

Warszawa, dn. 24.06.2026 r.
KIGEiT/1149/06/2026

Ministerstwo Cyfryzacji

ul. Królewska 27
00-060 Warszawa

Dotyczy: Konsultacji założeń konkursu na wdrożenie open source.

Szanowni Państwo,

w odpowiedzi na opublikowane przez Ministerstwa Cyfryzacji zaproszenie do konsultacji¹ w sprawie planowanego konkursu Ministerstwa Cyfryzacji w zakresie pilotażowych wdrożeń oprogramowania open source w jednostkach samorządu terytorialnego, Sekcja AI Poland działająca w ramach Krajowej Izby Gospodarczej Elektroniki i Telekomunikacji (SAIPL²) przedkłada stanowisko w przedmiotowej sprawie.

Pytanie	Odpowiedź
1. W jaki sposób dobrze zdefiniować open source na potrzeby konkursu? Czy definicja powinna odwoływać się do konkretnych licencji lub zapisów licencyjnych?	Właściwe zdefiniowanie oprogramowania open source (OSS) w konkursie dla Jednostek Samorządu Terytorialnego (JST) ma kluczowe znaczenie prawne i technologiczne. Błędna lub zbyt ogólna definicja może otworzyć drzwi dla podmiotów oferujących rozwiązania typu pseudo-open-source (np. na licencjach source-available lub Business Source License), które ograniczają swobodę dysponowania kodem, narzucają ukryte opłaty lub limitują liczbę użytkowników. Rekomendujemy, aby Ministerstwo Cyfryzacji skonstruowało definicję dwupoziomową: opartą na kryteriach funkcjonalno-prawnych oraz referencyjnym katalogu uznanych licencji. Kryteria funkcjonalno-prawne (Cztery wolności i standard OSI) Definicja w regulaminie konkursu powinna wprost odwoływać się do Open Source Definition (OSD) wypracowanej przez <i>Open Source Initiative (OSI)</i>. Zapewni to pełną zgodność z unijnymi ramami prawnymi, w tym z aktem w sprawie interoperacyjnej Europy (Interoperable Europe Act) Oparcie się na OSD gwarantuje spełnienie kluczowych postulatów suwerenności cyfrowej: • Brak uzależnienia od jednego dostawcy: OSD gwarantuje prawo do swobodnego używania, modyfikowania i dystrybucji oprogramowania przez dowolny podmiot. • Bezpieczeństwo łańcucha dostaw: Możliwość pełnego audytu kodu źródłowego przez państwowe instytucje certyfikujące. • Rozwój kompetencji: Otwartość na modyfikacje wspiera budowę lokalnego know-how w polskim sektorze IT. Wdrożone w samorządach oprogramowanie musi gwarantować JST cztery fundamentalne wolności:

¹ <https://www.gov.pl/web/cyfryzacja/rozpoczynamy-konsultacje-zalozen-konkursu-na-wdrozenie-open-source>

² <https://kigeit.org.pl/sekcja-ai-poland-kigeit-saipl/>

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	1. Wolność uruchamiania programu w dowolnym celu (brak limitów użytkowników czy stanowisk w urzędzie). 2. Wolność analizowania działania programu oraz jego modyfikacji (pełen dostęp do kodu źródłowego). 3. Wolność rozpowszechniania kopii (możliwość bezpłatnego przekazania systemu innemu samorządowi). 4. Wolność ulepszania programu i publicznego udostępniania zmian (wspieranie zasady <i>Public Money, Public Code</i>). Definicja powinna wykluczać licencje, które zabraniają komercyjnego wykorzystania lub ograniczają wdrażanie kodu w chmurze przez podmioty trzecie. Samorząd musi mieć prawo zlecić wsparcie techniczne dowolnemu lokalnemu MŚP, a nie tylko pierwotnemu twórcy kodu. Zwracamy jednak szczególną uwagę, że na potrzeby administracji publicznej definicja ta powinna zostać rozszerzona o pojęcie Open Source klasy korporacyjnej (Enterprise Open Source). Czysty kod społecznościowy (tzw. community) często nie spełnia rygorystycznych wymogów państwa, w szczególności w zakresie określonym przez dyrektywę NIS-2. Dlatego konkurs powinien premiować rozwiązania Open Source, ale takie, za którymi stoi profesjonalny producent gwarantujący m.in.: • Umowy o gwarantowanym poziomie świadczenia usług (SLA). • Długoterminowe wsparcie techniczne (Long Term Support - LTS). • Krytyczne poprawki bezpieczeństwa (reakcja na podatności CVE w określonym czasie). • Certyfikacje bezpieczeństwa wymagane w infrastrukturze krytycznej (np. Common Criteria, FIPS 140-3). Odwołanie do konkretnych zapisów licencyjnych Definicja powinna odwoływać się do konkretnych, powszechnie uznanych licencji. Rekomendujemy wskazanie, że oprogramowanie kwalifikujące się do pilotażu musi być udostępniane na jednej z licencji zatwierdzonych przez OSI lub Komisję Europejską. Ministerstwo powinno unikać tworzenia własnych katalogów "dozwolonych zapisów", ponieważ prawo autorskie i licencjonowanie oprogramowania to bardzo skomplikowana materia prawna. Określenie wymogu zgodności z istniejącymi licencjami OSI (GPL, Apache, MIT, itd.) rozwiązuje ten problem systemowo. Ważne jest, aby konkurs odróżniał i wykluczał licencje typu "Source-Available" (np. BSL - Business Source License czy SSPL - Server Side Public License). Nie są to licencje Open Source, ponieważ wprowadzają ograniczenia co do komercyjnego wykorzystania lub skali wdrożenia. Dopuszczenie takich pseudo-otwartych licencji stworzyłoby ryzyko ukrytego uzależnienia ("vendor lock-in"), co stoi w bezpośredniej sprzeczności z celem budowania suwerenności cyfrowej państwa. Zwracamy też uwagę na ryzyko, które pojawiło się niedawno na rynku rozwiązań Enterprise. Sama zgodność licencji oprogramowania z definicją OSI przestała być gwarantem otwartości ponieważ pojawiły się praktyki, w których globalni dostawcy, oferując oprogramowanie na wolnych licencjach (np. GPL), wprowadzają do umów świadczenia usług (tzw. Terms of Service lub EULA) zapisy, które de facto te wolności odbierają. Dla administracji państwowej kluczowa jest możliwość swobodnego współdzielenia kodu. Jeśli np. Centralny Ośrodek Informatyki (COI) zmodyfikuje otwarty system operacyjny na własne potrzeby, powinien mieć prawo udostępnić ten kod innym jednostkom (np. Ministerstwu Finansów czy Zdrowia) bez żadnych konsekwencji. Analogicznie, instytucje odpowiadające za cyberbezpieczeństwo kraju (np. CERT Polska, CSIRT NASK, CSIRT GOV) muszą mieć możliwość nieprzerwanego, niezależnego audytowania kodu źródłowego komponentów budujących infrastrukturę krytyczną. Zastosowanie praktyk około licencyjnych przez niektórych dostawców sprawia, że skorzystanie z tego podstawowego

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	prawa skutkuje natychmiastowym wypowiedzeniem umowy subskrypcyjnej, odcięciem od krytycznych łatek bezpieczeństwa i utratą wsparcia technicznego. Taki model czyni państwo zakładnikiem dostawcy w zakresie, który powinien być chroniony licencją Open Source. Dlatego naszą rekomendacją jest dodanie dwóch dodatkowych wymogów: • Oprogramowanie oraz towarzyszący mu model świadczenia usług (w tym umowy subskrypcyjne i SLA) nie może zawierać klauzul zniechęcających do redystrybucji kodu (tzw. Anti-Redistribution Clauses). • Wymagane jest, aby pełny kod źródłowy rozwiązań Open Source (w tym wprowadzane na bieżąco poprawki bezpieczeństwa) był publicznie i bezwarunkowo dostępny w otwartych repozytoriach, bez konieczności posiadania aktywnej subskrypcji komercyjnej u dostawcy. Wartą poruszenia kwestią jest też niezależność od rodzaju infrastruktury - zapewnienie, że oprogramowanie może być uruchamiane na powszechnie dostępnym sprzęcie, oraz w dowolnym modelu wdrożenia (on-prem, private cloud, air-gap itp.) Dlaczego precyzyjna definicja chroni projekt? Precyzyjne ramy licencyjne pozwolą uniknąć problemów systemowych, które opisuje m.in. raport fundacji Instra z 2025 r.. Raport ten ujawnił, że polski sektor publiczny rutynowo wpisuje w dokumentację zamówieniową (SWIZ/SWZ) nazwy konkretnych, zamkniętych marek (np. Microsoft czy Google), co de facto wyklucza uczciwą konkurencję. Poprawna definicja open source w tym konkursie przyniesie realne korzyści: • Uniemożliwi ukryty vendor lock-in: gwarantuje, że po roku darmowego wsparcia finansowanego z konkursu, gmina nie zostanie zmuszona do płacenia wysokich sum za licencje stanowiskowe, ponieważ kod pozostanie jej własnością. • Zapewnia zgodność z Cyber Resilience Act (CRA): otwarte licencje ułatwiają generowanie transparentnej specyfikacji oprogramowania (SBOM – Software Bill of Materials), co podnosi odporność cyfrową samorządu. • Umożliwia replikację (Efekt skali): jeśli jedno JST wypracuje świetną konfigurację chmury prywatnej i pakietu biurowego (na wzór udanej migracji Danii na LibreOffice w 2025 r.), precyzyjna otwarta licencja pozwoli bezkosztowo skopiować to rozwiązanie do innych polskich gmin. Wprowadzenie wymogu, by oprogramowanie posiadało licencję zgodną z listą OSI lub unijną licencją EUPL, to najprostszy i najskuteczniejszy bezpiecznik prawny, jaki Ministerstwo Cyfryzacji może zastosować w regulaminie.
2. Jakie czynniki będą służyły skuteczności wdrożeń?	Wzorując się na europejskich doświadczeniach oraz wnioskach zawartych w stanowisku KIGEIT na temat Open Source złożonym do Komisji Europejskiej w 2026 roku³ skuteczność wdrożeń w Jednostkach Samorządu Terytorialnego (JST) będą determinować następujące czynniki: a. Przełamanie luki produktowej poprzez wsparcie klasy Enterprise • Zapewnienie SLA i dokumentacji: samo doskonale oprogramowanie to za mało; urzędy potrzebują gotowego, stabilnego produktu. Skuteczne wdrożenie wymaga profesjonalnej dokumentacji technicznej, intuicyjnych interfejsów użytkownika oraz jasno zdefiniowanych umów o gwarantowanym poziomie świadczenia usług (SLA). • Rola lokalnego ekosystemu IT: choć Ministerstwo Cyfryzacji finansuje wsparcie przez pierwszy rok, kluczowe jest zaangażowanie lokalnych dostawców usług IT i MŚP, którzy będą w stanie przejąć utrzymanie

³ <https://kigeit.org.pl/2026/02/05/stanowisko-kigeit-ws-european-open-digital-ecosystem-strategy/>

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	systemów w kolejnych latach. Zapobiegnie to powrotowi do rozwiązań zamkniętych po zakończeniu finansowania projektu. b. Kompleksowe zarządzanie zmianą i szkolenia (czynniki ludzkie) • Wychodzenie z przyzwyczajenia: urzędnicy od lat są szkoleni na narzędziach konkretnych dostawców (tzw. <i>classroom-to-cloud pipeline</i>), co rodzi naturalny opór przed zmianą. Pilotaż nie może być tylko operacją IT – musi być projektem edukacyjnym. • Doświadczenia międzynarodowe: przykłady takie jak udana migracja duńskiego Ministerstwa Cyfryzacji na LibreOffice w 2025 r. pokazują, że kluczem do sukcesu jest nacisk na szkolenia z transferowalnych umiejętności cyfrowych i płynne dostosowanie codziennych procedur (workflow), a nie tylko instalacja nowych programów. c. Pakietyzacja i otwarta interoperacyjność • Brak opłat łącznikowych: ponieważ konkurs kładzie nacisk na rozwiązania pakietowe, kluczowa jest bezszwowa wymiana danych między chmurą prywatną, pocztą a edytorem tekstu. Wykorzystanie otwartych standardów i otwartych API eliminuje tzw. <i>connector fees</i> (opłaty za integrację), które w systemach zamkniętych często sztucznie podnoszą koszty. • Wspólne repozytorium kodu: samorządy biorące udział w pilotażu nie powinny działać w izolacji. Efektywne wdrożenie zakłada, że wypracowane konfiguracje i poprawki będą trafiać do wspólnego, publicznego repozytorium (zgodnie z zasadą <i>Public Money, Public Code</i>), dzięki czemu kolejna gmina będzie mogła wdrożyć ten sam pakiet o wiele szybciej i taniej. d. Centralna inwentaryzacja i koniec z silosami • Model SIST: skuteczność wdrożenia wymaga pełnej przejrzystości w zakresie posiadanych już zasobów i licencji IT. Wykorzystanie systemów na wzór polskiego SIST (System Inwentaryzacji Systemów Teleinformatycznych) pozwala JST precyzyjnie zmierzyć całkowity koszt posiadania (TCO) infrastruktury i uniknąć dublowania kosztów w różnych wydziałach urzędu. • Agregacja popytu: sukces pilotażu wzrośnie, jeśli samorządy będą mogły wspólnie zamawiać usługi wsparcia dla wdrożonego pakietu open source, zyskując silniejszą pozycję negocjacyjną wobec firm wdrożeniowych. e. Dostosowanie kryteriów oceny ofert (zastąpienie metryk korporacyjnych) • Zdrowie ekosystemu zamiast stabilności jednej firmy: tradycyjne przetargi w JST często wymagają od oferenta ogromnych obrotów finansowych, co dyskwalifikuje mniejsze, wyspecjalizowane firmy wdrożeniowe open source. • Nowe wskaźniki: skuteczny konkurs musi oceniać oferty pod kątem dojrzałości ekosystemu oprogramowania (częstotliwość aktualizacji kodu, zaangażowanie społeczności, łatwość migracji danych) oraz jakości planu zarządzania podatnościami na zagrożenia (cyberbezpieczeństwo), a nie samej wielkości korporacji stojącej za produktem. f. Unikanie pułapki „Integrator Lock-in” Uciekając od uzależnienia od wielkich, globalnych korporacji dostarczających oprogramowanie zamknięte (vendor lock-in), instytucje publiczne często wpadają w sidła jednego, lokalnego wykonawcy, który jako jedyny rozumie stworzony dla urzędu kod. Dlatego kryteria konkursowe muszą wymagać, aby tworzony lub modyfikowany kod był dobrze udokumentowany i zdeponowany w publicznie (lub administracyjnie) dostępnych repozytoriach (np. na platformach typu GitLab).

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	Architektura musi być modułowa i oparta na otwartych standardach wymiany danych (API). Dzięki temu urząd zawsze ma możliwość zmiany firmy serwisującej system na inną na konkurencyjnym, europejskim rynku. g. Powołanie OSPO (Open Source Program Office) i standardy reużywalności Brak centralnej koordynacji sprawia, że różne urzędy dublują swoją pracę, tworząc izolowane "wyspy" systemowe. Utworzenie w strukturach administracji państwowej (np. przy Ministerstwie Cyfryzacji lub NASK) tzw. OSPO, który powinien katalogować sprawdzone, bezpieczne oprogramowanie Open Source, dbać o audyty bezpieczeństwa kodu i stosować zasadę „Public Money, Public Code”. Jeśli jedna gmina sfinansuje wdrożenie modułu w otwartym kodzie, każda inna jednostka w Polsce powinna móc z niego skorzystać bez ponoszenia dodatkowych kosztów. h. Zmiana modelu finansowania (Świadomość TCO) Błędne przekonanie, że Open Source jest rozwiązaniem „darmowym”, prowadzi do niedofinansowania utrzymania systemów i ich powolnej degradacji. Z tego powodu administracja powinna przyjąć model Całkowitego Kosztu Posiadania (TCO). Pieniądze z licencji należy reinwestować w polski i europejski rynek IT – w usługi wdrożeniowe, audyty bezpieczeństwa, dostosowywanie kodu (customizację) oraz wsparcie techniczne zapewniane przez europejskie firmy. To stymuluje naszą gospodarkę i buduje faktyczną, technologiczną niezależność Europy. i. Rewolucja w Zarządzaniu Zmianą (Change Management) Doświadczenia takich miast jak Monachium (projekt LiMux) udowodniły, że technologia to zaledwie 20% sukcesu, a 80% to czynnik ludzki. Dlatego należy zadbać o przesunięcie środków zaoszczędzonych na licencjach na intensywne szkolenia, helpdesk oraz materiały edukacyjne. Użytkownicy (urzędnicy) muszą czuć, że nowe narzędzia pomagają im w pracy, a w razie problemów otrzymają natychmiastową pomoc. j. Strategia iteracyjna: Od infrastruktury do użytkownika Najczęstszym błędem we wdrożeniach Open Source jest rozpoczynanie od wymiany oprogramowania biurowego na komputerach urzędników, co natychmiast generuje ogromny opór przed zmianą nawyków. Dlatego pierwsze, pokazowe wdrożenia powinny dotyczyć warstwy serwerowej, bazodanowej i chmurowej (np. Linux, PostgreSQL, Kubernetes, rozwiązania typu Nextcloud do wymiany plików). Dla użytkownika końcowego zmiana na zapleczu jest niewidoczna, a administracja natychmiast uwalnia się od kosztów licencyjnych i buduje własne kompetencje. Zmiany oprogramowania na stacjach roboczych (front-office) powinny być etapem końcowym.
3. W jaki sposób zapewnić maksymalną replikowalność wdrażanych rozwiązań?	Aby pilotaż w Jednostkach Samorządu Terytorialnego nie zakończył się na etapie odizolowanych wdrożeń, minister cyfryzacji musi od samego początku wpisać w regulamin konkursu mechanizmy umożliwiające łatwe kopiowanie i skalowanie wypracowanych rozwiązań. Warto w tym celu sięgnąć do najlepszych europejskich praktyk (m.in. francuskiego <i>CodeGouv</i> czy włoskiego <i>Developers Italia</i>). Opierając się również na strategii "European Open Digital Ecosystem Strategy", rekomendujemy podjęcie poniższych kroków. k. Wymogi technologiczne i architektoniczne Skopiowanie kodu to za mało – inna jednostka musi móc go łatwo uruchomić. • Konteneryzacja i Infrastructure as Code (IaC): Wymogiem wdrożeniowym powinno być dostarczenie oprogramowania w postaci skonteneryzowanej tam, gdzie to możliwe (do uruchomienia na Kubernetes), wraz ze skryptami automatyzującymi budowę infrastruktury (np. Ansible, Terraform, Salt). Dzięki temu inna jednostka administracji może wdrożyć system "pod klucz" na własnych serwerach lub w chmurze rządowej w ułamku czasu pierwotnego wdrożenia.

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	• Podejście API-first i modułowość: Aplikacje muszą komunikować się przez ustandaryzowane interfejsy API. Logika biznesowa powinna być oddzielona od specyficznego dla danej gminy czy ministerstwa interfejsu. Rozwiązania muszą od razu być projektowane z myślą o integracji z krajową infrastrukturą (Węzeł Krajowy, ePUAP, mObywatel). • Agnostyczność środowiskowa: Oprogramowanie nie może być na stałe powiązane z usługami konkretnego dostawcy chmury komercyjnej (tzw. cloud lock-in), chyba że używa otwartych standardów do komunikacji z tymi usługami. Ekosystem i standaryzacja wiedzy, utworzenie centralnego repozytorium kodu dla administracji Większość prób przeniesienia systemu z jednego urzędu do drugiego upada z powodu braku odpowiedniej dokumentacji dla różnych grup odbiorców. • Krajowy Katalog Oprogramowania (KKO): Utworzenie jednej, centralnej i łatwo przeszukiwalnej platformy (np. na dane.gov.pl lub jako osobny portal opensource.gov.pl), gdzie każda jednostka może zgłosić i pobrać gotowe do użycia, zweryfikowane pod kątem bezpieczeństwa rozwiązania. Niezbędne elementy ułatwiające kopiowanie to szczegółowy opis funkcjonalny, wytyczne dotyczące wdrażania, zestawienie komponentów oprogramowania (SBOM) oraz punkty kontaktowe oferujące wsparcie przy ponownym wykorzystaniu. • Trójstopniowa dokumentacja wdrożenia: Repozytorium musi zawierać nie tylko kod, ale obowiązkowo: – Dokumentację instalacyjną dla działów IT (jak postawić system). – Dokumentację użytkową dla urzędników (jak w tym pracować). – Dokumentację prawno-proceduralną (jakie polityki prywatności, klauzule RODO czy regulaminy są potrzebne do wdrożenia). Model wsparcia i utrzymania Jednostki publiczne często boją się Open Source z obawy o to kto naprawi błąd, gdy system przestanie działać. • Centralne umowy ramowe na wsparcie: zamiast kazać każdej gminie osobno szukać dostawcy wsparcia dla otwartego systemu, Ministerstwo może wyłonić w przetargach pulę certyfikowanych wykonawców świadczących wsparcie (SLA), poprawki i rozwój dla konkretnych rozwiązań Open Source z Krajowego Katalogu. • Współfinansowanie rozwoju (crowdfunding publiczny): stworzenie mechanizmów prawnych, w których kilka jednostek samorządowych może zrzucić się na zapłacenie wykonawcy za napisanie nowego modułu do istniejącego systemu Open Source. Raz napisany moduł trafia do głównego repozytorium i służy od razu wszystkim innym jednostkom w kraju. Skupienie pierwszych konkursów na projektach, które udowodnią, że instalacja oprogramowania w nowym urzędzie zajmuje godziny, a nie miesiące, będzie najlepszym dowodem na skuteczność transformacji w stronę suwerennego oprogramowania. Wdrożenie zasady publiczne pieniądze, publiczny kod Kluczowym warunkiem replikowalności jest wymóg prawny, aby całe oprogramowanie, a także skrypty integracyjne czy szablony konfiguracyjne powstałe za publiczne środki z pilotażu, otrzymały otwartą licencję. Ministerstwo musi zagwarantować, że samorządy lub firmy wdrażające nie zatrzymają praw majątkowych do wypracowanych konfiguracji. Pozwoli to kolejnym gminom pobrać gotowy zestaw narzędzi bez ponoszenia ponownych kosztów na rozwój. Wymóg otwartych standardów i interoperacyjności Rozwiązania wdrożone w jednej gminie zadziałają w innej tylko wtedy, gdy będą oparte na wspólnych standardach wymiany danych. Regulamin konkursu musi narzucać wymóg stosowania otwartych interfejsów programowania aplikacji (API).

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	Takie podejście redukuje trudności w integracji i umożliwia kompatybilność w całym ekosystemie. Dzięki temu kolejna jednostka administracji publicznej będzie mogła z łatwością połączyć nowy komunikator czy pakiet biurowy z posiadanymi już, lokalnymi systemami dziedzicznymi. Stworzenie otwartych centrów kompetencji i sieci OSPO Technologia to zaledwie jeden aspekt przy powielaniu wdrożeń; kolejnym są braki kompetencyjne w urzędach. Władze centralne powinny powołać sieć biur do spraw oprogramowania open source (OSPO) na szczeblu krajowym lub regionalnym. Jednostki te pełnią rolę centrów kompetencji, które zapewniają wsparcie techniczne, szkolenia i pomoc dla kolejnych samorządów chcących zaadaptować oprogramowanie open source. Takie centra budują most pomiędzy rozwiązaniami stworzonymi w ramach projektów społecznościowych a rygorystycznymi wymaganiami wdrożeniowymi w administracji publicznej. Wprowadzenie oceny możliwości ponownego wykorzystania Aby wymusić replikowalność w przyszłości, państwo powinno wprowadzić obowiązek przeprowadzania oceny przed każdym nowym przetargiem informatycznym na kwotę powyżej miliona euro. Zanim gmina ogłosi postępowanie na nowy system pocztowy czy chmurowy, musi sprawdzić, czy odpowiednie oprogramowanie typu open source nie jest już dostępne w innej jednostce. Mechanizm ten pozwala unikać marnotrawstwa finansowego polegającego na tym, że różne instytucje wielokrotnie płacą za stworzenie i wdrożenie niemal identycznych platform.
4. Jakie czynniki bezpieczeństwa należy uwzględnić?	Podmioty realizujące pilotaże wdrożeniowe w polskich samorządach muszą uwzględnić szerokie spektrum czynników bezpieczeństwa, aby zbudować odporną i suwerenną infrastrukturę cyfrową. Opierając się na europejskiej strategii cyfrowej, wykonawcy muszą wziąć pod uwagę pięć głównych obszarów ryzyka i zapobiegania zagrożeniom. Przejrzystość łańcucha dostaw i generowanie SBOM Podmioty wdrażające oprogramowanie open source w urzędach muszą zagwarantować pełną przejrzystość w łańcuchu dostaw oprogramowania. Organizacje te mają obowiązek przygotowywać i udostępniać maszynowo czytelne zestawienia komponentów oprogramowania (SBOM). Raportowanie w tym standardzie pozwala samorządom sprawnie identyfikować zależności, audytować otrzymany kod oraz weryfikować jego właściwości pod kątem bezpieczeństwa. Ciągłe zarządzanie podatnościami Realizatorzy pilotażu muszą wdrożyć rygorystyczne zasady nadzoru i zintegrować procedury zarządzania oprogramowaniem open source z całym cyklem życia projektu. Wymagamy, aby podmioty wdrożyły zautomatyzowane śledzenie zależności i traktowały szybkie aktualizacje oprogramowania jako priorytet. Otwarty kod pozwala na bardzo szybkie reagowanie na ewentualne luki w zabezpieczeniach, ponieważ cała społeczność programistów potrafi dostarczyć poprawki błyskawicznie, bez czekania na powolne cykle wydawnicze konkretnych dostawców. Zapewnienie audytów i stabilnego utrzymania kodu Poważną barierą we wdrażaniu bezpiecznego open source jest zjawisko systemowego braku finansowania samej konserwacji kodu, tworzenia łat bezpieczeństwa i naprawiania błędów. Zwycięzcy konkursu, otrzymując środki na wdrożenie i roczne wsparcie techniczne, muszą zaplanować ciągłe testy bezpieczeństwa oraz przeznaczyć odpowiednie zasoby kadrowe na szybkie usuwanie podatności. Doskonałym wzorcem działania są tu unijne inicjatywy typu EU-FOSSA, które opierają się na regularnym audytowaniu kluczowych komponentów oprogramowania. Architektura bezpieczeństwa od fazy projektowania Firmy wdrażające pakiety biurowe, komunikatory czy chmury prywatne muszą opierać swoje usługi na najnowocześniejszych modelach ochrony. Architektura ta

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	musi uwzględniać zasady ograniczonego zaufania (zero trust) oraz bezpieczeństwa gwarantowanego już na wczesnym etapie projektowania (security by design). Otwarty charakter oprogramowania daje twórcom możliwość łatwej adaptacji tych mechanizmów w celu radzenia sobie z przyszłymi zagrożeniami cyfrowymi w urzędach. Zgodność z certyfikacjami i regulacjami Wdrażane rozwiązania Open Source muszą ułatwiać instytucjom spełnienie rygorów dyrektywy NIS2, dyrektywy CER oraz wymogów Krajowego Systemu Cyberbezpieczeństwa. Pilotáže powinny preferować rozwiązania i dystrybucje posiadające międzynarodowe certyfikaty bezpieczeństwa (np. Common Criteria, FIPS 140-3), co drastycznie obniża ryzyko i koszty późniejszych audytów. Gwarancja długoterminowego wsparcia (LTS) i rygorystyczne SLA Bezpieczeństwo w czasie to zdolność do natychmiastowego reagowania na nowe zagrożenia (podatności zero-day). Podmioty realizujące pilotáže muszą od początku uwzględniać model utrzymania – np. poprzez komercyjne subskrypcje wsparcia typu enterprise dla rozwiązań Open Source, które zapewniają rygorystyczne czasy łatania luk (patch management) bez wymogu angażowania dodatkowych zasobów własnych urzędu. Bezpieczeństwo łańcucha dostaw oprogramowania (Software Supply Chain Security) Warto mieć kontrolę nad pochodzeniem i integralnością używanego kodu. Należy weryfikować proces budowania oprogramowania i stosować mechanizmy kryptograficzne do podpisywania komponentów. Warto rozważyć opieranie się na zweryfikowanych dystrybucjach od stabilnych, sprawdzonych dostawców, którzy gwarantują certyfikatami (np. Common Criteria EAL4+) brak ingerencji osób trzecich w łańcuch dostaw. Transparentność i ciągły audyt kodu (DevSecOps) Pilotáže powinny w pełni wykorzystywać największą zaletę Open Source, jaką jest przejrzystość. Wymagane jest wdrożenie procesów ciągłego, zautomatyzowanego skanowania kodu i zależności pod kątem podatności oraz przeprowadzanie niezależnych audytów bezpieczeństwa przed przeniesieniem pilotażu na skalę produkcyjną
5. Jak ułatwić pracownikom przejście na open source?	Przejście z rozwiązań własnościowych na oprogramowanie open source w jednostkach samorządu terytorialnego wymaga odpowiedniego podejścia do pracowników, którzy przez lata przyzwyczaili się do komercyjnych interfejsów. Samorządy muszą wdrożyć przemyślaną strategię adaptacji, która zminimalizuje opór i zagwarantuje ciągłość pracy urzędu. Edukacja oparta na umiejętnościach uniwersalnych Kluczowym elementem ułatwiającym pracownikom migrację jest zmiana filozofii szkoleń. Władze centralne oraz podmioty wdrażające pilotaż powinny zrezygnować z uczenia urzędników obsługi jednego konkretnego programu komercyjnego. Należy zamiast tego kłaść nacisk na rozwój uniwersalnych umiejętności cyfrowych i logiki działania systemów otwartych. Takie podejście bezpośrednio zapobiega utrwalaniu uzależnienia od jednego dostawcy oprogramowania i buduje rzeczywiste kompetencje technologiczne kadry urzędniczej. Zapewnienie oprogramowania klasy korporacyjnej Pracownicy muszą czuć, że nowe narzędzia są równie stabilne i bezpieczne co rozwiązania zamknięte. Jeśli pracownicy będą mieli poczucie, że dostają rozwiązanie gorsze, "powiązane sznurkiem", będą na każdym kroku utrudniać i blokować wdrożenia. Dlatego kluczowe jest oparcie pilotażu na rozwiązaniach oferujących profesjonalne wsparcie techniczne i rygorystyczne SLA. Gwarantuje to szybkie rozwiązywanie problemów, eliminuje frustrację użytkowników w pierwszych, krytycznych dniach po migracji i buduje zaufanie do stabilności technologii Open Source w administracji. Stopniowa migracja i płynne wdrażanie

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

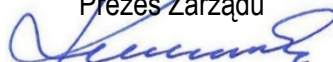
Pytanie	Odpowiedź
	Gwałtowne odcięcie pracowników od starych systemów zazwyczaj paraliżuje pracę urzędu. Zmiany technologiczne organizatorzy muszą wprowadzać etapami. Doskonałym wzorcem jest tutaj duńskie ministerstwo do spraw cyfryzacji, które latem 2025 r. rozpoczęło zastępowanie komercyjnych pakietów rozwiązaniem LibreOffice. Duńczycy początkowo przenieśli na nowy system zaledwie połowę pracowników, planując pełną migrację dopiero na jesień tego samego roku. Tak rozłożony w czasie harmonogram daje zespołom przestrzeń na naukę i dostosowanie codziennego obiegu dokumentów do nowych realiów. Budowa regionalnych centrów kompetencji Samorządy i ich pracownicy potrzebują miejsca, do którego mogą zwrócić się z codziennymi problemami. Administracja publiczna powinna utworzyć regionalne centra kompetencji, które zajmą się bezpośrednim wsparciem technicznym oraz szkoleniami dla urzędników korzystających z oprogramowania open source. Ośrodki te będą stanowić merytoryczne zaplecze, pomagając pracownikom w płynnym przyswojeniu nowych narzędzi pocztowych, chmurowych czy wideokonferencyjnych. Gwarancja profesjonalnego wsparcia technicznego Urzędnicy poczują się znacznie bezpieczniej, mając świadomość, że w przypadku awarii mogą liczyć na błyskawiczną interwencję specjalistów. Ministerstwo Cyfryzacji zaplanowało już roczne finansowanie takiego wsparcia, ale samorządy powinny w tym procesie postawić na budowanie relacji z lokalnym rynkiem usług IT. Przykład z Barcelony udowadnia, że zaangażowanie lokalnych przedsiębiorstw oraz zatrudnienie zewnętrznych deweloperów do opieki nad systemem znakomicie ułatwia adaptację urzędu do nowych warunków i gwarantuje płynność pracy administracyjnej. Ewolucyjne wdrażanie i interoperacyjność Migracja nie powinna być rewolucją zaburzającą ciągłość pracy. Sukces publicznych wdrożeń zależy od bezproblemowej integracji z istniejącymi systemami. W pierwszej kolejności warto migrować infrastrukturę serwerową i backendową, co jest przezroczyste dla użytkownika końcowego. W przypadku narzędzi biurowych kluczowe jest zagwarantowanie bezwzględnej kompatybilności formatów wymiany danych z systemami zewnętrznymi.
6. Jakie wsparcie zapewnić podmiotom realizującym pilotaże?	Wsparcie dla podmiotów realizujących pilotaże w samorządach musi obejmować znacznie więcej niż tylko roczne finansowanie. Aby wdrożenia chmury prywatnej, edytorów tekstu czy komunikatorów zakończyły się sukcesem, państwo musi otoczyć gminy stałą opieką techniczną, prawną i analityczną. Wsparcie z regionalnych centrów kompetencji Władze centralne mogą powołać regionalne centra kompetencji zapewniające bezpośrednią pomoc techniczną dla administracji publicznej wdrażającej rozwiązania otwarte. Ośrodki te zaoferują samorządom usługi w zakresie integracji systemów, dostosowywania oprogramowania do lokalnych potrzeb urzędów oraz szkolenia dotyczące prawidłowego zarządzania tymi platformami. Transfer wiedzy i budowa suwerenności kompetencyjnej Wsparcie eksperckie nie może ograniczać się do outsourcingu usług. Musi ono obejmować intensywne szkolenia, warsztaty i mentoring (tzw. shadowing) dla wewnętrznych zespołów IT w administracji publicznej. Długofalowe uniezależnienie od pozaeuropejskich korporacji wymaga zbudowania lokalnych kompetencji, które po fazie pilotażu pozwolą na samodzielne skalowanie i utrzymywanie infrastruktury. Doradztwo architektoniczne i migracyjne Niezbędne jest wsparcie ekspertów przy planowaniu całego procesu przejścia z systemów zamkniętych na otwarte standardy (np. zarządzanie infrastrukturą opartą na systemach Linux czy klastrach Kubernetes). Wsparcie to powinno koncentrować się na tworzeniu rozwiązań wykluczających uzależnienie od jednego dostawcy (vendor lock-in), co jest kluczowe dla prawdziwej suwerenności cyfrowej.

Krajowa Izba Gospodarcza Elektroniki i Telekomunikacji

Pytanie	Odpowiedź
	Wsparcie w obszarze cyberbezpieczeństwa i zgodności Zespoły wdrażające powinny otrzymać wsparcie w zakresie dostosowania wdrażanych rozwiązań do wymogów prawnych (np. dyrektywa NIS2). Konieczne jest zapewnienie bezpiecznego łańcucha dostaw oprogramowania (z europejskimi korzeniami), dostępu do audytowanych repozytoriów oraz gwarancji ciągłego dostarczania krytycznych poprawek bezpieczeństwa. Pomoc prawna i doradztwo zakupowe Ministerstwo dostarczy urzędnikom dedykowaną pomoc prawną, która pozwoli samorządom właściwie zarządzać kwestiami własności intelektualnej przy publikacji kodu. Wsparcie to ułatwi również prowadzenie trudnych negocjacji kontraktowych z dotychczasowymi dostawcami oprogramowania komercyjnego, którzy często blokują przejście na technologie open source. Dodatkowo państwo sformuje zespoły analityczne dostarczające jednostkom samorządowym strategiczny wywiad zakupowy oraz merytoryczne wsparcie w negocjacjach. Centralne umowy ramowe na usługi Aby zredukować urzędowe bariery proceduralne, administracja centralna przygotuje wstępnie rozstrzygnięte umowy ramowe na wsparcie techniczne, personalizację, integrację oraz utrzymanie głównych platform open source, w tym takich systemów jak LibreOffice, Nextcloud czy Kubernetes. Dzięki temu mechanizmowi samorządy zlecać niezbędne prace serwisowe wykwalifikowanym dostawcom bez konieczności każdorazowego organizowania skomplikowanych przetargów informatycznych. Bezpośrednia ścieżka eskalacji "upstream" Zaleca się aby podmioty realizujące pilotaż miały dostęp do inżynierów, którzy aktywnie współtworzą kod źródłowy wykorzystywanych rozwiązań open source na poziomie globalnym. Umożliwi to szybkie identyfikowanie i łatanie błędów u samego źródła oraz realny wpływ polskiej administracji na rozwój używanych narzędzi.

Z wyrazami szacunku

Prezes Zarządu



Stefan Kamiński