

# Plan testów systemowych Krajowego Systemu e-Fatur „KSEF”

Wersja: 1.0

Data zatwierdzenia: 03.01.2023

Opracował: Łukasz Jaskuła

# Spis treści

1. WSTĘP I OPIS OGÓLNY.....	3
1.1 Cel dokumentu.....	3
1.2 Skrócony opis przedmiotu testów.....	3
2. PRZEDMIOT TESTÓW.....	4
2.1 Opis systemu KSeF.....	4
2.2 Przedmiot testów.....	4
3. ZAKRES TESTÓW PODSTAWA TESTÓW.....	5
3.1 Zakres planu testów obejmuje.....	5
3.2 Wymagania.....	5
3.3 Podstawa testów.....	5
3.4 Wyłączenia z planu testów.....	5
4. PODEJŚCIE DO TESTÓW.....	7
4.1 Poziom i typ testów.....	7
4.2 Poszczególne wątki procesu testowego.....	7
4.3 Techniki projektowania testów.....	7
4.4 Zastosowane strategie testów.....	8
4.5 Podejście do automatyzacji testów.....	9
4.6 Podejście do testów statycznych instrukcji użytkownika.....	9
5. KRYTERIA ZALICZENIA / NIEZALICZENIA TESTÓW.....	10
6. KRYTERIA WEJŚCIA I WYJŚCIA.....	11
7. ŚRODOWISKO TESTOWE.....	12
7.1 Narzędzia wersje testowanych aplikacji.....	12
7.2 Narzędzia środowiska testowego.....	12
7.3 Narzędzia wspomagające proces testowy.....	12
7.4 Narzędzie zbierające metryki.....	12
8. SKRÓCONA LISTA WYMAGAŃ / FUNKCJONALNOŚCI DO PRZETESTOWANIA.....	13
9. HARMONOGRAM TESTÓW.....	14
9.1 Czynności i harmonogram.....	14
9.2 Osoby odpowiedzialne za monitorowanie i nadzór.....	15
10. RAPORTY Z TESTÓW.....	16
11. ZARZĄDZANIE INCYDENTAMI I BŁĘDAMI.....	17
11.1 Potrzeby szkoleniowe członków zespołu.....	17
11.2 Schemat zgłoszeń błędów.....	17
12. ROLE I ODPOWIEDZIALNOŚĆ.....	18
13. RYZYKA.....	19
14. DEFINICJE.....	20

# 1. WSTĘP I OPIS OGÓLNY

## 1.1 Cel dokumentu

Celem tego dokumentu jest zaprezentowanie planu testów systemowych Krajowego Systemu Faktur Elektronicznych Ministerstwa Finansów.

Dokument określa:

- Zakres testów oraz szczególnie kluczowe obszary
- Zakresy odpowiedzialności
- Typy testów dla poziomu testów systemowych
- Podejście do procesu testowania aplikacji i zastosowane techniki
- Kryteria wejścia i wyjścia
- Analiza ryzyk produktowych i projektowych
- Terminy wykonania poszczególnych

## 1.2 Skrócony opis przedmiotu testów

Krajowy System e-Faktur (KSeF) jest systemem, który umożliwia wystawianie i udostępnianie faktur ustrukturyzowanych. Przedmiotem testów systemowych opisanych w niniejszym dokumencie są testy systemowe aplikacji na przeglądarkę jak i udostępnionego systemu API.

## 2. PRZEDMIOT TESTÓW

### 2.1 Opis systemu KSeF

Krajowy System e-Faktur (KSeF) jest systemem, który umożliwia wystawianie i udostępnianie faktur ustrukturyzowanych. Dostęp do systemu możliwy jest zarówno z poziomu przeglądarki po uprzednim uwierzytelnieniu i zalogowaniu jak i za pomocą API w celu integracji systemu z systemami informatycznymi. Formatem wymiany informacji dla systemu jest faktura elektroniczna w ustandaryzowanym formacie XML. Dzięki standaryzacji faktur możliwe jest automatyczne rachowanie faktur w systemach fiskalnych Państwa Polskiego, zmniejszenie ilości zużywanego papieru (po wprowadzeniu obowiązku używania systemu faktur elektronicznych, faktury tradycyjne przestaną być wymagane). Dodatkowo system przyspieszy wymianę danych pomiędzy kontrahentami, gdyż po wystawieniu faktury odbiorca (kupujący) usługi bądź towaru będzie mógł otrzymać fakturę bez konieczności oczekiwania na wersję papierową. Dodatkowo system uniezależnia obieg faktur od innych metod obiegu jak poczta, tradycyjna bądź elektroniczna. Wprowadzenie systemu wiąże się z wypełnieniem zobowiązań Państwa Polskiego wobec cyfryzacji organów administracyjnych.

Dodatkowo system ma zwiększyć wykrywalność nadużyć, oszustw podatkowych bądź nieuczciwości w przypadku podmiotów, które nie regulują należności wobec swoich dostawców.

### 2.2 Przedmiot testów

Przedmiotem testów objętych tym planem jest:

- Wersja od Brokera oraz API serwisu Wersja **1.0.9**
  - <https://int-brokerpefexpert.efaktura.gov.pl/zaloguj>

Witryna internetowa zezwala na podstawowe operacje na koncie podatkowym powiązonym z numerem NIP jak ręczne wystawianie i sprawdzanie faktur, wystawianie korekt, pobieranie faktur w formacie XML.

W przypadku wersji API funkcjonalność jest rozszerzona o dodatkowe elementy.

- Instrukcja obsługi systemu

### 3. ZAKRES TESTÓW PODSTAWA TESTÓW

#### 3.1 Zakres planu testów obejmuje

W przypadku systemu planowane jest sprawdzenie, czy system spełnia wymagania funkcjonalne przedstawione w instrukcji obsługi, która jest jednocześnie podstawą testów i służy do określenia warunków testowych. Czynności testowania będą dotyczyć tylko własności funkcjonalnych systemu.

Zostaną także wykonane testy нефункционалне systemu. Zostanie nimi objęty graficzny interfejs użytkownika na różnych platformach z wykorzystaniem różnych przeglądarek internetowych w celu oceny użyteczności dla różnych grup użytkowników jak i przenaszalności i możliwości pracy na różnych środowiskach.

Ponadto wykonane zostaną testy statyczne instrukcji obsługi mające na celu sprawdzenie czy dokumentacja systemu jest spójna, jednoznaczna oraz zgodna z docelowym systemem udostępnianym użytkownikom.

#### 3.2 Wymagania

Czynności testowe mają wykazać czy wymagania funkcjonalne zostały spełnione. Wyczerpującą informację na tematy wymagań funkcjonalnych oraz matrycę śledzenia powiązań można uzyskać w dokumentacji testerskiej.

W przypadku wymagań нефункционалных testy mają wykazać, że system jest przyjazny dla niedoświadczzonego użytkownika, informacje wyświetlane są czytelnie.

#### 3.3 Podstawa testów

Podstawą testów w przypadku aplikacji internetowej są historyjki użytkownika oraz przypadki użycia. W przypadku wersji od brokera podstawą testów jest udostępniona instrukcja użytkownika.

#### 3.4 Wyłączenia z planu testów

Typy testów nieobjęte testami są następujące:

- Testy нефункционалне wydajnościowe: sprawdzenie systemu jak zachowuje się w różnych stanach obciążenia jest niemożliwe ze względu na brak dostępności sprzętu i oprogramowania do przeprowadzenia takich testów. Testy te zostaną zlecone firmie zewnętrznej, która przeprowadzi testy za pomocą narzędzia Jmeter oraz LoadUI.

- Testy niefunkcjonalne bezpieczeństwa: wykonanie testów penetracyjnych przez wzgląd na brak doświadczonego personelu oraz strategiczny charakter systemów zostanie zlecony firmie zewnętrznej specjalizującej się w testach bezpieczeństwa.

## 4. PODEJŚCIE DO TESTÓW

### 4.1 Poziom i typ testów

Rozpatrywanym poziomem testów są testy systemowe aplikacji oraz instrukcji obsługi. Wcześniejsze poziomy testów zostały zrealizowane na poprzednich etapach rozwoju oprogramowania. Nie zgłoszono żadnych defektów w oprogramowaniu, które zostały przeniesione na poziom testów systemowych. Wszystkie błędy w kodzie na niższych poziomach zostały usunięte a kolejny przebieg testów potwierdzających i regresji nie wykryły nowych defektów w oprogramowaniu.

### 4.2 Poszczególne wątki procesu testowego

Ze względu na złożoność systemu i dwie aplikacje do jego obsługi proces testowy zostanie podzielony na wątki. Poszczególne wątki zostaną podzielone pomiędzy zespoły i część z nich będzie realizowana równolegle dla przyspieszenia prac testowych.

Wątek	Przedmiot testów	Narzędzie testowe / opis
1	Instrukcja obsługi	Testy statyczne
2	Aplikacja Brokera	Testy manualne
3	Aplikacja Brokera	Testy automatyczne
4	Aplikacja Brokera	Testy niefunkcjonalne GUI
5	Aplikacja Brokera	Testy API

### 4.3 Techniki projektowania testów

Wykorzystywane techniki i przykłady zastosowania

W celu przetestowania wymagań zostaną wykonane odpowiednie testy funkcjonalne z zastosowaniem technik opartych na specyfikacji – technik czarnoskrzynkowych.

Testowanie czarnoskrzynkowe				
	Podział na klasy równoważności	Ręczne wpisanie wartości podatku	nieobowiązującego w systemie podatkowym	VAT

	Analiza wartości brzegowych	Wskazanie zerowej wartości zakupionego towaru / usługi dla przypadku negatywnego
	Testowanie przejść pomiędzy stanami	Sprawdzenie przejścia pomiędzy poszczególnymi etapami w aplikacji dla wprowadzenia faktury.
Testowanie oparte na doświadczeniu		
	Zgadywanie błędów	Sprawdzenie najbardziej newralgicznych miejsc w oparciu o doświadczenie testerów.
	Lista kontrolna	Sprawdzenie systemu w oparciu o dwie listy kontrolne: Dotycząca funkcjonalności serwisu internetowego oraz czytelności i poprawności wyświetlania się serwisu.
	Testowanie eksploracyjne	Przeprowadzenie 3 sesji w celu wykrycia potencjalnych błędów nieobjętych innymi testami.

#### 4.4 Zastosowane strategie testów

W przypadku projektowania przypadków testowych zastosowano następujące strategie testowania:

- Analityczna: oparto się na analizie wymagań stawianych wobec systemu
- Oparta na modelu: podstawowym modelem jest proces obiegu faktury zaczynając od jej utworzenia w firmie (dostawcy usługi) do firmy, która jest odbiorcą usługi. Rozpatrywanym odcinkiem procesu jest moment wprowadzania faktury do systemu oraz jej odbiór po stronie kupującego.
- Metodyczna: wykorzystano listę kontrolną stosowaną w organizacji do oceny aplikacji internetowych
- Reaktywna: w przypadku wykrycia defektów w oprogramowaniu nakłady na testowanie danej funkcji zostaną zwiększone w celu położenia większego nacisku na moduł oraz minimalizację ryzyka przeniesienia niewykrytych błędów na poziom akceptacyjny. Skutkiem ubocznym tego podejścia jest możliwość wydłużenia się procesu testowego w przypadku wykrycia większej ilości błędów.
- Konsultatywna: W związku z tym, iż system dotyczy testów systemu podatkowego do udziału w specyfikacji danych testowych zaproszono specjalistów do spraw księgowości i podatków
- Minimalizacji regresji: Przy testowaniu najważniejszych funkcjonalności w celu zapewnienia powtarzalności oraz dokładności oceny wyników



## 4.5 Podejście do automatyzacji testów

W przypadku kryteriów, którym będą podlegać testy skierowane do automatyzacji są to przypadki:

- powtarzalnych czynności jak rejestracja użytkownika bądź logowanie
- ścieżki, w przypadku których powtarzalność ma duże znaczenie
- których zautomatyzowanie pozwoli zaoszczędzić czas i zasoby w projekcie

## 4.6 Podejście do testów statycznych instrukcji użytkownika

Instrukcja użytkownika zostanie przetestowana statycznie w procesie przeglądu technicznego przy udziale facylitatora. Celem głównym przeglądu jest sprawdzenie czy instrukcja jest spójna, czy nie zawiera błędów oraz w sposób wystarczający jest środkiem, który pozwoli użytkownikom na pomoc przy używaniu systemu.

Zastosowane techniki to przegląd oparty na liście kontrolnej dla dokumentacji użytkowników opracowany w firmie dodatkowo technika oparta na rolach, uwzględniająca różnych użytkowników końcowych jak: księgowa, właściciel firmy, użytkownik niedoświadczony.

## 5. KRYTERIA ZALICZENIA / NIEZALICZENIA TESTÓW

Kryteria zaliczenia oraz niepowodzenia testów dla każdego konkretnego przypadku testowego zostaną opisane osobno w poszczególnych przypadkach testowych.

W przypadku testów funkcjonalnych kryteriami warunkującymi zaliczenie testów pozytywnych jest spełnienie wymagań stawianych przed systemem. Dla przytoczenia jednego przykładu jest to możliwość sprawdzenia wprowadzonej faktury oraz zbieżność wszystkich danych zapisanych w systemie wraz z wyświetlonymi. Dla testów negatywnych będzie to sprawdzenie zachowania systemu w przypadku wprowadzenia niepełnych danych, błędnych danych lub danych które w jakikolwiek inny sposób sprawiają, że faktura z punktu widzenia systemu jest nieważna. Przykładowo negatywnymi przypadkami testowymi będą przypadki w których wprowadzono błędne dane NIP lub numerów kont kontrahentów. W przypadku braku odpowiedzi systemu i komunikaty dotyczącego błędu przypadek testowy zostanie uznany za niezaliczony.

W przypadku testów нефункциональных czytelności i używalności systemu przez różnych użytkowników sprawdzony zostanie interfejs użytkownika a wszystkie uwagi dotyczące czytelności zostaną zgłoszone w końcowym raporcie procesu testowego wraz z ewentualnymi zaleceniami zmian. Kryterium zaliczenia / niezaliczenia jest w tym przypadku zdolność systemu do używania go przez różnych użytkowników na różnych platformach i różnym sprzęcie.

W przypadku całości aplikacji testy na poziomie systemowym zostaną uznane za zaliczone a system przekazany do kolejnego etapu w przypadku nie wystąpienia żadnych błędów testów powiązanych z wymaganiami określonymi jako krytyczne i ważne. W przypadku innych decyzja zostanie podjęta po uprzednim przedstawieniu informacji o niezaliczonych przypadkach testowych interesariuszom.

Dla testów środowiska API koniecznym jest uzyskanie prawidłowości w pracy z tokenami uwierzytelniającymi oraz poprawnością przyjmowania żądań wysyłanych za pomocą narzędzi wspomagających testowanie API dla przygotowanych danych.

## 6. KRYTERIA WEJŚCIA I WYJŚCIA

Kryteria wejścia dla rozpoczęcia procesu testowego

- Działające środowisko testowe
- Dostępność danych testowych
- Dostępność zespołu testowego
- Dostępność dokumentacji
- Zakończenie szkolenia personelu przydzielonego do szkoleń uzupełniających
- Przekazanie systemu do fazy testów systemowych
- Brak nierozwiązanych defektów z testów integracyjnych
- Zaakceptowana dokumentacja testowa
- Zaakceptowany plan testów

Kryteria wyjścia

- Przetestowano 98% wymagań w tym 100% wymagań powiązanych z wymaganiami krytycznymi i ważnymi
- Wszystkie błędy krytyczne i ważne zostały usunięte
- Testy statyczne instrukcji obsługi nie wykazały uchybień lub zostały one usunięte
- Wykorzystano całkowity przewidziany czas na testowanie

Kryteria zawieszenia

- Wykryto krytyczny błąd w systemie, który uniemożliwia przeprowadzenie wiarygodnych testów i wymaga natychmiastowego usunięcia przez zespół programistów. Wznowienie może nastąpić po przeprowadzeniu testu potwierdzającego przypadku testowego, który ujawnił ww. błąd.

## 7. ŚRODOWISKO TESTOWE

### 7.1 Wersje testowanych aplikacji

- Wersja od Brokera oraz API serwisu Wersja **1.0.9**

### 7.2 Narzędzia środowiska testowego

Testy manualne zostaną zrealizowane na następujących systemach:

- System Windows 10 kompilacja 19043.1766
- System Ubuntu 22.04 LTS
- Przeglądarka Firefox: 102 64 bit
- Przeglądarka Brave: 1.40.109
- Komputer z monitorem 21 cali

Testy automatyczne zostaną przeprowadzone za pomocą

- Python 3.10.5
- Selenium 4.2.0
- Robot Framework

Testy API zostaną wykonane z pomocą

- Postman 9.24.0

### 7.3 Narzędzia wspomagające proces testowy

Przypadki testowe zostaną zapisane w Test Link

Matryca śledzenia powiązań między wymaganiami a przypadkami testowymi zostanie utworzona w Microsoft Excel

Komunikacja zespołu oraz delegowanie zadań zostanie wykonane w systemie Jira

### 7.4 Narzędzie zbierające metryki

Metryki zostaną zebrane w przygotowanym zewnętrznym formularzy Excel wraz z generowanym automatycznie raportem graficznym.

## 8. SKRÓCONA LISTA WYMAGAŃ / FUNKCJONALNOŚCI DO PRZETESTOWANIA

Poniżej przedstawiono skrócone wymagania co do aplikacji. Szczegółowy opis znajduje się w osobnym dokumencie „Specyfikacja KSEF BROKER”. W dokumencie rozpisano szczegółowe wymagania poszczególnych elementów aplikacji. Poniższy spis ma na celu jedynie unaocznić czytelnikom planu testów podstawowego celu wdrożenia systemów w użycie i korzyści dla użytkowników.

1. System umożliwia wprowadzanie wystawionych faktur (sprzedaży) podmiotów i ich automatyczne przesyłanie do Izby Skarbowej.
2. System umożliwia odczytywanie faktur (zakupu) podmiotów.
3. System umożliwia współpracę z zewnętrznym API do obsługi faktur. Wysyłki oraz odbioru.
4. System jest zabezpieczony przed niepożądanym dostępem.
5. Umożliwione jest wysyłanie powiadomień mailowych do podmiotów.
6. Faktury pomiędzy powiązаныmi podmiotami (kupujący / sprzedający) są przekazywane i są spójne.
7. System umożliwia przeglądanie faktur, zarówno sprzedaży jak i zakupu podmiotów gospodarczych.
8. System wprowadza możliwość samofakturowania przez kontrahentów. Uprawnienia te można nadać jak i odebrać w dowolnym momencie.
9. GUI systemu umożliwia filtrowanie dokumentów po kategoriach
10. GUI systemu umożliwia sortowanie dokumentów po kategoriach

### Przykładowe historyjki użytkownika

Jako	Zalogowany użytkownik na konto podmiotu gospodarczego o wyspecyfikowanym numerze NIP
Chcę	Wprowadzić ręcznie dane faktury sprzedaży
Aby	Przekazać dane do Izby Skarbowej Państwa Polskiego oraz przesłać fakturę do kontrahenta

Jako	Zalogowany użytkownik na konto podmiotu gospodarczego o wyspecyfikowanym numerze NIP
Chcę	Sprawdzić status faktury sprzedaży
Aby	Dowiedzieć się czy kupujący otworzył fakturę

Jako Zalogowany użytkownik na konto podmiotu gospodarczego o wyspecyfikowanym numerze NIP

Chcę Wprowadzić korektę faktury sprzedaży

Aby Skorygować wartość wcześniej wystawionej faktury i przekazać dane do Izby Skarbowej Państwa Polskiego oraz do kontrahenta

Jako Zalogowany użytkownik na konto podmiotu gospodarczego o wyspecyfikowanym numerze NIP

Chcę Sprawdzić faktury zakupowe

Aby Móc je opłacić, zaksięgować oraz wyliczyć należny podatek

## 9. HARMONOGRAM TESTÓW

Przedstawiony harmonogram testów wraz z datami realizacji.

### 9.1 Czynności i harmonogram

		Początek procesu	Koniec procesu
1.	Planowanie testów		
2.	Analiza testów		
2.1	Dokonanie analizy podstawy testów		
2.2	Definiowanie warunków testowych aplikacja Brokera		
3.	Projektowanie testów		
3.1	Aplikacja Brokera – identyfikacja danych testowych		
3.2	Aplikacja Brokera – projektowanie przypadków testowych		
3.3	Stworzenie matrycy śledzenia powiązań		
3.4	Aplikacja Brokera – projektowanie przypadków testowych dla GUI		
4.	Implementacja testów		
4.1	Aplikacja Brokera – utworzenie procedur testowych		
4.2	Aplikacja Brokera – utworzenie skryptów automatycznych		
4.3	Sprawdzenie danych testowych		
4.4	Aktualizacja matrycy dwukierunkowego śledzenia powiązań		
5.	Wykonanie testów		
5.1	Testy statyczne instrukcji użytkownika		
5.2	Aplikacja Brokera – wykonanie testów manualnych		
5.3	Aplikacja Brokera – uruchomienie skryptów automatycznych		
5.4	Aplikacja Brokera – wykonanie testów GUI		
5.5	Aplikacja Brokera – API wykonanie testów z POSTMAN		
5.5	Powtórzenie czynności testowych dla testów potwierdzających i regresji		
6.	Ukończenie testów		
6.1	Zebrań danych z wykonanych czynności testowych		
6.2	Utworzenie raportu z testów		
6.3	Archiwizacja danych		
6.4	Przekazanie testaliów		
6.5	Przeanalizowanie wniosków z procesu testowego dla wykorzystania w kolejnych projektach		
7.	Przekazanie systemu do testów akceptacyjnych		

## 9.2 Osoby odpowiedzialne za monitorowanie i nadzór.

Osobami odpowiedzialnymi za monitorowanie i nadzór nad testami są kierownik działu testów oraz lider testów. Odpowiadają za realizowanie zadań zgodnie z przedstawionym harmonogramem i podejmują działania w celu jego dotrzymania.

Funkcja	Imię	Nazwisko
Kierownik dz. testów		
Lider zespołu 1		
Lider zespołu 2		



## 10. RAPORTY Z TESTÓW

Poniżej przedstawiono charakterystykę raportu z testów, który zostanie dostarczony po ukończeniu testów. Za zatwierdzenie planu testów odpowiadają lider testów oraz kierownik działu testów.

Zawartość raportu:

- podsumowanie wykonanych testów
- zestawienie wszystkich przypadków testowych i ich statusów
- czasu realizacji poszczególnych wątków procesu testowego
- informacje na temat wykrytych zdarzeń
- poziom wykonania zaplanowanych prac testowych
- informacje o odstępstwach od planu
- informacje o czynnikach blokujących proces testowy
- informacje na temat ryzyka (resztkowego)
- informacje o jakości produktu z punktu widzenia kryteriów wyjścia

## 11. ZARZĄDZANIE INCYDENTAMI I BŁĘDAMI

Błędy zgłaszane są przez poszczególnych testerów bądź do team liderów, odpowiedzialnych za poszczególne wątki, w przypadku jego nieobecności do lidera testów. Błędy zgłaszane zostaną w systemie Jira a na potrzeby dokumentacji zostaną opisane w znormalizowanym formacie wskazanym poniżej.

Za testy potwierdzające odpowiadają testerzy, którzy bezpośrednio zgłosili defekt.

### 11.1 Potrzeby szkoleniowe członków zespołu

Osoba	Szkolenie	Czas realizacji
Tester 1	Postman	

### 11.2 Schemat zgłoszeń błędów

Błędy będą zgłaszane w znormalizowanym formacie. Informacje będą przedstawiane tak jak poniżej.

Numer zgłoszenia		Rezultat oczekiwany
Wymaganie		
Przyp. Test.		
Data wykrycia		Rezultat uzyskany
Poziom ryzyka		
Zgłaszający		
		Opis zgłaszającego
Identyfikacja elementu testowego		
Inne informacje pomocne w odtworzeniu błędu		

## 12. ROLA I ODPOWIEDZIALNOŚĆ

Dział testów			
Osoba	Przełożony	Zespół	Odpowiedzialność
Kierownik dz. testów	-/-	-/-	Nadzór nad całością procesu testowego
Lider zespołu 1	Kierownik dz. testów	Zespół 1	Nadzór nad pracą zespołu 1
Tester 1	Lider zespołu 1	Zespół 1	Wątki 1, 5, 6, 7
Tester 2	Lider zespołu 1	Zespół 1	
Tester 3	Lider zespołu 1	Zespół 1	
Tester 4	Lider zespołu 1	Zespół 1	
Lider zespołu 2	Kierownik dz. testów	Zespół 2	Nadzór nad pracą zespołu 2
Tester 5	Lider zespołu 2	Zespół 2	Wątki procesu testowego 2, 3, 4
Tester 6	Lider zespołu 2	Zespół 2	
Tester 7	Lider zespołu 2	Zespół 2	
Tester 8	Lider zespołu 2	Zespół 2	

### Pozostałe osoby w projekcie

Product owner - kontakt z klientem, informacja o postępie i wykrywanych defektach

Konsultant ds. księgowości i podatków – merytoryczna pomoc w formułowaniu zapytań.

## 13. RYZYKA

Ryzyka projektowe: zaliczane są do nich ryzyka związane z przekroczeniem terminów, zmian w harmonogramie lub niskiej jakości pracy zespołu:

- niedoświadczenie testerów
- brak umiejętności w obsłudze narzędzi do testowania
- braki kadrowe – urlopy, zwolnienia chorobowe
- brak gotowości środowiska
- opóźnienia w dostarczeniu wymaganych narzędzi

Ryzyka produktowe: związane z obszarem awarii dostarczonego oprogramowania

- dostarczenie wadliwego oprogramowania
- oprogramowanie nie będzie wystarczająco wydajne (obsługa wielu zapytań)
- problem ze spójnością danych (znormalizowany format faktur)

## 14. DEFINICJE

KSEF – Krajowy System e- Faktur