

ZAŁĄCZNIK NR 1

Opis przedmiotu zamówienia

Przedmiotem zamówienia jest:

1. Zaprojektowanie, zbudowanie oraz wdrożenie aplikacji klienckiej „Concept Note” pracującej w trybie OFFLINE (opis formularza w załączniku nr 2)
2. zaprojektowanie, zbudowanie i wdrożenie aplikacji webowej jako odrębnego narzędzia pozwalającego na import, eksport i modyfikację danych zgromadzonych z aplikacji klienckiej „Concept Note”.

W ramach przedmiotu umowy Wykonawca:

1. przeprowadzi konsultacje z Zamawiającym dotyczące przedmiotu zamówienia, w tym dokona analizy potrzeb i wymagań oraz uzgodni koncepcję poszczególnych elementów zamówienia,
2. opracuje dokumentację projektowo-techniczną dla poszczególnych komponentów przedmiotu zamówienia,
3. opracuje, wykona, wdroży aplikację kliencką „Concept Note” pracującą w trybie OFFLINE dla systemów operacyjnych Windows XP (32-bit, 64-bit), Windows 7 (32-bit, 64-bit) lub nowsze,
4. opracuje, wykona, wdroży aplikację webową działającą na serwerze www zamawiającego do zarządzania, raportowania i gromadzenia danych zebranych z aplikacji klienckiej „Concept Note”,
5. opracuje architekturę informacji dla wymienionych aplikacji,
6. opracuje dokumentację powykonawczą aplikacji klienckiej oraz aplikacji webowej do gromadzenia danych,
7. umożliwi Zamawiającemu przeprowadzenie testów dotyczących działania obu aplikacji
8. przekaże licencje na dostarczone oprogramowanie oraz zobowiąże się do jego aktualizacji do 31.12.2019r,
9. przeprowadzi szkolenia dla Zamawiającego (około 30 osób),
10. przeniesie na Zamawiającego majątkowe prawa autorskie do opracowanego oprogramowania wraz z prawem do zezwalania na wykonywanie praw zależnych,
11. przeniesie na Zamawiającego własność nośników, na których przekazane zostaną Zamawiającemu wszystkie aplikacje wytworzone w ramach umowy,
12. świadczyć będzie usługę serwisu gwarancyjnego do 31.12.2019r.

Etap 1: budowa aplikacji klienckiej „Concept Note”

Opracowanie i dostarczenie Zamawiającemu dokumentacji Etapu 1, w szczególności:

- a) Dokumentacja Analityczna (procesy biznesowe, diagramy)
- b) Dokumentacja Techniczna (wymagania techniczne, sprzętowe, programowe)
- c) Plan Testów Akceptacyjnych

Opis aplikacji klienckiej w trybie OFFLINE,

- a) Formularz w aplikacji jest rozbudowany i wypełnienie pól jednorazowo przez użytkownika może nie być możliwe. Użytkownicy muszą mieć w dowolnym momencie wypełniania formularza możliwość zapisu danych bez ostatecznego zatwierdzenia/przesłania formularza. Podczas takiego zapisu niektóre walidacje pól mogą zostać pominięte. Formularze powinny uwzględniać m. in.:
- tytuł pola
 - opis pola
 - pomoc kontekstową (tooltip) - dla każdego pola może pojawić się dodatkowa informacja o tym, jakie dane przyjmuje pole, do czego służy, co użytkownik powinien w nie wprowadzić, itp.
- b) Wpisanie lub wybór wartości w polu A może skutkować pokazaniem/ukryciem pól B i C, zmianą ich opisu, ustawieniem ich, jako tylko do odczytu lub zmianą wartości, które można wybrać w tych polach oraz wartości domyślnych (np. Wybór w polu "Kraj" powoduje załadowanie do pola "Region" listy regionów przypisanych do tego kraju, zmiana wyboru w polu "Region" powoduje pokazanie pola „Powiat” i załadowanie do niego listy powiatów w danym regionie; zaznaczenie pola "Inne koszty" powoduje pojawienie się dodatkowego pola na wpisanie opisu kosztów, itp.).
- Zmiana wartości w polu, od którego zależą inne pola może wywoływać ostrzeżenie dla użytkownika o możliwej utracie już wpisanych danych (np. zmiana wyboru kraju może spowodować usunięcie wartości z pól regionu i powiatu, które bezpośrednio i pośrednio zależą od wyboru kraju).
- c) Formularz powinien być wzorowany na istniejącym formularzu o nazwie „Concept Note”
Załącznik nr2
- d) Na formularzach mogą występować m. in. następujące rodzaje pól:
- pola tekstowe jednoliniowe
 - pola tekstowe wieloliniowe
 - listy wyboru jednokrotnego
 - listy wyboru wielokrotnego
 - pola zaznaczania wielokrotnego (lista checkbox)
 - pola zaznaczania jednokrotnego (radio button)
 - listy wyboru z wyszukiwaniem
 - tabele
- e) Formularz powinien obsługiwać standardowe zachowanie klawiatury wg. ustawień regionalnych systemu
- Formularz powinien mieć następujące skróty klawiszowe:
- Ctrl + S - zapis danych formularza
 - Ctrl + C - skopiowanie wartości do schowka systemowego
 - Ctrl + V - wklejenie wartości ze schowka systemowego
 - Ctrl + X - wycięcie wartości i skopiowanie do schowka systemowego
 - Ctrl + Z - cofnięcie ostatniej operacji
 - Ctrl + Y - powtórzenie ostatniej operacji
 - Tab - przejście do kolejnego pola formularza

- shift + Tab - przejście do poprzedniego pola formularza

Formularz powinien w wybranych miejscach pozwalać na dodawanie i usuwanie wierszy oraz kolumn w tabelach.

- f) Wartości wpisane w pola formularza powinny być walidowane przez system pod kątem:
- wymagalności - czy pole wymaga wpisania wartości
 - zakresu danych

Walidacja pola może zależeć od wartości innego pola formularza (np. pole A może być wymagane, jeżeli pole B jest wypełnione, wartość pola A nie może być większa od wartości pola B i nie może być mniejsza od sumy wartości pól C i D).

Walidacja pola może być blokująca lub ostrzegająca - rozwiązanie może ostrzec, że w polu znajduje się nieprawidłowa wartość i pozwolić zapisać zmiany lub też nie przyjąć wprowadzonej wartości i nie pozwolić na zapis. Ustawienie walidacji może być różne przy zapisie i zapisie z eksportem.

- g) Po akceptacji wprowadzonych danych przez użytkownika aplikacja uruchomi walidację pól zgodnie z wytycznymi w Załączniku NR 2 (formularz Concept Note)
- h) Jeśli walidacja przebiegła pomyślnie zostanie wygenerowany dokument w formie PDF gotowy do wydruku zgodny z wytycznymi w Załączniku nr 2 formularz Concept Note z widoczną wygenerowaną sumą kontrolną (wygenerowanego pliku XML).
- i) Wygenerowany poprawnie plik XML zostanie zapisany na dysku użytkownika z sumą kontrolną jako nazwa pliku <suma_kontrolna>.xml, która jest identyczna i widoczna na wygenerowanym dokumencie PDF.
- j) Plik XML po przesłaniu do Centrum Projektów Europejskich zostanie zaimportowany do aplikacji webowej.

Technologia tworzenia aplikacji klienckiej „Concept Note” musi zapewnić poprawne działanie w systemie operacyjnym Windows XP 32-bit oraz 64-bit, Windows 7 32-bit oraz 64-bit lub nowszy.

Jeżeli aplikacja kliencka będzie wymagała instalacji, to powinna być realizowana przez kreator.

Kreator powinien pozwolić użytkownikowi na:

- instalację, deinstalację i naprawę instalacji
- wybór lokalizacji zainstalowanych plików

Proces instalacji lub uruchomienia aplikacji powinien być prosty i zrozumiały dla użytkownika.

Wszystkie potrzebne moduły, biblioteki, pliki do prawidłowego działania muszą być instalowane lub uruchamiane razem z aplikacją kliencką „Concept Note” bez konieczności pobierania dodatkowego oprogramowania firm trzecich.

Etap 2: Budowa aplikacji webowej do zarządzania zaimportowanymi danymi.

Akceptowana technologia tworzenia aplikacji webowej do zarządzania zebranymi danymi z aplikacji klienckiej „Concept Note”:

PHP5 lub nowszy, SQL, JavaScript, HTML, CSS, JAVA, AJAX

Aplikacja uruchomiona będzie na serwerze Zamawiającego w sieci lokalnej.

Opracowanie i dostarczenie Zamawiającemu dokumentacji Etapu 2, w szczególności:

- a) Dokumentacja Analityczna
- b) Dokumentacja Techniczna
- c) Plan Testów Akceptacyjnych

Funkcjonalności aplikacji webowej

- a) Połączenie do aplikacji webowej musi odbywać się za pomocą protokołu szyfrującego HTTPS.
- b) Loginy i hasła nadawane są przez administratora aplikacji webowej.
- c) Podczas uruchomienia aplikacji webowej system powinien wymagać podania przez użytkownika loginu i hasła. Po zweryfikowaniu poprawności dostarczonych danych system powinien weryfikować przydzielone uprawnienia i udostępnić właściwe funkcjonalności.
- d) Rozwiązanie nie powinno wymagać od użytkowników bezpośredniego wprowadzania danych do bazy danych lub zmiany wartości plików na centralnym serwerze. Cała interakcja z systemem powinna odbywać się za pośrednictwem aplikacji webowej.
- e) Uprawnieni użytkownicy powinni mieć prawo dodawać, edytować i usuwać wybrane elementy list z poziomu interfejsu aplikacji wewnętrznej, bez potrzeby łączenia się bezpośrednio z bazą danych.
Rozwiązanie powinno uwzględniać fakt, że różne grupy użytkowników mogą mieć różne uprawnienia do oglądania i edycji wpisów oraz do poszczególnych atrybutów każdego z obiektów.
- f) Każdy obiekt w systemie powinien posiadać informacje o: autorze, dacie stworzenia, dacie ostatniej modyfikacji oraz osobie dokonującej ostatniej modyfikacji.
- g) Zaimportowane dane powinny być automatycznie wprowadzone przez system. W przypadku wystąpienia problemów z importem niektórych danych, system powinien zaimportować poprawne dane i wyróżnić pola formularza, do których dane nie zostały zaimportowane oraz wyświetlić prawdopodobny powód wystąpienia problemu (brak danych, zły format danych, wartość spoza dozwolonego zakresu, itp.). Przed rozpoczęciem importu danych, plik do zaimportowania powinien zostać zweryfikowany pod kątem poprawności jego struktury.
- h) Aplikacja webowa powinna przechowywać historię zmian. Historia powinna uwzględniać m. in. informacje:
 - kto dokonał zmiany
 - kiedy zmiana została wykonana
 - które atrybuty zostały zmienione oraz jaką nową wartość otrzymały.

Historia zmian powinna uwzględniać także stworzenie nowego obiektu.

- i) Zawartość list obiektów powinna być wyświetlana w formie widoków. Jedna lista może mieć wiele widoków, przeznaczonych dla różnych użytkowników. Użytkownik powinien móc na żądanie filtrować i posortować widok wg. innych parametrów. Filtrowanie powinno pozwalać

na użycie różnych operatorów (=, <, >, zawiera, zaczyna się od, itp.) oraz ustalenie wielu parametrów na raz (np. nr zaczyna się od 2016 i data wprowadzenia > 2016-06-12).

- j) Widok listy w bieżącej strukturze oraz z wyświetlanymi danymi powinno dać się wyeksportować do aplikacji Excel lub pliku CSV.
- k) W aplikacji webowej powinna istnieć możliwość obsługi wielu naborów (zbierania danych z aplikacji klienckiej) np. nabór I (rok 2016), nabór II (rok 2017). Aplikacja powinna umożliwiać użytkownikom wybierać, z którym naborem chcą pracować. Administratorzy rozwiązania powinni móc z poziomu interfejsu graficznego dodać nowy nabór.
- l) Użytkownicy powinni mieć możliwość dołączania plików do wybranych obiektów (np. umowy, wniosku, raportu). Użytkownicy powinni móc wybrać pliki z lokalnych zasobów komputera użytkowników, nośników przenośnych lub udziałów sieciowych i przekazać je do bazy danych równocześnie łącząc z wybranym obiektem. Załączone pliki powinny być dostępne dla wszystkich użytkowników mogących oglądać wybrany obiekt, bez potrzeby posiadania dodatkowych uprawnień (pliki powinny być przechowywane w centralnej bazie danych).
- m) Użytkownicy powinni móc oznaczyć w aplikacji webowej, czy wniosek spełnia kryteria i przeszedł ocenę techniczną. Użytkownicy powinni móc także oznaczyć wniosek jako odrzucony i móc podać powody odrzucenia jako komentarz.
- n) Kod źródłowy powinien być pisany zgodnie z dobrymi praktykami.
- o) Rozwiązanie powinno przetrzymywać wszystkie wprowadzone informacje w jednej centralnej bazie danych.
- p) Rozwiązanie powinno posiadać 3 odseparowane warstwy - prezentacji, logiki biznesowej i warstwy danych.
- q) Przed importowaniem pliku XML z aplikacji klienckiej sprawdzana jest suma kontrolna i wyświetlona komunikatem przed akceptacją importu przez użytkownika, tak by można było zweryfikować poprawność dokumentu w wersji papierowej z importowanym plikiem XML.
- r) zaimportowane dane będą wyświetlane w sposób czytelny dla użytkowników aplikacji webowej.
- s) Aplikacja musi umożliwić generowanie zestawień, raportów zgodnie z wytycznymi jako przykład w Załączniku 8 raport.xls

Aplikacja kliencka Concept Note oraz aplikacja webowa muszą być wdrożone i poprawnie działać u Zamawiającego do 20 września 2016 roku.

Zamawiający dopuszcza zmiany w funkcjonalności aplikacji klienckiej oraz aplikacji webowej zaproponowane przez Wykonawcę po uprzedniej zgodzie pisemnej.