

PLAN TESTÓW – APLIKACJA MrBuggy 3

Tytuł: Plan testów do aplikacji MrBuggy 3

Wersja	Data	Autor	Komentarz
1.05.2001	14.03.2022	Klaudia Kurkowska	

Wprowadzenie

Niniejszy dokument zawiera plan testów aplikacji MrBuggy 3, którego celem jest określenie poprawności działania aplikacji poprzez przeprowadzenie testów, których zadaniem jest znalezienie jak największej liczby defektów. Aplikacja ta służy do przeprowadzenia i automatycznego sprawdzenia wyników pierwszej rundy Mistrzostw Polski w Testowaniu Oprogramowania.

Zakres testów

Testowanymi elementami aplikacji MrBuggy 3 będą:

- interfejs;
- wprowadzanie i zapis danych;
- wydajność aplikacji;
- bezpieczeństwo i niezawodność aplikacji.

Aplikacja testowana będzie na podstawie dokumentu http://mrbuggy.pl/mrbuggy3/dfiles/Specyfikacja_Mr_Buggy_3.pdf.

Aplikacja MrBuggy 3 zostanie przetestowana w następującym środowisku testowym:

- Windows 7;
- Windows 8/8.1.

Aplikację należy przetestować zarówno w przypadku działającego, jak i niedziałającego połączenia sieciowego. Przetestować należy wszystkie wersje aplikacji w części klienckiej oraz serwerowej.

Testowane cechy/funkcje

- wygląd interfejsu aplikacji oraz jego zgodność ze specyfikacją;
- funkcja raportowania defektów w różnych trybach aplikacji oraz ocena ich

- poprawności;
- generator i implementacja zadań;
- rejestracja czynności w dzienniku zdarzeń części serwerowej oraz prowadzenie klasyfikacji;
- wydajność aplikacji w oparciu o dostarczoną specyfikację;
- wymagania bezpieczeństwa aplikacji w oparciu o specyfikację;
- wymagania niezawodności.

Podejście testowe

Testy aplikacji MrBuggy 3 będą opierać się na dostarczonej specyfikacji (http://mrbuggy.pl/mrbuggy3/dfiles/Specyfikacja_Mr_Buggy_3.pdf). Przeprowadzone zostaną testy manualne, testy funkcjonalne oraz niefunkcjonalne, w tym testy wydajnościowe, testy bezpieczeństwa, testy integracyjne i testy eksploracyjne.

Kryteria zaliczenia oraz niezaliczenia testów

Testy zaliczane będą na podstawie zgodności z dostarczoną specyfikacją aplikacji MrBuggy3, po uzyskaniu wyników zgodnych z oczekiwanym rezultatem testów. Rozbieżności pomiędzy rezultatami oczekiwanymi i uzyskanymi skutkować będzie niezaliczeniem testu oraz zgłoszeniem błędu zgodnie z ich priorytetyzacją.

Priorytetyzacja błędów:

- **niski** – błędy kosmetyczne (np. literówki, błędy w nazwie);
- **średni** – błąd nieblokujący dalszej pracy aplikacji, występujący podczas typowego działania aplikacji;
- **wysoki** – jednoznaczny błąd nieblokujący dalszej pracy aplikacji, często występujący podczas typowego użytkowania aplikacji;
- **krytyczny**- jednoznaczny błąd, który uniemożliwia dalszą pracę lub korzystanie z kluczowej funkcji aplikacji.

Dla każdej funkcjonalności zostanie przygotowany przypadek testowy na podstawie informacji zawartych w specyfikacji aplikacji. Aplikacja musi przejść pozytywnie co najmniej 90% testów. Kryteria zaliczenia i niezaliczenia testów:

- co najmniej 90% przypadków testowych zostało wykonanych;
- co najmniej 80% błędów o priorytecie niskim i średnim została naprawiona;
- 100% błędów o priorytecie wysokim i krytycznym została naprawiona;
- wszystkie naprawione błędy muszą zostać przetestowane ponownie podczas testów

regresji.

Niespełnienie wyżej wymienionych kryteriów spowoduje, że testy nie zostaną zaliczone, ze względu na niespełnienie wymagań.

Kryteria zawieszenia i wznowienia testowania

Rozpoczęcie procesu testowego następuje po spełnieniu warunków wstępnych:

- zespół testerski zapoznał się ze specyfikacją aplikacji oraz planem testowania;
- działająca aplikacja została dostarczona przez deweloperów;
- środowisko testowe zostało skonfigurowane, udostępnione i zweryfikowane;
- zespół testerski został wyposażony w niezbędne narzędzia testowe i zasoby;
- harmonogram testów został zatwierdzony przez kierownika testów;

Proces testowy ulega zawieszeniu gdy:

- wykryto niezidentyfikowane dotąd nieścisłości między specyfikacją, a testowaną aplikacją;
- wykorzystanie budżetu;
- upływ zaplanowanego czasu przewidzianego na przeprowadzenie wszystkich testów;
- wykrycie błędów uniemożliwiających dalsze testowanie aplikacji;

Zawieszony proces testowy może zostać wznowiony gdy:

- zidentyfikowane nieścisłości zostaną wyjaśnione;
- błędy uniemożliwiające dalsze testowanie zostaną naprawione;
- zostanie przyznany dodatkowy budżet na realizację procesu testowego;
- zostanie przedłużony czas na przeprowadzenie procesu testowego.

Produkty testowania

Po przeprowadzeniu testów dostępne będą następujące produkty:

- plan testów;
- procedury testowe;
- lista przypadków testowych;
- harmonogram przeprowadzenia testów;
- raport nieścisłości;
- raport z przeprowadzonych testów;
- raporty z przeglądów formalnych;

- udokumentowana lista defektów;
- lista naprawionych defektów;
- dziennik testów.

Zadania testowania

Zadania, które będą towarzyszyć procesowi testowania aplikacji MrBuggy 3:

- przygotowanie przypadków oraz danych testowych;
- przygotowanie środowisk testowych;
- przeprowadzenie testów;
- raportowanie defektów.

Środowisko testowe

Komputer z systemem operacyjnym Windows 7 lub Windows 8/8.1, połączenie sieciowe, zainstalowana aplikacja MrBuggy3.

Odpowiedzialności

Zespół testowy odpowiedzialny za przetestowanie aplikacji MRBuggy3 będzie składał się z kierownika testów oraz trzech testerów, w tym dwóch senior testerów i jednego junior testera. Ogólna odpowiedzialność za prawidłowe przeprowadzenia procesu testowego spoczywa na kierowniku testów. Do jego zadań należą:

- planowanie i koordynacja procesu testowego;
- sporządzenie i aktualizacja planu testów;
- monitorowanie przebiegu procesu testowego;
- opracowanie raportu z procesu testowego;
- podejmowanie decyzji dotyczących wszystkich faz procesu testowego.

Do zadań testerów należą:

- udział w przygotowaniu planu testów;
- analizowanie, dokonywanie przeglądu i ocenianie wymagań, historyjek użytkownika i kryteriów akceptacji, specyfikacji oraz modeli pod kątem testowalności;
- przygotowanie i konfiguracja środowisk testowych zgodnych ze specyfikacją aplikacji;
- projektowanie i implementowanie przypadków testowych;
- przygotowanie i pozyskiwanie danych testowych;
- wykonywanie testów, ocenianie rezultatów oraz dokumentowanie przebiegu testów;

- korzystanie z odpowiednich narzędzi, niezbędnych do przeprowadzenia procesu testowego;

Potrzeby szkoleniowe i zapotrzebowanie na zasoby

W celu sprawnego przeprowadzenia procesu testowego niezbędne będzie zorganizowanie szkolenia dla junior testera z zakresu testowania aplikacji desktopowych.

Harmonogram testów

Przeprowadzenie procesu testowego będzie przebiegało w następujących fazach, zgodnie z uzgodnionym w zespole harmonogramem:

- zapoznanie się z dokumentacją;
- wyjaśnienie nieścisłości w specyfikacji;
- przygotowanie środowisk testowych;
- I etap testowania;
- przegląd po skończonym I etapie testów;
- II etap testów;
- zgłaszanie błędów;
- sporządzenie dokumentacji końcowej oraz raportów.

Ryzyka i plany awaryjne

Ryzyka związane z przeprowadzeniem procesu testowego:

- niedostępność pracowników w związku z planowanymi i nieplanowanymi urlopami;
- niepoprawnie działająca część serwerowa aplikacji;
- problemy z przygotowaniem środowiska testowego;
- duża ilość błędów krytycznych uniemożliwiająca dalsze przeprowadzenie testów;
- dostępność narzędzi;
- brakująca lub niepełna dokumentacja;

Plany awaryjne:

- zaangażowanie dodatkowych testerów;
- dodanie do harmonogramu kolejnych etapów testowania;
- ograniczenie możliwości urlopowych;
- przeprowadzenie dodatkowych szkoleń.

Zatwierdzenie planu

Plan testów powinien zostać zatwierdzony i monitorowany przez kierownika zespołu testerskiego.