

## **Załącznik nr 16**

### **Wzór Dokumentacji Technicznej**

1. Dokumentacja Techniczna będzie składać się z podstawowego dokumentu opisującego podział Systemu na poszczególne Moduły oraz osobnej dokumentacji każdego z wyszczególnionych Modułów. Dokument podstawowy będzie dokumentem głównym, do którego referować będą DT Modułów. Definicje Systemu i Modułu na potrzeby wytwarzania DT wskazano w sekcji „Definicje w zakresie DT”.
2. Kompletna dokumentacja techniczna powinna zostać opracowana zgodnie z dobrymi praktykami inżynierskimi tak, aby w oparciu o kompletną dokumentację techniczną ARiMR mogła samodzielnie uruchomić własne środowisko (zainstalować, skonfigurować i uruchomić System).
3. DT powinna być traktowana całościowo, natomiast w przypadku modyfikacji powinna być dostarczana nowa wersja DT z wyszczególnieniem zmian (w sekcji „Historia wersji (zmian) dokumentu” oraz ze zmianami w dokumentach Word zapisanymi w trybie rejestracji zmian (TRZ)) w poszczególnych modułach.
4. Dokumentacja Techniczna powinna być przygotowana zgodnie z szablonem określonym w niniejszym dokumencie (szablonem Dokumentu Podstawowego DT oraz szablonem DT Modułu).
5. Definicje w zakresie DT:
  - 5.1. „System” - w kontekście DT definiuje określoną aplikację lub grupę funkcjonalności zrealizowaną na wspólnej platformie.
  - 5.2. „Moduł” - w kontekście DT definiuje określony obszar biznesowy w ramach dokumentowanego Systemu.
6. Założenia w zakresie DT:
  - 6.1. Zakłada się, że diagramy będą tworzone w narzędziu Enterprise Architect i będą załączane w formie załączników HTML oraz XMI (dla części załączników, w określonych przypadkach, np. w zakresie bazy danych lub dokumentacji kodów źródłowych, dopuszcza się zastosowanie dodatkowych narzędzi oraz inne formaty załączników).
  - 6.2.** Dokumentacja DT Modułów będzie realizowana w osobnych dokumentach niż DT Systemu Podstawowego.

## Wzór

<b>ZAMAWIAJĄCY</b>	<b>Dokumentacja Techniczna</b> Numer rrrr.mm.nr	<b>WYKONAWCA</b>
--------------------	--	------------------

### Dane dokumentu

<b>Nazwa projektu:</b>	
<b>Wykonawca:</b>	<b>Numer wersji:</b>
<b>Data wersji:</b>	

### Historia wersji (zmian) dokumentu

Nr wersji	Data wersji	Opis	Nazwa pliku

## SPIS TREŚCI

<b>DT – SZABLON I ZAKRES DOKUMENTU PODSTAWOWEGO.....</b>	<b>4</b>
1. <b>Ogólny opis przeznaczenia systemu, jego zastosowania oraz przyjętego sposobu realizacji .....</b>	<b>4</b>
2. <b>Opis modelu logicznego systemu .....</b>	<b>4</b>
3. <b>Opis wzorców projektowych .....</b>	<b>5</b>
4. <b>Opis technologii i narzędzi implementacyjnych .....</b>	<b>5</b>
5. <b>Dokumentacja dotycząca kodu źródłowego .....</b>	<b>5</b>
6. <b>Opis konfiguracji i parametryzacji systemu .....</b>	<b>6</b>
7. <b>Rozmieszczenie poszczególnych pakietów oprogramowania: .....</b>	<b>6</b>
8. <b>Dokumentacja warstwy Infrastruktury .....</b>	<b>6</b>
1) <b>Specyfikacja sprzętu.....</b>	<b>6</b>
2) <b>Warstwa bazodanowa .....</b>	<b>7</b>
3) <b>Warstwa aplikacyjna .....</b>	<b>7</b>
4) <b>Sieć LAN.....</b>	<b>7</b>
5) <b>System backup’owy .....</b>	<b>7</b>
6) <b>SAN.....</b>	<b>7</b>
7) <b>Schematy środowisk .....</b>	<b>8</b>
8) <b>Zapotrzebowanie na licencje / oprogramowanie .....</b>	<b>8</b>
<b>DT – SZABLON I ZAKRES DOKUMENTACJI MODUŁU SYSTEMU.....</b>	<b>8</b>
1. <b>Szczegółowy opis modułu systemu realizującego określone funkcjonalności: .....</b>	<b>8</b>
2. <b>Szczegółowa specyfikacja bazy danych wraz z diagramem relacyjnym (ERD).....</b>	<b>9</b>
3. <b>Dokumentacja kodów źródłowych .....</b>	<b>9</b>
4. <b>Lokalizacja danych dla formatek na interfejsie użytkownika.....</b>	<b>9</b>

## **DT – szablon i zakres dokumentu podstawowego.**

### **1. Ogólny opis przeznaczenia systemu, jego zastosowania oraz przyjętego sposobu realizacji**

*Zwięzły biznesowy opis przeznaczenia systemu oraz jego zastosowania. Dodatkowo ogólny techniczny opis sposobu realizacji systemu (ogólna wysoko-poziomowa architektura systemu - zastosowane technologie, źródła danych, interfejs użytkownika). Dodatkowo ogólny diagram architektury systemu (z lotu ptaka).*

### **2. Opis modelu logicznego systemu**

#### **1) Zależności systemu od obiektów zewnętrznych (perspektywa zewnętrzna systemu)**

*Diagram z określeniem elementów zewnętrznych (nie stanowiących części systemu) od których system zależy, np. zewnętrzny serwer LDAP, zewnętrzny koncentrator danych, szyna usług innego systemu itp.*

*Należy zaprezentować diagram systemu oraz otoczenia peryferyjnego koncentrując się na warstwach i komponentach odpowiedzialnych za komunikację zewnętrzną.*

*Dla każdego interfejsu należy przedstawić opis techniczny warstw wymiany danych (opisać warstwy komunikacyjne, scharakteryzować techniczną definicję przesyłanych obiektów).*

#### **2) Podział na moduły**

*Tabela modułów systemu zgodnie z wcześniejszą definicją Modułu wraz ze zwięzłym opisem biznesowym każdego z modułów*

#### **3) Schematu powiązań między podsystemami (perspektywa wewnętrzna integracji)**

*Diagram ze schematem powiązań z innymi podsystemami (techniczne rodzaje interfejsów systemu, np. interfejs EJB, interfejs JMS, interfejs usługowy, interfejs bazodanowy itp.)*

*Należy zaprezentować diagram podsystemów koncentrując się na warstwach i komponentach odpowiedzialnych za komunikację.*

*Dla każdego interfejsu należy przedstawić opis techniczny (opisać warstwy komunikacyjne, scharakteryzować techniczną definicję przesyłanych obiektów)*

*Granulacja podziału na podsystemy powinna być zgodna z zaprezentowaną ontologią w podpunkcie 2.2*

#### **4) Diagram przepływu informacji**

*Diagram zorientowany na źródła danych - przedstawiający możliwe techniczne przepływy danych w systemie oraz pomiędzy systemem i innymi systemami (uwzględniający np. bazę danych, interfejs użytkownika, plik importu, pliki eksportu, bazę zewnętrzną, repozytorium LDAP, repozytorium dokumentów, szynę usług itp.)*

*Diagram powinien szczegółowo prezentować warstwy i komponenty istotne z punktu widzenia przetwarzania informacji w systemie. Diagram wraz ze szczegółowym opisem odnoszącym się do technicznych aspektów realizacji oprogramowania, powinien obrazować przepływ informacji w systemie w oparciu o reprezentatywne dla obiegu informacji diagramy sekwencji. Przepływ powinien obejmować warstwę GUI, wszystkie warstwy i komponenty klienta, komponenty serwerów aplikacyjnych, komponenty warstw middleware, warstw bazodanowych oraz warstw komunikacyjnych z systemami zewnętrznymi lub podsystemami.*

#### **5) Słownik pojęć/rejestr definicji modułów (jeden wspólny słownik dla wszystkich modułów)**

*Tabela z pojęciami i definicjami stosowanymi w DT*

#### **6) Specyfikacja protokołów i interfejsów komponentów wraz ze schematami typów danych (perspektywa wewnętrzna systemu/podsystemów)**

*Diagram komponentów technicznych systemu (np. komponent Fasada Usług EJB RMI, Fasada Usług WSDL, Komponent Web GUI, Komponent Procesów BPEL, Komponent DAO itp.) ze wskazaniem protokołów komponentów (np. SOAP, WSDL, RMI, JDBC itp.) oraz ze wskazaniem typów danych (np. XML, TXT, Java POJO, SQL itp.)*

### 3. Opis wzorców projektowych

#### 1) Opis wzorców z podziałem na warstwy, koncepcje kontrolerów ekranów, fasady, fabryki obiektów

*Opis - zastosowane w systemie wzorce projektowe ze wskazaniem komponentów systemu, które je implementują  
Dla każdego z zastosowanych wzorców (frameworków) należy wskazać warstwy i komponenty oraz przykładowe klasy (pakiety) stanowiące ich realizację.*

#### 2) Przyjęte konwencje nazewnictwa

*Opis - przyjęte konwencje nazewnictwa dla kodów źródłowych systemu oraz metadanych bazy danych*

### 4. Opis technologii i narzędzi implementacyjnych

*Opis - wskazanie technologii i narzędzi implementacyjnych ze zwięzłym wskazaniem ich zastosowania. Szerszy opis/dokumentacja technologii i narzędzi jest dostarczana przez dostawcę oprogramowania standardowego.*

### 5. Dokumentacja dotycząca kodu źródłowego

*Specyfikacja kodu źródłowego (np. kod Java modułu Moduł1 w katalogu Katalog1, biblioteki jar w katalogu Katalog2, biblioteki dll w katalogu3, skrypty np.:Maven lub inne narzędzia użyte w procesie budowania oraz wszystkie potrzebne elementy umożliwiające zbudowanie aplikacji w Katalogu4, itd.). Wszystkie wymienione tu elementy powinny pozwolić na zbudowanie aplikacji. Przez budowę aplikacji rozumie się wygenerowanie wszystkich potrzebnych plików (dll, exe, jar, war, sar, lib, xml itp.), które są potrzebne do deploymentu (czyli uruchomienia aplikacji).*

*Instrukcja zbudowania aplikacji (jeśli istnieją skrypty do budowy aplikacji to należy wskazać, który skrypt uruchomić, opisać możliwe do zastosowania parametry skryptu, opisać co i gdzie w wyniku budowy powinno powstać) oraz instrukcja deploymentu aplikacji powinna pojawić się w dokumentacji Administratora.*

*Będą zawarte następujące załączniki:*

- *JavaDoc w pliku ZIP. Załącznik/Kod źródłowy - opis w kodach źródłowych pakietów dla klas publicznych Javy przy metodach oraz atrybutach publicznych - zawarte opisy tak aby JavaDoc zawierał treść a nie tylko szkielet klas i ich metod. W szczególności pola o charakterze wyliczeniowym typu status, typ itp. winny mieć wymienione możliwe wartości wraz z ich opisem. (nie dotyczy słowników dokumentowanych w AOM lub edytowalnych przez użytkownika) Analogicznie do JavaDoc powinny być udokumentowane kody w innych językach (np. Basic, C, PL/SQL, Javascript, itp.) oraz powinny zawierać/uwzględniać wszystkie warstwy oprogramowania (dokumentacja kodów źródłowych dla innych języków niż Java może mieć inną formę, w szczególności może znajdować się bezpośrednio w kodzie źródłowym jako komentarze). Załączone arkusze stylu XSL stosowane w aplikacji i służące do transformacji XML-i (np.: do zobrazowania w postaci HTML). W szczególności których XSD (XML-i) dotyczą arkusze XSL. Będą wersjonowane pakiety sql ze wskazaniem przyczyny zmiany w historii zmian.*
- *Export bazodanowy DDL w formie HTML zakłada się że załącznik będzie generowany przy użyciu standardowego narzędzia, np. Enterprise Architect oraz TOAD . Dodatkowe wymagania w zakresie załącznika (poniżej specyfikacja XML) nie możliwe do wygenerowania z narzędzia np. EA lub TOAD będą dołączane jako dodatkowe sekcje w dokumencie lub dodatkowe załączniki. W przypadku gdy w kolumnach bazodanowych zawarte są XML-e powinny być dołączone do dokumentacji technicznej pliki xsd-schemy lub dtd wraz z opisami znaczników pól i struktur XML (ze względów technicznych opisy pól będą realizowane w sekcji opisu klasy XSD). W szczególności pola o charakterze wyliczeniowym typu status, typ itp. winny mieć wymienione możliwe wartości wraz z ich opisem (nie dotyczy słowników dokumentowanych w AOM lub edytowalnych przez użytkownika)*

*Należy przedstawić (w opisowej formie) odwzorowania artefaktów programistycznych kodu źródłowego na warstwy i komponenty techniczne zaprezentowane w punkcie 2.6.*

*Instrukcja Budowy Aplikacji będzie zawierać:*

- *szczegółowy opis środowiska budowy (system operacyjny, wersje oprogramowania narzędziowego wraz z wersjami wymaganych bibliotek), ustawienia plików konfiguracyjnych związanych z procesem budowy*
- *szczegółowa instrukcja „krok po kroku”, jakie czynności należy wykonać, jakich sprawdzeń należy dokonać w poszczególnych etapach budowy systemu (kryteria weryfikacji)*
- *szczegółowy opis wynikowych plików (to co jest niezbędne do rozpoczęcia procesu deploymentu).*

## 6. Opis konfiguracji i parametryzacji systemu

### 1) Opis komponentów aplikacji do zarządzania konfiguracją i parametryzacją

Opis - komponentów systemu odpowiedzialnych za parametryzację i konfigurację systemu, np. komponent Parametryzacji Biznesowej, Komponent Konfiguracji Systemu itp.

### 2) Opis plików konfiguracyjnych aplikacji

Opis roli pliku konfiguracyjnego/parametryzacyjnego w aplikacji (nie dotyczy plików konfiguracyjnych oprogramowania standardowego). Wskazanie które komponenty aplikacji zarządzają którymi plikami. W przypadku gdy plik konfiguracyjny jest w formacie XML powinny być dołączone do dokumentacji technicznej pliki xsd-schemy lub dtd wraz z opisami znaczników pól i struktur XML (ze względów technicznych opisy pól będą realizowane w sekcji opisu klasy XSD). W szczególności pola o charakterze wyliczeniowym typu status, typ itp. winny mieć wymienione możliwe wartości wraz z ich opisem (nie dotyczy słowników dokumentowanych w AOM lub edytowalnych przez użytkownika)

### 3) Opis tabel bazodanowych o charakterze konfiguracyjnym w aplikacji

Opis metadanych oraz jaka jest rola tabeli konfiguracyjnej / parametryzacyjnej w aplikacji (nie dotyczy tabel konfiguracyjnych oprogramowania standardowego). Wskazanie które komponenty aplikacji zarządzają którymi tabelami. W szczególności pola o charakterze wyliczeniowym typu status, typ itp. winny mieć wymienione możliwe wartości wraz z ich opisem (nie dotyczy słowników dokumentowanych w AOM lub edytowalnych przez użytkownika)

## 7. Rozmieszczenie poszczególnych pakietów oprogramowania:

Diagram „deploymentu” (wraz z opisem) z:

- nazwami hostów
- nazwami oprogramowania systemu, np. komponent Web, komponent OSB, komponent usług biznesowych itp.
- nazwami serwerów aplikacyjnych, np. WLS, SOA Suite, OSB, LDAP itp.
- Nazwami systemów operacyjnych, np. Windows, Linux itp. na warstwie sprzętowej

## 8. Dokumentacja warstwy Infrastruktury

### 1) Opis rozwiązania

[Ogólny opis rozwiązania w zakresie infrastruktury; przypisanie ilości sprzętu w podziale na elementy środowiska produkcyjnego i środowisk pomocniczych, schemat rozmieszczenia elementów]

### 1) Specyfikacja sprzętu

[Konfiguracja i zestawienie parametrów technicznych elementów infrastruktury]

Tabela zapotrzebowanie na infrastrukturę (należy podać typ i konfigurację.):

Rola	Rodzaj serwera	Licencje	Środowisko			
			Produkcyjny	Odbiorowy	Przedprodukcyjny	Szkoleniowy
[Np. HTTP proxy]	[Serwer Proxy]	[Web Tier]	.	.	.	.
[LDAP]	[np. Korzystamy z istniejącej infrastruktury]					

Tabela zawiera zestawienie parametrów technicznych maszyn na których uruchomiona jest aplikacja.

Maszyna	Typ maszyny	Typ procesora	Ilość core	Ilość RAM	Przeznaczenie
Nazwa maszyny/serwera	Typ serwera	jw	Ilość rdzeni procesora	Ilość pamięci ram	Jaką rolę będzie pełnił serwer i czy jest to serwer fizyczny czy wirtualny oraz czy jest to dodatkowy element SI

## 2) Warstwa bazodanowa

[Opis rodzaju i konfiguracji baz danych, ich wersji, klastrów bazodanowych, użytych systemów plików, parametry konfiguracyjne bazy danych]

## 3) Warstwa aplikacyjna

[opis oprogramowania wspomagającego (systemy operacyjne, middleware itp.), opis komunikacji pomiędzy elementami systemu i ich konfiguracja]

[Diagram rozmieszczenie komponentów na serwerach, mapowanie oprogramowania aplikacyjnego (bazy danych serwery aplikacyjne, aplikacja) na infrastrukturę.]

[Opis/schemat powinien pokazywać sprzęt niezbędny do eksploatacji aplikacji oraz mapowania, co najmniej, wszystkich modułów, które zidentyfikowano]

## 4) Sieć LAN

[adresacja, połączenia pomiędzy poszczególnymi elementami sieci, rodzaje połączeń, obciążenie łącz, opis infrastruktury sieciowej, w tym na infrastrukturze obszaru Bezpiecznego dostępu do Internetu (BI)]

## 5) System backup'owy

[opis systemu backupu, opis polityki backup'owej]

## 6) SAN

[informacje odnośnie przestrzeni dyskowych, ilości woluminów, przypisania woluminów do serwerów, infrastruktura SAN, opis konfiguracji przestrzeni dyskowej po stronie systemu operacyjnego]

Tabela zapotrzebowanie na przestrzeń:

	Produkcyjny (nazwa serwerów z adresami portów do których należy wystawić przestrzeń)	Odbiorowy (nazwa serwerów z adresami portów do których należy wystawić przestrzeń)	Przedprodukcyjny (nazwa serwerów z adresami portów do których należy wystawić przestrzeń)	Szkoleniowy (nazwa serwerów z adresami portów do których należy wystawić przestrzeń)
[np. Przestrzeń dla (bazy danych X – referencja do komponentu)]	[X TB]	[X TB]	[X TB]	[X TB]

## 7) Schematy środowisk

*[schematy środowisk pomocniczych i środowiska produkcyjnego, mapowanie środowisk na infrastrukturę]*

## 8) Zapotrzebowanie na licencje / oprogramowanie

Aplikacja będzie używała następujących produktów i bibliotek – *[większość z tych rzeczy pojawia się później w instrukcji administratora, należy przedstawić mapowanie licencji na infrastrukturę - schemat]:*

Produkt / biblioteka	Czy obecnie używane w SI	Uzasadnienie
<i>[Np.Oracle WLS]</i>	<i>[tak (wersja X.Y), wymagana licencja]</i>	<i>[Standardowy komponent do ...]</i>

Tabela zapotrzebowanie na licencje (wskazanie wszystkich licencji wraz wyszczególnieniem ewentualnych dodatkowym modułów zarówno warstwy aplikacyjnej jak i infrastrukturalnej w tym systemu operacyjnego):

Rola	Rodzaj/nazwa serwera	Licencje	Środowisko
<i>Rola w SI</i>	<i>jw.</i>	<i>Nazwa rodzaj licencji</i>	<i>[podział wykorzystania i zapotrzebowanie na licencje w zależności od środowiska]</i>

## DT – szablon i zakres dokumentacji modułu systemu.

### 1. Szczegółowy opis modułu systemu realizującego określone funkcjonalności:

#### 1) Przeznaczenie modułu i sposób jego wykorzystania

*Opis biznesowy modułu i sposobu wykorzystania modułu.*

#### 2) Charakterystyka działania – opis zastosowanych wzorców projektowych oraz algorytmów

*Opis - referencja do dokumentu podstawowego ze wskazaniem zastosowanych wzorców projektowych oraz dodatkowy opis zastosowanych specyficznych w określonym module algorytmów (rozwiązań) wraz ze wskazaniem klas implementacyjnych np. sposobu generowania raportu, sposobu generowania wydruku do określonego formatu, eksportu danych, importu danych, wyszukiwania danych itp.*

#### 3) Struktury danych

*Opis/Załącznik - specyfikacja struktur danych specyficznych dla modułu (poza określonymi w punkcie b)) np. struktura XSD pliku XML, struktura TXT pliku importu/eksportu itp.*

#### 4) Diagramy statyczne i dynamiczne (diagram sekwencji, diagram klas z podziałem na warstwy)

*Diagramy statyczne klas dla modułu w zakresie warstw: warstwy wymiany danych - np. encje OB, XBVO oraz diagramy sekwencji dla reprezentatywnych przebiegów funkcjonalności w zakresie modułu.*

*Dla każdej kategorii diagramu należy przedstawić (scharakteryzować) zastosowaną notację opisu oraz dokonać szczegółowego opisu każdego diagramu (w szczególności diagramu sekwencji, prezentując uczestnika procesu z zachowaniem pełnej zgodności z obiektami uwzględnionymi w diagramach statycznych, np. diagramie klas). Należy zwrócić uwagę na ścisłą korelację wszystkich artefaktów projektowych ujętych w prezentowanych diagramach modułu z diagramami zobrazowanymi w głównym dokumencie systemu.*



## **2. Szczegółowa specyfikacja bazy danych wraz z diagramem relacyjnym (ERD)**

*Załącznik z diagramem ERD generowany z narzędzia Enterprise Architect lub TOAD oraz Referencja do załącznika Dokumentu Głównego (Exportu bazodanowym DDL w HTML).*

- 1) Diagram ERD opisujący relacje**
- 2) Lista tabel**
- 3) Lista sekwencji**
- 4) Lista synonimów**
- 5) Lista widoków**
- 6) Lista wyzwalaczy**
- 7) Lista pakietów PL/SQL**

*Niezależnie od komentarzy zawartych w kodzie źródłowym pakietów (specyfikacja, implementacja) należy przedstawić szczegółową charakterystykę każdej z procedur i funkcji pakietu uwzględniając opis modelu ich użycia (zarówno w kontekście funkcjonalnym jak i architektonicznym [w którym komponencie i na poziomie której warstwy następuje ich egzekucja])*

## **3. Dokumentacja kodów źródłowych**

*Szczegółowy opis pakietów, klas z punktu widzenia realizowanej funkcjonalności biznesowej oraz technicznych aspektów zastosowanej architektury, wzorców projektowych i frameworków.*

## **4. Lokalizacja danych dla formatek na interfejsie użytkownika**

*Tabela/Załącznik - dla każdej formatki/zakładki (np. dla parametrów wyboru) powinna być wskazana lokalizacja danych w źródle danych (wskazanie tabeli i kolumny w której znajdują się prezentowane dane lub tabel i kolumn wraz z warunkami ich wyboru, jeśli źródłem danych jest więcej niż jedna tabela/kolumna w zależności od kontekstu). Dla pól wyliczanych zgodnie z AOM (nie przechowywanych w bazie danych) wskazywany będzie algorytm wyliczania na zasadzie wskazania odpowiedniego artefaktu analitycznego i/lub reguły.*