

Aplikacja webowa z bazą danych

Dokumentacja projektowa

Tomasz Chojnowski

Kacper Górnik

Mateusz Maksimowski

20.01.2023

Streszczenie

Poniższy dokument zawiera analizę projektu "Aplikacja webowa z bazą danych", realizowanego przez zespół w składzie: Tomasz Chojnowski, Kacper Górnik, Mateusz Maksimowski. Analiza zawiera opis celu biznesowego projektu, a także wymagania funkcjonalne i нефункционалне.

Spis treści

1. Opis biznesowy	4
2. Słownik	5
3. Harmonogram	6
4. Metodyka	7
5. Wymagania funkcjonalne	8
6. Wymagania нефunkcjonalne	9
7. Diagramy	10
8. Użytkowanie	16
9. Podsumowanie	17

1. Opis biznesowy

Celem aplikacji jest wyświetlenie prognozy pogody dla wybranego miasta, wpisanego w pole tekstowe przez zalogowanego użytkownika. Wyświetlenie danych pogodowych oraz mapy z zaznaczoną lokalizacją wpisanego miasta. System łączy się z bazą danych aby uwierzytelnić dane logowania.

2. Słownik

API - specyfikacja wytycznych, jak powinna przebiegać interakcja między komponentami programowymi (w jaki sposób programy lub podprogramy komunikują się ze sobą).

3. Harmonogram

- 08.11** - Wybór tematu oraz frameworku React do jego realizacji
- 10.11** - Wyszukanie pogodowego API, przestudiowanie struktury zwracanej informacji
- 13.11** - Dodanie API do Aplikacji
- 17.11** - Wyświetlanie prognozy pogody na bieżący dzień
- 22.11** - Wyświetlanie prognozy pogody na następne dni
- 27.11** - Dodanie CSS'a do podstrony z prognozą
- 04.12** - Połączenie się z bazą danych i proste wyświetlanie danych
- 20.12** – Wstawienie mapy wraz z pinezką wyszukiwanego miasta
- 01.01** – Dodanie opcji logowania oraz rejestracji
- 02.02** – Dodanie CSS'a do stron z logowaniem oraz rejestracją

4. Metodyka

Do stworzenia aplikacji wykorzystaliśmy środowisko React.js. Jest to javascriptowa biblioteka służąca do tworzenia interfejsów użytkownika. Aplikacja została stworzona przy edytorze kodu Visual Studio Code. Baza danych stworzona została w MySQL. Do pobrania pogody wykorzystujemy darmowe API: <https://openweathermap.org/>.

5. Wymagania funkcjonalne

- 5.1 Logowanie – do wyświetlenia prognozy wymagane jest zalogowanie do systemu
- 5.2 Rejestracja – umożliwienie skorzystania z aplikacji przez nowego użytkownika
- 5.3 Wyszukiwarka – możliwość wyszukania miasta
- 5.4 Baza danych zbierająca dane użytkownika
- 5.5 Wyświetlanie prognozy pogody
- 5.6 Pobieranie prognozy z API
- 5.7 Wyświetlanie mapy z pinezką wyszukanego miasta

6. Wymaganie нефункционалне

- 6.1 Wszystkie funkcjonalności powinny być dostępne dla użytkowników przeglądarek internetowych Chrome i Firefox.
- 6.2 Czas wypisania prognozy nie może przekraczać 3 sekund
- 6.3 Czas logowania nie może przekraczać 10 sekund
- 6.4 Dostępny na wszystkie systemy operacyjne

7. Diagramy

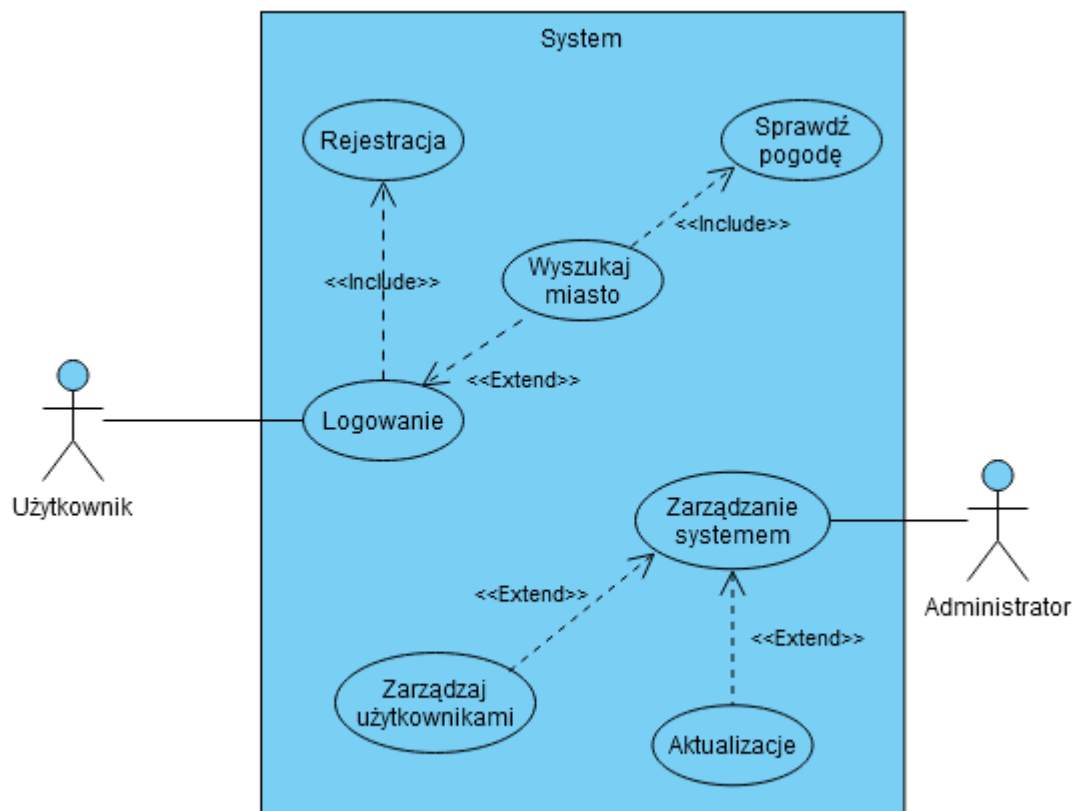


Diagram przypadków użycia

Użytkownik ma możliwość logowania, chyba że nie posiada konta to może je stworzyć. Użytkownik bez konta nie może skorzystać z aplikacji. Po zalogowaniu użytkownik może wpisać miasto i sprawdzić dla niego prognozę pogody.

Administrator wprowadza aktualizacje i zarządza kontami użytkowników.

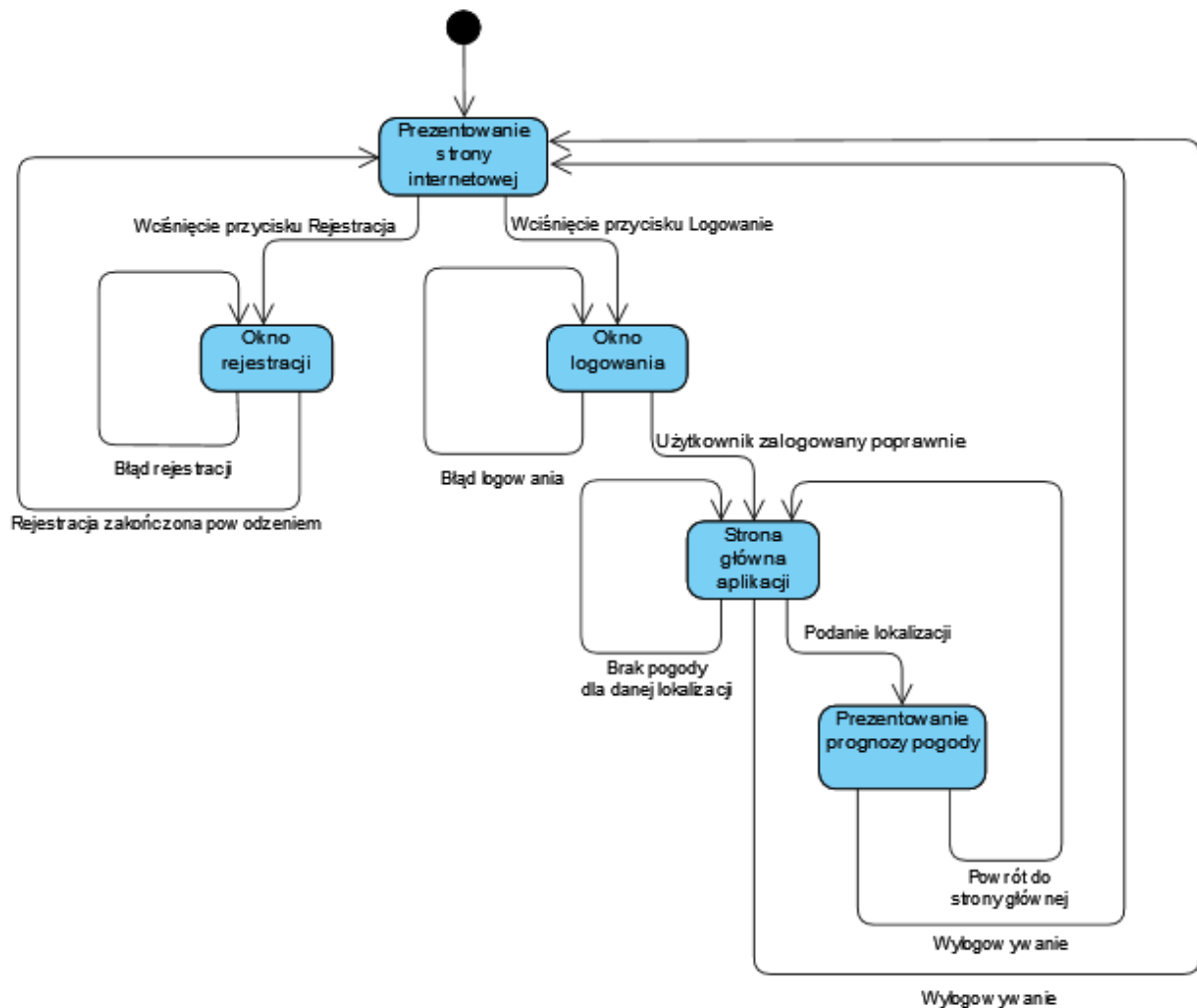


Diagram stanów

Po wejściu na stronę wyświetlana jest strona startowa. Użytkownik może założyć konto i po udanej rejestracji powraca na stronę główną. Użytkownik już posiadający konto przechodzi do logowania i po poprawnie wprowadzonych danych zostaje przekierowany do głównej strony aplikacji, gdzie ma możliwość sprawdzenia prognozy pogody. Po źle podanych danych logowania nie przechodzi dalej, a pozostaje nadal w logowaniu (podobnie jak w rejestracji). Gdy zostanie wprowadzona nie poprawna lokalizacja pogoda nie zostanie wyświetlona.

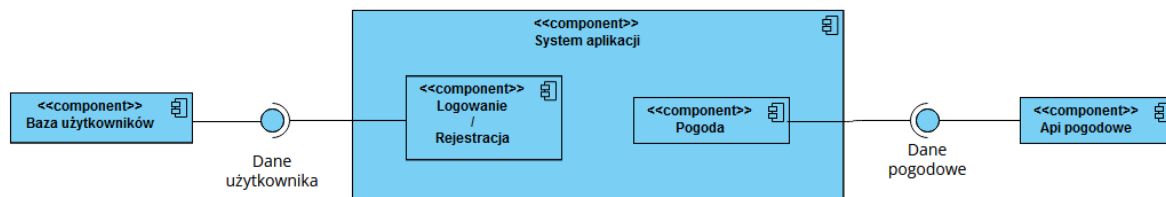


Diagram komponentów

Aplikacja składa się z elementów logowania/rejestracji i łączy się z bazą danych, przechowującą dane użytkowników. Pogoda pobiera dane z API pogodowego, które zostają wyświetlone w aplikacji internetowej.

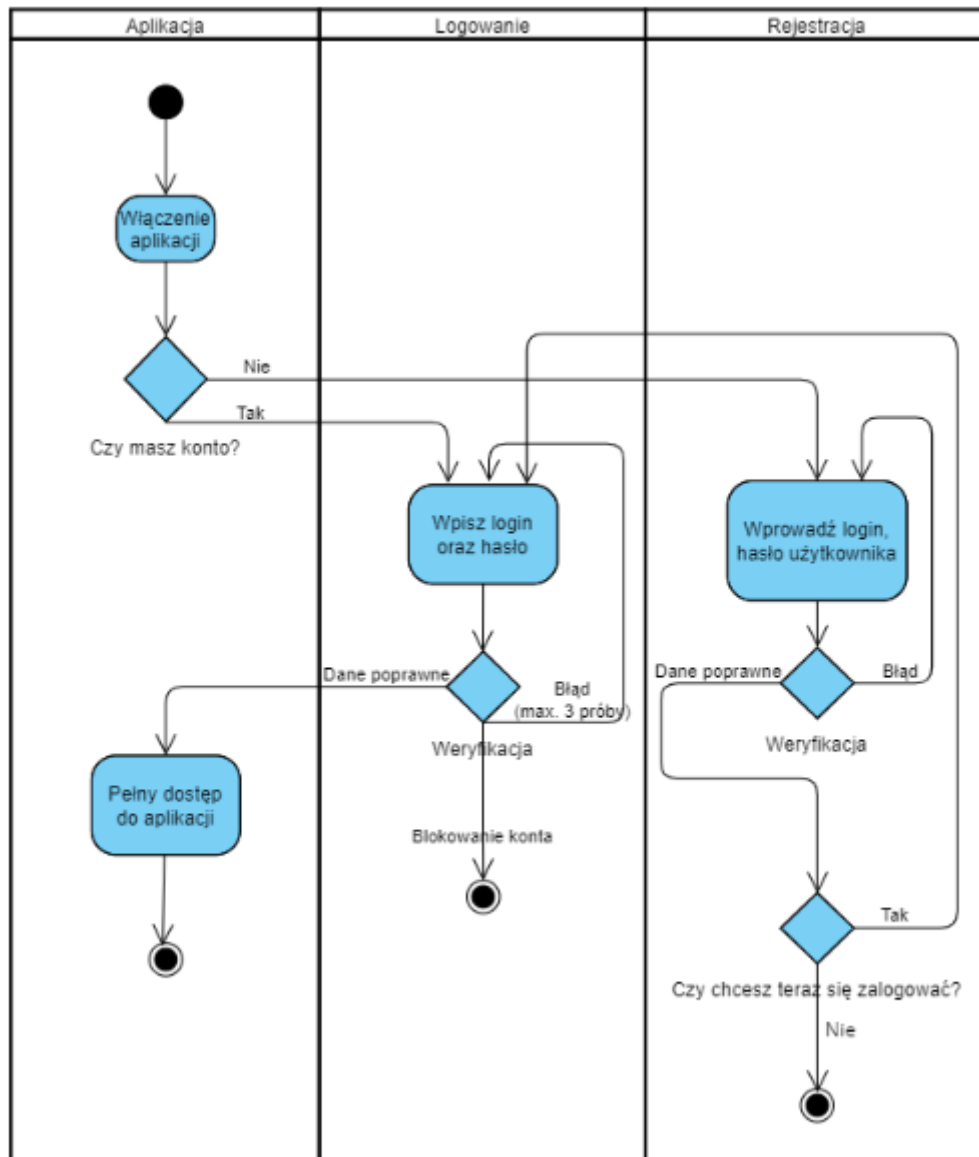


Diagram czynności

Po uruchomieniu strony użytkownik może się zalogować lub zarejestrować. Podczas rejestracji użytkownik podaje dane takie jak: login, hasło. Logując się użytkownik ma 3 próby zalogowania, po czym zostaje zablokowane jego konto. Użytkownik poprawnie zalogowany uzyskuje pełen dostęp do aplikacji, może sprawdzić prognozę pogody.

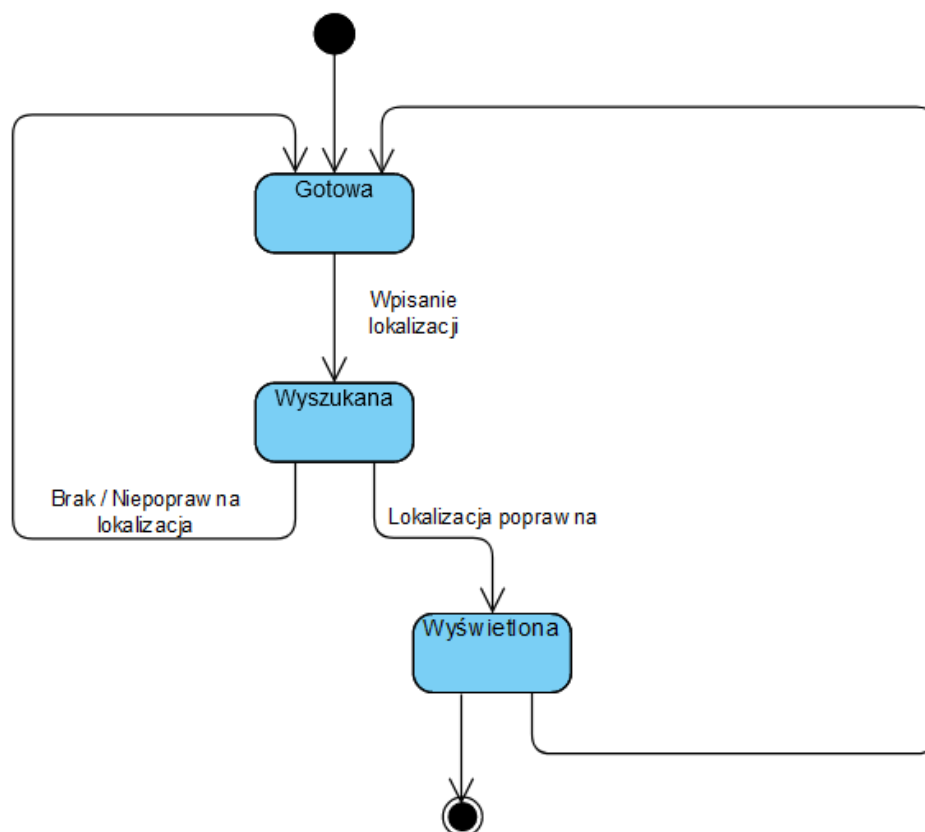


Diagram protokołowej maszyny stanowej

Aplikacja po uwierzytelnieniu użytkownika „czeka” na podanie lokalizacji, dla której będzie wyszukiwana prognoza pogody. Jeżeli lokalizacja jest poprawna i jest ona w API prognoza zostaje wyświetlona. W przypadku jej nie podania, lub braku w bazie pozostaje w trybie Gotowa.

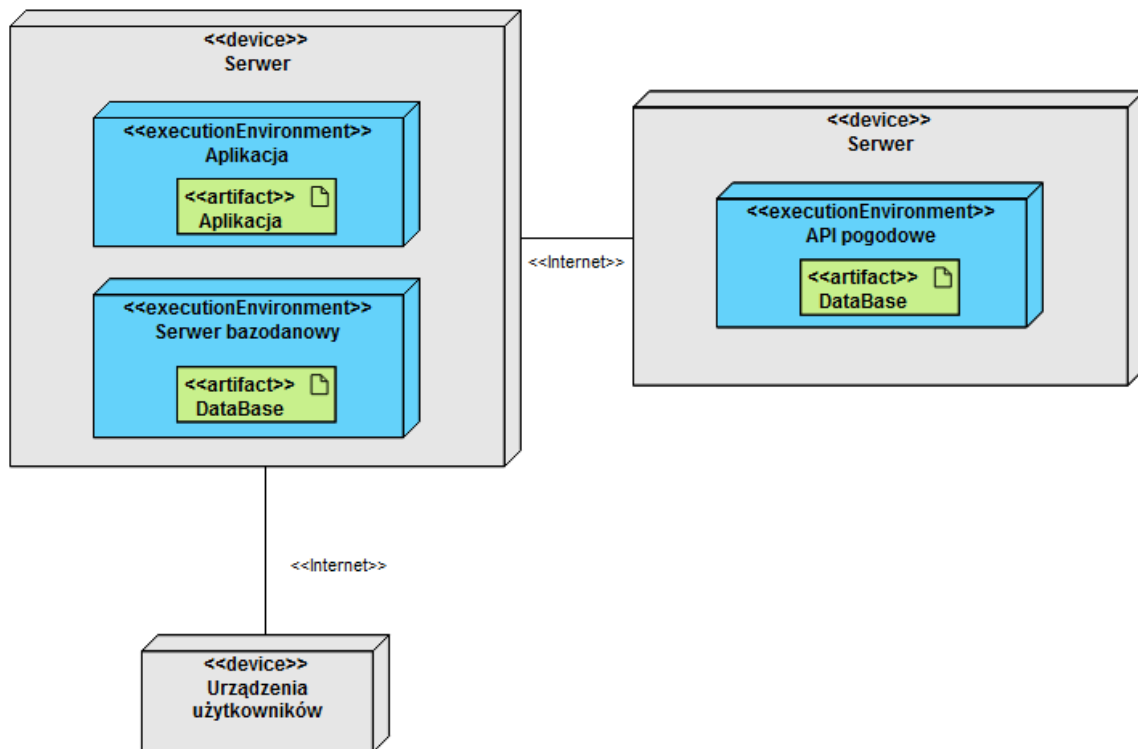


Diagram rozlokowania

Na serwerze jest aplikacja pogodowa i baza danych. Na odrębnym serwerze jest baza danych API pogodowego, z którego nasza aplikacja czerpie dane.

8. Użytkowanie

Aby uruchomić projekt za pomocą Visual Studio Code pobrać plik z dysku Google i go rozpakować. Projekt zawiera wszystkie potrzebne biblioteki do działania. Następnie wystarczy wpisać komendy *npm start* w folderze backend oraz app, a także posiadać bazę danych w MySQL.

Po wejściu na stronę aplikacji należy się zalogować, lub jeżeli jest się nowym użytkownikiem należy się zarejestrować klikając w link rejestracji. Podczas rejestracji jest wymagany login oraz hasło. Login – który nie jest w bazie jako, już użytkowy. Po stworzeniu konta można już się zalogować i przejść od razu do głównej strony aplikacji pogodowej. Należy podać lokalizację i wcisnąć enter, aby sprawdzić prognozę pogody. Jeżeli została podana prawidłowa wartość wyświetlane są dane pogodowe oraz mapa z zaznaczoną lokalizacją. Po zakończeniu korzystania z aplikacji wystarczy wylogować się.

9. Podsumowanie

W projekcie udało się stworzyć działającą aplikację wyświetlającą pogodę dla danej lokalizacji wraz z wyświetlaniem mapy z zaznaczoną lokalizacją. Wprowadzona została opcja logowania oraz rejestracji konta przez potencjalnych użytkowników. Baza danych zapisuje i przechowuje dane użytkowników. Aplikację można dalej rozwijać w stronę zaawansowanej prognozy pogody, o takie rzeczy jak poziom smogu, możliwe opady i wiele innych, na przykład wykorzystując przy tym innego rodzaju API.