

1 Przeglądarka wiadomości (full-stack developer)

1.1 Treść zadania

Przygotuj prostą aplikację wyświetlającą wiadomości regionalne.

Celem zadania jest przygotowanie serwera usług REST pośredniczącego w wywołaniach do zdalnego API oraz prostej aplikacji Web prezentującej wyniki.

W realizacji wykorzystaj dostępne powszechnie REST API <https://newsapi.org/>¹

Przygotuj usługę w postaci:

endpoint	GET /news/{country}/{category}/
odpowiedź	<pre>{ country: string?, category: string?, articles: { author: string, title: string, description: string, date: string(yyyy-MM-dd), sourceName: string, articleUrl: string, imageUrl: string }[] }</pre>

W ramach realizacji zadania przyjmij:

- filtrowanie po kraju "pl",
- filtrowanie po kategorii "technology",

możesz zaszyć te parametry jako stałe po stronie UI. Aplikacja frontendowa powinna być utworzona zgodnie z koncepcją Responsive Web Design - przyjmij breakpoints dla szerokości 768px i 1200px.

Dodatkowo, oczekujemy, że zbudowanie i deployment aplikacji sprowadzają się do jednego polecenia. Do budowania aplikacji użyj narzędzia Maven.

Kryteriami akceptacji zadania są:

- przygotowanie działającego projektu, zawierającego README jak projekt budować i uruchamiać,
- umieszczenie kodów na GitHub razem z historią zmian,
 - repozytorium powinno być prywatne - po wykonaniu zadania prześlemy informację komu należy przekazać uprawnienia w celu zrobienia *Code Review*,

¹ <https://newsapi.org/docs/endpoints/top-headlines>

- przygotowanie części serwerowej usług REST (backend) z wykorzystaniem następujących technologii: Java 8+, Spring 5+,
- przygotowanie części klienckiej prezentującej wyniki (frontend) z wykorzystaniem następujących technologii: Angular 7+, TypeScript, SASS/SCSS,
- przygotowanie konfiguracji uruchamialnego kontenera Docker,
- wywołanie zdalnej usługi wystawionej w publicznym API.

W trakcie oceny zwrócimy uwagę na:

- zgodność projektu z założeniami zadania,
- zaproponowaną architekturę rozwiązania, w tym umiejętności wykorzystania zastosowanego stosu technologicznego,
- jakość przygotowanego kodu,
 - stosowanie dobrych praktyk,
 - stosowanie wzorców projektowych,
- obecność i jakość testów,
- przygotowanie dokumentacji żyjącej z kodem, przykładowo:
 - wygenerowana/wykonywalna dokumentacja REST API,
 - komentarze w kodzie (o ile zasadne!)
- proces budowania i deploymentu aplikacji, w tym stosowanie dobrych praktyk związanych z zastosowanymi narzędziami

Rzeczy które nie będą kluczowe w realizacji projektu:

- UI/UX przygotowywanego rozwiązania, szukamy programistów, a nie grafików 😊
 - funkcjonalne wykorzystanie gotowego template jest jak najbardziej ok, jednakże chcielibyśmy ocenić twoje umiejętności w zakresie technologii frontendowych, więc oczekujemy również fragmentów kodu twojego autorstwa

Potencjalne rozszerzenia zakresu, w razie gdybyś chciał wykazać się bardziej:

- przygotuj po stronie UI listę kategorii z możliwością przejścia do dowolnej z nich,
- możliwość stronicowania wyników,
- możliwość przeglądania źródeł artykułów,
- wyszukiwarka wiadomości.

1.2 Szacunkowy czas realizacji

Zadanie	Czas
Zakres podstawowy	8h
Przygotowanie repozytorium, budującego się projektu, readme	1h
Przygotowanie usługi REST	2h
Przygotowanie części Frontend	3h
Przygotowanie uruchamialnego kontenera	1h

Testy, refaktoring, dokumentowanie kodu	1h
Zakres rozszerzony (tylko dla chętnych, każdy punkt jest niezależny)	Xh
Każda przykładowa kropka jest warta tyle samo	~2h
Zakres rozszerzeń zaproponowany przez kandydata nie jest do zwymiarowania	?