

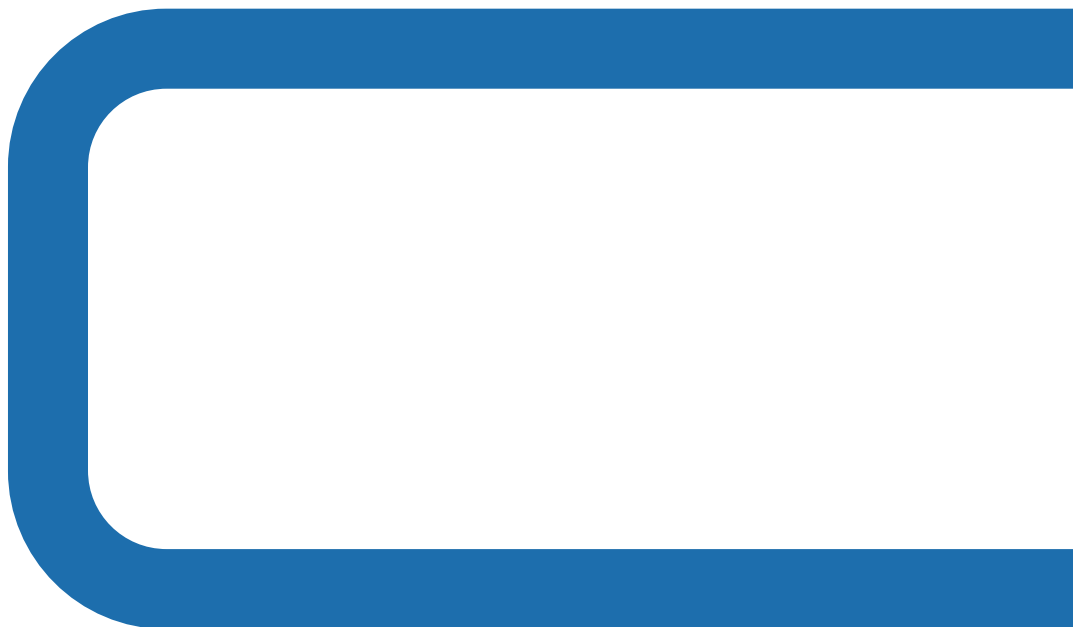
### **ODPOWIEDŹ wersja 1.0**

**do Wstępnych Konsultacjach Rynkowych  
na usługę „Budowy i utrzymania systemu  
informatycznego dla Inspekcji Weterynaryjnej”**

Przygotowane: Maciej Tomczak

[maciej.tomczak@euvic.pl](mailto:maciej.tomczak@euvic.pl)

Data: 05.03.2021



## Odpowiedzi na pytania

### 1. Jaka będzie preferowana przez Wykonawcę metodyka prowadzenia projektu.

Proponuje się zastosowanie modelu mieszanego zarządzania tj. metodyki kaskadowej (waterfall) z zastosowaniem metodyk zwinnych (agile). Projekt jako całość byłby prowadzony wg metodyki kaskadowej, natomiast poszczególne moduły, wytwarzane w ramach pojedynczych faz byłyby wytwarzane wg metodyk zwinnych. Takie podejście pozwoli w sposób optymalny wykorzystać zasoby i czas (krótki) na wytworzenie wymaganych przez Zamawiającego produktów.

W szczególności proponuje się zastosować metodykę Prince2, dojrzałą i dopracowaną metodykę kaskadową, która jest powszechnie stosowana również w powiązaniu z metodykami zwinnymi.

#### **Model iteracyjno-przyrostowy (metodyki zwinne)**

1. Prace nad implementacją poszczególnych produktów (modułów) warto prowadzić w modelu iteracyjno-przyrostowym bazującym na metodyce SCRUM (np. zgodnie z „Scrum Guide” 2020, dostępny m.in. w j. polskim na stronie <http://www.scrumguides.org/>).
2. Praca podzielona będzie na krótkie (maksimum 4 tygodniowe, zalecane 2 tygodniowe) odcinki czasowe (sprinty), zakończone powstaniem nowej, gotowej do wdrożenia wersji oprogramowania.
3. Na początku każdego sprintu określany będzie zakres prac które mają być wykonane w ciągu rozpoczynanego sprintu.
4. Każdy sprint zakończony jest prezentacją funkcjonalności zrealizowanych w ramach sprintu.
5. Jeśli nowa wersja została zaakceptowana przez Zamawiającego instalowana jest w środowisku testowym; środowisko testowe jest przez cały czas dostępne dla Zamawiającego i zawiera gotowe, zakończone funkcjonalności systemu.
6. Po przeprowadzeniu testów akceptacyjnych wersja ze środowiska testowego jest instalowana w środowisku produkcyjnym zgodnie z przyjętym harmonogramem. Tu proponuje się, aby produkcyjnie były udostępniany jeden lub kilka powiązanych modułów. Może również zostać przyjęte udostępnienie systemu na koniec Etapu.

Takie podejście do projektu pozwoli na właściwe zarządzanie całością projektu, przy maksymalizacji wykorzystania dostępnych zasobów zarówno po stronie Zamawiającego jak i Wykonawcy, jednocześnie zapewniając jak najszybsze otrzymanie użytecznych wyników biznesowych częściowych produktów (przyrostów po każdym sprincie).

## 2. Zespół projektowy:

### a) po stronie Zamawiającego

Poniżej oczekiwane role projektowe po stronie Zamawiającego, z pominięciem ról związanych z Zarządzaniem Projektem. Ważne, aby osoby oddelegowane do projektu mogły poświęcić wystarczającą część czasu pracy dla projektu (w okresie testów testerzy powinni móc pracować w wymiarze pełnego etatu wyłącznie na rzecz projektu).

**Główny Product Owner** – osoba odpowiedzialna za całość rozwiązania, dobrze znająca całość biznesu Zamawiającego, rozumiejąca powiązania pomiędzy modułami, zależności od systemów zewnętrznych oraz uwarunkowania związane z budowanym systemem. Osoba decyzyjna w zakresie wyboru rozwiązań w obszarze biznesowym.

Liczba osób w roli: jedna (dokładnie jedna), osoba na pełen etat w okresie trwania całego projektu.

**Product Owner** – osoba biznesowa odpowiadająca za wymagania związane z danym modułem, dobrze znająca biznes związany z danym obszarem.

Liczba osób w roli: jedna dla każdego obszaru biznesowego (równoważne z modułem lub grupą modułów powiązanych).

**Zespół testowy** – osoby wykonujące testy (w ramach fazy projektu oraz w ramach testów akceptacyjnych). Dobrze jeśli zna stronę biznesową testowanego obszaru.

Liczba osób w roli: kilka do kilkunastu (liczba dostosowana do liczby funkcji w danych obszarze, co najmniej 1- 2 osoby dobrze znające obszar od strony biznesowej).

**Zespół ds. infrastruktury** – osoby odpowiadające za infrastrukturę sprzętowo-systemową/chmurę po stronie Zamawiającego, administratorzy, architekci techniczni. Zespół IT będzie opiniował projekt techniczny systemu, dbał o spójność z rozwiązaniami technicznymi Zamawiającego, nadzorował konfigurację systemu, wykonywał czynności administracyjne, odbierał dokumentację eksploatacyjną. Po zakończeniu projektu przejmie pełne administrowanie i zarządzanie systemem na poziomie technicznym.

Liczba osób w roli: jeden architekt techniczny, kilka osób – specjaliści w różnych obszarach IT.

**Ekspert ds. bezpieczeństwa** – specjalista w dziedzinie bezpieczeństwa systemów informatycznych.

Liczba osób w roli: jeden

### b) po stronie Wykonawcy

**Kierownik Projektu** – osoba kierująca projektem ze strony Wykonawcy, z co najmniej 10 letnim doświadczeniem, mogąca się wykazać prowadzeniem co najmniej trzech projektów informatycznych (wytworzenie nowego oprogramowania lub utrzymanie i rozwój istniejącego oprogramowania), w tym co najmniej jeden o wartości pow. 5 mln PLN brutto. Posiadająca ważny certyfikat PRINCE2® Practitioner. Wymagane wykształcenie wyższe.

Liczba osób w roli: jeden

**Architekt systemów informatycznych** – osoba posiadająca minimum 5-letnie doświadczenie w projektowaniu architektury złożonych, skalowalnych systemów IT działających w środowisku zintegrowanym, brała udział jako architekt rozwiązań w co najmniej 2 projektach obejmujących swoim

zakresem zaprojektowanie i wdrożenie systemów informatycznych, posiadająca certyfikat TOGAF lub równoważny. Wymagane wykształcenie wyższe.

Liczba osób w roli: jeden

**Scrum MASTER** – w przypadku decyzji o zastosowaniu metodyki zwinnej scrum - osoba posiadająca co najmniej 3-letnie doświadczenie w roli Scrum Mastera, mogąca się wykazać udziałem w co najmniej 2 projektach jako Scrum Master. Posiadająca certyfikat PROFESSIONAL SCRUM MASTER. Wymagane wykształcenie wyższe.

Liczba osób w roli: jeden

**Analitik** – minimum 3-letnie doświadczenie w zakresie analizy i projektowania rozwiązań informatycznych, potwierdzone udziałem w co najmniej dwóch projektach informatycznych o podobnym zakresie. Osoba znająca metodyki analizy projektowej, posługująca się biegłe narzędziami z zakresu modelowania (np. UML). Wymagane wykształcenie wyższe.

Liczba osób w roli: minimum 2

**Inżynier d/s bezpieczeństwa** – osoba, która pracowała na stanowisku inżyniera d/s bezpieczeństwa co najmniej 5 lat, posiadająca co najmniej 3-letnie doświadczenie w co najmniej dwóch projektach budowy systemów informatycznych jako inżynier ds. bezpieczeństwa, posiadająca doświadczenie dot. bezpieczeństwa systemów informatycznych z zakresu prowadzenia audytu bezpieczeństwa informacji CISA lub CISSP poświadczona ważnym certyfikatem.

**Kierownik testów** – osoba z co najmniej 3-letnim doświadczeniem na podobnym stanowisku; brała udział w co najmniej jednym projekcie polegającym na budowie lub rozwoju systemu informatycznego o wartości minimum 5 mln PLN brutto w roli kierownika testów, posiadająca certyfikat - ISTQB Advanced Level lub ISEB Intermediate Level.

**Specjalista ds monitorowania systemów IT** – osoba, która pracowała jako Administrator IT lub DevOps co najmniej 3 lata, w tym w wyżej wymienionych rolach brała udział w co najmniej 2 projektach, mająca doświadczenie w monitorowaniu systemów IT. Posiadane kompetencje poświadczone certyfikatem np. ZABBIX 4.0 Certified Specialist. Wymagane wykształcenie wyższe informatyczne lub pokrewne.

### 3. Harmonogram projektu – dla wszystkich 3 etapów projektu.

Etapy i fazy odbywają się wg metodyki kaskadowej. Prace nad modułami (grupami modułów) wg metodyki zwinnej.

| Faza | Nazwa   | Miesiące | Opis   |
|------|---|----------|--|
|      | <b>ETAP 1</b>   |          |  |
| 1    | Harmonogram / analiza wstępna   | 1 - 2    | Organizacja projektu / wykonanie analizy wstępnej  |
| 2    | Dostawa, instalacja, konfiguracja infrastruktury sprzętowo-systemowej | 2 - 3    | Projekt infrastruktury sprzętowo-systemowej z uwzględnieniem chmury, uruchomienie i uzgodnienie  |
| 3    | Moduł(y) A  | 3 - 5    | W jednej fazie przygotowanie jednego lub kilku powiązanych merytorycznie modułów; faza obejmuje kompletny cykl realizacyjny metodyką zwinną (np. scrum) – analiza i projektowanie, |

|          |   |  |   |
|----------|---|--|---|
|          |   |  | wykonanie przyrostowe modułów, testy przyrostów, dokumentacja użytkownika   |
| 4<br>... | Moduł(y) B                                      | 3 - 5  | j.w. kolejny moduł lub moduły, sekwencja modułów ułożona wg zależności merytorycznych pomiędzy modułami   |
| 5        | Testy akceptacyjne (UAT) / wdrożenie pilotażowe | 2 - 4  | <p>2 – 6 tygodni testów UAT, w tym testy wydajnościowe i audyt bezpieczeństwa. Testy podzielonych na 2 – 3 cykle testowe. Po pozytywnym zakończeniu uruchomienie pilotażowe systemu.</p> <p>Do wdrożenia pilotażowego powinna zostać wybrana niewielka grupa reprezentatywnych użytkowników. Pilotaż powinien trwać 1 – 2 miesiące. Po pilotażu powinien być przewidziany czas na wprowadzenie ewentualnych korekt systemu przed wdrożeniem masowym oraz ich odbiór (testy) przez Zamawiającego.</p> <p>Jeśli jest to uzasadnione, to przeprowadzenie UAT i wdrożeń po wybranych fazach przygotowania modułu (modułów).</p> |
| 6        | Wdrożenie masowe                                | 1 - 2  | Po zakończeniu pilotaży system powinien zostać udostępniony dla użytkowników masowych.  |
| 7        | Szkolenia                                       |  | <ol style="list-style-type: none"> <li>1. Dla osób po stronie Zamawiającego – po każdym zakończonym module.</li> <li>2. Dla użytkowników – po wdrożeniu produkcyjnym, w postaci eLearningu (materiały do zapoznania się, ćwiczenia, testy)</li> </ol>   |
| 8        | Utrzymanie                                      | 36 lub zgodnie z oczekiwaniami Zamawiającego | Od momentu wdrożenia utrzymanie systemu wg przyjętego zakresu i reguł SLA.  |
|          | <b>ETAP 2</b>                                   |  |   |
|          | Fazy etapu 2 [...]                              |  | Analogicznie jak dla etapu 1. Dla firm realizujących projekt danego etapu przygotowanie, implemetacja, wdrożenie wygląda podobnie, jak dla etapu 1.   |
|          | <b>ETAP 3</b>                                   |  |   |
|          | Fazy etapu 3 [...]                              |  | j.w.  |

Uwagi odnośnie etapu 2 i 3:

1. Należy wymagać od wykonawców etapu 2 i 3 dostosowania technologii do wybranej w etapie pierwszym.
2. Należy wymagać od wykonawców kolejnych etapów użycia tej samej chmury (o ile będzie to przewidziane) lub rozbudowy infrastruktury z etapu 1 (tu należy wymagać od wykonawcy etapu 1, aby infrastruktura lub chmura zapewniała możliwości rozbudowy).

3. Należy wymagać od wykonawców etapu 2 i 3 integracji z Help Deskem dostarczonym przez wykonawcę etapu 1.

#### 4. Budżet projektu w rozbiciu na:

##### a) ETAPY – prace deweloperskie,

1. Szczegółowe oszacowanie kosztów prac deweloperskich biorąc pod uwagę udostępnione informacje o planowanym systemie nie jest możliwe. Wg bardzo zgrubnej oceny prace deweloperskie powinny zamknąć się w przedziale od 2 do 4 mln PLN w każdym z etapów.
2. Proponuje się, aby płatności następowały po zakończeniu każdej fazy w etapie.
3. Nie zalecamy nakładania ograniczeń na kwotę za fazę typu "płatność za fazę X nie może być większa niż 50% kwoty za etap" - prowadzi to do problemów z rozłożeniem rezultatów prac przy wyborze różnych rozwiązań przez wykonawców.
4. Proponuje się, wydzielenie opcjonalnych prac rozwojowych. O momencie wykorzystania tych prac decydowałby Zamawiający (narzucenie ograniczenia, że mogą zostać użyte dopiero w fazie utrzymania spowoduje brak formalnej możliwości ich uruchomienia wcześniej, gdyby zaszła taka konieczność).
5. W zależności od źródła finansowania za utrzymanie systemu może być płatność kwartalna lub włączona w ogólną kwotę umowy.

##### b) chmura [sprzęt],

Infrastrukturę sprzętową na potrzeby niniejszego projektu identyfikować należy jako niezbędną moc obliczeniową z wymaganą pamięcią operacyjną oraz wymaganą przestrzenią na dane, która powinna zostać dostarczona w postaci **usługi** bezpośredniej (IaaS) lub pośredniej (PaaS, SaaS) zależnie od przyjętego przez Oferenta modelu/koncepcji rozwiązania, o ile inne wymagania Zamawiającego nie stanowią inaczej. Podejście takie pozwala Zamawiającemu skupić się na kryteriach ogólnych związanych z parametrami świadczonych usług oraz istotnych dla Zamawiającego wzorcach i standardach, zamiast na kreowaniu architektury rozwiązania w szczegółach. Z drugiej strony Oferent na podstawie przekazanych danych będzie mógł dokonać estymacji kosztów oferowanej infrastruktury.

Na potrzeby realizacji powyższego Zamawiający powinien zdefiniować:

1. Standardy architektoniczne i technologiczne wymagane/preferowane przez Zamawiającego
  - a. Wymagana liczba środowisk (PRD, PRE-PRD, TST, DEV, SZKO, ...)
  - b. Architektura mikrouslugowa
  - c. Lokalizacja (Unia Europejska, Polska, itp.)
  - d. inne
2. Parametry SLA
  - a. Redundancja
  - b. Wysoka Dostępność (należy zważyć wymagania w tym zakresie i jeżeli dopuszczalna jest niedostępność systemu, to SLA powinno ją odzwierciedlać)
  - c. Ciągłość działania
  - d. Bezpieczeństwo danych / backup
3. Graniczne dane ilościowe
  - a. 1-szy rok produkcyjnego działania systemu
    - i. Liczba wszystkich użytkowników systemu

- ii. Maksymalna liczba jednoczesnych użytkowników systemu w okresie 1 godziny
- iii. Maksymalna dzienna liczba dokumentów rejestrowanych w systemie
- iv. Maksymalna miesięczna liczba dokumentów rejestrowanych w systemie
- v. Maksymalna roczna liczba dokumentów rejestrowanych w systemie
- vi. Szacowany roczny wolumen danych wprowadzanych do systemu
- vii. Szacowany roczny wolumen plików/załączników wprowadzanych do systemu
- viii. Szacowana dzienna maksymalna wielkość transferu danych
- ix. Szacowana dzienna średnia wielkość transferu danych
- x. Wolumetria związana w procesami raportowania, analityki oraz algorytmów uczenia maszynowego
- xi. Inne
- b. 3-ci rok produkcyjnego działania systemu
  - i. jw. w punkcie 3.a
- c. 5-ty rok produkcyjnego działania systemu
  - i. jw. w punkcie 3.a

### c) chmura [oprogramowanie]

Stos oprogramowania/usług chmurowych musi w pełni zabezpieczać realizację każdego z etapów projektu. Dlatego Wybór docelowej chmury powinien uwzględniać możliwość wykorzystania następujących rozwiązań:

- Konteneryzacja
- Rozwiązania hybrydowe (pozwalająca łączyć rozwiązania oparte o chmurę jak i architekturę on-premise)
- Implementację rozwiązań workflow bezpośrednio na danym rozwiązaniu chmurowym
- Zarządzanie API
- Zarządzanie tożsamością

### d) chmura [systemy wspierające],

Chmura powinna zapewnić funkcjonalność szyny danych. Integracja z częścią rejestrów powinna zapewnić możliwość funkcjonowania części systemu w trybie offline, co za tym idzie zawartość części rejestrów powinna być synchronizowana okresowo w trybie pełnym lub przyrostowym. Architektura rozwiązania powinna przewidzieć możliwość przechowywania pełnych rejestrów aby przyspieszyć działanie poszczególnych procesów biznesowych. Dla celów optymalizacji procesów importów danych niezbędne może okazać się zastosowanie rozwiązań połączenia rozwiązań chmurowych z przestrzenią dyskową zlokalizowaną on-premise

#### Raportowanie i analityka

Mechanizmy raportowania mogą zostać użyte na poziomie danych szczegółowych oraz analityki statystycznej i wnioskowania bazującego na uczeniu maszynowym.

Dostęp do raportów i analiz powinien być kontrolowany zarówno na poziomie raportu oraz na poziomie danych, tzw. row level security. Zamawiający powinien określić oczekiwane rodzaje raportów na poziomie poszczególnych poziomów raportowania w podziale na poszczególne moduły. Część rozwiązań raportowych może być wykorzystywana w obrębie wszystkich 3 etapów. Wskazaniem było by wprowadzenie jak najbardziej jednolitych mechanizmów generujących poszczególne zestawienia/raporty.



Określenie typów wszystkich raportów, zakresu danych raportowych oraz mechanizmów raportowania na etapie SIWZ pozwoli wykonawcom zaproponować adekwatne rozwiązanie. Tutaj istotna uwaga – rozwiązanie raportowe zaproponowane w Etapie I może okazać się możliwe do wykorzystania we wszystkich etapach projektu, ale jednocześnie może podnieść istotnie cenę rozwiązania Etapu I. Tanie rozwiązanie na Etapie I może skutkować koniecznością wdrożenia kolejnego/odmiennego rozwiązania w kolejnych etapach. Podnosząc koszt całościowy projektu.

Określenie na etapie SIWZ zakresu raportowanych danych pozwoli lepiej zwymiarować objętość baz danych. W zależności od tego czy przechowywane muszą być wszystkie dane atomowe czy jedynie wybrane agregaty na przestrzeni lat użytkowania systemu, może istotnie wpłynąć na objętość baz danych.

## e) helpdesk

Szczegółowy opis usług Help Desku znajduje się w rozdziale 5 „Technologie” pkt d „helpdesk”.

## f) sugerowane podejście do szkoleń, szkolenia (materiały, instruktaże, instrukcje, słowniczki, FAQ, szkolenia),

W pierwszej kolejności powinno nastąpić przygotowanie i uzgodnienie z Zamawiającym planu szkoleń, obejmującego:

- odpowiedni podział godzin na poszczególne komórki organizacyjne,
- odpowiedni podział na grupy użytkowników,
- odpowiednie zakresy tematyczne,
- podział na odpowiednie etapy realizacji kolejnych modułów (lokalizację, formę szkoleń dla poszczególnych grup, daty, godziny, czas trwania).

W trakcie realizacji szkoleń wymagane jest prowadzenie rejestru wykonanych szkoleń przez wykonawcę. Sporządzanie protokołów po każdym szkoleniu. Zalecane jest również aby szkolenia kończyły się egzaminem (lub inną formą sprawdzenia wiedzy).

Forma szkolenia w zależności od roli użytkownika i stopnia skomplikowania zadań do wykonania w systemie:

- Szkolenia stanowiskowe dla administratorów technicznych, centralnych, lokalnych, itp. obejmujące m.in.
  - Procedury: wykonywania backupu, wykonywania innych cyklicznych czynności administracyjnych.
  - Instrukcje instalacji, konfiguracji, aktualizacji sprzętu i oprogramowania.
- Webinarium – szkolenia zdalnie z możliwością zadawania prowadzącemu pytań na czacie.
- Dla masowych użytkowników końcowych proponuje się szkolenia w formie eLearnigu w postaci przykładów przejścia poszczególnych procesów z zrzutami ekranów z aplikacji opatrzonymi komentarzem oraz ćwiczeń do samodzielnego przejścia krok po kroku w systemie.

Do ww. rodzajów szkoleń dołączane są materiały w postaci scenariuszy/ćwiczeń do samodzielnego przejścia krok po kroku w systemie.

Ponadto dla użytkowników powinny być dostępne:



- Pomoc kontekstowa w samym systemie [tooltips] - po najechaniu myszką nad np. Pole w systemie użytkownik otrzymuje w dymku informację o danym elemencie systemu.
- Utrzymywana i na bieżąco aktualizowana bazy wiedzy.
- Strona z najczęściej zadawanymi pytaniami [FAQ].

## g) inne nie uwzględnione powyżej (opisać jakie, istotne dla realizacji projektu).

1. Zarządzanie tożsamością
  - a. W zakresie zarządzania tożsamością użytkowników systemu, w szczególności w celu ich uwierzytelnienia oraz nadania odpowiednich ról, proponujemy wykorzystanie danych pochodzących z rejestrów zewnętrznych (np. rejestr lekarzy weterynarii). Użytkownik rejestrując konto w systemie będzie potwierdzał swoją tożsamość za pomocą systemu Login.gov.pl [Profil Zaufany]. Na podstawie potwierdzonych danych osobowych (numer PESEL), w systemach zewnętrznych wyszukane zostaną informacje nt. takiej osoby i na tej podstawie przydzielone zostaną odpowiednie role.
  - b. W przypadku braku rejestru zewnętrznego dla określonej klasy użytkowników systemu, uwierzytelnianie oraz nadanie ról w systemie może być realizowane za pomocą oświadczenia osoby (złożenie odpowiedniego dokumentu na etapie rejestracji konta w systemie) lub poprzez akcję użytkownika wewnętrznego systemu za pośrednictwem dedykowanego modułu oprogramowania stanowiącego konsolę pracownika.
  - c. Na etapie opracowywania wizji systemu zalecamy wykonanie analizy pod kątem istnienia rejestrów zewnętrznych dla planowanych grup użytkowników systemu. Na podstawie wyników takiej analizy możliwe będzie zdefiniowanie wymagań dotyczących np. wspomnianego powyżej modułu konsoli pracownika (zarządzanie rolami) oraz rozpoczęcie działań mających na celu podpisanie stosownych porozumień pomiędzy GIW a Gestorami rejestrów zewnętrznych w zakresie wymiany danych.
2. Migracja – jest możliwa do przeprowadzenia wyłącznie wtedy, jeśli są dostępne w systemach źródłowych dane wysokiej jakości (w jednolitym formacie, pełne, spójne, z małą liczbą błędów). Jeśli nie są spełnione powyższe wymagania dot. jakości danych, wtedy budowany system będzie zasilany danymi „od początku” (nie należy dopuścić do zasilania nowobudowanego systemu danymi śmieciowymi).
3. Pilotaże - doświadczenie pokazuje, że wdrożenie zwłaszcza systemu, który ma być użytkowany masowo, warto na początek uruchomić dla małej liczby, wybranych użytkowników (pilotaż). Pilotaż pozwoli sprawdzić przyjęte założenia do systemu na małej liczbie użytkowników, zebrać uwagi, które pozwolą nieznacznie skorygować system przed wdrożeniem masowym.
4. Wymagania w OPZ - ważne jest zapewnienie możliwości kontroli wymagań na etapie realizacji projektu (możliwość sprawdzenia, czy wszystkie wymagania są spełnione oraz możliwość powiązania wymagań z artefaktami powstającymi w projekcie np. z przypadkami użycia, diagramami klas, scenariuszami testowymi, itd.). W konsekwencji pozwoli to na stwierdzenie na etapie odbioru systemu, czy zostały spełnione wszystkie przewidziane w OPZ wymagania. Aby to osiągnąć należy w OPZ:
  - a. zapewnić opis jednego wymagania w jednym punkcie,
  - b. każde wymaganie ponumerować
  - c. najłatwiej zarządzić wymaganiami, jeśli będą w formie tabelarycznej
  - d. powyższe dotyczy to zarówno wymagań funkcjonalnych, jak i niefunkcjonalnych.

- e. generalnie OPZ powinien zawierać:
  - i. akapity z opisami i wyjaśnieniami oraz
  - ii. numerowane wymagania (na etapie projektu artefakty/produkty projektowe będą wiązane z ww. numerami, odbiory będą oparte o te numery).
- 5. Metodyki projektowe / dokumentacja analityczna / projektowa  
Proponowany przez nas model prowadzenia projektu (hybryda metodyki kaskadowej i zwinnej) wymaga dostosowania do niego trybu pracy nad dokumentacją projektowo-analityczną. W trakcie dwóch pierwszych faz powinna zostać wytworzona pełna wymagana dokumentacja (zarówno analityczna, jak i techniczna), która to będzie stanowić bazę (szkielet) dla późniejszych prac rozwojowych. Oczywiście wszelkie dokumenty będą uszczegóławiane w ramach prac iteracyjno-przyrostowych w dalszych fazach – ważne jest, aby od samego początku nakreślić docelową strukturę i zawartość poszczególnych dokumentów, co pozwoli na wyznaczenie właściwego kierunku i priorytetyzacji prac. Istotne jest również, aby projekt był prowadzony z użyciem metodyk projektowych ze sformalizowaną dokumentacją (np. UML). Wszelkie wymagane diagramy, modele itp. powinny być tworzone i przechowywane z wykorzystaniem odpowiedniego oprogramowania (np. Enterprise Architect) i powinny stanowić integralną część wersjonowanego repozytorium projektowego (wraz z całą dokumentacją).
- 6. Udostępnianie materiały dla użytkowników końcowych
  - a. Instrukcje użytkownika (jeden rodzaj, obecnie nikt nie czyta instrukcji, najlepszym rodzajem dokumentacji są tooltip-y (wbudowane podpowiedzi w aplikację) oraz FAQ.
  - b. Dokumentacja API – powinna zawierać przykłady wywołań API oraz projekty testów do wykorzystania przez programistów . W zależności od rodzaju API dokumentacja powinna składać się z kodu samodokumentującego (np. dla API wg standardów SOAP mogą to być WSDL-e).
  - c. Nie zalecamy wymagania przygotowania materiałów w postaci filmów, instrukcji interaktywnych itp. – są one bardzo kosztowne w przygotowaniu i utrzymaniu ich aktualności, a użytkownicy korzystają z takich materiałów sporadycznie lub wcale.

## 5. Technologia

### a) (proponowana chmura obliczeniowa)

Ponieważ Zamawiający sam nie wskazuje preferencji w zakresie wyboru dostawcy usług chmurowych, proponujemy dokonanie wyboru kierując się możliwościami dostarczenia następujących obszarów usług:

1. Wirtualne zasoby obliczeniowe
2. Konteneryzacja
3. Zarządzanie tożsamością
4. DevOps
5. Zarządzanie aplikacjami w tym aplikacjami mobilnymi
6. Monitoring i zarządzanie

Ponadto ocenie powinna podlegać dostępność predefiniowanych architektur referencyjnych oraz ich weryfikowalna gotowość do szybkiego wdrożenia w zakresie potrzeb niniejszego projektu.

### b) system operacyjny, baza danych dla rozwiązania

Ze względów ekonomicznych oraz ewentualnych alternatyw migracji rozwiązań pomiędzy dostawcami chmury, rekomendujemy stosowanie oprogramowania licencjonowanego w modelu Open Source i rekomendujemy:

- Dla systemów operacyjnych, zastosowanie RHEL, Debian lub pochodne edycje
- Dla baz danych - PostgreSQL lub MySQL

### c) systemy wspierające – monitoring, zarządzanie,

Systemy wspierające muszą być dostosowane do wybranej architektury chmurowej, może to być Zabbix + ELK stack, natywne Azure Monitor for containers, Prometheus lub inne. Zarządzanie w oparciu o Terraform (IaC) i CI/CD (np. gitlab). Na tym etapie nie należy określać konkretnych narzędzi wspierających gdyż każdy z dostawców może zaoferować adekwatny własny zestaw narzędzi. Istotne elementy jakie powinny zostać zaadresowane:

- rozwiązania nie powinny być zamknięte, aby kolejny wykonawca mógł kontynuować dalszy rozwój z wykorzystaniem danego rozwiązania bez ponoszenia kosztów lub rozwiązanie zostanie przekazane zamawiającemu z możliwością udostępniania go kolejnemu wykonawcy
- narzędzia powinny pozwolić na wizualizowanie (dla zespołu nadzorującego) stanu poszczególnych komponentów (w postaci dashboardów), składowych systemu w czasie zbliżonym do czasu rzeczywistego z dokładnością np. do minuty. Konieczne jest oczywiście wysyłanie powiadomień (email, wywołanie API) przy przekroczeniu konfigurowalnych parametrów działania poszczególnych elementów systemu (w tym interfejsów)
- Zamawiający powinien mieć wgląd w wynik działania narzędzi - ciągłej integracji (CI) - ciągłego dostarczania (CD) na środowiskach Zamawiającego
- Wykonawca powinien dostarczyć wizję mechanizmów wersjonowania całego rozwiązania aby możliwe było ocenienie czy mechanizmy wersjonowania są przemyślane i mają szansę być wykorzystywane w ramach całego systemu w sposób jednolity, na przestrzeni wszystkich etapów realizacji projektów łącznie z podziałem na rozwój oraz usuwanie ewentualnych błędów.
- Zamawiający powinien otrzymać dostęp do repozytorium kodu wykonawcy. Repozytorium kodu powinno wspierać mechanizmy tagowania oraz tworzenie gałęzi kodu na potrzeby rozwoju i usuwania błędów.

### d) helpdesk

#### 1. Zakres usługi Helpdesk obejmuje:

- Przyjmowanie od użytkowników zgłoszeń dotyczących wspieranej aplikacji.
- Rejestracja zgłoszeń w aplikacji ITSM.
- Zdalne wsparcie użytkowników – telefoniczne rozwiązywanie problemów oraz udzielanie telefonicznych konsultacji w zakresie wspieranej aplikacji (po przeszkoleniu konsultantów Help Desk Euvic IT przez dostawcę aplikacji, przekazaniu bazy wiedzy oraz instrukcji i procedur postępowania).
- Przekazywanie nierozwiązanych zgłoszeń do kolejnych linii wsparcia (2 / 3 linia wsparcia po stronie dostawcy aplikacji).
- Zamykanie procesu realizacji zgłoszeń.
- Raportowanie liczby zarejestrowanych i obsłużonych zgłoszeń.
- Udostępnienie narzędzi:
  - Aplikacja ITSM.
  - System telekomunikacyjny z IVR.
  - Zdalna konsola.

#### Okno dostępności usługi:

| Okno dostępności  |
|---|
| Dni robocze 8:00 – 20:00<br>Możliwość przekazywania zgłoszeń przez formatkę www, e-mail: 24/7/365 |

#### Parametry SLA / KPI:

| Kryterium   | Wskaźnik   |
|---|--|
| Średni czas oczekiwania na połączenie   | 60 sekund  |
| Maksymalny % nieodebranych telefonów po 60 sek.   | 20%  |
| Czas reakcji  | Do 60 minut  |
| Czas realizacji   | SLA w tym czasy realizacji proponowane dla Państwa projektu zawarte w rozdziale 11. Umowa utrzymaniowa |
| Terminowość realizacji zgłoszeń na 1 linii wsparcia Help Desk   | >90%   |
| Rozwiązywalność zgłoszeń na 1 linii wsparcia Help Desk (dotyczy zgłoszeń możliwych do rozwiązania zdalnie przez 1 linię wsparcia) | >50%   |

## 2. Umowa SLA

Umowa SLA (Service Level Agreement) jest narzędziem do zarządzania relacjami pomiędzy firmą Euvic jako dostawcą usług a Klientem. W szczególności umowa ma na celu określenie zasad i zakresu pomiaru jakości usług świadczonych przez Euvic na rzecz Klienta oraz wpływu jakości świadczeń na cenę usług.

#### Cele wdrożenia umowy SLA:

- Zdefiniowanie parametrów, służących jako miary jakości dla poszczególnych usług.
- Analiza wartości mierzonych parametrów i porównywanie z wartościami modelowymi lub przyjętymi jako punkt odniesienia.
- Ustalenie sposobu obniżania wynagrodzenia tzn. relacji pomiędzy ceną za świadczone usługi a odchyleniami od przyjętych modeli jakości.
- Ustalenie kluczowych usług i aspektów związanych z obsługą środowiska IT Klienta, na podstawie analizy uzyskiwanych rezultatów.
- Zdefiniowanie procesów lub mechanizmów związanych z obsługą środowiska IT wymagających dopracowania lub rozwinięcia.

## 3. Raportowanie

Pomiary jakości świadczonych usług opierać się będą na raportach przygotowywanych przez Euvic. Szczegóły dotyczące raportowania, w tym zakres i częstotliwość zostaną uzgodnione z Klientem w etapie przygotowawczym, poprzedzającym uruchomienia usługi.

Dane na potrzeby raportowania będą pochodziły z aplikacji ITSM oraz systemu telekomunikacyjnego.

Przykładowy zakres raportu może obejmować:

- Informacje nt. zgłoszeń typu incydent.
- Informacje nt. zgłoszeń typu wniosek o usługę.
- Informacje nt. zgłoszeń typu wniosek o zmianę.

- Raport z obsługi zgłoszeń:
  - Ilość zgłoszeń zarejestrowanych.
  - Ilość zgłoszeń zamkniętych.
  - Podział zgłoszeń w zależności od źródła incydentu.
  - Podział zgłoszeń w zależności od zasobów realizujących (1 / 2 / 3 linia wsparcia).
  - Czasy obsługi incydentów.
- Wskaźniki telefoniczne:
  - Średni czas odbierania telefonu.
  - Maksymalny czas oczekiwania na połączenie.
  - Średni czas rozmowy z konsultantem Help Desk.
- Raport TOP 10:
  - Lista 10 najczęściej zgłaszających użytkowników.
  - Lista 10 najczęściej występujących incydentów/problemów.

Szczegółowa analiza dostarczanych raportów umożliwia wyciąganie konkretnych wniosków i formułowanie właściwych rekomendacji.

#### 4. Badanie satysfakcji użytkowników

Badanie poziomu zadowolenia użytkowników z usług świadczonych przez Euvic będzie odbywać w uzgodnionej z Klientem częstotliwości i formie np.

- Ankieta wypełniana przez użytkownika w systemie ITSM każdorazowo po zakończeniu usługi.
- Okresowe analizy zadowolenia użytkownika w formie anonimowej ankiety skierowanej do użytkowników.

#### e) aplikacje mobilne,

Realizując aplikację mobilną należy zadbać o dystrybucję jej na dwie najpopularniejsze platformy Android oraz iOS. Efektywne pisanie, utrzymanie i rozwój aplikacji na obie platformy równolegle sugeruje użycie technologii zapewniającej możliwość współdzielenia kodu pomiędzy wersjami dla obu środowisk. Obecnie istnieje na rynku wiele sprawdzonych technologii pozwalających na takie podejście do realizacji aplikacji mobilnych, m.in.: React Native, Flutter lub Xamarin. Pozwalają one na faktyczne zmniejszenie kosztów napisania i dalszego utrzymania aplikacji oraz zapewniają spójność aplikacji pomiędzy wersjami dedykowanymi dla różnych systemów.

Warto w ramach realizacji przetargu ograniczyć zapewnienie kompatybilności wstecz (np. systemy nie starsze niż 4 lata, co na chwilę obecną zawiera urządzenia z systemami iOS – wersja 10+ lub Android 8.0). Poszerzanie kompatybilności do starszych wersji urządzeń i systemów, powoduje znaczne ograniczenie w doborze technologii wykonania (zarówno natywnego API, ale również technologii cross-platformowych) i wprowadzanie kolejnych ograniczeń w wykorzystaniu nowoczesnych sposobów interakcji z użytkownikiem.

#### f) jakie rozwiązania off-line,

Jako rozwiązanie do pracy off-line proponujemy aplikację mobilną instalowaną na urządzeniu mobilnym (smartfonie lub tablecie) pracownika. Aplikacja udostępniałaby 2 tryby pracy: off-line i on-line. W ramach trybu off-line wszystkie dane wprowadzone przez użytkownika zapisywane byłyby lokalnie w pamięci urządzenia. W chwili uzyskania dostępu do Internetu, użytkownik przełączałby aplikację w tryb on-line, w którym udostępniona byłaby funkcja wysłania zgromadzonych lokalnie danych do komponentu centralnego systemu.

### g) jakie rozwiązania zapewniające integralność i zabezpieczenia danych po przejściu na tryb on-line i upload danych do systemu?

Standardowe rozwiązania spójne z pozostałymi mechanizmami systemu - elementem synchronizacji danych po uzyskaniu dostępu do Internetu będą zarówno walidacje biznesowe, jak i kontrola dostępu do usług zasilających centralne rejestry (uwierzytelnienie, autoryzacja). Ponadto w przypadku wykrycia sytuacji w której w systemie centralnym znajdują się nowsze dane zaimplementowany zostanie mechanizm umożliwiający porównanie różnic i manualne ich scalenie. Dane wrażliwe będą przechowywane w aplikacji w postaci zaszyfrowanej.

### h) rozwiązania uwierzytelniające użytkowników, wprowadzanie danych

W zakresie mechanizmów uwierzytelniających i autoryzujących użytkowników proponujemy standardowe rozwiązania - OAuth 2.0 lub OpenID Connect (w zależności od specyficznych potrzeb). Natomiast jeśli chodzi o wprowadzanie danych - tutaj również proponujemy standardowe formatki i formularze.

## 6. Kryteria oceny oferty – propozycje, wraz z podaniem wag tych kryteriów i sposobu oceny ofert

1. Wymagania dotyczące personelu Wykonawcy – wyszczególnione w punkcie 2 a – Zespół projektowy – po stronie Wykonawcy
2. Wyżej punktowane rozwiązania oparte na Open Source.
3. Premiowana wyższa liczba godzin rozwojowych deklarowana przez wykonawcę w ramach ceny oferty
4. Premiowana niższa stawka za godzinę prac opcjonalnych
5. Referencje dot. wykonanych projektów (wymagana liczba projektów, o wymaganej minimalnej kwocie)
6. Ocena kompetencji osób
7. Koncepcja rozwiązania np.:
  - a. w obszarze bezpieczeństwa
  - b. projekt interfejsu użytkownika
  - c. rozwiązania bazodanowe
  - d. zaprojektowanie rozwiązania wybranego problemu

## 7. Proponowane zapisy umowy (na co zwrócić uwagę).

Wytyczne dotyczące niektórych zapisów do umowy:

- Możliwość udzielenia zabezpieczenia/gwarancji należytego wykonania umowy
- Warunki świadczenia usług utrzymaniowych
- Elastyczność przy konieczności wymiany członków zespołu
- Ograniczenie wysokości kar oraz próg uprawniający do wypowiedzenia umowy
- Uwzględnienie zasad współpracy związanych z pandemią (zdalne spotkania, dokumenty wyłącznie w formie elektronicznej, szkolenia w formie eLearningu, itp.)



## 8. Zespół do utrzymania systemu – po stronie Zamawiającego (skład takiego zespołu).

Skład zespołu utrzymania po stronie Zamawiającego jest uzależniony od modelu utrzymania który zostanie przyjęty przez Zamawiającego. Poniżej przedstawiono kilka możliwości:

1. Jeśli infrastruktura sprzętowo-systemowa będzie w zasobach własnych Zamawiającego – konieczny jest zespół IT. W skład takiego zespołu musi wejść kilku administratorów, dostępnych w trybie 24/7, czy to na miejscu, czy w formie dyżurów zdalnych.
2. Jeśli administracja merytoryczna będzie po stronie Zamawiającego, to Zamawiający musi zatrudniać osoby merytoryczne, które znają system, są w stanie wykonywać czynności administracyjne w systemie (np. zakładać, usuwać konta użytkowników, nadawać uprawnienia, itp.). Osoby te będą stanowiły tzw. II linię Help Desku.
3. Jeśli Zamawiający nie będzie korzystał z Help Desków zewnętrznych, to będzie musiał zapewnić sobie osoby, które będą obsługiwały I linię Help Desku, w szczególności umiały odpowiadać na pytania użytkowników dot. systemu, umiały korzystać z bazy wiedzy, itd. Dodatkowo dochodzi baza narzędziowa Help Desku (aplikacje do obsługi zgłoszeń, baza wiedzy, dedykowana poczta elektroniczna, dedykowana linia telefoniczna, itd.)
4. Oprócz powyższego zawsze w każdym przypadku będą potrzebne osoby (te same role jak w projekcie):
  - a. Główny Product Owner – osoba która zna całość systemu, nadzoruje utrzymanie całości systemu
  - b. Product Ownerzy – zgłaszają potrzeby rozwojowe systemu, tworzą wymagania na zmiany, odbierają zmiany w systemie.
  - c. Osoby wspierające z działu IT – osoby monitorujące działanie systemu (nawet jeśli jest to wyłącznie nadzór na firmą zewnętrzną dostarczającą chmurę)
  - d. Testerzy – osoby sprawdzające poprawność aktualizacji (dostarczonych poprawek błędów oraz zmian).

## 9. Zespół po stronie Wykonawcy – I linia wsparcia/rozwój.

Opis I i II linii wsparcia zawarty w opisie systemu Help Desk (rozdz. 5. „Technologia” pkt. d „helpdesk”).

## 10. Zespół po stronie Zamawiającego – II linia wsparcia/rozwój.

Opis I i II linii wsparcia zawarty w opisie systemu Help Desk (rozdz. 5. „Technologia” pkt. d „helpdesk”).

## 11. Umowa utrzymaniowa - propozycje/minimalny okres.

IPU w zakresie utrzymania powinno obejmować:

1. Usuwanie wad systemu. W praktyce sprawdzają się trzy priorytety wad: awaria, wysoki i niski. Dla podanych priorytetów powinny być wymagane czasy reakcji na zgłoszenie (czas od zgłoszenia do powzięcia działań naprawczych przez Wykonawcę) oraz czas usuwania wady. Dla wad typu awaria ważne jest wprowadzenie pojęcia rozwiązania obejściowego (w przypadku niemożności usunięcia wady w czasie przewidzianym na ich usunięcie dopuszczalne byłoby rozwiązanie tymczasowe przywracające sprawność systemu). Ważne jest również dopuszczenie możliwości zmiany priorytetów zgłoszeń wad za obopólnym uzgodnieniem Zamawiającego i Wykonawcy. Jeśli system ma być dostępny w 24h na



dobę / 7 dni w tygodniu, ważne jest aby wymagać obsługi awarii w ww. trybie [tylko awarii, wady niższych priorytetów powinny być obsługiwane wyłącznie w dni robocze, w godzinach roboczych]. Proponowane SLA:

| Priorytet wady | Czas reakcji    | Czas rozwiązania   | Informacje dodatkowe  |
|----------------|-----------------|--|---|
| Awaria         | 2h              | 8h, jeśli niemożliwe, dopuszczalne jest rozwiązanie obejściowe | Obsługa 24/7. W przypadku zastosowania obejścia, priorytet wady zmienia się na Wysoki |
| Wysoki         | 4h              | 3 dni robocze  | Obsługa w dni robocze w godzinach pracy (9:00 – 17:00)                                |
| Niski          | 1 dzień roboczy | W najbliższej planowej aktualizacji                            |   |

- Konsultacje powinny obejmować odpowiedzi na pojawiające się pytania Zamawiającego dotyczące działania systemu.
- Zamówienia rozwojowe – powinna zostać przewidziana pula godzin na rozwój systemu w okresie utrzymania. Pula powinna być opcjonalna, z określoną maksymalną liczbą godzin w ciągu roku (pule godzin w okresach, aby uniknąć ich kumulacji na koniec okresu utrzymania). Dla ww. godzin na etapie postępowania powinna zostać określona stawka godzinowa za ww. prace.
- Inne prace, w zależności od potrzeb Zamawiającego:
  - przeglądy systemu w zakresie konfiguracji, bezpieczeństwa, itp. w podanej liczbie w ciągu roku [zwykle 2 lub 1 rocznie]
  - prace administracyjne, konfiguracyjne, monitorowania działania systemów, itp.
  - szkolenia dla użytkowników końcowych.

## 12. Etapy realizacji projektu.

Opisane w odpowiedzi na punkt 4 a Budżet projektu - Etapy

## 13. Proces testowania i wdrożenia systemu.

Testy oprogramowania powinny być przeprowadzane w ramach sprintów (dwutygodniowych okresów prac przy przyjęciu metodyki scrum w projekcie). Do ich przeprowadzenia przygotowywane byłyby scenariusze testowe - dla każdego wymagania co najmniej jeden pozytywny i jeden negatywny. Na ich podstawie byłyby przeprowadzane testy wewnętrzne. Po pomyślnym zakończeniu testów wewnętrznych, scenariusze byłyby przekazywane do Zamawiającego, który przeprowadzałby testy zrealizowanej w sprincie części oprogramowania (bazując na scenariuszach). Po zakończeniu budowy modułu (lub modułów) następowałyby testy akceptacyjne przeprowadzane przez Zamawiającego polegające na wykonaniu zarówno scenariuszy testowych, jak i przeprowadzeniu własnych testów opartych na wiedzy użytkowników o wykonywanych czynnościach biznesowych.

**Wytyczne do testów:**

- Testy powinny być przeprowadzane w najpopularniejszych przeglądarkach, np. próg 5% udziału w rynku przeglądarek - chrome, firefox, safari, edge (OPZ powinien zawierać wyszczególnienie, w których przeglądarkach powinien system działać).
- Testy powinny obejmować sprawdzenie zgodności ze standardami WCAG 2.1.
- Dla kluczowych i powtarzalnych procesów biznesowych wymagane w OPZ powinno być dostarczenie testów automatycznych.
- Po wgraniu wersji na środowisko testowe przeprowadzane są testy akceptacyjne - funkcjonalne i integracyjne. Testy przeprowadzane przez Zamawiającego ze wsparciem Wykonawcy, zakończone raportem z testów akceptacyjnych.
- Testy powinny być wspierane organizacyjnie przez narzędzia (np. JIRA, jeśli Wykonawca nie dysponuje takimi narzędziami, to powinny być wymagane od Wykonawcy). Narzędzia powinny umożliwiać: zarządzanie testami (podział testów na cykle testowe, przydział scenariuszy do cyklu, przydział scenariuszy do testerów, raportowanie stanu testów wg scenariuszy, zgłaszanie błędów wg przyjętych kategoryzacji błędów, w tym powiązanie ich ze scenariuszami, których dotyczą, raportowanie statusów zgłoszeń).

## 14. Rozbudowa systemu:

### a) koszty/funkcjonalne,

Proponujemy, aby po zakończeniu etapu budowy i wdrożenia inicjalnej wersji systemu, rozbudowa i wszelkie modyfikacje odbywały się w ramach ustalonej puli roboczogodzin, której wielkość Zamawiający zdefiniuje w SIWZ. SIWZ zawierałby definicję procedury realizacji usługi rozwoju np.:

- zlecenie zmiany przez Zamawiającego,
- przygotowanie projektu przez Wykonawcę, w tym określenie terminu realizacji oraz szacunku pracochłonności,
- akceptacja projektu przez Zamawiającego.

Dodatkowo, w przypadku wykorzystania w systemie usług chmurowych, oprócz puli roboczogodzin powinna zostać zdefiniowana dodatkowa kwota, z której pokryte byłyby ewentualne dodatkowe koszty utrzymania środowiska chmurowego na skutek zastosowania w zadaniach rozbudowy systemu dodatkowych usług chmurowych.

### b) koszty/techniczne – serwery, macierze itp.

W referencji do treści rozdziałów poprzednich, a szczególnie rozdziału chmura (sprzęt), bez wymaganych danych ilościowych oszacowanie kosztów sprzętu jest na tym etapie niemożliwe. Ponadto przyjmując, że preferowaną infrastrukturą jest infrastruktura chmurowa, szacowanie sprzętu jest niezasadne. Oczywiście przy zmianie strategii z chmurowej na on-premise oraz zdefiniowaniu wymaganych danych ilościowych, oszacowanie kosztów infrastruktury sprzętowej jest możliwe i zostanie przeprowadzone.

## 15. Obsługa helpdesk – propozycja rozliczenia.

Poniżej aktualny cennik (jako przykład) stosowanych przez EUVIC IT usług Help Desk wraz z opisem rozliczania.

| Liczba zgłoszeń  |                                   | PLN netto         |
|--|-----------------------------------|-------------------|
| Opłata miesięczna za gotowość i obsługę zgłoszeń         | Opcja 1: do 1 000 zgłoszeń / m-c  | 24 200 PLN / m-c  |
|  | Opcja 2: do 2 500 zgłoszeń / m-c  | 53 400 PLN / m-c  |
|  | Opcja 3: do 5 000 zgłoszeń / m-c  | 99 500 PLN / m-c  |
|  | Opcja 4: do 10 000 zgłoszeń / m-c | 177 200 PLN / m-c |
|  | Opcja 5: do 20 000 zgłoszeń / m-c | 344 300 PLN / m-c |
| Pakiet zawierający 10 zgłoszeń (po przekroczeniu limitu) |                                   | 200,00 PLN        |

Ryczałt miesięczny obejmuje:

- Realizację usług opisanych w ofercie zgodnie z parametrami SLA/KPI.
- Utrzymanie w gotowości zespołu konsultantów Help Desk.
- Udostępnienie i utrzymanie narzędzi: aplikacji ITSM, systemu telekomunikacyjnego oraz zdalnej konsoli.
- Zarządzanie zespołem poprzez Lidera Zespołu Help Desk ora oddelegowanie Kierownika Projektu po stronie Euvic IT odpowiedzialnego za nadzór operacyjny nad realizowanymi usługami.

## 16. Zabezpieczenia systemu.

W ramach cykliów wytwórczych oprogramowania należy uwzględnić utwardzanie systemu oraz infrastruktury. Proponujemy również uwzględnić audyty bezpieczeństwa w każdym etapie harmonogramu. Należy mieć na uwadze, że implementacja uwag z raportu bezpieczeństwa (będącego wynikiem audytu) może wpłynąć na czas zakończenia poszczególnych etapów (szczególnie w początkowych fazach).

## 17. Architektura rozwiązania docelowego.

System powinien mieć modułową architekturę, co umożliwi jego rozwój w dowolnym kierunku w przyszłości. Ogólny zarys modułów biznesowych powinien zostać przedstawiony w OPZ, ponadto do każdego modułu powinny zostać przypisane konkretne wymagania (zarówno funkcjonalne jak i нефункционалне) umożliwiające weryfikację zgodności oprogramowania z potrzebami oraz wycenę kosztów wytwórczych.

Architektura powinna zostać dostosowana do wymagań systemu (szczególnie wydajnościowych oraz bezpieczeństwa) i umożliwiać reakcję na potencjalne problemy (np. bezpieczeństwa) lub okresowe/stałe zwiększenie obciążenia aplikacji. Warto unikać niszowych rozwiązań i decydować się

Euvic Solutions – Odpowiedź 1.0 - Wniosek o dopuszczenie do udziału we Wstępnych Konsultacjach Rynkowych

na popularne, sprawdzone już rozwiązania (co pozwoli też rozszerzyć grupę podmiotów będących w stanie utrzymywać i rozwijać system w przyszłości.)

Na etapie wykonawstwa architektura systemu powinna zostać opisana (zamodelowana) w spójny sposób z wykorzystaniem obowiązujących standardów i narzędzi [UML, C4 model].

## 18. Czy Wykonawca sugeruje użycie silnika workflow do realizacji rozwiązania

Silniki workflow sprawdzają się w sytuacjach, gdy procesy biznesowe są mocno zmienne w czasie i wymagają mechanizmów wersjonowania, elastyczności w zmianach przepływów itp. W tym przypadku procesy są dokładnie zdefiniowane, więc nie zachodzi potrzeba wykorzystania tego typu rozwiązań, które niepotrzebnie podrażałyby rozwiązanie.

---

Z wyrazami szacunku

Maciej Tomczak  
Dyrektor Handlowy  
Euvic Solutions S.A.

[dokument został podpisany podpisem elektronicznym kwalifikowanym]