

# Specyfikacja нефunkcjonalna

Niniejsza specyfikacja нефunkcjonalna powinna zawierać poniższe informacje:

1. technologie, jakie będą wykorzystane do tworzenia przedmiotu zamówienia,
2. elementy techniczne, jakie przedmiot zamówienia będzie zawierał,
3. parametry techniczne, jakie będzie spełniał przedmiot zamówienia,
4. wymagania w zakresie konfiguracji stosu technologicznego i środowiska realizacji (developerkiego, preprodukcyjnych oraz produkcyjnych) przedmiotu zamówienia,
5. innych, istotnych zagadnień mających wpływ na funkcjonowanie systemu.

<b>Technologie</b>	<b>3</b>
Stos technologiczny:	3
Biblioteki używane po stronie backend	3
Biblioteki używane po stronie frontend	3
Otoczenie projektowanego systemu	3
Integracje	3
<b>Elementy techniczne</b>	<b>4</b>
Widok logiczny systemu	4
Szczegóły dotyczące infrastruktury	4
Wykorzystywana warstwa sprzętowa	4
Warstwa systemu operacyjnego	4
Warstwa aplikacji	4
Warstwa bazy danych	4
Ograniczenia w zakresie architektury	4
Wykaz wymaganych licencji	4
<b>Parametry techniczne</b>	<b>5</b>
Lista obsługiwanych przeglądarek	5
Założenia RWD	5
Bezpieczeństwo	5
Skalowalność	5
Dostępność	5

Wydajność	5
Założenia co do jakości	6
<b>Konfiguracja Środowiska</b>	<b>7</b>
Lista Środowisk z podziałem na lokalizację	7
Odpowiedzialność za utrzymanie	7
Strategia wersjonowania	7
Repozytorium	7
Parametry backupu środowisk	7
<b>Pozostałe</b>	<b>8</b>
Dokumentacja	8
Szkolenia	8

# Technologie

Rozdział ma na celu zebranie informacji o technologiach wykorzystywanych w projekcie oraz zależności między systemami.

## Stos technologiczny:

Uzupełnia: Lider Techniczny

*Zawrzeć informacje odnośnie stosu technologicznego projektu w każdej z tworzonych warstw.*

## Biblioteki używane po stronie backend

Uzupełnia: Lider techniczny

*Wypisać główne technologie oraz znane biblioteki planowane do użycia.*

*W szczególności wymienić wszystkie biblioteki i narzędzia płatne.*

## Biblioteki używane po stronie frontend

Uzupełnia: Lider techniczny / Frontend developer

*Wypisać główne technologie oraz znane biblioteki planowane do użycia.*

*W szczególności wymienić wszystkie biblioteki i narzędzia płatne,*

## Otoczenie projektowanego systemu

Uzupełnia: Architekt / Lider Techniczny

*Określić sposób komunikacji z systemami zewnętrznymi - można narysować diagram ekosystemu wraz z określeniem formy wymiany danych.*

## Integracje

Uzupełnia: Architekt / Lider Techniczny

*Wyszczególnione na diagramie integracje wypisać wraz z podaniem źródła i formatu danych.*

# Elementy techniczne

## Widok logiczny systemu

Uzupełnia: Architekt / Lider Techniczny

Narysować diagram logiczny systemu:

- podział na moduły. Każdy moduł powinien posiadać skrótowo opisaną odpowiedzialność.
- systemy z którymi się integruje i kierunek integracji

Jeżeli diagram obszerny (zajmuje A4 lub A3) może być jako odrębna strona lub załącznik do tej specyfikacji. Musi być czytelny i posiadać legendę.

## Szczegóły dotyczące infrastruktury

### Wykorzystywana warstwa sprzętowa

Uzupełnia: Architekt / Lider Techniczny / Hosting

Sprecyzować wymagania co do infrastruktury sprzętowej - minimalną ilość węzłów, ich typów oraz parametrów.

### Warstwa systemu operacyjnego

wyspecyfikować wymagania co do systemu operacyjnego.

### Warstwa konteneryzacji (opcjonalnie)

wyspecyfikować wymagania co do platformy konteneryzacji (np. Kubernetes)

### Warstwa aplikacji

wyspecyfikować wymagania co do serwera aplikacyjnego i innych aplikacji wymaganych do działania przedmiotu zamówienia.

### Warstwa bazy danych

wyspecyfikować wymagania co do bazy danych (silnik - dostawca, wersja, specyficzna konfiguracja).

### Ograniczenia w zakresie architektury

Wypisać znane ograniczenia architektury (np pliki binarne przechowywane w bazie danych, brak możliwości działania bez danego elementu aplikacji etc.)

## Wykaz wymaganych licencji

Uzupełnia: Architekt / Lider Techniczny / Hosting / Sprzedaż

*Wyszczególnić wszystkie znane licencje - własne oraz firm trzecich.*

# Parametry techniczne

## Lista obsługiwanych przeglądarek

Uzupełnia: Architekt / Lider Techniczny / Hosting

*Wyszczególnić listę przeglądarek, które poprawnie współpracują z aplikacją. Jeżeli listy różnią się pomiędzy back office a front office to należy przygotować dwie listy. Listy powinny zawierać nazwę oraz ostatnią obsługiwaną wersję - np. Firefox od wersji 45.0.2. Lista powinna zawierać przeglądarki desktop jak i mobilne.*

## Założenia RWD

Uzupełnia: Koncepcja

*Podać tzw breakpoints - szerokości ekranów dla których, dajemy gwarancję, że aplikacja będzie wyglądać poprawnie.*

## Bezpieczeństwo

Uzupełnia: Architekt / Lider Techniczny

*Opisać zastosowane mechanizmy zwiększające bezpieczeństwo aplikacji, spełniany OWASP ASVS etc.*

## Skalowalność

Uzupełnia: Architekt / Lider Techniczny / Hosting

*Opisać zastosowane mechanizmy umożliwiające skalowanie aplikacji (np serwery aplikacyjne w klastrze, OpenShift etc.)*

## Dostępność

Uzupełnia: Architekt / Lider Techniczny / Sprzedaż

*Opisać minimalną zakładaną dostępność systemu (SLA) oraz wskazać obiektywną metrykę pomiaru (SLI), która będzie weryfikowalna co najmniej post-mortem*

## **Wydajność**

Uzupełnia: Architekt / Lider Techniczny / Sprzedaż

*Opisać znane założenia wydajności - np. ruch na stronie, ilość requestów API na sekundę etc, wraz z oczekiwanymi czasami odpowiedzi. Warto dodać też definicję pomiaru - czy mierzony jest czas do pierwszej czy ostatniej odpowiedzi.*

## **Założenia co do jakości**

Uzupełnia: Architekt / Lider Techniczny / Sprzedaż

*Określić poziom jakości projektu - np. dopuszczalna ilość błędów z sonara, stopień pokrycia kodu testami. Opis metryki jakościowej powinien umożliwiać jej weryfikację przez Klienta (tam gdzie to możliwe opisać możliwe sposoby sprawdzenia przez Klienta)*

# Konfiguracja środowiska

## Lista środowisk z podziałem na lokalizację

Uzupełnia: Architekt / Lider Techniczny / Hosting

*Jeżeli aplikacja hostowana jest przez E-Point wylistować środowiska (imp, tst, stg, prod, etc) oraz wskazać fizyczną lokalizację serwerów. Jeżeli umowa wymaga DRP (disaster recovery plan) to należy wskazać również zapasowe lokalizacje.*

## Odpowiedzialność za utrzymanie

Uzupełnia: Sprzedaż

*Wskazać, która strona (Zleceniodawca lub zleceniobiorca) odpowiedzialny jest za utrzymanie przedmiotu zamówienia. Należy wskazać zarówno odpowiedzialność za infrastrukturę jak i aplikację.*

## Strategia wersjonowania

Uzupełnia: Lider Techniczny

*Opisać strategię wersjonowania.*

## Repozytorium

Uzupełnia: Architekt / Lider Techniczny

*Jeżeli umowa wymaga przekazania kodu lub współdzielenia repozytorium to opisać zasady przekazania lub współdzielenia (np - przekazujemy kod źródłowy bez historii git / z historią git).*

## Parametry backupu środowisk

Uzupełnia: Architekt / Lider Techniczny / Hosting

*Opisać jak często tworzone są backupy oraz jak długo są przechowywane. Należy również opisać miejsce przechowywania backupów.*

# Pozostałe

## **Dokumentacja**

Uzupełnia: Architekt / Lider Techniczny / Hosting

*Opisać jaka dokumentacja zostanie dostarczona w ramach projektu.*

## **Szkolenia**

Uzupełnia: Architekt / Lider Techniczny / Hosting

*Opisać ilość dni szkoleniowych oraz zakres szkolenia (np szkolenie tylko dla redaktorów, administratorów, lub szkolenia dla programistów) wynikających z zakresu.*