

WYKONAWCY

ubiegający się o zamówienie publiczne

**POWIADOMIENIE
o zmianach SWZ**

Dotyczy: postępowania o udzielenie zamówienia publicznego, prowadzonego w trybie przetargu nieograniczonego na „**Modernizację infrastruktury teleinformatycznej oraz modernizację i wdrożenie e-usług medycznych realizowaną w ramach projektu pn. „Rozwój e-usług medycznych w Samodzielnym Publicznym Zespole Zakładów Opieki Zdrowotnej w Nisku – znak sprawy Z.II.260.037.Zp.2022**”

Zamawiający, **Samodzielny Publiczny Zespół Zakładów Opieki Zdrowotnej**, działając na podstawie art. 137 ust. 1 i 2 ustawy z dnia 11 września 2019 r. Prawo zamówień publicznych (t.j. Dz.U. z 2021 r. poz. 1129 z późn. zm.), informuje o dokonaniu zmian w zapisach Specyfikacji warunków zamówienia w następującym zakresie:

- 1. W Załączniku nr 1 (Szczegółowy Opis Przedmiotu Zamówienia) w Zadaniu nr 1 - Oprogramowanie do serwerów bazodanowych – 1 kpl. (4 licencje) w pkt. 4 „Oprogramowanie serwerów bazodanowych” było:**

4.	Oprogramowanie serwerów bazodanowych	Oferowany motor bazy danych musi być dostępny zarówno na platformy systemów operacyjnych Windows i Linux.		TAK / NIE
		Oferowany motor bazy danych musi mieć możliwość rozbudowy do wersji wspierającej możliwość synchronicznej replikacji danych w dwóch niezależnych centrach danych.		TAK / NIE
		Oferowany motor bazy danych posiada komercyjne wsparcie producenta. Nie dopuszcza się zastosowania RBD typu open-source.		TAK / NIE
		Oferowany motor bazy danych ma możliwość realizacji kopii bezpieczeństwa w trakcie działania (na gorąco).		TAK / NIE
		Oferowany Motor bazy danych generuje kopie bezpieczeństwa automatycznie (o określonej porze) i na żądanie operatora oraz umożliwia odtwarzanie bazy danych z kopii archiwalnej, w tym sprzed awarii.		TAK / NIE
		Oferowany motor bazy danych umożliwia eksport i import danych z bazy danych w formacie tekstowym z uwzględnieniem polskiego standardu znaków.		TAK / NIE
		Administrator posiada możliwość wyboru danych, które mają być monitorowane w logach systemu z dokładnością do poszczególnych kolumn w tabelach danych, a zarządzanie nimi może odbywać się z poziomu narzędzi do zarządzania bazami danych (dopuszcza się narzędzie na poziomie motoru bazy danych).		TAK / NIE

	Oferowany motor bazy danych posiada możliwość rozbudowy do wersji wspierającej możliwość synchronicznej replikacji danych w dwóch niezależnych centrach danych.		TAK / NIE
	Oferowany motor bazy danych posiada komercyjne wsparcie producenta. Nie dopuszcza się zastosowania RBD typu open-source.		TAK / NIE
	Niezależność platformy systemowej dla oprogramowania klienckiego / serwera aplikacyjnego od platformy systemowej bazy danych.		TAK / NIE
	Możliwość przeniesienia (migracji) struktur bazy danych i danych pomiędzy ww. platformami bez konieczności rekompilacji aplikacji bądź migracji środowiska aplikacyjnego.		TAK / NIE
	Przetwarzanie z zachowaniem spójności i maksymalnego możliwego stopnia współbieżności. Modyfikowanie wierszy nie może blokować ich odczytu, z kolei odczyt wierszy nie może ich blokować do celów modyfikacji. Jednocześnie spójność odczytu musi gwarantować uzyskanie rezultatów zapytań odzwierciedlających stan danych z chwili jego rozpoczęcia, niezależnie od modyfikacji przeglądanego zbioru danych.		TAK / NIE
	Możliwość zagnieżdżania transakcji – powinna istnieć możliwość uruchomienia niezależnej transakcji wewnątrz transakcji nadrzędnej. Przykładowo – powinien być możliwy następujący scenariusz: każda próba modyfikacji tabeli X powinna w wiarygodny sposób odłożyć ślad w tabeli dziennika operacji, niezależnie czy zmiana tabeli X została zatwierdzona czy wycofana.		TAK / NIE
	Wsparcie dla wielu ustawień narodowych i wielu zestawów znaków (włącznie z Unicode).		TAK / NIE
	Możliwość migracji zestawu znaków bazy danych do Unicode.		TAK / NIE
	Możliwość redefiniowania przez klienta ustawień narodowych – symboli walut, formatu dat, porządku sortowania znaków za pomocą narzędzi graficznych.		TAK / NIE
	Skalowanie rozwiązań opartych o architekturę trójwarstwową: możliwość uruchomienia wielu sesji bazy danych przy wykorzystaniu jednego połączenia z serwera aplikacyjnego do serwera bazy danych.		TAK / NIE
	Możliwość otworzenia wielu aktywnych zbiorów rezultatów (zapytań, instrukcji DML) w jednej sesji bazy danych.		TAK / NIE
	Wsparcie protokołu XA.		TAK / NIE
	Wsparcie standardu JDBC 3.0.		TAK / NIE

		Zgodność ze standardem ANSI/ISO SQL 2003 lub nowszym.		TAK / NIE
		Motor bazy danych powinien umożliwiać wskazywanie optymalizatorowi SQL preferowanych metod optymalizacji na poziomie konfiguracji parametrów pracy serwera bazy danych oraz dla wybranych zapytań. Powinna istnieć możliwość umieszczania wskazówek dla optymalizatora w wybranych instrukcjach SQL.		TAK / NIE
		Brak formalnych ograniczeń na liczbę tabel i indeksów w bazie danych oraz na ich rozmiar (liczbę wierszy).		TAK / NIE
		Wsparcie dla procedur i funkcji składowanych w bazie danych. Język programowania powinien być językiem proceduralnym, blokowym (umożliwiającym deklarowanie zmiennych wewnątrz bloku), oraz wspierającym obsługę wyjątków. W przypadku, gdy wyjątek nie ma zadeklarowanej obsługi wewnątrz bloku, w razie jego wystąpienia wyjątek powinien być automatycznie propagowany do bloku nadrzędnego bądź wywołującej go jednostki programu.		TAK / NIE
		Procedury i funkcje składowane powinny mieć możliwość parametryzowania za pomocą parametrów prostych jak i parametrów o typach złożonych, definiowanych przez użytkownika. Funkcje powinny mieć możliwość zwracania rezultatów jako zbioru danych, możliwego do wykorzystania jako źródło danych w instrukcjach SQL (czyli występujących we frazie FROM). Ww. jednostki programowe powinny umożliwiać wywoływanie instrukcji SQL (zapytania, instrukcje DML, DDL), umożliwiać jednoczesne otwarcie wielu tzw. kursorów pobierających paczki danych (wiele wierszy za jednym pobraniem) oraz wspierać mechanizmy transakcyjne (np. zatwierdzanie bądź wycofanie transakcji wewnątrz procedury).		TAK / NIE
		Możliwość kompilacji procedur składowanych w bazie do postaci kodu binarnego (biblioteki dzielonej).		TAK / NIE
		Możliwość deklarowania wyzwalaczy (triggerów) na poziomie instrukcji DML (INSERT, UPDATE, DELETE) wykonywanej na tabeli, poziomie każdego wiersza modyfikowanego przez instrukcję DML oraz na poziomie zdarzeń bazy danych (np. próba wykonania instrukcji DDL, start serwera, stop serwera, próba zalogowania użytkownika, wystąpienie specyficznego błędu w serwerze). Ponadto mechanizm wyzwalaczy powinien umożliwiać oprogramowanie obsługi		TAK / NIE

		instrukcji DML (INSERT, UPDATE, DELETE) wykonywanych na tzw. niemodyfikowalnych widokach (views).		
		W przypadku, gdy w wyzwalaczu na poziomie instrukcji DML wystąpi błąd zgłoszony przez motor bazy danych bądź ustawiony wyjątek w kodzie wyzwalacza, wykonywana instrukcja DML musi być automatycznie wycofana przez serwer bazy danych, zaś stan transakcji po wycofaniu musi odzwierciedlać chwilę przed rozpoczęciem instrukcji, w której wystąpił ww. błąd lub wyjątek.		TAK / NIE
		Powinna istnieć możliwość autoryzowania użytkowników bazy danych za pomocą rejestru użytkowników założonego w bazie danych.		TAK / NIE
		Baza danych powinna umożliwiać na wymuszanie złożoności hasła użytkownika, czasu życia hasła, sprawdzanie historii hasła, blokowanie konta przez administratora bądź w przypadku przekroczenia limitu nieudanych logowań.		TAK / NIE
		Przywileje użytkowników bazy danych powinny być określone za pomocą przywilejów systemowych (np. prawo do podłączenia się do bazy danych - czyli utworzenia sesji, prawo do tworzenia tabel itd.) oraz przywilejów dostępu do obiektów aplikacyjnych (np. odczytu / modyfikacji tabeli, wykonania procedury). Baza danych powinna umożliwiać nadawanie ww. przywilejów za pośrednictwem mechanizmu grup użytkowników / ról bazodanowych. W danej chwili użytkownik może mieć aktywny dowolny podzbiór nadanych ról bazodanowych.		TAK / NIE
		Możliwość wykonywania i katalogowania kopii bezpieczeństwa bezpośrednio przez serwer bazy danych. Możliwość zautomatyzowanego usuwania zbędnych kopii bezpieczeństwa przy zachowaniu odpowiedniej liczby kopii nadmiarowych - stosownie do założonej polityki nadmiarowości backup'ów. Możliwość integracji z powszechnie stosowanymi systemami backupu (Legato, Veritas, Tivoli, OmniBack, ArcServe itd). Wykonywanie kopii bezpieczeństwa powinno być możliwe w trybie offline oraz w trybie online (hot backup).		TAK / NIE
		Odtwarzanie powinno umożliwiać odzyskanie stanu danych z chwili wystąpienia awarii bądź cofnąć stan bazy danych do punktu w czasie. W przypadku odtwarzania do stanu z chwili wystąpienia awarii odtwarzaniu może podlegać cała baza danych bądź pojedyncze pliki danych.		TAK / NIE
		W przypadku, gdy odtwarzaniu podlegają pojedyncze pliki bazy danych, pozostałe		TAK / NIE

		pliki baz danych mogą być dostępne dla użytkowników.		
		Wbudowana obsługa wyrażeń regularnych zgodna ze standardem POSIX dostępna z poziomu języka SQL jak i procedur/funkcji składowanych w bazie danych.		TAK / NIE
		Motor bazy danych na poziomie wskazanego numeru wersji wydania musi zapewnić wbudowany, własny mechanizm umożliwiający uruchomienia środowiska w konfiguracji klastrowej wykorzystującej dwa węzły jako podstawę architektury rozwiązania.		TAK / NIE
		Motor bazy danych musi oferować wbudowane rozwiązanie pozwalające na przechowywanie, analizę i wizualizację danych geolokalizacyjnych.		TAK / NIE
		Motor bazy danych musi oferować wbudowane rozwiązanie pozwalające na analizę powiązań, relacji między danymi wraz z możliwością wizualizacji tych powiązań i przedstawienia ich wraz z relacjami w postaci graficznej.		TAK / NIE
		Motor bazy danych musi oferować wbudowane rozwiązanie pozwalające na wykorzystanie zaawansowanych mechanizmów statystycznych, modeli danych, funkcjonalności wartościującej, celem tworzenia analiz, modelowania i predykcji danych przechowywanych w silniku bazy.		TAK / NIE

W Załączniku nr 1 (Szczegółowy Opis Przedmiotu Zamówienia) w Zadaniu nr 1 - Oprogramowanie do serwerów bazodanowych – 1 kpl. (4 licencje) w pkt. 4 „Oprogramowanie serwerów bazodanowych” jest:

4.	Oprogramowanie serwerów bazodanowych	Oferowany motor bazy danych musi być dostępny zarówno na platformy systemów operacyjnych Windows i Linux.		TAK / NIE
		Oferowany motor bazy danych musi mieć możliwość rozbudowy do wersji wspierającej możliwość synchronicznej replikacji danych w dwóch niezależnych centrach danych.		TAK / NIE
		Oferowany motor bazy danych posiada komercyjne wsparcie producenta. Nie dopuszcza się zastosowania RBD typu open-source.		TAK / NIE
		Oferowany motor bazy danych ma możliwość realizacji kopii bezpieczeństwa w trakcie działania (na gorąco).		TAK / NIE
		Oferowany motor bazy danych generuje kopie bezpieczeństwa automatycznie (o określonej porze) i na żądanie operatora oraz umożliwia odtwarzanie bazy danych z kopii archiwalnej, w tym sprzed awarii.		TAK / NIE
		Oferowany motor bazy danych umożliwia eksport i import danych z bazy danych w formacie tekstowym z uwzględnieniem polskiego standardu znaków.		TAK / NIE

	Administrator posiada możliwość wyboru danych, które mają być monitorowane w logach systemu z dokładnością do poszczególnych kolumn w tabelach danych, a zarządzanie nimi może odbywać się z poziomu narzędzi do zarządzania bazami danych (dopuszcza się narzędzie na poziomie motoru bazy danych).		TAK / NIE
	Oferowany motor bazy danych posiada możliwość rozbudowy do wersji wspierającej możliwość synchronicznej replikacji danych w dwóch niezależnych centrach danych.		TAK / NIE
	Oferowany motor bazy danych posiada komercyjne wsparcie producenta. Nie dopuszcza się zastosowania RBD typu open-source.		TAK / NIE
	Niezależność platformy systemowej dla oprogramowania klienckiego / serwera aplikacyjnego od platformy systemowej bazy danych.		TAK / NIE
	Możliwość przeniesienia (migracji) struktur bazy danych i danych pomiędzy ww. platformami bez konieczności rekompilacji aplikacji bądź migracji środowiska aplikacyjnego.		TAK / NIE
	Przetwarzanie z zachowaniem spójności i maksymalnego możliwego stopnia współbieżności. Modyfikowanie wierszy nie może blokować ich odczytu, z kolei odczyt wierszy nie może ich blokować do celów modyfikacji. Jednocześnie spójność odczytu musi gwarantować uzyskanie rezultatów zapytań odzwierciedlających stan danych z chwili jego rozpoczęcia, niezależnie od modyfikacji przeglądanego zbioru danych.		TAK / NIE
	Możliwość zagnieżdżania transakcji – powinna istnieć możliwość uruchomienia niezależnej transakcji wewnątrz transakcji nadrzędnej. Przykładowo – powinien być możliwy następujący scenariusz: każda próba modyfikacji tabeli X powinna w wiarygodny sposób odłożyć ślad w tabeli dziennika operacji, niezależnie czy zmiana tabeli X została zatwierdzona czy wycofana.		TAK / NIE
	Wsparcie dla wielu ustawień narodowych i wielu zestawów znaków (włącznie z Unicode).		TAK / NIE
	Możliwość migracji zestawu znaków bazy danych do Unicode.		TAK / NIE
	Możliwość redefiniowania przez klienta ustawień narodowych – symboli walut, formatu dat, porządku sortowania znaków za pomocą narzędzi graficznych.		TAK / NIE
	Skalowanie rozwiązań opartych o architekturę trójwarstwową: możliwość uruchomienia wielu sesji bazy danych przy wykorzystaniu jednego połączenia z		TAK / NIE

		serwera aplikacyjnego do serwera bazy danych.		
		Możliwość otworzenia wielu aktywnych zbiorów rezultatów (zapytań, instrukcji DML) w jednej sesji bazy danych.		TAK / NIE
		Wsparcie protokołu XA.		TAK / NIE
		Wsparcie standardu JDBC 3.0.		TAK / NIE
		Zgodność ze standardem ANSI/ISO SQL 2003 lub nowszym.		TAK / NIE
		Motor bazy danych powinien umożliwiać wskazywanie optymalizatorowi SQL preferowanych metod optymalizacji na poziomie konfiguracji parametrów pracy serwera bazy danych oraz dla wybranych zapytań. Powinna istnieć możliwość umieszczania wskazówek dla optymalizatora w wybranych instrukcjach SQL.		TAK / NIE
		Brak formalnych ograniczeń na liczbę tabel i indeksów w bazie danych oraz na ich rozmiar (liczbę wierszy).		TAK / NIE
		Wsparcie dla procedur i funkcji składowanych w bazie danych. Język programowania powinien być językiem proceduralnym, blokowym (umożliwiającym deklarowanie zmiennych wewnątrz bloku), oraz wspierającym obsługę wyjątków. W przypadku, gdy wyjątek nie ma zadeklarowanej obsługi wewnątrz bloku, w razie jego wystąpienia wyjątek powinien być automatycznie propagowany do bloku nadrzędnego bądź wywołującej go jednostki programu.		TAK / NIE
		Procedury i funkcje składowane powinny mieć możliwość parametryzowania za pomocą parametrów prostych jak i parametrów o typach złożonych, definiowanych przez użytkownika. Funkcje powinny mieć możliwość zwracania rezultatów jako zbioru danych, możliwego do wykorzystania jako źródło danych w instrukcjach SQL (czyli występujących we frazie FROM). Ww. jednostki programowe powinny umożliwiać wywoływanie instrukcji SQL (zapytania, instrukcje DML, DDL), umożliwiać jednoczesne otwarcie wielu tzw. kursorów pobierających paczki danych (wiele wierszy za jednym pobraniem) oraz wspierać mechanizmy transakcyjne (np. zatwierdzanie bądź wycofanie transakcji wewnątrz procedury).		TAK / NIE
		Możliwość kompilacji procedur składowanych w bazie do postaci kodu binarnego (biblioteki dzielonej).		TAK / NIE
		Możliwość deklarowania wyzwalaczy (triggerów) na poziomie instrukcji DML (INSERT, UPDATE, DELETE) wykonywanej na tabeli, poziomie każdego wiersza modyfikowanego przez instrukcję		TAK / NIE

		DML oraz na poziomie zdarzeń bazy danych (np. próba wykonania instrukcji DDL, start serwera, stop serwera, próba zalogowania użytkownika, wystąpienie specyficznego błędu w serwerze). Ponadto mechanizm wyzwalaczy powinien umożliwiać oprogramowanie obsługi instrukcji DML (INSERT, UPDATE, DELETE) wykonywanych na tzw. niemodyfikowalnych widokach (views).		
		W przypadku, gdy w wyzwalaczu na poziomie instrukcji DML wystąpi błąd zgłoszony przez motor bazy danych bądź ustawiony wyjątek w kodzie wyzwalacza, wykonywana instrukcja DML musi być automatycznie wycofana przez serwer bazy danych, zaś stan transakcji po wycofaniu musi odzwierciedlać chwilę przed rozpoczęciem instrukcji, w której wystąpił ww. błąd lub wyjątek.		TAK / NIE
		Powinna istnieć możliwość autoryzowania użytkowników bazy danych za pomocą rejestru użytkowników założonego w bazie danych.		TAK / NIE
		Baza danych powinna umożliwiać na wymuszanie złożoności hasła użytkownika, czasu życia hasła, sprawdzanie historii haseł, blokowanie konta przez administratora bądź w przypadku przekroczenia limitu nieudanych logowań.		TAK / NIE
		Przywileje użytkowników bazy danych powinny być określone za pomocą przywilejów systemowych (np. prawo do podłączenia się do bazy danych - czyli utworzenia sesji, prawo do tworzenia tabel itd.) oraz przywilejów dostępu do obiektów aplikacyjnych (np. odczytu / modyfikacji tabeli, wykonania procedury). Baza danych powinna umożliwiać nadawanie ww. przywilejów za pośrednictwem mechanizmu grup użytkowników / ról bazodanowych. W danej chwili użytkownik może mieć aktywny dowolny podzbiór nadanych ról bazodanowych.		TAK / NIE
		Możliwość wykonywania i katalogowania kopii bezpieczeństwa bezpośrednio przez serwer bazy danych. Możliwość zautomatyzowanego usuwania zbędnych kopii bezpieczeństwa przy zachowaniu odpowiedniej liczby kopii nadmiarowych - stosownie do założonej polityki nadmiarowości backup'ów. Możliwość integracji z powszechnie stosowanymi systemami backupu (Legato, Veritas, Tivoli, OmniBack, ArcServe itd). Wykonywanie kopii bezpieczeństwa powinno być możliwe w trybie offline oraz w trybie online (hot backup).		TAK / NIE
		Odtwarzanie powinno umożliwiać odzyskanie stanu danych z chwili		TAK / NIE

		wystąpienia awarii bądź cofnąć stan bazy danych do punktu w czasie. W przypadku odtwarzania do stanu z chwili wystąpienia awarii odtwarzaniu może podlegać cała baza danych bądź pojedyncze pliki danych.		
		W przypadku, gdy odtwarzaniu podlegają pojedyncze pliki bazy danych, pozostałe pliki baz danych mogą być dostępne dla użytkowników.		TAK / NIE
		Wbudowana obsługa wyrażeń regularnych zgodna ze standardem POSIX dostępna z poziomu języka SQL jak i procedur/funkcji składowanych w bazie danych.		TAK / NIE
		Motor bazy danych na poziomie wskazanego numeru wersji wydania musi zapewnić wbudowany, własny mechanizm umożliwiający uruchomienia środowiska w konfiguracji klastrowej wykorzystującej dwa węzły jako podstawę architektury rozwiązania.		TAK / NIE
		Motor bazy danych musi oferować wbudowane rozwiązanie pozwalające na przechowywanie, analizę i wizualizację danych geolokalizacyjnych.		TAK / NIE
		Motor bazy danych musi oferować wbudowane rozwiązanie pozwalające na analizę powiązań, relacji między danymi wraz z możliwością wizualizacji tych powiązań i przedstawienia ich wraz z relacjami w postaci graficznej.		TAK / NIE
		Motor bazy danych musi oferować wbudowane rozwiązanie pozwalające na wykorzystanie zaawansowanych mechanizmów statystycznych, modeli danych, funkcjonalności wartościującej, celem tworzenia analiz, modelowania i predykcji danych przechowywanych w silniku bazy.		TAK / NIE

Zmodyfikowana treść SWZ została zamieszczona na stronie internetowej prowadzonego postępowania pod adresem <https://e-propublico.pl> w dniu 31/10/2022 r.

**Z upoważnienia Dyrektora
SPZZOZ w Nisku**

Główna Księgowa

Ewa Świąder

/podpisano elektronicznie/