

# **PLAN TESTÓW APLIKACJI MRBUGGY 3**

*Jan Nowak*

## Spis treści

<b>INFORMACJA O WERSJI.....</b>	<b>3</b>
<b>1 WPROWADZENIE .....</b>	<b>3</b>
1.1 CELE.....	3
1.2 CZŁONKOWIE ZESPOŁU .....	4
<b>2 ZAKRES TESTÓW .....</b>	<b>4</b>
<b>3 PODEJŚCIE TESTOWE.....</b>	<b>4</b>
3.1 SPECYFIKACJA.....	4
3.2 WERSJA DEMONSTRACYJNA .....	4
3.3 WERSJA KOŃCOWA .....	4
<b>4 KRYTERIA ZALICZENIA ORAZ NIEZALICZENIA TESTU .....</b>	<b>5</b>
<b>5 KRYTERIA ZAWIESZENIA I WZNOWIENIA TESTOWANIA .....</b>	<b>5</b>
<b>6 PRODUKTY TESTOWANIA .....</b>	<b>5</b>
<b>7 ZADANIA TESTOWANIA.....</b>	<b>5</b>
<b>8 ŚRODOWISKO TESTOWE .....</b>	<b>6</b>
<b>9 ODPOWIEDZIALNOŚCI .....</b>	<b>6</b>
<b>10 POTRZEBY SZKOLENIOWE, ZAPOTRZEBOWANIE NA ZASOBY .....</b>	<b>7</b>
10.1 SENIOR DEVELOPER.....	7
10.2 JUNIOR DEVELOPER.....	7
10.3 SENIOR TESTER.....	7
<b>11 HARMONOGRAM .....</b>	<b>7</b>
<b>12 RYZYKA I PLANY AWARYJNE.....</b>	<b>8</b>
<b>13 ZATWIERDZENIE PLANU.....</b>	<b>8</b>

## Informacja o wersji

Data	Wersja	Autor	Komentarz
01.05.2021	0.1	Jan Nowak	Szablon dokumentu
02.05.2021	0.2	Jan Nowak, Adam Kowalski	Opis zakresu testów oraz określenie podejścia testowego
05.05.2021	0.3	Adam Kowalski	Poprawki po pierwszym przeglądzie. Określenie kryteriów zaliczenia oraz niezaliczenia testu
07.05.2021	0.4	Jan Nowak, Adam Kowalski, Marek Nowak	Określenie kryteriów zawieszenia i wznowienia testowania, opis produktów oraz zadań testowania, wybór środowisk testowych
15.05.2021	0.5	Jan Nowak, Marek Nowak	Poprawki po drugim przeglądzie. Rozpisanie odpowiedzialności oraz potrzeb szkoleniowych
16.05.2021	0.6	Jan Nowak	Ustalenie harmonogramu
17.05.2021	0.7	Jan Nowak, Adam Kowalski, Marek Nowak	Poprawki po trzecim przeglądzie. Określenie ryzyka, planów awaryjnych
20.05.2021	0.8	Jan Nowak	Poprawki po czwartym przeglądzie
23.05.2021	1.0	Jan Nowak, Adam Kowalski, Marek Nowak	Drobne poprawki stylistyczne. Wersja końcowa dokumentu

## 1. Wprowadzenie

Plan Testów został stworzony w celu przekazania członkom zespołu informacji dotyczących aplikacji MrBuggy 3. Obejmuje on zakres testów, podejście, środowisko, ryzyka oraz harmonogram. Dokument ten wyraźnie określa kryteria, które są niezbędne w celu zaliczenia testu. Proces odbywa się w oparciu o specyfikację produktu.

### 1.1 Cele

MrBuggy3 ma za zadanie umożliwić przeprowadzenie i automatyczne sprawdzenie wyników pierwszej rundy Mistrzostw Polski w Testowaniu oprogramowania oraz wyłonienie uczestników rundy finałowej. Zespół ma za zadanie sprawdzić działanie poszczególnych funkcjonalności aplikacji o określonej rozdzielczości w różnych środowiskach.

W pierwszym etapie zadanie zespołu będzie polegało na dostarczeniu odpowiednich informacji dotyczących uruchamiania aplikacji na wymaganych systemach operacyjnych i czy zostaje poprawnie wyświetlona w docelowej rozdzielczości.

## 1.2 Członkowie zespołu

Imię i nazwisko	Rola
Tomasz Jasiński	Product Owner
Adam Kowalski	Senior Developer
Marek Nowak	Junior Developer
Kamil Nowak	Junior Developer
Jan Nowak	Senior Tester

## 2. Zakres testów

Elementy podlegające testom:

- Specyfikacja produktu
- Aplikacja kliencka – wersja demonstracyjna
- Aplikacja kliencka – wersja końcowa
- Proces buildowania aplikacji

Elementy nie podlegające testom:

- Wersja serwerowa aplikacji

Podstawą, na której zespół opiera swoje testy specyfikacji stanowi dokument dostarczony przez Product Ownera – Specyfikacja\_Mr\_Buggy\_3.pdf

([http://mrbuggy.pl/mrbuggy3/dfiles/Specyfikacja\\_Mr\\_Buggy\\_3.pdf](http://mrbuggy.pl/mrbuggy3/dfiles/Specyfikacja_Mr_Buggy_3.pdf))

## 3. Podejście testowe

### 3.1 Specyfikacja

- Przegląd – wyłapanie nieścisłości, zdobycie brakujących informacji.

### 3.2 Wersja demonstracyjna

- Testy przenaszalności – instalacja przenośnej aplikacji na różnych środowiskach,
- Testowanie użyteczności,
- Testy funkcjonalne – funkcjonalność przycisków, generowanie dziennika zdarzeń,
- Testy integracyjne – poprawna funkcjonalność na różnych systemach operacyjnych,
- Testy eksploracyjne – używanie aplikacji przez 2 godziny,
- Testy akceptacyjne.

### 3.3 Wersja końcowa

- Przegląd,
- Testy przenaszalności – instalacja przenośnej aplikacji na różnych środowiskach,
- Testy obciążenia – uruchomienie aplikacji na 5 różnych komputerach jednocześnie,
- Testy przeciążeniowe – uruchomienie aplikacji na 20 różnych komputerach jednocześnie,
- Testy funkcjonalne – funkcjonalność przycisków, generowanie dziennika zdarzeń,
- Testy integracyjne – poprawna funkcjonalność na różnych systemach operacyjnych,
- Testy eksploracyjne – używanie aplikacji przez 3 godziny,

- Testy niezawodności – monitorowanie aplikacji po wdrożeniu na środowisko zbliżone do produkcyjnego,
- Testy związane ze zmianami – testowanie każdej wprowadzonej zmiany,
- Testy pielęgnowalności,
- Testy akceptacyjne.

## 4 . Kryteria zaliczenia oraz niezaliczenia testu

Zostanie przygotowanych 100 testów akceptacyjnych. Aby uznać aplikację za gotową do wysłania na Mistrzostwa Polski, powinna przejść pomyślnie 90% z nich, czyli 90 testów potwierdzających poprawne działanie kluczowych funkcji oraz zgodność ze specyfikacją.

Aplikacja zawierająca chociaż 1 błąd krytyczny bądź też wynik testów akceptacyjnych będzie poniżej 90% nie przejdzie pomyślnie testu.

## 5. Kryteria zawieszenia i wznowienia testowania

Proces testowania może zostać przerwany z następujących powodów:

- Niejasna specyfikacja
- Testowana funkcjonalność blokuje następne funkcje
- Braki w zespole
- Niewystarczające zasoby
- Niewystarczająca ilość czasu, którą można poświęcić na dane testy

Wznowienie testowania nastąpi wtedy, gdy:

- Niejasności w specyfikacji zostaną wyjaśnione
- Testowana funkcjonalność przejdzie pomyślnie testy
- Zespół będzie w komplecie bądź też zostaną przydzielone nowe osoby do projektu
- Zostanie przeznaczona większa ilość zasobów
- Termin dostarczenia gotowej aplikacji zostanie przesunięty na późniejszą datę

## 6. Produkty testowania

- Plan testów
- Raporty z przeglądu specyfikacji
- Raporty z testów wersji demonstracyjnej
- Raporty z testów wersji końcowej

## 7. Zadania testowania

- Przygotowanie środowisk testowych – organizacja oprogramowania oraz sprzętu
- Przygotowanie przypadków testowych – pozytywnych oraz negatywnych
- Automatyzacja testów funkcjonalnych
- Wykonanie testów manualnych
- Wykonanie testów automatycznych
- Akceptacja poprawności aplikacji MrBuggy3

## 8. Środowisko testowe

Do przeprowadzania wszystkich testów zgodnie z harmonogramem oraz dostępną ilością osób w projekcie wymagane są następujące zasoby sprzętowe:

- 20 x Komputer (specyfikacja poniżej)
- 20 x Monitor obsługujący rozdzielczość 1024x768
- 2 x Pendrive 2GB

Specyfikacja komputera (minimalne wymagania):

- Procesor AMD Athlon 3000+
- Karta graficzna Nvidia GT210
- 2048 MB RAM
- HDD 120GB
- Port USB 2.0
- Karta sieciowa
- System operacyjny Windows 7/Windows 8/Windows 8.1

## 9. Odpowiedzialności

Rola	Liczba	Odpowiedzialność
Product Owner	1	- Dostarczenie oprogramowania oraz sprzętu - Decyzja o rozpoczęciu oraz zakończeniu testowania - Zdobywanie zasobów - Składanie raportów klientowi
Senior Developer	1	- Wybranie technologii - Automatyzacja testów - Implementacja zmian - Naprawa znalezionych błędów
Junior Developer	2	- Udzielanie pomocy Senior Developerowi - Naprawa znalezionych błędów - Udzielanie pomocy Senior Testerowi - Wykonywanie testów
Senior Tester	1	- Implementacja testów - Wykonanie testów - Składanie raportów Product Ownerowi z przeprowadzonych testów - Sugerowanie możliwych sposobów na usunięcie defektów

## 10. Potrzeby szkoleniowe, zapotrzebowanie na zasoby

### 10.1 Senior Developer

- Szkolenie uzupełniające z wybranej technologii
- Szkolenie zaawansowane z automatyzacji testów

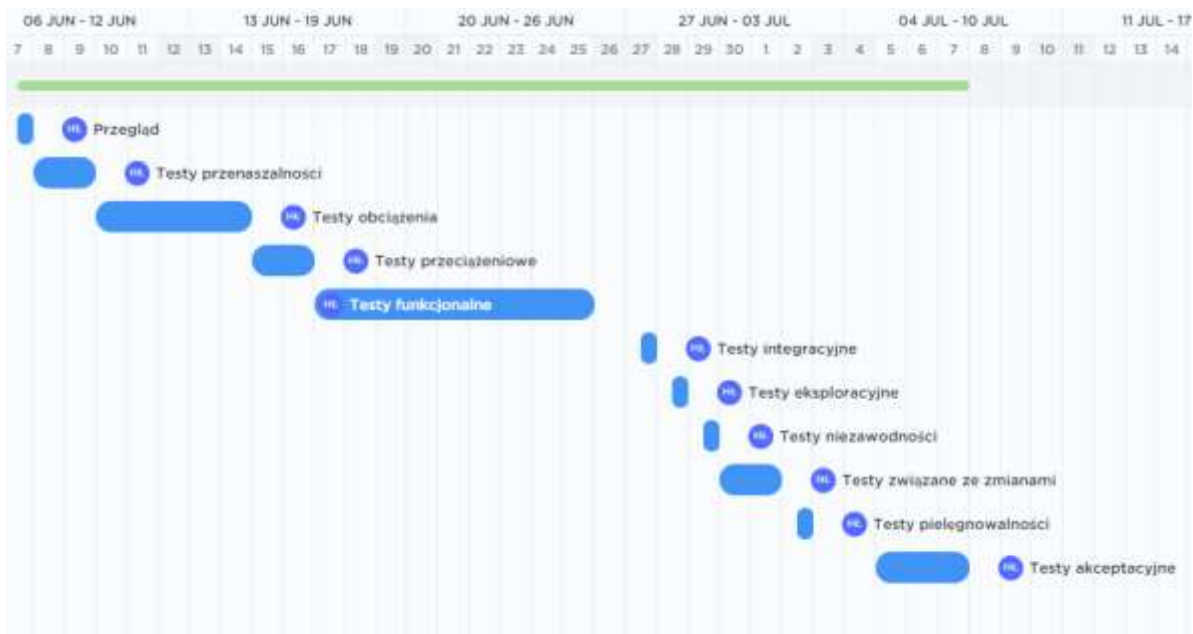
### 10.2 Junior Developer

- Szkolenie podstawowe z automatyzacji testów
- Szkolenie podstawowe z manualnego testowania
- Szkolenie średnio-zaawansowane z programowania

### 10.3 Senior Tester

- Szkolenie zaawansowane z testowania przeciążeniowego
- Szkolenie uzupełniające z testowania obciążeniowego
- Szkolenie podstawowe z programowania

## 11. Harmonogram



## 12. Ryzyka i plany awaryjne

Ryzyko	Wpływ	Skutek	Sugestia
Niewystarczająca ilość członków w zespole	Wysoki	Opóźnienie wydania produktu	Przypisanie większej ilości ludzi do projektu
Zbyt mała ilość testerów	Wysoki	Przeoczenie części defektów	Przypisanie większej ilości testerów do projektu
Zbyt mała ilość developerów	Średni	Opóźnienie automatyzacji testów oraz implementacji zmian	Przypisanie większej ilości developerów bądź przeszkolenie przypisanych już juniorów w szerszym zakresie
Niesprecyzowana specyfikacja od klienta	Średni	Opóźnienie pracy z powodu zbyt małej ilości informacji	Spotkanie się z klientem i wyjaśnienie zaznaczonych przez testera kwestii

## 13. Zatwierdzenie planu

Lista osób, które są niezbędne do zatwierdzenia planu testów

- Tomasz Jasiński – Product Owner
- Jan Nowak – Senior Tester