

PLAN TESTÓW	
PROJEKT:	PLATFORMA „CODERS GURU”
WERSJA:	WERSJA 1.0

Metryka dokumentu	3
Historia zmian dokumentu	3
Lista załączników	4
Wykaz użytych skrótów, symboli i terminów	4
Wprowadzenie	5
Cel i zakres dokumentu	5
Oczekiwania	5
Załączniki:	
Projekt techniczny/Wymagania	5
Przedmiot testów	6
2.1. Ogólna charakterystyka produktu	6
2.2. Użytkownicy i cele produktu	6
2.3. Elementy podlegające testowaniu	6
Zakres planowanych testów	7
Podejście do testów	7
4.1. Poziom i typ testów	7
4.2. Wykorzystane techniki projektowania testów oraz narzędzia	7
4.3. Fazy testów	7
Kryteria	8
5.1. Kryteria zaliczenia / niezaliczenia testu	8
5.2. Warunki rozpoczęcia testów	8
5.3. Warunki zakończenia testów	8
5.4. Kryteria zawieszenie i wznowienia testów	8
Produkty procesu testowego	8
Czynności i zadania testowe	8
Środowisko testowe	9
Role i odpowiedzialności, potrzeby szkoleniowe	9
9.1. Podział obowiązków procesu testowego	9
9.2. Potrzeby szkoleniowe	9
Harmonogram	9
Rejestr ryzyk	10

Metryka dokumentu

METRYKA DOKUMENTU			
Nazwa dokumentu:	CG-ProjektTechniczny_1.0		
Nr ID dokumentu:	CG-ProjektTechniczny		
Streszczenie:	Platforma łącząca doświadczonych programistów z osobami, które potrzebują wsparcia w rozwiązaniu konkretnego problemu. Umożliwia zdalne połączenie dwóch osób za pomocą czatu wideo i tekstowego, wymiany plików i edycji kodu na żywo.		
Projekt:	Budowa systemu CodersGuru		
Właściciel dokumentu:	ONL_TEM_S_06		
Sporządził:	ONL_TEM_S_06		
Nr wersji:	1.0	Data sporządzenia:	23.03.21
Status:		Data ostatniej modyfikacji:	23.03.21
Zatwierdził:	Wojciech Jeziorski	Data zatwierdzenia:	23.03.21

Historia zmian dokumentu

HISTORIA ZMIAN DOKUMENTU					
Nr wersji	Data	Opis zmian	Działanie *	Rozdział**	Autor
1.0	23.03.2021	Stworzenie planu testów	N	W	ONL_TEM_S_06

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

** Numer rozdziału lub W – wszystkie rozdziały

Lista załączników

LISTA ZAŁĄCZNIKÓW		
Lp.	Załącznik	Uwagi
1	Projekt "CodersGuru"	

Wykaz użytych skrótów, symboli i terminów

WYKAZ UŻYTYCH SKRÓTÓW, SYMBOLI I TERMINÓW	
Skrót/termin	Znaczenie
W-FUN	Wymaganie funkcjonalne
W-INT	Wymaganie dotyczące integracji z systemami zewnętrznymi
W-UX	Wymaganie dotyczące użyteczności serwisu
W-BD	Wymaganie dotyczące baz danych
W-WYD	Wymagania wydajnościowe
W-BEZ	Wymagania bezpieczeństwa

1. Wprowadzenie

Cel i zakres dokumentu

- Celem tego dokumentu jest nakreślenie ogólnego zarysu testów, sprawdzających poprawność implementacji Systemu. Są tu określone podstawowe fazy, ich cel oraz elementy Systemu, które wymagają sprawdzenia.

Oczekiwania

- Budowa bazy potencjalnych wykładowców

Osoby chcące udzielać korepetycji to idealny materiał dla naszego HR do rekrutacji wykładowców.

- Zwiększenie przejścia potencjalnych kursantów

Osobom, które dopiero zastanawiają się nad swoim początkiem w IT serwis umożliwi porozmawianie z doświadczonym wykładowcą, które nie rodzi dużych kosztów i zobowiązań - niwelujemy barierę dla osób, które przychodzą na kurs nie wiedząc czego się spodziewać.

Załączniki:

[Projekt techniczny/Wymagania](#)

2. Przedmiot testów

2.1. Ogólna charakterystyka produktu

Platforma łącząca doświadczonych programistów z osobami, które potrzebują wsparcia w rozwiązaniu konkretnego problemu. Umożliwia zdalne połączenie dwóch osób za pomocą czatu wideo i tekstowego, wymiany plików i edycji kodu na żywo.

Rozliczenia za czas połączenia z wykładowcą.

2.2. Użytkownicy i cele produktu

- Potencjalni kursanci - osoby, które myślą o nauce programowania ale nie wiedzą czym ono jest.
- Uczący się samodzielnie - osoby potrzebujące w procesie edukacji wsparcia mentora.
- Pakiet dla firm - zbliżone do Grupy 3 – opłacone przez firmy jako benefit dla pracowników.

2.3. Elementy podlegające testowaniu

Testowaniu będą podlegały następujące elementy:

- Specyfikacja wymagań,
- Wymagania funkcyjne,
- Baza danych,
- Integracja z systemem płatności on-line,
- Użyteczność serwisu,
- Wydajność aplikacji,
- Bezpieczeństwo aplikacji.

3. Zakres planowanych testów

Wymagania znajdują się w załączniku - [Projekt "CodersGuru"](#)

4. Podejście do testów

4.1. Poziom i typ testów

- Testy modułowe
- Testy integracyjne
- Testy systemowe
- Testy akceptacyjne

4.2. Wykorzystane techniki projektowania testów oraz narzędzia

Testy statyczne:

- przegląd koleżeński
- przegląd techniczny
- inspekcja

Techniki oparte na specyfikacji, czarnoskrzynkowe:

- Podział na klasy równoważności
- Analiza wartości brzegowych
- Testowanie w oparciu o tablice decyzyjną
- Testowanie przejść pomiędzy stanami
- Testowanie w oparciu o przypadki testowe

Techniki oparte na doświadczeniu

4.3. Fazy testów

- Sprint 0 (Instalacja narzędzi testowych, przygotowanie środowisk testowych, rezerwacja zasobów)
- Sprint 1 (testy in-house)
- Sprint 2 (Testy u klienta)
- Wydanie produktu

5. Kryteria

5.1. Kryteria zaliczenia / niezaliczenia testu

- Test będzie zaliczony na podstawie dokumentacji CG-ProjektTechniczny_1.0, po spełnieniu wymagań, sprawdzeniu czy główna funkcjonalność została przetestowana z wynikiem pozytywnym oraz na podstawie akceptacji przez klienta
- Test będzie niezaliczony

5.2. Warunki rozpoczęcia testów

- dostępność i gotowość środowiska testowego
- gotowość narzędzi testowych w środowisku testowym
- dostępność testowalnego kodu
- dostępność danych testowych

5.3. Warunki zakończenia testów

- Zebranie danych pochodzących z wykonanych czynności testowych w celu usystematyzowania i połączenia zdobytych doświadczeń, testaliów oraz innych istotnych informacji.

5.4. Kryteria zawieszenie i wznowienia testów

- Testy zostaną zawieszone w przypadku gdy liczba błędów przekracza określone limity do czasu aż nastąpi poprawienie aplikacji, a następnie dana faza testów rozpoczynana jest od nowa.

6. Produkty procesu testowego

plan testów,
przypadki testowe,
raporty błędów,
raporty z przeglądów,
baza danych i dane konfiguracyjne,
pliki (projekt techniczny),
linki i załączniki,
środowisko testowe

7. Czynności i zadania testowe

- planowanie testów;
- monitorowanie testów i nadzór nad testami;
- analiza testów;
- projektowanie testów;
- implementacja testów;

- wykonywanie testów;
- ukończenie testów.

8. Środowisko testowe

- Windows 10 (64 bit) (10.0 kompilacja 19041/)
- Android 10
- Przeglądarka Edge wersja 89.0.774.57 (64-bit) / Android 46.02.4.5152
- Przeglądarka Google Chrome wersja 1.21.77 Chromium: 89.0.4389.90 (64-bit) / (Android 89.0.4389.105)
- Przeglądarka Firefox wersja 86.0.1 (64-bit)/(Android 86.1.1)
- Przeglądarka Safari wersja 14.0/14.0.3 (iOS 14.4.1/macOS Big Sur 11.2.3)
- Responsywność sprawdzona za pomocą Google Resizer (1024×768, 1366×768, 1680×1050, 1920×1080)

9. Role i odpowiedzialności, potrzeby szkoleniowe

9.1. Podział obowiązków procesu testowego

Project manager - zarządzanie i monitorowanie etapów przebiegu projektu.

Team leader - zlecanie poszczególnych czynności do wykonania podległemu zespołowi. Kontrolowanie pracy analityków, testerów i przegląd wyników.

Analitik testów - ustalenie kryteriów wejścia, wyjścia, wymagań, ogólne ustalenie podstawy testów. Współpraca z testerami na wstępnych etapach procesu testowego.

Tester - wyszukiwanie defektów w kodzie, specyfikacji projektu, projektowanie i implementacja testów, raportowanie team leaderowi.

9.2. Potrzeby szkoleniowe

- dokładniejsze zrozumienie specyfikacji projektu przez testerów,
- dobranie właściwej strategii przeprowadzenia projektu,
- niwelacja barier dla osób nie branżowych,

10. Harmonogram

Wersja 1.0 powinna zostać oddana do dnia podanego przez klienta. Nie zawiera ona jeszcze personalizacji ustawień.

Wersja 1.1 powinna zostać oddana do dnia podanego przez klienta. Zawiera ona wszystkie funkcjonalności, aby mogła zostać przetestowana integralności Systemu.

Wersja 1.2 powinna zostać oddana do dnia podanego przez klienta. Wszystkie funkcjonalności zostały już wprowadzone i przetestowane z względu na ich integralności z resztą Systemu oraz wydajność ich pracy.

11. Rejestr ryzyk

11.1. Ryzyka projektowe

- wprowadzenie w ostatniej chwili zmian wymagających dokonania licznych przeróbek;
- niedoszacowany budżet projektu;
- braki kadrowe w zespole deweloperskim wynikające z sezonu grypowego i pandemii/sezonu urlopowego;
- źle oszacowana pracochłonność projektu skutkującą skróceniem czasu testowania;
- problemy komunikacyjne z klientem, różnice w interpretacji wymagań platformy;

11.2. Ryzyka produktowe

- problem z aktualizacją kalendarzy wykładowców;
- problem z dużą wymianą wykładowców;
- zbyt długi czas oczekiwania na połączenie z systemem płatności on-line;
- platforma nie jest intuicyjna w obsłudze dla użytkowników;
- platforma źle działa pod dużym obciążeniem;
- problem z jakością połączenia video;
- problem z oprogramowaniem typu open source;
- struktura serwisu jest trudna do czytania przez maszyny Google'a przez co potencjalni zainteresowani użytkownicy nie mogą trafić na platformę;

12. Zatwierdzenie planu

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

Imię i nazwisko	Funkcja	Organizacja	Podpis
Wojtek Jeziorski	Nauczyciel	CodersLab	