

## Wymagania funkcjonalne

1. Gość powinien mieć możliwość czytania informacji dodanych przez bibliotekarza.
2. Gość powinien mieć możliwość rejestracji w bibliotece.
3. Gość powinien mieć możliwość logowania/wylogowania w bibliotece.
4. Gość powinien mieć możliwość przeglądania listy książek dostępnych w bibliotece.
5. Gość powinien mieć możliwość wyszukiwania książek po następujących elementach: tytuł, autor, wydawnictwo, rok wydania.
6. Użytkownik powinien mieć możliwość przeglądania listy książek dostępnych w bibliotece.
7. Użytkownik powinien mieć możliwość wyszukiwania książek po następujących elementach: tytuł, autor, wydawnictwo, rok wydania.
8. Użytkownik powinien mieć możliwość wypożyczenia książki z biblioteki.
9. Użytkownik powinien mieć możliwość przeglądania listy wypożyczonych książek.
10. Bibliotekarz powinien mieć możliwość dodawania nowej książki do kolekcji.
11. Bibliotekarz powinien mieć możliwość usunięcia książki z biblioteki.
12. Bibliotekarz powinien mieć możliwość modyfikacji książki w bibliotece.
13. Bibliotekarz powinien mieć możliwość zablokowania użytkownika.

## Wymagania niefunkcjonalne

1. System musi działać stabilnie.
2. System musi być odporny na popularne błędy użytkowników.
3. System powinien wypożyczyć jeden egzemplarz książki dokładnie jednemu użytkownikowi.
4. Użytkownik może wypożyczyć maksymalnie N książek.
5. System musi być elastyczny, tj. umożliwiać rozszerzanie o dodatkowe moduły w przyszłości.
6. System musi być testowalny w celu wykrycia ewentualnych błędów programistycznych.
7. System musi być dostępny z poziomu różnych przeglądarek użytkownika.
8. System musi być intuicyjny („user-friendly”), posiadać prosty i wygodny interfejs użytkownika.

9. System musi być zabezpieczony przed popularnymi atakami typu CSRF, XSS, Session Hijacking czy SQL Injection.
10. System musi być zoptymalizowany.
11. System musi być odporny na ewentualne awarie.
12. System musi być zabezpieczony przed nieautoryzowanym dostępem.

## Wymagania dotyczące ograniczeń systemu:

1. Front-end aplikacji będzie wykonany przy pomocy technologii HTML5, CSS3, JS.
2. Back-end systemu zostanie wykonany w języku programowania Java.
3. System nie będzie posiadał wsparcia technicznego, wszelka pomoc będzie dostępna w dokumentacji.
4. System będzie zoptymalizowany do stopnia zapewniającego relatywnie szybki czas reakcji przy stosunkowo niewielkim obciążeniu serwera.
5. System uruchomi się na serwerze obsługującym język programowania Java oraz system relacyjnej bazy danych.

# Diagram encji

Diagram Encji

