|  |  |
| --- | --- |
| **PLAN TESTÓW** | |
| PROJEKT: | **PLATFORMA „CODERS GURU”** |
| WERSJA: | **WERSJA 3.0** |

**SPIS TREŚCI**

1. Wprowadzenie

1.1. Cel i zakres dokumentu

1.2. Oczekiwania

2. Przedmiot testów

2.1. Ogólna charakterystyka produktu

2.2. Użytkownicy i cele produktu

2.3. Elementy podlegające testowaniu

3. Zakres planowanych testów

3.1. Wymagania objęte zakresem testów

3.2. Wyłączenie z zakresu testów

4. Podejście do testów

4.1. Poziom i typ testów

4.2. Wykorzystane techniki projektowania testów oraz narzędzia

4.3. Fazy testów

5. Kryteria

5.1. Kryteria zaliczenia / niezaliczenia testu

5.1. Warunki rozpoczęcia testów

5.2. Warunki zakończenia testów

5.3. Kryteria zawieszenie i wznowienia testów

6. Produkty procesu testowego

7. Czynności i zadania testowe

8. Środowisko testowe

9. Role i odpowiedzialności, potrzeby szkoleniowe

9.1. Podział obowiązków procesu testowego

9.2. Potrzeby szkoleniowe

10. Harmonogram

11. Rejestr ryzyk

11.1. Ryzyka projektowe

11.2. Ryzyka produktowe

12. Zatwierdzenie planu

**Metryka dokumentu**

|  |  |  |  |
| --- | --- | --- | --- |
| **METRYKA DOKUMENTU** | | | |
| Nazwa dokumentu: | Plan testów do projektu platformy „Coders Guru” | | |
| Nr ID dokumentu: | CG-PlanTestów | | |
| Streszczenie: | Opisuje zakres, metody, zasoby i harmonogram zamierzonych czynności testowych | | |
| Projekt: | Platforma „Coders Guru” | | |
| Właściciel dokumentu: | Wykonawca projektu | | |
| Sporządził: | Magdalena Kmiecik | | |
| Nr wersji: | 3.0 | Data sporządzenia: | 19.10.2022 r. |
| Status: | W toku | Data ostatniej modyfikacji: | 25.10.2022 r. |
| Zatwierdził: | Jan Kowalski | Data zatwierdzenia: | 25.10.2022r. |

**Historia zmian dokumentu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **HISTORIA ZMIAN DOKUMENTU** | | | | | |
| **Nr wersji** | **Data** | **Opis zmian** | **Działanie\*** | **Rozdział\*\*** | **Autor** |
| 1.0. | 20.10.2022 r. | Dodanie elementu (Podstrona Regulamin) do przeprowadzenia testów funkcjonalnych | N | W | Magdalena Kmiecik |
| 2.0. | 22.10.2022 r. | Dodanie nowego typu testów – testy eksploracyjne | N | W | Magdalena Kmiecik |
| 3.0. | 25.10.2022 r. | Dodanie nowego typu testów – testy akceptacyjne dotyczące UI/UX | N | W | Magdalena Kmiecik |

\* N - nowy, M – modyfikacja, W – weryfikacja

\*\* Numer rozdziału lub W – wszystkie rozdziały

**Lista załączników**

|  |  |  |
| --- | --- | --- |
| **LISTA ZAŁĄCZNIKÓW** | | |
| **Lp.** | **Załącznik** | **Uwagi** |
| 1 | Projekt techniczny CG\_ProjektTechniczny\_1.0. | - |

**Wykaz użytych skrótów, symboli i terminów**

|  |
| --- |
| **WYKAZ UŻYTYCH SKRÓTÓW, SYMBOLI I TERMINÓW** |

|  |  |
| --- | --- |
| **Skrót/termin** | **Znaczenie** |
| CG | Coders Guru |
| W-FUN | Wymagania funkcjonalne |
| W-INT | Wymagania dotyczące integracji z systemami zewnętrznymi |
| W-UX | Wymagania dotyczące użyteczności serwisu |
| W-BD | Wymagania dotyczące baz danych |
| W-WYD | Wymagania wydajnościowe |
| W-BEZ | Wymagania bezpieczeństwa |
| UI/UX | Interfejs użytkownika / Doświadczenie użytkownika od ang.  *User interface / User experience* |
| Priorytety błędów: |  |
| Highest | Blokuje wykonywanie dalszych testów systemu |
| High | Znaczący błąd, nie można przetestować kluczowej funkcjonalności systemu |
| Medium | Średni problem, nie dotyczy kluczowych funkcji systemu. Możliwe jest kontynuowanie testów |
| Low | Niewielka usterka, minimalny wpływ na system |
| Lowest | Błąd typu literówka w treści strony |

# 1. Wprowadzenie

* 1. Cel i zakres dokumentu

Cel dokumentu: określenie podejścia do procesu testowego systemu Coders Guru.

Zakres dokumentu: określenie przedmiotu testów, zakresu planowanych testów, podejścia do testów, sformułowanie kryteriów wejścia i wyjścia, zdefiniowanie produktów procesu testowego oraz czynności i zadań testowych a także środowiska testowego, opracowanie harmonogramu oraz podejście do zarządzania ryzykiem.

1. **Przedmiot testów**

2.1. Ogólna charakterystyka produktu

Platforma łącząca doświadczonych programistów z osobami, które zgłosiły się po poradę w zakresie rozwiązania konkretnego problemu.

2.2. Użytkownicy i cele produktu

Użytkownicy:

* Potencjalni kursanci: osoby, które myślą o nauce programowania, ale nie wiedzą czym ono jest oraz uczący się samodzielnie – osoby potrzebujące w procesie edukacji wsparcia mentora. Pakiet dla firm- opłacone przez firmy jako benefit dla pracowników;
* Wykładowcy - doświadczeni programiści.

Cele: Funkcja edukacyjna oraz marketingowa platformy.

2.3. Elementy podlegające testowaniu

Testowane elementy dla testowania dynamicznego w przeglądarkach Chrome i Opera:

* Strona główna
* Podstrony z formularzem rejestracyjnym
* Podstrona Jak to działa
* Podstrona Cennik
* Podstrona Regulamin
* Podstrona Zostań mentorem
* Intergacja z systemem płatności on-line
* Komunikacja z bazą danych
* Charakterystyki dotyczące użyteczności

Testowane elementy dla testowania statycznego:

* Projekt techniczny

Nietestowane elementy:

* Charakterystyki dotyczące wydajności oraz bezpieczeństwa – klient zrealizuje te testy po swojej stronie.
* Inne – wcześniej nieokreślone.

1. **Zakres planowanych testów**

3.1. Wymagania objęte zakresem testów

Wymagania objęte zakresem testów: W-FUN, W-INT, W-UX, W-BD.

Dokładny spis wymagań:

* W-FUN.1 – Możliwość zarejestrowania prywatnego użytkownika;
* W-FUN.2 – Możliwość zarejestrowania firmowego użytkownika;
* W-FUN.3 – Możliwość zalogowania zarejesrtowanego użytkownika;
* W-FUN.4- Możliwość oznaczenia dni i godzin dostępności poszczególnych mentorów ( z poziomu baz danych);
* W-FUN.5- Możliwość wybrania mentora z listy dostępnych mentorów;
* W-FUN.6- Możliwość filtrowania/wyszukiwania mentorów według kursów;
* W-FUN.7-Profil wykładowcy zawiera zdjęcie, doświadczenie, technologie oraz najbliższe wolne terminy;
* W-FUN.8- Podstrona cennik- zawierająca informację o cenie konsultacji;
* W-FUN-9- Podstrona Regulamin – zawierająca regulamin usługi;
* W-FUN-10 – System, po zalogowaniu a przed przejściem do profilu, wyświetla komunikat dotyczący konieczności potwierdzenia aktualnych danych kontaktowych;
* W-FUN-11 – System umożliwia zarządzanie profilem klienta „Moje konto”;
* W-FUN-12 – System umożliwia eksport zaplanowanych konsultacji do formatu iCalendar;
* W-INT.1. - Integracja z systemem płatności on-line;
* W-UX.1 – System powinien spełniać wymagane standardy W3C w kontekście struktury dokumentu HTML5;
* W-UX.2 – System powinien spełniać wymagane standardy W3C w kontekście struktury layoutu CSS 3.0 lub nowszej;
* W-UX.3 – System umożliwia dostęp do wszystkich funkcjonalności portalu przez przeglądarkę internetową;
* W-UX-4 – System jest przystosowany do wyświetlania treści na różnorodnych urządzaniach mobilnych.

3.2. Wyłączenie z zakresu testów

Wymagania wyłączone z zakresu testów: W-WYD , W-BEZ.

Nie będziemy realizować testów wydajnościowych bezpieczeństwa z uwagi na brak wystarczających kompetencji w zespole. Klient zrealizuje te testy po swojej stronie z wykorzystaniem swoich zasobów kompetencyjnych / sprzętowych.

1. **Podejście do testów**

4.1. Poziom i typ testów

Typy testów:

* Statyczne;
* Dynamiczne;
* Funkcjonalne;
* Niefunkcjonalne;
* Związane ze zmianami (retesty, testy regresji).

Poziomy testów:

* Modułowe;
* Integracyjne;
* Systemowe;
* Akceptacyjne.

4.2. Wykorzystane techniki projektowania testów oraz narzędzia

Wykorzystane techniki projektowania testów:

* Techniki białoskrzynkowe ;
* Techniki czarnoskrzynkowe;
* Techniki oparte na doświadczeniu.

Wykorzystane narzędzia:

* Narzędzia do zarządzania testami: Testlink;
* Narzędzia do zarządzania defektami: Jira;
* Narzędzia do wykonywania testów: SoapUI;
* Narzędzia mierzące pokrycie (np. pokrycie wymagań lub kodu): CodeCover.

4.3. Fazy testów

Praca w modelu iteracyjno-przyrostowym (SCRUM).

Fazy wytwarzania oprogramowania:

* Rozpoczęcie: testy statyczne, zapoznanie się z dokumentacją;
* Opracowanie: opracowanie zestawu testów (przypadków testowych dla funkcjonalności, które później sprawdzimy na działającym oprogramowaniu);
* Budowa: testy modułowe, integracyjne dot. modułów, testy systemowe na działającym oprogramowaniu, testy integracyjne systemów;
* Przekazanie: testy akceptacyjne.

1. **Kryteria**

5.1. Kryteria zaliczenia / niezaliczenia testu

* 100 % pokrycia przypadkami testowymi dla scenariuszy pozytywnych i negatywnych;
* 100 % wykonania testów;
* Brak otwartych błędów o priorytecie HIGH i HIGHEST.

5.2. Warunki rozpoczęcia testów

* Działające środowisko testowe;
* Dostępne niezbędne narzędzia;
* Dostępny zespół ludzi o odpowiednich kompetencjach technicznych i biznesowych;
* Dostępność danych testowych.

5.3. Warunki zakończenia testów

Aby uznać system za gotowy do wdrożenia powinien przejść pomyślnie przez 80 % testów i testy dla wymagań o najwyższym priorytecie tj.1 muszą zostać zaliczone.

Uzyskanie wysokich wskaźników testów niefunkcjonalnych.

5.4. Kryteria zawieszenie i wznowienia testów

W przypadku wykrycia awarii należy zawiesić proces testowy. Po naprawieniu przyczyn awarii należy wznowić proces testowy.

1. **Produkty procesu testowego**

* Produkty pracy planowania testów : plan testów;
* Produkty pracy monitorowania testów i nadzoru nad testami : raporty z testów, w tym raporty o postępie testów oraz sumaryczne raporty ;
* Produkty pracy analizy testów: zdefiniowane i uszeregowane według priorytetów warunki testowe, możliwość dwukierunkowego śledzenia powiązań między tymi warunkami a pokrywanymi przez nie elementami podstawy testów, karta opisu testów dla testów eksploracyjnych;
* Produkty pracy projektowania testów: przypadki testowe, z których są utworzone zestawy testowe zdefiniowane na etapie analizy testów;
* Produkty pracy implementacji testów :procedury testowe oraz kolejność ich wykonywania, zestawy testowe, harmonogram wykonania testów;
* Produkty pracy wykonywania testów : dokumentacja dotycząca statusu poszczególnych przypadków testowych lub procedur testowych, raporty o defektach;
* Produkty pracy ukończenia testów : sumaryczny raport z ukończenia testów.

1. **Czynności i zadania testowe**

Wykonywane w iteracjach w metodyce SCRUM.

* Planowanie testów;
* Monitorowanie testów i nadzór nad testami;
* Analiza testów;
* Projektowanie testów;
* Implementacja testów;
* Wykonywanie testów;
* Ukończenie testów.

Uszczegółowienie poszczególnych czynności w pkt. 10 (Harmonogram).

1. **Środowisko testowe**

System operacyjny Windows NT 10 64-bitowy, przeglądarki Chrome oraz Opera.

1. **Role i odpowiedzialności, potrzeby szkoleniowe**

9.1. Podział obowiązków procesu testowego

Role i odpowiedzialności:

* Kierownik testów – 1 os.;
* Planowanie i logistyka, składanie raportów Zamawiającemu;
* Analityk – 1 os. Określanie idei poszczególnych testów, ocena jakości produktu;
* Projektant – 2 os. Określenie konkretnej struktury i budowy testów, wybranie technologii, projekt danych do testów automatycznych;
* Tester – 6 os. Implementacja testów, wykonanie testów, składanie raportów Kierownikowi testów z przeprowadzonych testów;
* Programista – 3 os. Wykonywanie testów jednostkowych, debugowanie.

9.2. Potrzeby szkoleniowe

Potrzeby szkoleniowe: w zakresie wcześniej niestosowanego w zespole narzędzia SoapUI.

1. **Harmonogram**

Planowanie testów

* Utworzenie zespołu testowego 1 dn.
* Utworzenie planu testów 2 dn.

Monitorowanie testów i nadzór nad testami: przez cały czas trwania projektu.

Analiza testów:

* Zapoznanie się z dokumentacją / Testy statyczne 7 dn.
* Utworzenie warunków testowych oraz utworzenie karty opisu testów 4 dn.

Projektowanie testów:

* Utworzenie zestawów testowych 4 dn.

Implementacja testów:

* Utworzenie środowiska testowego 1 dn.
* Utworzenie procedur testowych 2 dn.

Wykonywanie testów w metodologii SCRUM:

* Testy I etap (wraz z modyfikacjami i retestami) 14 dn.
* Testy II etap (wraz z modyfikacjami i retestami) 14 dn.
* Testy III etap (wraz z modyfikacjami i retestami) 14 dn.
* Testy regresji całości 6 dn.
* Testy akceptacyjne 10 dn.
* Modyfikacje 3 dn.
* Retesty 3 dn.
* Testy regresji 6 dn.

Ukończenie testów.

* Utworzenie sumarycznego raportu z ukończenie testów 4 dn.

1. **Rejestr ryzyk**

11.1. Ryzyka projektowe

* Absencja członków zespołu;
* Opóźnienia w developmencie;
* Czynniki dostawcy.

11.2. Ryzyka produktowe

* Oprogramowanie dostarczone na rynek zawiera defekty powodujące awarie;
* Słabe parametry oprogramowania (np. związane z tym że testy wydajnościowe oraz bezpieczeństwa wykonywane są „na zewnątrz”);
* Oprogramowanie nie spełniające swoich założeń.

1. **Zatwierdzenie planu**

Poniżej zamieszczona jest tabela interesariuszy, którzy muszą zatwierdzić plan, aby mógł on wejść w życie.

|  |  |  |
| --- | --- | --- |
| **Imię i nazwisko** | **Funkcja** | **Organizacja** |
| Jan Kowalski | Kierownik testów | Wykonawca projektu |
| Adam Włodek | Kierownik działu IT | Wykonawca projektu |