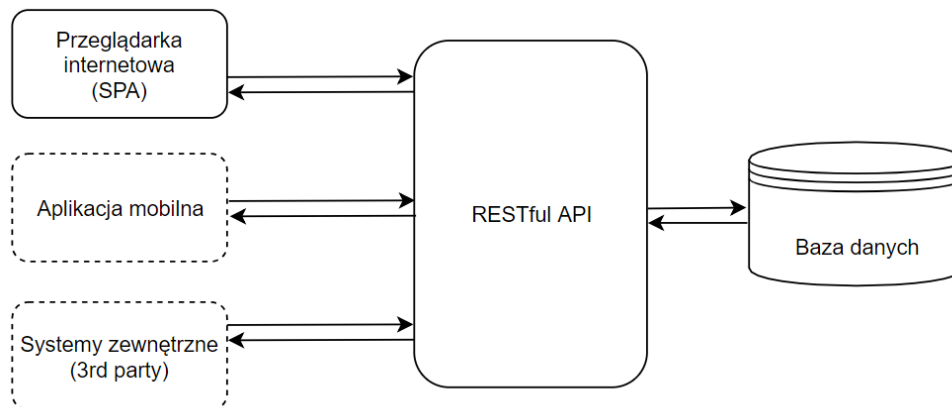


**Spis treści**

<b>1</b>	<b>Architektura systemu</b>	<b>2</b>
1.1	RESTful API . . . . .	2
<b>2</b>	<b>Stos technologiczny</b>	<b>2</b>
2.1	Spring Boot . . . . .	2
2.2	Angular . . . . .	3
2.3	MySQL . . . . .	3

# 1 Architektura systemu

Ze względu na potrzebę szerokiej dostępności platformy zostanie ona zrealizowana jako system webowy w architekturze klient - serwer. Interfejsem użytkownika końcowego będzie aplikacja kliencka typu SPA - Single Page Application uruchamiana w przeglądarce internetowej. Aplikacja ta będzie komunikować się z API wystawionym przez aplikację backendową umieszczoną na serwerze w sieci. Aplikacja backendowa z kolei będzie komunikować się z bazą danych w celu odczytu oraz zapisu informacji.



Rysunek 1: Diagram ogólnej architektury systemu

Elementy otoczone linią kreskowaną na diagramie nie będą przedmiotem tej pracy, jednak podkreślają uniwersalność API oraz wskazują możliwości rozwojowe oraz integracyjne systemu.

## 1.1 RESTful API

API wystawiane przez część serwerową będzie zaprojektowane w oparciu styl architektoniczny REST, który zakłada komunikację klient-serwer z uwzględnieniem następujących zasad:

- użycie podstawowych metody protokołu HTTP czyli - GET, PUT, POST oraz DELETE,
- identyfikacja zasobów poprzez URL,
- komunikacja bezstanowa (brak sesji).

Użycie podstawowych metod protokołu HTTP pozytywnie wpływa na czytelność oraz intuicyjność API. Projektowanie z uwzględnieniem powyższych zasad pozwala również zminimalizować powiązania pomiędzy serwerem oraz klientem, API staje się uniwersalne. Otwiera to możliwości rozwoju systemu na inne platformy, np. utworzenie aplikacji klienckich dla systemów mobilnych Android oraz iOS (Rysunek 1).

# 2 Stos technologiczny

## 2.1 Spring Boot

Aplikacja backendowa zostanie utworzona w języku Java (w wersji 1.8) z użyciem frameworka Spring Boot (w wersji 2.0.3). Technologie te zostały wybrane ze względu na następujące czynniki:

- rozwiązania open source,
- duże grono użytkowników oraz baza materiałów w sieci,
- dobra dokumentacja,
- duża ilość dostępnych modułów Springa np. do komunikacji z bazami danych,
- chęć poszerzenia wiedzy na temat tych technologii.

## **2.2 Angular**

Główną technologią wykorzystywaną po stronie front endu będzie framework do tworzenia SPA rozwijany przez Google - Angular (w wersji 6.1.0). Framework ten ułatwia budowę skalowalnych i szybkich aplikacji z bogatym interfejsem użytkownika.

W celu usprawnienia procesu rozwoju aplikacji zostanie wykorzystana biblioteka ngrx (w wersji 6.1.0), wspomagająca zarządzanie stanem aplikacji. Wykorzystanie tej biblioteki znacznie ułatwia analizę działania aplikacji oraz diagnozowanie błędów.

## **2.3 MySQL**

Relacyjna baza danych została wybrana ze względu na przewidywaną dużą ilość powiązań między encjami w systemie. MySQL od firmy Oracle jest darmowym, bezpiecznym oraz wydajnym systemem zarządzania bazą danych. Istotnym uzasadnieniem wyboru tej technologii jest również bardzo dobra integracja z frameworkiem Spring.