|  |  |
| --- | --- |
| **PLAN TESTÓW** | |
| PROJEKT: | **PLATFORMA „CODERS GURU”** |
| WERSJA: | **WERSJA 1.0** |

**SPIS TREŚCI**

[1.](#_heading=h.gjdgxs) Wprowadzenie 5

[1.1.](#_heading=h.30j0zll) Cel i zakres dokumentu 5

[1.2.](#_heading=h.1fob9te) Oczekiwania 5

[2.](#_heading=h.3znysh7) Przedmiot testów 5

[2.1. Ogólna charakterystyka produktu 5](#_heading=h.2et92p0)

[2.2. Użytkownicy i cele produktu 5](#_heading=h.tyjcwt)

[2.3. Elementy podlegające testowaniu 5](#_heading=h.3dy6vkm)

[3.](#_heading=h.1t3h5sf) Zakres planowanych testów 5

[3.1. Wymagania objęte zakresem testów 5](#_heading=h.4d34og8)

[3.2. Wyłączenie z zakresu testów 5](#_heading=h.2s8eyo1)

[4.](#_heading=h.17dp8vu) Podejście do testów 5

[4.1. Poziom i typ testów 5](#_heading=h.3rdcrjn)

[4.2. Wykorzystane techniki projektowania testów oraz narzędzia 5](#_heading=h.26in1rg)

[4.3. Fazy testów 5](#_heading=h.lnxbz9)

[5.](#_heading=h.35nkun2) Kryteria 5

[5.1. Kryteria zaliczenia / niezaliczenia testu 5](#_heading=h.1ksv4uv)

[5.2. Warunki rozpoczęcia testów 5](#_heading=h.44sinio)

[5.3. Warunki zakończenia testów 5](#_heading=h.2jxsxqh)

[5.4. Kryteria zawieszenie i wznowienia testów 5](#_heading=h.z337ya)

[6.](#_heading=h.3j2qqm3) Produkty procesu testowego 5

[7.](#_heading=h.1y810tw) Czynności i zadania testowe 5

8[.](#_heading=h.2xcytpi) Role i odpowiedzialności, potrzeby szkoleniowe 6

8[.1. Podział obowiązków procesu testowego 6](#_heading=h.1ci93xb)

8[.2. Potrzeby szkoleniowe 6](#_heading=h.3whwml4)

9[.](#_heading=h.2bn6wsx) Harmonogram 6

[10.](#_heading=h.qsh70q) Rejestr ryzyk 6

10[.1. Ryzyka projektowe 6](#_heading=h.3as4poj)

10[.2. Ryzyka produktowe 6](#_heading=h.1pxezwc)

[11.](#_heading=h.49x2ik5) Zatwierdzenie planu 6

**Metryka dokumentu**

|  |  |  |  |
| --- | --- | --- | --- |
| **METRYKA DOKUMENTU** | | | |
| Nazwa dokumentu: | Plan testu platformy “Coders Guru” | | |
| Nr ID dokumentu: | CG-Projekt techniczny | | |
| Streszczenie: | Niniejszy plan testów definiuje cele testowania i określa czynności testowe potrzebne do realizacji projektu stworzenia systemu “CodersGuru”. Dokument opisuje zakres, metody, zasoby, harmonogram i ryzyka zamierzonych czynności testowych. | | |
| Projekt: | Budowa systemu CodersGuru | | |
| Właściciel dokumentu: |  | | |
| Sporządził: | Bartosz Frąckowiak, Danuta Grzeszak, Aleksandra Juchacz, Paweł Orman | | |
| Nr wersji: |  | Data sporządzenia: | 23.02.2020 |
| Status: |  | Data ostatniej modyfikacji: | 23.02.2020 |
| Zatwierdził: |  | Data zatwierdzenia: |  |

**Historia zmian dokumentu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **HISTORIA ZMIAN DOKUMENTU** | | | | | |
| **Nr wersji** | **Data** | **Opis zmian** | **Działanie\*** | **Rozdział\*\*** | **Autor** |
| 1 | 23.02.2020 |  | N |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

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

\*\* Numer rozdziału lub W – wszystkie rozdziały

**Lista załączników**

|  |  |  |
| --- | --- | --- |
| **LISTA ZAŁĄCZNIKÓW** | | |
| **Lp.** | **Załącznik** | **Uwagi** |
| 1 | CG\_Projekt techniczny\_1\_0 |  |
| 2 | Harmonogram wykonywania testów |  |
|  |  |  |

**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**
   1. Cel i zakres dokumentu

Celem tego dokumentu jest określenie ogólnego zarysu testów sprawdzających poprawność wymagań. Zostały w nim określone wymagania, podejście do testów, kryteria, produkty procesu testowego, czynności i zadania testowe, środowisko, role, harmonogram oraz rejestr ryzyk.

* 1. Oczekiwania

Faza testowania ma dowieść, że wykonany System spełnia wymagania funkcjonalne i pozafunkcjonalne, przedłożone przez Zamawiającego. Wymagania te, określone w dokumencie “CG\_Projekt techniczny\_1\_0” (Załącznik 1) zostaną zweryfikowane przez zespół testujący. Wyniki testów zostaną przedstawione Zamawiającemu w postaci pisemnych raportów

**2. Przedmiot testów**

2.1. Ogólna charakterystyka produktu

Platforma łącząca doświadczonych programistów z osobami, które potrzebują wsparcia

w rozwiązaniu konkretnego problemu informatycznego. Umożliwia zdalne połączenie dwóch osób za pomocą czatu wideo i tekstowego, wymiany plików i edycji kodu na żywo.

2.2. Użytkownicy i cele produktu

Jako *programista* z doświadczeniem

chcę mieć możliwość udzielania korepetycji

aby pomagać osobom uczącym się programowania i zdobyć dodatkowe źródło zarobku.

Jako *pracownik HR*

chcę mieć dostęp do bazy korepetytorów

aby mieć możliwość rekrutowania wykładowców.

Jako *początkujący kursant*

chcę dowiedzieć się czym jest programowanie

aby mieć większą wiedzę na jej temat

Jako*uczący się samodzielnie*

chcę uzupełnić brakujące obszary wiedzy

aby uzyskać pełną wiedzę i umiejętności w zakresie założonego obszaru IT

Jako *firma*

chcę umożliwić swoim pracownikom podniesienie kompetencji z zakresu programowania

aby mieli możliwość rozwijania swoich umiejętności przez co będą mogli efektywniej wykonywać swoje zadania

Jako *administrator*

chcę mieć dostęp do wszystkich kont użytkowników

aby swobodnie zarządzać platformą.

2.3. Elementy podlegające testowaniu

* Profil wykładowcy
* Formularz rejestracyjny - mentor
* Formularz rejestracyjny prywatny - użytkownik
* Formularz rejestracyjny firmowy- użytkownik
* Oznaczenia dni i godzin mentora
* Lista dostępnych mentorów
* Profil klienta
* Podstrona Cennik
* Podstrona Regulamin
* System płatności on-line

3. **Zakres planowanych testów**

3.1. Wymagania objęte zakresem testów

Testowane będą wymagania:

1. funkcjonalne

* formularz rejestracyjny
* logowanie zarejestrowanego użytkownika
* oznaczenie dni i godzin dostępności
* wybranie mentora z listy
* filtrowanie/wyszukiwanie mentorów
* uzupełnianie i wyświetlanie profilu mentora
* podstrona Cennik
* podstrona Regulamin
* komunikat o konieczności aktualizacji danych kontaktowych
* zarządzanie profilem “Moje konto”
* eksport do aplikacji iCalendar

1. dotyczące integracji z systemami zewnętrznymi

* płatności on-line

1. dotyczące użyteczności serwisu

* standardy W3C w kontekście HTML5 i CSS 3.0
* interpretacja GUI na różnych przeglądarkach i ich odpowiednikach mobilnych
* responsywność

1. dotyczące baz danych

* działanie z jedną z baz danych w SQL działający w Open Source

1. wydajnościowe

* testy obciążeniowe (500 użytkowników jednocześnie)

1. bezpieczeństwa

* zgodność z RODO (ang. GDPR)
* możliwość samodzielnej zmiany hasła
* walidacja haseł niedopuszczalnych

3.2. Wyłączenie z zakresu testów

Z zakresu testów wyłączone zostają testy statyczne – przegląd dostępnych dokumentów został wykonany na wstępnym etapie wytwarzania oprogramowania.

**4. Podejście do testów**

4.1. Poziom i typ testów

1. systemowe:

* funkcjonalne
* niefunkcjonalne

1. integracji systemów:

* funkcjonalne
* niefunkcjonalne

1. akceptacyjne:

* beta
* funkcjonalne
* niefunkcjonalne

1. pielęgnacyjne:

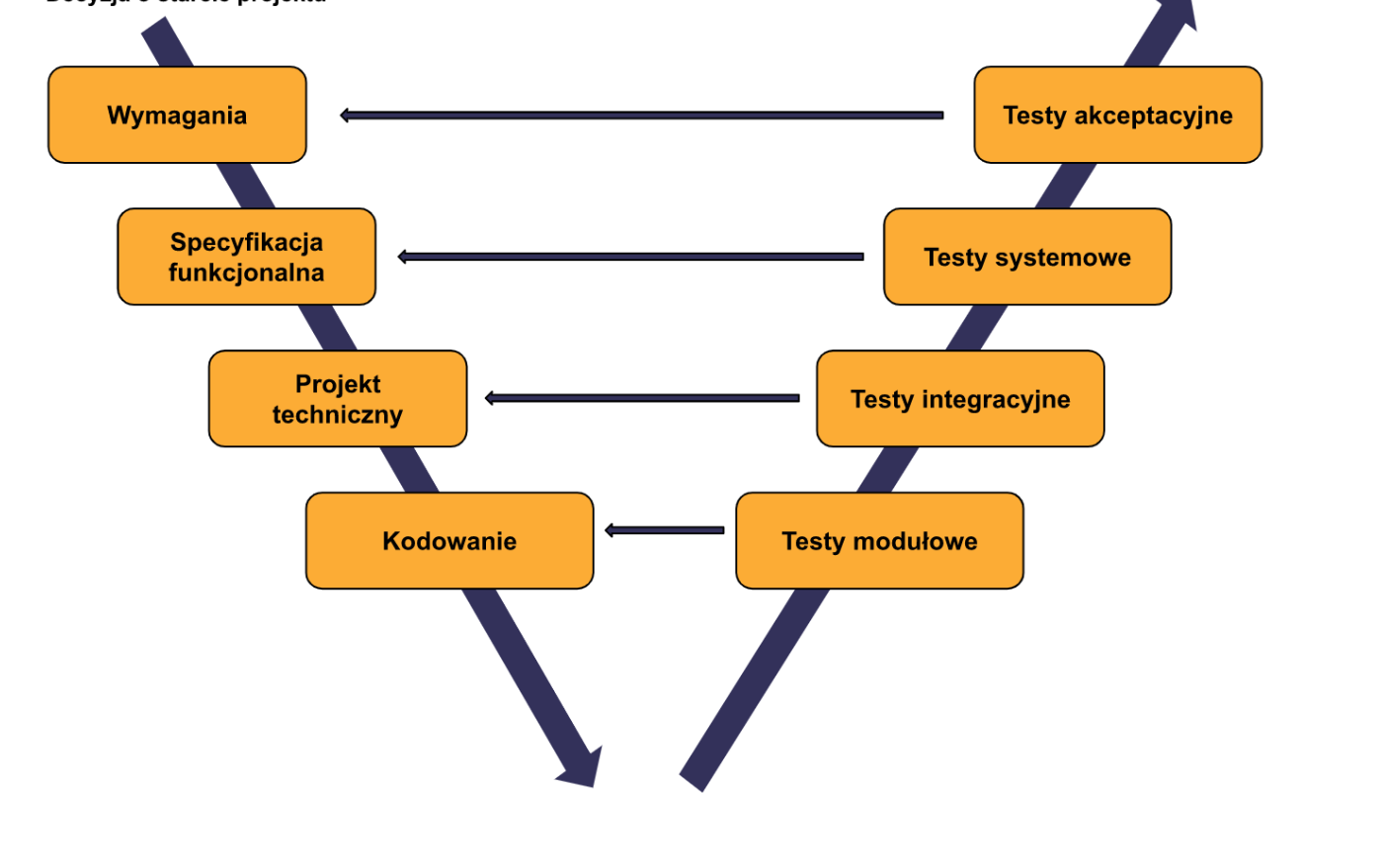
* migracji

4.2. Wykorzystane techniki projektowania testów oraz narzędzia

* eksploracyjne
* białoskrzynkowe
* czarnoskrzynkowe
* komparator
* wydajność
* bezpieczeństwo

4.3. Fazy testów (Danka)

(biorąc pod uwagę, że pracujemy wyłącznie na modelu V)



**Wymagania:** sprawdzenie jak będzie działał system w chwili wystąpienia awarii, czy wszystkie wymagania biznesowe, funkcjonalne zostały spełnione, czy cały proces biznesowy będzie działał na w pełni zintegrowanym systemie, czy wszystkie formularze zostały prawidłowo sporządzone i wprowadzone, czy zostały spełnione wszystkie kryteria uwzględnione w umowie z klientem, czy platforma będzie działać w różnych środowiskach produkcyjnych;

**Specyfikacja funkcjonalna**: testy instalacyjne, testy e2e (sprawdzenie następujących czynności: założenie konta, logowanie i wylogowanie się z systemu, wykonanie wszystkich określonych w wymaganiach czynności w systemie), testy wydajnościowe (badanie procesów pod kątem wykonania w akceptowalnym czasie), testy bezpieczeństwa (uwierzytelniania, autoryzacji),

**Projekt techniczny:** testowanie architektury systemu, sprawdzenie poprawności działania interfejsów pomiędzy modułami i systemami oraz integracji systemu z pakietami i organizacjami zewnętrznymi, czynności związane z konfiguracją systemu;

**Kodowanie:** weryfikacja oprogramowania pod kątem funkcjonalności (modułów, obiektów), testy strukturalne (pokrycie decyzji, instrukcji kodu), testy ze wsparciem bibliotek do testów jednostkowych;

**5. Kryteria (Ola)**

5.1. Kryteria zaliczenia / niezaliczenia testu

Przygotowanych zostanie 90 testów (po 5 testów na każde wymaganie). Aby uznać platformę za gotową do użytkowania, powinien przejść pomyślnie przez 75% testów, czyli 72 (zaokrąglenie 71,25) .

5.2. Warunki rozpoczęcia testów

* przygotowanie środowiska testowego z uwzględnieniem narzędzi wspomagających testowanie
* dostarczenie dokumentacji do wszystkich członków zespołu testowego
* odbycie spotkania przez wszystkich członków zespołu projektowego w celu omówienia backlogu
* wyznaczenie lidera testów

5.3. Warunki zakończenia testów

* spełnienie kryteriów zakończenia testów (75% testów zakończonych powodzeniem)
* koniec budżetu na kontynuację testów
* do release date zostało 2 tygodnie

5.4. Kryteria zawieszenie i wznowienia testów

* zawieszenie: brak supportu narzędzi wspomagających testowanie, wznowienie: znalezienie zastępstwa dla tego narzędzia

**6. Produkty procesu testowego (Paweł)**

6.1. Strategia testowania

W skład testów będzie wchodził 4 osobowy zespół. Testy powinny zająć 6 godzin. Podczas testów generowane będą raporty zawierające podsumowanie prac grup testerów wraz z informacją zwrotną dla kierownika projektu – szczegółową listą przeprowadzonych testów, ich rezultatów oraz wykazem błędów

6.2. Środowisko testowe i narzędzia.

* Wersja testowanej platformy: V 1.0 CodersGuru
* Wersja systemu operacyjnego, na którym będą prowadzone testy: macOS Catalina wersja 10.15.2
* Przeglądarka internetowa: Chrome Wersja 80.0.3987.122 (Oficjalna wersja) (64-bitowa)

Dodatkowe narzędzia:

* wykorzystanie platformy [https://gtmetrix.com](https://gtmetrix.com/) (zostaną dokonane na niej pomiary przy odbiorze szablonu)
* JIRA (zarządzanie testami i zgłaszanie błędów
* Postman (testowanie REST API)
* Selenium (automatyzacja testów regresji)

6.3. Plan testów

Testy będą planowane przed stworzeniem platformy. Testy będą obejmować sprawdzenie każdego modułu z osobna, oraz przetestowanie spójności modułów. Kompletna platforma zostanie poddana całościowym testom. Czynności:

• Wykonanie zestawu testów dla poszczególnych modułów

• Przeprowadzenie testów dla każdego zakończonego modułu

• Przeprowadzenie testów dla całej platformy

**7. Czynności i zadania testowe**

* **planowanie** - wybór narzędzi, metryk, ustalenie technik testowania, harmonogramu testów
* **kontrola** - sprawdzanie przez cały cykl tworzenia platformy czy wszystkie zadania idą zgodnie z planem, raportowanie testów (monitorowanie), działania podejmowane w celu osiągnięcia zamierzonych celów projektu (nadzór)
* **analiza** - wyprowadzenie warunków testowych po dokonaniu przeglądu podstawy testów
* **projektowanie** - wybór poziomu szczegółowości przypadków testowych, techniki projektowania, definiowanie przypadków testowych
* **implementacja** - opracowanie i priorytetyzacja procedur testowych, zbudowanie środowiska testowego, definiowanie i załadowanie danych testowych do środowiska testowego, przygotowanie harmonogramu wykonania testów
* **wykonanie** - testy ręczne i z pomocą narzędzi, weryfikacja uzyskanych wyników, sporządzenie raportów o potencjalnych defektach, logowanie wyników wykonania testów, (testy ujęte w harmonogramie pkt 9)
* **ukończenie testów** - potwierdzenie zamknięcia wszystkich raportów o defektach, przygotowanie raportu podsumowującego, archiwizacja środowiska testowego, danych testowych, infrastruktury i narzędzi wykorzystanych w procesie testowym;

**8. Role i odpowiedzialności, potrzeby szkoleniowe (Paweł)**

8.1. Podział obowiązków procesu testowego

Do przeprowadzenia testów będzie potrzebny zespół osób, który będzie generował testy, zajmował się ich wykrywaniem, planowaniem:

**·Kierownik** – planowanie testów, ustalanie obowiązków w zespole, zdobywanie odpowiednich zasobów, ocena przeprowadzonych testów.

**·Analityk testów** – ustalanie szczegółów przeprowadzonych testów, określenie wyników testów.

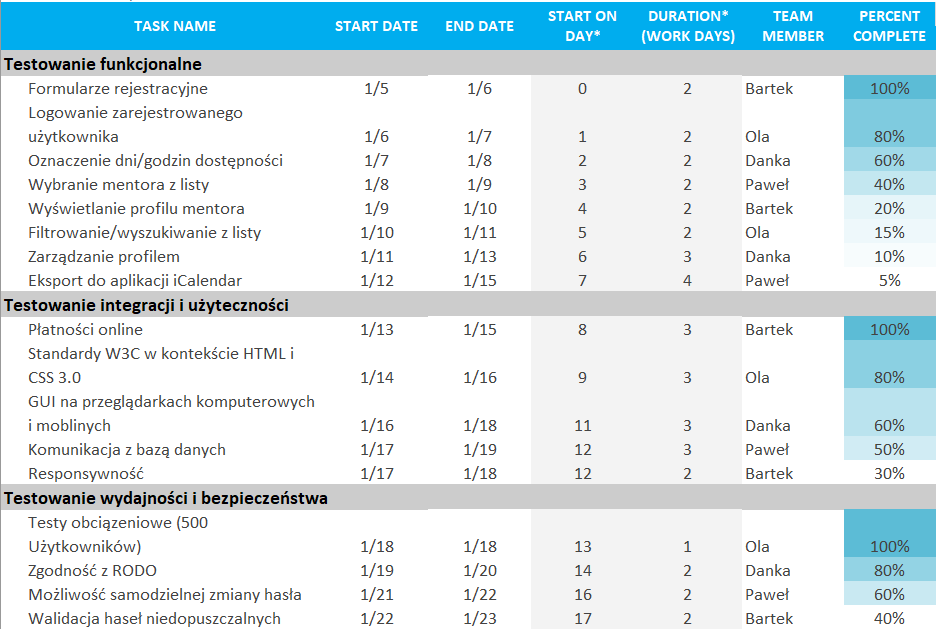
**·Projektant testów** – ustalanie technicznych szczegółów testów

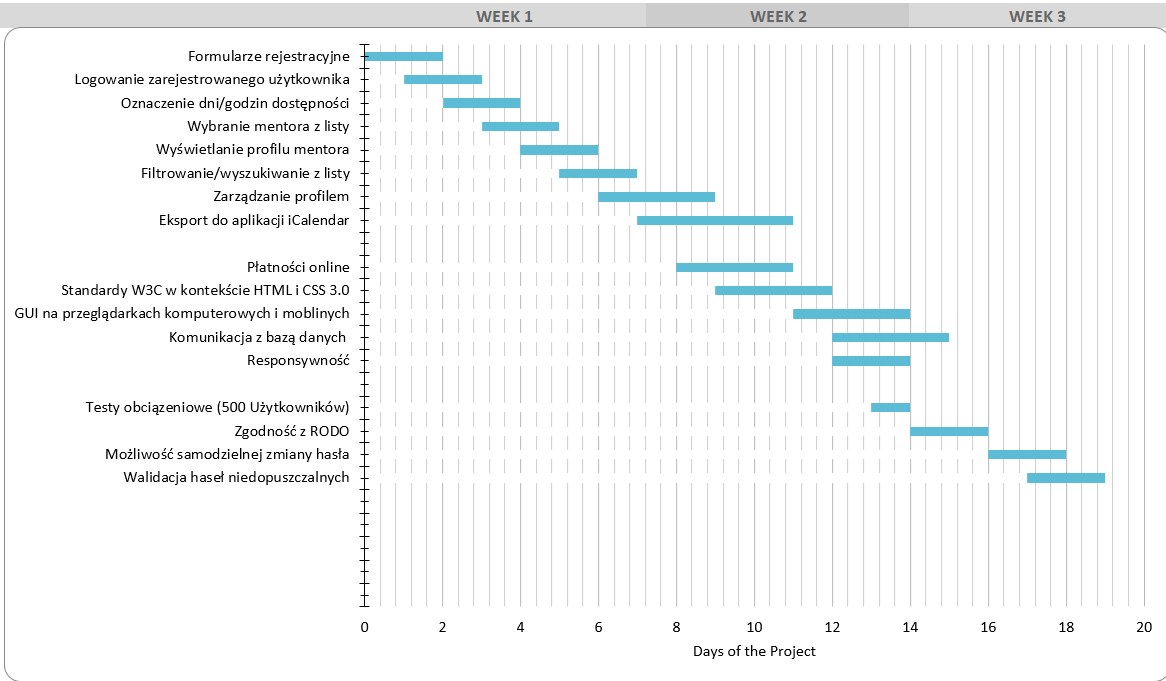
**·Tester** – implementacja testów, przeprowadzanie testów, rejestracja wyników testów

8.2. Potrzeby szkoleniowe

Szkolenie z obsługi narzędzia Selenium WebDriver w Javie

**9. Harmonogram (Bartek)**





**10. Rejestr ryzyk (Danka)**

10.1. Ryzyka projektowe:

Obszary decydujące o możliwości osiągnięcia celów projektu:

* problemy techniczne związane z konwersją danych, planowaniem migracji oraz rozwojem i testowaniem narzędzi do migracji i konwersji danych,
* problemy z komunikacją testerów w związku ze swoimi potrzebami i rezultatami testów,
* stopień w jakim mogą zostać spełnione wymagania przy aktualnych zasobach/ograniczeniach,
* przekroczenie terminu gotowości środowiska testowego;

10.2. Ryzyka produktowe

Potencjalne obszary wystąpienia awarii:

* niska jakość, brak spójności danych (problemy z przekazywaniem danych, problemy z migracją danych, naruszenie standardów danych),
* niewystarczające atrybuty oprogramowania z zakresu funkcjonalności, niezawodności, użyteczności, wydajności;
* niedokładnie przebadane narzędzie niekomercyjne, które chcemy zastosować w projekcie;

**11. Zatwierdzenie planu**

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

|  |  |  |
| --- | --- | --- |
| **Imię i nazwisko** | **Funkcja** | **Organizacja** |
|  |  |  |
|  |  |  |