Procedura udostępniania aplikacji dla użytkowników Politechniki Wrocławskiej

1. Cel procedury

Celem niniejszej procedury jest określenie działań i odpowiedzialności zapewnianiających, że wszystkie aplikacje udostępniane wewnętrznym użytkownikom Politechniki Wrocławskiej są bezpieczne, wydajne, zgodne z wytycznymi wewnętrznymi i spełniają określone wymagania funkcjonalne. Procedura ma na celu minimalizację ryzyka związanego z lukami w zabezpieczeniach oraz zapewnienie płynnego działania aplikacji.

2. Procedura

Poniżej przedstawiono poszczególne kroki w procesie akceptacji i udostępniania aplikacji.

2.1 Złożenie wniosku o zgodę na udostępnienie aplikacji

Każde udostępnienie nowej aplikacji musi rozpocząć się od złożenia formalnego wniosku przez właściciela biznesowego aplikacji do Działu Informatyzacji (DZI). Wniosek powinien zawierać co najmniej:

- Nazwę
- Opis aplikacji oraz jej przeznaczenie biznesowe
- Listę przewidywanych użytkowników/grup odbiorców
- Opis kluczowych funkcjonalności
- Informacje o architekturze aplikacji i wykorzystanych technologiach
- Określenie wymaganych integracji z innymi usługami np. USOS API, SSO
- Wymagania poza funkcjonalne, np. przewidywane obciążenie

Wniosek może zostać złożony osobiście lub przesłany droga elektroniczną.

Następstwem wniosku może być stworzenie kont do integracji w środowiskach developerskich (jeśli nie zostały stworzone i udostępnione wcześniej w ramach tworzenia aplikacji)

2.2 Wstępna analiza i planowanie testów

Po otrzymaniu wniosku, zespół DZI przeprowadza wstępną analizę i planuje zakres testów i dokumentów niezbędnych do uzyskania zgody. Na tym etapie:

- Analizowane jest ryzyko związane z aplikacją, biorąc pod uwagę jej wpływ na działalność Uczelni i rodzaj przetwarzanych danych
- Analizowane jest pokrycie funkcjonalności aplikacji z innymi działającymi i planowanymi systemami uczelni
- Określana jest lista dokumentów i testów wraz z harmonogramem, uwzględniający dostępne zasoby, niezbędnych do akceptacji aplikacji
- Określana jest potrzeba skonsultowania i uzyskania akceptacji szaty graficznej z działem uczelni odpowiedzialnym za prezentację uczelni na jej zewnątrz
- Określana jest potrzeba dokonania ustaleń odnośnie wymagań i rejestracji bazy danych osobowych z IOD
- Jeśli jest potrzeba definiowane są konkretne scenariusze testowe zarówno dla testów funkcjonalnych, bezpieczeństwa i/lub wydajności.

2.3 Analiza ryzyka

Ocenianie jest zagrożenie, jakie aplikacja może stanowić dla organizacji. Analizę przeprowadza zgłaszający aplikację, w oprogramowaniu do analizy ryzyka udostępnionym przez DZI.

Analiza ryzyka powinna obejmować:

1. Identyfikacja zasobów - określenie, co aplikacja przetwarza i do czego ma dostęp (np. bazy danych studentów, baza przedmiotów itp).
2. Identyfikacja zagrożeń - wskazanie potencjalnych negatywnych zdarzeń (np. nieautoryzowany dostęp, wyciek danych, awaria systemu, atak DoS).
3. Identyfikacja podatności - wskazanie słabości, które mogą zostać wykorzystane przez zagrożenia (np. brak szyfrowania, luki w bibliotekach zewnętrznych, słabe uwierzytelnianie).
4. Ocena prawdopodobieństwa i wpływu - oszacowanie, jak prawdopodobne jest wystąpienie każdego z zagrożeń i jakie byłyby jego konsekwencje finansowe, wizerunkowe i dla danych osobowych
5. Określenie poziomu ryzyka – klasyfikacja ryzyka poprzez przypisywanie poziomu
6. Określenie strategii postępowania z ryzykiem:
    - Mitygacja - zmniejszenie ryzyka przez wdrożenie odpowiednich rozwiązań
    - Akceptacja - świadome przyjęcie ryzyka
    - Unikanie - rezygnacja z funkcjonalności lub wdrożenia (jeśli ryzyko jest nieakceptowalne)

Wynikiem analizy ryzyka jest wygenerowany raport, który należy przekazać do zespołu DZI. Analiza jest akceptowana lub kierowana do poprawy.

2.4 Testy bezpieczeństwa (jeśli wymagane)

Celem jest identyfikacja i eliminacja potencjalnych luk w zabezpieczeniach aplikacji. Zalecane jest przeprowadzenie następujących testów:

- skanowanie podatności - automatyczne skanowanie kodu aplikacji i jej zależności w poszukiwaniu znanych podatności
- testy penetracyjne - symulacja ataków na aplikację w celu znalezienia i wykorzystania luk w zabezpieczeniach
- przegląd konfiguracji - weryfikacja czy serwery i usługi, na których działa aplikacja, są poprawnie skonfigurowane pod kątem bezpieczeństwa

Poziom i inne wymagania odnośnie do testów określane są indywidulanie na podstawie analizy wniosku i raportu z analizy ryzyk.

2.5 Testy wydajności (jeśli wymagane)

Mają na celu sprawdzenie, jak aplikacja zachowuje się pod obciążeniem i czy spełnia oczekiwania dotyczące szybkości działania. W ramach tych testów może zajść potrzeba przeprowadzenia:

- testów obciążeniowych - określenie jak aplikacja radzi sobie z oczekiwanym, normalnym obciążeniem
- testów przeciążeniowych - przeciążenie aplikacji w celu zidentyfikowania jej granic wytrzymałości i sprawdzenia, jak zachowuje się w ekstremalnych warunkach
- testów stabilności - długotrwałe testy pod normalnym obciążeniem, mające na celu wykrycie problemów z wyciekami pamięci lub innymi zasobami

Poziom i inne wymagania odnośnie do testów określane są indywidulanie na podstawie analizy wniosku i raportu z analizy ryzyk.

3.Podsumowanie

Na podstawie raportów Dział Informatyzacji podejmuje decyzję o pozwoleniu na udostępnieniu aplikacji lub ewentualnym wstrzymaniu wdrożenia do czasu wprowadzenia niezbędnych poprawek. Jeśli aplikacja wymaga dostępów do innych usług (np. USOS API) następuje utworzenie odpowiednich kont produkcyjnych i przekazanie ich wnioskodawcy. Dodatkowo DZI może podjąć decyzję o podłączeniu/integracji z wykorzystywanymi na uczelni systemami monitoringu lub innymi wspomagający zarządzanie systemami informatycznymi np. Zabix lub systemem kolejkowym OTOBO, a

także konieczność przygotowania materiałów pomocy dla użytkowników. Przygotowanie systemu do integracji spoczywa na Wnioskującym

Po wdrożeniu aplikacja powinna być na bieżąco monitorowana pod kątem bezpieczeństwa i wydajności przez jej właściciela. Wszelkie incydenty lub problemy z działaniem zgłaszane przez jednostki uczelni lub użytkowników końcowych powinny być natychmiast analizowane. Zaleca się również cykliczne powtarzanie testów bezpieczeństwa, zwłaszcza po wprowadzeniu istotnych zmian w aplikacji lub jej otoczeniu.