

Aplikacja do zarządzania listami zadań do wykonania

1. Nazwa i temat serwisu/aplikacji:

Aplikacja do zarządzania listami zadań do wykonania

2. Cel istnienia serwisu/aplikacji z punktu widzenia właściciela:

Zbieranie różnorodnej społeczności, sprzedaż dodatkowych funkcjonalności wewnątrz aplikacji, zbieranie statystyk dotyczących wykonywanych zadań w pracy różnych grup społecznych.

3. Główna grupa docelowa, dla której jest przeznaczony (cechy charakterystyczne):

- Główną grupą docelową są osoby potrzebujące aplikacji porządkujących harmonogram dnia w różnym wieku.
- Osoby, potrzebujące formy kalendarza z funkcjami notatnikowymi.
- Osoby, chcące poprawić swoją wydajność i zarządzanie czasem oraz otrzymywać statystyki z tym związane.

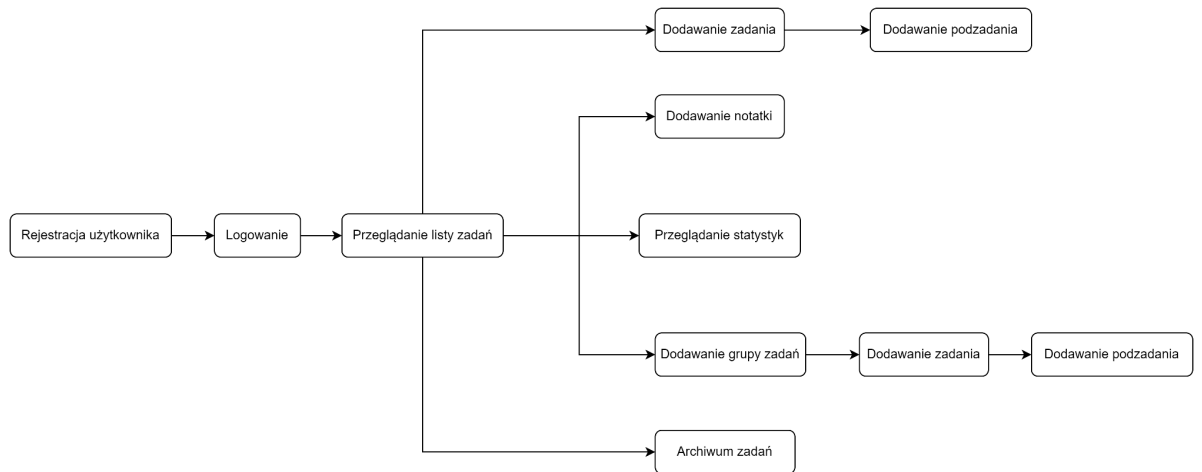
4. Przegląd rozwiązań konkurencyjnych oraz opis możliwości zdobycia przewagi konkurencyjnej:

- ToDolst: Serwis oferujący dodawanie zadań na konkretny dzień lub cyklicznych, wklejanie notatek do już istniejących zadań oraz dziennik aktywności.
- Microsoft To Do: Popularny serwis Microsoftu oferujący listy zadań, inteligentne planowanie dnia, udostępnianie list zadań między użytkownikami oraz integrację z serwisami Microsoft.
- Google Tasks Client: Aplikacja desktopowa i internetowa posiadająca funkcje związane z tworzeniem list zadań oraz pełną integrację z serwisami Google.

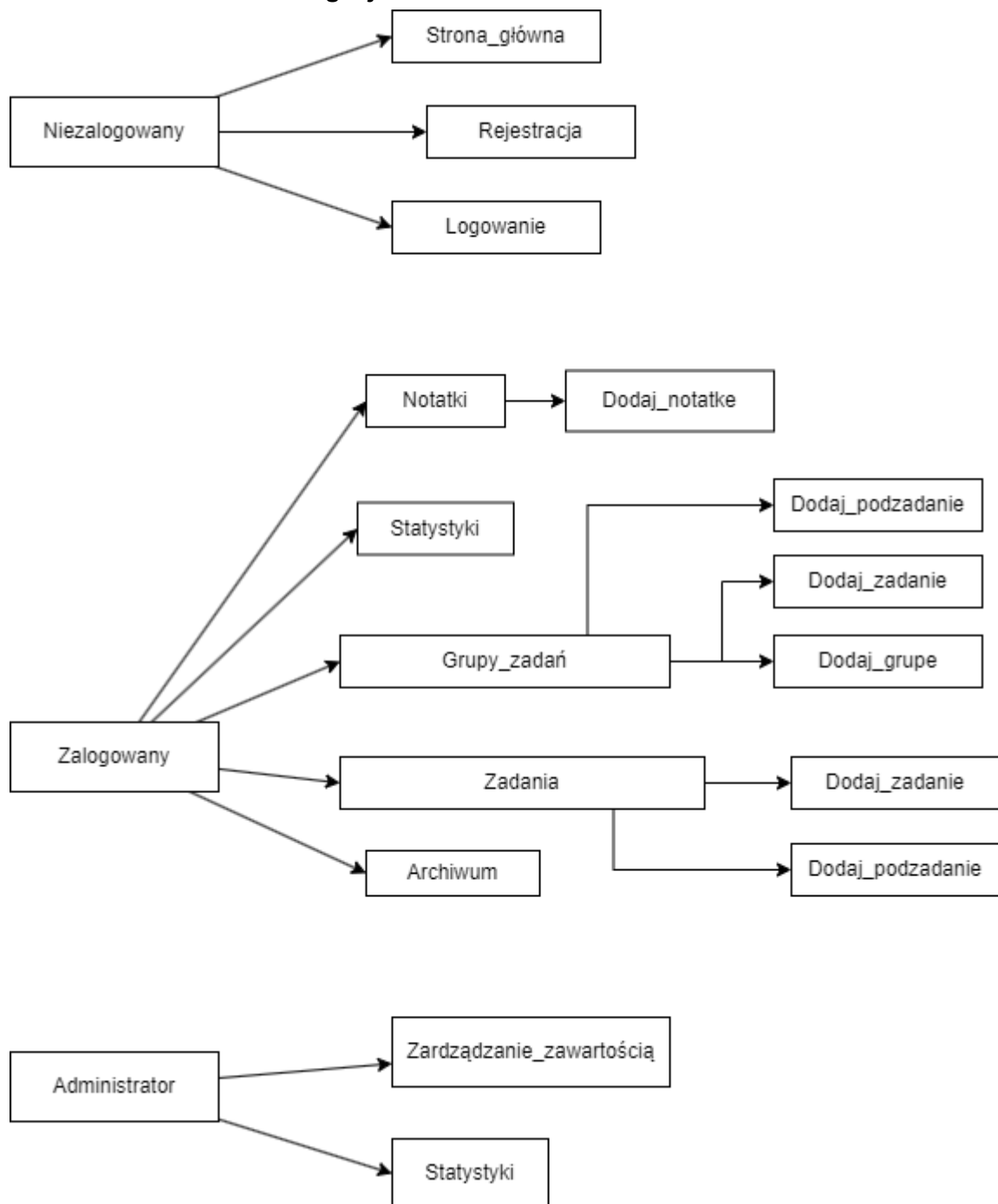
Przewagi konkurencyjne:

- Brak reklam.
- Łatwa możliwość dodawania notatek.
- Szybkie działanie serwisu, ze względu na brak zewnętrznych usług.
- Kontakt z developerem.
- Niski koszt utrzymania.

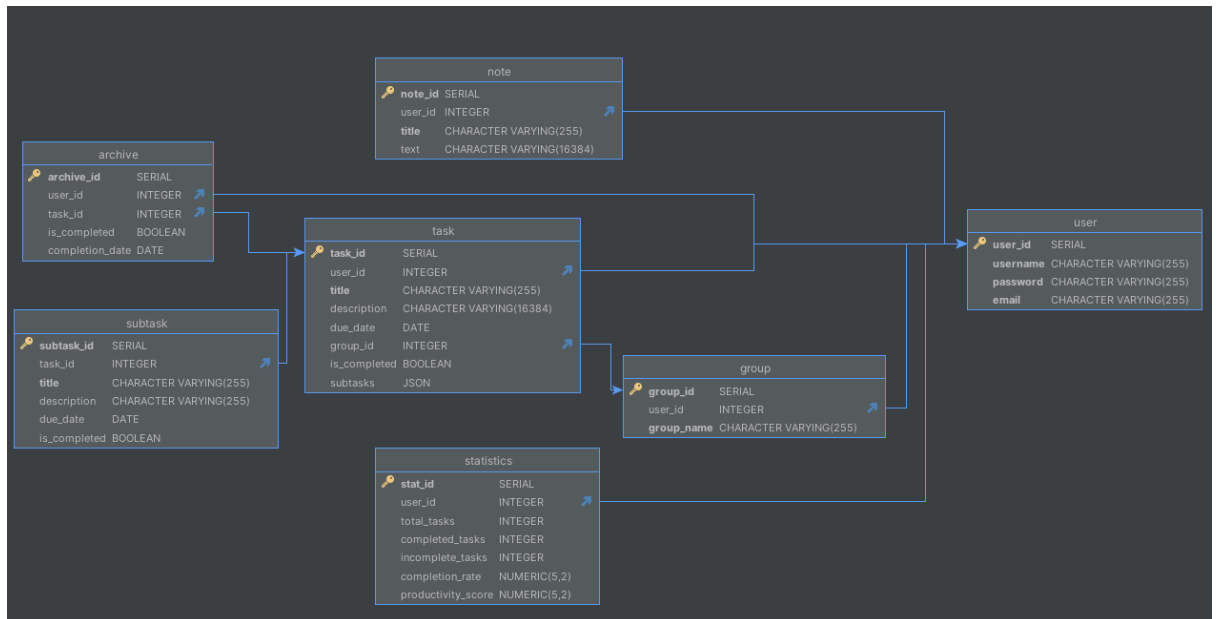
5. Diagram przypadków użycia UML:



6. Schemat nawigacji:



7. Model bazy danych (diagram klas UML):



8. Schematy graficzne stron:

- Widok główny z zadaniami do wykonania na najbliższe dni.
- Strona profilu z historią aktywności i statystykami
- Formularz dodawania zadania, notatki
- Panel administracyjny do zarządzania treścią.

9. Lista planowanych technologii:

Frontend: React.js

Backend: Spring Boot

Baza danych: PostgreSQL

Autentykacja: Spring Boot Security

Responsywny design: Custom

Komunikacja między serwerem a klientem: RESTful API