Załącznik nr 7a do SIWZ – Informacja o produkcie – wzór

.....
(pieczęć wykonawcy)

INFORMACJE O OFEROWANYM PRODUKCIE

Ja (My), niżej podpisany(ni)

.....

działając w imieniu i na rzecz :

.....

(pełna nazwa wykonawcy)

.....

(adres siedziby wykonawcy)

w odpowiedzi na ogłoszenie o przetargu nieograniczonym na:

„Zaprojektowanie, budowę, dostawę, wdrożenie, utrzymanie, serwisowanie systemów i aplikacji tworzących architekturę platform technologicznych oraz infrastruktury sprzętowej IT dla centrum danych SPNT Sp. z o.o. w ramach projektów: 1. "Przetwarzanie w chmurze dla rozwoju miast cyfrowych - faza rozwoju" - Działanie 5.1 Programu Operacyjnego Innowacyjna Gospodarka 2. „Budowa i wyposażenie I Etapu Pomerania Technopark w Szczecinie przy ul. Niemierzyńskiej - Poddziałanie 1.2.1 Regionalnego Programu Operacyjnego Województwa Zachodniopomorskiego" Część I przedmiotu zamówienia, przedstawiamy informacje o produkcie:

INNOWACYJNA GOSPODARKA
NARODOWA STRATEGIA SPÓJNOŚCI
TECHNOPARK POMERANIA
UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ ROZWOJU REGIONALNEGO

Projekt „Przetwarzanie w chmurze dla rozwoju miast cyfrowych - faza rozwoju" współfinansowany jest ze środków Europejskiego Funduszu Rozwoju Regionalnego w ramach Programu Operacyjnego Innowacyjna Gospodarka „Fundusze Europejskie - dla rozwoju innowacyjnej gospodarki"

PROGRAM REGIONALNY
NARODOWA STRATEGIA SPÓJNOŚCI
Pomorze Zachodnie
TECHNOPARK POMERANIA
UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ ROZWOJU REGIONALNEGO

Projekt „Budowa i wyposażenie I etapu Pomerania Technopark w Szczecinie przy ul. Niemierzyńskiej" współfinansowany ze środków Europejskiego Funduszu Rozwoju Regionalnego w ramach Regionalnego Programu Operacyjnego województwa zachodniopomorskiego na lata 2007-2013

1. Routery brzegowe – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Router brzegowy klasy operatorskiej.	
Obudowa dedykowana do montażu w szafie typu rack 19".	
Architektura modularna, pozwalająca na instalację modułów w postaci: - kart liniowych, - sprzętowych redundantnych modułów zarządzających, - modułów medium transmisyjnego (transreceiverów).	
Architektura wewnętrzna zapewniająca odseparowanie funkcji kontrolnych (control plane) takich jak routing, sygnalizacja, zarządzanie od obsługi ruchu użytkowego (data plane).	
Rozbudowa urządzenia o dodatkowe karty interfejsów powinna odbywać się bez konieczności instalacji dodatkowej matrycy przełączającej oraz konieczności wymiany modułów zarządzających i przy zachowaniu wszystkich wymogów funkcjonalnych i wydajnościowych zawartych w niniejszej specyfikacji.	
Wsparcie dla HA poprzez obsługę przełączania kart procesorowych w tryb active/active.	
Minimum 4 sloty na karty liniowe (nie uwzględniając procesorowych).	
Wymagana obsługa prędkości portów w urządzeniu:	

10Mbps/100Mbps/1Gbps/10Gbps.	
Ilość pamięci operacyjnej w kartach procesorowych: minimalnie 4GB.	
Urządzenie wyposażone w następującą ilość portów światłowodowych (minimum): 8x10Gbps i 20x1Gbps, porty muszą mieć możliwość obsługi diagnostyki modułów optycznych dostarczonych w ramach projektu; porty muszą mieć możliwość obsługi modułów światłowodowych innych producentów niż producent urządzenia z zachowaniem gwarancji na sprzęt.	
Urządzenie wyposażone w min 2 zasilacze AC umożliwiające pracę urządzenia w pełnej konfiguracji (tzn. przy obsadzeniu modułami wszystkich slotów urządzenia).	
Urządzenie wyposażone w min 2 karty procesorowe.	
Wydajność matrycy przełączającej: minimum 75Gbps.	
Wydajność przełączania pakietów: minimum 50Mpps.	
Sprzętowe wsparcie dla pełnego routingu IPv4/IPv6, co najmniej: static, OSPF, IS-IS, BGP.	
Wielkość tablic w sprzętowej tablicy przełączania FIB dla IPv4/IPv6: min. 1 miliona dla IPv4 i 256 tysięcy dla IPv6; podane parametry powinny być spełnione dla każdego protokołu niezależnie, również przy jednoczesnym wykorzystaniu obu protokołów; wielkość tablic RIB powinna wynosić min 2x wielkość tablic FIB dla poszczególnych protokołów.	
BFD musi być obsługiwane dla IPv4/IPv6	
Sprzętowa realizacja przełączania przełącznika.	
Wsparcie dla MTU 9216 bajtów.	
Obsługa sieci VLAN zgodna z IEEE 802.1q dla 4094 sieci VLAN jednocześnie; obsługa dowolnej translacji identyfikatorów VLAN, również w przypadku zastosowania	

mechanizmu Q-in-Q.	
Obsługa Spanning Tree 802.1d, 802.1w, 802.1s, ramki BPDU pomiędzy sieciami VLAN muszą być przenoszone także przy użyciu MPLS/VLPS.	
Obsługa sprzętowego eksportu statystyk ruchu (eksport danych do kolektora flow); dla łącza o przepływności 1Gbps - bez próbkowania (samplingu), dla ruchu powyżej 1Gbps - możliwe wykorzystanie próbkowania (samplingu).	
Wsparcie dla: PBR, VRRP.	
Obsługa standardów: L2 VPN, L3 VPN, MPLS VPN, MPLS, MPLS TE, VPLS.	
Obsługa standardów: kolejkowanie na portach: min 8 kolejki na port, limitowanie ruchu ingress/egreess na portach/vlanach, shaping lub policing ruchu per port.	
Obsługa mechanizmów kolejkowania ruchu, jego filtrowania oraz znakowania w oparciu o 802.1p, DSCP, ToS, MPLS EXP na wszystkich portach oraz dla poszczególnych sieci VLAN.	
Obsługa SNMP v1/v2/v3 (informacje o MIB, np. specyfikacje OID, muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Obsługa oraz integracja z sieciami SDN, przynajmniej poprzez otwarty protokół.	
Obsługa RADIUS/TACACS/TACACS+	

2. Switche szkieletowe – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Switch szkieletowy klasy operatorskiej.	
Obudowa dedykowana do montażu w szafie typu rack 19''.	
Zasilanie prądem przemiennym 230V AC, wyposażone w minimum dwa redundantne zasilacze.	
Architektura modularna.	
Regulacja prędkości wentylatorów w zależności od temperatury wewnątrz obudowy.	
Minimum 4 GB pamięci RAM.	
Min. 8 slotów do obsługi kart liniowych.	
Min. ilość obsługiwanych portów : 40x 1Gbps, 40x 10 Gbps, 24x 40Gbps; architektura urządzenia gotowa na obsługę min. 2x100Gbps.	
Min. ilość portów zainstalowanych: 16x 40Gbps, 24x 10Gbps, 16x 1Gbps.	
Redundantna praca zasilaczy - awaria jednego modułu nie może wpłynąć na pracę switcha w pełnej konfiguracji wymiana w trybie HotSwap - bez konieczności wyłączania Systemu.	
Możliwość redundancji kart procesorowych - praca w trybie active/active i active/standby.	
Przepustowość matrycy przełączającej min. 4,8 Tb/s.	
Przepustowość kart liniowych min. 100 Gb/s na slot.	
Sprzętowa obsługa ilości tras IPv4/IPv6: min. 128 000 dla IPv4 i 64 000 dla IPv6; podane parametry powinny być spełnione dla każdego protokołu niezależnie, również przy jednoczesnym wykorzystaniu obu protokołów.	
Wszystkie porty obsługiwane z pełną prędkością łącza (wire speed).	
Sprzętowa realizacja modułu przełączania.	

Obsługa nie mniej niż 100 tysięcy adresów MAC na każdą kartę liniową lub nie mniej niż 1 mln. na cały przełącznik.	
Obsługa ramek Jumbo (min. MTU 9000 bajtów).	
Obsługa sieci VLAN zgodna z IEEE 802.1q dla 4000 sieci VLAN jednocześnie oraz Q-in-Q, możliwa dowolna translacja vlanów.	
Obsługa standardu 802.3ad.	
Obsługa Spanning Tree 802.1d, 802.1w, 802.1s, ramki BPDU pomiędzy sieciami VLAN muszą być przenoszone także przy użyciu MPLS/VLPS.	
Sprzętowa obsługa QoS ingress/egress, możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL.	
Sprzętowa realizacja routingu.	
Obsługiwane protokoły routingu IPv4: static, RIP, OSPF, IS-IS, BGP.	
Obsługiwane protokoły routingu IPv6: static, RIP, OSPF, IS-IS, BGP.	
Obsługa mechanizmu BFD.	
Wymagana możliwość stworzenia klastra VRRP.	
Reguły ACL w oparciu o kryteria warstw 2-4 dla ingress i egress.	
Obsługa SNMP v1/v2/v3 (informacje o MIB, np. specyfikacje OID, muszą być dostępne dla Zamawiającego), CLI (minimum ssh/telnet), syslog, NTP.	
Obsługa oraz integracja z sieciami SDN, integracja poprzez otwarty protokół.	
Obsługa RADIUS/TACACS/TACACS+	

3. Switch top of rack – 6szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U, wielkość urządzenia 1U.	
Minimum dwa wewnętrzne zasilacze 230V AC, z redundancją zasilania.	
Wyposażony w następującą ilość portów (z obsługą diagnostyki modułów): min. 40 portów 10Gbps (z czego przynajmniej 20 musi mieć obsługę modułów 1Gbps i 10Gbps zamiennie), oraz minimum 4 porty 40Gbps.	
Tablica adresów MAC: min. 100000 pozycji.	
Urządzenie musi się charakteryzować wydajnością "wire speed" dla funkcji przełącznika, wydajność: min. 500Gbs, przepustowość: min. 500Mpps.	
Możliwość wsparcia dla pełnego routingu IPv4/IPv6, co najmniej: static, OSPF, IS-IS, BGP.	
Wsparcie dla multicast routing: PIM-SM lub PIM-DM.	
Obsługa VRRP.	
Obsługa PBR, IPv6 tunneling, BFD.	
Obsługa DHCP Snooping, Option 82, DHCP Relay, Option 82, DHCP Snooping Trust.	
Obsługa IGMP v1/v2/v3.	
Obsługa 4094 jednoczesnych VLAN, translacji VLAN.	

Obsługa Spanning Tree, 802.1d, 802.1w, 802.1s.	
Obsługa DCBX, 802.1Qbb, IEEE 802.1Qaz.	
Obsługa 802.3ad.	
Kontrola mac adresów w poszczególnych VLANACH.	
Wsparcie dla MTU 9000 bajtów.	
Możliwość obsługi BFD dla RIP, OSPF, BGP, IS-IS, VRRP, MPLS.	
Funkcjonalność przełączania pakietów non-STP, zgodność z protokołem IETF TRILL lub SPB (IEEE 802.1aq).	
Wielkość tablic w sprzętowej tablicy przełączania FIB dla IPv4/IPv6: min. 16000 dla IPv4 i 8000 dla IPv6; podane parametry powinny być spełnione dla każdego protokołu niezależnie, również przy jednoczesnym wykorzystaniu obu protokołów.	
Możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL.	
Klasyfikacja QOS po ACL, 802.1p, IP, DSCP, ToS.	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Obsługa oraz integracja z sieciami SDN, poprzez otwarty protokół.	
Obsługa RADIUS/TACACS/TACACS+	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	
Dedykowany port do zarządzania RJ-45.	

4. Firewall/UTM – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U.	
Obsługa portów 1Gbps i 10Gbps, urządzenie musi posiadać minimum 2 porty 10Gbps i 16 portów 1Gbps.	
Zasilanie 230V AC z możliwością wymiany w czasie pracy urządzenia (tzw. hot-swap); urządzenie musi być wyposażone w minimum dwa zasilacze AC.	
Minimum 3Mpps dla pakietów 64 bajtów.	
Minimum 2mln jednoczesnych połączeń/sekundę.	
Minimum 5Gbps dla ruchu szyfrowanego VPN.	
Minimum 2Gbps dla ruchu IPS.	
Sprzętowe wsparcie dla pełnego routingu IPv4/IPv6, co najmniej: static,OSPF, IS-IS, BGP.	
Obsługa minimum 500 sesji BGP.	
Obsługa mechanizmów QOS (policing, kolejkowanie, shaping), obsługa DSCP, IP ToS, 802.1p, WRED, obsługa tworzenia osobnych kolejek dla różnych klas ruchu.	
Ochrona przed atakami DoS/DDoS realizowana sprzętowo o wydajności min. 5Gbps. Dopuszcza się realizację tego wymagania poprzez wdrożenie dedykowanego (niezależnego od UTM) urządzenia lub systemu (również działającego w trybie redundancji).	
Urządzenie musi posiadać funkcję wykrywania i blokowania	

ataków intruzów (IPS, intrusion prevention) wspomaganą sprzętowo. System zabezpieczeń musi identyfikować próby skanowania, penetracji i włamań, ataki typu exploit (poziomu sieci i aplikacji), ataki destrukcyjne i destabilizujące (D)DoS oraz inne techniki stosowane w atakach na sieć. Ustalenie blokowanych ataków (intruzów, robaków) musi odbywać się w regułach polityki bezpieczeństwa. Baza sygnatur IPS musi być utrzymywana i udostępniana przez producenta urządzenia firewall.	
Obsługa statefull firewall dla ruchu IMIX z wydajnością nie mniejszą niż 10Gbps.	
System zabezpieczeń musi identyfikować próby skanowania, penetracji, włamań, ataki typu exploit, ataki destrukcyjne i destabilizujące.	
Obsługa pracy w trybie HA Active-Active, tak by przełączanie pomiędzy urządzeniami odbywało się przezroczyście dla ruchu użytkowników; mechanizm ochrony przed awariami musi monitorować i wykrywać uszkodzenia elementów sprzętowych i programowych systemu zabezpieczeń oraz łączy sieciowych.	
Wsparcie dla VPN-IPSEC.	
Obsługa IPv4, IPv6.	
Obsługa SNMP v1/v2/v3 (informacje o MIB, np. specyfikacje OID, muszą być dostępne dla zamawiającego), CLI (minimum ssh/telnet), syslog, NTP.	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	

5. Load balancers – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Musi być urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U lub oprogramowaniem dedykowanym do wirtualizacji i przystosowanym do uruchomienia w ramach infrastruktury IaaS.	
W przypadku urządzenia dedykowanego: powinno być wyposażone w porty minimum 2x 10Gbps oraz minimum 2x 1Gbps.	
Minimalnie 8GB RAM.	
Minimum dwa wewnętrzne zasilacze 230V AC, z redundancja zasilania.	
Minimum 250GB pojemności dysku twardego.	
Wsparcie load balancing dla: TCP, UDP, HTTP, HTTPS.	
Minimum 2Gbps dla ruchu L4/L7.	
Minimum 4 miliony jednoczesnych połączeń L4.	
Zapewnienie dowolnej liczby sesji load balance na poziomie TCP, UDP, HTTP, HTTPS.	
Obsługa algorytmów load balance: round robin, wagowany round robin, random losowy, najmniejsza liczba połączeń, wagowany.	
Obsługa zarządzania sesją użytkownika przez adres źródłowy lub HTTP Cookie.	
Obsługa zarządzania LB wraz z opcjami automatycznego podłączenia/odłączenia load balancowanych nodów powinna być realizowana przez funkcjonalność urządzenia, instancji	

wirtualnej lub dostarczony system zarządzania - w każdym przypadku wymagana jest integracja z IaaS.	
Obsługa terminacji SSL dla One-Way SSL i Two-Way SSL.	
Obsługa certyfikatów SSL dowolnego dostawcy.	
Zarządzanie nagłówkami HTTP przekazywanymi do node'ów (w tym nagłówkami z polami certyfikatów SSL).	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Monitoring urządzenia powinien być realizowany przez dedykowane Podsystemy monitoringu Centrum Danych (np. za pomocą SNMP).	

6. Serwery – 64szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	**Oferowany parametr/funkcjonalność:**
Architektura serwerowa, zalecany jest dobór serwerów typu "high density servers" z grupowaniem serwerów per obudowa ze wspólnym zasilaniem w celu optymalizacji zużycia energii elektrycznej.	
Minimum 6 dysków hot-swap z obsługą SAS/NL-SAS/SATA/SSD dla pojedynczego node serwera.	
2 porty USB.	
1 port VGA.	
1 slot PCI-E.	

Redundantne zasilanie (per obudowa) 230V AC.	
Zainstalowane 2 porty 10Gbps, wspierające sprzętowo: - 802.1q VLAN - TCP segmentation offload - IPv6 obsługa dla IP/TCP i IP/UDP receive checksum offload - wsparcie dla wirtualizacji oraz obsługa ramek jumbo o rozmiarze min.9000 bajtów.	
Zainstalowane 2 porty 1Gbps miedziane.	
Zainstalowany 1 port dedykowany na zarządzanie.	
Zainstalowane minimum 64GB RAM 1866Mhz ECC Registered, obsługa maksimum 256GB 1866Mhz ECC Registered w jednym serwerze, sumaryczna ilość pamięci RAM w całym środowisku nie może być mniejsza niż: 3840GB.	
Minimum 16 rdzeni procesora na serwer (nie uwzględniając wielowątkowości procesora, tzw. HT), taktowanie: min. 2.5Ghz na rdzeń.	
Minimum 16MB last level cache per processor.	
Oferowany model procesora musi osiągać w teście CPU MARK wynik bazowy minimum 10000pkt. (wynik zaproponowanego procesora musi znajdować się na stronie http://www.cpubenchmark.net/ w dniu składania ofert. Wydruk ze strony załączyć do oferty).	
Ewentualna rozbudowa o dodatkowy procesor lub inny osprzęt musi odbywać się bez dodatkowych licencji.	
Wsparcie dla procesora do wirtualizacji.	
Minimum 46TB pojemności dysków enterprise SATA SSD (pojemność RAW, sumarycznie na wszystkich serwerach) oraz minimum 768TB pojemności dysków SATA enterprise (pojemność RAW, sumarycznie na wszystkich serwerach); rozmieszczenie położenia dysków w poszczególnych	

obudowach powinno zostać dobrane na etapie projektu technicznego z uwzględnieniem wybranego podejścia do rozproszonego storage; przykładowy podział: 48 serwerów w obudowach po 8 sztuk każda, wyposażonych w 2x 480GB SSD każdy oraz 16 serwerów wyposażonych w 16x 3TB SATA każdy. MTBF dla każdego z dysków zastosowanych w serwerach minimum 1.4 miliona godzin dla enterprise SATA i minimum 2 miliony godzin dla enterprise SATA SSD. Dla dysków SSD parametr TBW (Total Bytes Written) powinien wynosić min. 270TB (w rozumieniu standardów JESD218 oraz JESD 219).	
Wyposażony w sprzętowy kontroler dysków z minimum 1GB pamięci, obsługujący dyski SAS, NL-SAS, SATA, SSD z obsługą RAID przynajmniej poziom 0, 1, 10, 5.	
Obsługa dysków SAS/SATA/SSD dowolnego producenta, w tym możliwość użycia dysków tzw. OEM (niekoniecznie dedykowanych/markowych); wykorzystanie dysków OEM nie powinno zmniejszać programowo lub licencyjnie wydajności/przepustowości Systemu oraz nie powinno powodować utraty gwarancji.	
Zarządzanie urządzeniami poprzez IPMI (Intelligent Platform Management Interface), zgodność z IPMI 2.0, KVM over LAN / KVM over IP.	
Obsługa IPv4, IPv6.	
Zarządzanie po dedykowanym, wbudowanym porcie Ethernet.	
Obsługa interfejsu WWW dla zarządzania.	
Zarządzanie serwerem przy wyłączonej maszynie, możliwe uruchomienie zdalne systemu, restart, wyłączenie, włączenie, możliwość montowania obrazu oraz instalacji systemu operacyjnego za pomocą interfejsu zarządzania.	

7. Macierz – 1szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Uzupełnienie istniejącej infrastruktury IBM w postaci macierzy IBM V3700 z 4 modułami rozszerzeń.	
Modułowa do instalacji w standardowej szafie Rack 19".	
Dwa redundantne kontrolery udostępniające w sumie nie mniej niż 8 połączeń FC 8Gb i 4 połączenia iSCSI minimum 1Gb. RAID 0,1,5,6,10. Wymagane jest, aby architektura wewnętrzna macierzy wykorzystywała standard SAS 2.0.	
Co najmniej 16GB pamięci „cache". Pamięć „cache" przeznaczona dla procesu zapisu musi być zabezpieczona przed skutkami awarii jednego z kontrolerów.	
Macierz musi obsługiwać dyski SSD (o pojemnościach 200GB i 400GB), SAS (o pojemnościach 146GB, 300GB, 600GB, 900GB) i NL-SAS lub SATA (o pojemnościach 500GB, 1TB, 2TB, 4TB) pozwalając na rozbudowę do co najmniej 120 dysków. Macierz musi obsługiwać dyski 2,5" i 3,5". Macierz musi umożliwiać mieszanie dysków SAS, NL-SAS i SSD w ramach jednej półki dyskowej. Dostawa ma obejmować 48 dysków SATA każdy o pojemności 4TB oraz 24 dysków SSD każdy o pojemności 400GB.	
Macierz musi mieć możliwość wykonywania migracji woluminów w ramach zasobów dyskowych bez zatrzymywania aplikacji z nich korzystających. Macierz musi posiadać	

możliwość migracji danych z zewnętrznych zasobów dyskowych.	
Macierz musi obsługiwać min 50 kopii migawkowych na macierz. Licencja na tę funkcjonalność musi być zawarta w cenie. Kopie danych typu „snapshot„ muszą być wykonywane przez macierz jako pojedyncza operacja w co najmniej trzech możliwych trybach: • kopia pełna, • kopia wskaźnikowa, • przyrostowa kopia pełna. Macierz musi mieć możliwość odtworzenia zawartości woluminu logicznego z kopii typu „snapshot" bez konieczności kopiowania danych za pośrednictwem serwera.	
Funkcjonalność konfigurowania woluminu dyskowego posiadającego dwie kopie fizyczne na różnych grupach dyskowych i różnego typu (np.: jedna kopia z nadalokacją druga bez). W przypadku zapisu macierz zapisuje do obu kopii synchronicznie. W przypadku odczytu czyta tylko z jednej kopii. Kiedy jedna kopia jest niedostępna macierz automatycznie korzysta tylko z dostępnej kopii a po naprawie brakującej kopii automatycznie synchronizuje dane. Wolumin mirrorowany może być przekształcony w zwykły wolumin poprzez usunięcie jednej kopii albo poprzez wyodrębnienie jednej kopii w osobny wolumin.	
Funkcjonalność dynamicznej alokacji przestrzeni dyskowej większej niż jest dostępna fizycznie oraz możliwość wyłączenia tej funkcjonalności dla wybranych woluminów.	
Wszystkie krytyczne komponenty takie jak: kontrolery dyskowe, pamięć „cache", zasilacze i wentylatory muszą być zdublowane w taki sposób, aby awaria pojedynczego elementu nie wpływała na funkcjonowanie całego Systemu. Brak	

pojedynczego punktu awarii. Wszelkie połączenia pomiędzy elementami składowymi macierzy (wszystkie ścieżki) muszą być redundantne. Wsparcie dla zasilania z dwóch niezależnych źródeł prądu poprzez nadmiarowe zasilacze typu „hot-swap". Wentylatory typu „hot-swap".	
Interfejs zarządzający GUI, CLI nie wymagający instalacji dodatkowego oprogramowania na stacji zarządzającej. Możliwość zmiany mikrokodu bez przerywania dostępu do danych. Monitorowanie stanu pracy za pośrednictwem protokołu SNMP. Automatyzacja procesu informacji o stanie urządzenia, w tym informacji o awariach za pomocą wiadomości przesyłanych drogą elektroniczną.	
Komplet wkładek FC, okablowanie zasilające, światłowodowe FC	

8. Rozbudowa macierzy – 1szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Dwa redundantne kontrolery udostępniające w sumie nie mniej niż 8 połączeń FC 8Gb i 2 połączenia iSCSI minimum 1Gb. RAID 0,1,5,6,10. Wymagane jest, aby architektura wewnętrzna macierzy wykorzystywała standard SAS 2.0. Rozbudowana macierz nie powinna posiadać programowych ograniczeń wydajności.	
Obudowa modułowa do instalacji w standardowej szafie Rack	

19".	
Rozbudowana macierz musi obsługiwać dyski SSD (o pojemnościach 200GB i 400GB, 800GB), SAS (o pojemnościach 146GB, 300GB, 600GB, 900GB i 1,2 TB) i NL-SAS lub SATA (o pojemnościach 500GB, 1TB, 2TB, 4TB) pozwalając na rozbudowę do co najmniej 120 dysków. Macierz musi obsługiwać dyski 2,5" i 3,5". Macierz musi umożliwiać mieszanie dysków SAS, NL-SAS i SSD w ramach jednej półki dyskowej. Dostawa macierzy musi obejmować minimum 9TB SSD pojemności RAW oraz minimum 43TB SAS pojemności RAW.	
Należy dostarczyć komplet wkładek FC, okablowanie zasilające, światłowodowe FC.	

9. Switche MGMT – 6szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	**Oferowany parametr/funkcjonalność:**
Switch klasy operatorskiej; min. ilość urządzeń tych urządzeń powinna być zgodna z zestawieniem ilościowym; w przypadku gdy projekt techniczny wykaże większe zapotrzebowanie należy dostarczyć ilość zgodną z projektem technicznym.	
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U, wielkość urządzenia 1U.	
Minimum dwa wewnętrzne zasilacze AC, z redundancją	

zasilania.	
Wyposażony w następującą ilość portów: min. 48 portów 10/100/1000Mbps , oraz minimum 4 porty 1Gbps.	
Wielkość tablicy mac-adresów minimum 12000 pozycji.	
Urządzenie musi być wire speed dla funkcji przełącznika, wydajność minimum 50Gbs, przepustowość minimum 50Mpps.	
Obsługa IPv4/IPv6.	
Obsługa DHCP Snooping, Option 82, DHCP Relay, Option 82, Dhcp Snooping Trust.	
Obsługa 4000 jednoczesnych VLAN, translacji VLAN.	
Obsługa Spanning Tree, 802.1d, 802.1w, 802.1s.	
Obsługa 802.3ad.	
Kontrola MAC adresów w poszczególnych VLANACH, obsługa wykrywania pętli tzw. loop detection.	
Wsparcie dla MTU 9000 bajtów.	
Tablica routingu minimum 4000 wpisów dla IPv4 oraz 1000 wpisów dla IPv6.	
Możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL	
Klasyfikacja QOS po ACL, 802.1p, IP, DSCP, ToS;	
Funkcja mirroringu portów: 1 to 1 Port mirroring, Many to 1 port mirroring	
Możliwość wyboru sposobu obsługi kolejek – Strict Priority; Weighted Round Robin;	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp	
Obsługa oraz integracja z sieciami SDN, poprzez otwarty protokół	

Obsługa RADIUS/TACACS/TACACS+	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	
Dedykowany port do zarządzania RJ-45	

10. Switche MGMT – agregacja – 1szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	**Oferowany parametr/funkcjonalność:**
Switch klasy operatorskiej.	
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U, wielkość urządzenia 1U.	
Minimum dwa wewnętrzne zasilacze AC, z redundancją zasilania.	
Wyposażony w następującą ilość portów: min. 20 portów 1Gbps.	
Wielkość tablicy MAC adresów minimum 12000 pozycji.	
Urządzenie musi być wire speed dla funkcji przełącznika, wydajność minimum 20Gbs, przepustowość minimum 20Mpps.	
Obsługa IPv4/IPv6.	
Obsługa DHCP Snooping, Option 82, DHCP Relay, Option 82, Dhcp Snooping Trust.	
Obsługa 4000 jednoczesnych VLAN, translacji VLAN.	
Obsługa Spanning Tree, 802.1d, 802.1w, 802.1s.	

Obsługa 802.3ad.	
Kontrola MAC adresów w poszczególnych VLANACH, obsługa wykrywania pętli tzw. loop detection.	
Wsparcie dla MTU 9000 bajtów.	
Tablica routingu minimum 4000 wpisów dla IPv4 oraz 1000 wpisów dla IPv6.	
Możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL.	
Klasyfikacja QOS po ACL, 802.1p, IP, DSCP, ToS.	
Funkcja mirroringu portów: 1 to 1 Port mirroring, Many to 1 port mirroring.	
Możliwość wyboru sposobu obsługi kolejek – Strict Priority; Weighted Round Robin.	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Obsługa oraz integracja z sieciami SDN, poprzez otwarty protokół.	
Obsługa RADIUS/TACACS/TACACS+	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	
Dedykowany port do zarządzania RJ-45.	

11. Konwertery, moduły (transceivers)

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Konwertery powinny poprawnie współpracować z portami sieciowymi do których zostaną podłączone. W przypadku konwerterów optycznych należy zależnie od wykorzystanych połączeń światłowodowych dobrać odpowiedni rodzaj (single/multi mode), moc optyczną (LX, SX, ZX) i jeżeli jest taka potrzeba wprowadzić tłumiki optyczne.	
Ilość i rodzaj konwerterów powinny zostać dobrane w trakcie przygotowywania projektu technicznego w taki sposób aby: - spełniać założone wymogi przepływnościowe pomiędzy węzłami i warstwami sieci, - gwarantować zachowanie wymogów dostępności i redundancji, - zapewnić połączenia do istniejącej infrastruktury. Do ilości wynikającej z projektu technicznego należy doliczyć 5% każdego rodzaju konwertera.	
Producent dostarczonych urządzeń musi dopuszczać możliwość wykorzystania modułów SFP/SFP+/QSFP/QSFP+/XFP pochodzących od innych producentów, tzn. obsługę modułów typu OEM bez utraty gwarancji czy supportu dla urządzenia.	
Dostarczone moduły optyczne muszą wspierać diagnostykę DDM (Digital Diagnostic Monitor) lub równoważną , tzn. monitorować kluczowe parametry pracy modułu takich jak moc optyczna sygnału nadawanego, moc optyczna sygnału odbieranego, temperatura pracy, napięcie zasilania, prąd lasera i w tym zakresie funkcjonalnym muszą poprawnie współpracować z urządzeniami wyposażonymi w te moduły.	

12. Bramka SMS – 1szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Dedykowane urządzenie do komunikacji GSM do wysyłania wiadomości SMS.	
Obsługa minimum dwóch kart SIM różnych operatorów działających na terenie Polski. Brak blokady sim-lock.	
Obsługa komunikatów SMS.	
Wyposażona w port minimum 2x 10/100 Mbps.	
Możliwość zasilania 230V AC i/lub DC z funkcją podtrzymania bateryjnego.	
Obsługa IPv4/IPv6.	
Obsługa NAT oraz VPN (przynajmniej PPTP).	
Obsługa logowania wysłanych wiadomości SMS.	
Obsługa funkcji routera przez dostęp do Internetu z sieci GSM, w celu awaryjnego dostępu do sieci.	
Możliwość integracji z systemem monitoringu poprzez API (automatyzacja wysyłania komunikatów) przez dedykowany port komunikacyjny.	
Dedykowany port do zarządzania.	
Monitoring stanu kart SIM GSM.	

13. Kompleks biurowy SPNT: switche dostępowe – 15szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U, wielkość urządzenia 1U.	
Minimum dwa wewnętrzne zasilacze AC, z redundancją zasilania.	
Wyposażony w następującą ilość portów: min. 48 portów 10/100/1000Mbps , oraz minimum 2 porty 10Gbps.	
Możliwość zbudowania jednego wirtualnego przełącznika z 4 przełączników tego samego typu tzw. funkcja stackowania; przełącznik wirtualny powinien być widziany przez inne urządzenia sieciowe jako pojedyncze urządzenie zarówno pod kątem mechanizmów warstwy L2 jak i warstwy L3; do podłączenia przełączników jako przełącznik wirtualny, musi być wykorzystany dedykowany port, połączenie przez dedykowany port musi być połączeniem w ringu, aby zminimalizować awarie w trakcie uszkodzenia kabla/przełącznika; przepustowość portu dedykowanego do połączenia między poszczególnymi urządzeniami musi być minimum 10Gbps.	
Wielkość tablicy MAC adresów minimum 12000 pozycji.	
Urządzenie musi być wire speed dla funkcji przełącznika, wydajność minimum 100Gbs, przepustowość minimum 100Mpps.	
Obsługa IPv4/IPv6.	

Obsługa DHCP Snooping, Option 82, DHCP Relay, Option 82, Dhcp Snooping Trust.	
Obsługa IGMP v1/v2/v3.	
Obsługa 4000 jednoczesnych VLAN, translacji VLAN.	
Obsługa Spanning Tree, 802.1d, 802.1w, 802.1s.	
Obsługa 802.1x, 802.3ad.	
Kontrola MAC adresów w poszczególnych VLANACH, obsługa wykrywania pętli tzw. loop detection.	
Wsparcie dla MTU 9000 bajtów.	
Tablica routingu minimum 4000 wpisów dla IPv4 oraz 1000 wpisów dla IPv6.	
Możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL.	
Klasyfikacja QOS po ACL, 802.1p, IP, DSCP, ToS.	
Funkcja mirroringu portów: 1 to 1 Port mirroring, Many to 1 port mirroring.	
Możliwość wyboru sposobu obsługi kolejek – Strict Priority; Weighted Round Robin.	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Obsługa oraz integracja z sieciami SDN, poprzez otwarty protokół.	
Obsługa RADIUS/TACACS/TACACS+	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	
Dedykowany port do zarządzania RJ-45.	

14. Kompleks biurowy SPNT: switche dostępowe POE – 9szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U, wielkość urządzenia 1U.	
Minimum dwa wewnętrzne zasilacze AC, z redundancją zasilania.	
Wyposażony w następującą ilość portów: min. 48 portów 10/100/1000Mbps, oraz minimum 2 porty 10Gbps.	
Możliwość zbudowania jednego wirtualnego przełącznika z 4 przełączników tego samego typu tzw. funkcja stackowania; przełącznik wirtualny powinien być widziany przez inne urządzenia sieciowe jako pojedyncze urządzenie zarówno pod kątem mechanizmów warstwy L2 jak i warstwy L3; do podłączenia przełączników jako przełącznik wirtualny, musi być wykorzystany dedykowany port, połączenie przez dedykowany port musi być połączeniem w ringu, aby zminimalizować awarie w trakcie uszkodzenia kabla/przełącznika; przepustowość portu dedykowanego do połączenia miedzy poszczególnymi urządzeniami musi być minimum 10Gbps.	
Wielkość tablicy MAC adresów minimum 12000 pozycji.	
Urządzenie musi być wire speed dla funkcji przełącznika, wydajność minimum 100Gbs, przepustowość minimum	

100Mpps.	
Obsługa IPv4/IPv6.	
Obsługa DHCP Snooping, Option 82, DHCP Relay, Option 82, Dhcp Snooping Trust.	
Obsługa IGMP v1/v2/v3.	
Obsługa 4000 jednoczesnych VLAN, translacji VLAN.	
Obsługa Spanning Tree, 802.1d, 802.1w, 802.1s.	
Obsługa 802.1x, 802.3ad.	
Kontrola MAC adresów w poszczególnych VLANACH, obsługa wykrywania pętli tzw. loop detection.	
Wsparcie dla MTU 9216 bajtów.	
Tablica routingu minimum 4000 wpisów dla IPv4 oraz 1000 wpisów dla IPv6.	
Możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL.	
Klasyfikacja QOS po ACL, 802.1p, IP, DSCP, ToS.	
Funkcja mirroringu portów: 1 to 1 Port mirroring, Many to 1 port mirroring.	
Możliwość wyboru sposobu obsługi kolejek – Strict Priority; Weighted Round Robin.	
Wsparcie dla 802.3af, 802.3at tzw. POE/POE+, minimalnie 14W na port miedziany, przy założeniu wykorzystania wszystkich portów moc jednego zasilacza musi wynosić minimum 700W; urządzenie wówczas musi być wyposażone w dwa zasilacze każdy po 700W.	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Obsługa oraz integracja z sieciami SDN, poprzez otwarty protokół.	

Obsługa RADIUS/TACACS/TACACS+	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	
Dedykowany port do zarządzania RJ-45.	

15. Kompleks biurowy SPNT: switche agregujące – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	**Oferowany parametr/funkcjonalność:**
Musi być dedykowanym urządzeniem sieciowym, przystosowanym do montażu w szafie RACK 19U, wielkość urządzenia 1U.	
Minimum dwa wewnętrzne zasilacze 230V AC, z redundancją zasilania.	
Wyposażony w ilość portów, z obsługą diagnostyki modułów minimum 24 porty 10Gbps.	
Tablica MAC adresów minimum 32000 pozycji.	
Urządzenie musi być wire speed dla funkcji przełącznika, wydajność minimum 480Gbs, przepustowość minimum 300Mpps.	
Obsługa IPv4/IPv6.	
Obsługa DHCP Snooping, Option 82, DHCP Relay, Option 82, Dhcp Snooping Trust.	
Obsługa IGMP v1/v2/v3.	
Obsługa 4000 jednoczesnych VLAN, translacji VLAN.	
Obsługa Spanning Tree, 802.1d, 802.1w, 802.1s.	

Obsługa 802.1x, 802.3ad.	
Kontrola MAC adresów w poszczególnych VLANACH.	
Wsparcie dla MTU 9000 bajtów.	
Funkcjonalność przełączania pakietów non-STP.	
Tablica routingu minimum 10000 wpisów dla IPv4 oraz 2000 wpisów dla IPv6.	
Możliwość ograniczania pasma na porcie (globalnie) oraz możliwość ograniczenia pasma dla ruchu określonego listą ACL.	
Klasyfikacja QOS po ACL, 802.1p, IP, DSCP, ToS.	
Obsługa SNMP v1/v2/v3 (informacje o MIB muszą być dostępne dla zamawiającego), cli (minimum ssh/telnet), syslog, ntp.	
Obsługa oraz integracja z sieciami SDN, poprzez otwarty protokół.	
Obsługa RADIUS/TACACS/TACACS+	
Zarządzanie przez system zarządzania i monitorowania pochodzący od tego samego producenta.	
Dedykowany port do zarządzania RJ-45.	

16. Kompleks biurowy SPNT: rozszerzenia licencji kontrolerów Wi-Fi – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	**Oferowany parametr/funkcjonalność:**
Wymagane 2 pakiety rozszerzenia licencji do obsługi 50 szt. Access Pointów do dwóch kontrolerów Rucus ZoneDirector	

3000.	

17. Kompleks biurowy SPNT: punkty dostępowe (AP) – 43szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	**Oferowany parametr/funkcjonalność:**
Punkt dostępowy współpracujący / w pełni kompatybilny z kontrolerem Rucus ZoneDirector 3000.	
Dwa tryby pracy: samodzielny (zarzadzanie punktem odbywa się poprzez interfejs przeglądarki internetowej, telnet i SSH) oraz zarządzania przez kontroler sieci bezprzewodowej.	
W trybie zarządzania przez kontroler sieci bezprzewodowej komunikacja z punktem dostępowym musi być szyfrowana.	
W trybie zarządzania przez kontroler sieci bezprzewodowej punkt dostępowy może znajdować się w tej samej, lub innej podsieci IP.	
Możliwość pracy jako cześć sieci kratowej (tj. „mesh") - bez podłączonego kabla Ethernet, z dynamicznym przełączeniem pomiędzy trybami i automatyczną konfiguracją.	
Równoczesna praca w pasmie 2,4 GHz i 5.x GHz.	
Praca w trybie MIMO 3x3:3.	
Wsparcie dla metody MRC (Maximal Ratio Combining).	
System musi zapewniać dostęp sygnału radiowego wokół punktu dostępowego, bez martwych pól.	
System musi zapewniać maksymalne wzmocnienie 9 dBi i filtrowanie interferencji na poziomie -15dBi.	

Automatyczna ochrona przed interferencjami sygnału.	
Anteny wbudowane i zintegrowane z punktem dostępowym.	
System antenowy musi się składać z nie mniej niż 12 elementów.	
Czułość odbiornika nie mniejsza niż -101dBm.	
Nie mniej niż 32 BSSID z własną polityką dostępu i regułami QoS.	
Nie mniej niż 4 kolejki QoS per stacja kliencka i wsparcie standardu 802.11e.	
Obsługa nie mniej niż 500 stacji, nie mniej niż 60 klientów głosowych jednocześnie.	
802.11d, 802.1Q, 802.1X.	
802.3af oraz 802.3at PoE.	
802.11a/b/g/n.	
WEP	
WPA-PSK	
WPA-TKIP	
WPA2-AES	
802.11i	
IEEE 802.11n: 2.4 – 2.484 GHz i 5.15 – 5.85 GHz	
IEEE 802.11a: 5.15 – 5.85 GHz	
IEEE 802.11b: 2.4 – 2.484 GHz	
802.11n: 6.5Mbps – 216,7Mbps (20MHz)	
802.11n: 13.5Mbps – 450Mbps (40MHz)	
802.11a: 54, 48, 36, 24, 18, 12, 9, 6Mbps	
802.11b: 11, 5.5, 2, 1 Mbps	
802.11g: 54, 48, 36, 24, 18, 12, 9, 6 Mbps	
Zasilanie poprzez PoE lub zasilacz 12V DC	
Maksymalna pobierana moc 13W.	
2 porty RJ-45, auto MDX, auto-sensing 10/100/1000 Mbps,	

jeden z możliwością zasilania PoE.	
Masa urządzenia nie większa niż 1 kg.	
Praca w temperaturze 0-50C, wilgotność do 95% (bez kondensacji).	
Homologacja do montażu w zamkniętych przestrzeniach UL 2043.	
Montaż bez konieczności użycia zewnętrznych akcesoriów i maskownic.	
Dynamicznego generowania kluczy Pre-Shared keys.	
Automatycznego wyboru najlepszego kanału pracy w oparciu o realną przepustowość/pojemność kanałów dla 2,4 GHz lub 5 GHz wraz z możliwością przeniesienia klienta na optymalny kanał z wykorzystaniem standardu 802.11h.	
Możliwość dobierania optymalnych kanałów transmisyjnych bez konieczności przerywania transmisji danych.	
Optymalizacja wydajności sieci przy różnych prędkościach dostępowych klientów (sterowanie czasem dostępu do punktu dostępowego na podstawie okien czasowych, nie ilości przesłanych danych).	
W celach diagnostycznych możliwość przechwycenia ramek 802.11/802.3 od/do klienta przesyłanych przez punkt dostępowy bez wpływu na trwającą komunikację.	

18. Szafy serwerowe – 4szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
DK TS8 kompatybilna z modułami LCP T3+	
Wyposażenie w wysokoszczelny szczotkowy moduł podłogi do wprowadzania kabli.	
Łączniki zewnętrzne.	
Uszczelki pionowe blokady strumienia powierzchni lewej i prawej strony LCP i poziom 19".	
2 moduły mocy PDU 3x32A gniazda 3x 6xC13, 3x 2xC19, funkcja wł.-wył. Pojedyncze gniazdo model CW-24VYM.	
2 moduł PDU - 19" z podłączeniem 32A , wyjście 4xC19 z funkcją włą-wył. Pojedyncze gniazdo.	
Czujnik temperatury i wilgotności.	
4 szyny poziome do prowadzenia kabli.	
10 wieszaków kablowych przelotowych.	
Funkcja automatycznego otwierania drzwi.	

19. Moduły chłodzące – 2szt.

Producent / model:.....

Dodatkowe informacje:.....

.....

Wymagany parametr/funkcjonalność:	Oferowany parametr/funkcjonalność:
LCP T3+ EC 47U 3300235, dla chłodzenia szaf serwerowych DK TS8 2200x1200, o mocy chłodniczej do 30 kW.	
Dwa aktywne obwody chłodzenia i prądowe.	
Wbudowane sterowniki monitoringu i zarządzania zdalną pracą przez Rittal CMC TC.	

Funkcja „Auto-Load-Balancing“.	
Funkcja „Auto-Recovery“.	
Ekran dotykowy do lokalnego zarządzania i monitorowania pracą wymiennika.	
Funkcja automatycznego otwierania drzwi szafy serwerowej dla gaszenia pożaru w szafie oraz dla chłodzenia awaryjnego.	
Kolor RAL 7035.	

20. IaaS

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.2 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.2 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	Oferowana funkcjonalność:
Wymagania ogólne	
IaaS musi być dedykowanym rozwiązaniem do budowy chmur publicznych (public clouds), prywatnych (private clouds), serwerów VPS, serwerów dedykowanych.	
IaaS musi zapewniać zarządzanie oraz monitorowanie dużą (powyżej 100 000) siecią maszyn wirtualnych bądź serwerów dedykowanych.	
W ramach wdrożenia Podsystemu IaaS muszą być zapewnione mechanizmy automatycznego provisioningu, zarządzania, monitoringu i skalowania maszyn wirtualnych cloud osadzonych na sprzęcie compute dla wszystkich wymaganych/wspieranych systemów operacyjnych.	
W ramach wdrożenia Podsystemu IaaS muszą być zapewnione mechanizmy automatycznego provisioningu, zarządzana, monitoringu i skalowania maszyn wirtualnych VPS osadzonych na sprzęcie compute dla wszystkich wymaganych/wspieranych systemów operacyjnych.	
W ramach wdrożenia Podsystemu IaaS muszą być zapewnione mechanizmy automatycznego provisioningu, zarządzana, monitoringu serwerów dedykowanych osadzonych na sprzęcie compute dla wszystkich wymaganych/wspieranych systemów operacyjnych.	
W ramach wdrożenia Podsystemu IaaS muszą być zapewnione mechanizmy automatycznego provisioningu, zarządzania, monitoringu i skalowania środowiska sprzętowego "Storage" dla maszyn wirtualnych cloud, VPS.	

W ramach wdrożenia Podsystemu IaaS muszą być zapewnione mechanizmy automatycznego provisioningu, zarządzania, monitoringu i skalowania środowiska sieciowego, w tym: sieci prywatnych (np. w oparciu o VLAN), adresacji prywatnej i publicznej IPv4 i IPv6 oraz sieci dostępowych VPN do usług. W przypadku protokołu IPv6 jeżeli oferowane rozwiązanie nie obsługuje na etapie składania oferty tej funkcjonalności, musi oferować ją na etapie realizacji Projektu.	
W ramach wdrożenia Podsystemu IaaS muszą być zapewnione mechanizmy automatycznego provisioningu, zarządzania, monitoringu i skalowania środowiska load balancerów.	
W ramach wdrożenia Podsystemu IaaS musi być zapewniona obsługa dwóch regionów: 1. "Głównego" w Centrum Danych Zamawiającego na sprzęcie dostarczonym w ramach niniejszego zamówienia (przedmiot niniejszego postępowania) oraz 2. "Zapasowego" na zakupionym, wdrożonym i rozbudowanym w ramach niniejszego zamówienia sprzęcie znajdującym się w innym budynku Zamawiającego. Ogólna specyfikacja istniejącej infrastruktury: - compute: serwery blade IBM w obudowie rack 19U, na potrzeby środowiska wirtualizacyjnego, - storage: macierz IBM obsadzona dyskami SAS, SATA, biblioteka taśmowa oraz serwery backupu z wbudowaną przestrzenią dyskową, - sieć: zbudowana na przełącznikach 1Gbps i 10Gbps, z wykorzystaniem łączenia portów w grupy oraz z zachowaniem redundancji połączeń. Dodatkowe uszczegółowienie realizacji tego wymagania	

znajduje się w rozdziale 4	
IaaS musi zapewnić zarządzanie środowiskiem storage większym niż 10 petabajtów.	
IaaS musi zapewnić zarządzanie oraz monitorowanie dużej (powyżej 10 000 hostów) sieci dla maszyn wirtualnych.	
IaaS musi zapewnić logiczną agregację maszyn wirtualnych w grupy (inaczej strefy/zones)	
IaaS musi zapewnić podział na regiony zapewniając osobną dedykowaną konfigurację dla compute, storage oraz warstwy sieci (w tym możliwość obsługi innego typu hypervisora oraz innej konfiguracji/sprzętu dla storage i sieci) dla danego regionu. Region musi zapewnić także osobny API endpoints, wspierając zarazem wspólne uwierzytelnianie użytkowników w oparciu o SSO oraz wspólny panel administracyjny dla wszystkich regionów.	
IaaS musi zapewnić definiowanie domyślnego hypervisor dla maszyn wirtualnych (cloud/vps) uruchamianych w danej grupie (w danej strefie/zone).	
IaaS musi zapewnić wiele organizacji (multitenant).	
IaaS musi zostać skonfigurowany w trybie wysokiej dostępności (High-Availability) - w szczególności dotyczy to każdego komponentu służącego do zarządzania i monitorowania infrastruktury IaaS.	
IaaS musi zapewnić pełną kontrolę każdego aspektu funkcjonalnego za pomocą API.	
IaaS musi wspierać pluginy dla każdej warstwy architektury (compute, storage, network) umożliwiające rozszerzanie funkcjonalności danej warstwy.	
IaaS musi zapewnić możliwość przydzielenia maszynom wirtualnym od 1 do co najmniej 128 procesorów wirtualnych.	
Licencja IaaS musi zapewniać możliwość ciągłego rozwoju	

Systemu przez Zamawiającego (własny zespół), dopuszczalne są dwa podejścia: przekazanie kodów źródłowych do całego Podsystemu IaaS wraz z przekazaniem autorskich praw majątkowych, bądź przekazanie kodów źródłowych do całego Podsystemu IaaS wraz z licencją umożliwiającą rozwój na własne potrzeby Zamawiającego.	
Kody źródłowe aplikacji muszą być przekazane w formie umożliwiającej rozwój. Kody źródłowe nie mogą być zaciemnione (nie mogą być poddane obfuskacji, z ang. obfuscation).	
Wraz z kodami źródłowymi wymagane jest przekazanie dokumentacji rozwojowej, w szczególności: opis architektury logicznej rozwiązania (modułów), wymaganych i użytych bibliotek i narzędzi, sposób budowania i dystrybucji aplikacji.	
IaaS musi zapewnić pełną obsługę następujących systemów operacyjnych (zarówno dla serwerów cloud, vps jak i serwerów dedykowanych): GNU Linux (w tym w szczególności: Debian GNU/Linux, RedHat Enterprise Linux, CentOS, Ubuntu, OpenSUSE, SUSE Enterprise Linux, Oracle Enterprise Linux), Microsoft Windows Server (2008, 2012 - wersje standard oraz enterprise), FreeBSD.	
IaaS musi obsługiwać systemy operacyjne w najnowszej stabilnej wersji dostępnej w dniu zakończenia etapu analizy.	
W ramach wdrożenia Podsystemu IaaS, Wykonawca musi przygotować obrazy dla maszyn wirtualnych (cloud/vps) oraz obrazy do instalacji serwerów dedykowanych dla wszystkich obsługiwanych systemów operacyjnych oraz umieścić je na usłudze biblioteki obrazów (image template/storage) IaaS.	
IaaS musi zapewnić zautomatyzowaną instalację/uruchomienie maszyn wirtualnych (cloud/vps) czy też serwerów dedykowanych bez ingerencji administratora dla każdego	

wymaganego systemu operacyjnego.	
IaaS musi umożliwić implementację/wdrożenie zarządzania wszystkimi funkcjonalnościami Podsystemu przez Podsystemy SelfService oraz Cloud API.	
IaaS musi zapewniać pełną integrację z Podsystemem Billing oraz udostępniać w sposób zautomatyzowany (poprzez API) wszystkie dane wymagane do prawidłowego działania i spełnienia wszystkich wymagań Podsystemu Billing.	
IaaS musi zapewnić możliwość obsługi wielu instancji różnych systemów operacyjnych na jednym serwerze fizycznym i musi wykorzystywać sprzętową wirtualizację zasobów.	
IaaS musi być niezależne od producenta platformy sprzętowej (poprawnie współpracować ze sprzętem dostarczanym przez wielu różnych producentów).	
IaaS musi umożliwiać przydzielenie większej ilości pamięci RAM dla maszyn wirtualnych niż fizyczne zasoby RAM serwera w celu osiągniecia maksymalnego współczynnika konsolidacji (tzw. memory overcommit).	
IaaS musi zapewnić możliwość monitorowania wykorzystania zasobów fizycznych infrastruktury wirtualnej jak i parametrów samych maszyn wirtualnych (cloud/vps) jak też serwerów dedykowanych w trybie real-time.	
IaaS musi zapewnić możliwość klonowania systemów operacyjnych wraz z ich pełną konfiguracją i danymi.	
IaaS musi umożliwiać zastosowanie w serwerach fizycznych dowolnej ilości procesorów o dowolnej ilości rdzeni.	
IaaS musi mieć możliwość uruchamiania fizycznych serwerów z centralnie przygotowanego obrazu poprzez protokół PXE	
IaaS musi umożliwiać automatyczne równoważenie obciążenia serwerów fizycznych pracujących jako platforma dla infrastruktury wirtualnej.	

IaaS musi pozwalać na definiowanie wzorców (typów) serwerów wirtualnych (cloud/vps) abstrahując od parametrów serwerów, np. typ „Large", „Medium", itp.. Wzorce mają wyjść naprzeciw oczekiwaniu, aby użytkownicy/klienci zamawiali podobne konfiguracje maszyn, unikając dużej fragmentacji zestawianych serwerów. Definicja wzorca serwera musi uwzględniać ogólne i szczegółowe parametry, np. dla serwera: CPU lub vCPU, pamięć RAM, przestrzeń dyskowa, adresy IPv4 i IPv6, VLANy, sieci prywatne, sieci dostępowe VPN.	
IaaS musi zapewnić uwierzytelnianie użytkowników w oparciu o SSO.	
IaaS musi zapewnić zarządzanie uprawnieniami dla użytkowników w oparciu o SSO.	
IaaS musi pozwalać na definiowanie praw dostępu do zasobów wchodzących w skład chmury dla ról/użytkowników. Definicje muszą obejmować co najmniej: dostęp do specyficznych zasobów (np. stref, sieci), maksymalne liczby poszczególnych typów zasobów (CPU, serwery, pamięć, przestrzeń dyskowa, sieci, adresy IP, VPN).	
IaaS musi pozwalać na definiowanie Usług na potrzeby budowania oferty Usług zachowując pełny cykl życia Usługi: • rejestrację Usługi • autoryzację Usługi • zamawianie i akceptacja Usługi • rezerwacja zasobów • uruchamianie Usługi • modyfikacje parametrów • zarządzanie Usługami • rezygnacja z Usługi • zwalnianie zasobów.	

IaaS musi pozwalać na przypisanie przygotowanych modeli kosztów do poszczególnych zasobów i ról/użytkowników.	
IaaS musi umożliwiać zdefiniowanie maksymalnej ilości jednostek do wykorzystania w ramach danej Usługi (np. 10GB transferu danych dla danego serwera cloud/vps).	
IaaS musi wspierać limitowanie wszystkich zasobów sprzętowych przydzielonych do danej instancji wirtualnej.	
Szczegółowe wymagania dla wdrożenia warstwy/Usługi chmury obliczeniowej (compute)	
IaaS musi zapewnić wsparcie następującej listy hypervisor: Microsoft Hyper-V, Citrix XenServer, Xen, KVM, VMWware ESXi, LXC, QEMU, Docker, Baremetal (czyli kontrolę maszyny fizycznej analogicznie jak w przypadku maszyny wirtualnej).	
IaaS dla następującej grupy hypervisorów musi zapewnić operacje: uruchomienia, restartu i zatrzymania: LXC, Baremetal, Docker.	
IaaS dla następującej grupy hypervisorów musi zapewnić następujące operacje dla maszyny wirtualnej działającej pod kontrolą danego hypervisora: uruchomienia, restartu, zatrzymania, resize (zmiana parametrów maszyny wirtualnej), suspend, reasume: Xen, KVM, QEMU, Hyper-V, ESXi.	
IaaS musi umożliwiać przenoszenia maszyn wirtualnych (cloud/vps) w czasie ich pracy (bez zatrzymania działania) pomiędzy serwerami fizycznymi (tzw. live migration), dla następujących Hypervisorów: KVM, XEN, XenServer, Hyper-V, QEMU.	
IaaS musi umożliwiać instancjonowanie maszyn wirtualnych cloud/vps zapewniając automatyczne skalowanie liczby procesorów i pamięci RAM.	
Instancje maszyn wirtualnych cloud/vps muszą mieć możliwość rozliczania w cyklu minutowym.	

IaaS musi udostępnić dla maszyn wirtualnych (vps/cloud) zdalny tryb/system rescue czyli system awaryjnego dostępu do serwera w celu złokalizowania i usunięcia usterek.	
Szczegółowe wymagania dla wdrożenia warstwy/Usługi serwery dedykowane	
IaaS musi udostępniać w sposób zautomatyzowany zdalną konsolę zarządzania sprzętem dla klientów końcowych (użytkowników) - czyli zdalna klawiatura, ekran i mysz, które pozwalają na dostęp do ekranu klawiszy/myszy serwera za pomocą bezpiecznego połączenia.	
IaaS musi udostępnić dla serwerów dedykowanych zdalny tryb/system rescue czyli system awaryjnego dostępu do serwera w celu zlokalizowania i usunięcia usterek.	
IaaS musi umożliwiać zdalny restart serwera dedykowanego dla maszyn dedykowanych.	
IaaS musi udostępnić możliwość zdalnego wyłączenia i ponownego uruchomienia serwera w dowolnej chwili.	
IaaS dla serwerów dedykowanych musi udostępnić poza jądrem dostarczanym domyślnie na dysku twardym, możliwość uruchomienia maszyny z sieci z jądrem przygotowanym przez Zamawiającego lub Wykonawcę (uruchomienie serwera dedykowanego z innym jądrem po sieci).	
Szczegółowe wymagania dla wdrożenia warstwy/Usługi danych (storage)	
IaaS musi świadczyć usługę repozytorium obrazów maszyn wirtualnych (image storage).	
IaaS musi zapewnić następujące typy obrazów dyskowych: raw, vhd, vmdk, vdi, qcow2.	
IaaS musi umożliwiać szyfrowanie dysków systemów wirtualnych pod jego kontrolą. Hasła powinny być nadawane przez użytkownika/klienta platformy IaaS za pomocą: dedykowanych funkcjonalności SelfService, API bądź interfejsu CLI.	

Wdrożenie IaaS musi obejmować wdrożenie Usługi block storage dla Klientów (udostępnianie urządzeń blokowych rozpoznawane przez system operacyjny jako dysk).	
Usługa block storage musi umożliwiać instancjonowanie przestrzeni dyskowych o rozmiarze min. 50 TB automatycznie podłączanych do systemu operacyjnego (oczywiście ograniczonych fizyczną przestrzenią dostępną przez oferowany sprzęt storage).	
Usługa block storage musi umożliwiać zmianę rozmiaru (powiększanie i zmniejszanie) przestrzeni dyskowej.	
IaaS musi wspierać protokoły dostępowe iSCSI, FC oraz RADOS Block Device.	
Wdrożenie IaaS musi obejmować wdrożenie object storage, który musi zostać zaimplementowany z użyciem protokołu RADOS Object Storage w celu zapewnienia kompatybilności z usługą Amazon S3.	
Usługa object storage musi umożliwiać zapis plików i obiektów o rozmiarze nie mniejszym niż 16TB.	
Usługa object storage musi zapewniać mechanizm autoryzacji dostępu.	
Transmisja obiektów z usługi object storage musi być możliwa za pomocą szyfrowanego połączenia.	
Szczegółowe wymagania dla wdrożenia warstwy/Usługi sieci (networking)	
IaaS musi zapewniać tworzenie wirtualnych sieci prywatnych (wirtualna prywatna chmura) pomiędzy maszynami wirtualnymi, serwerami dedykowanymi danego klienta oraz w ramach instancji danej aplikacji PaaS - implementacja Usługi Cloud Private Network (będź inaczej Private Network as a Service).	
IaaS musi zapewniać kontrolowanie sieci na warstwie drugiej modelu OSI.	

IaaS musi zapewnić automatycznie nadawanie prywatnej adresacji: IPv4 oaz IPv6 oraz publicznej adresacji IPv4 oraz IPv6 dla wszystkich Usług obsługiwanych przez IaaS. W przypadku standardu IPv6 - jeżeli oferowane rozwiązanie nie obsługuje na etapie składania oferty tej funkcjonalności, musi oferować ją na etapie realizacji Projektu.	
IaaS musi zapewnić usługę IP failover, polegającą na możliwości przenoszenia publicznych adresów IP między różnymi instancjami Usług.	
IaaS musi zapewnić Load Balancer as a Service dla każdego Klienta. Usługa ta powinna zapewniać tworzenie dowolnej liczby instancji load balancera na poziome: TCP, UDP, HTTP, HTTPS oraz następujące algorytmy load balancingu: Round Robin Wagowany Round Robin (umożliwiający określenie wagi dla każdego node), Random losowy, Preferowana najmniejsza liczba połączeń - który będzie powodował kierowanie ruchu na node z najmniejszą liczbą połączeń, Wagowany najmniejsza liczba połączeń - analogiczny do preferowana najmniejsza liczba połączeń, natomiast pozwalający przypisać wagi do poszczególnych node. Usługa load balancing musi zapewniać także zarządzanie sesją użytkownika (session persistance): source IP - zarządzanie sesją za pomocą śledzenia adresu źródłowego w celu określenia node docelowego oraz HTTP Cookie - śledzenie sesji za pomocą HTTP cookies (jedynie dla HTTP load balancing). Dodatkowo musi umożliwić zarządzanie monitoringiem oraz opcjami automatycznego podłączania/odłączania load balancowanych nodów (interwał weryfikacji nodów, liczba prób, timeout).	
IaaS musi zapewnić terminację SSL na load balancerach.	
IaaS musi zapewnić Usługę: Firewall as a Service dla każdego	

Klienta. Jeżeli oferowane rozwiązanie nie obsługuje na etapie składania oferty tej funkcjonalności, musi oferować ją na etapie realizacji Projektu.	
IaaS musi zapewnić możliwość konfiguracji certyfikatu SSL na poszczególnych instancjach load balancerów.	
IaaS musi zapewnić Usługę: VPN as a Service (VPNaaS - VPN Site-to-Site) umożliwiającą tworzenie i zarządzanie dedykowanych sieci VPN do sieci każdego z Klientów (zarządzanych przez Klientów) za pomocą: ipsec VPN. Wykorzystana technologia musi umożliwiać połączenie się do sieci VPN darmowymi, wbudowanymi w system operacyjny bądź open source narzędziami (klientami) IPSEC dostępnymi na następujące systemy operacyjne: GNU Linux, w szczególności: Debian GNU/Linux (Wheezy i nowszy), Fedora Linux (18 i nowsza), OpenSUSE (12 lub nowszy), Ubuntu Linux (13.10 lub nowszy), Windows 7 (i nowszy), Mac OS X (10.8 i nowszy). IaaS musi zapewniać także automatyczne generowanie plików konfiguracyjnych dla wybranego narzędzia ustalonego wspólnie z Zamawiającym na etapie analizy.	
IaaS musi zapewnić segmentację VLAN sieci Klienta.	
Load balancery zarządzane przez IaaS muszą mieć możliwość konfigurowania przekazywanych nagłówków HTTP.	
IaaS musi zapewnić możliwość zarządzania konfiguracją Two-Way SSL na load balancerach.	
W przypadku włączenia usługi two-way ssl na load balancerze, IaaS musi zapewnić konfigurację przekazywania nagłówków certyfikatów SSL klienta nawiązującego połączenie (np. pola Common Name, email itd).	
IaaS musi zapewniać możliwość definiowania, które z Usług (np. serwerów cloud bądź dedykowanych) mają być	

podłączone do load balancer.	
IaaS musi zapewniać zarządzanie regułami firewall dla pojedynczego serwera cloud/vps/dedykowanego jak i całych sieci prywatnych i publicznych klienta. Jeżeli oferowane rozwiązanie nie obsługuje na etapie składania oferty tej funkcjonalności, musi oferować ją na etapie realizacji Projektu.	
IaaS musi zapewnić rezerwację i przydzielanie zasobów jak również ich rozliczanie (zbieranie metryk udostępnianych Podsystemowi Biling): * Wirtualnych w oparciu o systemy wirtualizacji zasobów, z możliwością definicji parametrów, w tym: - vCPU – ilość - RAM – ilość - Storage - ilość wg typów storage - IOPS – ilość - Pasmo (gwarantowane, niegwarantowane) - Transfer danych - ilość - wewnątrz Usług - do sieci Internet - z sieci Internet - VLAN - ilość - Adresy IP publiczne IPv4 oraz IPv6 - ilość - Load Balancer - ilość (wraz z typami zarządzania sesji oraz terminacji SSL) - Firewall - ilość - Sieci prywatne - ilość - instancje VPN - ilość oraz typ * Fizycznych bez względu na producenta, w tym: - serwerów o zdefiniowanych parametrach	

- wolumenów dyskowych - wydzielonych - przydzielanych dynamicznie (integracja z storage) - Sieci prywatne - ilość - instancje VPN - ilość oraz typ - Firewall - ilość - Load Balancer - ilość (wraz z typami zarządzania sesji oraz terminacji SSL) - Adresy IP publiczne IPv4 oraz IPv6 – ilość.	
Szczegółowe wymagania dla panelu administracyjnego i zarządzania przez Zamawiającego IaaS	
IaaS musi zapewnić centralny interfejs administracyjny, umożliwiający zmianę każdego parametru konfiguracyjnego za pomocą dedykowanego interfejsu WEB (cienki klient).	
Interfejs administracyjny IaaS musi zapewnić funkcjonalność wersjonowania parametrów konfiguracyjnych (wartość, datę wartości, osobę/administratora, która wprowadziła wartość, unikalne ID dla zmiany, komentarz dla wprowadzonych zmian) wraz z przywracaniem wybranej wersji parametru konfiguracyjnego z poziomu panelu.	
Panel administracyjny IaaS musi zapewnić możliwość zdefiniowania poziomów akceptacji wprowadzanych zmian.	
Interfejs administracyjny IaaS musi zapewnić wizualizację zmian w konfiguracji, uwzględniając: - system/serwer/Usługę, której zmiana dotyczy, - datę powstania zmiany - osobę, która dokonała zmiany, - diff w postaci graficznej reprezentujący zmianę w stosunku do poprzednich zmian, - komentarz dot. danej zmiany. Po akceptacji zmiany musi nastąpić automatyczne wdrożenie zaakceptowanej konfiguracji za pomocą mechanizmu	

orkiestrującego (provisioning).	
Panel administracyjny musi zapewnić mechanizm ról i uprawnień, który w szczególności musi zapewnić uprawnienia: - wprowadzanie zmian w plikach konfiguracyjnych danego zasobu IaaS, - nadawanie uprawnień do wprowadzania zmian dla całej grupy, - nadawanie uprawnień do uruchamiania danych konfiguracji na danym serwerze, usłudze, systemem - należy rozróżnić fakt wprowadzania oraz uruchamiania danych konfiguracji.	
Panel administracyjny musi w trybie real-time wizualizować postęp jak i operacje (komunikaty usług wprowadzających zmiany) wykonywane podczas wprowadzanych zmian.	
Interfejs administracyjny musi zapewnić powiadomienie (notyfikację) wybranych administratorów o fakcie pojawienia się konfiguracji gotowych do wdrożenia (zarządzanie grupami zgodnie z wymaganiami dla wszystkich Podsystemów odbywać będzie się przez LDAP).	
Notyfikacje muszą być wysyłane na adresy email administratorów.	
Treści komunikatów będą mogły być edytowane za pomocą formularza na stronie Podsystemu i muszą być zapisywane jako szablony.	
Panel administracyjny musi umożliwić wyświetlanie na żywo logów wszystkich wykonywanych operacji.	
Panel musi udostępniać możliwość przeszukiwania i filtrowania logów wykonywanych operacji z uwzględnieniem uprawnień do nich.	
Panel administracyjny musi udostępniać możliwość rollback (przywrócenia poprzedniego stanu) - czyli uruchomienia konfiguracji, instalacji/aktualizacji poprzedniej wersji z	

systemu kontroli wersji.	
Panel administracyjny musi umożliwiać przetestowanie wprowadzonych zmian w konfiguracji z wykorzystaniem środowiska integracyjnego zbudowanego w ramach Projektu.	
Panel administracyjny musi zapewniać możliwość zarządzania i monitorowania Usług IaaS wszystkich klientów (oraz ich parametrów).	

21. PaaS

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.3 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.3 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Pozostałe informacje:

Wymagana funkcjonalność:	Oferowana funkcjonalność:
PaaS musi stanowić przyjazne, elastyczne, skalowalne i bezpieczne środowisko dla aplikacji SaaS.	
PaaS musi zapewnić: • cykl życia aplikacji • pakiety deploymentowe (Platform Deployment Package) • zasoby (Resources) zgodnie ze specyfikacją Cloud Application Mangement for Platforms (https://www.oasis-open.org/committees/download.php/47278/CAMP-v1.0.pdf).	
Licencja PaaS musi zapewniać możliwość ciągłego rozwoju Systemu przez Zamawiającego (własny zespół), dopuszczalne są dwa podejścia: przekazanie kodów źródłowych do całego Podsystemu PaaS wraz z przekazaniem autorskich praw majątkowych, bądź przekazanie kodów źródłowych do całego Podsystemu PaaS wraz z licencją umożliwiającą rozwój na własne potrzeby Zamawiającego.	
Kody źródłowe aplikacji muszą być przekazane w formie umożliwiającej rozwój. Kody źródłowe nie mogą być zaciemnione (nie mogą być poddane obfuskacji, z ang. obfuscation).	
Wraz z kodami źródłowymi wymagane jest przekazanie dokumentacji rozwojowej, w szczególności: opis architektury logicznej rozwiązania (modułów), wymaganych i użytych bibliotek i narzędzi, sposób budowania i dystrybucji aplikacji.	
PaaS musi integrować się bezpośrednio z IaaS oraz wykorzystywać infrastrukturę zarządzaną przez IaaS w celu zarządzania, monitorowania oraz automatycznego skalowania środowiska IaaS.	

Każda instancja aplikacji bądź bazy danych uruchamianej na Podsystemie PaaS musi działać na dedykowanym kontenerze w oparciu o wirtualizację na poziomie systemu operacyjnego GNU/Linux (właściwy system operacyjny zostanie wybrany i uzgodniony z Zamawiającym podczas etapu analizy).	
PaaS musi zapewnić automatyzację zarządzania, skalowania (w tym auto-skalowanie) i monitorowania środowiska IaaS dla serwerów aplikacyjnych, baz danych przeznaczonych dla aplikacji Klienta.	
PaaS musi zapewnić automatyzację zarządzania, skalowania (w tym auto-skalowanie) i monitorowania osadzonych na nim aplikacji oraz baz danych.	
PaaS musi zapewnić funkcjonalność konfiguracji reguł auto-skalowania dla osadzanych aplikacji oraz baz danych.	
PaaS musi zapewnić automatyczny provisioning, monitoring i zarządzanie środowisk aplikacyjnych oraz bazodanowych za pomocą Podsystemu SelfService.	
PaaS musi zapewnić automatyczny provisioning, monitoring i zarządzanie środowisk aplikacyjnych oraz bazodanowych za pomocą API.	
PaaS musi zapewnić automatyczny provisioning, monitoring i zarządzanie środowisk aplikacyjnych oraz bazodanowych za pomocą dedykowanej aplikacji CLI.	
PaaS musi zapewnić automatyczny provisioning i zarządzanie środowisk aplikacyjnych oraz bazodanowych za pomocą systemu VCS (system kontroli wersji) GIT.	
PaaS musi posiadać architekturę modułową oraz zapewnić wsparcie wielu języków programowania, w szczególności następujących języków: Java, Python, PHP. Zamawiający pisząc "zapewnić wsparcie" ma na myśli umożliwić uruchamianie, zarządzanie, monitoring oraz auto-skalowanie	

aplikacji napisanych w danym języku programowania.	
PaaS musi zapewnić wsparcie wielu wersji danego języka oprogramowania (dana aplikacja może działać na wybranej wersji danego języka oprogramowania).	
PaaS musi zapewnić wsparcie wielu wersji frameworków/bibliotek danego języka oprogramowania (dana aplikacja może działać na wybranej wersji danego framework/biblioteki oprogramowania).	
Architektura PaaS powinna w przyszłości (bez jej modyfikacji) umożliwiać rozszerzanie funkcjonalności Podsystemu o kolejne języki programowania, frameworki, systemy bazodanowe a także umożliwiać proste (za pomocą modułów/pluginów) rozszerzenie Podsystemu o nowe technologie, narzędzia i usługi (np. rozproszonych i asynchronicznych systemów zarządzania kolejkami zadań)	
Architektura PaaS musi zapewnić jego rozszerzenie o wsparcie nowych bibliotek/frameworków programowania danego języka. Zamawiający pisząc "zapewnić wsparcie" ma na myśli umożliwić uruchamianie, zarządzanie, monitoring oraz auto-skalowanie aplikacji napisanych w danym framework/bibliotece danego języku programowania.	
Architektura PaaS musi zapewnić jego rozszerzenie o wsparcie nowych Usług bazodanowych (w tym usługi bazodanowe SQL i noSQL). Zamawiający pisząc "zapewnić wsparcie" ma na myśli umożliwić uruchamianie (w tym tworzenie odpowiedniej schemy oraz import wskazanych danych), zarządzanie, monitoring oraz auto-skalowanie instancji baz danych w oparciu o dany system bazodanowy oraz możliwość automatycznego konfigurowania aplikacji działającej pod kontrolą Podsystemu PaaS aby wykorzystywała daną instancję bazy danych.	

Musi być dostarczona dokumentacja oraz przykład (szablon) w jaki sposób tworzyć moduły/rozszerzenia PaaS w celu zapewnienia wsparcia dla kolejnych (nie obsługiwanych jeszcze) języków programowania.	
Musi być dostarczona dokumentacja oraz przykład (szablon) w jaki sposób tworzyć moduły/rozszerzenia PaaS w celu zapewnienia wsparcia przykładowego framework dla danych języków programowania.	
Musi być dostarczona dokumentacja oraz przykład (szablon) w jaki sposób tworzyć moduły/rozszerzenia PaaS w celu zapewnienia wsparcia przykładowego serwera aplikacyjnego dla danego języka/framework.	
Musi być dostarczona dokumentacja oraz przykład (szablon) w jaki sposób tworzyć moduły/rozszerzenia PaaS w celu zapewnienia wsparcia dla kolejnych (nie obsługiwanych jeszcze) języków systemów baz danych.	
PaaS musi zapewnić wsparcie następujących frameworków języka Java: Sprint, Play!.	
PaaS musi zapewnić wsparcie następujących frameworków języka Python: Django, Flask.	
PaaS musi zapewnić wsparcie następujących frameworków języka PHP: CakePHP, Symphony.	
PaaS musi zapewnić wsparcie następujących serwerów aplikacyjnych dla języka Java: JBoos, Tomcat.	
PaaS musi zapewnić wsparcie następujących serwerów WSGI języka Python: Gunicorn, uWSGI.	
PaaS musi zapewnić wsparcie następujących frameworków języka PHP: CakePHP, Symphony.	
PaaS musi zapewnić wsparcie uruchamiania aplikacji języka PHP za pomocą oprogramowania: Apache2 wraz z mod_php oraz za pomocą PHP FastCGI Process Manager (FPM) wraz z	

proxy NGINX.	
PaaS musi zapewnić wsparcie dla uruchamiania aplikacji języka PHP na wydzielonym koncie użytkownika, przypisanym do wirtualnego hosta (mechanizm SuExec).	
PaaS musi zapewnić wsparcie następującej platformy bazodanowej SQL: PostgreSQL, MySQL.	
PaaS musi zapewnić wsparcie następującej platformy bazodanowej noSQL: MongoDB.	
PaaS musi zapewniać pełną automatyzację provisioningu i deploymentu aplikacji automatycznie konfigurując aplikację do działania.	
PaaS musi zapewniać pełną automatyzację provisioningu i deploymentu aplikacji automatycznie konfigurując i podłączając bazę danych aplikacji oraz samą aplikację z bazą danych.	
PaaS musi zapewnić pełną kontrolę każdego aspektu funkcjonalnego za pomocą API.	
PaaS musi zapewniać możliwość dostarczenia metryk dla Podsystemu Biling w celu rozliczania aplikacji działających na środowisku PaaS w modelu Pay as you go - czyli za faktycznie wykorzystane zasoby (CPU, RAM, przestrzeń dyskowa, I/O, wykorzystanie (transfer) danych wyjściowych, wykorzystanie (transfer) danych wejściowych, liczba instancji, czas działania, liczba obsłużonych połączeń).	
Musi być stworzona kompleksowa dokumentacja Podsystemu PaaS wraz z przykładami użycia dla każdego języka, framework oraz bazy danych.	
Muszą być dostarczone szkieletowe/przykładowe aplikacje dla każdego języka i framework (w połączeniu z każdą bazą danych) w celu ułatwienia bootstrapowania nowych aplikacji.	
Dokumentacja API oraz przykłady aplikacji muszą być	

przygotowane w języku polskim i angielskim.	
PaaS musi umożliwiać zdefiniowanie w dowolnym momencie liczbę instancji danej aplikacji. Zmiana liczby instancji musi powodować automatyczne uruchomienie bądź wyłączenie danej instancji.	
W przypadku zdefiniowania (automatycznie podczas skalowania bądź ręcznie) drugiej bądź kolejnej instancji danej aplikacji, musi być automatycznie podniesiona instancja load balancer'a oraz muszą być podłączone do niego wszystkie instancje aplikacji. Innym możliwym sposobem realizacji jest uruchamianie instancji load balancer od razu dla każdej nowej aplikacji a następnie podłączanie do niego kolejnych instancji aplikacji.	
Musi być możliwe dostarczenie definicji uruchomienia aplikacji na środowisku PaaS (w tym bazy danych) w jednym pliku konfiguracyjnym dołączonym do źródeł aplikacji - tzw. plik Manifestu.	
Podczas uruchamiania instancji aplikacji na środowisku, PaaS powinien na podstawie pliku Manifestu w sposób zautomatyzowany wykryć typ uruchamianej aplikacji (język, framework).	
Plik Manifestu musi umożliwiać skonfigurowanie typu aplikacji (język oraz wersja języka, framework oraz jego wersja, sposób obsługi instancji danej aplikacji), typ bazy danych, inicjalne dane do wczytania do bazy, liczbę instancji aplikacji, liczbę instancji baz danych, sposób działania load balancera (w szczególności sesji HTTP), konfigurację SSL load balancera, nazwę domeny aplikacji, port TCP/IP (domyślnie 80).	
Plik Manifestu musi być plikiem tekstowym.	
Plik Manifestu musi być w jednym z formatów: YAML, JSON.	

Musi być możliwość zalogowania się za pomocą SSH na kontener, na którym działa dana instancja aplikacji.	
W przypadku uruchomienia więcej niż jednej instancji danej aplikacji PaaS (maksymalnie 64 instancje) musi zapewnić poprawną obsługę sesji użytkowników oraz współdzielonych zasobów (np. danych na dyskach).	
W przypadku uruchomienia więcej niż jednej instancji danej aplikacji PaaS (maksymalnie 64 instancje) musi zapewnić automatyczne instancjonowanie sieci prywatnej (wirtualna prywatna chmura - private cloud network) oraz instancjonowanie dostępu do tejże sieci prywatnej za pomocą instancji VPN (instancja Usługi VPN as a Service).	
Szczegółowe wymagania dla panelu administracyjnego i zarządzania przez Zamawiającego PaaS	
PaaS musi zapewnić centralny interfejs administracyjny, umożliwiający zmianę każdego parametru konfiguracyjnego za pomocą dedykowanego interfejsu WEB (cienki klient).	
Interfejs administracyjny PaaS musi zapewnić funkcjonalność wersjonowania parametrów konfiguracyjnych (wartość, datę wartości, osobę/administratora, która wprowadziła wartość, unikalne ID dla zmiany, komentarz dla wprowadzonych zmian) wraz z przywracaniem wybranej wersji parametru konfiguracyjnego z poziomu panelu.	
Panel administracyjny PaaS musi zapewnić możliwość zdefiniowania poziomów akceptacji wprowadzanych zmian.	
Interfejs administracyjny PaaS musi zapewnić wizualizację zmian w konfiguracji, uwzględniając: - aplikację/system/serwer/Usługę, której zmiana dotyczy, - datę powstania zmiany - osobę, która dokonała zmiany, - diff w postaci graficznej reprezentujący zmianę w stosunku do poprzednich zmian,	

- komentarz dot. danej zmiany. Po akceptacji zmiany musi nastąpić automatyczne wdrożenie zaakceptowanej konfiguracji za pomocą mechanizmu orkiestrującego (provisioning).	
Panel administracyjny musi zapewnić mechanizm ról i uprawnień, w szczególności następujące uprawnienia: - wprowadzanie zmian w plikach konfiguracyjnych danego zasobu PaaS, - nadawanie uprawnień do wprowadzania zmian dla całej grupy, - nadawanie uprawnień do uruchamiania danych konfiguracji na danym serwerze, usłudze, systemie - należy rozróżnić fakt wprowadzania oraz uruchamiania danych konfiguracji.	
Panel administracyjny musi w trybie real-time wizualizować postęp jak i operacje (komunikaty usług wprowadzających zmiany) wykonywane podczas wprowadzanych zmian.	
Interfejs administracyjny musi zapewnić powiadomienie (notyfikację) wybranych administratorów o fakcie pojawienia się konfiguracji gotowych do wdrożenia (zarządzanie grupami zgodnie z wymaganiami dla wszystkich Podsystemów odbywać będzie się przez LDAP).	
Notyfikacje muszą być wysyłane na adresy e-mail administratorów.	
Treści komunikatów będą mogły być edytowane za pomocą formularza na stronie Podsystemu i muszą być zapisywane jako szablony.	
Panel administracyjny musi umożliwić wyświetlanie na żywo logów wszystkich wykonywanych operacji.	
Panel musi udostępniać możliwość przeszukiwania i filtrowania logów wykonywanych operacji z uwzględnieniem uprawnień do nich.	

Panel administracyjny musi udostępniać możliwość rollback (przywrócenia poprzedniego stanu) - czyli uruchomienia konfiguracji, instalacji/aktualizacji poprzedniej wersji z systemu kontroli wersji.	
Panel administracyjny musi umożliwiać przetestowanie wprowadzonych zmian w konfiguracji z wykorzystaniem środowiska integracyjnego zbudowanego w ramach Projektu.	
Panel administracyjny musi zapewniać możliwość zarządzania i monitorowania Usług PaaS wszystkich Klientów (oraz ich parametrów)	

22. Cloud API

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.4 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.4 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	Oferowana funkcjonalność:
Cloud API musi zapewnić warstwę abstrakcji, która pozwoli w spójny, łatwy do wykorzystania sposób udostępnić wewnętrzne interfejsy API wszystkich Podsystemów oraz funkcjonalności wchodzących w skład Systemu dla użytkowników zewnętrznych - w szczególności musi udostępniać przez API funkcjonalności dla Klientów wszystkich Usług opartych o IaaS, PaaS, informacje nt. rozliczeń oraz faktur (Billing) oraz pełną obsługę konta Klienta (organizacji Klienta) w ramach Podsystemu (SSO).	
Cloud API musi realizować wymagania Rozporządzenia Ministra Nauki i Informatyzacji z dnia 19 października 2005 r. w sprawie testów akceptacyjnych oraz badania oprogramowania interfejsowego.	
Wymagane jest dostarczenie dokumentacji Cloud API wraz z przykładami dla każdej metody (w języku angielskim).	
Muszą być dostarczony opis wszystkich kodów oraz statusów odpowiedzi każdej metody API (w języku angielskim).	
Udostępniany endpoint API musi być zabezpieczony certyfikatem SSL (dostarczonym przez Zamawiającego).	
Cloud API powinno stanowić spójny interfejs API - a zatem, jeżeli Wykonawca planuje wdrożyć Podsystemy, które posiadają różne typy danych w wywołaniach API, Cloud API musi zapewnić jeden wybrany typ danych dla wszystkich wywołań.	
Cloud API oprócz endpointów API musi dostarczyć	

aplikację/narzędzie działające w linii poleceń (CLI), instalowane przez użytkownika na jego komputerze i działające na najpopularniejszych systemach operacyjnych (Linux, Windows 7 (i nowsze), Mac OS X 10.8 (i nowszy), FreeBSD), zapewniające pełną funkcjonalność Cloud API z linii poleceń (metody powinny być odwzorowane w odpowiednich argumentach i opcjach). Aplikacja taka umożliwi np. administratorom bądź developerom wykonywać operacje na platformie bez potrzeby integracji z API a za pomocą gotowego narzędzia, które można użyć/wykorzystać we własnych aplikacjach/skryptach. Aplikacja ta jest aplikacją CLI pozwalającą zarządzać i monitorować (jak i weryfikować rozliczenia) wszystkich Usług dostarczanych w ramach Systemu i stanowi kolejny (oprócz SelfService i API) interfejs komunikacji z platformą.	
Cloud API musi być zintegrowane z rozwiązaniem Apache DeltaCloud.	
Cloud API w przypadku obsługi klienta/użytkownika danego Partnera musi ukryć metody dot. rozliczeń, jako że rozliczenia tego typu klientów (klientów Partnerów) będą dokonywane przez tychże partnerów a nie Zamawiającego.	
Licencja Cloud API musi zapewniać możliwość ciągłego rozwoju Systemu przez Zamawiającego (własny zespół), a dopuszczalne są dwa podejścia: przekazanie kodów źródłowych do całego Podsystemu Cloud API wraz z przekazaniem autorskich praw majątkowych, bądź przekazanie kodów źródłowych do całego Podsystemu Cloud API wraz z licencją umożliwiającą rozwój na własne potrzeby Zamawiającego.	
Kody źródłowe aplikacji muszą być przekazane w formie umożliwiającej rozwój. Kody źródłowe nie mogą być	

zaciemnione (nie mogą być poddane obfuskacji, z ang. obfuscation).	
Wraz z kodami źródłowymi wymagane jest przekazanie dokumentacji rozwojowej, w szczególności: opis architektury logicznej rozwiązania (modułów), wymaganych i użytych bibliotek i narzędzi, sposób budowania i dystrybucji aplikacji.	

23. SelfService

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.5 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.5 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	Oferowana funkcjonalność:
SelfService musi umożliwić Klientowi kompleksowe monitorowanie (wszystkich dostępnych przez daną Usługę KPI/metryk), zarządzanie Usługami oraz ich parametrami w ramach oferty biznesowej (w szczególności Usługami IaaS: serwery dedykowane, VPS, Cloud, Storage as a Service, Firewall as a Service, Load Balancer as a Service oraz Usługami PaaS). Musi służyć Klientom jako centralna i główna aplikacja do zarządzania, rozliczania oraz monitorowania wszystkich funkcjonalności udostępnianych przez IaaS, PaaS i Billing w zakresie obsługi produktów i Usług wykorzystywanych przez danego Klienta, uwzględniając przy tym posiadane przez niego uprawnienia dostępu.	
Licencja SelfService musi zapewniać możliwość ciągłego rozwoju Systemu przez Zamawiającego (własny zespół), a dopuszczalne są dwa podejścia: przekazanie kodów źródłowych do całego Podsystemu SelfService wraz z przekazaniem autorskich praw majątkowych, bądź przekazanie kodów źródłowych do całego Podsystemu SelfService wraz z licencją umożliwiającą rozwój na własne potrzeby Zamawiającego.	
Kody źródłowe aplikacji muszą być przekazane w formie umożliwiającej rozwój. Kody źródłowe nie mogą być zaciemnione (nie mogą być poddane obfuskacji, z ang. obfuscation)	
Wraz z kodami źródłowymi wymagane jest przekazanie dokumentacji rozwojowej, w szczególności: opis architektury logicznej rozwiązania (modułów), wymaganych i użytych bibliotek i narzędzi, sposób budowania i dystrybucji aplikacji.	
SelfService musi zapewnić dla każdej Usługi oferowanej przez	

IaaS oraz PaaS dedykowany obszar Podsystemu (np. zakładkę) zaprojektowany w celu jak najwygodniejszej obsługi wszystkich funkcjonalności oferowanych przez Usługę.	
SelfService musi zapewnić dla każdej z Usług dedykowaną funkcjonalność monitorowania i wizualizacji wszystkich metryk/KPI udostępnianych przez daną funkcjonalność IaaS bądź PaaS.	
SelfService musi zapewnić dla każdej z Usług stan rozliczenia (kwotowy oraz jednostki rozliczeniowe) danej Usługi wraz z możliwością filtrowania oraz eksportowania.	
SelfService musi zapewnić dedykowany obszar interfejsu agregujący wszystkie metryki/KPI w celu monitorowania zbiorczo stanu wszystkich Usług IaaS i PaaS. Dane powinny być wyświetlane w ujęciu tabelarycznym oraz za pomocą wykresów w ujęciu godzinnym, dniowym, tygodniowym, miesięcznym i rocznym. Musi być możliwość eksportu wyświetlanych danych.	
SelfService musi zapewnić dedykowany obszar interfejsu użytkownika dla historii rozliczeń, płatności, faktur i umów w ramach integracji z Podsystemem Biling.	
SelfService musi zapewnić dedykowany obszar interfejsu użytkownika umożliwiający przeglądanie, przeszukiwanie i filtrowanie logów każdej z Usług - zarówno wszystkich Usług zbiorczo jak i poszczególnych Usług osobno.	
SelfService musi realizować kontrolę dostępu do poszczególnych funkcjonalności na podstawie roli użytkownika określonej przez SSO.	
Zarządzanie Usługą: SelfService musi wyświetlać na bieżąco stan zleconego zadania dla Usługi (np. stworzenie serwera wirtualnego IaaS, deployment aplikacji na PaaS, itd.) za pomocą informacji w postaci procentowej zmieniającej się bez	

przeładowania strony (AJAX).	
Zarządzanie Usługą: SelfService musi wyświetlać na bieżąco logi oraz wykonywane operacje zleconego zadania dla Usługi (np. stworzenie serwera wirtualnego IaaS, deployment aplikacji na PaaS, itd.) bez przeładowania strony (AJAX).	
SelfService musi w sposób czytelny i widoczny wyświetlać wszystkie parametry danej Usługi (np. status danej Usługi, adres IPv4 i IPv6 serwera cloud, adres URL pod którym dostępna jest aplikacja PaaS po jej uruchomieniu, itp.).	
Wszystkie krytyczne operacje wpływające na działanie Usługi muszą być (mimo zalogowania) potwierdzane hasłem zalogowanego użytkownika (jeżeli ma uprawnienie do danej operacji) lub hasłem jednorazowym - jeśli jest wykorzystywane przez danego użytkownika. Przykładem krytycznych operacji Usług jest: usunięcie Usługi, zmiana hasła root/Administratora serwera, restart działającej Usługi, itp.	
SelfService musi implementować przeglądanie/filtrowanie i nawigację typu faceted (tzw. nawigacja fasetowa) zapewniając Klientom łatwą nawigację oraz filtrowanie Usług oraz ich parametrów.	
SelfService musi umożliwić przypisywanie tagów do usług w celu szybszego przeglądania/filtrowania dostępnych Usług.	
SelfService musi umożliwić zapisanie parametrów tworzonej Usługi jako szablonu, do ponownego użytku oraz umożliwić tworzenie nowych Usług z wykorzystaniem zapisanych szablonów.	
SelfService musi udostępniać wizualizację real time logów danej Usługi bez przeładowania strony (AJAX).	
SelfService musi zapewnić konfigurowalny mechanizm powiadomień mailowych nt. wybranych operacji na Usłudze bądź jej parametrach wraz z konfiguracją reguł powiadomień	

dla każdego z użytkowników osobno (np. notyfikacja email nt. restartu serwera, notyfikacja email nt. automatycznego stworzenia nowej instancji aplikacji podczas autoskalowania, itp).	
W przypadku łączenia się Usług (np. kilka serwerów wirtualnych podłączonych do load balancera, kilka instancji aplikacji osadzonej na PaaS podłączonych do instancji bazy danych) SelfService musi zapewnić reprezentację graficzną (w postaci diagramów) wraz z reprezentacją jakie Usługi w jaki sposób są połączone.	
SelfService musi zapewnić konfigurator (wizard) dla uruchomienia każdej z Usług, wizualizując liczbę ekranów konfiguracji/parametryzacji danej Usługi i procent zaawansowania.	
SelfService musi zapewnić pomoc kontekstową podczas zmiany każdego z parametrów Usługi.	
Wymagania dla Usług IaaS	
SelfService musi umożliwiać, po zakończonym provisioningu serwera (cloud/vps/dedykowanego/share-hosting), wyświetlenie za pomocą pop-up hasła dla użytkownika root (w przypadku serwerów Linux/FreeBSD) bądź administratora (dla serwerów Windows).	
SelfService musi umożliwić zarządzanie kluczami publicznymi SSH dla wszystkich typów serwerów uwzględniając nadane uprawnienia do serwerów.	
SelfService musi umożliwiać, po zakończonym provisioningu serwera (cloud/vps/dedykowanego/share-hosting), wysłanie na adres email użytkownika, który go stworzył hasła dla użytkownika root (w przypadku serwerów Linux) bądź administratora (dla serwerów Windows).	
SelfService musi udostępniać terminal maszyny wirtualnej	

cloud/vps w przeglądarce.	
SelfService musi udostępnić zdalną konsolę zarządzania sprzętem serwera dedykowanego w przeglądarce.	
SelfService musi udostępnić opcję zdalnej konsoli za pomocą protokołu VNC (serwery Linux) bądź Remote Desktop Connection (serwery Windows), umożliwiając definiowanie portu oraz adresów źródłowych IP, które mają dostęp do Usługi.	
Wymagania dla integracji z Podsystemem Billing oraz rozliczeń Usług	
SelfService musi zapewniać wyświetlanie obecnego stanu rozliczenia konta.	
SelfService musi udostępniać wygenerowane faktury (pro-forma, faktury oraz faktury korekt) dla danego Klienta (integracja z Podsystemem Billing).	
SelfService musi zapewniać automatyczne pobieranie płatności zgodnie ze zdefiniowanym trybem bilingowania (integracja z Podsystemem Billing).	
SelfService musi udostępniać historię rekordów rozliczeniowych, faktur oraz płatności (integracja z Podsystemem Billing).	
SelfService musi umożliwić płatność za pomocą integracji z Podsystemem Billing.	
SelfService musi udostępnić historię prób pobrania płatności (jeżeli były nieudane) oraz czytelnie zakomunikować dlaczego Usługi/konto zostały zablokowane.	
SelfService musi w przypadku blokady Usługi (bądź konta - czyli wszystkich Usług) w przypadku braku płatności zablokować wszystkie udostępniane funkcjonalności, oprócz: - możliwości dokonania opłaty za Usługi - możliwości pobrania faktur	
Wymagania dla Usług PaaS	

SelfService dla każdej uruchomionej aplikacji musi zapewnić dedykowany obszar w interfejsie użytkownika (np. zakładkę).	
SelfService dla każdej bazy danych aplikacji musi zapewnić dedykowany obszar w interfejsie użytkownika (np. zakładkę).	
SelfService musi udostępniać możliwość tworzenia snapshotów wybranej bazy danych.	
SelfService musi udostępniać możliwość exportu wybranej bazy danych.	
SelfService musi udostępniać możliwość importu wybranej bazy danych.	
Wymagania dla Programu Partnerskiego	
SelfService musi identyfikować (integracja z SSO) czy dany Klient jest Partnerem (uczestnikiem Programu Partnerskiego), klientem Partnera czy też Klientem Zamawiającego.	
SelfService musi zapewnić Partnerom dedykowany obszar interfejsu użytkownika umożliwiający zarządzanie klientami/organizacjami/użytkownikami Partner (integracja z SSO), zarządzanie oraz monitorowanie wszystkich usług klientów Partnera.	
SelfService musi zapewnić Partnerom dedykowany obszar interfejsu użytkownika pobieranie rekordów bilingowych każdego klienta (integracja z Billing).	
SelfService w przypadku obsługi się klienta/użytkownika danego Partnera musi ukryć obszar rozliczeń, jako że rozliczenia tego typu klientów (klientów Partnerów) będą dokonywane przez tychże Partnerów a nie przez Zamawiającego.	

24. Billing

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.6 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.6 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
Billing musi zapewniać elastyczne rozliczanie wszystkich Usług oferowanych przez System bazując na określonych atomowych jednostkach metrycznych oferowanych przez daną Usługę.	

Billing musi integrować się z każdym Podsystemem Platformy oprogramowania, który odpowiedzialny jest za dostarczanie Usług dla klienta w celu aktywacji/deaktywacji Usług.	
Billing musi umożliwić modelowanie rozliczeń na jednostkach rozliczeniowych za pomocą reguł matematycznych i logicznych w taki sposób aby umożliwić tworzenie złożonych konfiguracji modeli rozliczeniowych.	
Billing musi umożliwiać definiowanie produktów rozliczeniowych za pomocą reguł matematycznych i logicznych w taki sposób aby umożliwić tworzenie złożonych konfiguracji Usług.	
Billing musi umożliwiać grupowanie Usług w grupy Usług.	
Billing musi mieć modułową budowę systemu akwizycji danych (umożliwiać pobieranie danych z różnych źródeł w różnych formatach).	
Billing musi umożliwiać naliczanie Usług jednorazowych.	
Billing musi umożliwiać naliczanie Usług abonamentowych.	
Billing musi umożliwiać naliczanie Usług w planie taryfowym Pay-As-You-Go (za faktycznie wykorzystane zasoby).	
Billing musi umożliwiać wsparcie dla rozliczeń typu post-paid oraz pre-paid.	
Billing musi umożliwiać konfigurowanie mnożników kosztowych.	
Billing musi umożliwiać definiowanie okresów rozliczeniowych dla każdego Klienta indywidualnie (z dokładnością do minut).	
Billing wraz z Podsystemem SSO muszą zapewnić wsparcie obsługi oraz zarządzania Partnerami, w tym: umożliwiać określenie globalnej stawkę rabatów dla wszystkich Usług dla Partnera, dedykowaną stawkę rabatu dla poszczególnych Usług dla wybranego Partnera oraz zarządzanie Partnerami jak i jego	

klientami, w szczególności umożliwiać zmianę poziomu partnerstwa.	
Billing musi zapewnić generowanie, udostępnianie i przechowywanie dla Partnerów rekordów rozliczeniowych wszystkich Usług wszystkich klientów danego partnera w formacie CSV.	
Billing musi umożliwiać definiowanie promocji określając: czas trwania promocji; Usługi uwzględnione w promocji; typ rabatu globalnie (dla wszystkich Usług) bądź dla każdego z Usług, typy promocji to: zniżkowa (obniżenie wartości Usług) wraz z definicją wartości obniżenia: procentowa lub kwotowa - bądź typ: zwiększenie salda (czyli przypisanie wartości kwotowej dla salda); definicję dla kogo promocja jest dostępna - czy dla wszystkich klientów wg. stażu; możliwość przypisania promocji do wybranych Partnerów.	
Billing musi umożliwiać rozliczenie w walutach: PLN, EUR, USD - zgodnie z polskim prawem.	
Billing musi umożliwiać automatyczną obsługę kursów walut w oparciu o kurs NBP.	
Billing musi umożliwiać wsparcie dla wielu walut w tym wymagane: PLN, EUR, USD.	
Billing musi umożliwiać definiowanie oraz rozliczenie Usług pakietowych (połączenie kilku Usług) w modelu jednorazowym.	
Billing musi umożliwiać definiowanie oraz rozliczenie Usług pakietowych (połączenie kilku Usług) w modelu abonamentowym.	
Billing musi umożliwiać definiowanie oraz rozliczenie Usług pakietowych (połączenie kilku Usług) w modelu Pay-As-You-Go.	
Billing musi umożliwiać definiowanie oraz rozliczenie Usług	

pakietowych (połączenie kilku Usług) w modelu mieszanym (np. część produktów w modelu abonamentowym a część w modelu jednorazowym, itp).	
Billing musi umożliwiać automatyczne przełączanie na wyższy bądź poziom cenowy czy też pakiet na podstawie zdefiniowanych warunków.	
Billing musi dopuszczać Usługi, pakiety bezpłatne.	
Billing musi umożliwiać definiowanie oraz rozliczenie promocji do Usług łączonych.	
Billing musi umożliwiać ręczne oraz automatyczne fakturowanie zgodnie z prawem Polski oraz Unii Europejskiej.	
Billing musi umożliwiać generowanie raportów dla rozliczeń w zadanym okresie czasowym, dla zadanych Usług, w postaci tabelarycznej, wykresów oraz exportu do pliku CSV.	
Billing musi zapewnić produkty oraz konta testowe.	
Billing musi obsługiwać importowanie stawek/planów cenowych z plików CSV.	
Billing musi obsługiwać nielimitowaną ilość Usług, promocji, Klientów.	
Billing musi oferować wsparcie dla masowej edycji rekordów.	
Billing musi umożliwiać dla poszczególnych Usług określanie rozpoczęcia i daty końcowej w której Usługa jest aktywna.	
Billing musi oferować wsparcie dla definiowania i generowania automatycznie raportów okresowych uwzględniających co najmniej: zamówienia/odnowienia/rezygnację z Usług, płatności/rozliczenia Usług.	
Billing musi oferować wsparcie do automatycznej wysyłki raportów drogą e-mail.	
Billing musi udostępniać pełne API umożliwiającego kontrolę/konfigurację oraz dostęp do wszystkich funkcjonalności, danych rozliczeniowych oraz raportów.	

Billing musi zapewnić integrację z Podsystemem SSO w celu uwierzytelniania i autoryzacji do zasobów oraz identyfikacji klientów.	
Billing musi wspierać grupowanie klientów wg ich klasyfikacji biznesowej oraz nadawanie im parametrów rozliczeniowych (np. inne ceny na poszczególne Usługi, rabaty globalne, limity sprzedawanych Usług). Źródłem klasyfikacji biznesowej klientów jest Podsystem SSO.	
Billing musi oferować wsparcie dla overbookingu jako podstawy przyznawania rabatów.	
Billing musi oferować wsparcie dla kontrolowanego wywłaszczania (niższego priorytetu usług) jako podstawy przyznawania rabatów.	
Billing musi oferować wsparcie analityczne dla ustalania cen każdego zasobu rozliczeniowego w ujęciu czasowym.	
Billing musi umożliwić określenie sposobu migracji Usług: - downgrade - z końcem okresu rozliczeniowego lub z datą powstania zdarzenia. Pozostałe środki powinny być zaksięgowane na poczet następnego okresu rozliczeniowego - upgrade - z końcem okresu rozliczeniowego lub z datą powstania zdarzenia. Jeśli użytkownik wybierze datę powstania zdarzenia wymagane jest pobranie opłaty (różnicy).	
Billing musi umożliwić definicję wspieranych rodzajów płatności dla każdej Usługi.	
Billing musi umożliwić definiowanie czy jest okres próbny (trial) oraz jak długo trwa dla każdej Usługi (liczba dni).	
Billing musi umożliwiać wsparcie dla różnych formatów daty (w szczególności: DDMMYYYY, YYYYMMDD, DD-MM-YYYY, YYYY-MM-DD, DD.MM.YYYY, YYYY.MM.DD) oraz wsparcie przy czym zapisu miesiąca (MM) cyframi rzymskimi oraz arabskimi.	

Billing musi wspierać wiele stawek podatku VAT wraz z definicją okresu/przedziału czasowego wsparcia dla danej stawki.	
Billing musi umożliwiać zdefiniowanie maksymalnej ilości jednostek do wykorzystania w ramach danego produktu np. 1GB RAM, 10GB transferu danych z określeniem stawki za dalsze wykorzystanie poza limitem np. 10zł/1GB transferu.	
Billing musi umożliwiać określenie przedziałów cenowych za zużyty zasób np. transfer danych do 10GB koszt 10zł/GB, powyżej 8zł/GB (mnożniki kosztowe).	
Billing musi umożliwiać przyznawanie darmowych jednostek w danej Usłudze za korzystanie z innych Usług.	
Billing musi umożliwiać tworzenie produktów systemowych - dodane dla każdego użytkownika i konfigurowalne wspólnie dla wszystkich np. darmowy adres e-mail.	
Billing musi umożliwiać określenie minimalnego zużycia czasowego/jednostek w modelu Prepaid (np. naliczone z góry 1h użycia Usługi).	
Billing musi umożliwiać weryfikację czy stan finansowy pozwala na włączenie Usługi i na jaki okres w modelu Prepaid.	
Billing musi zapewnić automatyczne procesowanie ponagleń płatności wraz z powiadomieniami e-mail klientów.	
Billing musi umożliwiać określanie okresów karencji, po których następuje zablokowanie Usługi oraz sposobów notyfikacji e-mail wraz z ich treścią.	
Billing musi umożliwiać określanie ilości prób i okresów czasu po jakich następuje ponowna próba pobrania opłaty (po nieudanej próbie).	
Billing musi wspierać wznawianie Usług zawieszonych.	
Billing musi umożliwiać konfigurację szablonów e-mail (ponaglenia, podziękowania za płatność) - możliwość	

zdefiniowania szablonu kodem HTML z wykorzystaniem predefiniowanych atrybutów określających różne elementy konta np. imię, nazwisko, wymaganą kwotę i termin płatności.	
Billing musi umożliwiać obsługę płatności: karty kredytowe, przelewy tradycyjne bazujące na subkontach, płatności online. System obsługi płatności zostanie ustalony wspólnie z Zamawiającym na etapie analizy.	
Billing musi umożliwiać tworzenie raportu uwzględniającego: - płatności danego miesiąca, wybranego okresu - przeterminowane płatności - statystyki zużycia zasobów danego miesiąca, wybranego okresu - raporty finansowo-księgowe (uwzględnione korekty).	
Billing musi umożliwić wznawianie Usług - np. wyłączonych z powodu braku płatności (możliwy nowy termin początkowy okresu rozliczeniowego).	
Billing musi umożliwić zmniejszenie/zwiększenie ceny Usługi dla konkretnego Klienta o daną kwotę.	
Billing musi umożliwić ustawienie ceny Usługi dla konkretnego Klienta na daną kwotę.	
Billing musi umożliwiać edycję zdarzenia rozliczeniowego (źle naliczona opłata za użycie) wraz zachowaniem historii zmian oraz automatyzacją tworzenia faktur korygujących.	
Billing musi umożliwiać przeksięgowanie płatności na inną należność (wraz ze śledzeniem historii takowych operacji).	
Billing musi umożliwiać zwrot nadpłaty lub przeksięgowanie na kolejną płatność (wraz ze śledzeniem historii takowych operacji). Operacja zwrotu środków obsługiwana będzie przez system płatności ustalonym wspólnie z Zamawiającym podczas fazy analizy.	
Billing musi umożliwiać automatycznie naliczanie odsetek.	

Billing musi zapewnić migrację Usług - np. ze względu na wycofanie z oferty danej Usługi.	
Billing musi umożliwić wystawianie faktur (w tym faktury korygujące) i not odsetkowych (w tym faktur elektronicznych).	
Billing musi umożliwiać automatyczną wysyłkę faktur i not odsetkowych oraz ich wysyłkę na żądanie za pomocą e-mail oraz korespondencji masowej (papierowej).	
Billing musi umożliwiać wystawienie, przesyłanie i przechowywanie faktur w formie elektronicznej zgodnie z obowiązującymi przepisami prawa.	
Billing musi umożliwić generowanie oraz przechowywanie dla każdego stworzonego dokumentu (faktura, faktura korygująca, nota odsetkowa, itd) obrazu elektronicznego w postaci plików PDF.	
Billing musi umożliwiać udostępnianie obrazów elektronicznych faktur dla Podsystemu SelfService, który będzie udostępniał je Klientom.	
Billing musi umożliwić generowanie oraz przechowywanie dla każdego stworzonego dokumentu (faktura, faktura korygująca, nota odsetkowa, itd) plików umożliwiających import owych dokumentów w programie Symfonia wersja min. Premium 2014.	
Billing musi umożliwiać konfigurację szablonów faktur.	
Billing musi zapewnić definiowanie ograniczeń kwotowych i ilościowych per Klient, Usługa bądź łącznie (np. aby jeden Klient nie mógł wykupić zdefiniowanej liczby Usług - limit).	
Billing musi zapewnić definiowanie limitu kredytowego dla rozliczeń w modelu post-paid.	
Billing musi zapewnić windykację (przypomnienia, wyłączanie Usług, wznawianie Usług).	
Billing musi zapewnić możliwość przyszłej	

modyfikacji/dostosowania Podsystemu za pomocą: API oraz pluginów/modułów.	
Billing musi zapewnić zmienność względem: pory dnia, dnia tygodnia i dnia miesiąca.	
Billing musi zapewnić funkcjonalność kuponów rabatowych, które identyfikują Usługę bądź promocję oraz umożliwiają wprowadzenie ograniczeń czasowych dla kuponu oraz ograniczeń dla wykorzystania kuponu przez Klientów w zadanym stażu oraz wybranych Partnerów.	
Billing musi umożliwić przedstawienie raportu zasobów rozliczeniowych (wraz z przychodami) w zadanym okresie czasowym w celu określenia rentowności Usług.	
Billing musi umożliwiać przedstawianie powyższych raportów przekładając je na Usługi nieaktywne.	

25. ESB

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.7 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.7 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
Każdy Podsystem komunikujący się po API bądź udostępniający API musi mieć zaimplementowany endpoint na ESB i komunikować się przez ESB. W uzasadnionych przypadkach Zamawiający może, na etapie opracowywania projektu technicznego, dopuścić bezpośrednią komunikację między Podsystemami.	
ESB musi posiadać mechanizm definiowania, implementacji, wdrażania i zarządzania usługami realizującymi dostęp do integrowanych Podsystemów.	
ESB zakłada istnienie usług prywatnych i publicznych. Usługi prywatne są dostępne jedynie w obrębie Systemu i nie mogą być bezpośrednio wywoływane przez Klientów Systemu.	
ESB musi posiadać mechanizm umożliwiający planowe i cykliczne uruchamianie usług Systemu. Zarządzanie planowanymi do uruchomienia usługami musi odbywać się w sposób spójny z jednego miejsca Systemu na zasadzie definiowania harmonogramu wywołań.	
ESB musi zapewniać funkcjonalność transformacji komunikatów i transportowania ich używając różnych	

protokołów: HTTP, HTTPS, JDBC, JMS, Web Services, AMQP, FTP, SSH, SOAP, REST/JSON.	
ESB musi zapewnić możliwość tworzenia konektorów do usług, umożliwiających łatwe podłączanie kolejnych usług w oparciu o konfigurację danego konektora.	
ESB musi zapewniać usługi zarówno synchroniczne jak i asynchroniczne.	
ESB musi zapewnić GUI WEB zapewniające funkcjonalności zarządzania oraz monitorowania rozwiązania.	
ESB musi dostarczać usługi translacji protokołów.	
ESB musi umożliwiać filtrowanie komunikatów na podstawie zawartości, przy wykorzystaniu parametrów definiowanych przez użytkownika.	
ESB musi umożliwiać trwałe przechowywanie komunikatów.	
ESB musi umożliwiać tworzenie architektury wyjątków, która może przechwytywać wyjątki, generować transakcje kompensacyjne i generować raporty o błędach.	
ESB musi zapewniać mechanizm transakcji XA (XA Transactions).	
ESB musi umożliwiać zachowanie integralności, niezaprzeczalności, poufności i autentyczności komunikacji.	
ESB musi zapewnić API do tworzenia nowych konektorów oraz wsparcia dla nowych nieobsługiwanych protokołów.	

26. SSO

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.8 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.8 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
Architektura i integracja z pozostałymi Podsystemami	
SSO musi stanowić jedyny mechanizm uwierzytelniania dla wszystkich Podsystemów dostępnych poprzez przeglądarkę internetową.	
SSO jako podstawowy protokół musi wykorzystywać oAuth w wersji 2.	
SSO musi stanowić centralny rejestr wszystkich użytkowników wykorzystywanych w ramach Systemu.	

SSO musi stanowić centralny rejestr informacji o przetwarzaniu i udostępnianiu danych osobowych, wymagany przez Ustawę o ochronie danych osobowych.	
Podsystemy wchodzące w skład Platformy oprogramowania muszą rejestrować w SSO zdarzenia związane z przetwarzaniem danych osobowych użytkowników w zakresie określonym wytycznymi Ustawy o ochronie danych osobowych i aktów wykonawczych do tej Ustawy.	
SSO musi umożliwiać Podsystemom przeprowadzenie dodatkowego uwierzytelnienia użytkownika przed wykonaniem istotnych z perspektywy bezpieczeństwa operacji.	
SSO musi umożliwiać zdefiniowanie listy dostawców usług (Service Provider), którzy mogą korzystać z SSO jako dostawcy tożsamości (Identity Provider).	
SSO musi umożliwiać zarządzanie wyglądem każdej podstrony za pomocą szablonów.	
SSO musi być zintegrowane z bramką SMS ustaloną na etapie przygotowywania projektu technicznego.	
LDAP	
W skład Podsystemu SSO musi wchodzić usługa katalogowa oparta o protokół LDAP.	
SSO musi przechowywać wszystkie dane o użytkownikach w usłudze katalogowej LDAP.	
SSO musi przechowywać wszystkie informacje o zdefiniowanych organizacjach w katalogu LDAP.	
SSO musi przechowywać wszystkie informacje o członkostwie użytkowników w organizacji w katalogu LDAP.	
LDAP wchodzący w skład SSO musi zostać wdrożony w konfiguracji klastra multi-master replication.	
Role i grupy	
SSO musi zarządzać uprawnieniami, rolami, użytkownikami,	

grupami i organizacjami ze wszystkich Podsystemów.	
SSO musi umożliwiać definiowanie nowych ról użytkowników.	
SSO musi umożliwiać edycję zdefiniowanych wcześniej ról użytkowników.	
SSO musi umożliwiać wygodne zarządzanie powiązaniami ról z użytkownikami.	
SSO musi umożliwiać tworzenie grup użytkowników.	
SSO musi umożliwiać wygodne zarządzanie członkostwem użytkowników w grupach.	
SSO, w ramach procesu uwierzytelniania, musi udostępniać Podsystemom informacje o wszystkich rolach przypisanych (zarówno bezpośrednio, jak i za pośrednictwem grup) do uwierzytelnianego użytkownika.	
SSO musi zapewniać mechanizmy służące do oznaczania użytkowników w zależności od ich klasyfikacji biznesowej (np. "Klient", "Partner", "Złoty Partner" itp.).	
Organizacje	
SSO musi umożliwiać definiowanie nowych organizacji.	
SSO musi umożliwiać modyfikowanie nazwy organizacji.	
SSO musi umożliwiać definiowanie użytkowników należących do organizacji.	
SSO musi umożliwiać generowanie automatycznych stron logowania o adresach url typu: nazwa_organizacji.domenaplatformy.pl, które uwierzytelniają użytkowników logujących się przez taką stronę z organizacją o nazwie: nazwa_organizacji.	
SSO musi umożliwiać ustawienie loga dla automatycznie wygenerowanych stron logowania.	
SSO musi umożliwić Partnerom tworzenie organizacji dla ich klientów.	

SSO musi umożliwić Partnerom pełne zarządzanie organizacjami ich klientów (realizować wszystkie funkcje tak, jakby to klienci je realizowali).	
Uprawnienia w ramach SSO	
SSO musi umożliwiać nadanie uprawnień do każdej atomowej operacji wykonywanej w SSO.	
SSO musi umożliwiać nadanie uprawnień do odczytu każdego atomowego pola danych przechowywanych w SSO.	
SSO musi umożliwiać nadanie uprawnień do edycji każdego atomowego pola danych przechowywanych w SSO.	
SSO musi umożliwiać nadanie pojedynczego uprawnienia zezwalającego na dostęp do wszystkich organizacji zdefiniowanych w SSO.	
SSO musi umożliwiać nadanie uprawnienia dostępu do każdej pojedynczej organizacji zdefiniowanej w SSO.	
SSO musi umożliwiać nadanie uprawnienia dostępu do organizacji, do której należy dany użytkownik.	
SSO musi wymagać, by użytkownik wykonujący operację posiadał role, które łącznie dają mu uprawnienia do operacji i pól danych, których dotyczy operacja.	
SSO musi wymagać, by użytkownik wykonujący operację na koncie innym niż swoje posiadał uprawnienia dostępu do organizacji, do której należy to konto.	
SSO musi wyświetlać użytkownikowi tylko te pola danych, do których ma on dostęp.	
SSO musi umożliwiać użytkownikowi edycję tylko tych pól danych, do których ma on dostęp.	
SSO musi wyświetlać użytkownikowi tylko użytkowników należących do organizacji, do której ma on dostęp.	
Uwierzytelnianie: hasło	
SSO musi umożliwiać uwierzytelnienie użytkowników w	

oparciu o login i hasło.	
SSO musi przechowywać hasła w sposób uniemożliwiający ich odczytanie, z zastosowaniem silnej, kryptograficznie bezpiecznej funkcji skrótów oraz "soli" (salted hash).	
SSO musi umożliwiać użytkownikowi zmianę własnego hasła.	
SSO musi umożliwiać użytkownikowi posiadającemu odpowiednie uprawnienia zmianę hasła innego użytkownika.	
SSO powinno wyświetlać informację o sile nowego hasła podczas zmiany hasła użytkownika.	
SSO musi weryfikować zgodność nowego hasła podczas jego zmiany z określoną konfiguracyjnie polityką haseł.	
Mechanizmy określania polityki haseł w SSO powinny umożliwiać stworzenie polityki haseł odzwierciedlającej wytyczne GIODO.	
Mechanizmy określania polityki haseł w SSO powinny umożliwiać określenie czasu ważności haseł.	
SSO musi umożliwiać przypisanie różnych polityk haseł do różnych ról, grup i organizacji użytkowników.	
SSO musi udostępniać mechanizm resetowania hasła.	
Mechanizm resetowania hasła musi działać zgodnie z dobrymi praktykami rynkowymi w tym zakresie, np. zgodnie z wytycznymi OWASP Forgot Password Cheat Sheet.	
Mechanizm resetowania hasła musi być tak zaprojektowany, żeby zapobiegać powszechnie znanym atakom bezpieczeństwa.	
Uwierzytelnianie: zewnętrzni dostawcy tożsamości	
SSO musi umożliwiać rejestrację konta użytkownika z wykorzystaniem zewnętrznych dostawców tożsamości, w oparciu o protokół oAuth v2.	
SSO powinno automatycznie uzupełnić dane tworzonego konta danymi otrzymanymi od dostawcy tożsamości przy rejestracji za pomocą zewnętrznego dostawcy tożsamości.	

SSO powinno automatycznie uzupełnić dane rozliczeniowe danymi otrzymanymi od dostawcy tożsamości przy rejestracji za pomocą zewnętrznego dostawcy tożsamości.	
SSO powinno umożliwić użytkownikowi edycję automatycznie uzupełnionych danych przed ich zapisaniem przy rejestracji za pomocą zewnętrznego dostawcy tożsamości.	
SSO musi umożliwiać uwierzytelnianie z wykorzystaniem zewnętrznych dostawców tożsamości w oparciu o protokół oAuth v2.	
SSO musi obsługiwać zewnętrznego dostawcę tożsamości Facebook.	
SSO musi obsługiwać zewnętrznego dostawcę tożsamości Google.	
SSO musi obsługiwać zewnętrznego dostawcę tożsamości Windows Live.	
SSO musi obsługiwać zewnętrznego dostawcę tożsamości ePUAP.	
SSO musi umożliwiać użytkownikowi powiązanie jego konta z dowolną liczbą zewnętrznych dostawców tożsamości.	
SSO musi umożliwiać użytkownikowi usuwanie powiązań zewnętrznych dostawców tożsamości z jego kontem użytkownika.	
SSO musi umożliwiać administratorowi (na podstawie uprawnień) usuwanie powiązań zewnętrznych dostawców tożsamości z kontami użytkowników.	
Uwierzytelnianie: dwustopniowe uwierzytelnianie	
SSO musi umożliwiać użytkownikowi aktywację dwustopniowego uwierzytelniania.	
SSO musi obsługiwać dwustopniowe uwierzytelnianie w oparciu o kody SMS.	
SSO musi obsługiwać hasła jednorazowe oparte o algorytm	

HOTP (RFC 4226), w sposób kompatybilny z aplikacją Google Authenticator.	
SSO musi obsługiwać hasła jednorazowe oparte o algorytm TOTP (RFC 6238), w sposób kompatybilny z aplikacją Google Authenticator.	
SSO musi generować kody QR na potrzeby konfiguracji haseł jednorazowych.	
Panel konfiguracyjny	
Administrator musi dysponować panelem konfiguracyjnym, umożliwiającym modyfikację wszystkich opcji konfiguracyjnych SSO.	
SSO musi umożliwiać administratorowi określenie polityki haseł, definiującej reguły walidacji haseł ustawianych w SSO.	
SSO musi prowadzić rejestr udostępnień danych osobowych na potrzeby realizacji wymagań Ustawy o ochronie danych osobowych.	
SSO musi umożliwiać tworzenie wpisów w rejestrze udostępnień danych osobowych.	
SSO musi umożliwiać przeglądanie wpisów w rejestrze udostępnień danych osobowych.	
SSO musi umożliwiać oznaczanie wpisów w rejestrze udostępnień danych osobowych jako anulowanych.	
Konto użytkownika	
SSO musi umożliwiać użytkownikowi samodzielne utworzenie konta.	
SSO musi umożliwiać użytkownikowi o odpowiednich uprawnieniach utworzenie konta dla innego użytkownika.	
SSO musi umożliwiać wprowadzenie przy tworzeniu konta użytkownika jego danych osobowych oraz określania dla rejestrowanej organizacji czy jest to osoba fizyczna, firma z siedzibą w Unii Europejskiej bądź firma spoza Unii	

Europejskiej.	
SSO musi weryfikować europejskie numery NIP w oparciu o system VIES.	
SSO musi wymagać weryfikacji adresu e-mail dla konta założonego samodzielnie przez użytkownika.	
SSO musi wymagać weryfikacji adresu e-mail w przypadku jego samodzielnej zmiany przez użytkownika.	
SSO musi umożliwiać edycję danych konta użytkownika.	
SSO musi umożliwiać dezaktywację konta użytkownika.	
SSO musi umożliwiać (opcjonalnie) usunięcie danych osobowych użytkownika w momencie dezaktywacji jego konta.	
SSO musi umożliwiać reaktywację wcześniej dezaktywowanego konta użytkownika.	
SSO musi umożliwiać przeglądanie listy użytkowników.	
SSO musi umożliwiać filtrowanie listy użytkowników.	
SSO musi umożliwiać przeszukiwanie bazy użytkowników.	
SSO musi umożliwiać eksport danych użytkowników.	
SSO musi spełniać wymagania Ustawy o ochronie danych osobowych.	
SSO musi prowadzić rejestr wszystkich operacji wykonanych w ramach SSO.	
SSO musi udostępniać zalogowanemu użytkownikowi wszystkie dane, które należy mu udostępnić na podstawie Ustawy o ochronie danych osobowych.	
SSO musi udostępniać zalogowanemu użytkownikowi stronę umożliwiającą wgląd w jego dane zarejestrowane w SSO w zakresie wynikającym z nadanych mu uprawnień.	
SSO musi udostępniać zalogowanemu użytkownikowi możliwość edycji jego danych zarejestrowanych w SSO w zakresie wynikającym z nadanych mu uprawnień.	
SSO musi udostępniać zalogowanemu użytkownikowi	

możliwość podglądu rejestru zdarzeń, które dotyczą jego konta użytkownika.	
SSO musi udostępniać zalogowanemu użytkownikowi możliwość podglądu rejestru zdarzeń, które dotyczą jego danych osobowych.	
SSO musi umożliwiać zdefiniowanie terminu ważności konta użytkownika.	
SSO musi automatycznie dezaktywować konta, dla których upłynął termin ważności.	
Konto użytkownika: klucze SSH	
SSO musi umożliwiać zarejestrowanie dowolnej ilości kluczy publicznych SSH dla użytkownika.	
SSO musi obsługiwać klucze publiczne SSH w formacie zgodnym z aplikacją OpenSSH.	
SSO musi obsługiwać klucze publiczne SSH w formacie zgodnym z aplikacją PuTTY.	
SSO musi umożliwiać konwersję kluczy publicznych użytkownika między obsługiwanymi formatami.	
SSO musi przechowywać klucze publiczne użytkownika w formacie zgodnym z aplikacją OpenSSH.	
SSO musi przechowywać klucze publiczne użytkownika w LDAP.	
SSO musi umożliwiać przeglądanie listy zarejestrowanych kluczy publicznych użytkownika.	
SSO musi umożliwiać zmianę zarejestrowanych kluczy publicznych użytkownika.	
SSO musi umożliwiać usuwanie zarejestrowanych kluczy publicznych użytkownika.	
SSO musi umożliwiać wygenerowanie pary kluczy kryptograficznych algorytmu RSA dla użytkownika.	
SSO musi umożliwiać wygenerowanie pary kluczy	

kryptograficznych algorytmu DSA dla użytkownika.	

27. Portal

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.9 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.9 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	Oferowana funkcjonalność:
Portal będzie stanowić platformę do publikacji treści związanych z działalnością Zamawiającego w obszarze IaaS i PaaS.	
Zasilenie treścią	
Projekt techniczny Portalu musi zawierać opis architektury informacji Portalu.	
Projekt techniczny Portalu musi zawierać spis wszystkich treści, które są niezbędne do rozpoczęcia świadczenia Usług przez Zamawiającego.	
Wykonawca musi przygotować wszystkie treści (teksty, uzupełniające materiały graficzne) niezbędne do rozpoczęcia świadczenia Usług przez Zamawiającego.	
Wykonawca musi przygotować regulaminy świadczenia Usług przez Zamawiającego w celu ich publikacji na Portalu.	
Wykonawca musi przygotować politykę prywatności dla Usług świadczonych przez Zamawiającego w celu ich publikacji na Portalu.	
Portal w momencie odbioru musi zawierać wszystkie treści przygotowane przez Wykonawcę.	
Portal musi udostępniać wszystkie funkcjonalności niezbędne do zasilenia go treścią opracowaną przez Wykonawcę.	
Portal musi udostępniać wszystkie funkcjonalności niezbędne do zaimplementowania docelowej architektury informacji.	
Portal musi udostępniać wszystkie funkcjonalności niezbędne do zaimplementowania opracowanego projektu graficznego.	
Funkcjonalności administratora i uprawnienia	
Portal musi umożliwić zarządzanie dostępem do treści portalu przez wielu użytkowników, zgodnie z ich uprawnieniami.	
Portal musi realizować kontrolę dostępu do poszczególnych	

funkcjonalności na podstawie roli użytkownika określonej przez SSO.	
Portal musi umożliwiać nadanie każdej z ról uprawnień do korzystania z poszczególnych funkcjonalności.	
Portal musi umożliwiać nadanie każdej z ról uprawnień do odczytu wybranych rodzajów treści (np. artykuł, opis Usługi).	
Portal musi umożliwiać nadanie każdej z ról uprawnień do modyfikacji wybranych rodzajów treści (np. artykuł, opis Usługi).	
Portal musi umożliwiać nadanie każdej z ról uprawnień do modyfikacji treści utworzonych przez aktualnie zalogowanego użytkownika..	
Portal musi umożliwiać nadanie każdej z ról uprawnień dostępu do konkretnych treści (np. konkretnego artykułu).	
Portal musi definiować uprawnienie określające, kto może publikować treści.	
Tworzenie i zarządzanie treścią	
Portal musi umożliwiać tworzenie treści za pomocą edytora WYSIWYG.	
Portal musi poprawnie obsługiwać wklejanie sformatowanych i ostylowanych treści (co najmniej dla treści wklejanych z programu MS Word 2010 oraz przeglądarek internetowych).	
Portal musi zapewniać wersjonowanie wszystkich stron.	
Portal musi umożliwiać przywrócenie dowolnej z poprzednich wersji danej treści.	
Portal musi umożliwiać wyświetlenie różnic między dwoma wersjami treści.	
Portal musi zapewnić mechanizm podglądu zmian przed opublikowaniem.	
Portal musi umożliwiać zapisanie treści jako wersji roboczej (nieopublikowanej).	

Portal musi umożliwiać opublikowanie wersji roboczej treści.	
Portal musi umożliwiać opublikowanie treści bez zapisywania jej jako wersji roboczej.	
Portal musi zapewniać możliwość definiowania harmonogramów publikacji (wybór daty oraz czasu (z dokładnością do minuty), o której pojawi się dana treść na portalu oraz daty i czasu (z dokładnością do minuty), o której treść zniknie z portalu.	
Portal musi zabezpieczyć użytkownika przed przypadkową utratą treści w czasie ich wprowadzania (np. w efekcie przypadkowego przeładowania lub zamknięcia okna przeglądarki).	
Obsługa załączników	
Portal musi umożliwiać definiowanie repozytoriów plików, pełniących rolę stron z plikami "do pobrania".	
Portal musi umożliwiać modyfikację nazw plików umieszczonych w repozytorium plików.	
Portal musi umożliwiać definiowanie tekstowego tytułu pliku umieszczonego w repozytorium, odrębnego od nazwy pliku.	
Portal musi umożliwiać definiowanie tekstowego opisu pliku umieszczonego w repozytorium.	
Portal musi zapewnić możliwość publikowania galerii zdjęć wraz z mechanizmami pokazu slajdów (slideshow).	
Portal musi umożliwiać dodawanie załączników (plików) do poszczególnych treści z poziomu edytora WYSIWYG.	
Portal musi umożliwiać osadzanie w treści obrazów z poziomu edytora WYSIWYG.	
Portal musi umożliwiać osadzanie w treści filmów z serwisu YouTube z poziomu edytora WYSIWYG.	
Portal musi umożliwiać osadzanie w treści filmów z serwisu Vimeo z poziomu edytora WYSIWYG.	

Portal musi umożliwiać dodanie wielu plików jednocześnie.	
Portal musi umożliwiać przeglądanie wszystkich plików umieszczonych w Portalu.	
Dla każdego pliku umieszczonego w Portalu, Portal musi zapewnić możliwość łatwego zidentyfikowania zasobu (np. treści, repozytorium plików, galerii), w którym został wykorzystany dany plik.	
Portal musi umożliwiać usuwanie wybranych plików.	
Eksport danych	
Portal musi umożliwiać masowy eksport wybranych treści do formatu CSV obsługiwanego poprawnie przez aplikację MS Excel 2010.	
Portal musi umożliwiać masowy eksport wybranych treści do formatu XML.	
Adresy URL	
Portal musi automatycznie tworzyć semantyczne adresy URL dla każdej treści.	
Portal musi umożliwiać modyfikację adresu URL, pod którym widoczna jest treść.	
Portal musi zapewnić, że każda treść jest dostępna tylko pod jednym adresem URL.	
Portal musi umożliwiać tworzenie przekierowań z jednego adresu na inny.	
Portal musi automatycznie tworzyć przekierowanie ze starego adresu na nowy po zmianie adresu URL treści.	
Portal musi zapewniać zabezpieczenie przed pętlami przekierowań (powodującymi np. błąd ERR_TOO_MANY_REDIRECTS w przeglądarce Chrome).	
Kategoryzacja i tagowanie	
Portal musi umożliwiać definiowanie wielu kategorii tagów.	
Portal musi umożliwiać powiązanie rodzaju treści z wieloma	

kategoriami tagów.	
Portal musi umożliwiać opisywanie treści nieograniczoną ilością tagów.	
Portal musi umożliwiać opisywanie plików w repozytorium nieograniczoną ilością tagów.	
Portal musi umożliwiać tworzenie nowych tagów poprzez wpisanie ich w trakcie tworzenia/edycji treści.	
Portal musi umożliwiać tworzenie nowych tagów poprzez wpisanie ich w trakcie tworzenia/edycji pliku w repozytorium.	
Portal musi umożliwiać tworzenie i usuwanie nowych tagów z poziomu ekranu edycji kategorii tagów.	
Portal musi umożliwiać definiowanie wielu kategorii terminów.	
Portal musi umożliwiać definiowanie wielu terminów w jednej kategorii terminów.	
Portal musi umożliwiać organizowanie terminów z jednej kategorii terminów w strukturę drzewiastą.	
Portal musi umożliwiać powiązanie rodzaju treści z wieloma kategoriami terminów.	
Portal musi umożliwiać powiązanie treści z jednym lub wieloma terminami z powiązanych kategorii terminów.	
Portal musi umożliwiać tworzenie nowych terminów w trakcie tworzenia/edycji treści.	
Portal musi umożliwiać tworzenie nowych terminów w trakcie tworzenia/edycji pliku w repozytorium.	
Portal musi umożliwiać opisywanie plików w repozytorium nieograniczoną ilością terminów.	
Portal musi umożliwiać tworzenie i usuwanie nowych terminów z poziomu ekranu edycji kategorii terminów.	
Widoki	
Portal musi umożliwiać tworzenie widoków, pozwalających na wyświetlanie na jednej stronie różnych treści.	

Portal musi umożliwiać tworzenie galerii zdjęć.	
Portal musi umożliwiać tworzenie list zawierających wybrane atrybuty treści.	
Portal musi umożliwiać tworzenie pokazów slajdów (slideshow).	
Menu	
Portal musi umożliwiać definiowanie wielu niezależnych struktur menu.	
Portal musi umożliwiać dodawanie odnośnika do treści w dowolnym miejscu menu.	
Portal musi umożliwiać aktywację / deaktywację elementów menu.	
Portal musi umożliwiać przypisanie do każdego elementu menu obrazu / ikony reprezentującej ten element menu.	
Portal musi umożliwiać przypisanie do każdego elementu menu unikalnej klasy CSS.	
Portal musi umożliwiać łatwą reorganizację struktury menu.	
Portal musi umożliwiać generowanie ścieżki do aktualnej lokalizacji ("breadcrumb") na podstawie pozycji w strukturze treści.	
Wersje językowe	
Wszystkie teksty występujące w Portalu powinny być wyświetlane w aktywnym języku.	
Portal musi umożliwiać jednoczesne stosowanie wielu mechanizmów określania aktywnego języka interfejsu użytkownika, uwzględniając określone przez administratora priorytety poszczególnych mechanizmów (jeśli mechanizm o najwyższym priorytecie wykryje język, to ten język jest stosowany; jeśli nie - przetwarzany jest kolejny mechanizm itp.).	
Portal musi zawierać mechanizm określania aktywnego języka	

na podstawie struktury adresu URL.	
Portal musi zawierać mechanizm określania aktywnego języka na podstawie konfiguracji przeglądarki użytkownika.	
Portal musi zawierać mechanizm określania aktywnego języka na podstawie zapisanych preferencji użytkownika (dla użytkowników zalogowanych).	
Portal musi zawierać mechanizm określania aktywnego języka na podstawie ciasteczek (cookies) i/lub sesji przeglądarki.	
Portal musi udostępniać mechanizm pozwalający użytkownikowi na przełączanie się między wersjami językowymi.	
Portal musi umożliwiać wprowadzanie tłumaczeń treści na każdy z obsługiwanych języków.	
Portal musi automatycznie wyświetlać treści w wybranym przez użytkownika języku, o ile takie tłumaczenie jest dostępne.	
Portal musi umożliwiać tłumaczenie elementów menu na każdy z obsługiwanych języków.	
Komentarze	
Portal musi umożliwić włączenie/wyłączenie opcji komentowania danej strony.	
Portal musi umożliwiać komentowanie bezpośrednio w Portalu, za pomocą platformy Facebook i Disqus (w zależności od konfiguracji zdefiniowanej przez administratora).	
Portal musi integrować się z Podsystemem SSO w celu uwierzytelniania i autoryzacji do zasobów.	
Portal musi umożliwić dodawanie plików/załączników do komentarzy.	
Portal musi umożliwiać ocenianie poszczególnych treści.	
Formularze i ankiety	
Portal musi umożliwiać tworzenie formularzy elektronicznych.	

Portal musi rejestrować wszystkie dane wprowadzane do formularzy elektronicznych.	
Portal musi umożliwiać definiowanie powiadomień e-mail wysyłanych po wypełnieniu formularza elektroniczego.	
Portal musi umożliwiać definiowanie reguł walidacji dla poszczególnych pól formularza elektroniczego.	
Portal musi umożliwiać tworzenie i wypełnianie ankiet.	
Portal musi umożliwiać dodanie do ankiety listy opcji, które mogą wybrać użytkownicy.	
Portal musi umożliwiać wypełnianie ankiet zarówno nie zalogowanym, jak i zalogowanym użytkownikom.	
Portal musi umożliwiać ograniczenie ilości głosów w ankiecie (ilości "submitów") oddawanych przez jednego użytkownika (zarówno w przypadku użytkowników zalgoowanych, jak i nie zalogowanych).	
Portal musi umożliwiać definiowanie w ankietach elementów typu: text, radio, checkbox, date, datetime, time, email, number (typy te należy rozumieć zgodnie z definicjami opisanymi w standardzie HTML 5).	
Portal, dla elementów wielokrotnego wyboru, musi umożliwiać definiowanie ilości wyborów, które może zaznaczyć użytkownik.	
Portal musi umożliwiać definiowanie reguł widoczności elementów ankiety (wyświetlanie / ukrywanie elementów na podstawie wyboru dokonanego w innych elementach ankiety).	
Portal musi umożliwiać podliczanie ilości głosów oddanych na poszczególne wartości elementów ankiety (ilość zaznaczonych checkbox'ów w danym elemencie, suma i średnia wpisanych wartości w elementach liczbowych, ilość wystąpień każdej z wprowadzonych / wybranych wartości w danym elemencie).	
Portal musi umożliwiać eksportowanie wyników ankiet.	

Portal musi umożliwiać automatyczną publikację wyników ankiet.	
FAQ	
Portal musi umożliwiać tworzenie stron z najczęściej zadawanymi pytaniami i odpowiedziami na nie (FAQ).	
Forum	
Portal musi umożliwiać stworzenie forum dyskusyjnego wsparcia dla użytkowników.	
Portal musi umożliwiać moderowanie wpisów publikowanych na forum.	
Portal musi umożliwiać tworzenie kategorii forum.	
Portal musi umożliwiać tworzenie tematów dyskusji w ramach kategorii forum.	
Wpisy na forum powinny być tworzone z wykorzystaniem edytora WYSIWYG.	
Wyszukiwanie	
Portal musi umożliwiać przeszukiwanie bazy wiedzy wraz z przeszukiwaniem załączników oraz wizualizacją (zaznaczeniem) tekstu wyszukiwanego w skrócie/cytacie podczas wyświetlania wyników. W wyświetlanych wynikach nie jest wymagane zachowanie właściwego formatowania dokumentu źródłowego.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie opublikowanych treści.	
Portal w wynikach wyszukiwania powinien uwzględniać prawa dostępu zalogowanego użytkownika.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie PDF.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie DOC.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w	

zgromadzonych w Portalu plikach w formacie DOCX.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie RTF.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie HTML.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie OpenDocument.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie TXT.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie XLS.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie XLSX.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie PPT.	
Portal musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Portalu plikach w formacie PPTX.	
Portal musi uwzględniać w wynikach wyszukiwania tylko treści publicznie dostępne.	
Portal musi uwzględniać zmiany w indeksie wyszukiwania maksymalnie w ciągu godziny od wprowadzenia zmiany w treści.	
Wyszukiwarka powinna być skonfigurowana w taki sposób, aby czas wyszukiwania bazy 100000 artykułów o wielkości średnio 10 000 znaków i nie przekraczał 1000ms, niezależnie od formatu pliku.	
Portal musi umożliwiać filtrowanie wyników wyszukiwania za pomocą mechanizmów "facets", na podstawie zdefiniowanych atrybutów (pól) treści.	
Portal musi zaznaczać wystąpienia tekstu wyszukiwanego w skróceie/cytacie podczas wyświetlania wyników wyszukiwania.	

Mechanizmy wyszukiwania powinny poprawnie obsługiwać różne języki.	
Portal musi poprawnie obsługiwać polską fleksję w ramach mechanizmów wyszukiwania.	
Portal musi poprawnie obsługiwać angielską fleksję w ramach mechanizmów wyszukiwania.	
Portal, w ramach mechanizmów wyszukiwania, musi umożliwiać określenie wag dla poszczególnych atrybutów wyszukiwanych treści	
Statystyki	
Portal musi umożliwiać integrację z (wybraną na etapie przygotowania projektu technicznego) platformą służącą do zbierania i analizy statystyk odwiedzin.	
Obsługa wielu domen	
Portal musi umożliwiać wyświetlanie innych treści pod różnymi adresami domenowymi (mechanizm "multi-site"), przy wykorzystaniu wspólnej instancji bazy danych.	
Portal musi umożliwiać kierowanie odrębnych poddomen do fragmentów struktury Portalu	
Rozszerzenia i szablony	
Portal musi umożliwiać tworzenie rozszerzeń (pluginów, modułów) Podsystemu.	
W przypadku bazowania na istniejącym oprogramowaniu, cały kod wytworzony przez Wykonawcę w ramach Projektu musi zostać zrealizowany jako rozszerzenie (plugin, moduł itp) lub szablon do Portalu.	
Portal musi umożliwiać zainstalowanie wielu szablonów ("skórek").	
Portal musi umożliwiać wykorzystanie odrębnego szablonu w zależności od przeglądanej strony.	
Portal musi umożliwiać wykorzystanie odrębnego szablonu w	

zależności od pozycji strony w strukturze menu.	
Portal musi umożliwiać wykorzystanie odrębnego szablonu w zależności od roli użytkownika.	
Pozostałe	
Portal musi zawierać skuteczne zabezpieczenie wszystkich formularzy przed spamem o efektywności na poziomie co najmniej 99% (maksymalnie 1% postów może zostać niepoprawnie sklasyfikowanych), np. poprzez integrację z usługą typu Mollom	
Wszystkie funkcjonalności Portalu (w tym te związane z administracją Portalem oraz dostępem do treści Portalu) muszą być realizowane przez przeglądarkę Internetową.	
Portal musi zapewnić mechanizm cache'owania renderowanych szablonów.	
Portal musi udostępnić pełne API, pozwalające na jego kontrolę oraz konfigurację.	
SEO	
Portal musi zapewniać możliwość automatycznego opisywania treści metadanymi na potrzeby SEO.	
Portal musi zapewniać możliwość automatycznego opisywania treści metadanymi na potrzeby sieci społecznościowych (co najmniej w zakresie określonym przez standard OpenGraph i publikacje schema.org).	
Portal musi zapewniać możliwość ręcznego opisywania treści metadanymi na potrzeby SEO.	
Portal musi zapewniać możliwość ręcznego opisywania treści metadanymi na potrzeby sieci społecznościowych (co najmniej w zakresie określonym przez standard OpenGraph i publikacje schema.org).	
Portal musi automatycznie tworzyć sitemapy XML (artykuły oraz grafiki), oraz zgłaszać je po wygenerowaniu do Google.	

Portal, w ramach obsługi semantycznych adresów URL, nie musi pozwalać na indeksowanie stron w ścieżkach systemowych, lecz automatycznie przekierowywać z nich na prawidłowy, przyjazny adres, który będzie indeksowany.	
Wersja dla niepełnosprawnych	
Portal musi umożliwiać przełączenie serwisu do wersji o wysokim kontraście.	
Portal musi zapewnić standard Aria.	
Portal musi prawidłowo definiować nawigację po elementach serwisu za pomocą "tabindex".	
Portal musi ukrywać w wersji dla niepełnosprawnych zbędne elementy interfejsu, które mogłyby zaburzać pracę czytników bądź nawigację osób niepełnosprawnych po stronie.	
Wersja żałobna serwisu	
Portal musi zawierać specjalny zestaw styli dla wersji żałobnej serwisu, którą można będzie włączyć z poziomu panelu administracyjnego Podsystemu.	
Wersja żałobna Portalu powinna zmieniać także paletę kolorów dla obrazków wyświetlanych na stronie.	
Obsługa procesu zakupowego	
Portal musi zawierać funkcjonalności pozwalające na zakup Usług przez Klientów.	
Realizacja płatności elektronicznych musi odbywać się zgodnie ze standardem PCI Data Security Standard.	
Portal musi integrować się z SelfService w celu zrealizowania procesu sprzedażowego.	
Portal musi integrować się z Billingiem w celu pobierania informacji o cenach produktów.	
Portal musi zawierać mechanizmy umożliwiające Klientom oszacowanie kosztów wykorzystywanych Usług.	
Portal musi umożliwiać zapisywanie danych wprowadzanych	

przez Klientów w ramach szacowania kosztów Usług.	
Portal musi umożliwiać eksport zapisanych wcześniej danych wprowadzanych przez Klientów w ramach szacowania kosztów Usług do formatu CSV kompatybilnego z MS Excel 2010.	

28. Baza Wiedzy

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.11 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.11 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

Wymagana funkcjonalność:	Oferowana funkcjonalność:
Celem Bazy Wiedzy jest zbudowanie dwóch baz wiedzy: dla Zamawiającego i dla Klientów. W ocenie Zamawiającego rozwiązaniem optymalnym byłoby zbudowanie Podsystemu na bazie rozwiązania klasy "wiki".	
Zasilenie treścią	
Wykonawca musi przygotować wszystkie treści (teksty, uzupełniające materiały graficzne) i zasilić nimi Bazę Wiedzy w zakresie potrzebnym do rozpoczęcia świadczenia Usług przez Zamawiającego.	
Baza Wiedzy musi udostępniać wszystkie funkcjonalności niezbędne do zasilenia go treścią opracowaną przez Wykonawcę	
Baza Wiedzy musi udostępniać wszystkie funkcjonalności niezbędne do zaimplementowania docelowej architektury informacji.	
Baza Wiedzy musi udostępniać wszystkie funkcjonalności niezbędne do zaimplementowania opracowanego projektu graficznego.	
Funkcjonalności administratora i uprawnienia	
Baza Wiedzy musi realizować kontrolę dostępu do poszczególnych funkcjonalności na podstawie roli użytkownika określonej przez SSO.	
Baza Wiedzy musi umożliwiać nadanie każdej osobie i roli uprawnień do korzystania z poszczególnych funkcjonalności.	
Baza Wiedzy musi umożliwiać nadanie każdej osobie i roli uprawnień do odczytu wybranych części struktury bazy	

wiedzy.	
Baza Wiedzy musi umożliwiać nadanie każdej osobie i roli uprawnień do modyfikacji wybranych części struktury bazy wiedzy.	
Baza Wiedzy musi umożliwiać nadanie każdej osobie i roli uprawnień do kontroli dostępu do wybranych części struktury bazy wiedzy.	
Baza Wiedzy musi umożliwiać nadanie każdej osobie i roli uprawnień dostępu do konkretnych treści (np. konkretnego artykułu).	
Tworzenie i zarządzanie treścią	
Baza Wiedzy musi umożliwiać tworzenie treści za pomocą języka znaczników (ang. markup language).	
Baza Wiedzy musi umożliwiać tworzenie treści za pomocą edytora WYSIWYG.	
Baza Wiedzy musi poprawnie obsługiwać wklejanie sformatowanych i ostylowanych treści (co najmniej dla treści wklejanych z programu MS Word 2010 oraz przeglądarek internetowych)	
Baza Wiedzy musi zapewniać wersjonowanie wszystkich stron.	
Baza Wiedzy musi umożliwiać przywrócenie dowolnej z poprzednich wersji danej treści.	
Baza Wiedzy musi umożliwiać wyświetlenie różnic między dwoma wersjami treści.	
Baza Wiedzy musi zapewnić mechanizm podglądu zmian przed opublikowaniem.	
Baza Wiedzy musi umożliwiać zapisanie treści jako wersji roboczej (nieopublikowanej).	
Baza Wiedzy musi umożliwiać opublikowanie wersji roboczej treści.	
Baza Wiedzy musi definiować uprawnienie określające, kto	

może publikować treści.	
Baza Wiedzy musi zabezpieczyć użytkownika przed przypadkową utratą treści w czasie ich wprowadzania (np. w efekcie przypadkowego przeładowania lub zamknięcia okna przeglądarki).	
Baza Wiedzy musi umożliwiać wydzielenie w strukturze bazy wiedzy artykułów dostępnych tylko dla określonych użytkowników / grup.	
Baza Wiedzy musi umożliwiać organizację treści w bazie wiedzy w strukturę drzewiastą.	
SEO	
Baza Wiedzy musi zapewniać możliwość automatycznego opisywania treści metadanymi na potrzeby SEO.	
Baza Wiedzy musi zapewniać możliwość automatycznego opisywania treści metadanymi na potrzeby sieci społecznościowych.	
Baza Wiedzy musi zapewniać możliwość ręcznego opisywania treści metadanymi na potrzeby SEO.	
Baza Wiedzy musi zapewniać możliwość ręcznego opisywania treści metadanymi na potrzeby sieci społecznościowych.	
Baza Wiedzy musi automatycznie tworzyć sitemapy XML (artykuły oraz grafiki), oraz zgłaszać je po wygenerowaniu do wyszukiwarki Google.	
Baza Wiedzy w ramach obsługi semantycznych adresów URL, nie może pozwalać na indeksowanie stron w ścieżkach systemowych; powinien automatycznie przekierowywać z nich na prawidłowy, przyjazny adres, który będzie indeksowany.	
Obsługa załączników	
Baza Wiedzy musi umożliwiać definiowanie repozytoriów plików, pełniących rolę stron z plikami "do pobrania".	
Baza Wiedzy musi umożliwiać modyfikację nazw plików	

umieszczonych w repozytorium plików.	
Baza Wiedzy musi umożliwiać definiowanie tekstowego tytułu pliku umieszczonego w repozytorium, odrębnego od nazwy pliku.	
Baza Wiedzy musi umożliwiać definiowanie tekstowego opisu pliku umieszczonego w repozytorium.	
Baza Wiedzy musi umożliwiać dodawanie załączników (plików) do poszczególnych treści z poziomu edytora WYSIWYG.	
Baza Wiedzy musi umożliwiać osadzanie w treści obrazów z poziomu edytora WYSIWYG.	
Baza Wiedzy musi umożliwiać osadzanie w treści filmów z serwisu YouTube z poziomu edytora WYSIWYG.	
Baza Wiedzy musi umożliwiać osadzanie w treści filmów z serwisu Vimeo z poziomu edytora WYSIWYG.	
Dla każdego pliku umieszczonego w Bazy Wiedzy, Podsystem musi zapewnić możliwość łatwego zidentyfikowania zasobu (np. artykułu, repozytorium plików), w którym został wykorzystany dany plik.	
Baza Wiedzy musi umożliwiać usuwanie wybranych plików.	
Eksport danych	
Baza Wiedzy musi umożliwiać eksport każdego artykułu do formatu PDF.	
Baza Wiedzy musi umożliwiać eksport wybranego artykułu wraz z wszystkimi artykułami podrzędnymi do formatu PDF.	
Baza Wiedzy musi umożliwiać masowy eksport wybranych treści do formatu CSV obsługiwanego poprawnie przez aplikację MS Excel.	
Baza Wiedzy musi umożliwiać masowy eksport wybranych treści do formatu XML.	
Adresy URL	

Baza Wiedzy musi automatycznie tworzyć semantyczne adresy URL dla każdej treści.	
Baza Wiedzy musi umożliwiać modyfikację adresu URL, pod którym widoczna jest treść.	
Baza Wiedzy musi zapewnić, że każda treść jest dostępna tylko pod jednym adresem URL.	
Baza Wiedzy musi umożliwiać tworzenie przekierowań z jednego adresu na inny.	
Baza Wiedzy musi automatycznie tworzyć przekierowanie ze starego adresu na nowy po zmianie adresu URL treści.	
Baza Wiedzy musi zapewniać zabezpieczenie przed pętlami przekierowań (powodującymi np. błąd ERR_TOO_MANY_REDIRECTS w przeglądarce Chrome)	
Kategoryzacja i tagowanie	
Baza Wiedzy musi umożliwiać opisywanie treści nieograniczoną ilością tagów.	
Baza Wiedzy musi umożliwiać opisywanie plików w repozytorium nieograniczoną ilością tagów.	
Baza Wiedzy musi umożliwiać tworzenie nowych tagów poprzez wpisanie ich w trakcie tworzenia/edycji treści.	
Baza Wiedzy musi umożliwiać tworzenie nowych tagów poprzez wpisanie ich w trakcie tworzenia/edycji pliku w repozytorium.	
Baza Wiedzy musi umożliwiać definiowanie wielu kategorii.	
Baza Wiedzy musi umożliwiać definiowanie wielu terminów w jednej kategorii.	
Baza Wiedzy musi umożliwiać organizowanie terminów z jednej kategorii w strukturę drzewiastą.	
Baza Wiedzy musi umożliwiać powiązanie treści z jednym lub wieloma terminami.	
Baza Wiedzy musi umożliwiać tworzenie nowych terminów w	

trakcie tworzenia/edycji treści.	
Menu	
Baza Wiedzy musi umożliwiać definiowanie wielu niezależnych struktur menu.	
Baza Wiedzy musi umożliwiać dodawanie odnośnika do treści w dowolnym miejscu menu.	
Baza Wiedzy musi umożliwiać aktywację / deaktywację elementów menu.	
Baza Wiedzy musi umożliwiać przypisanie do każdego elementu menu obrazu / ikony reprezentującej ten element menu.	
Baza Wiedzy musi umożliwiać przypisanie do każdego elementu menu unikalnej klasy CSS.	
Baza Wiedzy musi umożliwiać łatwą reorganizację struktury menu.	
Baza Wiedzy musi umożliwiać generowanie ścieżki do aktualnej lokalizacji ("breadcrumb") na podstawie pozycji w strukturze treści.	
Wersje językowe	
Baza Wiedzy, w momencie wdrożenia, musi obsługiwać język polski i angielski.	
Baza Wiedzy musi umożliwiać rozszerzenie listy obsługiwanych języków o kolejne języki.	
Wszystkie teksty występujące w Bazie Wiedzy powinny być wyświetlane w aktywnym języku.	
Baza Wiedzy musi umożliwiać jednoczesne stosowanie wielu mechanizmów określania aktywnego języka interfejsu użytkownika, uwzględniając określone przez administratora priorytety poszczególnych mechanizmów (jeśli mechanizm o najwyższym priorytecie wykryje język, to ten język jest stosowany; jeśli nie - przetwarzany jest kolejny mechanizm	

itp).	
Baza Wiedzy musi zawierać mechanizm określania aktywnego języka na podstawie struktury adresu URL.	
Baza Wiedzy musi zawierać mechanizm określania aktywnego języka na podstawie konfiguracji przeglądarki użytkownika.	
Baza Wiedzy musi zawierać mechanizm określania aktywnego języka na podstawie zapisanych preferencji użytkownika (dla użytkowników zalogowanych).	
Baza Wiedzy musi zawierać mechanizm określania aktywnego języka na podstawie ciasteczek (cookies) i/lub sesji przeglądarki.	
Baza Wiedzy musi udostępniać mechanizm pozwalający użytkownikowi na przełączanie się między wersjami językowymi.	
Baza Wiedzy musi umożliwiać tłumaczenie treści na każdy z obsługiwanych języków.	
Baza Wiedzy musi automatycznie przekierowywać użytkownika do tłumaczenia treści w aktywnym języku, o ile takie tłumaczenie jest dostępne.	
Baza Wiedzy musi umożliwiać tłumaczenie elementów menu na każdy z obsługiwanych języków.	
Komentarze	
Baza Wiedzy musi umożliwiać włączenie/wyłączenie opcji komentowania danej treści.	
Baza Wiedzy musi zapewniać możliwość komentowania treści.	
Baza Wiedzy musi umożliwiać komentowanie bezpośrednio w Podsystemie, za pomocą platformy Facebook i Disqus (w zależności od konfiguracji zdefiniowanej przez administratora)	
Baza Wiedzy musi obsługiwać załączniki do komentarzy.	
Formularze i ankiety	
Baza Wiedzy musi umożliwiać tworzenie formularzy	

elektronicznych	
Baza Wiedzy musi rejestrować wszystkie dane wprowadzane do formularzy elektronicznych.	
Baza Wiedzy musi umożliwiać definiowanie powiadomień e-mail wysyłanych po wypełnieniu formularza elektroniczego.	
FAQ	
Baza Wiedzy musi umożliwiać tworzenie stron z najczęściej zadawanymi pytaniami i odpowiedziami na nie (FAQ).	
Wyszukiwanie	
Baza Wiedzy musi umożliwiać przeszukiwanie bazy wiedzy wraz z przeszukiwaniem załączników oraz wizualizacją (zaznaczeniem) tekstu wyszukiwanego w skrócie/cytacie podczas wyświetlania wyników. W wyświetlanych wynikach nie jest wymagane zachowanie właściwego formatowania dokumentu źródłowego.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie PDF.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie DOC.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie DOCX.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie RTF.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie HTML.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie OpenDocument.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie TXT.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie	

w zgromadzonych w Bazie Wiedzy plikach w formacie XLS.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie XLSX.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie PPT.	
Baza Wiedzy musi umożliwiać pełno tekstowe wyszukiwanie w zgromadzonych w Bazie Wiedzy plikach w formacie PPTX.	
Baza Wiedzy musi uwzględniać w wynikach wyszukiwania tylko treści opublikowane.	
Baza Wiedzy musi uwzględniać w wynikach wyszukiwania tylko treści dostępne dla aktualnego użytkownika.	
Baza Wiedzy musi uwzględniać zmiany w indeksie wyszukiwania maksymalnie w ciągu godziny od wprowadzenia zmiany w treści.	
Wyszukiwarka powinna być skonfigurowana w taki sposób, aby czas wyszukiwania bazy 100000 artykułów o wielkości średnio 10 000 znaków był liniowy i nie przekraczał 1000ms, niezależnie od formatu pliku.	
Baza Wiedzy musi umożliwiać filtrowanie wyników wyszukiwania za pomocą mechanizmów "facets", na podstawie zdefiniowanych atrybutów (pól) treści.	
Baza Wiedzy musi zaznaczać wystąpienia tekstu wyszukiwanego w skrócie/cytacie podczas wyświetlania wyników wyszukiwania.	
Mechanizmy wyszukiwania powinny poprawnie obsługiwać różne języki.	
Baza Wiedzy musi poprawnie obsługiwać polską fleksję w ramach mechanizmów wyszukiwania.	
Baza Wiedzy musi poprawnie obsługiwać angielską fleksję w ramach mechanizmów wyszukiwania.	
Baza Wiedzy w ramach mechanizmów wyszukiwania, musi	

umożliwiać określenie wag dla poszczególnych atrybutów wyszukiwanych treści	
Statystyki	
Baza Wiedzy musi umożliwiać integrację z (wybraną na etapie przygotowania projektu technicznego) platformą służącą do zbierania i analizy statystyk odwiedzin.	
Obsługa wielu domen	
Baza Wiedzy musi umożliwiać wyświetlanie innych treści pod różnymi adresami domenowymi, przy wykorzystaniu jednej bazy danych (mechanizm "multi-site")	
Baza Wiedzy musi być posadowiony w co najmniej 2 instancjach: wewnętrznej (dla pracowników Zamawiającego) i publicznej (dla Klientów).	
Rozszerzenia i szablony	
Baza Wiedzy musi umożliwiać tworzenie rozszerzeń (pluginów, modułów) Podsystemu.	
W przypadku bazowania na istniejącym oprogramowaniu, cały kod wytworzony przez Wykonawcę w ramach Projektu musi zostać zrealizowany jako rozszerzenie (plugin, moduł itp) lub szablon do Podsystemu.	
Baza Wiedzy musi umożliwiać zainstalowanie wielu szablonów ("skórek").	
Baza Wiedzy musi umożliwiać wykorzystanie odrębnego szablonu w zależności od przeglądanej strony.	
Baza Wiedzy musi umożliwiać wykorzystanie odrębnego szablonu w zależności od pozycji strony w strukturze menu.	
Baza Wiedzy musi umożliwiać wykorzystanie odrębnego szablonu w zależności od roli użytkownika.	
Wersja dla niepełnosprawnych	
Baza Wiedzy musi umożliwiać przełączenie serwisu do wersji o wysokim kontraście.	

Baza Wiedzy musi zapewnić obsługę standardu Aria	
Baza Wiedzy musi prawidłowo definiować nawigację po elementach serwisu za pomocą "tabindex".	
Baza Wiedzy musi ukrywać w wersji dla niepełnosprawnych zbędne elementy interfejsu, które mogłyby zaburzać pracę czytników bądź nawigację osób niepełnosprawnych po stronie.	
Wersja żałobna serwisu	
Baza Wiedzy musi zawierać specjalny zestaw styli dla wersji żałobnej serwisu, którą można będzie włączyć z poziomu panelu administracyjnego Podsystemu.	
Wersja żałobna serwisu powinna zmieniać także paletę kolorów dla obrazków wyświetlanych na stronie.	
Pozostałe	
Baza Wiedzy musi zawierać skuteczne zabezpieczenie wszystkich formularzy przed spamem o efektywności na poziomie co najmniej 99% (maksymalnie 1% postów może zostać niepoprawnie sklasyfikowanych).	
Wszystkie funkcjonalności Bazy Wiedzy (w tym te związane z administracją oraz dostępem do treści) muszą być realizowane przez przeglądarkę Internetową.	
Baza Wiedzy musi zapewnić mechanizm cache'owania renderowanych szablonów.	
Baza Wiedzy musi udostępnić pełne API, pozwalające na jego kontrolę oraz konfigurację.	
Baza Wiedzy musi integrować się z Podsystemem SSO w celu uwierzytelniania użytkowników.	

29. Trouble Ticketing

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.12 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.12 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
Analiza w zakresie Podsystemu Trouble Ticketing powinna uwzględniać analizę procesów biznesowych związanych z obsługą zgłoszeń.	
Analiza w zakresie Podsystemu Trouble Ticketing powinna zawierać katalog proponowanych rodzajów zgłoszeń.	

Analiza w zakresie Podsystemu Trouble Ticketing powinna zawierać spis atrybutów (pól), właściwych dla poszczególnych rodzajów zgłoszeń.	
Analiza w zakresie Podsystemu Trouble Ticketing powinna zawierać propozycję konfiguracji kolejek zgłoszeń.	
Analiza w zakresie Podsystemu Trouble Ticketing powinna zawierać propozycję polityki SLA do zaimplementowania.	
Analiza w zakresie Podsystemu Trouble Ticketing powinna zawierać propozycję raportów do zaimplementowania.	
Projekt techniczny musi zawierać projekt graficzny interfejsu użytkownika uwzględniający kwestie użyteczności (usablility).	
Obsługa projektów	
Trouble Ticketing musi umożliwiać tworzenie projektów grupujących zgłoszenia.	
Trouble Ticketing musi umożliwiać definiowanie nowych rodzajów zgłoszeń.	
Trouble Ticketing musi umożliwiać określenie, które rodzaje zgłoszeń są dostępne w którym z projektów.	
Trouble Ticketing musi umożliwiać przypisanie użytkownikom ról w ramach projektu.	
Trouble Ticketing musi umożliwiać przypisanie uprawnień do ról w ramach projektu.	
Trouble Ticketing musi umożliwiać przypisanie w ramach projektu konkretnego przepływu pracy (workflow) do rodzaju zgłoszenia.	
Trouble Ticketing musi umożliwiać tworzenie projektów, w ramach których będą realizowane prace programistyczne.	
Trouble Ticketing musi umożliwiać tworzenie projektów, w ramach których będzie realizowana obsługa zgłoszeń serwisowych użytkowników (funkcjonalność "Service Desk")	
Trouble Ticketing musi uniemożliwiać dostęp do informacji o	

projekcie (w tym do zgłoszeń) użytkownikom nieposiadającym odpowiedniego uprawnienia w tym projekcie.	
Tworzenie zgłoszeń	
Trouble Ticketing musi zapewnić możliwość uwierzytelniania użytkowników w oparciu o Podsystem SSO.	
Trouble Ticketing musi zapewnić możliwość utworzenia nowego zgłoszenia.	
Trouble Ticketing musi zapewnić możliwość zdefiniowania listy atrybutów (pól) dostępnych do wypełnienia na formularzu tworzenia nowego zgłoszenia.	
Trouble Ticketing musi zapewnić możliwość podglądu statusu zgłoszenia przez użytkownika, który je utworzył.	
Trouble Ticketing musi umożliwić dołączanie plików/załączników do zgłoszeń.	
Trouble Ticketing musi umożliwić tworzenie powiązań między zgłoszeniami (np. duplikaty, zgłoszenia blokujące itp.)	
Trouble Ticketing musi umożliwić tworzenie zgłoszeń podrzędnych ("podzadania").	
Trouble Ticketing musi umożliwić opisywanie zgłoszeń z wykorzystaniem tagów.	
Trouble Ticketing musi umożliwić definiowanie poziomów bezpieczeństwa zgłoszeń, umożliwiających ograniczenie zbioru użytkowników widzących dane zgłoszenie do wybranych ról projektowych.	
Komentarze	
Trouble Ticketing musi zapewnić możliwość tworzenia komentarzy do zgłoszenia przez twórcę zgłoszenia i pracowników obsługujących zgłoszenie.	
Trouble Ticketing musi umożliwić ograniczenie widoczności komentarzy do wybranej roli w ramach projektu.	
Trouble Ticketing musi umożliwiać śledzenie historii dla	

każdej zmiany dotyczącej zgłoszenia.	
Serwis Klienta	
Trouble Ticketing musi zawierać wydzieloną logicznie część, przeznaczoną do obsługi użytkowników zewnętrznych (Serwis Klienta).	
Serwis Klienta musi być wizualnie zintegrowany z Podsystemem SelfService.	
Integracja między Serwisem Klienta a Podsystemem SelfService nie może bazować na mechanizmie IFRAME.	
Serwis Klienta musi umożliwiać Klientom tworzenie różnych rodzajów zgłoszeń.	
Serwis Klienta musi umożliwiać Klientom tworzenie zgłoszeń we właściwych projektach (np. "wsparcie techniczne", "wsparcie handlowe").	
Serwis Klienta musi zapewnić w formularzu tworzenia zgłoszenia możliwość wybrania z listy obiektów (Usług, faktur, płatności) należących użytkownika tego obiektu, którego dotyczy problem.	
Serwis Klienta musi umożliwiać Klientom wgląd w aktualny status zgłoszenia.	
Serwis Klienta musi umożliwiać Klientom wgląd w przeznaczone dla Klientów komentarze do zgłoszenia.	
Serwis Klienta musi umożliwiać Klientom utworzenie komentarza do zgłoszenia.	
Serwis Klienta musi umożliwiać Klientom dodanie załącznika do zgłoszenia.	
Serwis Klienta musi umożliwiać Klientom oznaczenie zgłoszenia jako wycofane/anulowane.	
Serwis Klienta musi umożliwiać Klientom przeszukiwanie zgłoszeń.	
Serwis Klienta musi umożliwiać określenie, które atrybuty	

(pola) zgłoszenia są widoczne dla Klienta.	
Serwis Klienta Trouble Ticketing i Podsystem SelfService muszą być zaimplementowane i zaprojektowane graficznie w taki sposób, żeby dla użytkownika wyglądały jak elementy jednej i tej samej aplikacji.	
Integracja	
Trouble Ticketing musi być zintegrowany z innymi Podsystemami w zakresie pobierania informacji o Usługach użytkownika.	
Trouble Ticketing musi być zintegrowany z innymi Podsystemami w zakresie pobierania informacji o fakturach użytkownika.	
Trouble Ticketing musi być zintegrowany z Podsystemem SSO w celu uwierzytelniania użytkowników.	
Trouble Ticketing musi udostępniać pełne API udostępniające wszystkie funkcjonalności Podsystemu (zarówno operacyjne jak i administracyjne/konfiguracyjne).	
Trouble Ticketing musi być zintegrowany z mechanizmami obsługującymi środowisko integracyjne (continuous integration).	
Przepływy pracy (workflow)	
Trouble Ticketing musi umożliwiać definiowanie nowych przepływów pracy (workflow).	
Trouble Ticketing musi umożliwiać definiowanie statusów zgłoszeń dostępnych dla każdego przepływu pracy.	
W ramach przepływu pracy Trouble Ticketing musi umożliwiać zdefiniowanie przejść między statusami zgłoszenia.	
W ramach przepływu pracy Trouble Ticketing musi umożliwiać zdefiniowanie formularza wyświetlanego użytkownikowi przy konkretnym przejściem między statusami.	

Jako "zdefiniowanie formularza" należy rozumieć tutaj określenie, jakie atrybuty zgłoszenia będą wyświetlane na formularzu, wraz z możliwością określenia kolejności tych atrybutów i umieszczania ich w osobnych "zakładkach".	
W ramach edycji przepływu pracy Trouble Ticketing musi umożliwiać określenie ról, które mogą dokonać przejścia między statusami zgłoszenia.	
W ramach edycji przepływu pracy Trouble Ticketing musi umożliwiać automatyczne przypisanie zgłoszenia do wskazanej osoby po przejściu między statusami zgłoszenia.	
Trouble Ticketing musi umożliwiać definiowanie i edycję przepływów pracy (workflow) z poziomu przeglądarki internetowej, bez konieczności instalacji jakichkolwiek aplikacji i dodatków na stacji użytkownika.	
Trouble Ticketing musi umożliwiać wdrożenie zdefiniowanych przepływów pracy (workflow) z poziomu przeglądarki internetowej użytkownika.	
Trouble Ticketing, w ramach wdrożenia nowego przepływu pracy, musi przeprowadzić migrację istniejących zgłoszeń do nowego przepływu pracy.	
Trouble Ticketing musi umożliwiać definiowanie przepływów pracy bez prowadzenia prac programistycznych.	
W ramach definiowania przepływów pracy Trouble Ticketing musi umożliwiać utworzenie akcji uaktualnienia wybranych atrybutów (pól) zgłoszenia.	
W ramach definiowania przepływów pracy Trouble Ticketing musi umożliwiać utworzenie akcji wysłania informacji o zgłoszeniu w formacie JSON do zewnętrznego Podsystemu.	
Trouble Ticketing, dla każdego zgłoszenia, musi wizualizować wykorzystywany przepływ pracy w sposób graficzny wraz z zaznaczeniem aktualnego statusu danego zgłoszenia.	

Trouble Ticketing musi umożliwić ponowne otwarcie zamkniętego wcześniej zgłoszenia.	
Trouble Ticketing musi umożliwiać określanie kierowników obszarów merytorycznych, do których są automatycznie przypisywane nowe zgłoszenia w danym obszarze.	
Trouble Ticketing musi umożliwić użytkownikowi przejście między statusami zgłoszenia, zgodnie z posiadanymi uprawnieniami i konfiguracją przepływu pracy.	
Powiadomienia e-mail	
Trouble Ticketing musi wysyłać powiadomienia e-mail o zdarzeniach związanych z cyklem życia zgłoszenia.	
Trouble Ticketing musi umożliwiać określenie w konfiguracji projektu, jakie role projektowe będą otrzymywać powiadomienia o poszczególnych zdarzeniach.	
Trouble Ticketing musi umożliwiać "śledzenie" zgłoszeń przez użytkowników, powodujące powiadomienie ich o zdarzeniach dotyczących danego zgłoszenia niezależnie od konfiguracji powiadomień.	
Trouble Ticketing musi umożliwiać określenie ról, do których będzie wysyłać powiadomienia o przypadkach przekroczenia SLA.	
Trouble Ticketing musi umożliwiać grupowanie wielu powiadomień w jedną wiadomość e-mail (ang. email digest)	
Trouble Ticketing musi umożliwiać indywidualne włączenie/wyłączenie grupowania powiadomień dla każdego użytkownika osobno.	
Obsługa zgłoszeń Klientów przez użytkowników wewnętrznych	
Trouble Ticketing musi umożliwiać przeszukiwanie zgłoszeń.	
Trouble Ticketing musi umożliwiać filtrowanie wyników wyszukiwania zgłoszeń według poszczególnych atrybutów (pól) zgłoszenia.	

Trouble Ticketing musi umożliwiać zapisanie aktualnej konfiguracji wyszukiwania i filtrowania jako "zapisanych filtrów".	
Trouble Ticketing musi umożliwiać udostępnianie "zapisanych filtrów" innym użytkownikom.	
Trouble Ticketing musi umożliwiać definiowanie zapisanych filtrów przy wykorzystaniu języka skryptowego, umożliwiającego co najmniej: filtrowanie wg atrybutów zgłoszenia (operatory: równy, nierówny, większe, większe lub równe, mniejsze, mniejsze lub równe, należące do zbioru wartości, nienależący do zbioru wartości, zawierający tekst, niezawierający tekstu), stosowanie operatorów AND OR NOT EMPTY NULL, sortowanie według atrybutów, a także filtrowanie według historycznych wartości atrybutów (atrybut ma obecnie lub miał w przeszłości daną wartość).	
W przypadku części Trouble Ticketing przeznaczonej dla użytkowników wewnętrznych (obsługa klienta, inżynierowie wsparcia technicznego, developerzy itp.), Zamawiający może - na etapie akceptacji projektu technicznego - częściowo odstąpić od wymagania zapewnienia pełnej spójności graficznej wszystkich Podsystemów.	
W części przeznaczonej dla użytkowników wewnętrznych Trouble Ticketing musi udostępniać konfigurowalny przez użytkownika panel kontrolny (dashboard).	
Panel kontrolny musi umożliwiać użytkownikowi osadzanie dowolnej liczby instancji widgetów.	
Trouble Ticketing musi udostępniać widget dla panelu kontrolnego pozwalający wyświetlać wyniki wyszukiwania zrealizowane w oparciu o zapisane wcześniej filtry	
Trouble Ticketing musi udostępniać widget dla panelu kontrolnego pozwalający wyświetlać zgłoszenia przypisane do	

aktualnie zalogowanego użytkownika.	
Trouble Ticketing musi udostępniać widget dla panelu kontrolnego pozwalający wyświetlać zawartość poszczególnych kolejek zgłoszeń.	
Trouble Ticketing musi udostępniać widget dla panelu kontrolnego pozwalający wyświetlać wykresy obrazujące liczbę realizowanych zgłoszeń.	
Trouble Ticketing musi umożliwiać "udostępnienie" zgłoszenia - wysłanie e-maila z linkiem do zgłoszenia do wybranego użytkownika.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi prezentować użytkownikom wewnętrznym pełne informacje o usłudze, której dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi prezentować użytkownikom wewnętrznym pełne informacje o fakturach za Usługę, której dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi prezentować użytkownikom wewnętrznym pełne informacje o płatnościach za Usługę, której dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać użytkownikom wewnętrznym możliwość wyświetlenia pełnych informacji dotyczących wszystkich Usług użytkownika, którego dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać użytkownikom wewnętrznym możliwość wyświetlenia pełnych informacji dotyczących wszystkich faktur użytkownika, którego dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać użytkownikom wewnętrznym możliwość wyświetlenia pełnych informacji dotyczących wszystkich płatności użytkownika, którego dotyczy zgłoszenie.	

Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać użytkownikom wewnętrznym możliwość wyświetlenia pełnych informacji dotyczących wszystkich zgłoszeń użytkownika, którego dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać użytkownikom wewnętrznym możliwość wyświetlenia pełnych informacji dotyczących konta użytkownika, którego dotyczy zgłoszenie.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać możliwość przejścia bezpośrednio do obiektu (np. Usługi, faktury, płatności, konta użytkownika) w ramach odpowiedniego Podsystemu dziedzinowego, którego dotyczy przedmiot zgłoszenia.	
Trouble Ticketing, przy wyświetlaniu zgłoszenia, musi udostępniać możliwość zmiany danych użytkownika zewnętrznego, który utworzył zgłoszenie.	
Trouble Ticketing, prezentując informacje o Usługach, użytkownikach, fakturach i płatnościach, musi uwzględnić uprawnienia do tych zasobów posiadane przez aktualnie zalogowanego użytkownika.	
Trouble Ticketing musi umożliwić przekazanie zgłoszenia do realizacji innemu użytkownikowi, o ile aktualny użytkownik ma odpowiednie uprawnienia.	
Ewidencja czasu pracy	
Trouble Ticketing musi zapewnić ewidencjonowanie czasu poświęconego na realizację danego zgłoszenia na każdym etapie jego realizacji.	
Trouble Ticketing musi zapewniać integrację z Podsystemem Knowledge Base, umożliwiając w łatwy sposób wskazywanie artykułów z bazy wiedzy w komentarzach zgłoszeń (wyświetlanie pełnej struktury bazy wiedzy, łatwe	

przeszukiwanie struktury i osadzanie automatycznie linków po wybraniu danego elementu drzewa bazy wiedzy).	
Liczba użytkowników	
Pierwsza wersja Podsystemu Trouble Ticketing musi zapewniać możliwość pracy dla minimum 100 nazwanych użytkowników.	
Ostateczna wersja Podsystemu Trouble Ticketing (na dzień odbioru końcowego Projektu) musi zapewniać możliwość pracy dla minimum 500 nazwanych użytkowników.	
Obsługa kolejek i SLA	
Trouble Ticketing musi umożliwiać definiowanie kolejek zgłoszeń.	
Trouble Ticketing musi umożliwiać określenie, z wykorzystaniem języka skryptowego, jakie zgłoszenia trafiają do poszczególnych kolejek zgłoszeń.	
Trouble Ticketing musi umożliwiać określenie, jakie atrybuty zgłoszenia są wyświetlane w widoku kolejki zgłoszeń.	
Trouble Ticketing musi umożliwiać pracownikowi pobranie z kolejki zgłoszenia do realizacji.	
Trouble Ticketing musi umożliwiać definiowanie metryk SLA.	
Trouble Ticketing musi umożliwiać określenie zdarzeń (co najmniej: utworzenie zgłoszenia, zmiana statusu, utworzenie komentarza, przypisanie zgłoszenia do użytkownika, zamknięcie zgłoszenia) rozpoczynających, wstrzymujących i kończących liczenie czasu dla danej metryki SLA.	
Trouble Ticketing musi umożliwiać określenie zdarzeń (co najmniej: utworzenie zgłoszenia, zmiana statusu, utworzenie komentarza, przypisanie zgłoszenia do użytkownika, zamknięcie zgłoszenia), wstrzymujących (pauzujących) liczenie czasu dla danej metryki SLA.	
Trouble Ticketing musi umożliwiać określenie zdarzeń (co	

najmniej: utworzenie zgłoszenia, zmiana statusu, utworzenie komentarza, przypisanie zgłoszenia do użytkownika, zamknięcie zgłoszenia) kończących liczenie czasu dla danej metryki SLA.	
Trouble Ticketing musi umożliwiać określenie wymaganych czasów reakcji dla zgłoszeń spełniających określone kryteria (zdefiniowane z wykorzystaniem języka skryptowego).	
Trouble Ticketing musi umożliwiać definiowanie kalendarzy określających w jakie dni i w jakich godzinach liczony jest upływ czasu reakcji.	
Trouble Ticketing musi umożliwiać definiowanie w kalendarzach SLA dni świątecznych, w których upływ czasu reakcji nie jest liczony.	

30. Centralny system logów

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.13 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.13 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
Centralny system logów musi umożliwiać zbieranie logów przesłanych za pomocą następujących formatów danych: syslog, json.	
Centralny system logów musi umożliwiać podłączanie (za pomocą API/modułów/pluginów) innych formatów danych wejściowych.	
Centralny system logów musi umożliwiać zbieranie logów zarówno z infrastruktury sprzętowej jak i Podsystemów oprogramowania.	
Centralny system logów musi zapewniać pełno tekstową wyszukiwarkę logów poprzez GUI oraz API z zachowaniem filtrowania, min. źródło, data, poziom alertu, treść.	
Centralny system logów musi zapewniać parsowanie i katalogowanie przechowywanych logów.	
Centralny system logów musi zapewniać archiwizację, rotację i kompresję logów.	
Centralny system logów musi zapewniać szybkie	

wyszukiwanie logów poprzez wykorzystanie mechanizmów indeksujących (np. poprzez elasticsearch).	
Centralny system logów musi zapewniać zarządzanie dostępem do poszczególnych źródeł logów oraz poziomów logowania.	
Centralny system logów musi umożliwiać eksport informacji o określonych rodzajach logów poprzez mechanizm triggeringu do Podsystemu NOC-monitoring (na potrzeby wygenerowania alarmu).	
Centralny system logów musi umożliwiać definiowanie reguł filtrowania logów w czasie rzeczywistym (oraz wyświetlania ich bez przeładowania strony - AJAX) oraz zapisywania tych reguł.	
Centralny system logów musi umożliwiać definiowanie alertów, których spełnienie spowoduje wysłanie powiadomienia e-mail do administratora.	
Centralny system logów musi zawierać panel kontrolny (dashboard) umożliwiający osadzanie widgetów, w których wyświetlane są logi spełniające określone kryteria wyszukiwania bądź wykresy bazujące na tych kryteriach.	
Centralny system logów musi umożliwiać definiowanie kryteriów wyszukiwania według poszczególnych atrybutów wpisu w logach (np. data, serwer, treść komunikatu itp.).	
Centralny system logów musi umożliwiać definiowanie kryteriów wyszukiwania z wykorzystaniem wyrażeń regularnych.	

31. NOC: Monitoring

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.14 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.14 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
Oprogramowanie do monitoringu musi umożliwiać przedstawienie w formie graficznej statystyk dla wszystkich istotnych elementów sieci, takich jak: - przepustowość łącz pomiędzy poszczególnymi urządzeniami, - przepustowość łącz do operatorów, - temperaturę pracy urządzeń, - utylizację pracy urządzeń oraz komponentów w urządzeniach:	

- utylizacja danej karty, - wolne/zajęte miejsce na dysku, - wolna/zajęta pamięć, - wolne/zajęte przerwania, - inne, istotne z punktu widzenia szacowania obciążenia danego urządzenia. - statystykę błędów na danym porcie/karcie, - mapę połączeń wraz z procentowym zajęciem danego łącza, - statystykę zasilania urządzenia.	
Oprogramowanie do monitoringu musi umożliwiać wykrywanie sieci: - automatyczne wykrywanie urządzeń działających zgodnie z protokołem TCP/IP, - automatyczne tworzenie mapy przedstawiającej logiczną strukturę sieci, - półautomatyczne tworzenie mapy przestawiającej fizyczną strukturę sieci, - wykrywanie usług sieciowych, - automatyczna klasyfikacja urządzeń i usług (funkcja, typ, producent);	
Oprogramowanie do monitoringu musi umożliwiać budowanie konfiguracji Systemu na podstawie automatycznie wykrytych elementów sieci.	
Oprogramowanie do monitoringu musi zapewniać wsparcie dla protokołów SNMP (v2 i v3) oraz ICMP.	
Oprogramowanie do monitoringu musi zapewniać możliwość importowania własnych baz MIB SNMP do Podsystemu.	
Oprogramowanie do monitoringu musi zapewniać umożliwianie monitorowania wszystkich istotnych parametrów urządzeń poprzez możliwość tworzenia własnych wtyczek, template'ów i skryptów.	

Oprogramowanie do monitoringu musi zapewniać możliwość tworzenia tzw. liczników wirtualnych – liczników opartych o wartości liczników rzeczywistych powiązanych operacjami matematycznymi.	
Oprogramowanie do monitoringu musi zapewniać możliwość samodzielnej konfiguracji wyglądu alarmów, układu alarmów, sposobu ich reprezentacji.	
Oprogramowanie do monitoringu musi zapewniać możliwość tworzenia własnych kwerend SNMP.	
Oprogramowanie do monitoringu musi dostarczać wiele metod raportowania: - raportowanie przepływności, - raportowanie dostępności, - raportowanie czasów odpowiedzi, - raportowanie obciążenia procesorów, - raportowanie zajętości powierzchni, - raportowanie historyczne, - raportowanie w czasie rzeczywistym.	
Oprogramowanie do monitoringu musi zapewniać możliwość generowania raportów przekrojowych i skrośnych.	
Oprogramowanie do monitoringu musi zapewniać możliwość generowania raportów automatycznych, zgodnie z ustalonym terminarzem.	
Oprogramowanie do monitoringu musi zapewniać możliwość eksportowania raportów i zebranych danych do formatu CSV.	
Oprogramowanie do monitoringu musi zapewniać współpracę z relacyjną bazą danych,	
Oprogramowanie do monitoringu musi zapewniać wsparcie dla hierarchicznego modelu uprawnień.	
Oprogramowanie do monitoringu musi zapewniać współpracę z operatorem poprzez przeglądarkę WWW z możliwością	

stworzenia spersonalizowanej strony domowej.	
Oprogramowanie do monitoringu musi zapewniać umożliwianie graficznego przedstawienia schematu sieci, wraz umieszczeniem informacji o jej aktualnych parametrach.	
Oprogramowanie do monitoringu musi zapewniać reagowanie na określone wcześniej graniczne parametry i po ich przekroczeniu.	
Oprogramowanie do monitoringu musi zapewniać wysyłanie powiadomień - informowanie o wystąpieniu zdarzenia, wraz z określeniem poziomu alarmu: - na monitorze, - poprzez wysyłanie wiadomości e-mail, - poprzez wysyłanie wiadomości SMS (poprzez zintegrowane urządzenie dedykowane GSM).	
Oprogramowanie do monitoringu musi zapewniać możliwość wysłania powiadomień gdy zajdą określone wcześniej zdarzenia.	
Oprogramowanie do monitoringu musi zapewniać możliwość określenia użytkownika bądź grupy użytkowników, do których wysyłany jest komunikat na podstawie zdefiniowanych filtrów wysyłania powiadomień.	
Oprogramowanie do monitoringu musi zapewniać możliwość integracji z systemem paszportyzacji.	
Oprogramowanie do monitoringu musi zapewniać wsparcie dla ściany wizyjnej.	
Oprogramowanie do monitoringu musi zapewniać wsparcie dla systemu dodatków (wtyczek), możliwość tworzenia dodatków min. w językach: C, python, perl, bash, php, java.	
Oprogramowanie do monitoringu musi zapewniać monitorowanie sprzętu poprzez protokół IPMI.	
Oprogramowanie do monitoringu musi zapewniać możliwość	

obsługi map przy wykorzystaniu bibliotek D3 lub dowolnych bibliotek służących do wizualizacji danych.	
Oprogramowanie do monitoringu musi zapewniać dostęp do API, możliwość tworzenia własnych modułów do Podsystemu, personalizację Podsystemu.	
Oprogramowanie do monitoringu musi zapewniać możliwość monitorowania odseparowanych systemów poprzez serwery proxy.	
Oprogramowanie do monitoringu musi zapewniać wsparcie dla IPv6/dual stack, Podsystem musi być dostępny poprzez adresację IPv4 oraz IPv6.	
Oprogramowanie do monitoringu musi zapewniać wsparcie dla LDAP.	
Logowanie użytkowników do Podsystemu monitoringu musi zostać zrealizowane na podstawie autoryzacji z centralną bazą danych (np. LDAP), wspólną przynajmniej dla: Podsystemu monitoringu, Podsystemu zarządzania i Podsystemu DCIM.	
Oprogramowanie musi zostać zintegrowane z dedykowanym urządzeniem do wysyłania powiadomień SMS (infrastruktura: bramka SMS).	
Oprogramowanie do monitoringu musi zostać skonfigurowane do monitorowania całej infrastruktury oraz wszystkich Podsystemów oprogramowania.	
Oprogramowanie do monitoringu musi umożliwiać identyfikację każdego z urządzeń w Podsystemie NOC-DCIM (np. poprzez odnośnik).	

32. NOC: Podsystem zarządzania

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.15 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.15 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	**Oferowana funkcjonalność:**
NOC: Podsystem zarządzania musi umożliwiać konfigurację, zarządzanie oraz ewidencję wszystkich urządzeń pracujących w ramach infrastruktury Centrum Danych. Podsystem może stanowić jeden spójny system zarządzania lub zbiór systemów zarządzania dostarczonych przez producentów poszczególnych urządzeń wdrożonych w ramach infrastruktury.	
W ramach NOC: Podsystem zarządzania dopuszcza się	

wdrożenie wielu rozwiązań dostarczanych przez producentów dostarczanego sprzętu, przy czym w przypadku wdrożenia kilku systemów zarządzania należy zadbać o ich spójność w zakresie bazy danych użytkowników i uprawnień (wspólna baza danych użytkowników i uprawnień) np. w oparciu o LDAP.	
W ramach NOC: Podsystem zarządzania wszystkie zintegrowane systemy zarządzania muszą umożliwiać śledzenie zmian w konfiguracji sprzętu który obsługują z zachowaniem spójnej informacji: kto dokonał zmiany, jaka to była zmiana i kiedy nastąpiła. Rekomendowane jest wykorzystanie systemu GIT.	
W ramach NOC: Podsystem zarządzania wymagane jest cykliczne realizowanie kopii zapasowej konfiguracji całej infrastruktury (kopie konfiguracji poszczególnych urządzeń). Kopia zapasowa konfiguracji musi być niezależna od systemów zarządzania (odtworzenie konfiguracji z takiej kopii powinno być możliwe również bez pośrednictwa systemu zarządzania)	
Składowe elementy NOC: Podsystem zarządzania musi być w pełni funkcjonalne w odniesieniu do wymagań i nie powinny posiadać żadnych ograniczeń wynikających z braku licencji.	
Każdy system zarządzania wchodzący w skład oprogramowania NOC: Podsystem zarządzania musi być dostarczony wraz systemem operacyjnym na jakim pracuje, jeśli takiego wymaga; jeśli jest wymagana dodatkowa licencja do oprogramowania zarządzającego np. licencja systemu operacyjnego, musi być ona dostarczona razem z oprogramowaniem zarządzającym.	
Logowanie użytkowników do NOC: Podsystem zarządzania musi zostać zrealizowane na podstawie autoryzacji z centralną bazą danych (np. LDAP), wspólną przynajmniej dla:	

Podsystemu monitoringu, Podsystemu zarządzania i Podsystemu DCIM.	

33. NOC: DCIM

Czy oprogramowanie bazuje na dostępnym produkcie (nazwa / wersja / wariant)?

.....

Opis realizacji w przypadku oprogramowania dedykowanego (jeżeli dotyczy):

.....

Deklarowany stopień pokrycia wymagań z części 5.2.16 OPZ przy użyciu oprogramowania open source (jeżeli dotyczy):

.....

Wykorzystane oprogramowanie w celu pokrycia zadeklarowanego przez Wykonawcę stopnia pokrycia wymagań z części 5.2.16 OPZ oprogramowaniem open source (jeżeli dotyczy):

.....

Sposób licencjonowania rozwiązania:

.....

Przekazanie autorskich praw majątkowych do rozwiązania (TAK/NIE)

.....

Przekazanie kodów źródłowych (TAK/NIE)

.....

Pozostałe informacje:

.....

Wymagana funkcjonalność:	Oferowana funkcjonalność:
DCIM musi zapewniać inwentaryzację sprzętu i jego identyfikację w szafach.	
DCIM musi zapewniać monitoring zajętości szaf.	
DCIM musi obsługiwać rezerwację miejsca w szafach.	
DCIM musi zapewniać monitoring obciążenia CDU/PDU: - z pomiaru - teoretyczny	
DCIM musi zapewniać śledzenie połączeń Ethernet.	
DCIM musi zapewniać śledzenie połączeń elektrycznych.	
DCIM musi raportować bilans cieplny.	
DCIM musi zapewniać monitoring użycia portów switchy.	
DCIM musi zapewniać graficzną prezentację serwerowni.	
DCIM musi wspierać protokół SNMP v1/2c do monitorowania temperatury i wilgotności.	
DCIM musi wspierać protokół SNMP v1/2c do monitorowania PDU/CDU.	
DCIM musi być w pełni niezależne od producentów PDU/CDU/czujników.	
DCIM musi wspierać raportowanie zasobów.	
DCIM musi zapewniać raport kosztów pracy DC w tym: koszt miejsca, koszt energii elektrycznej.	
DCIM musi zapewniać interfejs WWW.	
DCIM musi zostać zintegrowane pod względem obsługi alarmów i monitoringu do Podsystemu NOC-monitoring.	
DCIM musi zostać zintegrowane z istniejącą infrastrukturą pasywną Zamawiającego.	
Logowanie użytkowników do DCIM musi zostać zrealizowane na podstawie autoryzacji z centralną bazą danych (np. LDAP), wspólną przynajmniej dla: Podsystemu monitoringu, Podsystemu zarządzania i Podsystemu DCIM.	

....., dn. .....

.....

(podpis(y) osób uprawnionych do reprezentacji wykonawcy, w przypadku oferty wspólnej- podpis pełnomocnika wykonawców)