niewykonalne poza trywialnymi przypadkami”.
- d) NIEPOPRAWNE – To zdanie jest sprzeczne z zasadą 1: „Testowanie może pokazać, że istnieją usterki, ale nie może dowieść, że oprogramowanie nie posiada defektów”.

Pytanie 2 K1

Które z poniższych zdań opisuje NAJWAŻNIEJSZY cel zespołu testowego?

- a) Rozstrzygnięcie, czy wykonano dostatecznie wiele testów modułów w czasie testów systemowych.
- b) Spowodowanie tak wielu awarii jak to tylko możliwe, aby można było zidentyfikować i naprawić usterki.
- c) Udowodnienie, że wszystkie usterki zostały zidentyfikowane.
- d) Udowodnienie, że pozostawione usterki nie spowodują żadnej awarii.

Uzasadnienie

- a) NIEPOPRAWNE - Testy modułów nie są częścią testów systemowych.
- b) POPRAWNE – To jest podstawowe zadanie zespołu testowego.
- c) NIEPOPRAWNE – W zasadzie 1 stwierdza się, że testy gruntowne są niemożliwe, więc nigdy nie można dowieść, że wszystkie usterki zostały zidentyfikowane.
- d) NIEPOPRAWNE – By oszacować, czy usterka będzie powodowała awarię czy nie, należy ją najpierw wykryć. Stwierdzenie, że pozostawione usterki nie spowodują awarii implikuje fakt, że wszystkie usterki zostały wykryte, co jest sprzeczne z zasadą 1.

Pytanie 3 K1

Które z poniższych zadań będzie – Twoim zdaniem – wykonywane podczas analizy i projektowania testów według podstawowego procesu testowego?

- a) Zdefiniowanie celów testowania.
- b) Przeglądanie podstawy testów.
- c) Tworzenie zestawów testów z procedur testowych.
- d) Analizowanie wniosków z poprzednich projektów dla naprawy procesu.

Uzasadnienie

- a) NIEPOPRAWNE – To zadanie jest wykonywane podczas planowania testów (p. 1.4.1).
- b) POPRAWNE – To zadanie jest wykonywane podczas analizy i projektowania testów (p. 1.4.2).
- c) NIEPOPRAWNE – To zadanie jest wykonywane podczas implementacji i wykonania testów (p. 1.4.3).
- d) NIEPOPRAWNE – To zadanie jest wykonywane podczas czynności wykonywanych na zakończenie testowania (p. 1.4.5).

Pytanie 4 K2

Poniżej podana jest lista problemów, które można zaobserwować podczas testowania lub pracy operacyjnej. Który z nich jest awarią?

- a) Produkt zepsuł się, gdy użytkownik wybrał opcję w okienku dialogowym.
- b) Plik z kodem źródłowym dołączony do nowej wersji (build) był w złej wersji.
- c) Algorytm używał złych danych wejściowych.
- d) Deweloper nieprawidłowo zinterpretował wymaganie algorytmu.

Uzasadnienie

- a) POPRAWNE – Awaria to uzewnętrznienie się usterki. Zepsucie się produktu jest łatwo zauważalne przez użytkownika.
- b) NIEPOPRAWNE – Ten typ usterki nie musi koniecznie prowadzić do widocznej lub zauważalnej awarii, na przykład jeżeli zmiany w nowej wersji źródłowej dotyczą tylko komentarzy.
- c) NIEPOPRAWNE – Użycie niepoprawnych zmiennych wejściowych nie musi koniecznie prowadzić do widocznej lub zauważalnej awarii. Na przykład, jeżeli nikt nie używa tego konkretnego algorytmu lub źle użyte zmienne wejściowe mają podobne wartości do poprawnych danych wejściowych lub nikt nie używa złych wyników algorytmu. „Usterki w oprogramowaniu mogą prowadzić do wystąpienia awarii, ale nie wszystkie usterki powodują awarie”
- d) NIEPOPRAWNE – Ten typ pomyłki nie musi koniecznie prowadzić do widocznej lub zauważalnej awarii. Na przykład, gdy nikt nie używa tego konkretnego algorytmu.

Pytanie 5 K1

Która z poniższych sytuacji lub akcji – o ile zostanie zauważona podczas przeglądów lub testów – może prowadzić do problemów (lub konfliktu) w mieszanym zespole programistów i testerów?

- a) Testerzy i programiści są dociekliwi i skupieni na znajdowaniu defektów.
- b) Testerzy i programiści są dostatecznie wykwalifikowani, by znajdować usterki i awarie.
- c) Testerzy i programiści przedstawiają defekty jako krytykę osób, a nie produktu.
- d) Testerzy spodziewają się, że w oprogramowaniu mogą być defekty, których programiści nie znaleźli i nie naprawili.

Uzasadnienie

- a) NIEPOPRAWNE – To nie jest sytuacja prowadząca do konfliktu. Testerzy i programiści powinni skupiać się na znajdowaniu usterek.
- b) NIEPOPRAWNE – Testerzy i programiści powinni być dostatecznie wykwalifikowani, by znajdować defekty i awarie. Nie jest to sytuacja prowadząca do konfliktu.
- c) POPRAWNE – zgodnie z sylabusem testerzy i programiści powinni współpracować, a przedstawianie defektów jako krytyki osób może prowadzić do konfliktów.
- d) NIEPOPRAWNE – Rolą testera w zespole jest szukanie tych defektów w oprogramowaniu, które nie zostały znalezione i poprawione przez programistów. Przedstawiona sytuacja jest jak najbardziej naturalna i pożądana.

Pytanie 6 K2

Które z poniższych zdań są PRAWDZIWE?

- A. Testerzy mogą być zobowiązani do przetestowania oprogramowania pod kątem wymagań prawnych lub kontraktowych.
- B. Testowanie oprogramowania jest głównie potrzebne by poprawić jakość efektów pracy programistów.
- C. Rygorystyczne testowanie i naprawa znalezionych usterek może pomóc w zmniejszaniu ryzyka pojawiania się problemów w środowisku produkcyjnym.
- D. Rygorystyczne testowanie jest czasami stosowane by udowodnić, że wszystkie awarie zostały wykryte.

- a) A, B i C są prawdą; D są fałszem.
- b) A jest prawdą; B, C i D są fałszem.
- c) A i C są prawdą; B i D są fałszem.
- d) C i D są prawdą; A i B są fałszem.

Uzasadnienie

- A. POPRAWNE – Testowanie oprogramowania może polegać na sprawdzaniu wymagań prawnych i kontraktowych.
- B. POPRAWNE – Testowanie oprogramowania jest głównie potrzebne, by poprawić jakość efektów pracy programistów.
- C. POPRAWNE – Jednym z głównych celów testowania oprogramowania jest zmniejszanie ryzyka pojawiania się problemów w środowisku produkcyjnym.
- D. NIEPOPRAWNE – Nie można udowodnić, że wszystkie awarie zostały wykryte.

Stąd:

- a) POPRAWNE
- b) NIEPOPRAWNE
- c) NIEPOPRAWNE
- d) NIEPOPRAWNE

Pytanie 7 K2

Które z poniższych zdań poprawnie opisuje różnicę pomiędzy testowaniem a debugowaniem?

- a) Testowanie namierza i identyfikuje źródło usterki. Debugowanie analizuje awarie i proponuje działania prewencyjne.
- b) Testowanie dynamiczne ujawnia awarie spowodowane usterekami. Debugowanie odszukuje, analizuje i usuwa powody awarii z oprogramowania.
- c) Testowanie usuwa usterki. Debugowanie identyfikuje przyczyny awarii.
- d) Testowanie dynamiczne zapobiega przyczynom awarii. Debugowanie usuwa awarie.

Uzasadnienie

- a) NIEPOPRAWNE – Testowanie nie namierza ani nie identyfikuje źródła usterki.
- b) POPRAWNE – Testowanie dynamiczne pokazuje awarie spowodowane usterekami. Debugowanie odszukuje, analizuje i usuwa przyczyny awarii z oprogramowania.
- c) NIEPOPRAWNE – Testowanie nie usuwa usterek.
- d) NIEPOPRAWNE – Testowanie dynamiczne nie zapobiega powodom awarii.

Odpowiedzi "Testowanie w cyklu życia"

Pytanie 8 K1

Które z poniższych zdań NAJLEPIEJ opisuje testowanie нефункционалне?

- a) Testowanie нефункционалне to процес testowania zintegrowanego systemu by sprawdzić, czy spełnia on wyspecyfikowane wymagania.
- b) Testowanie нефункционалне to процес testowania sprawdzający zgodność systemu ze standardami kodowania.
- c) Testowanie нефункционалне to testowanie bez odwoływania się do wewnętrznej struktury systemu.
- d) Testowanie нефункционалне to testowanie takich атрибутов systemu jak użyteczność, niezawodność czy pielęgnownalność.

Uzasadnienie

- a) NIEPOPRAWNE – To jest definicja testowania systemowego.
- b) NIEPOPRAWNE – To wchodzi w skład zadań analizy statycznej.
- c) NIEPOPRAWNE – To jest definicja testowania czarnoskrzynkowego.
- d) POPRAWNE – Testowanie takich атрибутов systemu jak użyteczność, niezawodność czy pielęgnownalność to testowanie нефункционалне.

Pytanie 9 K1

Co jest istotne podczas pracy z modelem wytwarzania oprogramowania?

- a) Jeżeli to konieczne, dostosować model do kontekstu projektu i cech produktu.
- b) Wybrać model wodospadowy (kaskadowy), ponieważ to najbardziej sprawdzony model.
- c) Rozpocząć z modelem V, a potem przejść do modelu iteracyjnego lub przyrostowego.
- d) Zmienić strukturę organizacji, by dostosować się do modelu, a nie odwrotnie.

Uzasadnienie

- a) POPRAWNE – Modele dostarczają ogólnych wskazówek – nie musi być to dokładny процес krok po kroku, który trzeba stosować literalnie.
- b) NIEPOPRAWNE – Model wodospadowy (kaskadowy) jest jednym z możliwych modeli, który zespół może wybrać.
- c) NIEPOPRAWNE – Model V nie jest zgodny z modelem iteracyjnym, więc opisanie przejście nie ma sensu.
- d) NIEPOPRAWNE – Modele są wybierane by dostosować się do sytuacji i projektu a nie odwrotnie.

Pytanie 10 K1

Które z poniższych jest cechą dobrego testowania i stosuje się w dowolnym modelu cyklu życia wytwarzania oprogramowania?

- a) Testy akceptacyjne to zawsze ostatni stosowany poziom testów.
- b) Wszystkie poziomy testów są zaplanowane i wykonywane dla każdej cechy wytwórczej.
- c) Testerzy są angażowani, gdy tylko pierwszy fragment kodu może być wykonany.
- d) Dla każdej aktywności wytwórczej istnieje odpowiadająca jej działalność testowa.

Uzasadnienie

- a) NIEPOPRAWNE – To jest poprawne tylko dla projektów, w których są testy akceptacyjne. Istnieją projekty, w których nie ma tego poziomu testów.
- b) NIEPOPRAWNE – Istnieją sytuacje, w których nie wszystkie poziomy testów są konieczne. Na przykład, gdy otrzymano kod od firmy trzeciej, testy modułowe mogą nie być konieczne.
- c) NIEPOPRAWNE – Testerzy powinni być angażowani dużo wcześniej, niż gdy kod staje się dostępny. Na przykład, testerzy mogą być angażowani w przeglądy specyfikacji wymagań.
- d) POPRAWNE – „Dla każdej czynności związanej z wytworzeniem oprogramowania istnieją odpowiadające jej czynności związane z testowaniem” (p. 2.1.3).

Pytanie 11 K1

Które z poniższych jest przykładem testowania pielęgnacyjnego?

- a) Sprawdzenie, czy poprawiono defekty podczas tworzenia nowego systemu.
- b) Sprawdzanie rozszerzeń dla działającego w środowisku docelowym systemu.
- c) Rozstrzyganie skarg dotyczących jakości systemu, zgłaszanych podczas testów akceptacyjnych.
- d) Integracja funkcji podczas tworzenia nowego systemu.

Uzasadnienie

- a) NIEPOPRAWNE – Testowanie nowego systemu to nie jest testowanie pielęgnacyjne.
- b) POPRAWNE – Testowanie możliwości działania systemu po zmianie środowiska jest przykładem testowania pielęgnacyjnego.
- c) NIEPOPRAWNE – Zajmowanie się awariami podczas testów akceptacyjnych to nie jest testowanie pielęgnacyjne.
- d) NIEPOPRAWNE – Integracja funkcji nie jest czynnością testową.

Pytanie 12 K2

Które z poniższych zdań są PRAWDZIWE? [K2]

- A. Testowanie regresywne i retestowanie to to samo.
 - B. Testy regresji pokazują, czy wszystkie wykryte awarie zostały rozwiązane.
 - C. Testy regresji są dobrym kandydatem do automatyzacji.
 - D. Testy regresji wykonuje się, by wykryć defekty będące wynikiem zmian w programie.
 - E. Testy regresji nie powinny być wykonywane podczas testów integracyjnych.
-
- a) A i B są prawdą; C, D i E są fałszem.
 - b) A, C i E są prawdą; B i D są fałszem.
 - c) C i D są prawdą; A, B i E są fałszem.
 - d) B, D i E są prawdą; A i C są fałszem.

Uzasadnienie

Zdanie A jest niepoprawne – **testowanie regresywne** to powtarzanie testów na już przetestowanym programie, po zmianach, w celu wykrycia usterek wprowadzonych lub ujawnionych w wyniku zmian w programie. **Retesty** wykonuje się, by potwierdzić, że wykryte defekty zostały usunięte (p. 2.3.4).

Zdanie B jest niepoprawne – opisuje retesty i nadzór nad testami.

Zdanie C jest poprawne – zestawy testów regresji można wykonywać wiele razy i w ogólności zmieniają się one wolno, więc testy regresji dobrze nadają się do automatyzacji (p. 2.3.4).

Zdanie D jest poprawne – to jest definicja testów regresji (p. 2.3.4).

Zdanie E jest niepoprawne – „Testowanie regresywne można wykonywać na wszystkich poziomach testów i dla wszystkich typów testów: funkcjonalnych, нефункциональных i strukturalnych” (p. 2.3.4).

Stąd:

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) POPRAWNE
- d) NIEPOPRAWNE

Pytanie 13 K2

Które z poniższych porównań testowania modułowego i testowania systemowego są PRAWDZIWE?

- a) Testy modułowe sprawdzają działanie modułów, obiektów lub klas, które można testować w izolacji, podczas gdy testowanie systemowe sprawdza interfejsy pomiędzy modułami i interakcje różnych części systemu.
- b) Przypadki testowe dla testowania modułowego są zwykle wyprowadzane ze specyfikacji modułów, specyfikacji projektowej lub z modelu danych, podczas gdy przypadki testowe dla testów systemowych są zwykle wyprowadzane ze specyfikacji wymagań, specyfikacji funkcjonalnej lub przypadków użycia.
- c) Testowanie modułowe ogranicza się tylko do atrybutów funkcjonalnych, podczas gdy testowanie systemowe ukierunkowane jest na cechy funkcjonalne i niefunkcjonalne.
- d) Za testowanie modułowe na ogół odpowiadają testerzy, podczas gdy za testy systemowe na ogół odpowiadają użytkownicy systemu.

Uzasadnienie

- a) NIEPOPRAWNE – Testowanie systemowe nie testuje interfejsów pomiędzy modułami i interakcji różnych części systemu; to jest zadanie testów integracyjnych.
- b) POPRAWNE (p.p. 2.2.1 i 2.2.3).
- c) NIEPOPRAWNE – Testowanie modułowe nie ogranicza się wyłącznie do charakterystyk funkcjonalnych.
- d) NIEPOPRAWNE – Za testowanie modułowe na ogół odpowiadają programiści, podczas gdy za testy systemowe na ogół odpowiadają testerzy.

Odpowiedzi "Testowanie statyczne"

Pytanie 14 K1

Które z poniższych są głównymi fazami przeglądu formalnego?

- a) Inicjacja, status, indywidualne przygotowanie, spotkanie przeglądowe, poprawki, sprawdzenie.
- b) Planowanie, indywidualne przygotowanie, spotkanie przeglądowe, poprawki, zamknięcie, sprawdzenie.
- c) Planowanie, rozpoczęcie, przygotowanie indywidualne, spotkanie przeglądowe, poprawki, sprawdzenie.
- d) Indywidualne przygotowanie, spotkanie przeglądowe, poprawki, zamknięcie, sprawdzenie, analiza przyczynowo – skutkowa.

Uzasadnienie

Główne fazy przeglądu formalnego to planowanie, rozpoczęcie, indywidualne przygotowanie, spotkanie przeglądowe, obróbka, sprawdzenie (p. 3.2.1).

Stąd:

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) POPRAWNE
- d) NIEPOPRAWNE

Pytanie 15 K2

Który z poniższych typów przeglądów NAJLEPIEJ nadaje się do przeglądania modułów krytycznych ze względu na bezpieczeństwo w projekcie software'owym?

- a) Przegląd nieformalny.
- b) Przegląd koleżeński.
- c) Inspekcja.
- d) Przejrzenie.

Uzasadnienie

Do przeglądów modułów krytycznych ze względu na bezpieczeństwo najbardziej pasuje formalny, udokumentowany przegląd, stąd:

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) POPRAWNE
- d) NIEPOPRAWNE

Pytanie 16 K1

Które z poniższych zdań o analizie statycznej jest FAŁSZEM?

- a) Wspomagana narzędziami analiza statyczna może być używana jako środek zapobiegawczy w odpowiednim procesie.
- b) Wspomagana narzędziami analiza statyczna może znaleźć usterki, które nie dadzą się łatwo wykryć w testowaniu dynamicznym.
- c) Wspomagana narzędziami analiza statyczna skutkuje ograniczeniem kosztów przez wczesne znajdowanie usterek.
- d) Wspomagana narzędziami analiza statyczna to dobra metoda na wymuszanie awarii w oprogramowaniu.

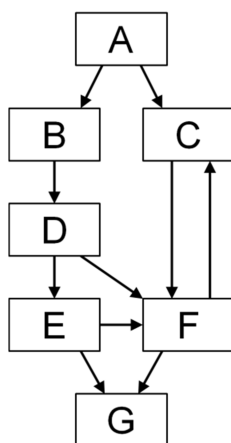
Uzasadnienie

- a) NIEPOPRAWNE – To zdanie jest prawdziwe, wspomagana narzędziami analiza statyczna może być używana jako środek zapobiegawczy.
- b) NIEPOPRAWNE - To zdanie jest prawdziwe, wspomagana narzędziami analiza statyczna może znaleźć usterki, które nie dadzą się wykryć w testowaniu dynamicznym.
- c) NIEPOPRAWNE – To zdanie jest prawdziwe, wspomagana narzędziami analiza statyczna skutkuje ograniczeniem kosztów przez wczesne znajdowanie usterek.
- d) POPRAWNE – Podczas analizy statycznej nie wykryje się żadnej awarii, ponieważ kod nie jest uruchamiany.

Odpowiedzi "Techniki projektowania testów"

Pytanie 17 K4

Jednym z celów testowania w projekcie jest osiągnięcie 100% pokrycia decyzji. Następujące trzy testy zostały wykonane dla grafu przepływu sterowania pokazanego na poniższym rysunku.



Test α pokrywa ścieżkę: A, B, D, E, G.

Test β pokrywa ścieżkę: A, B, D, E, F, G.

Test γ pokrywa ścieżkę: A, C, F, C, F, C, F, G.

Które z poniższych zdań odnoszących się do celu pokrycia decyzji jest poprawne?

- a) Decyzja D nie jest w pełni przetestowana.
- b) Osiągnięto 100% pokrycia decyzji.
- c) Decyzja E nie jest w pełni przetestowana.
- d) Decyzja F nie jest w pełni przetestowana.

Uzasadnienie

W diagramie mamy następujące 4 decyzje: A, D, E i F.

W teście α pokryte są przejścia $A \rightarrow B$, $D \rightarrow E$ oraz $E \rightarrow G$.

W teście β pokryte są przejścia $A \rightarrow B$, $D \rightarrow E$ oraz $F \rightarrow G$.

W teście γ pokryte są przejścia $A \rightarrow C$, $F \rightarrow C$ oraz $F \rightarrow G$.

Tym samym decyzja A jest pokryta w całości ($A \rightarrow B$ oraz $A \rightarrow C$), decyzja E jest pokryta w całości ($E \rightarrow G$ oraz $E \rightarrow F$), decyzja F jest pokryta w całości ($F \rightarrow C$ oraz $F \rightarrow G$). Decyzja D nie jest pokryta w całości, bo mamy pokryte tylko przejście $D \rightarrow E$, natomiast przejście $D \rightarrow F$ nie jest pokryte.

Stąd:

- a) POPRAWNE
- b) NIEPOPRAWNE
- c) NIEPOPRAWNE
- d) NIEPOPRAWNE

Pytanie 18 K3

Podczas testowania znaleziono usterkę: system padł podczas ściągania danych klienta z serwera. Usterka została naprawiona przez poprawienie kodu, który sprawdzał dostępność sieci podczas transferu danych. Istniejące przypadki testowe pokrywały 100% instrukcji w istniejącym module. Kilka nowych testów zostało zaprojektowanych, dodanych do zestawu testowego i wykonanych, by sprawdzić poprawkę i zapewnić bardziej obszerne pokrycie.

Jakie typy testów opisane są w powyższym akapicie?

- A. Testowanie funkcjonalne.
 - B. Testowanie strukturalne.
 - C. Retestowanie.
 - D. Testowanie wydajnościowe.
-
- a) A i B, ale nie C i D.
 - b) A i C, ale nie B i D.
 - c) A, B i C, ale nie D.
 - d) B, C i D, ale nie A.

Uzasadnienie

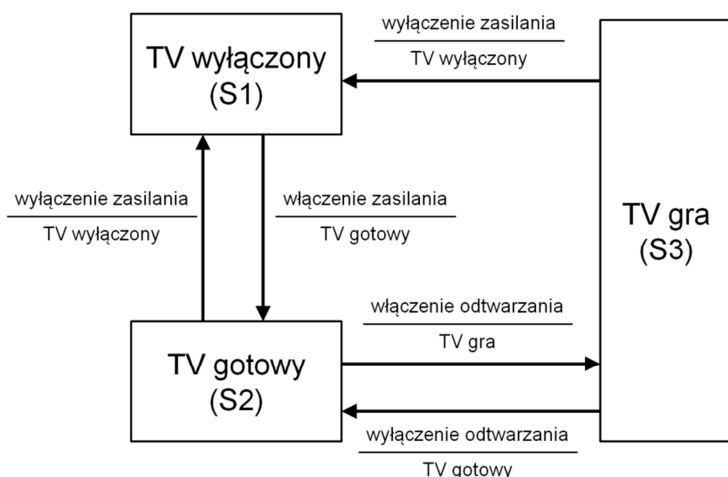
- A. Jest prawdziwe. Ściąganie danych klienta z serwera to typowe testowanie funkcjonalne.
- B. Jest prawdziwe. W opisie problemu jest powiedziane: „*Istniejące przypadki testowe pokrywały 100% instrukcji w istniejącym module*”, testowanie instrukcji to testy strukturalne.
- C. Jest prawdziwe. W opisie problemu jest powiedziane: „*By sprawdzić poprawkę i zapewnić bardziej obszerne pokrycie, kilka nowych testów zostało zaprojektowanych, dodanych do zestawu testowego i wykonanych*”, a więc mamy do czynienia z retestami.
- D. Jest nieprawdziwe. W opisie problemu nie ma mowy o testach wydajnościowych.

Stąd

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) POPRAWNE
- d) NIEPOPRAWNE

Pytanie 19 K3

Które z poniższych zdań o danym diagramie stanów i tablicy przypadków testowych jest PRAWDZIWE?



Przypadek testowy	1	2	3	4	5
Stan początkowy	S1	S2	S2	S3	S3
Wejście	Włączenie zasilania	Wyłączenie zasilania	Włączenie odtwarzania	Wyłączenie odtwarzania	Wyłączenie zasilania
Wynik oczekiwany	TV jest gotowy	TV wyłączony	TV gra	TV jest gotowy	TV wyłączony
Stan końcowy	S2	S1	S3	S2	S1

- a) Zaprojektowane przypadki testowe mogą być wykorzystane do pokrycia poprawnych i niepoprawnych przejść na diagramie stanów.
- b) Zaprojektowane przypadki testowe pokrywają wszystkie możliwe poprawne przejścia na diagramie stanów.
- c) Zaprojektowane przypadki testowe pokrywają tylko wybrane przejścia na diagramie stanów.
- d) Zaprojektowane przypadki testowe reprezentują sekwencje par przejść między stanami na diagramie stanów.

Uzasadnienie

Zaproponowane przypadki testowe pokrywają wszystkie 5 możliwych pojedynczych przejść na diagramie stanów (S1→S2, S2→S1, S2→S3, S3→S2, S3→S1)

Stąd:

- a) NIEPOPRAWNE
- b) POPRAWNE
- c) NIEPOPRAWNE
- d) NIEPOPRAWNE

Pytanie 20 K2

Które z poniższych zdań są PRAWDZIWE w odniesieniu do techniki podziału na klasy równoważności?

- A. Podział możliwych wejść na klasy, w taki sposób, że dla wszystkich elementów danej klasy spodziewamy się podobnego zachowania systemu.
- B. Wykorzystuje zarówno poprawne jak i niepoprawne klasy równoważności.
- C. Musi używać w testach przynajmniej dwóch wartości z każdej klasy równoważności.
- D. Może być używana do testowania klas równoważności wejść wprowadzanych tylko z Graficznego Interfejsu Użytkownika.

- a) A, B i D są prawdą; C jest fałszem.
- b) A jest prawdą; B, C i D są fałszem.
- c) B i C są prawdą; A i D są fałszem.
- d) A i B są prawdą; C i D są fałszem.

Uzasadnienie

Podział na klasy równoważności dzieli możliwe wejścia na klasy w ten sposób, by każdy element z klasy powodował to samo zachowanie się systemu. Używamy zarówno poprawnych jak i niepoprawnych klas. Technika wymaga, aby w testach użyć co najmniej jednej wartości z każdej klasy. Wprowadzanie danych z Graficznego Interfejsu Użytkownika jest możliwe, ale metoda podziału na klasy równoważności może testować wejścia pochodzące z innych niż GUI źródeł (np. przy użyciu API).

Stąd:

- A. jest prawdą.
- B. jest prawdą.
- C. jest fałszem.
- D. jest fałszem.

Tym samym

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) NIEPOPRAWNE
- d) POPRAWNE

Pytanie 21 K1

Która z poniższych list technik projektowania testów może być w całości zakwalifikowana jako techniki czarnoskrzynkowe?

- a) Podział na klasy równoważności, testowanie w oparciu o tablicę decyzyjną, testowanie w oparciu o przejścia pomiędzy stanami, analiza wartości brzegowych.
- b) Podział na klasy równoważności, testowanie w oparciu o tablicę decyzyjną, pokrycie instrukcji, testowanie w oparciu o przypadki użycia.
- c) Podział na klasy równoważności, pokrycie decyzji, testowanie w oparciu o przypadki użycia.
- d) Podział na klasy równoważności, pokrycie decyzji, analiza wartości brzegowych.

Uzasadnienie

- a) POPRAWNE – Wszystkie wymienione techniki są czarnoskrzynkowe (p. 4.3).
- b) NIEPOPRAWNE – Pokrycie instrukcji jest techniką białoskrzynkową (p. 4.4).
- c) NIEPOPRAWNE – Pokrycie decyzji jest techniką białoskrzynkową (p. 4.4).
- d) NIEPOPRAWNE – Pokrycie decyzji jest techniką białoskrzynkową (p. 4.4).

Pytanie 22 K3

Należy wyliczyć premię pracownika. Premia nie może być ujemna, ale może być wyliczona na zero. Premia wyliczana jest w oparciu o staż pracy. Są następujące kategorie zatrudnienia: zatrudniony na okres nie większy niż 2 lata; więcej niż 2 ale mniej niż 5; 5 do 10 lat; więcej niż 10 lat. W zależności od okresu zatrudnienia, pracownik otrzymuje odmienny poziom premii. Ile poprawnych klas równoważności jest potrzebnych do testowania obliczeń premii?

- a) 3
- b) 5
- c) 2
- d) 4

Uzasadnienie

- a) NIEPOPRAWNE – Patrz właściwy podział opisany w (d).
- b) NIEPOPRAWNE – Patrz właściwy podział opisany w (d).
- c) NIEPOPRAWNE – Patrz właściwy podział opisany w (d).
- d) POPRAWNE – Zgodnie z opisem w zadaniu podział jest następujący:
 - $0 < \text{staż pracy} \leq 2$
 - $2 < \text{staż pracy} < 5$
 - $5 \leq \text{staż pracy} \leq 10$
 - $\text{staż pracy} > 10$

Informacja o tym, że premia nie może być ujemna nie ma w tym przypadku żadnego znaczenia dla analizy, ponieważ podział dokonywany jest dla stażu pracy, a nie wysokości premii.

Pytanie 23 K2

Które z poniższych zdań opisujących korzyści z projektowania przypadków testowych z przypadków użycia jest prawdą, a które fałszem?

- A. Projektowanie przypadków testowych z przypadków użycia wspomaga testy systemowe i testy akceptacyjne.
- B. Projektowanie przypadków testowych z przypadków użycia wspomaga tylko testy automatyczne.
- C. Projektowanie przypadków testowych z przypadków użycia wspomaga testy modułowe.
- D. Projektowanie przypadków testowych z przypadków użycia wspomaga testowanie integracji.

- a) A i D są prawdą; B i C są fałszem.
- b) A jest prawdą; B, C i D są fałszem.
- c) A i B są prawdą; C i D są fałszem.
- d) C jest prawdą; A, B i D są fałszem.

Uzasadnienie

Zdanie A jest POPRAWNE – przypadki użycia opisują zwykle interakcje użytkowników z całym systemem, dlatego też najlepiej pasują do definiowania testów na poziomie systemowym. Co więcej, „Przypadki użycia bardzo przydają się w projektowaniu testów akceptacyjnych, w których ma brać udział klient/użytkownik” (p. 4.3.5).

Zdanie B jest FAŁSZEM – Przypadki użycia można testować zarówno ręcznie jak i automatycznie.

Zdanie C jest FAŁSZEM – na poziomie testów modułowych wyprowadzamy przypadki testowe tak, by zwiększyć pokrycie kodu. Wyprowadzamy je patrząc na kod, a nie w oparciu o definicję lub przypadek użycia, być może pozwalający sprawdzić pewien fragment kodu.

Zdanie D jest POPRAWNE – Przypadki użycia „pomagają również wykrywać defekty integracji spowodowane interakcją i interferencją różnych modułów” (p. 4.3.5).

Stąd:

- a) POPRAWNE
- b) NIEPOPRAWNE
- c) NIEPOPRAWNE
- d) NIEPOPRAWNE

Pytanie 24 K1

Która z poniższych opcji jest najlepszą podstawą do testowania atakiem usterek?

- a) Doświadczenie, dane o usterkach i awariach, wiedza o awariach oprogramowania.
- b) Identyfikacja ryzyka wykonana na początku projektu.
- c) Przypadki użycia wyprowadzone z przepływów biznesowych przez ekspertów dziedzinowych.
- d) Wyniki testów i porównanie z wynikami oczekiwanymi uzyskanymi z istniejącego systemu.

Uzasadnienie

- a) POPRAWNE – „Listy defektów i awarii można budować na podstawie doświadczenia, dostępnych danych na temat usterek i awarii oraz na ogólnej wiedzy dlaczego oprogramowanie nie działa” (p. 4.5).
- b) NIEPOPRAWNE – Identyfikacja ryzyka oznacza obszary lub własności na których trzeba się koncentrować, ale nie mówi jak je testować.
- c) NIEPOPRAWNE – Testowanie przepływów biznesowych nie koncentruje się na znanych słabościach w oprogramowaniu. Ono po prostu sprawdza pewne przypadki użycia.
- d) NIEPOPRAWNE – To zdanie opisuje jedną z możliwych sytuacji, gdy test przeszedł lub nie i nie ma nic wspólnego z atakiem usterek.

Pytanie 25 K1

Pracujesz w projekcie, w którym jest uboga specyfikacja i duża presja czasu. Które spośród poniższych jest najlepszym podejściem do testów w takiej sytuacji?

- a) Testowanie w oparciu o przypadki użycia.
- b) Pokrycie instrukcji.
- c) Testowanie eksploracyjne.
- d) Testowanie decyzji.

Uzasadnienie

- a) NIEPOPRAWNE – W projekcie mamy ubogą specyfikację, więc jest małe prawdopodobieństwo, że przypadki użycia istnieją.
- b) NIEPOPRAWNE – Testowanie instrukcji zajmuje dużo czasu, a w projekcie jest duża presja czasu.
- c) POPRAWNE – Testowanie eksploracyjne jest dobrym pomysłem, gdy jest uboga specyfikacja i duża presja czasu (p. 4.5).
- d) NIEPOPRAWNE – Testowanie decyzji zajmuje dużo czasu, a w projekcie jest duża presja czasu.

Pytanie 26 K1

Która z poniższych technik jest techniką opartą na strukturze?

- a) Testowanie decyzji.
- b) Analiza wartości brzegowych.
- c) Podział na klasy równoważności.
- d) Testowanie przejść pomiędzy stanami.

Uzasadnienie

- a) POPRAWNE – Testowanie decyzji jest techniką białoskrzynkową (p. 4.4).
- b) NIEPOPRAWNE – Analiza wartości brzegowych jest techniką czarnoskrzynkową (p. 4.3).
- c) NIEPOPRAWNE – Podział na klasy równoważności jest techniką czarnoskrzynkową (p. 4.3).
- d) NIEPOPRAWNE – Testowanie przejść między stanami jest techniką czarnoskrzynkową (p. 4.3).

Pytanie 27 K3

Rozpoczynasz testowanie programu w oparciu o specyfikację. Program `calcNWD (A, B)` wylicza największy wspólny dzielnik (NWD) dwóch liczb całkowitych `A` i `B`, większych od zera.

Zdefiniowano następujące przypadki testowe (TC):

TC	A	B
1	1	1
2	INT_MAX	INT_MAX
3	1	0
4	0	1
5	INT_MAX - 1	1
6	1	INT_MAX - 1

gdzie `INT_MAX` jest największą liczbą całkowitą. Która technika testowa została zastosowana do zdefiniowania przypadków testowych 1 - 6?

- a) Analiza wartości brzegowych.
- b) Testowanie przejść między stanami.
- c) Testowanie w oparciu o przypadki użycia.
- d) Testowanie w oparciu o tablicę decyzyjną.

Uzasadnienie

- a) POPRAWNE – Podane wartości (0,1,INT_MAX-1, INT_MAX) to typowe wartości brzegowe dla zmiennych będących dodatnimi liczbami całkowitymi.
- b) NIEPOPRAWNE – Patrz (a) dla uzasadnienia.
- c) NIEPOPRAWNE – Patrz (a) dla uzasadnienia.
- d) NIEPOPRAWNE – Patrz (a) dla uzasadnienia.

Pytanie 28 K3

Pracownik organizacji otrzymuje premię, jeżeli pracuje dłużej niż jeden rok i osiągnął uprzednio uzgodniony z nim cel.

Do testowania systemu zaprojektowana została następująca tablica decyzyjna:

		T1	T2	T3	T4	T5	T6	T7	T8
WARUNKI									
W1	Zatrudnienie ponad 1 rok	TAK	NIE	TAK	NIE	TAK	NIE	TAK	NIE
W2	Uzgodniono cel?	NIE	NIE	TAK	TAK	NIE	NIE	TAK	TAK
W3	Osiągnięto cel?	NIE	NIE	NIE	NIE	TAK	TAK	TAK	TAK
AKCJA									
A1	Wypłata premii?	NIE	NIE	NIE	NIE	NIE	NIE	TAK	NIE

Które z przypadków testowych można wyeliminować z powyższej tablicy decyzyjnej ze względu na niemożliwość ich zajścia?

- a) T1 i T2.
- b) T3 i T4.
- c) T7 i T8.
- d) T5 i T6.

Uzasadnienie

W przypadkach testowych zajście akcji zależy od zajścia kombinacji warunków. W przypadkach T5 i T6 opisana jest sytuacja osiągnięcia celu, który nie został wcześniej uzgodniony. Taka sytuacja nie może mieć miejsca. Stąd:

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) NIEPOPRAWNE
- d) POPRAWNE

Odpowiedzi "Zarządzanie testowaniem"

Pytanie 29 K1

Które z poniższych NAJLEPIEJ opisuje podział zadań pomiędzy liderem testów a testerem?

- a) Lider testów planuje aktywności testowe i wybiera odpowiednie standardy postępowania, podczas gdy tester wybiera narzędzia, które będą użyte i sprawuje nad nimi nadzór.
- b) Lider testów planuje, organizuje i zarządza aktywnościami testowymi, podczas gdy tester specyfikuje i wykonuje testy.
- c) Lider testów planuje, monitoruje i zarządza aktywnościami testowymi, podczas gdy tester projektuje testy i decyduje o zatwierdzeniu kryteriów zakończenia.
- d) Lider testów planuje i organizuje testowanie oraz specyfikuje przypadki testowe, podczas gdy tester priorytetyzuje i wykonuje testy.

Uzasadnienie

- a) NIEPOPRAWNE – wybór narzędzia to zadanie lidera testów (p. 5.1.2).
- b) POPRAWNE – patrz p. 5.1.2.
- c) NIEPOPRAWNE – Decyzja o zatwierdzeniu kryteriów zakończenia to zadanie lidera testów.
- d) NIEPOPRAWNE – Lider testów nie specyfikuje przypadków testowych.

Pytanie 30 K2

Które z poniższych może zostać zakwalifikowane jako ryzyko produktowe?

- a) Niska jakość wymagań, projektu, kodu i testów.
- b) Problemy polityczne, a także opóźnienia - głównie w złożonych obszarach produktu.
- c) Obszary podatne na błędy, potencjalne szkody dla użytkownika, słabe parametry jakościowe produktu.
- d) Problemy w definiowaniu poprawnych wymagań, obszary możliwych awarii w oprogramowaniu lub w systemie.

Uzasadnienie

- a) NIEPOPRAWNE – Niska jakość wymagań to ryzyko projektowe (p. 5.5.1).
- b) NIEPOPRAWNE – To są ryzyka projektowe.
- c) POPRAWNE – To są ryzyka produktowe (p. 5.5.2).
- d) NIEPOPRAWNE – Problemy z wymaganiami to ryzyka projektowe (p. 5.5.1).

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

Pytanie 31 K2

Która z poniższych są typowymi kryteriami zakończenia testów?

- a) Miary pokrycia testowego, miary niezawodności, koszt testów, harmonogram, stan naprawy usterek i pozostałe ryzyko.
- b) Miary pokrycia testowego, miary niezawodności, stopień niezależności testerów i złożoność produktu.
- c) Miary pokrycia testowego, miary niezawodności, koszt testów, dostęp do testowalnego kodu, czas do wypuszczenia produktu na rynek i kompletność (gotowość) produktu.
- d) Czas do wypuszczenia produktu na rynek, pozostałe usterek, kwalifikacje testerów, stopień niezależności testerów, miary pokrycia testowego oraz koszt testów.

Uzasadnienie

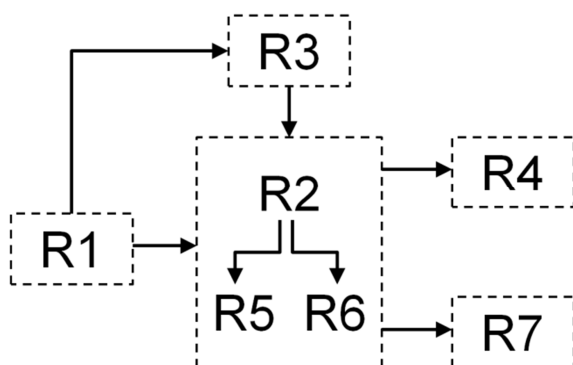
- a) POPRAWNE – Patrz sylabus (p. 5.5.1).
- b) NIEPOPRAWNE – Stopień niezależności testerów nie ma znaczenia jako kryterium wyjścia.
- c) NIEPOPRAWNE – Dostęp do testowalnego kodu to kryterium wejściowe.
- d) NIEPOPRAWNE – Zarówno stopień niezależności testerów jak i kwalifikacje testerów nie mają znaczenia jako kryteria wyjścia.

Pytanie 32 K3

Jako lider testów masz następujące wymagania do przetestowania:

- R1 – Nieprawidłowości procesu
- R2 – Synchronizacja
- R3 – Potwierdzenie
- R4 – Problemy
- R5 – Dane finansowe
- R6 – Diagram danych
- R7 – Zmiany profilu użytkownika

Wymagania są logicznie zależne ($A \rightarrow B$ oznacza, że B jest zależne od A).



Jaki harmonogram wykonywania testów zaproponujesz zgodnie z zależnościami pomiędzy wymaganiami?

- a) $R3 > R2 > R1 > R7 > R5 > R6 > R4$
- b) $R2 > R5 > R6 > R4 > R7 > R1 > R3$
- c) $R1 > R3 > R2 > R5 > R6 > R4 > R7$
- d) $R1 > R2 > R5 > R6 > R3 > R4 > R7$

Uzasadnienie

- a) NIEPOPRAWNE – Wszystko jest zależne od R1, więc każda sekwencja nie zaczynająca się od R1 jest niepoprawna.
- b) NIEPOPRAWNE - Wszystko jest zależne od R1, więc każda sekwencja nie zaczynająca się od R1 jest niepoprawna.
- c) POPRAWNE – Testy są ustawione zgodnie z zależnościami.
- d) NIEPOPRAWNE – R2 jest zależne od R3, więc R3 powinno być testowane przed R2.

Pytanie 33 K1

Jaka jest korzyść z niezależnego testowania?

- a) Zostanie wykonane więcej pracy, bo testerzy nie przeszkadzają deweloperom przez cały czas.
- b) Niezależni testerzy starają się być bezstronni i znajdują inne usterki niż deweloperzy.
- c) Niezależni testerzy nie potrzebują dodatkowego wykształcenia i treningu.
- d) Niezależni testerzy redukują wąskie gardło w procesie zarządzania incydentami.

Uzasadnienie

- a) NIEPOPRAWNE – Niezależność nie oznacza braku współpracy.
- b) POPRAWNE – To jest jeden z powodów niezależności.
- c) NIEPOPRAWNE – Testerzy potrzebują wiedzy i szkoleń niezależnie od tego, czy są testerami niezależnymi, czy nie.
- d) NIEPOPRAWNE – Nie ma związku pomiędzy niezależnością testerów a redukcją wąskiego gardła w procesie zarządzania incydentami.

Pytanie 34 K1

Które z poniższych może być zakwalifikowane jako ryzyko projektowe?

- a) Braki umiejętności i niedobór personelu.
- b) Niedostateczne właściwości oprogramowania (np. użyteczność).
- c) Podatność dostarczonego oprogramowania na awarie.
- d) Możliwe problemy niezawodności (usterki).

Uzasadnienie

- a) POPRAWNE – Braki umiejętności i niedobór personelu to typowe ryzyka projektowe.
- b) NIEPOPRAWNE – To jest ryzyko produktowe.
- c) NIEPOPRAWNE – To jest ryzyko produktowe.
- d) NIEPOPRAWNE – To jest ryzyko produktowe.

Pytanie 35 K2

Poproszono Cię – jako lidera testów – o końcowy raport z testów. Ograniczając się do aktywności testowych, co powinno być NAJWAŻNIEJSZĄ informacją zawartą w Twoim raporcie, zgodnie z normą IEEE 829?

- a) Liczba wykonanych przypadków testowych i ich wyniki.
- b) Podsumowanie podstawowych aktywności testowych, zdarzeń i ich status w stosunku do spełnienia zamierzonych celów.
- c) Całkowite oszacowanie każdego elementu pracy deweloperów.
- d) Szkolenia odbyte przez członków zespołu testowego, które wspierały wysiłek testowy.

Uzasadnienie

- a) NIEPOPRAWNE – Liczba wykonanych przypadków testowych i ich wyniki mogą być zawarte w końcowym raporcie z testów zgodnie z IEEE 829, ale to nie jest najważniejsza część raportu.
- b) POPRAWNE – Końcowy raport z testów musi zawierać informacje o podstawowych aktywnościach testowych, zdarzeniach i ich statusie w stosunku do spełnienia zamierzonych celów.
- c) NIEPOPRAWNE – Oszacowanie każdego elementu pracy deweloperów nie jest częścią końcowego raportu z testów.
- d) NIEPOPRAWNE – Szkolenia nie są istotne w końcowym raporcie z testów.

Pytanie 36 K3

Jesteś testerem w projekcie tworzącym oprogramowanie krytyczne ze względu na bezpieczeństwo. Podczas wykonywania testów stwierdziłeś/aś, że jeden z oczekiwanych wyników nie został osiągnięty. Pisziesz raport o incydencie, który miał miejsce. Jak sądzisz, jaka będzie NAJBARDZIEJ istotna informacja w tym raporcie zgodnie z normą IEEE 829?

- a) Wpływ, opis incydentu, data i czas, twoje nazwisko.
- b) Unikalny identyfikator zgłoszenia, specjalne wymagania i dane osoby, która wykryła defekt.
- c) Przekazywane elementy (ang. transmitted items), twoje nazwisko i twoje odczucia nt. prawdopodobnego źródła błędu.
- d) Opis incydentu, środowisko, oczekiwane wyniki testów.

Uzasadnienie

- a) POPRAWNE – Najważniejszą informacją, która musi być zawarta w raporcie o incydencie jest wpływ.
- b) NIEPOPRAWNE – Te informacje powinny być zawarte w raporcie o incydencie, ale nie są tak istotne jak wpływ (patrz (a)).
- c) NIEPOPRAWNE – Patrz (a), raport o incydencie powinien zawierać rzeczowe informacje, nie „odczucia” testerów nt. prawdopodobnego źródła błędu.
- d) NIEPOPRAWNE – Te informacje powinny być zawarte w raporcie o incydencie, ale nie są tak istotne jak wpływ (patrz (a)).

Odpowiedzi "Testowane wspomagane narzędziami"

Pytanie 37 K1

Z poniższej listy wybierz zalecane zasady wprowadzania w organizacji wybranego narzędzia testowego

- A. Wdróż narzędzie w całej organizacji w tym samym czasie.
- B. Rozpocznij od projektu pilotażowego.
- C. Dostosuj i popraw procesy, by dopasować użycie narzędzia.
- D. Zapewnij szkolenia i wsparcie dla nowych użytkowników.
- E. Niech każdy zespół określi własne metody używania narzędzia.
- F. Monitoruj, czy koszty nie przekroczyły początkowych kosztów nabycia.
- G. Zbieraj wnioski z użycia narzędzia od wszystkich zespołów.

- a) A, C, D, E
- b) B, E, F
- c) B, C, D, G
- d) A, F, G

Uzasadnienie

Zdanie A jest NIEPOPRAWNE – zaleca się zacząć od pilotażu przed wdrożeniem narzędzia w całej organizacji.

Zdanie B jest POPRAWNE – patrz powyżej.

Zdanie C jest POPRAWNE – „Ocena czy i jak narzędzie pasuje do obowiązujących procesów i praktyk oraz ustalenie, co ewentualnie należałoby zmienić” (p. 6.3).

Zdanie D jest POPRAWNE – Zapewnienie szkoleń jest jednym z czynników sukcesu przy wdrożeniu narzędzia (p. 6.3).

Zdanie E jest NIEPOPRAWNE – Jeżeli każdy będzie określał własne metody użycia narzędzia, to skończy się bałaganem. „Ustalenie standardów użycia” jest jednym z czynników sukcesu przy wdrożeniu narzędzia (p. 6.3).

Zdanie F jest NIEPOPRAWNE – Koszt wdrożenia narzędzia może być większy niż sam koszt zakupu. Nieuwzględnienie tego faktu to jedno z ryzyk związanych z wdrożeniem narzędzia (p. 6.1).

Zdanie G jest POPRAWNE – „Zbieranie wniosków z wykorzystania narzędzia przez wszystkie zespoły” to jeden z czynników sukcesu przy wdrożeniu narzędzia (p. 6.3).
Stąd:

- a) NIEPOPRAWNE
- b) NIEPOPRAWNE
- c) POPRAWNE
- d) NIEPOPRAWNE

Pytanie 38 K1

Które z poniższych NAJLEPIEJ charakteryzuje narzędzia do wykonywania testów w oparciu o słowa kluczowe?

- a) Tablica zawierająca dane wejściowe do testów, słowa opisujące akcje oraz wartości oczekiwane sprawdzające działanie testowanego systemu.
- b) Działania testerów nagrane w skrypcie, który jest wykonywany wielokrotnie.
- c) Działania testerów nagrane w skrypcie, który jest wykonywany wielokrotnie z różnymi zbiorami testowych danych wejściowych.
- d) Możliwość logowania wyników i porównania ich z wynikami oczekiwanymi, przechowywanymi w zbiorze tekstowym.

Uzasadnienie

- a) POPRAWNE – „W podejściu sterowanym słowami kluczowymi arkusz kalkulacyjny zawiera słowa kluczowe opisujące akcje do wykonania (nazywane również słowami akcji) oraz dane testowe.” (p. 6.2.2)
- b) NIEPOPRAWNE – To jest definicja automatyzacji w oparciu o skrypty testowe.
- c) NIEPOPRAWNE – To jest definicja automatyzacji testowania sterowanego danymi.
- d) NIEPOPRAWNE – To jest opisanie części tego, co robi struktura do automatyzacji testów.

Pytanie 39 K1

Które z poniższych NIE jest celem Projektu Pilotażowego do oceny narzędzia?

- a) Oszacowanie, na ile narzędzie pasuje do istniejących procesów i praktyk.
- b) Określenie sposobu użycia, zarządzania, przechowywania i pielęgnacji narzędzia i testaliów.
- c) Oszacowanie, czy korzyści będą osiągnięte przy rozsądnych kosztach.
- d) Redukcja odsetka wykrytych usterek w Projekcie Pilotażowym.

Uzasadnienie

- a) NIEPOPRAWNE – To zdanie jest prawdziwe, patrz sylabus p. 6.3.
- b) NIEPOPRAWNE – To zdanie jest prawdziwe, patrz sylabus p. 6.3.
- c) NIEPOPRAWNE – To zdanie jest prawdziwe, patrz sylabus p. 6.3.
- d) POPRAWNE – Redukcja odsetka wykrytych usterek nie jest celem Projektu Pilotażowego.

Certyfikowany tester Przykładowy egzamin poziomu podstawowego	International Software Testing Qualifications Board Stowarzyszenie Jakości Systemów Informatycznych
---	--

Pytanie 40 K2

Poniżej dana jest lista celów poprawy efektywności testów, które to cele zamierza osiągnąć organizacja wytwarzająca i testująca oprogramowanie. Który z tych celów najlepiej wspomogą narzędzia do zarządzania testowaniem?

- a) Budowanie powiązań pomiędzy wymaganiami, testami i błędami.
- b) Optymalizacja zdolności testów do wykrywania usterek.
- c) Szybsze rozwiązywanie usterek.
- d) Automatyzacja wyboru tych przypadków testowych, które będą wykonane.

Uzasadnienie

- a) POPRAWNE – Bo budowanie powiązań pomiędzy wymaganiami, testami i błędami jest funkcjonalnością narzędzia do zarządzania testowaniem (p. 6.1.3).
- b) NIEPOPRAWNE – Bo to nie jest możliwe w narzędziu do zarządzania testowaniem.
- c) NIEPOPRAWNE – Bo to nie jest rozwiązywane w narzędziu do zarządzania testowaniem.
- d) NIEPOPRAWNE – Bo wybór przypadków testowych nie jest wspomagany przez narzędzie do zarządzania testowaniem.